

Organic Complex Systems

**A Comprehensive Theoretical Apparatus
for Modeling the Organization and Dynamics
of Living and Lifelike Systems**

Part II: Organodynamics
A Mathematical Realization of the OCS Theory

**Part II is a work-in-progress and not ready
for distribution.**

Revision 29

**Grant Holland
Santa Fe, NM
October 2011**

Contents

<u>OCS REVIEW AND SUMMARY</u>	<u>8</u>
<u>PREFACE</u>	<u>11</u>
A MATHEMATICAL IMPLEMENTATION OF THE OCS ORGANIZING PRINCIPLES	11
MASSIVELY COMPLEX DOMAINS AND COMPREHENSIVE MODELS	11
MATHEMATICS ACCESSIBLE TO NON-MATHEMATICIANS	12
THE ORGANODYNAMICS FRAMEWORK.....	12
<u>PREVIEW OF ORGANODYNAMICS</u>	<u>14</u>
THE ISSUE ADDRESSED	14
COMPREHENSIVELY MODELING THE SEVEN OCS ORGANIZING PRINCIPLES	15
ORGANODYNAMIC WEBS: THE GRAND SCHEME OF ORGANODYNAMICS.....	16
APPROXIMATION LEVELS IN ORGANODYNAMIC WEBS.....	24
UNCERTAINTY AND INFORMATION.....	31
UNCERTAINTY AND THE SIX APPROXIMATIONS.....	32
INCREMENTAL EXAMPLE MODEL DEVELOPMENT	33
<u>SYSTEM STATE AND ORGANIZATION.....</u>	<u>34</u>
THE ISSUE ADDRESSED	34
SYSTEMS	35
SYSTEM POPULATION	36
SYSTEM ORGANIZATION	37
SIMPLE NON-BIOCHEMICAL SYSTEM EXAMPLES	43
SIMPLE BIOCHEMICAL SYSTEM EXAMPLE	52
FORWARD-LOOKING NOTE ON SYSTEM DYNAMICS.....	56
ADVANCED SYSTEM TOPICS.....	57
SOME LIMITATIONS	59
EXTENDING THE SYSTEM ORGANIZATION CONCEPT	60
RELATIVE DEGREES OF SYSTEM ORGANIZATION.....	63
CONCLUSIONS.....	64
INCREMENTAL EXAMPLE MODEL DEVELOPMENT	64
PROGRESS CHECK.....	64
<u>SYSTEM DYNAMICS AS REORGANIZATION.....</u>	<u>66</u>
THE ISSUE ADDRESSED	66
SYSTEM PROCESSES AND TRANSFORMS.....	67
PRIMITIVE SYSTEM TRANSFORMS.....	73
A STARTER SET OF PRIMITIVE SYSTEM TRANSFORM CATEGORIES	74

THE SYSTEM TRANSFORM CATEGORIES	79
SYSTEM PROCESS WEBS.....	82
SIMPLE NON-BIOLOGICAL SYSTEM EXAMPLES	82
INCREMENTAL EXAMPLE MODEL DEVELOPMENT	87
PROGRESS CHECK.....	87
<u>SYSTEM DYNAMICS AS UNCERTAINTY.....</u>	<u>89</u>
THE ISSUE ADDRESSED.....	89
STOCHASTIC MODELS AND UNCERTAINTY.....	92
SYSTEM DYNAMICS AS UNCERTAIN REORGANIZATION	101
INTRODUCING STOCHASTIC PROCESSES IN ORGANODYNAMICS	104
SIMPLE NON-BIOLOGICAL SYSTEM EXAMPLE OF AN ORGANODYNAMIC PROCESS	111
INTRODUCING CONDITIONAL STOCHASTIC PROCESSES.....	114
INTRODUCING MARKOV CHAINS.....	118
INCREMENTAL EXAMPLE MODEL DEVELOPMENT	120
PROGRESS CHECK.....	121
<u>THE CANONICAL ORGANODYNAMICS MODEL.....</u>	<u>124</u>
THE ISSUE ADDRESSED.....	124
THE BIASES OF THE ORGANODYNAMICS FRAMEWORK.....	124
INTRODUCING A CANONICAL BIOLOGICAL EXAMPLE.....	126
INCREMENTAL EXAMPLE MODEL DEVELOPMENT	132
PROGRESS CHECK.....	132
<u>THE UNCERTAINTY MODEL GRADIENT</u>	<u>133</u>
THE ISSUE ADDRESSED.....	133
THREE MODEL TYPES OF DECREASING UNCERTAINTY	135
THE UNCERTAINTY MODEL GRADIENT	136
LEVELS OF APPROXIMATION AND THE UNCERTAINTY MODEL GRADIENT	137
ELEMENTS OF THE THREE UNCERTAINTY GRADIENTS MODELS	138
EXAMPLE MARKOV AND CONDITIONAL STOCHASTIC MODELS	141
PROMOTING A MARKOV MODEL TO AN UNCONDITIONAL MODEL	150
MODEL DEVELOPMENT: BUILDING THE FIRST TWO APPROXIMATIONS	155
MODEL DEVELOPMENT: BUILDING THE FIRST APPROXIMATION	155
MODEL DEVELOPMENT: BUILDING THE SECOND APPROXIMATION	167
UNCERTAINTY VERSUS ORGANIZATION IN ORGANODYNAMICS	171
INCREMENTAL EXAMPLE MODEL DEVELOPMENT	174
PROGRESS CHECK.....	174
<u>AUTOCOGENERATION: HOW ORGANIC SYSTEMS REORGANIZE</u>	<u>176</u>
THE ISSUE ADDRESSED.....	176
FOUNDATIONS OF AUTOCOGENERATION	177
GENERAL DEFINITION OF AUTOCOGENERATION.....	178

AUTOCOGENERATION AS REGULARITY AND IRREGULARITY	179
EXAMPLES OF AUTOCOGENERATION IN SCIENCE, ENGINEERING AND MATHEMATICS	180
MECHANISMS OF AUTOCOGENERATION IN BIOLOGICAL SYSTEMS	186
THE MECHANISMS THAT IMPLEMENT REGULARITY IN ORGANODYNAMICS	191
STRATEGY FOR MODELING AUTOCOGENERATION USING MARKOV CHAINS.....	193
INCREMENTAL EXAMPLE MODEL DEVELOPMENT	200
PROGRESS CHECK.....	200

MODELING AUTOCOGENERATION USING MARKOV CHAINS.....202

THE ISSUE ADDRESSED	202
THE ROLE OF MARKOV CHAINS IN ORGANODYNAMICS	204
CLASSIFICATION OF MARKOV CHAINS AND THEIR STATES.....	204
BEHAVIORAL DYNAMICS OF HOMOGENEOUS MARKOV CHAINS.....	209
REGULARITY, IRREGULARITY AND AUTOCOGENERATION IN MARKOV CHAINS	212
STATIONARY DISTRIBUTIONS AND REGULARITY	213
SPECIFIC TYPES OF REGULARITY.....	219
COMPLEX TOPICS IN REGULARITY	222
INCREMENTAL EXAMPLE MODEL DEVELOPMENT	227
PROGRESS CHECK.....	227

ORGANODYNAMIC GRAPHS.....229

THE ISSUE ADDRESSED	229
SYSTEM TRANSFORMS AS NODES THAT CONNECTED SYSTEM PROCESSES.....	233
CONTEMPLATING NODES TO CONNECT STOCHASTIC PROCESSES.....	234
GENERAL DEFINITIONS OF ORGANODYNAMIC TRANSFORMS	235
JOIN AND SPLIT TRANSFORMS ON PIECEWISE-HOMOGENEOUS MARKOV CHAINS	236
TYPES OF SPLITS AND JOINS IN ORGANODYNAMICS.....	239
IMPLEMENTING SPLITS AND JOINS AS ORGANODYNAMIC TRANSFORMS	241
THE NECESSARY NONHOMOGENEITY OF AN ORGANODYNAMIC TRANSFORM	244
PATTERNS IN DEFINING THE ORGANODYNAMIC TRANSFORMS.....	245
ORGANODYNAMIC TRANSFORM CATEGORIES	248
ORGANODYNAMIC GRAPHS.....	250
TOPOLOGY OF AN ORGANODYNAMIC GRAPH	251
SIMPLE NON-BIOLOGICAL SYSTEM EXAMPLES OF ORGANODYNAMIC PROCESSES	252
INCREMENTAL EXAMPLE MODEL DEVELOPMENT	253
PROGRESS CHECK.....	253

SIMPLEX ORGANODYNAMIC WEBS.....255

THE ISSUE ADDRESSED	255
ORGANODYNAMIC NETWORKS	257
ROUTINE	257
SIMPLEX ORGANODYNAMIC WEBS	261
ORGANODYNAMIC WEBS AND THE SEVEN OCS ORGANIZING PRINCIPLES.....	263
BUILDING MODELS WITH ORGANODYNAMIC WEBS.....	264
INCREMENTAL EXAMPLE MODEL DEVELOPMENT	265

PROGRESS CHECK.....	265
<u>NESTED SYSTEMS</u>	<u>267</u>
THE ISSUE ADDRESSED	267
SYSTEM NESTING CONSIDERATIONS	270
EXAMINING THE LIMITATIONS OF THE SIMPLEX SYSTEM NOMENCLATURE	271
A COMPOSITE REPRESENTATION FOR OUR CANONICAL BIOLOGICAL SYSTEM	272
MORE THAN MERELY A PANDORA’S BOX.....	277
COMPOSITE AND COMPONENT SYSTEMS	278
COMPOSITE SYSTEMS	278
COMPONENT SYSTEM REPRESENTATION.....	278
DEEPLY-NESTED COMPOSITE SYSTEMS	282
ALGORITHM FOR CONSTRUCTING A COMPOSITE SYSTEM	282
COMPOSITE SYSTEMS REPRESENTATION IN THE UNCERTAINTY MODEL GRADIENT.....	283
THE INHERENT COMPLEXITY OF MODELING NESTED SYSTEMS.....	285
INCREMENTAL EXAMPLE MODEL DEVELOPMENT	286
PROGRESS CHECK.....	286
<u>CONDITIONAL DEPENDENCY RELATIONSHIPS IN COMPOSITE SYSTEMS.....</u>	<u>288</u>
THE ISSUE ADDRESSED	288
ORGANIZATIONAL DEPENDENCIES IN NESTED SYSTEMS	291
EXAMPLES OF DEPENDENCY CONDITIONS IN NESTED SYSTEMS	292
CONSTRAINTS AND LIBERTIES ON ORGANIZATIONS IN COMPOSITE SYSTEMS	300
STOCHASTIC DEPENDENCIES AND COMPOSITE SYSTEM ORGANIZATION.....	301
THE SEMANTICS OF COMPONENT RELATIONSHIPS	302
THE MECHANISMS OF SYSTEM INTEGRATION	304
DEFINING DOMAIN-DEFINED RELATIONSHIPS.....	308
DEFINING PROPAGATED COMPONENT RELATIONSHIPS.....	308
DEFINING ACCESSIBILITY-ENHANCED COMPONENT RELATIONSHIPS	309
FURTHER COMPONENT RELATIONSHIPS CHARACTERIZATIONS	314
COMPONENT RELATIONSHIPS IN THE UNCERTAINTY MODEL GRADIENT.....	316
INCREMENTAL EXAMPLE MODEL DEVELOPMENT	317
PROGRESS CHECK.....	317
<u>COMPOSITE ORGANODYNAMIC WEBS</u>	<u>319</u>
THE ISSUE ADDRESSED	319
COMPOSITE SYSTEMS IN ORGANODYNAMIC WEBS	320
COMPOSITE ORGANODYNAMIC WEBS	324
THE UNCERTAINTY MODEL GRADIENT FOR A NESTED SYSTEM STEP.....	325
INCREMENTAL EXAMPLE MODEL DEVELOPMENT	326
PROGRESS CHECK.....	326
<u>JOINT COMPOSITE ORGANODYNAMIC WEBS</u>	<u>328</u>

THE ISSUE ADDRESSED	328
ANALYSIS OF THE ISSUE	332
STRATEGY	334
THE RESOLUTION	335
ASSEMBLING THE UNCERTAINTY MODEL GRADIENT	339
INCREMENTAL EXAMPLE MODEL DEVELOPMENT	340
PROGRESS CHECK	340

MEASURING UNCERTAINTY IN ORGANODYNAMIC WEBS.....342

THE ISSUE ADDRESSED	342
UNCERTAINTY AND INFORMATION	343
SHANNON ENTROPY AND THE UNCERTAINTY MODEL GRADIENT	350
UNCERTAINTY IN ORGANODYNAMIC BASIC STRUCTURES	353
UNCERTAINTY IN ORGANODYNAMIC APPROXIMATION LEVELS	355
UNCERTAINTY, ORGANIZATION AND SHANNON ENTROPY	359
INCREMENTAL EXAMPLE MODEL DEVELOPMENT	362
PROGRESS CHECK	362

ENTROPY FOUNDATIONS AND CONSTRAINTS IN STOCHASTIC PROCESSES ...364

THE ISSUE ADDRESSED	364
THE INFORMATION RANDOM VARIABLE	364
THE APPLICABILITY OF UNCERTAINTY AND INFORMATION TO ORGANIC SYSTEMS	372
INCREMENTAL EXAMPLE MODEL DEVELOPMENT	377
PROGRESS CHECK	377

EMERGENCE.....379

THE ISSUE ADDRESSED	379
SYSTEMIC VERSUS PROCESS EMERGENCE	379
THE PRACTICAL INEVITABILITY OF SYSTEMIC EMERGENCE	380
OPPORTUNITIES FOR AND LIKELIHOOD OF SYSTEMIC EMERGENCE	384
COUNTER-EMERGENCE	384
PROCESS EMERGENCE	385
INCREMENTAL EXAMPLE MODEL DEVELOPMENT	385
PROGRESS CHECK	386

PERSISTENCE.....387

THE ISSUE ADDRESSED	387
THE PERSISTENCE OF ORGANIZATION	388
CONSPICUOUS MECHANISMS OF PERSISTENCE IN BIOLOGICAL SYSTEMS	389
OTHER ORGANIZING PRINCIPLES SUPPORT OF PERSISTENCE	391
DIRECTED GRAPH OF OCS ORGANIZING PRINCIPLES	393
THE ORGANIZING PRINCIPLES AS A SYSTEM	400
THE ORGANIZING PRINCIPLES AS A MARKOV MODEL	400

CONCLUSIONS.....	407
INCREMENTAL EXAMPLE MODEL DEVELOPMENT	408
PROGRESS CHECK.....	408
<u>REVIEW OF ORGANODYNAMICS.....</u>	<u>410</u>
THE BEINGS OF ORGANODYNAMICS.....	411
THE DOINGS OF ORGANODYNAMICS	415
THE KNOWINGS OF ORGANODYNAMICS	424
COMPLEXITY	426
SUMMARY OF ORGANODYNAMICS STRUCTURES AND METHODOLOGY	429
CONCLUSIONS.....	429
<u>APPENDIX 1: SYSTEM TRANSFORMS.....</u>	<u>430</u>
INTRODUCTION TO SYSTEM TRANSFORMS	430
SYSTEM TRANSFORM CATEGORIES.....	431
<u>APPENDIX 2: ORGANODYNAMIC TRANSFORMS</u>	<u>447</u>
INTRODUCTION TO ORGANODYNAMIC TRANSFORMS.....	447
ORGANODYNAMIC TRANSFORM CATEGORIES	449
<u>APPENDIX 3: THE ORGANODYNAMICS MODELING STRUCTURES</u>	<u>463</u>
<u>APPENDIX 4: THE ORGANODYNAMICS MODELING METHODOLOGY</u>	<u>468</u>

OCS Review and Summary

Organic Complex Systems is a theoretical framework for modeling systems that are “organic” - that exhibit lifelike properties.

Such systems must, of course, include biological life on earth. But OCS seeks to go beyond – to represent possible “lifelike” domains that may exist outside of carbon chemistry; indeed, outside of chemistry. OCS seeks to develop a formal abstraction of the living.

As a discipline, OCS is at the intersection of systemics, dynamical systems theory, complexity science, information theory, cybernetics and systems biology. It calls upon a number of other scientific disciplines and mathematical research. In the process, it presents a systemic argument for the properties that it identifies as *organizing principles*; and then it develops and presents a mathematical modeling framework.

Such a framework could see a number of applications. Some of these are: non-carbon biochemical protolife on other planets; the engineering of artificial lifelike systems; complex adaptive systems in compute space and cyber space; information and memetics; and the modeling of various contending origins-of-life theories including “genetics first”, “metabolism first”, RNA worlds, the thioester world, the iron-sulfur world, clay life, heterotrophic versus autotrophic origins, and others.

OCS intends to present a platform for addressing such questions as: Can the stock market be considered in any sense to be lifelike? Can an economy? What about a tornado or a game of Go, an autocatalytic set, or the planet Earth?

The general intention of OCS is to empirically observe biological systems, to abstract a set of organizing principles from these observations, and finally to generalize a theory from these organizing principles that is applicable to any systems, biological or not, that exhibit these principles.

Part I of this treatise develops these seven OCS organizing principles. We shall summarize them here by reprinting the following table:

Principles of Organization of Organic Systems

Systemic Property	Organizing Principle
Organized	<i>Any organic entity is comprised of 1) a population of two or more components, as well as 2) a set of relationships among those components. This set of relationships specifies the organization of the entity. Thus, an organic entity constitutes a system; and shall be referred to as an organic system.</i>
Emergent	<i>Any organic system exhibits one or more properties that none of its components exhibits.</i>

<i>Nested</i>	<i>Any organic system has at least one component that is another system.</i>
<i>Reorganizational</i>	<i>Organic systems are dynamic – constantly undergoing change. This change involves some manner of reorganization of its components, or their components. The result is, generally, some form of buildup or breakdown – composition, decomposition, de-nesting, re-nesting, or any combination thereof.</i>
<i>Autocogenerative</i>	<i>The generative agent that initiates the reorganization of an organic system is the system itself. This reorganization is accomplished by virtue of the components of the system co-generating and co-organizing each other. Since the components co-generate and co-organize the other components, then the result is that they co-generate and co-organize the system. An organic system is co-generated by its components-in-relationship, and thus by itself. This phenomenology obviates the philosophical problem of a thing generating itself prior to its own existence. Consequently, the nature of this systemic reorganization, or dynamism, is that it is autonomous. All organic systems decay, or decompose, autonomously. But they also autonomously recombine in varieties of ways. These autonomous regenerations ramify into many flows that branch and loop. Some changes promote the propensity of the system to remain the same with respect to certain properties (regulation); while other changes cease such promotion (deregulation). Some changes promote the continued persistence of the system (adaptation), while some retard it (extinction).</i>
<i>Uncertain</i>	<i>All living processes depend upon both chance events as well as deterministic ones. Living processes operate along a spectrum between complete randomness and complete determinism, uncertainty and certainty. Where along that spectrum an organic system lies at any time is uncertain. Thus, the degree of predictability or uncertainty of an organic system is itself uncertain. (“Life is unpredictably unpredictable.”)</i>
<i>Persistent</i>	<i>The nature of the autocogeneration and uncertainty of an organic system is that it engenders the ongoing existence of these systems. Taken together, all of these static and dynamic elements have resulted in, at the least, a limited persistence of life itself. While no organic system persists, the existence of organic systems has.</i>

A second intention of OCS is to develop one or more mathematical frameworks that are founded on the seven OCS organizing principles. Such a framework

would, then, represent a general mathematical system that is suitable for modeling lifelike systems.

Such a mathematical framework, named *Organodynamics*, is developed and described below in the present Part II of this treatise.

Preface

Part I of this text presents the basic OCS theory in terms of a collection of organizing principles that characterize seven systemic properties that a broad class of living and lifelike systems should be expected to exhibit. These seven principles are put forth in this text as distinctions that characterize systems that exhibit them as worthy of being deemed *organic*.

The approach of OCS is to identify these essential systemic properties, and then to generalize them into a class of *organic, or "lifelike"* systems. In other words, OCS intends to be a generalized theory of the "living".

A Mathematical Implementation of The OCS Organizing Principles

This Part II presents one possible implementation of the basic OCS theory and its organizing principles, into a mathematical framework – named *Organodynamics* - that is intended to provide a suitable apparatus for developing models of this class of lifelike systems. Thus, Organodynamics is a development framework for models of organic systems.

The OCS framework is an integrated family of mathematical structures that are intended for modeling systems that comply with the seven organizing principles. The framework should be useful for the mathematical modeling of such systems as: biological systems, artificially engineered organic systems (Artificial Life) or lifelike systems in non-biochemical domains.

Massively Complex Domains and Comprehensive Models

The systems that Organodynamics intends to provide models of are dynamical, organic and massively complex – even bewilderingly so. In addition, Organodynamics intends to enable the construction of comprehensive models of these systems – thus further increasing the levels of complexity of the resulting models.

These levels of complexity are necessary if one desires to exhibit the systemic properties of these systems that render them as lifelike. They are also necessary if one desires such a model to provide a plan or prototype for the development of such systems.

Traditionally, modeling has been an exercise to produce vastly simplified likenesses of object systems - and not ones that are comprehensive and holistic.

However, the level of detailed fidelity and comprehensive complexity that is attempted by the Organodynamics framework in order to model organic systems with high fidelity represents a departure from traditional approaches. So, the reader is advised to expect a unique treatment in this regard and to expect the support for levels of complexity not often seen in model development systems.

This situation demands computer support. Thus, a library of software routines and modeling tools is planned for this Organodynamics framework.

Mathematics Accessible to Non-mathematicians

Departing from the promise of Part I, Part II will present a mathematical treatment of this framework. However, concepts and relationships will be motivated and explained so that, hopefully, non-mathematicians who have digested Part I will be able to follow along and not lose any significant points. The number of equations is kept to an absolute minimum. Instead, all mathematical concepts and constructs are explained in prose narrative and copiously exemplified by both biological and logical examples.

This Part II relies on a number of specific mathematical structures and theories, including, basic set theory, basic concepts of probability distributions, information theory, and basics of the theory of conditional stochastic processes – in particular the theory of Markov chains. However, even these subjects – although covered with considerable mathematical thoroughness - are introduced in a manner that should make the reading accessible to non-mathematicians.

Thus, these disciplines *are not* prerequisites. They will be introduced and explained in the text to the level required for the understanding of the Organodynamics theory and practice presented.

Let us now proceed to introduce *Organodynamics*.

The Organodynamics Framework

It should be obvious that a number of mathematical frameworks, all of which are faithful to the seven organizing principles as presented, are possible. One such framework has been developed at this time, *Organodynamics*, which is the subject of the present Part II of this text. The development of additional mathematical frameworks that also comply with the OCS organizing principles is left as further research.

Organodynamics is a dynamical system theory. It attempts to articulate a general theory of organic systems. It takes as its initial inspiration the architecture and dynamics of biological life on earth. It then abstracts from biology a set of systemic properties that it then uses as the hypothetical bases for a more general theory.

Thus, as an investigative exercise, Organodynamics studies the static and dynamic aspects of biological life and describes it as a process that originated as a population of organized entities that, through their encounters (biomolecules under the persuasion of Brownian motion and other chance phenomena) in a “primordial soup”, somehow managed over the eons to gradually, if non-monotonically, but spontaneously, emerge into a population of often-near-deterministic, highly-organized complex of interdependent processes.

At the very least, OCS interprets that living systems evolved in an environment involving both chance and determinism to states of high levels of organization and reorganization and to exhibit a spectrum of degrees of uncertainty/certainty that range between complete randomness and complete determinism. Organodynamics is constructed to be a mathematical framework that can model these aspects.

In the following chapters, we shall introduce Organodynamics by incrementally developing the elements of its machinery according to the seven organizing principles of Part I. Specifically, Organodynamics is a complex of mathematical elements, or apparatus, each of which is designed to contribute one or more of the seven OCS organizing principles.

For expository reasons, we shall initially roll out these mathematical elements one at a time, describe them and explain the OCS operating principle(s) that they support. The order that these organizing principles show up in this presentation is not necessarily the same as presented in Part I.

Of course, a regaling of the individual organizing principles does not reveal their total systemic impact, since they collectively cavort to reinforce each other in sometimes-unexpected ways. So, after the initial exposition, we shall have to circle back around to subsequently consider some integrated effects of their collaboration.

The constructs presented in this text have already been motivated by a number of examples in Part I, which shall not be repeated here. I trust this will allow the following expositions to proceed with certain efficiency, without loss of understanding.

Preview of Organodynamics

The Issue Addressed

Since Organodynamics is a framework for building models of organic complex systems, then the models that it generates are necessarily complex – and so is the framework itself.

Because of this, the framework will be developed in this text in a series of chapters. In order to ease the understanding of these chapters, the present chapter is provided as a *preview*.

In addition, at the completion of the presentation of the several framework content chapters, a *review* chapter will also be presented. Its organization will reflect the present preview chapter. But the review chapter will fill out the branches of the limbs of the present preview chapter with a recounting of some of the significant content of the intervening framework development chapters.

Biological Example of the Issue

All biological systems are composed of cells. Eukaryotic cells are composed of components called *organelles* as well as macromolecules and other smaller molecules. The organelles are, in turn, composed of macromolecules and other molecules. Each of these macromolecules can be, and often are, during the life of the cell chemically decomposed further into their components – and are built up from component molecular building blocks.

Prokaryotic cells are simpler. They do not have organelles. However, macromolecules and smaller molecules also constitute them. The molecules and many of the smaller molecules can be understood as constituted by smaller components.

All of this structure represents a high level of organization. And, this organization changes often, in both unpredictable and near-predictable degrees of organization. In fact, the degree and manner of organization in living systems is one of its most characteristic properties.

This organization and change of organization is self-managed by the cell so as to obviate the cessation of its life processes and to promote its persistence. This self-management is carried out by this very organization and change of organization that constitutes these life processes. At the same time, these cells are not totally self-contained, because they require a continued supply of external resources in order to persist.

Not only does this self-management occur within cells, but also in higher level organisms, such as multicellular organisms, but also in super-organisms, such as in ecological communities.

Specific Challenges

Developing a mathematical system that is capable of modeling this level of complexity, or modeling these varying degrees of organization and disorganization, of representing these varying degrees of uncertainty and certainty, and that is capable of developing comprehensive models of whole living organisms will be especially challenging.

The Organodynamic Approach to Modeling the Issue

Leverage dynamical systems theories, significant elements from probability theory, information theory and the theory of stochastic processes to fashion a highly flexible and customizable framework that can serve as a template for developing dynamical system models of biological systems as well as non-biological systems that exhibit the seven organizing principles of OCS.

Comprehensively Modeling the Seven OCS Organizing Principles

Organodynamics is a complex mathematical theory and modeling framework. In order to describe the theory, it will be necessary to present an elaborate and detailed exposition of its elements. This will be done according to the seven OCS organizing principles, since the mathematical structures of this framework have been contrived so as to embody and express these principles.

Admittedly, the ordering of this presentation will differ from the order in which the organizing principles were presented in Part I. The reason for this is the nature and complexity of the various mathematical constructs that have been contrived to comprise the theory, and how they fit together and interplay.

It is simpler to introduce them in a particular sequence and to gradually build up the edifice that is Organodynamics. It turns out that each of the mathematical structures involved in the Organodynamics framework can be reasonably presented as primarily, though not exclusively, related to one of the organizing principles.

The order in which the seven organizing principles first make their appearance is determined by how the mathematical structures that comprise Organodynamics are constructed. This order is:

1. Organizing Principle #1: Organization
2. Organizing Principle #4: Reorganization
3. Organizing Principle #6: Uncertainty
4. Organizing Principle #5: Autocogeneration
5. Organizing Principle #2: Nestedness
6. Organizing Principle #3: Emergence
7. Organizing Principle #7: Persistence

In addition, some of these organizing principles require several chapters; and because of the way that the mathematics interweaves, some of chapters describing the mathematics related to some of these organizing principles are interspersed.

Organodynamic Webs: The Grand Scheme of Organodynamics

The intention of Organodynamics is to provide a framework for building models of living and lifelike (organic) systems – specifically systems that exhibit the seven OCS operating principles. The models that are constructed using this framework shall be fashioned from mathematical elements of the framework. Any system for which an Organodynamic model is being constructed will be referred to as, variously, the *object system*, or the *target system*.

Like a physical modeling system, Organodynamics will provide a toolkit of foundational elements that can be judiciously composed by modelers into model instances of their own design. Because of its complexity, this text will develop the Organodynamics framework incrementally.

Being composed of mathematical elements, the Organodynamic framework is abstract. Nevertheless, as a framework, it has many things in common with various construction systems whose purpose is to enable builders to build physical entities (e.g. carpentry, plumbing, mechanical engineering and civil engineering).

Introducing a Construction Kit Analogy

As “development kits” go, we shall find that Organodynamics is particularly complex. This would normally be considered to be unfortunate. However, given the difficulty of the task that this framework has taken on – to enable the construction of models of living and lifelike systems, we can see that it must be so.

Hopefully, the use of the framework can be made as tractable as possible. However, it must have a complex nature in order to be able to develop complex models. It is a primary task of this framework to make sure that it presents and manages its elements and constructs in an efficient, even elegant, manner so that it provides a tractable and powerful modeling tool.

Or course, this complexity makes the description and explanation of Organodynamics itself challenging. Because of this, we shall find it from time to time helpful during the portions of this text that introduce various aspects of this framework, to have an analogy of a familiar example “construction kit”.

Having such an analogy will be useful when trying to explain the role of newly introduced Organodynamics mechanisms, their role within the framework and some motivation behind why they are needed and how they can be expected to be used during the construction of lifelike models.

The specific *analogy* that we shall use is a hobbyists’ kit for designing and building electronic circuits. In this text, we are behaving as the designer of the Organodynamics framework whose users are building models of lifelike systems. Just so, the manufacturers of this analogous electronic circuit hobbyist’s kit are designing the kit for users who are going to develop electronic circuits.

Thus, in our analogy, the author of this text is analogous to the kit manufacturers, Organodynamics is analogous to the hobbyist's kit and our system modelers are analogous to the hobbyists who build circuits.

So, while developing the Organodynamics framework throughout the this text, we shall occasionally compare ourselves to the manufacturers of the hobbyist's circuit construction kit; we shall compare Organodynamics to the hobbyist's kit; we shall compare the hobbyist's kit users to mathematical modelers; and we shall compare electronic circuits to lifelike (organic) system models built using the Organodynamics framework.

This hobbyist's circuit building toolkit analogy will help us to track our progress in the development of the Organodynamics framework as we go along in the text. There are seventeen chapters that incrementally add major mathematical mechanisms to the framework throughout this text.

After the first nine or so of these chapters, we shall have enough apparatus to be able to use the framework to construct a basic *Organodynamic* web – the basic structure in the framework that can provide a comprehensive model of a near-full organic system. (See below in this chapter for an introduction to Organodynamic webs.) The remaining chapters add more mechanisms in order to enable Organodynamic webs to be more robust modeling structures.

Considering what is required to develop the Organodynamics Framework

In preparation for what is in store for us below as we construct the Organodynamics framework, let's give some thought to what it would be like to design and build our analogical hobbyist's kit for building electronic circuits.

In the first place, as manufactures of the hobbyist's toolkit, we would not start from scratch, but would build upon the work of Faraday, Maxwell and others and recognize the basic elements we are working with - including the movement of electron currents through wires. Electrons and wires would be our foundational elements.

Of course, the functionality of electronics pertains to driving electrons through wires to power appliances. Thus, we would provide wires (with electrons), as well as connecting "appliances" to both. We would further realize that electronics deals with *networked systems*, and provide in our toolkit a way for these wires to join together and split apart via some type of *nodes*.

Furthermore these nodes need to provide smart choices, so we would provide a mechanism for developing logic gates. At this point our hobbyist's kit product is getting mature enough that we are willing to declare it as adequate, if basic. And we are willing to anoint it as an "electronic circuit building kit". Later, we would add more bells and whistles to this toolkit to make it a robust product that is attractive to hobbyists and that is competitive.

In the same manner, throughout this text we shall gradually build up the intellectual apparatus of Organodynamics sufficient to have a framework for developing models of living and lifelike systems. Initially, we shall only have

basic elements. But beyond a certain point in the text (the chapter entitled “Simplex Organodynamic Webs”), we shall finally have enough apparatus to be able to produce basic, yet comprehensive, organic system models.

Of course, we can subsequently add improvements, and we shall. For example, a game-changing improvement to our analogous electronic circuit construction kit would be to be able to interconnect multiple circuits. Even more advanced would be adding the use of transistors.

In Organodynamics, we also have our “basic building blocks”. But instead of being electrons, they are the abstract concept of “system” and the elements that comprise them: components, and relationships between components, which we shall call *system organizations* – or *organizations* for short. These are the “electrons” of Organodynamics.

Organodynamics does not need to invent these building blocks. In fact, Organodynamics invents very little. Rather, it builds upon a number of mechanisms, concepts and principles from mathematics – especially probability theory, information theory, systems theory and systems engineering.

Another basic element of Organodynamics is the probability distribution. This entity enables us to model the sixth organizing principle of OCS: *uncertainty*. Together, we use *system organizations* and *probability distributions* as our “connections”; or in the language of our analogy – as our “wires”.

When we are finished developing the Organodynamics framework, we shall have provided the tools to design and develop the ultimate apparatus of Organodynamics – the *Organodynamic web*.

The Grand Scheme of Organodynamics

The ultimate goal of the circuit building toolkit is to enable hobbyists to construct a final apparatus – which is called an “electronic circuit”. In the same way, the ultimate goal of the Organodynamics framework is to enable builders to construct a final apparatus – the *Organodynamic web*.

The Organodynamic web is the final apparatus, the *grand scheme*, of Organodynamics. Any model of an organic system built using the Organodynamics framework will ultimately embody the structure of an Organodynamic web.

An Organodynamic web can be a comprehensive model of an “organism”. That is, except for needed “external supplies” (food), an Organodynamic web is sufficiently comprehensive so as to be “self-managing” – or in OCS terminology, it can be *autocogenerative*.

However, an Organodynamic web is necessarily a very complex structure. Because of this, the framework provides a sequence of other structures that gradually build up to the final Organodynamic web structure. Each of the structures in this sequence is comprised of numerous instances of the more primitive of these structures in the sequence.

The art of developing an Organodynamic web model of an organic system is to understand the interrelationships among this sequence of framework structures and how each models various aspects of the target organic system being modeled.

The Organodynamics Model-building Methodology

The user of the circuit-building kit hasn't produced anything useful until he has a completed circuit which functions as he desires. Just so, the user (modeler) of the Organodynamic framework doesn't have a completed model until he has an Organodynamic web that models his specific organic system.

Of course, the user of the circuit-building kit will usually build a small circuit first – one that does not have all of the desired features. Then gradually the user enhances it in a sequence of improvements. In this way, the circuit starts out small, and then grows to the desired complexity.

Of course, such a strategy is a *methodology*. In fact, the manufacturers of the circuit-building kit will, undoubtedly, notice this methodology and encourage it by adding features to the kit that does so.

As we shall see, modeling methodology is important to Organodynamics also. And various features will be incorporated into the framework to enable and support certain modeling methodological considerations.

For example, the practice just described of building a small prototype circuit first, and then incrementally enhancing it is a widely adopted engineering practice. It enables the concept of “multiple approximations”. The initial simple circuit is a first approximation. Additional incremental circuits produce further levels of approximation.

In a subsection below, we shall lay out a modeling methodology for Organodynamics that supports this multiple approximations. We shall also see how certain aspects of the framework exist to support this methodological approach.

What Organodynamic Webs are made of

We have explained that an Organodynamic web is ultimate apparatus of an Organodynamic model. Being composed of all of the more elementary mathematical structures of the framework, it is capable of modeling an entire self-managing organic system that exhibits all seven of the OCS organizing principles.

As indicated, an Organodynamic web is a network structure – analogous to an electronic circuit. One can use the Organodynamics framework to construct models of pretty much any “lifelike” system that one can imagine – as long as it exhibits the seven OCS organizing principles.

To say “network structure” implies that there will be at least these: “wires” and “junctions”, or to use the language of Graph Theory, “edges” and “nodes”. As

the text unfolds, we shall see that an Organodynamic web indeed consists of *edges*, *nodes* and other network-like elements.

However, like looking through a really powerful electron microscope at a protein molecule, what initially looks like a linear construct gives way to a chain of nucleotide molecules as the focus is tightened. The “line” that is an *edge* becomes fat – populated by things with breadth, depth and *activity* (“It’s moving around!”).

When the Organodynamics framework starts to unfold, what we have been so far describing in graphical terms (“edges” and “nodes”) begin to reveal themselves as *algebraic and analytical structures*.

In fact, the mathematical elements are really presented first. Gradually, we begin to notice that they seem to “fit together” in a manner that suggests network semantics. Thus, even though we give these higher-level structures names that suggest that they are network elements (“edges” and “nodes”), inside they are constructed of mathematics.

The Edges of the Graph as Time Sequences

At the first turn of “the microscope”, we shall see that the *edges* (of the network that is an Organodynamic web) begin to reveal themselves as *sequences* of some type of mathematical structure. (At first we don’t know what kind of structure. But we do know that it is a sequence.)

Moreover, we learn that each element of these sequences represents a distinct moment in time. So the, in actuality these sequences of mathematical structures represent time sequences.

The Elements of the Time Sequences

Upon a second refocus of the microscope, we see that the elements of these sequences are *probability distributions*. Each *edge*, then, represents the passage of time, and is a sequence of probability distributions, each representing an incremental event in time.

(One type of model that we use below – the Markov chain - promotes these probability distributions to probability matrices. But we shall leave the explanation of this to the body of the text, and stay with the simpler case in this preview.)

We have to dig even deeper to see what event in time is represented by each of these probability distributions. At this point we shall introduce the answer to this question. It lies at the heart of the Organodynamics framework.

In brief, each probability distribution represents the state of a system at one moment in time. And an entire edge represents the changing states of that system over time.

To be more precise, each probability distribution represents a set of possible *system states* that the system being modeled can attain to at the next moment in time following the current moment.

The System States

Every dynamical system theory – and Organodynamics is one – must define how it will define *system state* – and how it will define *change* of system state.

Since OCS is primarily interested in how a system is organized at any point in time, then Organodynamics will define system state as its *organization*. In Organodynamics, system organization is a well-defined concept. In the text below, considerable time is spent on defining *system organization*, so we shall not do so now. For the time being suffice it to say that a system organization is an arrangement – “configuration”, “structure”, etc. – of the *components* of the system.

In other words, Organodynamics considers a system to be comprised of a set of components that can be organized in a number of ways. However, at any moment in time, the system is organized in exactly one of these ways. In addition, which of these organizations is the currently realized state can change at each moment in time.

The System Processes

In Organodynamics, we are going to represent the “life process” of an organic entity as a mathematical time series. This will consist of a sequence of contiguous “time steps”. The representation of each time step will be some mathematical expression that represents the *state* of the organic system at that moment in time.

We have already said that we represent system state in Organodynamics as a particular *system organization* of the underlying system being modeled. Thus, we shall represent a *system process* as a sequence of system organizations – one for each time step of the process.

State Probabilities for the Next Moment in Time

But, Organodynamics is interested in the uncertainty of organic systems. And, Organodynamics chooses to introduce the element of uncertainty right in the middle of the change-of-state of a system process.

The way it is going to approach this is: instead of saying that we know deterministically exactly *which* system organization is going to occur at the next step, we are going to say that we know *probabilistically* which system organization is going to occur at the next step.

The way we express this is to replace the specific system organization a time step with a probability distribution. In this section we are going to describe that probability distribution and give it a name.

But, at this point we have define the *state* of a system at one time step as being a specific *system organization* formed from the population of that system. We have not discussed how to form such an organization, and are delaying that explanation until the body of the text. Suffice it to say that a system organization is some type of combination of the components of the system's population.

What we are going to do now is to replace this single system organization of an underlying system with the set of all possible system organizations of that underlying system.

Moreover, in order to introduce the uncertainty that we require, at this point we are going to assign probabilities to each of the system organizations in this set of all possible organizations.

Thus, for a specific time step, we now have a probability distribution that represents the set of all of the possible *organizations* that can be formed from a particular underlying system. This probability distribution describes the probability that each possible system organization of the underlying system is the one that will be realized as at the next time step.

We are going to name this probability distribution the *organodynamic distribution of the underlying system*.

This probability distribution is one of the most important and foundational structures of Organodynamics. So, lets review what we have just said about it.

Given a time step within a system process, and a system "S" whose state we want to describe during that time step, we are going to define the collection of all possible *system organizations* that can be created from the population of system S at that time. We shall call this set the *organodynamic space* of the underlying system S.

In addition, we are going to assign probability values to each of the members of this organodynamic space of S in such a manner that they are non-negative and sum to 1. (These assignments can have either an empirical or a theoretical basis.)

Thus, we now have a probability distribution whose sample space is the set of all possible system organizations constructible from system S. This probability distribution is called, as indicated above; the *organodynamic distribution of S*. S is called the *underlying system* of this distribution.

Organodynamic Process

At the beginning of the previous subsection, we had a sequence of time steps representing some part of the life of an organic system. We called this sequence a *system process*. Each of these time steps is represented by exactly one of the possible *system organizations* of the underlying system being modeled.

We then proceeded to replace one of its time steps with a probability distribution that we named the *organodynamic distribution of the system*.

In this subsection, we are going to replace all of the time steps of the above system process with these *organic distributions*. The result will be a new kind of process that we call an *organodynamic process*. An organodynamic process takes a system process and introduces *uncertainty* into it by replacing all of its deterministic system states with probabilistic system states – *organodynamic processes*.

Any mathematical process whose terms contain probability distributions is called a *stochastic process*. Thus organodynamic processes are a special case of stochastic processes. There is a rich mathematical theory of Markov chains that we can draw upon – which we shall.

The text is going to explain that there are many kinds of stochastic processes – some of them exhibit more or less degrees of uncertainty than the others. In Organodynamics, we are very interested in degrees of uncertainty. So we are going to exploit these various kinds of stochastic processes and create various kinds of organodynamic processes, so that we can accommodate the variations in the degrees of uncertainty that we find in organic systems.

A type of stochastic process that exhibits considerable degrees of uncertainty is called the Markov chain. Initially, we are going to emphasize organodynamic processes that are implemented as Markov chains. We will also show how to promote Markov chains to other types of stochastic processes that embody reduced degrees of uncertainty. Thus, *organodynamic processes* will take many forms within our modeling framework.

Summarizing the Structures

All higher-level structures in the Organodynamic framework consist of *organodynamic processes*. So, these structures are fundamental to the framework.

In fact, we are going to see that the *segments*, *edges* and *nodes* (network structures) that we previewed above are defined internally as organodynamic processes. And, since the higher-level structures of the framework, including Organodynamic webs, are composed of these network element structures, then they too are ultimately defined as organodynamic processes.

And, don't forget, organodynamic distributions are merely sequences of organodynamic probability distributions of underlying systems. Thus, the hierarchy of structures within the framework is revealed.

Supporting the Seven OCS Organizing Principles

We shall also see that each of these mathematical substructures supports one or more of the seven organizing principles of OCS. At the same time, each structure also contributes to a methodology for building models of living and lifelike systems that contributes to their verifiability and constructability.

Summarizing the Organodynamic Web

Like a “circuit” in electronics, an Organodynamic web is a highly customizable collection of specific component types that can be fashioned into any conceivable configuration for the purpose of accommodating practically any imaginable application in its problem domain – that domain being the construction of models of organic systems.

Structurally, an Organodynamic web has an “outer shell” that can be articulated graph-theoretically as “edges” that consist of “segments” that are interconnected via “nodes”. However, inside of these edges and nodes are several layers of structured mathematical entities that provide the building materials and prefabricated forms to construct dynamical models of living and lifelike systems.

Approximation Levels in Organodynamic Webs

Let’s now take a closer look at the process of using the Organodynamic framework to construct a comprehensive model of an organic system.

The structures provided by the framework as building blocks for these models, as well as the Organodynamic modeling methodology, start out with a simple model and then gradually add more structures and assemble them together into a network. This is a very general network structure called an *Organodynamic web* that can take on any configuration desired to comprehensively model an organic entity.

Thus, every comprehensive organic system model that is built using the framework will result in an instance of an Organodynamic web. This is analogous to saying that every comprehensive structure built by our analogous electronic circuit building toolkit is a “circuit”. In both cases, the product is extremely flexible, general and customizable.

For the remainder of this text, we shall speak in terms of “building Organodynamic webs”. The methodological approach that we shall use will build this Organodynamic web structure in an incremental fashion and in stages. At each stage, we shall add more intellectual equipment, so that each stage will progressively represent an increased level of fidelity to the complexities of organic system being modeled.

As a result, each new stage will represent an *increased approximation* to “real life”. And each stage in the sequence will represent an *nth approximation*. The Organodynamics model-building methodology provides a fixed number of these approximations that the modeler can progress through. In fact, there are *six* of these *approximation levels* – *approximation one* through *approximation six*.

Admittedly, the number and contents of these six approximation levels is arbitrary. However, it is simpler to establish a fixed number of approximations with fixed contents in order to make the exposition of the methodology simpler and more consistent. If after using these six approximations, the modeler decides that a different partitioning is more serviceable, then one is of course

free to reorganize them to fit your needs. However, this text will assume these six approximation levels, and build a methodology around them.

The first several approximations start slowly – beginning with simple model elements and progressing with increasing elaboration to more complex models. Up through the third approximation the models are elemental and do not yet represent a comprehensive structure.

However, at the fourth approximation, a model matures into a basic version of the Organodynamic web structure. At this point, one finally has a near-comprehensive model of an organic system. The fifth and sixth approximations add additional capabilities to make the structure more robust.

The first three approximations gradually build toward an *Organodynamic web* structure. As such, these early *approximations* are designed to have modeling value of their own. However none of them is expected to represent a whole and comprehensive “organism”. That capability is reserved for the Organodynamic web structure that we achieve in the fourth approximation and refine in the fifth and sixth approximations.

The methodology encourages the modeler to start at the first approximation and to proceed incrementally up through the approximation levels – not skipping any approximations. If each approximation is implemented correctly, skipping any of them will not save any time. They are designed to progress the modeler through them in the most efficient manner.

Certainly, the methodology supports stopping the modeling exercise after any of these approximation levels that satisfies the needs of the modeling project. Not all six approximations need to be implemented if the goals of the modeling project are completed at some earlier approximation level.

This approach will permit modelers to develop Organodynamic models incrementally, with each increased approximation building upon the previous.

This methodology starts slowly, with the first three approximations taking small steps, but implementing a considerable amount of content. The final three approximations integrate the first three into a comprehensive and complex entity, the Organodynamic web that constitutes a whole dynamical organic process.

Let us now briefly describe each of the six approximations.

First Approximation: Segment of a Markov Chain

A segment of a Markov chains is defined in Organodynamics to be a maximal homogeneous contiguous sequence of time steps within a Markov chain. By maximal is meant that the last step preceding and next succeeding this sequence, if they exist, have an *organodynamic distribution* that differs from the time steps of the sequence.

Within an Organodynamic web, the *segment* is the smallest “connector” that can bridge two nodes. Segments are the “wire pieces” of the network structure.

The first approximation is introduced in the chapter below entitled “The Uncertainty Model Gradient”.

Second Approximation: Edge of a Markov Chain

An *edge* is a piecewise-homogeneous finite Markov Chain. This means that an edge is a finite Markov chain that can be partitioned into a sequence of contiguous *segments*. This is called an *edge* because it will – at the fourth approximation – become the edge of a graph.

Within an Organodynamic web, the *edge* is a complete “connector” that bridges two nodes. Edges are the “wires” of the network structure. However, an *edge* can be, and usually is, made up of a “string of contiguous *segments*”.

That is, a *segment* is the smallest “easily-calculated” section of a single “wire”; whereas an *edge* is the entirety of a single wire between two nodes.

The reason for “breaking an edge into segments” is that segments are mathematically tractable. Usually, an edge is calculated by breaking into segments, calculating each segment, and then putting the results together to calculate the entire edge.

Specifically, in one version of the organodynamic process, an *edge* is represented within and Organodynamic web as a piecewise-homogeneous Markov chain; whereas the “homogeneous pieces” of that edge are the *segments*. There is considerable mathematical theory for homogeneous Markov chains that can be brought to bear on segments. However, as a whole, an edge is nonhomogeneous and as such generally more intractable. Thus, it is profitable to partition an edge into multiple segments, and to calculate the edge by calculating the individual segments.

The second approximation is also introduced in the chapter below entitled “The Uncertainty Model Gradient”.

Third Approximation: Organodynamic Graphs

At the third approximation, the concept of *node* is added to the picture. A node represents the multiplexing of one or more edges through a single time step. That is, multiple concurrent *edges* can come into the node from the past, and multiple concurrent edges can exit the node toward the future.

Of course, this means that the one or more edges that come into the node from the past exist concurrently. It also means that all edges of a concurrent set have ending time steps that all occur at the same time, say time $t-1$. It also means that the “node” is occurs at a single time that immediately follows time $t-1$, say at time t .

This also means that one or more edges that come out from the node at time $t+1$ are concurrent; and that the initial time step of each of these occurs at time $t+1$.

When multiple concurrent edges from the past combine into a single node, we shall call this a *join* operation. When a node furcates into multiple concurrent edges in the future, we shall call this a *split* operation. Obviously, a single node can be a join operation, a split operation or both.

Collectively, the set of edges that come into or go out of a node, plus the node itself is called an *Organodynamic graph*, which is the third approximation.

The first step in developing a third approximation is to develop multiple concurrent edges. Each of these edges is a model of a process or subprocess within the larger system that one is modeling. For example, if one were modeling a collection of biomolecules and how they interact, then each of the biomolecules could be modeled as a distinct edge. These edges will be concurrent. This means that they share some of the same time steps.

The second step in developing a third approximation is to identify the event that two or more of these edges can combine into a single edge at some specific time step. For example, if one were modeling a collection of biomolecules and how they interact, then one might want to model the event of their chemically combining into a larger molecule. This is an example of a *node* in the network. Specifically, it is a *join node*.

Another example of a node in the network is the splitting of a single edge into multiple edges. For example, if one were modeling a collection of biomolecules and how they interact, then one might want to model the event of a biomolecule splitting apart into several smaller biomolecules. This is also an example of a *node* in the network. Specifically, it is a *split node*.

In this third approximation, we have added additional edges and also have added some nodes that join some of those edges and allow other edges to split apart. Such a network constitutes a *graph* as defined in Graph Theory, because it consists of directed edges with nodes that can collectively represent a directed network.

Let's now look briefly at how we relate the components of Organodynamics that we have already introduced above into such a graph. What we have to explain is how our set theoretic, algebraic and analytical entities become involved in the graph-theoretic idea of a graph.

In Organodynamics, a set of concurrent edges, all of which end at time $t-1$, can join to a single node at time t . This set of concurrent edges is called *node inputs*. Subsequently, another set of concurrent edges can all split from the node at time t and connect to distinct time steps at $t+1$. This second set of concurrent edges is called *node outputs*.

Any node represents an organodynamic transform, or operation, between probability matrices of the ending or starting time steps of concurrent edges.

Generally, such *operations* represent ways that organic systems can divide (split) or combine (join) – for example, cell division and chemical combinations.

Generally, these split and join system operations have inputs and/or outputs. When the probability matrices of ending time steps of concurrent edges join, they form a single combined probability matrix of the subsequent time step. This “joining” represents an *operation* whose inputs are the ending matrices that are being joined, and whose output is the new combined matrix that occurs at the next time step.

Similarly, when a probability matrix at a particular time step “divides” or “splits” into multiple new matrices (representing new organic systems), they come into being at the next time step.

Together, these join and split *operations*, together with their inputs and outputs at previous and subsequent time steps form the *nodes* of an Organodynamic web. And the *edges* whose ending time steps provide the “input matrices” for a join operation are called the *input edges* for the operation node. Likewise, the *edges* whose beginning time steps are the “output matrices” for a split operation are called the *output edges* for the operation node.

Together, the input edges, the output edges and the node are called an organodynamic *graph*.

Being “joiners” and “splitters” of systems, these graphs are able to bring the *edges* of Organodynamics together to form networks.

The third approximation is introduced in the chapter below entitled “Organodynamic Graphs”.

Fourth Approximation: Simplex Organodynamic Web

At the fourth approximation, we finally construct a near-comprehensive, if incomplete, model of the whole system we are modeling. In fact, this is the first approximation that develops a version of the Organodynamic web structure.

In this case, the Organodynamic web is a simple version that usually does not represent a comprehensive model – mainly because nestedness is not yet modeled. That is, within any edges of this approximation, only *simplex systems* (non-nested) are present.

However, this approximation does model the entire target system at a high level of detail.

This approximation improves upon the third approximation in two ways. The first improvement is that it adds enough new edges and nodes so that the entire system being modeled is included somewhere. One can approach this by modeling all of the “sub-parts” of the target system with multiple individual graphs, and then make sure that all of these graphs connect to each other in a larger comprehensive graph that covers the entirety of the target system.

The second way that this approximation improves in the third approximation is that it adds any looping mechanisms necessary to model any feedback and regulation phenomena exhibited by the target system. You might recall that this type of behavior is characteristic of the autocogeneration organizing principle of OCS.

One way to approach this addition to the model is to observe any occasions of the same edges being executed multiple times within your network. If so, then your graph can be simplified by adding loops that enable these edges to be represented only once in your graph. The addition of additional paths to enable the loopbacks will also be required.

This fourth approximation is introduced in the chapter below entitled “Simplex Organodynamic Webs”.

Fifth Approximation: Composite Organodynamic Web

The third OCS organizing principle, *nestedness*, provides for the components of some systems being systems in their own right. This *system nesting* can continue to any finite number of levels.

This approximation adds nested systems, also known as composite systems, to the model. Nested systems can add considerable complexity to an Organodynamic web. Thus an entire approximation is dedicated to their addition.

This fifth approximation is introduced in the chapter below entitled “Composite Organodynamic Webs”.

Sixth Approximation: Joint Composite Organodynamic Web

Recall that we created nodes in the third approximation above by taking the ending time steps of sets of concurrent edges and joining them together through *nodes*.

A limitation of that approximation is that we have to specify in advance the exact times when these *joining* or these *splitting apart* events occur. That is, we have to specify at what time the *nodes* of the network will occur in advance, when the network is designed.

We did not have to do that with the other events – the changes between time steps that occurred as each system changed. All possible changes of a single system were handled probabilistically; no matter what the change is, and no matter what time it occurred.

It would be desirable if we could remove this restriction and allow the joint occurrences of multiple events when they split or join to occur at unpredictable times also. Such is the purpose of the sixth approximation.

Specifically, we need to have the same flexibility for nodes to occur – or not occur – at whatever time they will. Remember that *nodes* represent the joining

together of individual systems into a single system (such as the chance combination of two molecules); or the chance splitting of a single system into many (such as cell division).

Clearly, assuming that we know in advance when two systems are going to join, or one is going to split is ultimately an unsupportable assumption. Imagine two molecules doing their Brownian dance in a primordial soup. Who knows when or if they will ever collide? And if they do, who knows whether they will ever bond?

And, clearly the occurrence of these is just as unpredictable as the occurrence of specific system organizations. None of this is determined. It begs to be modeled stochastically.

The problem with approximation five, the Composite Organodynamic Web, is that the time steps in which these nodes occur must be determined in advance. We must repair this restriction so that the occurrence of nodes can be modeled as uncertain, unpredictable, along with all other events.

This problem is solved by using a single probability space for the system as a whole – including all time steps for all edges and for all nodes.

We shall have to take a new point of view – a point of view of the whole system.

Fortunately, probability theory provides the apparatus we need for this. It is called a *joint probability* distribution.

Thus, our approach to doing this in approximation six is via a joint probability distribution. The whole system viewpoint can be modeled as the joint probability of all of the systems that exist in the process at each time step.

This joint probability space is a new kind of Organodynamic web – one that represents all of these views at once. We shall call it the *joint composite Organodynamic web*.

This structure has the advantage that it can treat the occurrence of the organodynamic transforms, the nodes of the web, stochastically. Thus, it creates a comprehensive model of organic system processes that can be of very high fidelity – higher than that of the fifth approximation.

This sixth approximation is less intuitive than the fifth approximation, and also more complex to develop. However, Organodynamics is not the first mathematical framework to deal with these issues – and to deal with them using joint probability distributions involving stochastic processes (specifically Markov chains!)

We have only to look at Statistical Mechanics, which uses both Markov chains and joint probability distributions – in an integrated model. Of course, there are many differences between the two theories. Both are dynamical theories, but their representations of system state are fundamentally different.

It is expected that many organic process-modeling applications will be content with the fifth approximation – even though it has an accuracy problem regarding the inability to represent join and split operations stochastically.

The sixth approximation will most likely be reserved for applications that demand it. While it is considerably more realistic, the use of joint probability distribution is more difficult to use by non-mathematicians owing to its non-intuitive nature.

The sixth approximation is introduced in the chapter below entitled “Joint Composite Organodynamic Webs”.

Uncertainty and Information

The sixth organizing principle of OCS states that lifelike systems are incessantly changing from uncertainty to certainty and back again, while visiting a continuum of degrees of uncertainty in between.

The existence of uncertainty may be due to:

1. Perceptual shortcomings in the mind of the observer,
2. The endemic uncertain nature of the organic process being observed, or,
3. Some combination of the two.

Nevertheless, regardless of which of the above three obtains, uncertainty and certainty persistently change into each other within organic systems. So states the sixth OCS organization principle. We do not have to settle this argument regarding psychology, epistemology and the philosophy of science in order to treat organic processes as stochastic.

Along this spectrum from certain to uncertain, from deterministic to random, uncertainty changes to certainty and back again. When, uncertainty changes to certainty, it is called *information*.

The relationship between uncertainty and information is profound and deeply seated. Non-intuitive to many, this relationship is the foundation of a discipline named *Information Theory*.

A. I. Khinchin [Khinchin 1957], renowned mathematician of the Russian school of probability theory, in his *Mathematical Foundations of Information Theory*, points out that the degree of uncertainty is the same as the degree of information that is revealed when uncertainty changes to certainty.

Khinchin explains:

Thus, we can say that the information given us by carrying out some experiment consists in removing the uncertainty which existed before the experiment. The larger this uncertainty, the larger we consider to be the amount of information obtained by removing it.

So, the degree of uncertainty and the resulting degree of information are equivalent. One could say that the *information* lies latent within the *uncertainty*. And this explains why the two are measured by the same function (Shannon's entropy).

Because of the nature of this uncertainty, Organodynamics relies on the mathematics of probability theory – and its sub-disciplines *information theory* and the *theory of stochastic processes* – to develop Organodynamic models.

When modeling an organic system, uncertainty gradually changes to information, and our modeling framework needs a way to incorporate this change into the model under construction. (And, in time, additional uncertainty is often revealed. The possible relationships between elements of this new information, itself, often breed new uncertainty.) This means, that each model presented must be able to accommodate *changes in uncertainty degree*.

Accordingly, Organodynamics needs a mechanism to model differing gradations of uncertainty. Thus, the Organodynamics framework provides three distinct mathematical mechanisms for modeling three different grades of uncertainty. This gradual mechanism is called the *uncertainty model gradient*.

Every model in Organodynamics is capable of moving through these three grades of uncertainty. Part of the methodology program of the framework describes how to establish the first of these grades, and then how to transition to the next two grades – gradually decreasing the degree of uncertainty of (and increasing the amount of information in) the models at each grade.

This uncertainty model gradient will be a key ingredient into a methodology for conducting a modeling exercise using the Organodynamics framework. This methodology will be explained in the chapter below on *The Uncertainty Model Gradient*.

Uncertainty and the Six Approximations

Let's put all of this together. When using the Organodynamic framework to develop a dynamical model of an organic system, one would normally initially develop the first approximation of the system.

To do this, one would gradually develop all three grades of the uncertainty model gradient. The first grade will involve the most uncertainty. It will consist of identifying the different conditions that make up the first approximation, and their probabilities.

The second grade will consist of describing how these conditions work together. This will further reduce the uncertainty of the first approximation. Finally, in the third grade, the outcome of the first approximation is realized, and all uncertainty changes to information.

Once, the model for the first approximation has passed through all three grades of uncertainty, it is then time to move on to the second approximation, where all three of its grades of uncertainty will be graduated through.

This scheme is repeated up through each level of approximation to whatever level the modeler elects to take the exercise.

Incremental Example Model Development

It will be helpful to the understanding of this framework if we incrementally develop an example Organodynamic model throughout the exposition of these several chapters of Part II.

We shall select a single biological organism to use as our incremental example. Specifically we shall one of the simplest living organisms known to biology: the *mycoplasma bacteria*.

Unfortunately, for the sake of the economy required in this text for such an example, we shall have to present a very high-level Organodynamic model of mycoplasma, owing to the fact that a detailed model is most likely a multi-person-year effort. However, such a high-level exposition should be sufficient to map out a typical plan of action for developing an Organodynamic model through all six levels of approximation suggested by our modeling methodology.

(The fact that a typical comprehensive Organodynamic model could be a multi-person-year project should alone tell you something about the Organodynamics framework. It is a complex framework that is capable of developing complex models of complex systems. And it is to be expected that a comprehensive modeling project that uses the Organodynamics framework can consume a lot of resources. Of course, a first, second or third approximation model will be much smaller. But a full-fledged fifth or sixth approximation model for a comprehensive lifelike system will require considerable resources. Such is the intention and the scope of Organodynamics. Welcome to the comprehensive modeling of complex systems!)

Near the end of most chapters in this Part II, there will appear a section named "Incremental Example Model Development". In such sections, our mycoplasma model will be enhanced with the mechanisms introduced and explained in the chapter.

Chapters that conclude an approximation level will most certainly present such a subsection. However, most approximation levels will be developed over several chapters incrementally. Usually, such incremental chapters will present this subsection as well.

System State and Organization

Organodynamics can be seen as the mathematical study of a certain kind of “thing” or entity – entities that exhibit the quality of being “organic”. We have already defined by we mean by this quality in Part I of this text.

We must characterize these entities so that they have a definite and basic constitution – and a name. This is the task of the present chapter.

The name that we shall give these entities is *Organodynamic system* – or just *system* for short. Every subject we discuss hereafter in this text will be a development of this idea of *Organodynamic system* and how such development can provide a rich and faithful mathematical representation of “the lifelike” and “the organic” as a dynamical system that can exhibit the seven OCS principles.

The Issue Addressed

The most conspicuous characteristic of living and lifelike systems – according to the perspective of OCS – is that they are highly organized. This has been articulated as the first of the seven OCS organizing principles. It is not insignificant to OCS that the words “organic” and “organization” have the same root.

According to OCS, the form that of organization that living entities take is that of a *system*. *System* is not a loosely defined idea to OCS. Rather it is a concept that has all the history and culture behind it of any theoretical discipline. In the case of *systems*, that discipline is called *systemics*. Originated in the 1930s [von Bertalanffy 1968], the discipline was originally named General Systems Theory and went on to inspire a number of related disciplines, including cybernetics and information theory.

Thus, the notion of *system* is well-defined idea that is at the heart of OCS and of Organodynamics. This chapter establishes the concept on a firm mathematical foundation so that the remainder of the Organodynamics theory can stand on solid ground.

OCS Organizing Principle Supported

Organizing Principle # 1: Organic systems are highly *organized*.

Biological Example of the Issue

Any biological system can be viewed as being constituted by a set of components that enjoy well-defined and established relationships among each other. These relationships constitute the organization, or structure, of these biological systems.

For example, a eukaryotic cell has is constituted by a number of cellular components, including organelles, macromolecules and small molecules. Within each cell, these components are organized so as to enjoy specific types

of relationships. Organodynamics uses the term *organization* to describe ea these relationships.

Some of these relationships/organizations are required in order for the cell to, in fact, be itself. If those relationships are destroyed, we no longer consider the remaining entity to “be a cell”. So, without those relationships, the cell itself is destroyed.

In general, the components of the cell by themselves do not make a cell. Those components must be organized into some relationships before a biologist is willing to call them a cell.

In other words, the components by themselves are not enough to make a cell. Certain relationships must be there as well.

Specific Challenges

Organodynamics must identify a mechanism by which organic entities can be defined as systems.

The Organodynamic Approach to Modeling the Issue

Organodynamics has chosen to use basic set theory from the foundations of mathematics to define the notion of *system*.

This is fortuitous on two counts. In the first place, set theory is sufficiently malleable so as to be able to model any aspect of the domain of this theory – so-called “organic systems”.

In the second place, since the end of the nineteenth century, all mathematical disciplines have been re-articulated to have a firm foundation in the theory of sets. Therefore, any mathematical sub-disciplines that we may wish to bring to bear on the development of the Organodynamics framework will find itself in friendly territory.

Systems

The first step in developing the Organodynamics framework is to define its basic entity – the *organic system*. The *organic system* is the object of study of Organodynamics. Every aspect of OCS is concerned with organic systems – how they are defined and how they can change. For modeling purposes, the *organic system* will be used to represent systems that exhibit lifelike properties.

Specifically, as we saw in the Preview chapter, the fundamental element of the Organodynamic web is, in fact, the simplex system.

A definition has already been presented in Part I, and will be recapped here. An *organic system* is a *system* that exhibits all seven organizing principles presented in Part I.

What is needed and what we shall provide at this point, then, is a working definition of *system*:

System: a collection of entities, together with a description of how those entities are *organized*.

The message in this definition is that a system is more than merely a set of components. Rather, it also consists of an entity that describes how those components are *organized* – or *related* to each other. Together, these two ideas define a system. If the *organization* is missing from the definition, then one does not have a system.

For example, suppose that you have an automobile that you have taken apart; and all of the parts are lying on your front lawn. Then you no longer have an automobile! You have a set of parts (components); but they no longer exist in relationship to each other. They are no longer *organized*. An automobile is a *system* – not merely a collection of parts. The set of components does exist; but not an automobile.

But let's return to our above "definition". This is fine for a loose definition. But we need a precise mathematical articulation of these ideas.

For this, we shall rely on the universal language of contemporary mathematics: *set theory*; where, we shall use the term *set* interchangeably with that of *collection*. We shall also use the term *element* interchangeably with *entity*.

Without further ado, then – and assuming the reader has some very basic understanding of elementary set theory, we shall present a definition of what OCS means by the term *system*:

system: an ordered pair (P; O), where P is a set of elements. P is called the *population* of the system. The elements of P are called the *components of the population* – and are also called the *components of the system*. O is called the *organization* of the system. O uses the idea of "ordered pairs of the population's components" in order to specify certain relationships between those components.

We shall forthwith define P and O in more detail.

System Population

The *population* of a system is its set of *components*. This is the "P" in the ordered pair (P; O) mentioned in the above definition of *system*.

In biological systems, the components are the elements of the system - the "things in relationship" of that biological system.

For example, in biological systems, it is often that a biochemical perspective is taken to understand the system. In this perspective, the basic components of the system are atoms.

We shall define a population P as a set of components. For example, if we have a set of five atoms of unspecified species named A , B , C , D and E , then

$$P = \{ A, B, C, D, E \}$$

System Organization

The essential idea of defining how a population of components is organized is to capture how the components individually relate to one another.

Systemic Relationships

For example, if we have a population of *atoms* – as we do in biochemistry, then these atoms can relate to each other via covalent bonds.

These relationships are ultimately binary. Therefore we can represent them as *pairs*. For example, if two atoms, A and B , bond to create a simple dimer molecule, then we can represent this dimer as a (A, B) . This symbolism is called an *ordered pair*.

There are other approaches that we could use to represent interrelationships between the components. However, pairs are very general. Virtually any kind of interrelationship can be described using pairs, even if it is not the most efficient approach for every type of relationship. Most of the types of entities whose relationships we want to model are can be understood as network structures. The use of pairs to represent network entities is very widespread. For example, graph theory also use pairs to represent network structures.

In set theory, the use of parentheses to articulate a pair implies that ordering is significant – that (A, B) is different from (B, A) . If we don't think that ordering, or directionality, is involved with our dimer, then we can use either pair to represent the unordered pair.

Lets get back to our population of atoms. Lets suppose that we have a population of atoms of unspecified species that are named A , B , C , D , E and F .

Generally, some of these atoms can be bonded to others, while others of these atoms remain unbounded. For example, suppose that A and B are bonded, and that so are C , D and E ; but that F is unbounded.

Then, the use of ordered pairs to represent these relationships could be:

$$(A, B), (C, D), (D, E) \text{ and } (C, E)$$

Since F is not bonded, then no pair contains it.

And, we can represent this *organization*, or description of relationships, of these atoms as the set that elements are these four pairs.

$$O_1 = \{(A, B), (C, D), (D, E), (C, E)\}$$

We used the subscripted name “ O_1 ” because there are many possible organizations of the population P , and we shall want to discuss them as well. For example there could be a different organization of this population names, say, O_2 or O_{135} .

System Organizations

We are now in a position to define the notion of *system organization*, or simply *organization*. It is important to keep in mind that the idea of an *organization* of a system is to capture and articulate how its components are *interrelated*.

We shall first formally define the idea of of *duple* and then define organization in terms of it.

Definition: *duple*: Given population P of components, a *duple* of P is an ordered pair of related components of P . Whether or not two components of P are *related* is defined by a domain application that is modeled by P .

Definition: *system organization*: Given population P of components, an organization O of P is a set of duples of P .

Terminology: Duples of P are also called *duples of S* . Organizations of P are also called *organizations of S* .

Modeling a Domain with an Organodynamic System

An important question is “What do the above definitions of *system*, *population*, *component*, *duple* and *organization* say about the kinds of real-live systems that we can model with Organodynamics?”

First, it says that we must have a “real-live” system that, in at least one way, can be thought of as “a set of components”. For, example, within a closed chemical system, it is reasonable to decide to say that the “components” of this system are its atoms. We can also say that its components are “small molecules”, or any other entities that we want to “start with” as its basic components.

Secondly, these definitions say that these components can be organized in a number of different ways; and that each of these and that we would like to be able to describe these “organizations” of that population so that these different ways can be distinguished from each other.

There are a number of other things that these definitions say about these “organizations”.

One is that they describe relationships among the components. Another is that each single organization describes the way that the *entire population* is “arranged”, “structured”, “configured” – or any other language that implies interrelationships among components - at a *specific point in time*. This means two things: 1) An *organization* is of the whole population – not just part of it. 2) From one point in time to another, the organization can change.

Point 2) is the subject of the next chapter.

In summary, if we have a “space” (a “lifelike” entity) that we want to model using the concept of an Organodynamic system that we have defined here, such a space must be able to modeled as a set of *components* that enjoy a binary (pairwise) relationship.

Components

Have some notion of *components* (e.g. “atoms”).

Population

The set of all of the components identified as constituting the system.

Duples

The components must have pairwise relationships (e.g. covalent bonds among atoms).

Organizations

The set of all of the duples called an organization. This set represents one way to organize the components. Any other way to organize the components can be represented by another distinct organization of the same components.

System Instance (aka “system”)

A population and an organization of that population. A system is articulated as the ordered pair (P; O), where P is a the population of the system and O is an organization of P.

Class of Systems on a Population

The set of all systems whose population is the same.

As another example, lets consider a eukaryotic cell as a collection of atoms. At any particular moment in time, it can be characterized as specific arrangement (organization) of its atoms as organized by covalent bonds. This means that the population of such system is the set of atoms involved, and the organization is the set of pairs of atoms where each pair represents a covalent bond between the atoms.

Notice that from one moment to the next, some of the atoms are bonded while others are unbounded. At a different moment in time, within the same eukaryotic cell, the bonding of atoms can change. That is, some of the pairs that were bonded before can become unbounded. Also new bonds can form. This means that at different moments in time, two distinct *system organizations* apply – even though the population has remained the same. This is an example of the concept of *change of system organization* that will be the subject of the next chapter.

In any event, this means that at each of those points in time, the same eukaryotic cell can be characterized by a population of molecules that can be organized into different sets of covalent bonds – or by different system organizations.

At each of those same points in time, the same eukaryotic cell can be characterized as a population of macromolecules that can be organized into duples of organelles.

At each of those same points in time, the same eukaryotic cell can be characterized as a population of atoms that are organized inside of small molecules that are organized inside of macromolecules that are organized inside of organelles.

Notice that we have not talked about the idea that the population of molecules can also be organized into a set of small molecules. Nor have we talked about the fact that these small molecules can be organized into a set of macromolecules, or that the macromolecules can be organized into a set of *organelles* within our eukaryotic cell. More importantly, we have not discussed the idea that our eukaryotic cell can be organized by all of these *levels of organization at the same time*.

Obviously, these are very significant ideas to the notion of the organization of our example eukaryotic cell – as it is to any system worthy of being called “organic”.

However, we shall delay the introduction of the necessary mechanism within our modeling framework to represent this kind of *nested systems*. The reason for this delay is because we want to first introduce a number of other mechanisms that build upon the notions of *system*, *population*, and *system organization* that we have introduced in this chapter. If we were to extend these basic systemic notions with the machinery of nested systems at this time, it would complicate our introduction of these other ideas.

Therefore, we shall postpone the introduction of *nested systems* machinery into Organodynamics until after we have introduced these other ideas – which include *dynamical systems* and *uncertainty*, each in its own set of chapters. After that, we shall return to include *nested system* into the framework, also in its own set of chapters.

We also want to keep the foundations of the machinery of this framework as “atomic” as possible. We want to introduce ideas in basic articulations, and then build more complex ideas as combinations of simpler ones. This enables us to manage the necessary complexity of this framework and at the same time render it accessible.

The downside is that simple ideas by themselves don’t have much “coverage”. For example, right now our eukaryotic cell example is begging for some mention of the idea of molecules. But we will not be able to model both atoms and molecules and how they work together until later when we introduce *nested systems*. However, disadvantage will be compensated for by providing a framework that is constructed of clear, well-defined and very malleable machinery.

With this approach, the ideas around *system dynamics* and around *uncertainty* can be explained without complicating them with nested systems. Then later, when nested systems are introduced, we can also incorporate system dynamics and uncertainty into that discussion.

The Mathematics a System Organizations

Recall that a *system organization* of a population is a set of ordered pairs of that population. This gives rise to the idea of “an ordered pair on a population” being an entity in its own right – and also of there being a space of all such entities. Such a space is called the *product space* of the population.

Set-theoretically, an *organization* is a subset the product space of the population of the system.

This statement is concise, but it deserves an explanation and an example.

The *product space* of the population of a system is the set of all ordered pairs of the components of that population. As such, the product space is a model of all of the pairwise *relationships* that are possible between the components of that population.

In other words, the product space of a system could just as well be called its “relationship space”, because its members are the possible binary relationships of the underlying components.

Consider the product space of our population of atoms. This is the set of all possible pairs of the atoms in the population. In our eukaryotic cell example, we decided that we wanted the pairs of atoms to represent the fact that a covalent bond is shared between them. (This was an application-level modeling decision. We could have chosen some other relationship between atoms to represent with these pairs.)

Thus, our example *system instance* (or, just *system*) consists of the population of all of the atoms within a prokaryotic cell. Its organization is the set of all pairs of these atoms that are covalently bonded. For our example, we shall make another application-level modeling decision and stipulate that the “donor atom” will be represented in the first entry of the pair.

Therefore, our *organization* of these atoms, being a set of ordered pairs of the atoms, can be viewed as a subset of the product space of the atoms.

Normally, there will be a large percentage of all possible pairings (potential covalent bonds) of these atoms that are not actually bonded at the specific moment in time represented by a specific system organization of our molecules in the cell. So, any one specific organization normally does not “have coverage” for a very large percentage of all of the possible pairings in the product space.

Nevertheless, it is very succinct to understand an *organization* of a system as a subset of the product space of the population of the system.

Figure and Background

It may appear that these ideas can do a pretty good job of representing elements of interest in a population (the “figures”), but that it omits the idea of any “background” in which these “figures” are set.

To wit, an *organization* of components (“figures”) describes a relationship *among the components*, rather than a relationship between the components and any “background” against which the “figures” reside.

For example, it may seem that this approach above might be incapable of modeling, say, a chess game. The model may be capable of representing the chess pieces and their relationships to each other. But no mention is made of the chessboard and the relationship of the pieces to the board.

And in chess, the relationship of the pieces to the board is paramount. It is not enough to merely describe the relationships of the pieces to each other. And, it would appear that the chess pieces are the *figures* while the chessboard is the *background*. Thus it would appear that Organodynamics couldn't adequately model a chess game.

(Moreover, it would seem that we would encounter the same problem in Thermodynamics, where the “background” is the space within which the ideal gas is contained, while the molecules of the gas are the “figures”.)

This is a reasonable objection, which we shall now address.

In the chess game, Organodynamics would model both the chess pieces and the squares of the chessboard as *components* of the population. As well, the chess pieces have relationships with each other, the squares of the chessboard have relationships with each other, and the chess pieces and the squares have relationships with each other.

Necessarily, then, an *organization of these components* represents a snapshot of the game after a specific “move”. As we shall see in the next chapter, a sequence of such *organizations* – one for each “move” - represents an entire chess game.

Of course in order to manage this, we need a nomenclature that provides for distinguishing between the two types of components (pieces and squares). And we shall address this issue in a section below regarding ways to extend this nomenclature.

However, for the sake of reducing the complexity of our examples, this text will generally avoid examples in which “backgrounds” are essential parts or the model. For example, below we shall consider a game of Scrabble®. Clearly the Scrabble® board is important to the game. However we shall consider that “it goes without saying” that we need to account for the board, and ignore it in the example – for the sake of reducing the complexity of the example.

We shall also present the game of straight poker as an example. However, this game has no notion of a background that is significant to the game. Many systems are like this.

Environments

Organodynamics does not treat *environment* as a “background” concept. Rather it treats an “environment of a component” as the set of all significant relationships that exist within the system of which that component is a member.

Therefore, we have this

Definition: *an environment of a component*: An organization of a system in which the component is an element of the system’s population.

Thus, an *environment* is an *organization*.

Also, if an entity has an environment, then it is a *component of a population*.

Moreover, given a system, all of its components have the same population.

Simple Non-Biochemical System Examples

At this point we have articulated a definition of some basic ideas from system theory: namely the idea of a *system* consisting of 1) a population of components and 2) an *organization* of those components.

The basic idea is that a *system* is more than merely a set of components. It also includes a set of relationships on those components. A collection of components is not enough to make a system. That they be organized by describing how they are related is also necessary to call them a *system*.

The way we have approached this can get pretty complicated for most systems. In the first place, even if the components of the system are individually simple entities, when one organizes them the way that we have specified into an *organization*, the situation can start to get pretty complicated.

After all, in order to imitate what we see in physical chemistry, biochemistry and all living cells, we decided above to define a *system organization* as a “set of dupes of the system’s components”. Admittedly this is already getting complicated.

But, unfortunately, as we will shortly see, in Organodynamics the components of our systems will generally not be simple. The components of *organic system* are usually complex in their own right. In fact, each component of an organic system is usually a complicated entity – consisting of some combination of components of more primitive systems.

Therefore, when one starts to create an *organization* of a population of complex components, the organizations can begin to get very complex right away –

mainly because their duples are pairs of complex entities. This is one of the main reasons why organic systems are so complex.

Therefore, before we start looking at actual organic systems, it is probably a good idea to take a look at some simpler systems that *are not* organic (because they don't exhibit all seven of the OCS organizing principles). However, these example systems will exhibit *some* of these principles.

In this way, we can become accustomed to working with simpler, more contained systems at first.

Two examples have been selected: 1) a game of straight poker, and 2) a game of Scrabble®.

Both of these games can be modeled as dynamical systems that exhibit the properties of organization, emergence, reorganization and uncertainty. However, nestedness, autocogeneration and persistence are not their strong suits. (Of course, in this chapter restricting our concerns to static properties. We are not dealing with any dynamical properties right now. That conversation will begin in a later chapter.)

But, we shall not get into any kind of strict test of their adherence to these principles. We are more interested here in providing examples of relatively simple systems and articulating them in the manner presented above.

In this chapter, we shall concentrate on articulating these two examples as a population and an organization, as defined above. In later chapters, we shall revisit these two examples to exemplify a couple of the dynamical OCS organizing principles: the fourth principle, reorganizational, and the sixth principle, uncertainty. After that, we shall usually resort to purely biological examples, since we shall by that time be discussing more complex properties.

Most of the examples in the remainder of this text involve complex organic systems. The purpose of these two non-biological examples is to develop some fluency in the language of Organodynamics while constraining the complexity of the examples used.

Poker

I hope that “real poker players” will accept my apologies at the outset. However, the rules of the game render it such a tantalizing example, that I must use it here.

Description of the Game Straight Poker

Briefly, let's say that poker is a card game, played with a standard 52-card European or American deck that is partitioned into four “suits” of 13 cards each. Any number of players is allowed. For simplicity of explanation, we will assume that there is a *dealer* who is not a player.

There are several versions of the game. In each version, each player eventually displays five cards (or ceases to play – “folds”). These played five cards are called a *hand* - a term that is overloaded to have at least three meanings. The players then compare their *hands*. The player with the winning hand wins the round.

The determination of the winning hand is made by the rules of poker. These rules specify a number of five-card combinations (“poker hands”), gives each of these combinations a name, and then imposes an ordering on these combinations. Some of these poker hands are called: “one pair”, “two pair”, “three of a kind”, “straight”, “flush”, “full house” and “straight flush”.

Most of these combinations (“hands”) involve some combination of multiple cards, and, as well involve some number of other cards that “left over”, or remain unrelated. For example, “three of a kind” involves three cards that all have the same value (from three different suits).

As well there will be two other cards in this hand that remain “unrelated” to any other card in the hand. “Two pair” also has one card “left over”. “One pair” has three cards left over. On the other hand, “straight”, “flush”, “full house” and “straight flush” have no “unrelated” cards.

The ordering rules of poker provide a well ordering for which type of poker hand has precedents over (beats) which other hands. Given any type of poker hand, it beats certain others and is beat by the remainder of types.

However, within each type of poker hand, the rules go further and articulate a precedence order for the case that two players have the same poker hand type. This precedence order specifies that the high card of each hand type will be compared. (Yes, each hand type has a high card.) The player with the highest high card – within the same hand type – wins over the others with the same hand type.

However, if the high cards of two players with the same hand type are equivalent (e.g. they are both Queens or sevens – but of different suits), then a tie is declared (and the pot is split by the winners).

Thus, the poker rules of precedent are – after all is said – a *partial ordering*, and not a *well ordering*. These issues will surface again in a later chapter when we discuss the notion of *random variables*. (There is an alternative, non-standard, set of rules that imposes an ordering on the four suits; and which is used to break these ties, and to therefore provide a well-ordering.)

Let’s assume that we are playing a version called “straight poker” which is possibly the simplest. Each player is dealt five cards. Then, after betting, all players compare their hands. The winning hand then “takes the pot”, or wins all of the money that was bet.

The game is then repeated many times until the players decide to stop.

Poker Modeled as an Organodynamic System

Poker presents a number of different types of entities all of which are naturally represented as systems. In this section, we shall survey some of these, and notice how they are of different sizes and constitutions.

In the next chapter on the *reorganization* OCS organizing principle, we shall see how a game of poker is a dynamical system (or, *process*) in which various of the examples of the systems that we visit in the present section will appear at different points in time during a poker game as it plays out. Or, as we say in the next chapter, these specific systems are *transformed* from one to another over time during a game of poker.

For now, though, let's simply look at some examples of various types of *Organodynamic systems* that can occur in the game of straight poker.

The 52-card Deck as a System

Let's begin by considering the 52-card deck as a population. We must then consider how, in some manner sensible to the game of poker, we can form some duples of these cards. To do this, we must identify some *binary relationship* that involves two cards (at a time) from the card deck that is also a *meaningful relationship* in the game of poker.

For this, we shall consider the following *open sentence*, where *X* and *Y* are open variables that can be filled by cards from the deck:

X can be in the same poker hand as *Y*.

The duples (*X*, *Y*) can be any pair of cards from the 52-card deck, as long as *X* and *Y* are not the same card – since there is only one of each card in the deck.

Thus we have identified a system whose population is the 52-card deck and whose organization is the set of all pairs of cards that may appear in any poker hand. This is an example of a very general (non-strict) system that might appear in a game of poker. In fact, this system models the situation at hand prior to the dealer dealing any cards.

A Poker Hand as a System

Once some cards are dealt, then some other – more specific – systems begin to appear. Let's look at some examples of these. For example, any poker hand can be modeled as a system.

In fact, it is true that any five-card hand constitutes a legal poker hand of some type. In particular, a five-card hand that does not meet the criteria identified in the list of poker hands in the above section still constitutes a poker hand. These fall into the category of a type of poker hand named "High card" – an example of which is: 7H, 5D, 4S, 3S, 2D.

From this explanation, it can be seen that, in fact, there are $52 \cdot 51 \cdot 50 \cdot 49 \cdot 48$ such hands, each of which can be modeled as a system of the form (P; O). For

example, the “Full House” hand 7D, 7H, 7S, QH, QD can be represented as follows:

$$\begin{aligned}
 P &= \{ 7D, 7H, 7S, QH, QD \} \\
 O &= \{ (7D, 7H), (7H, 7S), (QH, QD) \} \\
 \text{Thus, } S &= (P; O) = \\
 & \quad (\{ 7D, 7H, 7S, QH, QD \}; \{ (7D, 7H), (7H, 7S), (QH, QD) \})
 \end{aligned}$$

So far, we have seen two types of systems based on the 52-card deck. One is the very inclusive system whose population is the entire deck, and whose organization consists of all possible pairs of cards in the deck that can appear together in the same poker hand. The second type of system also consisted of populations of individual cards. These systems represented individual poker hands. However, the populations of these systems were limited to 5 cards, and their organizations depicted their relationships to each other in a poker hand.

All Poker Hands as a System

We shall now look at our third and final example if a system related to the game of poker. This time, the population will be different. Rather than individual cards, as before, we shall define the components of the population as an entire poker hand. In other words, in this system, each individual component is a complex combination of cards.

In fact, we are going to define our population as the set of all possible poker hands. As we have already discussed, any five-card hand constitutes a legal poker hand of some kind. Thus, our population is the set of all possible 5-card hands.

As we have seen, any population can generally have a large number of organizations. However, from a modeling perspective, any such organization is expected to “make sense” to the domain that it is modeling. Thus, since our domain in this example is the game of poker, we should develop an organization of our population that “makes sense” to the population of all poker hands within the world of the game of poker.

For example, our organization could specify the “value order” of the population of all poker hands. You may recall that in the above subsection where we described the rules of straight poker, we stated that, given any two poker hands, the rules of poker specify which “beats” the other.

Thus, we shall define an organization of the population of all paper hands that specifies their value order. This can be accomplished, for example, by listing all possible duples of poker hands where the “winning hand” is listed first in the duple and the losing hand is listed second. Any hands that do not appear as a duple are ties.

Comment about the Choice of Ordered Pairs

It turns out that the use of ordered pairs (which we call *duples*) is not the most efficient strategy for the representation of many, if not most, poker hands.

Rather, using the structures provided by Combinatorics – *permutations* or *combinations* – would have been a more efficient choice for this example. Permutations are represented by n-tuples, also called tuples; while combinations are represented by sets. Specifically, the use of *combinations* would have been an excellent choice, since orderings are not significant to the game.

The use of *combinations* would have provided a streamlined representation – certainly as compared with the use of ordered pairs (duples). For example, hands involving three-of-a-kind are easily represented by a single set. Consider a hand that contains three queens, say QH, QD and QS:

$$= \{ QH, QD, QS \}$$

But, if we stay with our use of ordered pairs (duples), then we have the following representation:

$$= \{ (QH, QD), (QD, QS) \}$$

Moreover, this representation is not unique.

However, even though combinatorics combinations is a more efficient strategy for this example, it is a poor choice for certain other types of applications – especially *network*, or *graph-theoretic*, ones. For these types, the use of ordered pairs is usually more efficient and also unique.

We also find that in living system, network and graph-theoretic configurations are prominent. Thus, while using combinatorics cannot easily model some of these, ordered pairs can model all of them, and is often a more efficient approach.

Thus, we shall stay with the use of duples to articulate *system organizations*, even though there are some examples, such as our poker game, for which they are not the most efficient approach.

Scrabble®

Scrabble® is a board game that consists of a set of 100 tiles, each of which has a letter of the English alphabet inscribed on it, along with a numerical value which serves to calculate a point score for the players at several places in the game.

These point scores pertain to the comparative rarity of occurrence of their letter within common English language usage. The more rare the usage of a letter, the more points is assigned to the letter.

In addition, since there are more tiles than letters in the alphabet, then some (most) letters are represented on multiple tiles. Generally, the more common is a letter's usage, the more tiles display it.

The players take turns arranging the letters on a grid, and individually amass

points by making full word combinations. Beyond this, the rules of the game will not be explained here.

As we did with poker, we shall identify a number of examples of systems occurring throughout a game of Scrabble® as it plays out. In fact, as we shall see in the next chapter, a Scrabble® game can be viewed as a sequence of these systems that is played out over time as one or more of these systems are transformed into others at each step.

For identification purposes, we can name each tile by the letter it displays, plus an index number. For example, all tiles that display the letter "S" would be named $S\langle n \rangle$, where n is the index. "S3" would be the third S tile.

Systems whose Population is all Tiles

Firstly, let's consider how the game looks at the beginning before any tiles are played when the tiles are laid out face down. However, the board upon which the game is played contains a table that describes the number of tiles for each alphabetic letter, as well as the letter's frequency of tiles.

There are a number of organizations of the game's tiles that the player should be aware of before starting play. Another way of saying this is that the complete set of tiles can be modeled as a number of distinct systems, each whose population is the entire set of tiles.

Each of these systems has its unique organization. For example, the tiles could be organized according to the letter displayed. In such an organization, any of the duples have the property that its two displayed letters are the same but their indices are different. All such duples would be present in this organization.

Another organization of this same population would have the organizing principle which states that its duples have the property that its displayed letters share the same point score. For each such duple, all index combinations would have a duple represented. For example, (A_5, S_2) , (A_1, S_2) and (A_2, S_3) would all be members of this organization, since A and S both share a point score of 1.

A third example of a system with this population has the organization of all pairs of tiles such that the first tile of the duple has a smaller point score than does the second tile of the duple. This organization imposes a partial ordering on the entire population of tiles.

Whether articulated as systems or not, these various organizations of the tiles are essential knowledge required to be able to play a game of Scrabble®. However, this subsection has shown how this essential knowledge can be expressed in the form of Organodynamic systems.

As can be imagined, the number of possible systems with the entire set of tiles as its population is large.

Word Systems

At any moment in time, a set of tiles that constitute a “word play” (either across or down) within the progress of the game constitutes a system. In such a system, the population is the set of tiles that make up the word – including any common tiles shares with other played words.

We shall call these small systems *word systems*, because each of them depicts a word that has been played.

The organization of such a *word system* of tiles would dictate the ordering of the tiles. This could be done, as before, by a set of duples whose first tile is the last tile of the previous duple and whose second tile is the first tile of the next duple. Of course, the first duple has no predecessor and its tile is the first tile of the word represented. The last duple is defined in a similar manner.

For example, after two plays, lets assume that the word “sand” has been played vertically, and the word “hand” has been played vertically, with the “n” being the common letter.

Take note that we need to uniquely identify each tile. This means that it is not sufficient to identify a tile via the letter that it displays. Instead, we shall identify each tile by the letter it displays together with a unique index number subscript.

Then, the following *organization* represents “sand”:

$$\{ (S_i, A_i), (A_i, N_i), (N_i, D_i) \}$$

And, the following *organization* represents “hand”:

$$\{ (H_i, A_i), (A_i, N_i), (N_i, D_i) \}$$

Thus, the *word system* for “sand” is the pair:

$$(\{ S_i, A_i, N_i, D_i \}; \{ (S_i, A_i), (A_i, N_i), (N_i, D_i) \})$$

Game State Systems

Finally, we shall describe a system that – at a particular point in time - represents the state of the game. We shall call these systems *game state systems*. Of course, there will be one of these types of systems after each word play of the game. The idea is that a game could be reconstituted exactly from one of these systems.

This time, the population will depict each of the words that have been played (until “this” point in time), while the organization will depict how these played words are related to each other on the game board.

There are a number of ways we could approach the definition of such a system. Our task, then, is to decide how we shall define both the population and the organization of such a system. We must tackle the population first, since the organization must be defined as a set of duples whose elements are components of the population.

We have already said that each component of our new *game state system* will represent an entire word – a word that has been played in the game. We must now decide how we shall articulate each of these “word” components.

Intuition may suggest that each component, then, be defined as the set whose elements are the names of the tiles that constitute the word represented by the component. For example, if the word is “sand”, then its component in this population would be $\{ S_i, A_i, N_i, D_i \}$.

Another way to say this is that we use the *population* of the *word system* representing the word “sand” as the *component* representing the word “sand” within the population of this new *game state system* that represents the progress of the game.

This approach would admittedly provide uniqueness. However, since sets do not prescribe any ordering on their members, then this set is the same set as $\{ N_i, A_i, S_i, D_i \}$ and as $\{ D_i, A_i, N_i, S_i \}$. Thus, this approach would not enable the reconstitution of the game (say, if the board got upset and the tiles went flying asunder).

What is missing in this approach is the *organization* of the tiles of the word “sand”. But, of course, such organization is captured in the *organization* of the word “sand”.

Another way to say this is that we use the *organization* of the *word system* representing the word “sand” (rather than its *population*) as the *component* of this new *game state system* then we will be able to reconstitute the game.

We have so far decided that the *components* of the population of this new *game state system* will be the *organizations* of the *word systems* of each word so far played in the game.

For example, suppose the word “sand” has been played. Then it deserves representation as a component within the population of the *game state system* that captures the state of the game until this moment. We have decided that its articulation within this population is: “ $\{ (S_i, A_i), (A_i, N_i), (N_i, D_i) \}$ ” – which is its *organization* within the *word system* for the word “sand”.

Our second and final task in creating the *game state system* is to define its *organization*. As always, this organization will contain duples whose elements are the components that we just defined.

However, the two components of any duple must be “related” to each other – as defined by the domain being modeled. In our case, the “domain being modeled” is our Scrabble® game as played to this moment.

What we want to capture in this organization is any intersection of two words on the Scrabble® board. This occurs if and only if two words share a tile.

The obvious approach to defining this organization is to have a duple represent any two words that share a tile. For example, suppose the words “logo” and

“owl” are played and share the second “o” of “logo”. Then the following duple could be a member of this organization:

$$(\{(L_2, O_1), (O_1, G_2), (G_2, O_2)\}, \{(O_2, W_1), (W_1, L_1)\})$$

What's Missing?

We've successfully defined a quite complete set of systems that can be used to describe a number of aspects of a game of as it is played out in time. However, there are some major elements of the game still missing.

These concern the positioning of the game upon the board. At the point that we have so far described, we could still “slide” (translate) the game to any squares on the board as long as all the tiles still fit within the boundaries of the board.

We didn't have this problem with poker, because that game defines no “background” such as a board.

The way that this significant aspect of is Scrabble® handled using Organodynamics has already been alluded to above, and we shall only mention it here.

The solution is to include the squares of the board as components of the system – as well as the letter tiles.

This is not a “cop-out” solution; but rather provides a complete and thorough systemic approach to modeling the game. In fact, the squares of the board actually are full-blown elements of the system and deserve to be represented as such.

The objection may be raised that such an approach is unnecessarily complex. However, the game of Scrabble® already *is* complex. What we are doing is capturing that complexity in a mathematical modeling system – rather than ignoring it. This model of the game is no more complex than – in fact, not even as complex as – the game itself. In fact, the model of Scrabble® presented above is a simplification of the game that only models essentials and omits ancillary game elements.

Additionally, as we shall see in the next chapter, this model is readily promoted to a dynamical one. Much of the preparation for that has already been accomplished in the present chapter.

Admittedly, the symbols involved can become complicated. However, that eventuality is best handled by the use of computers and software. As has already been mentioned elsewhere, a library package of modeling software is planned for Organodynamics in order to offset these complications.

Simple Biochemical System Example

Usually, we think of a closed chemical system as a collection of atoms and molecules that are sealed inside of some container from which they cannot

escape and into which no other chemicals can enter. However, such physical systems usually have an enormous number of atoms and molecules, and usually also have an enormous number of species of atoms and molecules.

Going forward in this text, we shall use this type of system as a canonical example for much of what is presented. But right now we need a simpler biochemical example.

Oxidation-Reduction Equations

Thus, we shall appeal to the idea of a “logically closed system” in which we refuse to consider any but a limited set of atoms and molecules at a selected point in time, even though may be “swimming” in a larger space.

Such an example is the classical oxidation-reduction (o-r) reactions studied by all students of physical chemistry and biochemistry. These reactions are often characterized as involving the transfer of one or more electrons from “donor” molecules to “recipient” molecules. (There is a more general explanation, but this will do for now.)

These types of reactions occur, for example, in almost any situation in which energy is released or work is done. As one might expect, these o-r reactions provide represent of a lot of the chemistry that takes place during cellular metabolism.

These reactions usually occur in a sequence of steps, where each step is a re-arrangement, or (in Organodynamics terminology, a *reorganization*) of the atoms involved in all of the steps considered. This can be viewed as having a specific set of atoms, and then observing them being “rearranged” at each step of a sequence.

In terms of the way we have defined *system*, we would say that the *population* of atoms remains the same across this sequence, and that the *organization* of the atoms changes. These o-r equations consist of two sets of chemical compounds that are separated by an arrow. The arrow indicates that a change of organization (chemical compounds) is occurring; and the chemical formulas on both sides of the arrow is the “chemistry way” of describing these chemical *organizations*.

Here is an example:



In this example, we can infer that we are dealing with a population of 14 atoms – four of potassium and ten of oxygen. On the left side of the o-r equation, one particular organization of these 14 atoms is described. This organization consists of six molecules. The first is a molecule consisting of four potassium atoms.

On the right side of the o-r equation, a different organization of the same population is depicted. This organization has one molecule. This molecule has four potassium atoms and ten oxygen atoms.

Let's look at how we would represent these same two organizations using the Organodynamics way.

First, we have a population P that consists of 14 atoms. Since there are many atoms of the same species, it is not enough to identify each atom merely by its species. Therefore, to name the atoms, let's use a combination of species symbol (i.e. "P") and an index number to distinguish the atoms of the same species. Thus "P1" will identify potassium atom number 1. Thus, in our set-theoretic terminology, our population of atoms P is:

$$P = \{P1, P2, P3, P4, O1, O2, O3, O4, O5, O6, O7, O8, O9, O10\}$$

Next, we have two *organizations* of this population to represent – one for the left side of the o-r equation and another for the right side.

First, let's develop the organization for the left side. Let's call this Org_1 . The left side of the equation reads $P_4 + 5 O_2$. This means that there is one molecule named P_4 and another five molecules, each of which are named O_2 .

In Organodynamics, the first molecule P_4 of the left side is represented by:

$$(P1, P2), (P2, P3), (P3, P4), (P4, P1)$$

The other five molecules, named $5 O_2$ in the chemical formula, all have the same simple "shape" (they are dimers). But since each consists of distinct molecules, we identify them separately as follows:

<grant – Need a chemist to verify these atomic structures.>

$$(O1, O2), (O3, O4), (O5, O6), (O7, O8), (O9, O10)$$

Since organization org_1 consists both of these sets ($P_4 + 5 O_2$), then

$$org_1 = \{ (P1, P2), (P2, P3), (P3, P4), (P4, P1), (O1, O2), (O3, O4), (O5, O6), (O7, O8), (O9, O10) \}$$

Let's also describe the Organodynamics way of expressing the right side of the equation, which we call org_2 . Chemically, org_2 is written " P_4O_{10} ".

<grant – Need a chemist to verify these atomic structures.>

$$org_2 = \{ (P1, P2), \}$$

(P2, P3),
 (P3, P4),
 (P4, P1),
 (P1, O1),
 (O1, O2),
 (O2, O3),
 (O3, O4),
 (O4, O5),
 (O5, O6),
 (O6, O7),
 (O7, O8),
 (O8, O9),
 (O9, O10),
 (O10, P1)
 }

The entire left side of the equation is completely represented by the system $S_1 = (P, \text{org}_1)$, and the entire right side of the equation is represented by the system $S_2 = (P, \text{org}_2)$. We can always expand these expressions by substituting their full expressions.

The reason for having the Organodynamic nomenclature for expressing these systemic relationships is that: 1) it emphasizes the (population; organizational) aspects of these systems as systems, and 2) the nomenclature is applicable to any domain that can be modeled by Organodynamics, and 3) the nomenclature is consistently applicable to any level of organization within a deeply nested system.

The chemistry nomenclature has its own advantages. For one, it is customized for this specific application – physical chemistry. On the other hand, it does not specify the precise relationships among the individual atoms, as does the Organodynamic nomenclature.

Space with Many O-R Reactions

In order to demonstrate how the Organodynamic system nomenclature can be applied to biochemical systems, we have been working with an arbitrarily small “closed system” that contains a set of atoms that one would encounter in a college chemistry textbook. This is obviously arbitrarily constraining.

However, we can enlarge this system and still have it be arbitrarily constrained so that we can still provide a simple explanation of it. Considering a “closed atmosphere” of a large number of the same set of atoms that we just considered could do this. For example, suppose we had 10 thousand sets of these atoms in a closed container. Between any two points in time, some small subset of these sets could undergo the o-r reaction depicted in the o-r equation $P_4 + 5 O_2 \longrightarrow P_4O_{10}$, while the others remained unchanged during that time segment. Then, during the next time segment, some others of these subsets of atoms could undergo the same reaction.

A more realistic example would be a closed container with 10 trillion subsets of these atoms in which we did not arbitrarily limit the possible reactions to $P_4 + 5 O_2 \longrightarrow P_4O_{10}$. Of course, this reaction would be one of many possible ones. Even though this expanded example is getting to be very complex, it is still arbitrarily constrained.

In an actual living cell, the population of molecules considered in the above examples does not exist in isolation from billions of other species of molecules. Nor is the cell itself a closed system. Rather, the cell has a semipermeable membrane that allows atoms (and molecules) to enter and leave.

However, we can model each span of time between when material enters and leaves, and consider such span a closed system – at least for that span of time. Such a system would be “piecewise closed”. And this is precisely what we shall do in general in Organodynamics.

Within such a piecewise closed system as an actual living cell (prokaryotic or eukaryotic), there may be, say, some billions of molecules undergoing chemical reactions across several time steps – all of which occur on a constant, unchanged, population of atoms.

Many of these reactions will be o-r reactions. Also, many of the products of some of these o-r reactions will intermittently participate in non-o-r reactions, and conversely. Overall, across the span of a closed population (which may last a few milliseconds) there will be a mixture of o-r and non-o-r reactions, with the products of both intermingling and participating in both types of interactions.

Thus, it can be seen that in a real cell, these systems are large and complex, with large and complex populations and with large and very complex organizations. As we shall see in the next chapter, not only can the entire cell be seen as a very large Organodynamic system, but also there are, possibly billions of, small and medium-sized systems within these systems.

Again, as with the social games examined above, as the life of the cell plays out, these myriad systems can be seen as transforming from into each other at each step of the process.

Forward-looking Note on System Dynamics

These discussions of system dynamics actually belong in the next chapter on the *reorganizational* OCS organizing principle. But it has been important to introduce these ideas in the present chapter in order to motivate the articulation of so many different system types in both the non-biological and the biological system examples that were presented.

We shall see these various types of systems unfold and transform into each other in sequence when we visit these same examples in the next chapter.

Advanced System Topics

Beginning with a basic system that has a population of components and an organization of those components, there are a number of directions that can be taken to expand on these ideas. This section presents four of these that shall reappear later in this text:

- Organizational state space
- Compound sets
- Organizational sets
- Organizational systems

The Organizational State Space of a System

Above, we presented an example *instance* of a *system* of atoms $S = (P; O_1)$, where O_1 is a particular *organization* of population (or of system S).

However, there are certainly many more such *organizations of P* that are possible besides O_1 . In fact, there are a quite a large number of possible *system organizations* of any interesting single population.

It is the *set of all of these possible organizations* of a system that we wish to concern ourselves with – not only in this subsection, but also in the remainder of this text.

This concept is so important that we shall define it as the *state space* with which we shall be primarily concerned in Organodynamics. We shall refer to it as the *state space of system organizations for a population*.

Definition: state space of organizations of a population:

Given system $S = (P; O)$, the set of all possible system organizations $\{ O_1, O_2, O_3, \dots, O_n \}$ of population P is called the *state space of organizations for population P*.

In Organodynamics, given a system $S = (P; O)$, we shall most often be interested in the *state space of its organizations for its population* than in the underlying population itself, or any particular one of its populations.

Compound Sets

As we mentioned above, we can create new entities by combining other entities in numerous ways. Let's call these new entities *compound entities*.

For example, the compound entities may be *permutations* of an existing set of simpler entities. Any set whose members are compound entities will be called a *compound set*.

For example, suppose we have an underlying set $A = \{ a, b, c \}$. Suppose we use the notion of permutations to form the compound entities. Of course, set A has six permutations:

(a, b, c), (a, c, b), (b, a, c), (b, c, a), (c, a, b) and (c, b, a)

An example compound set of the underlying set A is:

$$\{ (a, c), (b, c), (c, b) \}$$

Another example of a compound set whose underlying set is A is the compound set of the all permutations of A :

$$\{ (a, b, c), (a, c, b), (b, a, c), (b, c, a), (c, a, b), (c, b, a) \}$$

Another example of a type of compound set would use the notion of *combinations* of an initial simpler set. In fact, there is no end to the number of ways that one can form a compound set of entities from one or more initial sets of simpler entities.

In any event, starting with one or more simpler sets of entities, a new set consisting of compound entities can be formed.

Organizational Sets

Suppose one begins with an underlying set and then forms compound entities of it by creating *organizations* of it. This is obviously a third way to form compound entities of an underlying set – as opposed to forming permutations or combinations of it.

Another way to say this is to treat the underlying set as a *population* and then generate some possible *system organizations* of that population.

Thus,

Definition: *organizational set*: a set of system organizations of a population of components.

An organizational set is a special kind of compound set – one that uses *organizational* formation as its compounding principle. We are very interested in organizational sets within Organodynamics.

Organizational System

Suppose we have an organizational set P' of an underlying population P. Then P' is a set of organizations of P. Since P' is a set, it is possible to consider it as a population in its own right that can, in turn, have its own organization O'. This means that O' is a set of duples whose members are organizations of P. Thus, O' is an organization of organizations of P.

We also want to focus on the fact that the components of population P' are *organizations of P*.

Let summarize. We have a system S = (P; O). We also have a system S' = (P'; O') where the components in P' are *organizations of P* – one of which may or may not be O.

Moreover, the duples that are in O' have as their members organizations of P .

Let's call such a system as $S' = (P'; O')$ an *organizational system*.

Organodynamics is very interested in *organizational systems* like S' , and even uses the concept to develop its approach to defining nested systems.

For the time being, though, we will stay with simpler system in order to exemplify a number of points. But, it is useful at this time for the reader to see where we are going with these concepts later in our process of developing a theory of complex systems.

Some Limitations

Our definitions of *duple* and *organization* have been articulated to provide the essential machinery in order to model *organic*, or living and lifelike, systems. As currently specified, they provide the basic modeling mechanism needed – even for very complex system. Perhaps unexpectedly, we shall be able to ramify these notions into complex Organodynamic webs.

However, the notation does have its limitations. We shall choose to live with these for the remainder of this text – mostly due to the added complexity involved in extending them toward greater generality. After all, the concepts and devices presented in this text are complicated enough as it is. To extend these definitions further at this time would complicate the presentation beyond the level that is interesting to many readers.

Nevertheless, we need to, at least, briefly address the limitations of the present articulation of these definitions and ideas, indicate some possible extensions to them that will avoid certain of these limitations, and suggest further research into these issues.

The present section will outline some of these limitations and the problems that they present. The following section will present some potential extensions in order to obviate them.

Nested Molecular Systems

If you study closely the example system of atoms presented in the previous subsection – and in the more general definitions, it will become evident that complex nested molecules, where there are “small molecules within macromolecules within organelles within cells, etc.” cannot be represented with the nomenclature so far presented. Rather, the only organization that is represented is the covalent bonds of the atoms of the population.

For example, a protein macromolecule is composed of amino acid molecules that are composed of atoms. Using the approach presented above, we can specify that there is a set of atoms (the population) and that those atoms are related by some covalent bonds (their organizations). Or, we could view the same set of chemicals at a higher level of organization as a set of amino acids molecules (the population) that are bonded via peptide bonds to produce a

polypeptide macromolecule, or protein molecule (the organization of those amino acids).

However, at this point, one would have to choose which of these levels of organization to represent as a system. Currently, there is no way to represent a single nested system that involves “atoms organized into amino acids” at one level of organization and “amino acids organized into a protein molecule” at a higher level of organization *at the same time within a single structure*.

Organodynamics *is* capable of these nested systems representations. But in order to define them, it must provide more intellectual machinery than has been presented thus far. The provision of this machinery is delayed until a set of chapters that are presented later in this text. This machinery is collectively referred to as “nested systems” and “composite systems”, and relates to the third OCS organizing principle.

Conflict with Terminology used by other Investigators

The idea behind the concept of a *system organization* is that it describes how the various components of the system relate to each other, and thus their organization. In OCS, the notion of *organization* is static. Thus, we could have just as easily used the term “structure” instead, or even the terms “arrangement” or “configuration”.

However, other investigators have chosen to use the terms *system* and *organization* in a more dynamical sense. These choices are certainly legitimate, as is the uses of these terms in Organodynamics.

However, it is important to realize that these uses are different and not to confuse them. Just as Organodynamics has legitimate reasons for defining its terms in the manner that it desires, so do these other theories and investigators.

Here is a specific example of the differences:

[Maturana and Varela 1987] have defined “structure” to mean something different from the usage in OCS and Organodynamics. Also, some usages of the term “organization” include the notion of change [Morowitz 1992], [Maturana and Varela 1987]. But OCS restricts the notion of *organization* to that of static relationships. Change of organization is extremely important to OCS, but OCS uses the term *reorganization* to mean “change of organization”. You will recall *reorganization* being the fourth organizing principle, which we shall discuss at length later. This distinction between static and dynamic organization is significant in OCS.

Extending the System Organization Concept

We have established a basic definition and nomenclature to describe the various possible organizations of the population of a system.

Unfortunately, what we have defined so far is not adequate for characterizing the populations and organizations of complex system states and dynamics. There are two areas of inadequacy for which we must suggest amending mechanisms.

One of them (multiple relationships) pertains to the occasion of the same pair of components sharing multiple significant relationships. For example, both a covalent bond and a weaker bond such as a hydrogen bond may simultaneously bond two atoms.

The other limiting area (relationship details) pertains to complex relationships that require more description than simply the binary notion that they are related or not. For example, we may wish to model the strength of a bond between two atoms, and not merely the fact that they are bonded.

In this section, we shall make suggestions as to how to extend our nomenclature in order to include these two areas.

Multiple Relationships

The first area in which our nomenclature is limited is the fact that the representation mechanism for describing relationships between components, sets of ordered pairs, does not distinguish between two or more relationships on the same pair of components.

For example, suppose components “a” and “b” share two distinct relationships. Then the ordered pair nomenclature (a, b) doesn’t tell us which of the two relationships is referred to. Moreover, if we desire our organization to include descriptions of both relationships, there is no way to do so, since listing the same pair twice in an *organization* is not allowed by set theory.

One way to get around this is to use some kind of marker to distinguish between the two. We shall assume that there cannot be more than a countable number of relationships between the same two components. Thus, we can use an integer subscript to distinguish multiple relationships between any two components.

For example, suppose we have a system whose population is (a, b, c). Suppose further that an organization, O, of this population needs to describe two relationships between “a” and “b” and a single relationship between “b” and “c”. Then, we can define O as:

$$O = \{ (a, b)_1, (a, b)_2, (b, c)_1 \}$$

Of course, some other indexing mechanism could be utilized instead of integers.

However, in the interest of simplifying our explanation of Organodynamics, we shall not utilize this extended notation in the remainder of this text. Such a mechanism will be necessary to make Organodynamics a sufficiently robust

framework to model many, if not most, organic systems of interest. Extending this nomenclature for tis purpose is suggested for further research.

Relationship Details

A second inadequacy of our definition and nomenclature for system organizations is the inability to represent detailed attributes of the relationships between related components.

For example, the two components that we represent within an ordered pair of a system organization may reside in Euclidean space (as well as in algebraic, probability and other mathematical spaces that our system of interest may simultaneously reside in).

In such a case, we may be interested not only in specifying that two components are related but also in mathematically describing *how* they are related. We need a nomenclature that can specify not only *that* two components are related but also *how* those two components are related.

One possible way to extend the for defining system organizations is to build upon the “subscripting” idea suggested in the previous subsection.

The suggestion here is to provide an ordered pair for describing *how* two components are related, and then subscripting this ordered pair with another ordered pair that identifies the two components being related. Of course, this second pair is subscripted with an integer as suggested in the previous subsection.

For example, suppose that we have the same population as above, $P = \{a, b, c\}$. Further, suppose that we want to extend the organization O of P – and thereby of the system $(P; O)$ – that includes a description of *how* these pairs are related. Then, O becomes:

$$O = \{ (x, f(x))_{(a, b)1}, (x, g(x))_{(a, b)2}, (x, h(x))_{(b, c)1} \}$$

One way to understand this extended nomenclature of the system organization “ O ” as defined in the previous subsection is this:

Re-specify “ O ” by

1. Converting every entry of O into a subscript.
2. Preceding each of these subscripts with an ordered pair that describes a functional relationship between the pair of components in the subscript.

Of course, the “ordered pair” that precedes the subscript can be further extended to be an ordered tuple. In addition, the mathematics of this structure can bring to bear any of the machinery of real and complex analysis. As well, the relationships established by such disciplines as nonlinear dynamics can inform these relationships as well. The possibilities are open ended.

However, in the interest of simplifying our explanation of Organodynamics, we shall not utilize this extended notation in the remainder of this text. Such a mechanism will be necessary to make Organodynamics a sufficiently robust framework to model many organic systems of interest. Developing this nomenclature further is suggested for further research.

Relative Degrees of System Organization

Let's reflect upon the nature of system organizations. Intuitively, some organizations of a given population are "more organized" or "less organized" than other organizations of the same population. Equivalently, we may also say that one organization has "more structure" or "less structure" than another organization of the same population.

We can extend this idea. It is also possible to consider how to compare the *degree of organization* of two systems whose populations and organizations are different – even if the populations have different cardinalities.

In any event, by "more organized" we mean, in some sense, that a certain organization of the components has more "interrelationships" among its components than does some other organization of the same components. By internal "relationships" and "structure" of a system, we are referring to the duples (ordered pairs) of the elements of that system that make up an organization of a system in Organodynamics.

Let's look at an example that considers the idea of *organization* being quantified. In other words, consider the possibility of being able to say "one organization is *more organized* than another".

Consider the abstract population $C = \{a, b, c, d\}$. Also, consider O1, O2, O3 and O4, four possible organizations of C.

$$\begin{aligned} O1 &= \{ (a, b), (b, c) \} \\ O2 &= \{ (a, b), (c, d) \} \\ O3 &= \{ (b, d) \} \\ O4 &= \{ \} \end{aligned}$$

Two of these example organizations have two duples, another has one duple and the fourth is the empty set and has zero duples. It is intuitive that O1 has more "structure" than O3, because it has more pairs, or relationships, than does O3.

It is also intuitive that O4 has less "structure" or degree of organization than do any of the other three organizations – because it contains no duples. It would even make sense to say that O4 is "totally unorganized", or even "totally disorganized".

Another interesting question is whether or not O1 is "more organized" than O2. The answer to this question is more complicated and involves a tradeoff. On the one hand, we could say that O1 is more organized because it has "interlocking"

duples. (It has two duples that share a component, b). On the other hand, one could argue that O2 has a higher degree of organization since it includes more components of C than does O1. (“d” is not referenced in any duple of O1.)

We shall not attempt to define a measure of the degree of structuredness, or *degree of organization* of a system at this point. Organodynamics certainly needs such a definition. But for the time being, suffice it to have pointed out that the notion of “the degree of organization” of a system is a sensible idea, and that it is pertinent to the interests of Organodynamics.

In the end, it may not be easy to quantify the idea of *system organization*. Developing a well-defined measure of system organization is suggested for further research.

Conclusions

To recap, we have discussed what we mean by *system* in Organodynamics (and OCS). Recall that an *organic system* is a system that further exhibits the other six properties of OCS that were presented in Part I of this text.

For example, if we could show that a water molecule exhibits all six of the other organizing principles of OCS then we would declare it to be an *organic system*. Let’s assume that it would be difficult to make the claim that a water molecule exhibits organizing principles number 4, 5 and 6. If so, then we should declare that although a water molecule is a *system*, it is not an *organic system*.

Thus, with all of these considerations, the Organodynamics framework supports the *organized* organizing principle.

Incremental Example Model Development

<>

Progress Check

In Part II, Organodynamics, we are gradually and incrementally developing a mathematical framework for modeling living and lifelike systems according to the seven organizing principles of OCS. Once completed, this modeling framework will achieve the form of an abstract but highly flexible mathematical structure that we are calling an *Organodynamic web*.

We are using the example of a hobbyist’s toolkit for building electronics as an analogy to Organodynamics. Both are systems for “building something”, and in both cases the resulting “thing to be built” has a network structure and has been given a general name.

In the case of the electronics circuit toolkit, the general name of the “thing to be built” is “circuit”. The circuits built by this kit are highly customizable to specific applications. In the case of Organodynamics, the general name of the “thing to

be built” is “Organodynamic web”. The models built by this mathematical framework are also highly customizable to specific applications.

In this chapter, we have introduced and developed the most basic structure in Organodynamics – the *system*.

Later, we shall see that the *system* is what the probability matrices that were introduced in the Preview chapter are describing. And, since these probability matrices were at the very foundation of the Organodynamic web, the concept of *system* that we introduced in the present chapter is even *more* foundational to an Organodynamic web than is the probability matrix.

Once can conclude from this that we are at the very beginning of introducing mathematical structures that we shall “pile on top and inside of each other” in order to eventually erect this comprehensive structure called the Organodynamic web.

System Dynamics as Reorganization

The Issue Addressed

The fourth OCS organizing principle is named *reorganization*. It states that systems change their state by reorganizing.

This organizing principle describes the foundations of dynamics in Organodynamics. That is, how state change occurs from step to step within an organic process.

This organizing principle first observes that, since an organic process is a sequence of systems, then state change must incorporate the manner in which systems are allowed to change.

In OCS, systems are allowed to change via a change in their *organization*. This is what it means to say that an organic system is *reorganizational*.

Recall that a *system* in OCS is defined as a pair (P; O), where P is a population consisting of a set of components and O is a set of sets of ordered pairs, or *duples*, of related components (elements) of P. Thus, it follows that a change in an organic system must constitute a change in system S_1 at one step in the process to a system S_2 at the next step in the process.

[Note: As we shall see later, whether or not two components are “related” is defined within an application that Organodynamics is modeling. If the application says that two system components are related, then that fact is preserved within the Organodynamic framework. Organodynamics will define how such “relatedness” is preserved across various transformations. But, it is left to the application to define which components are related and which are not.]

We have defined the *state* of a system to be its *organization* – the “O” part of the ordered pair (P; O). Thus the *state* of a system changes if and only if its organization changes.

This does mean that a system can change without its state changing. For example, it is possible that a component can be removed or added to a system from one step to another, but that the organization of that system can remain the same. For example, we could remove a component that is not a member of any duples of its organization, and that would not constitute a state change, although it would be a change in the system definition. It is also possible to add a component to a system without adding any duples (relationships) to the system’s organization regarding that component. This would not constitute a change in the system state either, even though this change too would be a change to the system.

OCS Organizing Principle Supported

Organizing Principle # 4: Reorganization.

Biological Example of the Issue

In an oxidation-reduction reaction, a change from one step in time to another is represented by an oxidation-reduction (o-r) equation. One side of the equation represents a previous time step and the other side of the equation represents a subsequent step.

In such an equation all represented atoms are preserved across this change in time. One thing that can be said about such an equation is that these preserved atoms have been *reorganized* by the reaction – and represented by the o-r equation – from the previous time step to the subsequent one.

Specific Challenges

To represent a sufficient degree of the dynamics involved via the elements of set theory. To apply this representation to the manners in which systems can combine and split dynamically, yet still maintain “system-ness”.

The Organodynamic Approach to Modeling the Issue

This chapter will describe a deterministic version of the approach to system dynamics that is used by Organodynamics. Ultimately, however, the framework will use a probabilistic version of this approach to dynamics.

Because it is more basic, this chapter will introduce this deterministic version first. In the next chapter, we shall embellish this deterministic version with *choices* and *probabilities* – thus arriving at the probabilistic version of dynamics that will actually be used by the Organodynamic framework.

Thus, this chapter should be considered as preparation for the next. As such, it will not yet add any machinery to our development of the Organodynamic web structure. We shall wait until we have developed the probabilistic version in the next chapter before we update our continued construction of this structure.

We shall continue to utilize the elements of set theory to model dynamical phenomena. This will provide continuity and integration with the approach we took in the previous chapter when we defined the static concepts of Organodynamic – also using set theory. Some of the elements involved are set-theoretic *relations*, *functions* and *operations*. This articulation will be used in both this chapter and the next, as well as in the remainder of this text.

System Processes and Transforms

We would like to represent organic system *change* from one time step to the next within a system process. Moreover, we would like to view these changes as a finite set of contiguous but discrete time steps.

We shall also investigate the viability of the notion of “a system that changes”. That is, we shall look into the question of whether we can consistently speak of a “system” as “changing”, or whether we must be content with the notion of “change of systems”.

System Processes

A good way to represent system change is by using a finite sequence of system states. We shall call such a sequence a *system process*.

We have stipulated that in Organodynamics, we are representing system state as a *system organization*, which we have defined as a set of related duples (ordered pairs) of components from the population of an underlying system.

Therefore, in Organodynamics, we shall represent a *system process* as a sequence of system organizations – each element in the sequence being associated with a contiguous time step. These time steps may be defined in any way as long as they are contiguous.

Notice that these *organizations* that constitute the elements of the sequence that comprise a system process are assumed to represent the same underlying system. This means that each system organization O in such a system process sequence has a population P of components associated with it.

If, for a given system process, all of the associated populations are the same, then the system process is said to be *homopopulous*. If all of the *organizations* are the same, then the system process is said to be *homo-organizational*. It is possible for system processes to be either, both or neither.

Nevertheless, as discussed above, we have defined the *state* of the process is not considered to have changed unless at least one element (duple) of its organization changes.

More formally, we define:

System process: a sequence of system organizations. Thus, a system process is a sequence

$$O_1, O_2, O_3, \dots$$

Where O_1, O_2, O_3, \dots are organizations of the same underlying system whose populations are P_1, P_2, P_3, \dots ,

And where the $(P_i; O_i)$ are systems.

System Transforms

There is a second way to think about system processes: as the changes, or transitions, that occur *between* the elements (time steps) of the sequence that is a system process.

To articulate this, we must identify a way in which represent these transitions. Set theory again provides us with the mechanism that we need: the *function*, also known as a *transform*. We shall call these functions *system transforms*.

So, a system transform between time steps n and $n+1$ of a system process is a *function* whose inputs (*domain set*) are values of the system process at step n and whose output is the value of the system process at step $n+1$.

Recall, of course, that the value of a step of our system processes is a *system organization*. Therefore, a *system transform* is a function that maps a *system organization* at step n to another *system organization* at step $n+1$.

And this is true between all contiguous time steps in a system process.

In other words, a system process has a set of system transforms that occur between every two contiguous time steps of the process. In fact, this set of transforms also form a sequence when put in the order defined by their input elements.

System transforms provide an alternate way to define any given system process. To do this, define a specific system organization as the first element of the process, and then define a sequence of system transforms. The first of these transforms uses the first element as its input, and produces the second element of the system process as its output. Then the remaining system transforms are applied inductively to these results.

Before we go on, let's provide a definition of what we have already described:

System transform: A function that maps a system organization at one time step of a system process to the system organization at the next system in that process.

Example System Processes and Transforms

In order to make these concepts more realistic, let's discuss an example of a system process that models a chemical reaction by the sequence of molecules that result from step to step of the reaction.

Without getting too involved with a complicated chemical example, let's merely assume that two steps, say step 43 and step 44, of the reaction produces these two molecules: at step 43, we have a water molecule; and at step 44, we have an hydrogen dimer. Thus, at step 43, we have the atom H_2O ; while at step 44, we have H_2 . That is, the organic transform from step 43 to step 44 loses the oxygen atom.

Set theoretically, we represent this as follows:

Step 43 has system
 Water molecule = $(C_{43}; O_{43}) =$
 $(\{H1, H2, O1\}; \{ (H1, O1), (H2, O1) \})$

Step 44 has system
 H_2 dimer = $(C_{44}; O_{44}) = (\{H1, H2\}; \{ (H1, H2) \})$

Moreover, we can describe the organic transform T_{43} as:

$$T_{43}(O_{43}) = O_{43}$$

Furthermore, the *reorganization* that has occurred is characterized by the change that has occurred between the *organization* of step 43 - which is the set $\{(H1, O1), (H2, O1)\}$ – and the organization of step 44 – which is the set whose only duple is the pair $(H1, H2)$. That is $O_{43} = \{(H1, H2)\}$.

An alternative view of system process is described as the sequence of transforms that “connect” the steps of the process.

For example, suppose we have a system process

$$P = O_1, O_2, O_3, \dots$$

Then, there exist a sequence of system transforms

$$T_P = T_1, T_2, T_3, \dots, \text{ where}$$

$$T_1(O_1) = O_2$$

$$T_2(O_2) = O_3$$

$$T_3(O_3) = O_4$$

.

.

.

$$T_i(O_i) = O_{i+1}$$

and the codomain of transform T_i ; is the domain of transform for transform T_{i+1} for all i .

We shall name such a sequence the *transform sequence* T_P of the system process P .

System Identity

Colloquially, we speak of a “changing system”, and of “dynamical systems”. This notion implies that somehow an entity called a “system” can change in time, yet still be identified as “the same system” – at least for a while.

In this discussion, we shall assume that a system can be identified only by its attributes.

The basic idea is that such a “system”, in order to have an identity, must have two categories of attributes: 1) one category that contains the system’s *identity* and whose identifying values stay the same across the life of such system, and 2) a second category that contains ancillary attributes that can change over time without changing the system’s identity.

In computer software, particularly object-oriented programming, it is typical to have both of these categories of “system” (object) attributes.

However, in mathematics – and in nature, we do not have the identifying category of attributes. Rather, any attribute of such systems are allowed to

change. Unfortunately, this fact results in presenting a challenge to any attempt to rigorously define the notion of “system identity”. Let me explain further.

Any property of a systems in nature have can change at any time during the life of the system in question.

This fact results in having no way to define the identity of a “dynamical system”. The basic problem is that, in OCS, we have defined the notion of “system” as a static entity. When one of these systems “changes over time”, we call the sequence of changes a “process”. Thus, in OCS the term “dynamical system” is actually a misnomer. We should be using the term “process” instead. However, since OCS is attempting to provide one more theory of “dynamical systems”, we must use the terminology – or at least make the connection.

In any event, whenever we use the phrase *dynamical system* in this text, we are actually referring to a *system process*. Admittedly, this conundrum results from the fact that we have defined the term *system* statically. But we do need a term for the static notion of a population plus an organization of that population. And the term *system* is a reasonable one to choose for that purpose. This leaves us with the situation wherein a system cannot be “dynamic”. For that idea we have the term *process*.

This all leaves us with a situation wherein we encounter difficulty when we attempt to define the notion of “identity of a dynamical system”. Of course, this should be rearticulated to read “identity of a system process”.

We can illustrate what occurs if we try to nail down the notion of “the same system changing over time”. Of course we try to do this all the time when we, for example, give ourselves an identity - a name (such as Grant Holland) - and pretend that “I am the same person moment after moment”.

The problem is that there is no one (identifying) attribute that I possess that can be guaranteed to remain the same over time. Of course, my DNA is a pretty good candidate for such an identifying attribute. But even my DNA cannot be guaranteed against change over “my lifetime”. For example, all of my cells eventually “wear out” and must be replaced. However, such replacement requires gene replication – a process that is imperfect.

Let’s look at this issue theoretically by returning to abstract definitions of system from above. And let’s attempt to define “dynamical system identity”, or “process identity”, by working with the only thing that we have to work with, which is our definition of system as an ordered pair (P; O).

If we look at biological systems, organisms, we already have a problem. This is because we want to allow one of them to “change its organization” (i.e. allow metabolism and other chemical reactions to occur) and yet to keep their same identity (be the “same organism instance). Thus, we can’t use the “organization” part of the (P; O) pair as an “identity attribute”.

On the other hand, we also cannot use the “population” part of the (P; O) pair as an “identity attribute” either – because we want to be able to replace any

dissipating atoms with “fresh” ones. Therefore, we are left with no attributes of a system that we can point to as “staying the same” across time.

Some may try to argue that certain attributes of biological entities are, in fact, identity attributes. Coming to mind is DNA. We have already shown one problem with DNA above. But we also know that DNA cannot work as an identifying attribute because it does not guarantee uniqueness across distinct individual organisms.

For example, we know that “distinct” instances of organisms exist that share the same DNA configuration – because of the existence of identical twins. A second example is cell division. The two daughter cells of an initial cell have the same DNA configuration as the mother cell – whenever no “copy errors” occur.

At the collective level, DNA even has a difficulty as a mechanism for distinguishing species. For example, DNA is allowed to mutate for some number of replications before it is finally considered to have produced a new species. So, for some number of mutations, the species is held to be “the same”. Then all of a sudden, at some tolerance level, the degree of mutation is finally considered to be a new species.

Together, all of these difficulties and inconsistencies result in a great difficulty in rigorously defining the notion of “system identity” within the realm of dynamical systems.

An old folk vignette concerning a woodcutter and his axe exposes this difficulty. The woodcutter bought the axe and used it for a few months. Unfortunately, he broke the handle on a particularly onerous hardwood. So he went to the hardware store and replaced it. After a few more months he wore the axe head out by constantly sharpening it. So he went to the hardware store and replaced it too. Of course, at this point, none of the axe’s two components are the one that comprised the axe when he bought it. So, the question is – is the axe still the same axe? Maybe so – but none of its attributes attests to it.

Automobile manufacturers have a similar problem. Initially, they placed the vehicle identification number (VIN) in exactly one location in the car – often the dashboard. However, if this component is destroyed, then the vehicle cannot be identified. Lately they have decided to print the VIN onto multiple components. Of course, this approach is still not foolproof, and complicates the parts replacement process.

Because of all of these inconsistencies, the Organodynamics framework will deem such attempts to rigorously define the notion of system identity of dynamical systems as ill fated, and will not objectify the idea.

Above, we raised the question regarding the viability of the notion of “a system that changes”. We are concluding that we cannot find a consistent way, simply by observing the properties of a system, to define the notion of a “system changing”. This is because there is no identifying property about it that we can depend on remaining the same across time. This, we must be content with the

notion of a “system process” in which each time step brings a distinct system from the other time steps.

This conclusion implies that the phrase “dynamical system” is an oxymoron. “dynamical process” is consistent, but redundant. Nevertheless, in order to be consistent with widely adopted terminology, we shall – of course – continue to use the phrase “dynamical system”, but with the proper warning label.

Primitive System Transforms

Organodynamics is primarily interested in the ways in which dynamical systems can change from one state to another through reorganization. Toward this end, it is useful for Organodynamics to observe biological systems to see what reorganizational change looks like there. To no one’s surprise, attempts to observe and analyze the kinds of change that occur in living systems find them to be overwhelmingly complex.

However, it is worth the effort to attempt to identify some *patterns of organizational change* in biology. If we can do that, then we can generalize these patterns of change to a general dynamical theory of living systems. Such is the goal of Organodynamics.

A first observation is that change in biological systems occurs concurrently at many different levels of organization: organism, organ, tissue, cell, macromolecule, molecule, atom, etc. Going in the other direction of increased generality, change also occurs within families, groups, cultures and environments. The semantics of all of these levels of organization, of course, differs by level. Thus, any attempts to characterize the kinds of change that occur at all of these levels from a biological perspective that is peculiar to the semantics of each level can be very challenging.

This pertains to the second OCS organizing principle: Nestedness. Because of its complexity, we shall save the development of mathematical constructs to model system nestedness, or *composite systems*, until near the end of the development of Organodynamics. Essentially, this text will incrementally build up a dynamical mathematical structure that embodies all seven of the OCS organizing principles. Near the end of this process, we shall add the structure to model composite systems.

We shall approach the issue of characterizing the various ways that change can occur as a logical one, and we shall apply this characterization to the ways in which *systems*, as we have defined them, can change. We shall consider what the logical possibilities are; and these will form the basis for our dynamical theory. Even if all of these logically possible ways of changing system state cannot be found in biology, we can nevertheless permit them within the generalized theory of living systems that we are constructing in the form of Organodynamics.

The present chapter will identify and define some primitive organic transformations. The collection of *patterns of organization* that we shall identify should be considered to be a meager beginning. This endeavor can, in and of

itself, become a deep inquiry. Thus, in this text, we shall provide a candidate “starter set” set of these organic transforms. It is left to further research to revisit this set of transforms and to reorganize it to provide a more comprehensive collection than is presented here.

In fact, the candidate *system transforms* that we shall present here are properly regarded to be *categories* of these transforms, of which there are many potential subcategories and many potential manifestations. It is important that the reader understand that the set presented *are not themselves transforms*; but rather that they are *categories* of transforms. As such, we have abstracted characteristics that any specific transforms of a category will exhibit. And, for the purposes of this chapter, we are treating each category as though it were a transform that exhibits only those common characteristics.

The identification of actual transforms that populate each of these categories is left to further research.

In the next chapter, we shall embellish each of these transform categories with probabilistic characteristics. These probabilistic, or stochastic, versions of these transforms will become central to the development of the Organodynamic web structure that we are constructing throughout this text.

Therefore, the readers will find it profitable going forward to develop an understanding of these primitive transforms in this chapter so that they will be prepared to digest the more complex stochastic version of them that is presented in a later chapter when we introduce the Organodynamic web structure.

A Starter Set of Primitive System Transform Categories

In order to develop a set of primitive system transform categories, we must recall that this mechanism is the mechanism within Organodynamics through which one system state is changed into another.

This chapter represents a “thought experiment” toward identifying an exhaustive set of logical possibilities as to how one system organization can change to another. Another way to say this is that we shall try to identify all of the logically possible categories of system transforms that can exist between one system organization and another.

In this text, we do not actually intend to succeed in accomplishing exhaustiveness. Rather, we shall present a starter set of such categories of system transforms; and then leave it to further research to provide a comprehensive set of these transforms.

In any case, at this point, twelve categories of system transform are presented here. This set of categories will give us some transforms to work with in the development of the other seven organizing principles for Organodynamics in the remainder of this text. Later, when a more elaborate theory of system transform categories has been developed, then these can be applied to the remainder of the Organodynamics framework.

A set of categories of system transforms will now be presented. Any system transform that complies with the description of a category will be said to be an instance of that category.

In many of these examples, we treat a category as though it were an actual transform type, rather than a category. This is for pedagogical purposes. We accomplish this by abstracting the characteristics that any transform of the category will exhibit, and then treat such an abstraction as though it were a useful transform. At least, this approach will allow us to illustrate how the transforms operate, even though the abstracted transform is likely too general to be practical.

Transform Domains and Codomain

It will be useful to review some aspects of system transforms that we shall be discussing. A system transform is a set-theoretic function that maps the elements of one set ("space") to some elements in another. In our case, the first and second sets are either system organizations or some combination of them (including a product space).

It is standard to refer to the "input" space as the "domain" and the elements in the "output" space that the function maps these elements to as the "codomain".

It is significant that most of these transforms do not actually act upon the system organizations of the input systems. Rather, they simply work with the underlying *populations* of those input systems to impose restrictions on the populations of the systems that are the results of the transform category that is being defined.

In general, these definitions intend for the specific transforms that will be eventually defined for each category to do the work of imposing their transformational effects on the *organizations* of the output systems that are their results.

In other words, the *transform categories* presented below provide rules for the resulting *populations* of the outputs of all specific system transforms within the category; but it is left to these specific transforms to define the rules for what their resulting *system organizations* will be.

Join and Split Categories of Transforms

For all system transform categories described below, each transform category maps singletons, pairs, triplets, etc., of systems in the domain space to one system, or to a set of systems, in the codomain.

Join Transforms

Some of the system transforms that are presented are properly categorized as "operations", because they map a collection of two or more systems (in the domain of the transform) to a single system (in the codomain).

For example, a simple chemical reaction wherein two molecules combine into single larger molecule via a covalent bond is modeled in Organodynamics as a *join transform* or operation.

In this sense, these “operations” resulting in “joining” or “combining” multiples “inputs” into a single output. We shall label these types of system transforms as *join transforms*.

Join Transform: A *join transform* maps an n-tuple of system (S_1, S_2, \dots, S_n) in a domain product space S_D to a system S_C in the codomain.

Split Transforms

In defining our notion of a *split transforms*, we want to make sure that we model certain aspects of the behavior of biological entities when they divide or split. We shall do this by imposing some restriction on how a *split* transform accomplishes its “splits”.

We need to be guided by an example. From such an example, we shall abstract the properties that we want to preserve in our definition.

Let's use cell division as our example. When a cell divides, its components first reorganize themselves within the cytoplasm in a manner that forms a *partitioning* of the cells components. This is accomplished by some of the components moving to one side of the cell, and the rest to the other side.

In set theory, a *partitioning of a set* is a set of disjoint subsets whose union is the set being partitioned. We shall refer to these disjoint sets as *compartments of the partitioning*. We shall use this idea of a partitioning of a set and its compartments in our definition of the split transformation.

But let's get back to our example of cell division in biology. When we left off, our cell had partitioned itself into two compartments. Once the dividing cell has done this, it then splits in to two new cells, and it allocates the cell components of one of the compartments to one of the new cells, and the components of the other compartment to the other new cell.

A third thing to notice about these two new cells is that at this point they begin to have a life of their own – interdependent and concurrent with each other. The original cell, however, no longer exists.

We have just pointed out three distinct actions that all take place in sequence within a cell division. There are many other actions that we could also identify within cell division. But we want to abstract these particular three actions, and create a mathematical model of them. This will be our definition of a *split transform*.

Lets pause now to articulate these three actions into the language of systems and reorganization that we have developed so far.

- 1) *Partitioning*. This is a reorganization of the components of a system at step n into multiple *compartments*. (Our biology cell had two compartments; but we shall now generalize to $k > 0$ compartments.)
- 2) *Allocation*. This is the construction of k new systems. The k compartments of the partitioning become the populations of the k new systems. The organizations of the k new systems are constructed in any manner possible as long as they are organizations of their respective populations.
- 3) *Process creation*. This is the creation of k new system processes. The initial time steps of each of these new processes are the k new newly constructed systems from step 2) above. Moreover, the original system process terminates.

We shall immediately re-define step 3) to say that the initial system process “splits”, rather than saying that the initial process terminates and k new processes are created.

We shall further stipulate that all three of these actions have taken place within a single step – step n - of a system process. At the beginning of step n , we have a single process involved. At the end of step n we have k processes, each with its own system. These k processes are a result of partitioning the initial system.

Now, in a split transform, all three of these actions take place sequentially within the same step (say step n) of a system process. So, we should conveniently consolidate all three of them into a single transform, which shall then designate as the split transform. We can accomplish this by defining each of the three as a separate transform or function, and then use functional composition to combine them into a single transform.

So, we shall define the *partitioning* action as function f_1 . f_1 maps the initial system S of the step to a partitioning of S .

Thus, $f_1(S) = \{ P_1, P_2, \dots, P_k \}$, where $\{ P_1, P_2, \dots, P_k \}$ is a partitioning of S . f_1 is called a *partitioning function* of S .

Next we shall define the *allocation* action as f_2 . f_2 maps a partitioning to a collection of systems, each of whose populations are the compartments of the partitioning. The organizations of these systems can be any allowable organizations of their components.

Thus, $f_2(\{ P_1, P_2, \dots, P_k \}) = \{ (P_1; O_1), (P_2; O_2), \dots, (P_k; O_k) \}$, where $\{ O_1, O_2, \dots, O_k \}$ are arbitrary organizations on P_1, P_2, \dots, P_k , respectively. f_2 is called an *allocation function* of a partitioning of S .

Thirdly, we shall define the *process creation* action as f_3 . f_3 maps a collection of systems to a collection of system processes whose initial steps have states that are the systems in the input collection of systems.

Thus, suppose $\{M_1(O_1), M_2(O_2), \dots, M_k(O_k)\}$ is a set of system processes whose initial states are O_1, O_2, \dots, O_k , respectively. And $\{(P_1; O_1), (P_2; O_2), \dots, (P_k; O_k)\}$ is a set of system whose states are O_1, O_2, \dots, O_k , respectively. Then f_3 is called a *process creation function* of collection of systems $\{(P_1; O_1), (P_2; O_2), \dots, (P_k; O_k)\}$.

It follows, therefore, that the composite function

$$T(S) = f_3(f_2(f_1(S))) = \{M_1(O_1), M_2(O_2), \dots, M_k(O_k)\}$$

which maps an initial system S to a collection of system processes is a reasonable way to define any *split transform*.

Join and Split Transforms

Finally, it is possible for a transform to be both a *split* transform and a *join* transform at the same time. This is the case if the transform maps n-tuples of its domain space to a system partition in its codomain. Later, we shall utilize these split and join features to construct various types of system networks, or webs.

The System Transform Category Examples

The third OCS organizing principle says that organic systems are *deeply-nested – in other words, highly composite*. The message here is that organic systems consist of systems within systems within systems to many level of depth – or to many levels of organization.

The system transforms presented here are intended to apply at any level of organization with such a nesting hierarchy. However, the higher the level of organization, the more complex the system is.

We want to provide examples in this chapter that are relatively simple. Otherwise, the concepts of the transforms will get lost in the complexity of the example systems. In order to prevent this, we shall select examples of living systems from a fairly low level of biochemical organization.

This means that our examples will be selected mostly from biochemistry or at most cell biology, rather than from higher levels of biological organization such as physiology, anatomy or ecology.

It should be understood that these system transforms are meant to be applied to these higher levels of organization also. However, in the spirit of making the examples more understandable, we shall stick to lower levels of biological organization for our examples.

In Part III of these articles, we shall provide a unifying and integrated example for all seven organizing principles combined. This example will draw from a higher level of organization of biological systems.

The System Transform Categories

We shall use the remainder of this chapter to introduce a number of categories of these system transforms. More detailed definitions of each are documented in Appendix 1. Please consult this appendix for a deeper understanding of these transforms categories.

Each of these categories has a descriptive name that attempts to characterize the nature of the transformations being defined. Specifically, we shall define the following transform categories:

- Reform
- Catalyze
- Consume
- Expel
- Unite
- Divide
- Acquire
- Divest
- Compose, Integrate
- Decompose, Disintegrate
- Precipitate
- Dissolve

Any of these transforms can technically be either join transforms, split transforms or both. This is true because all of these transforms admit to a product space as their domain space, given that a product space of dimension one is an ordinary system space. This is also true of split transformations since a single system output can be considered as a system partitioning of cardinality one.

Generally, these categories describe constraints on the system populations of the resulting output (codomain) systems. The definitions of what happens to the system organizations of the resulting output systems are left to the definitions of the individual transforms of those categories.

Reform

Sometimes a system can reorganize without changing anything about its population. That is, it merely reorganizes its population. This is the Reform transform.

Replicate

Replicate is a category of *system transform* in which a new process for a new instance of a system is created. *Replicate* preserves both the population and the organization of the domain step of the domain process and creates the first step of a new process – which “operates concurrently with” the original process.

Catalyze

In biochemistry, catalysis is an important, even ubiquitous, transformation. This transformation permits the speed of a chemical reaction to be vastly increased by involving a special molecule, called a catalyst, to participate in the reaction. The result of the reaction when the catalyst is involved is the same as it would be without its involvement – except that the reaction time is vastly more efficient.

Consume

We need a transform to model the consumption of “food” by living entities. When a food component is taken in, it is added to the population. The consuming system is reorganized so that the newly added component can form essential relationships with the existing components of the initial system.

Expel

We need a transform to model the elimination of “waste” by living entities. The Expel transform is the inverse of the Consume transform. When “waste” is eliminated, it is removed from the population. The eliminating system is then reorganized so that the newly removed component is no longer represented in any relationships (duples) in the organization of the resulting system.

Unite

The Unite transform takes two systems from the domain space and creates a new system in the codomain whose population is the union of the other two populations.

Divide

We need a model of cell division and other types of division in biochemical processes. The idea is that the components of a system are partitioned into a set of mutually exclusive subsets whose union is the initial population. Then, new systems are formed whose populations are these subsets. Then any legal organizations are formed on these populations to result in a new set of systems.

Acquire

Acquire is a more sophisticated form of Consume, wherein the “consumed” entity is a system – not merely a “standalone” component.

Divest

Divest is a more sophisticated form of Expel, wherein the “expelled” entity is a system – not merely a “standalone” component.

Compose, Integrate

In biology, molecules can combine to form macromolecules, macromolecules can form cells, cells can form tissues, etc. These formations create hierarchies of systems within systems with multiple levels of organization – all continuing to exist at their original levels of organization at the same time that they form the composite hierarchy.

This is precisely what occurs when two systems “integrate”. Two systems come together to create a new “super-system”. The two original systems still exist as systems in their own right. But now they also participate in the new integrated system as though they are components of that new system. And, as components, they organize into relationships with each other within the new system. Thus, a new level of organization is also created – the new integrated system.

Decompose, Disintegrate

When complex biological entities “die”, they decompose. This decomposition is a result of the fact that these systems were initially formed through the Compose transform, also called the Integrate transform.

This means that a hierarchy of systems within systems constitutes them. When they decompose, they unwrap this hierarchy starting from the top.

The Decompose transform models this decomposition. It is the inverse of the Compose, or Integrate transform.

Precipitate

Within large system organizations, “cliques” can form. These are sub-organizations that can coalesce from some small number of interrelationships that already exist within the larger organization.

The sub-organizations can then begin to behave as “small systems” in their own right in such a manner that they distinguish their existence from the larger system. Moreover, as relationships, they become codified into a new system organization that segregates itself from the other interrelationships of the organization.

In this way, their members begin to emerge as a distinct population from the initial population. Thus, at this point both a new population and a new organization has emerged – and thus a new system. This new system is called a *precipitate* of the initial system. The transform that produced it is called the *precipitate transform*.

Precipitate is similar to Divest, except that it begins with the interrelationships and develops populations second.

Dissolve

Dissolve is the inverse of the *precipitate transform*. It takes two system and adds the duples of the organization of the second to the organization of the first. It also adds the elements of the duples of the organization of the second system to the population of the first.

System Process Webs

At this point we have developed enough intellectual machinery to be able to define the notion of a *system process network*, or *system process web*. This concept presents the notion of individual system processes “splitting apart” at certain points in time to become multiple system processes; and also of multiple system processes “coming together” or joining at certain points to form a single system process.

The points at which system processes “join” or “split”, as with any network structure, form the *nodes* of the network. The individual system processes that span two nodes become the *edges* of the network.

In this case, the actual nodes are the *system transforms* that we have defined above. And, each of the edges is a finite step system process in its own right.

Thus, we have taken a structure that consists strictly of set-theoretic elements (sets, relations and functions) and interconnected them algebraically; but have also notices that they fit a graph-theoretic articulation.

This *system process web* forms the intellectual foundation for the Organodynamic web that we are gradually constructing throughout this text. It is the “starter format” of the essential construct of Organodynamics. Ultimately, we shall add a number of features to this web structure that shall render it capable of providing a general apparatus for modeling living and lifelike systems.

We shall spend the remainder of this text embellishing this network structure to handle uncertainty, autonomy, self-organization, self-regulation, deep nesting, emergence and persistence – in other words, to comply with all seven of the organizing principles of OCS. As we add additional capabilities to this web structure, we shall alter its name to reflect its newly added dimensions.

Simple Non-Biological System Examples

This section will consider the dynamics of both of the two systems that we started developing in the previous chapter. The idea here is to see how, at some discrete number of time steps, the game can be modeled as system states that change across a transition between any two time steps. Each of these transitions is describable by one of the system transforms discussed above in this chapter.

The representation of systems as having a population of components as well as an organization of duples of those components is demonstrated in this section through the use of these two example dynamical systems.

This same approach is used in Organodynamics to represent organic systems as well. The purpose of this section is to demonstrate the use of this representation on systems that are simpler than the biological ones that we are ultimately interested in through the study of Organodynamics.

It is hoped that some familiarity with this nomenclature will be gained through its use on these simpler systems so that a higher comfort level is acquired by the time we apply this same nomenclature to organic systems.

Poker

A game of poker can be understood as a “dynamical system”, which in Organodynamics we call a *discrete process*, or *process* for short (since we only deal with discrete processes.) By *process* we mean that the game can be viewed as a sequence of “snapshots” across time, and that each one of these snapshots can be described as a *system* with a population and an organization.

Dynamical Systems within the Game of Poker

In the previous chapter, we explained the game of straight poker and also presented some of the “system snapshots” from the game of poker. In this chapter we shall build upon that and show how, as a game of poker plays out, that some of the “systems of poker” appear over time and that they can be viewed as the result of some of the *system transforms* presented in this chapter.

In fact, “playing poker” can be viewed as a set of *system transforms* that transition from a “poker system” at one time step to another poker system at the next time step. Here, by poker system we mean some combination of the examples we visited in the previous chapter, and by *system transforms* we mean the ones we discussed in the present chapter.

Poker Game Narrative

To demonstrate the presence of Organodynamic systems within the game of poker and how they are transformed into each other via the system transform of this chapter as the game plays out, we shall present a brief narrative of the beginning hands of an imaginary draw poker game. We leave the remaining hands of the game to the imagination of the reader.

First, we must explain what “draw poker” is. Draw poker enjoys the same rules of play as straight poker, with a single addition. After five cards are initially dealt to each player, they get to inspect their hands and decide if they would like to exchange any number of their five cards from some randomly selected cards. If so, they remove those cards from their hand, and return them to the dealer. The dealer then deals them replacement cards from the top of the deck.

(The reason for this draw poker game is to make poker more interesting than the “straight” form of the game – which affords players to have high-ranking hands only with extremely low probability. The idea is that allowing the players to choose cards from the initially-dealt hand to be exchanged should have a tendency to improve their hands.)

The initial state of the game, then, is for the dealer to hold the randomly organized deck without having dealt any cards as yet. In this chapter, we are ignoring randomness because it is introduced in the next chapter.

We know that the 52-card deck is a system with a population and an organization. When we say that we are “ignoring randomness”, what is actually mean is that 1) we know what the population of this system is. But that 2) we do not know what the organization of the system is.

In the next chapter, we shall treat the fact that we do not know what the organization of the system is by declaring such organization to be a *sample space* of all possible organizations of a 52-card deck.

At this point in the text, we do not yet know anything about *sample spaces* of possibilities. However, the cards in the deck do reside in some linear order. Therefore, the organization of this system is a set of duples that enforces this organization.

From the perspective of Organodynamics, then, the 52-card deck system has a population consisting of the 52 cards, and an organization consisting of 52-1 duples that define the order of the cards in the deck currently. Thus, currently, at time step zero of the system process that is this game of draw poker, the organization of this system is defined but unknown.

So, at step 0, the population P_0 contains 52 components, and the organization O_0 contains 51 duples. And, of course, the system at step 0 is $S_0 = (P_0; O_0)$.

Next, the dealer deals five cards to each of the players. (Let’s say that there are three players.) This deal can be seen as a *Divide* transform, where the three hands are separated from the remainder of the deck.

But, this act of dealing is an specific instance of the Divide transform category because it has created multiple (4) new systems from one (the card deck). Three of these new systems represent the five cards of the three players. The fourth new system represents the dealer’s deck after the three hands have been dealt.

The fourth output system of the Divide transform is represents the remaining 37-card deck after the three hands are dealt. It population is the set of $52 - 3 \cdot 5 = 37$ cards. Its organization is a set of 36 ordered pairs that specify the ordering of these cards within this deck.

Next, the game moves to time step 2, where each player discards any cards that they want to exchange for a new set of cards – hoping for a better hand.

This is modeled via the *Expel transform*, which is used for the elimination of extraneous components from a system – regardless of organization.

Here, each of the players applies the Expel transform to her five-card hand. The result is a new system whose *effluent system* consists of the discarded cards. Its population is its cards and its organization is a set of duples that represents its organization within a poker hand. The *Resident system* of this transformation has a population that consists of the non-discarded cards and an organization that represents its organization within a poker hand.

Next, the game moves to time step 3. Here, the dealer deals replacement cards to each of the three players. This is a repeat of the same transform as in transition from step 0 to step 1 where in the dealer deals three new “hands”. This time, however, the “hands” consist of replacement cards, rather than 5 new cards to everyone.

Thus, again the dealer, by dealing replacement cards to the players, performs another *Divide transform*. Again, three sets of cards are dealt – one to each player. However, each player receives the number of cards required to replace the discarded cards.

Thus, at the end of step 3, each player has two sets of cards: 1) the cards remaining in her “hand” after the discard, and 2) the replacement set. The total number of cards between these two sets is 5. As well, the fourth new system involved is the modified dealer deck after the replacement cards have been dealt. This system is defined as before, except that it has fewer cards than before.

Finally, the game makes one more transition to time step 4. This transition accomplishes the formation of a new 5-card poker hand on the part of each player. There is also the dealer’s deck to consider. However, it has remained unchanged since step 3. Each of these three 5-card poker hands is organized as already described, where the organization is a set of duples that describes the specific poker hand held by each player.

The game can then end, or continue in this manner for any number of steps until a final “hand” has been played.

Scrabble®

When the game begins at time step, the tiles have all been placed facedown on a surface waiting to be selected. It is certainly true that the identities of the tiles are unknown, and it is reasonable to assert that they are completely unorganized, or even disorganized. Let’s assume for the sake of example that there are three players.

Thus, as a system, it is reasonable to assert that their population is the set of 100 tiles, and that their organization, as a set of duples of tiles, is the empty set.

At time step 1, the players have each selected seven tiles. This action can be modeled as a *system transform* using the *Divide transform* – as with the poker

game. In this case, the Divide transform yields four systems: one each for the players and one for the remaining unselected tiles.

Each of the player's systems has a population of seven tiles. Once a player has obtained their seven tiles, she arranges them upon her "tile tray" in a linear ordering according to her own desires. This results in each player having a unique organization of tiles.

Thus, the *system* that represents the tiles of each player is specific to the player in both population and organization. Players are constantly rearranging their tiles in order to study the possibilities of constructing words to play. Each of these rearrangements constitutes a *reorganization* of the individual player's system. These rearrangements are modeled via a *Reform system transform*.

Specifically, between the time steps that a single player makes a word play and then draws replacement tiles, these reorganizations are homopopulous. However, when a player draws new tiles after a word play and then begins to reorganize her tiles, then the population of the system of tiles on her tray changes. Therefore, across the entire game, these instances of the Reform transform are neither homopopulous nor homo-organizational.

Another set of systems within the game that was pointed out in the previous chapter are the individual words that are played upon the Scrabble® board as the game proceeds. Each of these played words has a population whose components are the tiles played and whose organization is a set of $k-1$ duples that relate each pair of adjacent tiles.

Each of these words comes into existence by "precipitating out" from the tiles in the player's tray. Actually, what occurs is somewhat more complex than suggested here. What first occurs is that the player notices an "exposed" tile on the game board and "mentally" consumes the tile into her tray (using the *Consume transform*).

At this point, she reorganizes her tray to account for the abstractly consumed game board tile so that there is now a subset of duples with the new tray organization that represents the arrangement of tiles that can be played around the exposed board tile. This arrangement, in fact, includes the exposed board tile.

If this "arrangement" could be made into a system, then it could be played as a word in the game. But, the arrangement is a subset of duples. Thus, if a population is created whose components are the set of all components involved in this subset of duples discussed, then we would have both the population of tiles together with their organization that specifies the correct spelling. Therefore, we would have a system that specifies the word to be played.

All of this can be accomplished by applying the *Precipitate transform* to the abstract arrangement of tiles mentioned in the previous paragraph. In this case, the abstract arrangement of tiles is treated as the sub-organization that is removed from the tray's system organization, and whose tiles are removed from the tray's population.

As the result, this Precipitate transform produces two new systems: the first is the *precipitate*, which is the new word that is played. The second is the residual system, which is the remaining tiles left on the tray with a new organization that is imposed by the player.

Each of these transforms requires its own time step in the system process that is unfolding as the game is played.

The final system that we shall discuss for is Scrabble® the accumulation of tiles that constitute the growing set of words played at any point in time on the game board. This system, as described in the previous chapter, has a population whose components are entire words, and whose organization consists of dupes of these words, wherein a related pair of words shares a tile. This system is, in fact, a nested system – one whose components are, themselves, systems.

At the same time that each played word is formed via the *Precipitate transform* just discussed, a second new system is formed that represents the accumulated game.

This accumulated game at time t can be seen as the result of the a *Unite transform* whose inputs are 1) the previous accumulated game instance, and 2) the new word system that is played as a result of the *Precipitate transform* just discussed.

In this section, we have followed the narrative of the play of an instance of a game of Scrabble® as it has played out for a few time steps. Of course, the game continues in this way.

It is hoped that the point has been made that a game of Scrabble® can be views as a dynamical system – or actually a *system process* – in which the *system transforms* discussed in this chapter act upon various systems of tiles and sets of tiles at various time steps and which transforms these input systems to other systems of tiles and sets of tiles at new time steps in the process.

Incremental Example Model Development

<>

Progress Check

In Part II, Organodynamics, we are gradually and incrementally developing a mathematical framework for modeling living and lifelike systems according to the seven organizing principles of OCS. Once completed, this modeling framework will achieve the form of an abstract but highly flexible mathematical structure that we are calling an *Organodynamic web*.

We are using the example of a hobbyist's toolkit for building electronics as an analogy to Organodynamics. Both are systems for "building something", and in both cases the resulting "thing to be built" has a network structure and has been given a general name.

In the case of the electronics circuit toolkit, the general name of the “thing to be built” is “circuit”. The circuits built by this kit are highly customizable to specific applications. In the case of Organodynamics, the general name of the “thing to be built” is “Organodynamic web”. The models built by this mathematical framework are also highly customizable to specific applications.

Because of the complexity of organic systems, any model that represents them needs to be built in cumulative and increasingly accurate stages. Each stage is generally referred to as an “approximation”. Organodynamics as a model builder’s toolkit provides for six of these approximations. These approximation levels are well defined and occur at precise places in the methodology set forth by Organodynamics.

In this chapter we have introduced the machinery of the *system process* as a sequence of *organizations* of the components of an underlying system – each of which occurs at a sequentially contiguous time to its neighbors.

We also introduced the concept of a system transform that connects two time-consecutive system organizations within a system process. We showed how the sequence of these system organizations between consecutive time steps could serve as an alternate description of a system process.

In addition, we introduced the concept of *split transforms* and of *join transforms*. Join transform are system transforms that can take multiple system transforms as inputs. Split transforms are system transforms that product multiple system transforms as outputs.

As such, split and join transforms can sever as nodes in a network whose edges are finite state system processes – as long as the time steps of these system processes synchronize at these nodes.

These networks, called *system process webs*, form the foundation of the Organodynamic web structure that is the *grand scheme* of the Organodynamics framework.

However, we shall not elect to anoint this concept of *system process web* as one of the six *approximation* levels. Rather, we shall wait until we have developed a number of other mechanism that work together so as to be able to embellished the *system process web* structure with the requisite machinery that is so essential to Organodynamic structures.

This will finally occur at the chapter below, entitled “Simplex Organodynamic Webs”, in which we shall invest the notion of *system transform* with its probabilistic equipment.

System Dynamics as Uncertainty

The Issue Addressed

The sixth OCS organizing principle loosely states that any and every possible degree of uncertainty may be evident in the processes of organic systems. This chapter must put mathematical specificity to this idea.

We have made the case that there are many dynamical systems - particularly complex systems and especially organic systems - that exhibit considerable uncertainty and unpredictability. This may occur either 1) because the modeler does not have sufficient information concerning the nature of the underlying system, or 2) because of some inherent randomness or uncertainty in the underlying nature of the system itself, or 3) both.

This text will take this statement at face value and will not pose any argument as to which of the three is the case – preferring instead to leave such speculation to the philosophers - and the quantum physicists.

In order to develop a mathematical approach to modeling such dynamical systems, Organodynamics has chosen to leverage probability theory and its extensions information theory and theory of stochastic processes.

This *principle of uncertainty* specifically refers to what occurs between one time step and the next within living system processes, or organic processes. (Organodynamics models time series in discrete units.) The principle makes the assumption that the change in system organization between one time step and the next is unpredictable, or uncertain - and that this general uncertainty is a significant characteristic of organic systems.

You will also notice that the way that this uncertainty is described means that it refers to a *dynamical characteristic* of organic systems, because it is applying the notion of uncertainty to the occurrence of change from one time step to another. Thus, in Organodynamics, *uncertainty*, as applied, is dynamical.

Moreover, the uncertainty that occurs between one time step and another in these organic systems *has a degree*. It is not simply a binary condition of completely certain (deterministic), or completely uncertain (totally random). While these two conditions are permitted, they are considered to be extreme polarities of the possibilities of the degrees of uncertainty that will be supported, or modeled.

Thus, Organodynamics is interested in *measuring* this degree of uncertainty/certainty between any two contiguous time steps of an organic process (and within whole processes when that notion is sensible). And this measurement will become a central characterization of those time steps within the Organodynamic framework.

This chapter introduces the mathematical mechanisms of Organodynamics that support uncertainty and its degrees. These mechanisms are provided in the

form of various topics from probability theory, including probability distributions and stochastic processes; and information theory, which is based on these probabilistic concepts.

These probability mechanisms shall be used throughout Organodynamics in order to define other mathematical mechanisms and structures that model several OCS organizing principles, including nested systems, emergence, reorganization and autocogeneration. In this sense, the material developed in this chapter is foundational to the Organodynamic framework.

The reader may have noticed that the titles of this and the previous chapter may seem to be in conflict. One implies that Organodynamics will treat *system dynamics* as *reorganization*; while the other implies that Organodynamics will treat *system dynamics* as *uncertainty*.

Of course, Organodynamics will treat *system dynamics* as both *reorganization* and *uncertainty* simultaneously – or as *uncertain reorganization*. This chapter must explain how this is done and provide a mathematical model of system dynamics that does both simultaneously.

OCS Organizing Principle Supported

Organizing Principle # 6: Uncertainty.

Biological Example of the Issue

Biological processes exhibit every degree of uncertainty from deterministic and near deterministic to near random and completely random.

For example, DNA replication is near deterministic. If it were completely deterministic then all DNA replicas would be perfect copies of their source molecules. However, we know that this is not the case. Copy errors occur in DNA replication on an order of magnitude near one in a billion nucleotides.

And, it is fortuitous for life on earth that DNA replication is *not* perfect. If it were, then adaptation, diversification and evolutions of the species could never occur. There would only have been one species, and it would be extinct by now. Thus DNA replication is nearly deterministic, but exhibits a small degree of randomness. Organodynamics is interested in characterizing and measuring this degree.

On the other hand, meiosis exhibits high degrees of uncertainty. And there are biological processes that exhibit intermediate degrees of uncertainty. For example, the bonding of various biochemical compounds at the molecular level depends on chance encounters of the Brownian motion class. The long run probabilities of these encounters are nearly inevitable, but in the short they can be miniscule. Also, the relative presence of various enzymes can affect the degree of uncertainty in variable ways. Organodynamics is interested in characterizing and measuring all of these degrees of uncertainty as well.

Specific Challenges

To map the elements of organic systems to the essential mechanisms provided by probability theory and the theory of stochastic processes in a manner that provides a dynamical model.

Moreover, the structures and mechanisms developed for this purpose must be able to form the foundations of larger mathematical structures and machinery that can be used later in this text to model the other OCS organizing principles, including nesting, emergence, reorganization, autocogeneration and persistence.

The Organodynamic Approach to Modeling the Issue

A *system process*, as defined in the previous chapter, is a sequence of time steps, each of which is represented by a single system organization.

In the present chapter, we shall embellish the system process concept by adding *choice* to each time step. We shall do this by replacing the single system organization at each time step with an entire collection of system organizations – each of which is a possible realization of the system being modeled. We shall then assign probabilities to each of these possible organizations.

This, then, gives rise to a probability distribution for each time step of our new embellished organic process. In fact, we shall establish such a probability distribution for each of the time steps of the organic process. Such a sequence of probability distributions, then, provides a probabilistic version of our system process that we shall call an *organodynamic process*.

Since an *organodynamic process* is defined by probability distributions, it is a type of process called a *stochastic process*. We shall limit our organodynamic processes to a finite number of time steps. Therefore, they will be a special case of stochastic process called a *finite state finite step stochastic process*. This approach enables our model of organic systems to change step-by-step in time and also in a probabilistic manner.

Thus, at the end of the last chapter, we represented an organic process as a sequence of time steps, each described by a specific system organization.

But, in this chapter we shall create an enhanced representation of an organic process by replacing each the *system organizations* at each time step with a *probability distribution of system organizations* at that time step. Thus, our new representation will consist of a sequence of probability distributions – one for each time step.

We shall also see that having a single probability distribution for each time step is sometimes not enough. For these cases, we shall introduce the notion of conditional stochastic processes.

The simplest and most tractable type of these conditional stochastic processes is the Markov chain. The use of Markov chains in scientific and commercial

applications is widespread and well accepted. While the Markov chain will not cover all cases that we need for modeling dynamical organic systems, it nevertheless gives us a good start – and, as compared with more general conditional stochastic processes, it has the added advantage of being tractable.

Therefore, Organodynamics will concentrate on developing a dynamical stochastic foundation based upon finite state, finite step Markov chains – and leave the development of a more complete framework to further research.

The mechanisms developed in this chapter are the enabling mechanism for constructing the first two approximations of the Organodynamic model construction methodology. However, we shall wait until the end of the next chapter, entitled “The Uncertainty Model Gradient” before presenting these two approximations.

Stochastic Models and Uncertainty

Stochastic is another word for *probability*. Stochastic mathematical models use probability distributions in order to model the uncertain nature of processes that exhibit various degrees of randomness or uncertainty.

Being the sixth organizing principle of OCS, *uncertainty* is very important to Organodynamics. This principle states something about the *degree of uncertainty* of organic systems: namely that it is highly variable within an organic system as well as across any set of organic systems (such as a species or an ecology of them.)

Consequently, it behooves Organodynamics to 1) provide a measure of the degree of uncertainty, and 2) demonstrate how to show that this degree of uncertainty does, in fact, vary within an organic system and across related sets of them (species, etc.).

In this chapter, we shall define uncertainty of a system in mathematical terms and provide further discussion of its significance to dynamical systems.

Of course, *uncertainty* is a property of randomness. Mathematically, random processes differ from deterministic ones by virtue of the fact that repeated trials of the same experiment (repeating the same procedure with the same inputs) may produce different outcomes with each repetition. Whereas, in a deterministic process, repeated trials of the same process, using the same inputs, always produce the same outcome.

The set of multiple possible outcomes that an experiment (or system process) can produce in a stochastic model is called the *sample space*. Members of the sample space are called by several terms: *outcomes*, *logical possibilities*, *sample space members*, *sample points*, *sample space events* and others. These terms shall be used interchangeably in this text.

In Organodynamics, we start with the idea of a *system process*, but then we immediately imbue it probabilities and “promote” it to a *stochastic process*.

Whereas a *system process* models a dynamical system *deterministically*, a *stochastic process* models it *probabilistically*.

A *system process* is a time series whose individual terms represent the state of the system at specific time steps. In contrast, a *stochastic process* is a time series whose individual terms represent *multiple possible states* of the system at specific time steps – and the probabilities of each of those possible states at those time steps.

Theoretically, a deterministic process is a special case of a stochastic process whose sample space has exactly one member – the single result that is produced with every repetition, or trial of an experiment.

The essence of stochastic models and probability is that they model the idea of uncertainty and of its resulting unpredictability. The essential difference between a stochastic model and a deterministic one is that deterministic models represent certainty, while stochastic models represent uncertainty.

However, the theory of probability marries the two concepts and is able to represent an entire spectrum all the way from complete certainty to complete uncertainty. This is precisely the nature of living and lifelike systems as understood by OCS according to its sixth organizing principle. Therefore, stochastic modeling is appropriate for Organodynamics.

In fact, *degree of uncertainty* is a measurable attribute of any stochastic process – specifically, the *degree of uncertainty* regarding which member of its sample space will be the outcome (the realization) of any particular trial of the process.

Consequently, Organodynamics is very interested in the degrees of uncertainty of various specific aspects of the organic systems that it models. And Organodynamics needs for its mathematical models to provide a measure for the degree of uncertainty.

Probability Distributions

When conducting repeated trials of an experiment, some events may be more or less likely to be realized than are others. Thus, we have the notion of the likelihood, or probability, of an event (outcome, sample point, etc.)

Each sample event of an experiment, therefore, has a probability. The collection of all sample events of an experiment together with their probabilities is called a *probability distribution* of the experiment. The probability of an event is defined to be real numbers between zero and one such that the sum of the probabilities of all the sample events of an experiment is equal to one.

A Measure of the Degree of Uncertainty of a Probability Distribution

We have seen that the various sample events of a given probability distribution may have distinct probabilities.

A related question is whether all probability distributions have the same amount of total uncertainty inherent within them – or whether some probability distributions contain more or less uncertainty than others.

The answer is that some probability distributions are inherently more uncertain than others. If all of the events of a distribution have the same probability (they are equally likely), then the distribution is highly uncertain. It is inherently more random.

But if a probability distribution has one event with a very high probability, and all of the others are very unlikely, then the distribution itself has a low degree of uncertainty. It is inherently more deterministic.

This being the case, it begs the question as to whether we can develop (or find) a way to measure this *degree of uncertainty* of a probability distribution – and concomitantly of the underlying process that it models. The present subsection explores such a measuring function.

The Degree of Uncertainty Inherent in Random Processes

For example, in sexual reproduction the process of meiosis is very uncertain. Any possible combination of the right type of chromosomes – either from the father or the mother – is pretty much equally likely as any other. You would have to say that the degree of uncertainty inherent in meiosis is very high. Moreover, any function that purports to measure the degree of uncertainty of meiosis as a function of a probability distribution that models meiosis had better produce a relatively high value result. If not, then we would not regard such a function as a very good measure of uncertainty.

On the other hand, the process of DNA replication had better produce a very low degree of uncertainty – because this process should work the same way almost every time. That is, there should be very few mistakes (copy errors) in this process. Of course, there had better be at least *some* copy errors in DNA replication, or otherwise there will be no mutations – and therefore no diversity of species, and therefore no natural selection and therefore no evolution!

Consequently, any function that purports to measure the degree of uncertainty of DNA replication as a function of a probability distribution that models DNA replication had better produce a relatively low value result. The result should be a little larger than zero, since DNA replication - as we just explained - does exhibit *some* uncertainty. If not, then we would not regard such a function as a very good measure of uncertainty.

So, we are looking for some function to measure the degree of uncertainty of an experiment (system process). This function should take a probability distribution as its input, and produce some number (representing degree of uncertainty) as its output. Moreover, the probability distribution used should be one that models the experiment (system process) whose degree of uncertainty we are trying to measure.

Shannon's Measure of Uncertainty

Fortunately, Organodynamics does not have to invent such a measuring function because one already exists! Claude Shannon at Bell Telephone Laboratories contrived it and published his results in 1948 [Shannon 1953].

Shannon gave the name *entropy* to his measure of uncertainty. He selected this name because of the similarity of his function with another function of the same name developed by Gibbs [Gibbs 1902]. Essentially, Shannon set the value of a certain arbitrary constant in Gibbs function to 1 in order to arrive at his formula.

Whether or not Gibbs intended for his entropy function to measure uncertainty – or some other property – is a discussion for a later time. Nevertheless, Shannon makes it clear that his version of *entropy* does, in fact, measure the uncertainty of a finite sample space probability distribution.

In the present text, the use of the word *entropy* refers to Shannon's version along with its intention to measure the uncertainty of a system process (or experiment) by inspecting a probability distribution which models that process. To avoid any confusion, we shall occasionally use the phrases *Shannon's entropy*, or *information entropy* to distinguish it from other entropy measures.

We shall delve into Shannon's entropy measure in some depth in a later subsection. For now, let it suffice to provide a brief prosaic definition of it:

Shannon's entropy: A measure of the uncertainty inherent in a probability distribution. Entropy is calculated as the expected value ("average") of the logarithms of the reciprocals of the probabilities of the sample events of the distribution.

In other words, to calculate the *entropy* of a finite probability distribution, first calculate the reciprocals of the probabilities of each of its sample points. Next, calculate the logarithm (base 2) of each of those reciprocals. Thirdly, multiply each of those answers by its probability, resulting in a set of products. Finally, add all of those products together. (The last two steps constitute calculating the average.) This final sum is the *entropy* of the distribution (and of the underlying process for which it is a stochastic model.)

Note that any logarithm base will work properly to calculate entropies, as long as the same base is used consistently throughout. Shannon used a base of 2, as shall we.

There is a shortcut entropy formula that is most often referenced in texts. It involves negative logarithms. We shall discuss this and other aspects of Shannon's entropy – including *why* it is a measure of uncertainty - at a later time. For now, suffice it to say that 1) there already exists a widely accepted formula for calculating the uncertainty of a probability distribution – Shannon's entropy, and 2) we know how to calculate it.

Derivation of Shannon's Entropy Function

Shannon's entropy function can at first appear somewhat mysterious. And it is generally not intuitive as to why such an expression should be accepted as a measure of uncertainty. However, it turns out that the formula is built upon a simpler principle – the measure of the uncertainty of a single event.

It also turns out that this measure for a single event can be explained in a manner that can appeal to intuition. Fortunately, Shannon's entropy formula combines this measure for all of the events of a distribution and produces an integrated result.

In this chapter, we shall explain the simpler concept and show how it is combined to create Shannon's entropy formula described in the previous subsection.

For readers not interested in an intuitive explanation of why Shannon's entropy is a measure of the uncertainty inherent in a probability distribution, please proceed to the next section entitled "Choice, Uncertainty and Realization in Organodynamic Models".

Considering a Measure of the Uncertainty of a Single Event

In Part I, we put forth the idea that "uncertainty is inversely related to probability". Mathematically, this means that a measure of the uncertainty of an event E should be inversely proportional to its probability, Pr(E): the lower the probability, the higher the uncertainty. This is equivalent to saying that *the uncertainty of an event is proportional to the inverse of its probability*.

So, if our measure of the uncertainty of an event E is named U(E), then U(E) should be a function of 1/Pr(E).

First Attempt at Defining a Measure of Uncertainty of a Single Event

So, a first attempt at inventing U(E) – lets call it U'(E) - would be:

$$U'(E) = 1/\text{Pr}(E).$$

When we define a measuring function, we are generally free to define it how we like. On the other hand, there are certain properties that we want our measuring functions to have. Let's look at a couple of these desired properties to see how our candidate U'(E) "measures up" to these criteria.

First, there is "positive monotonicity". This means that, as the phenomena being measured gets bigger, so does so its measurement. This is true of U'(E). As the uncertainty of an event increases, its probability decreases and U'(E) increases. And this is what we want. So U'(E) passes the positive monotonicity test.

This is a good start. But, unfortunately, this function will not pass our next test. It does not have a second property that we want measuring functions to have: *additivity* – at least for independent events. Additivity for *events* means that the

measure of a compound event is the same as the sum of the measures of the individual events.

Of course, we don't want to be too strict with our criteria. We would not expect additivity to hold for events that depend on each other. But if two events are statistically independent, then we would like for the uncertainty inherent in their occurring together would be the same as the sum of their occurring alone.

For example, if "rolling a die" has a degree of uncertainty of, say "x", and "flipping a coin" has a degree of uncertainty of, say, "y", then an experiment that considers the joint outcomes of simultaneously "rolling a die and flipping a coin" should exhibit a degree of uncertainty of "x+y". That is, at least for two events that are statistically independent, we would like for the measure of uncertainty of their events taken individually to sum to the measure of the uncertainty of their events occurring jointly.

Mathematically, the *additivity* of $U'(E)$ for independent events, if true, would mean that, for independent events E_1 and E_2 , that $U(E_1 \wedge E_2)$ has the same value as $U(E_1) + U(E_2)$. Lets see if this is true by calculating them both and comparing the results.

First, lets calculate $U'(E_1 \wedge E_2)$:

$$U'(E_1 \wedge E_2) = 1/\text{Pr}(E_1 \wedge E_2) = \\ 1/(\text{Pr}(E_1) * \text{Pr}(E_2))$$

Then we'll calculate $U'(E_1) + U'(E_2)$:

$$U'(E_1) + U'(E_2) = 1/\text{Pr}(E_1) + 1/\text{Pr}(E_2) = \\ = (\text{Pr}(E_2) + \text{Pr}(E_1)) / (\text{Pr}(E_1) * \text{Pr}(E_2))$$

Note that the two final lines of the above are not equivalent for all independent events E_1 and E_2 . This demonstrates that our first attempt $U'(E)$ is not additive for independent events.

Therefore, $U'(E)$ is a failed attempt at inventing a measuring function for the uncertainty of an event that has all of the properties that we would like our measuring function to exhibit.

Consequently, we must look further for another such function.

Successful Attempt at Defining a Measure of Uncertainty of a Single Event

In this subsection, we shall "repair" our first attempt at inventing a measure of uncertainty of a single event so that it is additive for independent events. It is permissible to do this because we can make up our measuring function any way that we wish – as long as it follows reasonable expectations for measuring functions. And one of those expectations is that the function be additive – at least for independent events.

Anyway, the "repair" is very simple:

$$U(E) = \log(1/\text{Pr}(E)).$$

Notice that we simply took the logarithm of the first attempt. Of course, we have not specified which base to use for the logarithm. This is because it doesn't matter. Any base will work as long as it is used consistently. (In practice for the remainder of this text, we shall use a base of 2, as does Shannon.)

We must demonstrate that this function is additive for independent events. This is demonstrated by showing that, for all independent events E1 and E2, that:

$$U(E1 \wedge E2) = U(E1) + U(E2).$$

To show this, we shall calculate both expressions and show that they must be the same for all independent events:

First, we calculate $U(E1 \wedge E2)$:

$$\begin{aligned} U(E1 \wedge E2) &= \\ \log(1/\text{Pr}(E1 \wedge E2)) &= \\ \log(1) - \log(\text{Pr}(E1 \wedge E2)) &= \\ 0 - \log(\text{Pr}(E1 \wedge E2)) &= \\ - \log(\text{Pr}(E1 \wedge E2)) &= \\ - \log(\text{Pr}(E1) * \text{Pr}(E2)) &= \\ - (\log(\text{Pr}(E1)) + \log(\text{Pr}(E2))) & \end{aligned}$$

Next, we calculate $U(E1) + U(E2)$:

$$\begin{aligned} U(E1) + U(E2) &= \\ \log(1/\text{Pr}(E1)) + \log(1/\text{Pr}(E2)) &= \\ (\log(1) - \log(\text{Pr}(E1))) + (\log(1) - \log(\text{Pr}(E2))) &= \\ (0 - \log(\text{Pr}(E1))) + (0 - \log(\text{Pr}(E2))) &= \\ (- \log(\text{Pr}(E1))) + (- \log(\text{Pr}(E2))) &= \\ - (\log(\text{Pr}(E1)) + \log(\text{Pr}(E2))) & \end{aligned}$$

So, the final lines of the above two calculations are the same. Thus, $U(E) = \log(1/\text{Pr}(E))$ is additive for independent events.

More Required Properties

Besides additivity, there are two more properties that are generally required of measuring functions, and we must ensure that our repaired measure of uncertainty, $U(E)$, exhibits those two as well.

We shall leave the demonstration of these to the reader as exercises. However, they are straightforward, if not trivial.

The first of these is the *zero property*. This means that if an entity exhibits none of the property being measured, then the measuring function should assign a value of zero to that entity. In the case of $U(E)$, an event that exhibits absolutely *no uncertainty*, is one that is absolutely certain – and therefore has a probability

of one. The exercise for the reader, then, is to prove that if event E has probability 1, then $U(E) = 0$.

The other required property for a measuring function has already been discussed above – *positive monotonicity*. The reader is invited to prove that $U(E)$ is positively monotonic – also called *monotonically increasing*.

The Uncertainty of an Event - Simplified

This formula can be simplified to produce a format that is most often seen in texts. The derivation of this common simplification is:

$$\begin{aligned} U(E) &= \log(1/\text{Pr}(E)) = \\ &= \log(1) - \log(\text{Pr}(E)) = \\ &= 0 - \log(\text{Pr}(E)) \end{aligned}$$

Therefore, $U(E) = -\log(\text{Pr}(E))$.

The Determination of Shannon Entropy

Our function $U(E)$ is the measure of the uncertainty of a single event.

We have mentioned above that Shannon's entropy measures the uncertainty of an entire probability distribution (as well as the underlying phenomena for which it is a stochastic model).

The question is, how do we get from the measure of the uncertainty of a single event – our $U(E)$ function – to the uncertainty of an entire probability distribution?

Of course, a probability distribution is comprised of an entire set of sample events. So the above question resolves to:

“How do we apply $U(E)$ to every event of a probability distribution in order to arrive at a measure of the uncertainty of the distribution as a whole?”

The answer to this question is:

“Take their average”!

(Isn't that what we normally do with a sample of entities? For example, if a student has a set of grades from a course of study, we normally calculate a single grade for the student by taking the average of the set.)

Here, then, is how to take the average of the $U(E)$ values of all of the sample points of probability space:

Let A be a finite probability distribution with n sample points $E_1, E_2, E_3, \dots, E_n$. So, the expression for their average $U(E)$ values, or *mean*, named “ $H(A)$ ”, is:

$$H(A) = \sum_{i=1}^n \Pr(E_i) * U(E_i),$$

For each sample point E_i in the sample space of the distribution.

This expression can be filled in by substitution of $U(E_i)$, as follows.

$$H(A) = \sum_{i=1}^n \Pr(E_i) * U(E_i)$$

$$H(A) = \sum_{i=1}^n \Pr(E_i) * (-\log(\Pr(E_i)))$$

$$H(A) = -\sum_{i=1}^n \Pr(E_i) * \log(\Pr(E_i))$$

The reader may recognize the final line above as the formula for Shannon's entropy of a probability distribution!

Thus, an intuitive interpretation of *entropy* is that it is the *mean uncertainty* of the sample events of a probability distribution.

Moments of the Information Random Variable

Yes, Shannon's entropy is the mean of a function on the sample space of a probability distribution. More precisely, entropy is the *mean* of a specific *random variable* of a probability distribution. In this case, the random variable we used to calculate the mean has been the function $U(E)$.

Thus, Shannon's entropy is the mean – or *first moment* – of the random variable $U(E)$.

(An introduction to random variables, their meaning and their moments is beyond the scope of this text. However, for those readers who are knowledgeable concerning them, the present subsection may be interesting. Other readers may want to refer to the introduction section of [Doob 1953].)

But, $U(E) = -\log(\Pr(E))$ is the measure of *uncertainty* of an event. As we have seen in Part I, $U(E)$ is also the measure of *information* of an event – according to Shannon and to contemporary Information Theory.

Thus, if we were to give this random variable a name, the phrase "Uncertainty Random Variable" would be suitable. But so would the phrase "Information Random Variable". Let's choose the latter.

Thus, $U(E) = -\log(\Pr(E))$ shall henceforth be referred to as the *information random variable*, or *IRV*, of a probability distribution.

Moreover, the first moment of the IRV is Shannon's entropy – as we have shown above.

And, being a random variable, $U(E)$ has a countable sequence of other *moments*, and also a countable sequence of *central moments*. In mathematical statistics, it is a standard technique to use the sequence of central moments of a probability distribution as a characterization of that distribution.

This set of central moments (the first of which is *entropy*), then, is a useful mathematical characterization of their random variable, the IRV.

Choice, Uncertainty and Realization in Organodynamic Models

In Part I, after surveying some classical and contemporary observations of Information Theory, we concluded that

Choice *is* uncertainty. And uncertainty becomes manifest – *realized* – as *information*.

So we have a time sequence established here. First, there is *choice* and concomitant *uncertainty*. Subsequently, at some point when “the die is cast” or “the coin is flipped”, this *uncertainty* becomes *realized* as *information*.

Therefore, there is a time sequence that is important to modeling practices. Pertaining to the dynamical processes that we are modeling, this time sequence, then, directs how we as modelers need our models to unfold in time:

- Firstly, to first represent the *choices* provided by a process,
- Secondly, represent our *uncertainties* regarding which of those choices will be realized, and
- Thirdly, represent the outcome (information) revealed once that realization has obtained.

We shall ensure that Organodynamics provides a modeling program – or *modeling methodology* - that provides these three steps, or types of models. A program will be presented in which the modeler develops three levels of model, each representing one of the three stages indicated above.

The first stage will progress into the second, and the second into the third as the amount of information is increased gradually during the modeling process. This modeling practice, which is directly supported by the Organodynamics methodology, is detailed below in the next chapter, entitled “The Uncertainty Model Gradient”.

System Dynamics as Uncertain Reorganization

We have seen above that the dynamical aspects of Organodynamics focuses on a set of choices, each of which are possible states that a system can obtain at any specific time step within the process.

As a dynamical system, Organodynamics is particularly interested in characterizing which of these possible system states will be manifest, or realized, at the next time step in the process.

Because of the sixth OCS organizing principle, Organodynamics is especially interested in the uncertainty associated with which state will be manifested at the next time step. Therefore, Organodynamics is going to model the dynamical state change between the current step and the next in terms of this uncertainty.

And, as we have already discussed at length, in Information Theory *uncertainty* is defined in terms of probabilities. Thus, we are going to use a probability

distribution to describe the dynamics between each of the contiguous time steps of our process. Specifically, in Organodynamics we are going to use a probability distribution to model the *change-of-state* between any two contiguous time steps in organic processes.

But recall that in Organodynamics these states whose change we are modeling have been selected to be *system organizations*.

In other words, these *choices* whose *states* and whose *uncertainty* we are concerned about in Organodynamics are not simple states like “heads or tails”. Rather, they are complex states in their own right. Specifically, for each time step in a process, they are a *system organization*. And, we have a well-defined articulation of these *organizations* as sets of ordered pairs (duples) of the components of an underlying system.

In other words, our “choices” for a time step, the events of the sample spaces of our probability distributions, are complex things – system organizations, all of the same underlying system.

One thing to note here is that we are combining the notions of *organization* and *uncertainty* together in the same space. In this space we have the *uncertainty* of *which organization* of an underlying system becomes manifest at the next time step.

Thus, our probability space is a *compound space*. We are modeling the *uncertainty* of the occurrence of various *organizations* at the next step in time.

Organodynamic Probability Spaces and Distributions

Thus, the type of probability space that is central to our modeling framework is a special kind of distribution whose sample space is special.

This probability space is special in that:

1. It is based on the population of an underlying system.
2. Its sample space is a compound space based in on that population.
3. Specifically, it is the set of all *system organizations* that can be formed using the population of that underlying system.

We shall name this special probability space the *organodynamic probability space of a system*. We shall call its associated probability distribution the *organodynamic probability distribution of a system*.

Thus, we have the following definition:

Definition: the organodynamic sample space of a system:

Let $S = (P; O)$ be a system. Consider the set $O' = \{O'' \text{ such that } O'' \text{ is a possible organization of } P\}$. Then, O' is called the *organodynamic sample space of S*.

Definition: the organodynamic probability distribution of a system:

Let $S = (P; O)$ be a system. Let F' be any probability distribution on sample space O' , where O' is an *organodynamic sample space* of S . Then F' is called an *organodynamic probability distribution of system S*.

Organodynamic probability distributions are also known simply as *organodynamic distributions*.

The Organodynamic framework is primarily concerned with *organodynamic probability distributions of systems*.

In fact, all major constructs of the framework that are defined in the remainder of this text use *organodynamic probability distributions* as their building blocks.

These constructs include: organodynamic processes, segments, edges, nodes, Organodynamic graphs and all three forms of Organodynamic webs. The remainder of Part II of this text develops these constructs and discusses how they exhibit various of the seven OCS organizing principles.

Degrees of Uncertainty in Organodynamic Distributions

And, between any two time steps of a system process, the *degree of uncertainty* can either be high or low; and the *degree of organization* can either be high or low – and these two degrees can vary independently. That is, the amount of uncertainty and the amount of organization that is going on can both increase, can both decrease, or either can increase while the other decreases. We shall see that is matter is significant and we shall revisit it later.

On an historical note, it is interesting that statistical mechanics enjoys a similar situation. It deals with a compound space in which it models the uncertainty of various “configurations” (positions and momenta) of ideal gas molecules being manifest at certain point in time. Like *system organizations* in Organodynamics, these position and momenta situations represent system state at a moment in time. And it is the change of these system states (like *organizations*) that define the dynamical aspects of statistical mechanics.

The point is that, like Organodynamics, statistical mechanics deals with a compound space in which it is primarily interested in *uncertainty*, but it is the uncertainty of a complex state – a state that exhibits varying degrees of *organization*. This situation makes it tempting to confuse the notion of *uncertainty* and the notions of *organization*. And, tempts its participants into using terminology like “order” and “disorder” – which are sometimes used to convey “organization/disorganization”, and at other times used to convey “certainty/uncertainty”. More properly, these terms allude to “the uncertainty of organization/disorganization”.

Introducing Stochastic Processes in Organodynamics

We have discussed *system processes* and defined them as time series whose individual terms are the *states* of the system being modeled at specific time steps. We described system process as a deterministic model of systems, because the model represents complete certainty in its description of system state at each time step.

We have also discussed *stochastic processes* as an “upgrade” of a *system process* – an upgrade that adds some degree of uncertainty to the determinism of a *system process*. In many modeling situation, especially complex systems, such uncertainty is more realistic than a pretense that the underlying system dynamics is deterministic.

How does a modeler go about promoting a *system process* to a *stochastic process*? This section addresses this issue.

Promoting System Processes to Stochastic Processes

To promote a system process to a stochastic process, we must somehow associate probabilities to each of its time steps. That is, where those steps are now deterministic, we must make them probabilistic. Of course, what we are trying to accomplish is to reflect the uncertainty that is inherent in an underlying dynamical system that we are trying to model.

The way that we generally go about adding *uncertainty* to a system process in order to promote it to a stochastic process is to 1) add choices, and then 2) add probabilities to those choices. Let’s look at each of these enhancements.

Adding Choices

The *system process* model provides only one “choice” for each of its time steps. Thus when promoting it to a *stochastic process*, we shall add more choices to each of its time steps.

However, since we are attempting to capture uncertainty in our stochastic model, it can be argued that the notion of uncertainty make more sense when we are discussing the possible outcome of “the next time step” after the current time step.

Therefore, we are going to make an additional change as compared to what we do in a system process. That is, we are going to stipulate that the *system states* whose *possible choices* we are enumerating are possible system states that can be manifest at the *next time step*.

Adding Probabilities

In order to be precise about these choices, we must also assign probabilities to each of them. Thus we shall go further and assign probabilities to each of these newly added choices.

The usual approach for assigning probabilities is to develop them empirically from the underlying system that is being modeled. Sometimes, in research environments, these probabilities can be assumed to be defined by a specific theoretical distribution.

The Result

The result is that each time step of a stochastic process is represented by a *probability distribution* whose *sample space* is the set of choices now associated with the next time step. Understand that this probability distribution defines the state of the *current time step*, but the probabilities are “making predictions” regarding which of these possible states will be realized at the *next time step*.

We shall define such probability distributions to be our representation of *the state of the dynamical system* at that time step. Thus, the stochastic process that we shall use to model dynamical systems in Organodynamics shall be a sequence of probability distributions, each representing the *stochastic state* of the underlying dynamical system being modeled.

Promoting System Processes to an Organodynamic Processes

In Organodynamics, we utilize the two ideas of *system process* and *stochastic process* to a great extent.

However, we also do something very specific to these two ideas to make them into the process structure that we need in order to create the structure that we shall need going forward from here.

The specific thing that we do is: We define very precisely what is meant by *system state* in Organodynamics. This has been mentioned before, but we need to emphasize it again – because this is where it fits into our system and stochastic processes.

System state in Organodynamics is defined as:

System organization: a set of sets of ordered pairs (“duples”) of system components.

So, above, when we defined *system process* and *stochastic process*, we were intentionally vague about what we meant by the notion of *system state*.

But in Organodynamics, we get very specific about system state. Not only that, but we also get complicated! System state in Organodynamics is not simple. The whole idea is to define *system state* in Organodynamics as “the way that the set of components of the system are currently organized”. As indicated in the above description of system organization, it is defined using set theory as a set of sets of duples of components of the population of the system.

In fact, everything that we do throughout the following text will depend upon an understanding that Organodynamic system state is this complex thing: this set

of sets of duples. Every structure that is defined will, at its heart, be this structure.

Thus, what we are going to accomplish in this section is to put specificity to our *system process* concept and to our *stochastic process* concept by specifying that their *system states* be *system organizations* – that is, sets of duples of underlying system components.

In fact, once we put that specificity to these two concepts, we shall promote them to constructs with new names: *organic processes* and *organodynamic processes*.

Promoting System Processes to Organic Processes

We have defined a *system process* as a sequence of system states, all states of the same underlying system.

In Organodynamics, we have defined *system state* to be a *system organization*, which is a set of sets of ordered pairs (called *duples*) whose members are related components of the population of a system that is being modeled.

In Organodynamics we have already defined a special kind of system process, called an *organic process*, whose *states* are *system organizations*. Thus, an *organic process* has been defined as a sequence of *system organizations*, each of which represents the state of an organic system at a specific time step.

We also have a need to do the same kind of thing for *stochastic processes*. That is, we need to define a special kind of stochastic process whose sample spaces are *system organizations*. We shall name this type of stochastic process an *organodynamic process*.

Thus, an *organodynamic process* is defined as a sequence of probability distributions, one for each time step in the sequence. Moreover, the *sample spaces* of each of these distributions is a set of *system organizations* of an underlying system that is specific to that time step.

Thus, in Organodynamics, an *organic process* is the special case of the more general *system process* where state is defined to be a system organization. And, an *organodynamic process* is a special case of the more general *stochastic process* where state is defined to be a probability distribution whose sample space is a set of system organizations.

Whereas an *organic process* is a deterministic model of an organic dynamical system, an *organodynamic process* is a stochastic model of the organic dynamical system.

In Organodynamics, we model organic systems as *organodynamic processes*. However, we may begin with a simpler model and then eventually promote it to an organodynamic process. There are a couple of ways to do this. Both start with developing a system process model of the target dynamical system.

One of these approaches first promotes the system process to an organic process; and then converts the organic process to an organodynamic process. The second approach first promotes the system process to a stochastic process; and then promotes the stochastic process to an organodynamic process. Let's look at one of these approaches in the next section.

Promoting an Organic Process to Organodynamic Processes

This subsection will provide some detail to the first approach discussed above for starting with a simple process and promoting it to an organodynamic process.

This approach, like the other one, begins with a system process model of a target system. The next step is to promote that simple model to an organic process. The next step after that is to promote the organic model to an organodynamic model. We shall describe this approach in a little more detail here.

The first step of this exercise begins with a system process model of the dynamical system that we are modeling. Such a model is a deterministic model of the process we are modeling. For example, if the target process were, some well-known biochemical process, let's say the Krebs cycle, then the *system process model* would be a simple sequence whose time steps would each name the proper molecule for the biochemical process being modeled.

This system process model would be a deterministic model of the Krebs cycle. As such, it would be a sequence of the molecules in the Krebs cycle. Moreover, after the proper number of time steps, the sequence of molecules in this case would start to repeat, because this example is a cycle.

The second step is to promote this *system process* to an *organic process*. Recall that an *organic process* is special case of a system process where the terms of the sequence are articulated as *system organizations*. This means that we must describe each of the molecules of the Krebs cycle as an *organization of atoms* – and then figure out how to express this organization as a *set of duples*.

We have and shall discuss how to do this at length elsewhere. Then suffice it to say for now that such a set of duples is capable of representing which atoms bonded with which others to form the molecule being described.

Once this replacement of the molecule names in the system process has been made then the result is a sequence of *system organizations*, each of which is a more detailed representation of the molecules of the Krebs cycle than is a sequence of mere molecule names.

And, this new sequence is now an *organic process*. Note that it is still deterministic. And, this completes our first step: We have promoted a (mere) system process model of the Krebs cycle to an organic process model.

The third and final step is to promote this *organic process model* of the Krebs cycle into an *organodynamic process model*.

To do this, we want to end up with a probability distribution for every time step. Right now, we don't have that. Instead we have a single *system organization* at each time step. And that system organization represents a detailed description of the inter-atomic bonds that describe the molecule at that time step.

In order to promote each of these steps we must do two things: 1) add more choices, and 2) assign probabilities to each of those choices.

But, "where are we going to get additional *system organizations* for each of these time step?" you may be asking. In order to ask this question you must put yourself into a more realistic frame of mind that are the text books that teach the Krebs cycle. Those text books assume that one molecule of the Krebs cycle must always lead to the next molecule of the Krebs cycle.

In other words, the textbooks assume that, given that we have a specific molecule of the Krebs cycle, that the necessary chance phenomena will *always* occur that results in the desired chemical reactions (desired by the authors of the text book) so as to produce the next molecule depicted in the Krebs cycle.

But this simply is not true. The Krebs cycle often begins with the "right" sub-sequence of molecules, but then "abandons" it in the middle of the process. This is not a matter of intension on the part of the molecules in the process. Rather, it is a matter of *chance encounter*, of probability.

The fact is that each step in the Krebs cycle has many possible pathways that it can proceed – all with some probability (including the "correct"). The step we are not taking becomes aware of this fact. And, at this point it is necessary to identify all of these possible pathways that can be taken from the present molecule that we are at for the time step.

This is the first thing we must do at this step, described above as "add more choices". To do this, for each time step, we must identify the Krebs cycle molecule at that time step, and then determine all of the chemical reactions involving that molecule that could possible ensue, and what the chemical result would be.

One possibility is that the product is the correct result for the Krebs cycle. But there will usually be other possible products as well – depending upon which other compounds and catalysts probabilistically become available in the chemistry of the cell at that time.

All of the possibilities must be articulated as *system organizations* for the time step. As such, they become the sample space for the time step. The second thing that must be done is to assign probabilities to each of these system organizations. Then, you will have the desired probability distribution for the time step.

When this is done for all of the time steps of the Krebs cycle, then you will have the desired *organodynamic process model* for the Krebs cycle. And you will be done.

The Result

Going forward in this text, virtually all of our development below will be built upon the *organodynamic process*. It will lie at the very heart of all substructures that we shall build up to become the Organodynamic web, that we have described elsewhere as the grand scheme of Organodynamics.

Strongly stated, the *organodynamic process* is a fundamental *dynamical* entity of Organodynamics. In the language of our electronic circuit toolkit analogy, the *organic process* in Organodynamics is analogous to the “wires” in the electronics circuit toolkit.

We shall actually have little use for the notion of *organic process* going forward. For the most part, we have used the concept of *organic process* as a context for developing and understanding *organodynamic processes*.

Thus, it is recommended that the reader make sure that an understanding of the *organodynamic process* is grasped at this point in the text. The example that we are incrementally building that is presented toward the end of this chapter should help in this regard.

Modeling Dynamics with Probabilities and Uncertainty

It is important to understand that the uncertainty inherent in the probability distribution at each step of our *organic process* exists at a *current step* of the process regarding the outcome of the *next step* of the process. In other words, these probability distributions model the *uncertainty of change* within our organic processes between the current step and the next.

Thus, in Organodynamics we are going to use *probability* to model *change*, or *dynamics*.

From this it follows that any *measure* of the uncertainty inherent in one of these probability distributions is also a description of a particular aspect of the dynamics of the process. Specifically, such a measure of uncertainty describes the unpredictability dynamics between a current and the next step of a process.

As we shall see later in this text, a central interest regarding system dynamics in Organodynamics is how much uncertainty is at work in various parts of an Organodynamic web. Since a network of several organodynamic processes constitutes any Organodynamic web, we shall find that we can characterize the uncertainty of an entire Organodynamic web by collectively determining the uncertainties of its constituent organodynamic processes. We shall have more to say about how to do this later.

In any event, fundamental to any of this is the calculation of the *degree of uncertainty* of the constituent probability distributions of any of our structures. We have already seen above that this calculation is accomplished by using Shannon's formula for *entropy*. The result will be a set of entropy values that characterize the degrees of uncertainty throughout that Organodynamic web.

This will enable us to ascertain the various degrees of uncertainty exhibited across various aspects of a specific Organodynamic web, as required by the sixth organizing principle, in order for it to be considered to be "lifelike".

Of course, we have yet to describe what an Organodynamic web looks like – and shall spend the remainder of this text doing so. But, from what has been said, the reader can assume that it has a lot of "time steps" and they each have a probability distribution whose entropy can be calculated individually. Collectively, however, the sixth OCS principle says that these measure of uncertainty will vary across a single Organodynamic web.

This is the reason we have been so precise above in providing a mathematical definition of *uncertainty* in the form of Shannon's entropy. When we finally get around to defining the mathematical structure of Organodynamic webs, we shall visit how to apply entropy to the characterization of the uncertainty exhibited by an Organodynamic web.

<grant – Characterizing the uncertainty of an Organodynamic web.> As we shall see in a later chapter, the approach taken will be to treat this collection of entropies of the probability distributions involved in a single Organodynamic web as a random variable, and then characterize that random variable by its moments.

Piecewise Homogeneity and Segments in Organodynamics

In what we have developed so far, each time step in one of our *organodynamic processes* has its own unique probability distribution. However, it would simplify matters greatly if all of the time steps of a specific organodynamic process shared the same distribution.

If we can use the same probability distribution for all of the steps in an organodynamic process, then the process is called *time-homogeneous*, or simply *homogeneous*.

[Note: The attribute *time-homogeneous* as applied to a stochastic process is also referred to as *stationary*. However, the term *stationary* has a second distinct usage within the theory of stochastic processes – as usage that we shall have much need of later. Therefore, in order to reduce confusion, we shall not use the term *stationary* to mean *time-homogeneous*; and reserve its usage to the second meaning that shall introduce later.]

In the modeling of dynamical systems in science and engineering, we shall find that many organic processes are homogeneous. This is fortuitous indeed.

There is a rich body of mathematics to assist in dealing with homogeneous stochastic processes that can be brought to bear for homogeneous organodynamic processes.

However, many organodynamic processes are not homogeneous. In the most ill behaved case, the probability distribution is unique for every time step in the process.

But there is an intermediate case. Quite often, organic processes are *piecewise homogeneous*. That is, they are homogeneous for some number of time steps and then the probability distribution changes. Then they are subsequently homogenous for some more steps, and then the probability distribution changes again.

Piecewise-homogeneity enables modelers to partition an organodynamic process into its consecutive homogeneous pieces and apply the homogeneity theory to those in isolation. This ability is, in many ways, analogous to “linearity” in the theory of functions. But, overall, the process is still nonhomogeneous, so the technique has its limitations.

It seems that piecewise-homogeneity is quite frequent for biological systems. We shall take this further and assume that it is very frequent for organodynamic processes as well.

The bad news is that even if they are piecewise homogeneous, stochastic processes are still nonhomogeneous overall. But the good news is that there are whole stretches of steps that *are* homogeneous, and are therefore relatively tractable.

We shall use the name *segments* to refer to these homogenous subsequences within these piecewise-homogeneous chains; and we shall use the term *edge* to refer to a piecewise-homogeneous sequence of time steps. Thus, an *edge* can also be understood as a *sequence of segments*.

In Organodynamics, we shall mostly deal with piecewise homogenous finite stochastic processes, or *edges* – which can also be treated as sequences of homogeneous *segments*.

Simple Non-Biological System Example of an Organodynamic Process

The poker and Scrabble® game examples that we have been using provide good and comparatively simple examples of complex, but non-biological, webs of processes and transforms. And we shall use them again in a later chapter where we develop the stochastic versions of these transforms. However, in those two examples, the segments lengths are too short (typically one time step) for them to be typical examples of segments or of edges.

Therefore, for this chapter we shall use an alternative example simple non-biological system – the Diner’s Travel Club. We shall use this example twice in this chapter – once here to exemplify a segment of an organodynamic process, and later to exemplify a conditional stochastic process (Markov chain).

For an annual fee, members of the Diner's Travel Club can enjoy one excursion per month. Each excursion flies the member and one companion to three US cities in four days and three nights for dinner and a hotel service. These excursions are offered in six packages.

The cities involved in the program are New Orleans (No), Miami (M), San Francisco (Sf), Los Angeles (L), New York (Ny), Chicago (C), Kansas City (K) and Seattle (S). The travel year begins in December.

The six packages that are offered are:

- Miami to New Orleans to Kansas City (MNoK)
- Los Angeles to San Francisco to Seattle (LSfS)
- New York to New Orleans to Miami (NyNoM)
- San Francisco to Seattle to Chicago (SfSC)
- Miami to New York to San Francisco (MNYsf)
- San Francisco to New York to Miami (SfNYM)

After keeping records for a number of years, the club has found that the travel patterns during any of the four calendar seasons are quite stable. However, each of the seasons exhibits a distinctive travel pattern in each of the four seasons.

Therefore, travel in each of the four seasons is reasonably modeled via its own distribution describing the probability of traveling to each of the destinations during its season. The four probability distributions are:

Winter Travel Probability Distribution

Trip:	MNoK	LSfS	NyNoM	SfSC	MNySf	SfNyM
Prob:	.20	.15	.15	.15	.05	.05

Spring Travel Probability Distribution

Trip:	MNoK	LSfS	NyNoM	SfSC	MNySf	SfNyM
Prob:	.20	.15	.10	.15	.10	.05

Summer Travel Probability Distribution

Trip:	MNoK	LSfS	NyNoM	SfSC	MNySf	SfNyM
Prob:	.30	.10	.20	.10	.15	.05

Fall Travel Probability Distribution

Trip:	MNoK	LSfS	NyNoM	SfSC	MNySf	SfNyM
Prob:	.15	.05	.20	.20	.25	.10

Since travel occurs once per month, then there are 12 time steps in this organodynamic process per year. And, given that it is an organodynamic process (as we have described one so far), each of these time steps is a probability distribution. If we name the four probability distributions by the names "Winter", "Spring", "Summer" and "Fall", then the first 24 time steps of this one are:

Winter, Winter, Winter, Spring, Spring, Spring, Summer, Summer, Summer, Fall, Fall, Fall, Winter, Winter, Winter, Spring, Spring, Spring, Summer, Summer, Summer, Fall, Fall, Fall...

There are number of statements that we can make about this organodynamic process:

- It is an organodynamic process because the entities that comprise each of its time steps is a probability distribution whose sample space is a set of alternative organizations of an underlying sample space. The underlying sample space for all 24 distributions is the set of eight cities. The organizations of these cities that form the sample space is the six excursion packages.
- It is homopopulous. The underlying system of each time step of this stochastic process has a population whose components are the eight cities. This population never changes throughout.
- It is homo-organizational. The six excursion packages are each an organization of the population of eight cities. This collection of organizations forms the sample spaces of each distribution of each of the 24 time steps depicted.
- It is nonhomogeneous. This is because not all of its probability distributions are the same. Even though the sample spaces are the same, the probabilities change every three time steps.
- Time steps zero through two is a segment, because they are a contiguous set of time steps whose probability distributions remain unchanged.
- For any n , time step $0 \cdot n$ through time step $0 \cdot n + 2$ is a segment for the same reason.
- Time steps 0 through 23 form an *edge*, and they do not form a *segment* because as a sequence they are nonhomogeneous.

Introducing Conditional Stochastic Processes

Sometimes it is desirable to model a process as stochastic, but we don't have quite enough information for each time step to be able to define its probability distribution.

However, there are times when we do have quite a bit of information for certain specific conditions, if not for the whole picture. In fact, sometimes we even have an entire probability distribution at a time step for certain conditions – if not for the whole picture.

Example Conditional Stochastic Process

Let reuse the Diner's Travel Club to provide an example of a conditional stochastic process.

In order to convert the scenario into one that involves a conditional process, we must add a piece to it that provides information about how the probabilities are influenced by what has occurred recently. At the same time, however, we are going to take away some information that we had the first time around.

The information that we are going to take away for this example is the probability distributions for each trip. You may recall that four probability distributions were used, because, each season the travel pattern changed, and this fact was reflected in a change in the probability distribution.

But we are going to add some conditional information. At each time step (month), several conditional probability distributions will be provided – each predicting which excursion package will be selected for the next month depending upon which excursion package was selected for this month.

In fact, for each time step, there will be one conditional probability distribution for each excursion package. And each of these will have a probability for each of the excursion packages. Thus there will be six conditional probability distributions for a given time step, and each one will provide six probabilities.

Another change that we shall make (to make this example simpler) is that we shall declare that these probabilities are the same for all twenty-four time steps. In other words, this Markov chain is homogeneous.

Conditional Probability Distributions for the Diner's Travel Club Example

Above we established that at each time step of this conditional probability process, there are six possible excursions that a club member could have selected for that *current time step*.

And, for each of these six packages, he will make a selection from these same six packages for the next excursion package to be taken at the *next time step*. Moreover, for whichever package he has selected for the current time step, the probabilities that he will select each of the six packages for the next time step *depend upon* – or are *conditioned upon* – which package he selected for the current time step.

This means that each of the six possible *current conditions* must have its own conditional probability distribution – each of which has a sample space with six sample points (excursion packages) in it.

A 6 x 6 transition matrix, each of whose rows represents one of the six conditional probability distributions, can represent all of this. This further means that this transition matrix is a probability matrix.

Recall that this matrix applies to all twenty-four time steps in this example, because we have stipulated that the probabilities do not change across time steps – that this conditional stochastic process is homogeneous. If it were not then the transition matrix would be different for at least one of the time steps.

Here is this transition matrix:

	MNoK	LSfS	NyNoM	SfSC	MNySf	SfNyM
MNoK	.05	.35	.20	.15	.10	.15
LSfS	.20	.15	.30	.10	.10	.15
NyNoM	.25	.25	.10	.20	.05	.15
SfSC	.10	.15	.10	.15	.25	.25
MNySf	.35	.10	.15	.30	.05	.05
SfNyM	.20	.25	.20	.20	.10	.05

Notice a couple of things about this matrix. First, all of its rows sum to one and none of its values is negative. These two facts together mean that all of its rows can be individually interpreted as probability distributions. Such a matrix is called a *probability matrix*.

Therefore, the set of all of the conditional probability distributions for a single time step of our organodynamic process can be represented as a probability matrix.

Another interesting thing to note is that all of the rows are different from all of the other rows. This means that all six of the conditional probability distributions are distinct.

This fact does not always have to be the case. But, suppose that they were all equal to each other. This would mean that the probability outcome is actually not conditional. This would mean that it is never helpful to know which region the current trip is too, because the probability of going anywhere specific on the next trip is the same regardless of where one has been on the current trip. In this case, we would not need the matrix – because the same probability distribution would work no matter what the “current condition” is. In this case, we would have a single unconditional probability distribution and would not need to consider them separately as we are doing with the matrix.

The States of Time Steps of Conditional Stochastic Processes

It would be desirable to have a single unconditional probability distribution for the month (time step) based upon this set of conditional distributions. However, given that not all of them are the same distribution, there is not enough information contained in these set of six conditional distributions to be able to develop such an overall unconditional distribution.

It is possible – if we were to obtain certain *additional information* – that we could calculate such a single unconditional probability distribution that *would* correctly

model the next trip for the entire club. But right now we do not have that extra information. We will discuss this situation later. But for now, we are interested in describing the conditional situation that we have.

In other words, our situation is that we have enough information to develop six conditional probability distributions – one for each excursion package, but not enough information to develop a single unconditional probability distribution to describe the travel behavior of the entire club for the next monthly trip.

At this point, we have these six conditional distributions for a single time step. If we continue this process for several months (time steps), we shall have a sequence of time steps, each of which describes its own six conditional probability distributions. Of course, since this particular example process is homogeneous, then the same six conditional distributions – and thus the same transition matrix – applies to all of these time steps.

This sequence of sets of six conditional probability distributions as represented by their transition matrices, for the number of months that a particular individual is a club member, is called a *conditional stochastic process*.

In our example, our conditional stochastic process would have a set of six conditional probability distributions for each time step of the process. This is as contrasted with our previous example in this chapter where each time step has a *single probability distribution* for each step in the process, and which we named *unconditional distribution*.

Differences between Conditional and Unconditional Stochastic Processes

Thus, we have discussed two kinds of stochastic processes thus far. Previously in this chapter we introduced the stochastic process as a dynamical model with a sequence of time steps, each of which is represented by a probability distribution. And we have now refined its name to *conditional stochastic process*.

But, in this present section, we have introduced a new kind of stochastic process. It also has time steps, but each of each of them is represented by a set of probability distributions – each being a *conditional probability distribution*. We have named this a *conditional stochastic process*.

We have also demonstrated, using the Diner's Travel Club as an example, how to represent each time step of a *conditional probability distribution* as a *probability matrix* – where each row of the matrix represents one of the conditional distributions.

Since each time step of an *unconditional stochastic process* is an *unconditional probability distribution*, then such a stochastic process can be represented as a *sequence of vectors*, where each vector represents the unconditional probability distribution of a single time step.

And there is another difference between the *conditional stochastic process* and the earlier *unconditional stochastic process*. In the unconditional stochastic

process, what is *uncertain* at the *current time step* is the determination of which system state will be realized *at the next time step*. There is no uncertainty concerning the current step – only uncertainty regarding the next step. For this reason, a single (unconditional) probability distribution is needed that describes the probabilities for the next step.

On the other hand, in the *conditional stochastic process*, there is uncertainty as to which condition (state) is realized at the *current time step* as well as with the *next time step*. For this reason, a distinct conditional distribution is needed for each of the conditions, because the probabilities for the outcomes of the next step may be different for each of the conditions that could obtain for the current step.

Applicability of Conditional Stochastic Processes

Conditional stochastic processes show up very often in science and engineering, including systems biology. Therefore, it behooves us to look more closely at them. It is reasonable to suspect that they will also frequent organic but non-biological processes.

We have discussed conditional stochastic processes in light of their exhibiting more uncertainty than an unconditional stochastic process. The case we used involved increased uncertainty because for each time step we have multiple probability distributions to work with. And in an attempt to arrive at certainty, we have the additional problem of figuring out which of these several distributions to use. Whereas in the unconditional case we only have one probability distribution per time step and we do not have the additional problem of determining which distribution to use.

However, there is a use case of conditional probability distributions that increases our degree of certainty as compared to that of the unconditional distribution. This is when the current step has already been realized. In that case, we know exactly which of the conditional distributions to use. In such a case, the time step is reduced to a single distribution, and has been “promoted” to the unconditional case.

Thus, the use of conditional stochastic processes has a wide variety of use cases, and has become a flexible tool for use in science and engineering.

In fact, there is a special category of conditional stochastic processes (the simplest case, in fact) that enjoys widespread use in mathematics, engineering and the sciences. This is the so-called *Markov chain* – which we shall introduce next.

Introducing Markov Chains

In the example of the Diner’s Travel Club, we saw that knowing which excursion package a club member selected on the current trip *had an influence* on their select for the next trip.

Other ways to express this influence is to say that the next trip is *conditioned on* the current trip. This language gave birth to our choice of the phrases “conditional probability”, “conditional probability distributions” and “conditional stochastic processes”.

We could also have used the term “dependent, or “dependency”; and could have used the names “dependent probability”, “dependent probability distributions” and “dependent stochastic processes”. (In his study of languages, Shannon used the term “redundancy” to refer to this phenomenon [Shannon 1963].)

In any event, this “influence”, or “conditionality”, or “dependency” resulted in our having distinct probability distributions to predict the next step (the future) *depending on which* condition manifested in the current time step.

Our example was built upon the assumption that the *influence* on the choice of destination for the next trip *depended only on*, or was conditioned only on, the condition that was chosen (the trip that was traveled) on the current time step.

But limiting our “memory” of which selections have been made to the current time step is a special case of the more general situation in which we remember, or know, of all past selections. Moreover, this limitation results in a simplification.

The simplification may have been justified or not. If the conditional probability of choosing the trip for the next time step depends *only* on the current time step, and is not influenced by the choices of any trips before that, then our problem is simpler. This is because we would not have to take into account any more time steps prior to the current time step when developing our conditional probability distributions.

However, things may actually be more complicated. It may be, for example, that the member’s choice of next excursion package is influenced, not only by the current trip, but also by the last several recent trips. That is, the member may have a *memory* that goes further back than the most recent trip only – and that memory may influence his selection of the next trip destination.

For example, we may find that members who selected MNySf excursion packages on any of the last three trips were unlikely to select SfNyM for the next trip.

The situation would be much simpler if members who selected MNySf on the current excursion all showed the same probability of selecting SfNyM for the next excursions *no matter which package they selected on the previous excursion*, or any previous excursions.

If this were true, then knowledge of previous excursions before the current one would give *no useful information* in predicting the next selection. In such a case, even if you took the previous excursions into account, it would not change the conditional probability distributions that you developed by only considering the current excursion and ignoring any previous excursions.

This would be a situation that is easier to calculate, because we could always get by with only being concerned with the current time step when predicting the next – and not have to worry about any previous time steps.

Such a simplified situation is called the *Markov condition*.

A Markov process is a *conditional stochastic process* that “has no memory prior to the current step” because it does not need to. It does not need to because knowledge of previous steps does not change the calculated results as compared with only knowing the outcome for the current step.

This means that, for a non-Markov process, it is not sufficient to have the simple probability matrix that we presented above – where each row represents on the current time step. Rather, for the non-Markov case, it is necessary to complicate the meanings of each row. In such a case, each row must represent an outcome of several past time steps – for as far back as “memory” has an influence. We shall present an example of this in the next chapter.

However – and this is what is important, for a Markov process, it is sufficient for each row to represent a condition that describes a condition (outcome) of the current step only

In the previous chapter, we “pretended” that the Diner’s Travel Club process is Markov, because our conditional probability distributions were all conditioned upon a single package selection – that of the current excursion. If our pretense is incorrect then our calculation must change so as to take into account more previous outcomes than merely the current time step. Of course, this would complicate our calculation.

Because of the complexity of lifelike systems, and in order to provide a reasonably tractable approximation, in this text we are going to restrict the Organodynamic framework to the use of Markov processes, rather than the more general conditional stochastic processes that include the non-Markov case.

However, as we shall see in later chapters, there are many biological processes that are clearly *not* Markov. This is because process steps other than the previous step influence them. This is bad news for Organodynamics that seeks to be both tractable and accurate.

The solution to this dilemma, then, is that we shall leave the extension of Organodynamics to include the more general *conditional stochastic processes* to further research. It is unfortunate to have to make this trade-off. But even restricting the Organodynamic framework to Markov chains will prove quite challenging in terms of tractability. So this is a choice that we shall regrettably make at this time.

Incremental Example Model Development

<>

Progress Check

In Part II, Organodynamics, we are gradually and incrementally developing a mathematical framework for modeling living and lifelike systems according to the seven organizing principles of OCS. Once completed, this modeling framework will achieve the form of an abstract but highly flexible mathematical structure that we are calling an *Organodynamic web*.

We are using the example of a hobbyist's toolkit for building electronics as an analogy to Organodynamics. Both are systems for "building something", and in both cases the resulting "thing to be built" has a network structure and has been given a general name.

In the case of the electronics circuit toolkit, the general name of the "thing to be built" is "circuit". The circuits built by this kit are highly customizable to specific applications. In the case of Organodynamics, the general name of the "thing to be built" is "Organodynamic web". The models built by this mathematical framework are also highly customizable to specific applications.

Because of the complexity of organic systems, any model that represents them needs to be built in cumulative and increasingly accurate stages. Each stage is generally referred to as an "approximation". Organodynamics as a model builder's toolkit provides for six of these approximations. These approximation levels are well defined and occur at precise places in the methodology set forth by Organodynamics.

In this chapter we have begun to introduce the machinery that we need to mathematically represent the *uncertainty* that the sixth organizing principle of OCS observes as inherent in organodynamic processes.

In information theory, both uncertainty and information are described in terms of probabilities. In fact, information theory provides a measuring function for both of these, named *entropy*, that measures the amount of uncertainty inherent in an event in a manner that is inversely related to the probability of that event.

Consequently, both uncertainty and information need a probability distribution to work with. A probability distribution needs two things: choices and probabilities associated with those choices. In probability theory, the set of choices is called the sample space.

In order to obtain a set of *choices* for Organodynamics to work with, we decided to use a generalized set of *possible states* that an organic system can exhibit at a specific moment in time. Of course, in the previous chapter, we established that we would represent such a state as a *system organization* – which is a specific set of relationships among the components of the system being modeled.

Due to its centrality to the Organodynamic framework, let us reiterate this. The set of choices that will populate the sample space of one of our probability distributions is a set of possible system organizations of the underlying system that is being modeled. And, of course, the probability distribution assigns probabilities to this sample space.

So, such a probability distributions forms the model of a single time step of a system that we are modeling. But, since these systems are dynamic, then they consist of a sequence of time steps. Such a sequence, then, is modeled in Organodynamics as a sequence of probability distributions each of whose sample space is a set of system organizations of the underlying system that is being modeled.

Such as sequence is also described as an *unconditional stochastic process*. And the probability distributions that model the time steps of these processes are called *unconditional probability distributions*. The process being modeled by one of these *unconditional stochastic processes* is referred to as an *organodynamic process*.

We also considered a situation in which we do not have enough information to model the uncertainty of a time step with a single unconditional stochastic process. But, we may have enough conditional information to model the uncertainty of the time step with a set of *conditional probability distributions*.

We can then also consider a sequence of these conditional probability distributions that model an entire dynamical process as a time series. We called this a *conditional stochastic process*.

Whereas an unconditional stochastic process is a sequence of probability distributions that model the time steps of an unconditional process, the conditional stochastic process is a sequence of *sets of* conditional probability distributions. In this case, each *set* of distributions models a time step of the process.

In the case of the conditional stochastic process, we showed how to represent the set of conditional probability distributions as a probability matrix. Thus a conditional stochastic process can be represented as a sequence of probability matrices – one for each time step of the process.

Likewise, for an unconditional stochastic process, a vector can represent each time step. This is a probability vector that represents the unconditional probability distribution for that time step.

Thus, the result of this chapter is these two types of stochastic processes: the *conditional stochastic process* and the *unconditional stochastic process*. We shall use both of these going forward to model organodynamic processes - which are the subject of Organodynamics. We shall use the *conditional stochastic process* in certain circumstances when we have less information. Then, when we gain more information, we shall promote the *conditional stochastic process* to an *unconditional stochastic process*.

Either of these two stochastic process models forms an elemental portion of the Organodynamic web structure that we are gradually building within this text.

To use our analogy of the hobbyist's electronic toolkit, one of these stochastic processes is like a "copper wire" in the analogous toolkit. That is, these

stochastic processes become the “connections” between the “nodes” of the Organodynamic web.

Below, we shall see that the modeler of a complex organic entity will identify multiple subprocesses within that entity, and model each of them as one of these organodynamic processes. Subsequently, multiples of these processes will then be connected into a network by using a “node” structure that we shall also define below.

The Canonical Organodynamics Model

The Issue Addressed

Every modeling framework requires that the modeler adopt a particular viewpoint toward any target system that is to be modeled. This chapter discusses the modeling viewpoint required by Organodynamics.

It then discusses the basic steps that a modeler would then take to construct such a viewpoint of a target system, and then how begin to model the target system with the framework.

The Biases of the Organodynamics Framework

Every modeling system stakes out its own particular view of systems, and then provides specific modeling mechanisms that implement that view.

This view is always parochial to some extent, and establishes its own biases in the form of the modeling mechanisms that it supports. These biases force the modelers who use the modeling framework to develop an understanding, or articulation, of a target system-to-be-modeled in the terms of this bias.

We shall look at some of the biases of the Organodynamics framework. Then we shall look at how the modeler must first develop an understanding of his target system that accommodates these biases.

The System View

As laid out in the previous chapters, the Organodynamics systems view entails discerning the target system as a *set of components* called the *system population*. Organodynamics views this population of components as being assembled into a set of inter-relationships called a *system organization*, or *organization* for short.

Generally, any system has exactly one population of components at any moment. However, this population can be arranged, or structured, into many possible *organizations* – of which exactly one can be *realized*, or *made manifest*, at any one moment.

Organodynamics emphasizes these *organizations* of components more than it does the components themselves.

All of this is the Organodynamics way of viewing a system statically – at a single snapshot in time. We refer to these moments as *time steps*, or simply *steps*.

From a dynamical perspective, Organodynamics emphasizes that, from one time step to another, *which* of these system organizations is realized at each time step can change. And, given the system's population, which *organization*

is realized changes from moment to moment to other organizations of the same population.

In Organodynamics, this set of possible system organizations of a population (the “underlying system”) is called the *state space* of that population.

But there is a second wrinkle to the dynamical aspects of a system that Organodynamics is particularly keen about. At any given time step, each of these possible *system organizations* of an underlying population has a *probability* of being realized at the next time step.

So, in Organodynamics, since we are considering the state space as a set of mutually exclusive alternatives, exactly one of which can be realized at each time step of a process, and each of which is assigned a probability, then it is also appropriate to refer to this state space of system organizations as a *sample space*.

So, given a system and its population, for each time step there exists a probability distribution that describes the probabilities for each possible *system organization* of its underlying population at that time step.

There you have it. This is the viewpoint that Organodynamics requires a modeler to make of a target system that is to be modeled by the framework.

The first thing to look for is, then: Can your target system be viewed as a set of components that can take on different “arrangements”, structures or *organizations*?

The second thing to look for is: Does your system “change” from one of its organizations to others across time steps?

The third thing to look for is: Can you associate probability values to each of these different *organizations* of your components? In addition, this set of probabilities must add to one (1), so that you have a probability distribution over the sample space of all possible organizations of the underlying population.

If the answer to all three of these questions is “Yes”, then your target system is a reasonable candidate for being modeled by the Organodynamics framework.

However, there is also a fourth question that you will want to ask yourself: Are these biases interesting to me with respect to my target system?

It may take a little time and experience to be able to perceive the significance of the particular systems viewpoint of the Organodynamics framework. So you may want to digest more of this text and the examples herein before you try to answer the fourth question.

In any event, by way of example, lets now take a big-picture look at the basic steps that a modeler would take to apply the elements of the Organodynamics framework to develop a model of a target system – one that passed the first three questions just posed.

Introducing a Canonical Biological Example

In this section, we shall introduce a very general description of an example biological system that can hopefully serve as a canonical example of all biological systems that can be modeled with the Organodynamics framework.

Our biological example will initially consist of a closed system of atoms and molecules, large and small. This example will purposefully be very general, but we shall immediately be able to see that there are several biological entity types that fit this canonical example: cells, multi-cellular organs, whole organisms, ecologies, etc.

However, right now we shall just consider this example as broadly described. Specifically, we shall consider a biochemical system - *a system of atoms*. (Pretty much any biological – even physical – entity fits this description).

A Closed System of Atoms

Lets first consider this system of atoms as closed - any closed system of atoms. We'll open it up a little later.

Within this closed system, various chemical reactions can occur, based upon the chemical state of the system at any point in time. Being closed, these reactions can cause the molecular constitution to change over time – even though the type and number of atoms will not change, given that it is a closed system.

In this situation, then, at any point in time, the entire system will be in some “state” where some of the atoms will be bonded into molecules while others are “running free” and not bonded to any other atoms at all.

This state can be described as a “configuration”, or an “arrangement” of the entire system of atoms. Such an arrangement can be described in such a manner that it is clear which of all the atoms are currently “un-bonded” and which ones are currently bonded with other atoms into molecules – and *how* they are bonded into these molecules.

Such a configuration describes the *entire closed system* at a snapshot in time all at once – and every atom and every molecule is represented in this configuration. In fact, this configuration, or arrangement, of the atoms is precisely what we have termed a *system organization*, or *organization* in Organodynamics.

From an Organodynamics framework perspective, then, each *atom* is a *component*. The entire set of atoms in the closed system is the *population*. Any one of the *configurations* of atoms is an *organization* of that population. The collection of all of the *possible organizations* of the population is called the *sample space* of that population. *Together*, this population and one of its *organizations* is a *system instance*.

At this point, we have described this system *statically* – at a snapshot in time, using the ideas of component, population and organization. Lets now start to describe this system dynamically.

Of course, at each moment in time, exactly one of these *organizations* is manifested. However, across time – even within this closed system of atoms – different organizations, arrangements, of this same set of atoms appear. That is, there can be a “change its organization” over time.

In our example, the set of possible *organizations of atoms* is the *sample space* of the system in Organodynamic terminology. Admittedly, this is a quite complex sample space, because its elements, each being an *organization*, are complex. It is a sample space nevertheless.

If we observe this system for a while, it becomes evident that some of these organizations appear often, some seldom and some (perhaps) not at all. In other words, these *organizations of atoms* have *relative frequencies*.

But, these relative frequencies can be interpreted as *probabilities*. Thus, each element of our sample space, each system organization, now has a probability. Together these constitute a finite state *probability distribution*.

Since this is a *closed system* – the types and number of atoms by type never changes over time, if we watched this system long enough, the relative frequencies of these possible system organizations would “settle down” and begin to be consistent. That is the relative frequency of each specific organization of atoms stays about the same. We may find different relative frequencies for different atomic organizations. But the relative frequency of each will remain about the same after a while.

In other words, the probability distribution for our sample space remains the same over time. This means that we have a *homogeneous* situation. In Organodynamic terms, then, we have a *homogeneous finite state stochastic process*.

We could also take a slightly different view of this dynamism. We could notice that there is a dependency between the outcome of one step and the next. Specifically what we would notice is: *Knowing the outcome of one time step helps us to predict the outcome of the next step*.

In probability terms, this means that knowing the outcome at one step effects the probability distribution for the next step. And this means that we are in a position to define distinct probability distributions for the *next step* in the process, given that we know that certain of these atomic organizations occurred at the current time step.

If we knew this kind of thing for the cases of every possible atomic organizations occurring at the current time step, then we would be in a position to model the system of atoms as a Markov chain – which is the type of *conditional stochastic process* emphasized in the Organodynamics framework.

In this case, the Markov chain would be homogeneous because the same set of conditional distributions works for every time step.

So, we would use a homogeneous Markov chain to model our closed dynamical system of atoms.

Taking stock of what elements of the Organodynamic framework we have just discussed, we have a closed *system* of atoms that are comprised of a *population of components* (the atoms themselves); and these components can be arranged into various *organizations*. The set of all possible *organizations* of atoms comprises a *sample space* of organizations. It is this *sample space* that Organodynamics emphasizes, rather than the individual atoms.

Each of these organizations of atoms in the sample space has a probability of occurring at any point in time, and therefore forms a probability distribution for that moment in time. Thus, our closed system of atoms as a dynamical system is modeled as a sequence of these probability distributions – and is therefore called a *finite state stochastic process*.

However, since this system is closed (and the population of atoms does not change), then this probability distribution stays the same for all of the time steps involved in this stochastic process. Thus, it is a *homogeneous stochastic process*.

If we start looking at the stochastic process in terms of which atomic organization manifests at one time based upon which one manifested at the previous time, then we have what is called a *conditional stochastic process*. If this process depends only on the current time step, and not any earlier time steps, then we have a simpler special case that is called a Markov chain.

So adopting a viewpoint about one's target system in which all of these mechanisms can be used is the first step in using the Organodynamic framework to model your system.

Opening the Closed System of Atoms

However, we know that biological systems are open, not closed. Thus, we shall open up our system so that, from time to time, some of the existing atoms will leave the system, and some new ones will enter. In fact, we shall allow whole molecules to enter and leave as well.

Specifically, we shall “open the gate” during a single time step and then immediately close it during that same time step. So, generally we can provide a momentary opening of the system from time to time.

To do this, we adopt the viewpoint that “every now and then” a “gate” is opened in the boundary of our closed system, and new atoms and molecules from the outside are allowed to come in. Also, existing atoms and molecules from the inside are allowed to leave. This fact, at that point in time, changes our closed system to a different closed system.

Another way to view this is that we have a sequence of closed system processes. For a few time steps we have one closed system. Then we momentarily open the gates and then for a few more time steps we have a different closed system, etc.

We shall articulate this change by saying that the *population* has changed. It now has a new set of components that consists of all of the atoms that just entered, plus all of the atoms that were there before the opening, minus the ones that left – whether these atoms were part of molecules or standing alone.

Of course, usually, only a few new atoms enter and only a few leave whenever we open it up. So, usually, the population doesn't change very much at once. However, from a system perspective, the population, as a set, has changed.

But, changing the population affects the sample space, its probability distribution, and the resulting Markov chain. So even small changes to the population can have marked effects in the dynamical system.

But – and here's the point, until the next time step that the gate opens, we again have another *homogeneous Markov chain*.

It is, in general, a different homogeneous Markov chain, because the probabilities distributions have changed. So, on each side of the time step in which the gate was opened, we have two distinct homogeneous Markov chains.

One could view this another way and say that all of the time steps taken together is a single nonhomogeneous Markov chain. This would also be correct. We could also say that we have a single piecewise-homogeneous Markov chain. And this is the fact that we shall emphasize in Organodynamics.

Thus, in our system of atoms, we have applied the Organodynamics framework up to the point where we can model the system as a piecewise-homogeneous Markov chain. And this is what we shall do generally.

In fact, almost every biological and non-biological system that model using Organodynamics will be approached in exactly this same way. Simply go through the same steps and an Organodynamic model will result.

What we have Left Out So Far

We have not addressed a number of issues that need to be modeled in this example. More advanced features of the Organodynamic framework address more advanced modeling issues. There are several of these features – which we shall address in due time.

We shall describe only one here. It is the issue of “nested systems”, or “composite system”.

In the atomic system example, we made no distinction between small molecules and macromolecules. They were all treated the same. For example,

we made no distinction between amino acids and proteins or between RNA molecules and nucleotides.

In other words, the entire probability space was treated as having only two *levels of organization*: atoms and molecules. All of the various levels of organization (or levels of complexity) available in the world of molecules were thrown into one huge category: “molecules”. Clearly we need to be able to break these down into several layers of abstraction just among molecules.

Organodynamics *does* present such a mechanism in a later chapter. This mechanism accommodates the OCS principle of organization number 3: Nested Systems.

The Atomic System Example and the Organodynamic Web Construct

We have also not yet discussed where our atomic system example fits in with the Organodynamic web structure, which we shall now address.

We first introduced the atomic system as a closed system whose population of components is the atoms. An *organization* of this population is a single arrangement of all these atoms. Such an arrangement would include some number of molecules plus some number of free atoms that have not chemically combined with other atoms.

Notice that we have not said that a single molecule is an *organization* of these atoms. Such a statement would be *incorrect*.

A more correct way of thinking about what an *organization* of these atoms could be is a *partitioning* of the population into a mutually exclusive and exhaustive set of compartments. Any compartment that has exactly one atom in it represents a “free” atom. Any compartment that has more than one atom in it represents a molecule.

However, as described in the chapter on Systems and Components, we defined a specific set-theoretic mechanism that Organodynamics has decided to use to articulate such *organizations*. This mechanism segregates the “free atoms” by not representing them, and it represents any collection of atoms that is deemed to be a single *molecule* as a set of sets of ordered pairs called *duples*. Each duple represents a covalent bond between atoms with a molecule.

We then watched this system change over time for a few time steps without allowing any of the atoms to leave the population, and without allowing any new ones to enter. Each change from one time step to the next is an opportunity for a chemical reaction to occur among this population of atoms – and thus an opportunity for the mix of molecules to be *reorganized*.

In other words, each time step permitted an opportunity for the *organization* of the system to change, while the *population* stays the same.

We also noticed that a dependency is at work in this dynamical system. This dependency says that the probability distribution for estimating the outcome of

the next time step is dependent on, and only on, the *organization* of this system that is realized at the current time step. This statement says that this dynamical system is a *conditional (or dependent) stochastic process*. And the “only on” part of our assumption goes further and prescribes that we have a Markov chain.

We also assumed that, given any such organization at the current time step, that the probability distribution that estimates the outcomes of the next time step are true for any time step when it is treated as “current”. This statement is equivalent to saying that we have a *homogeneous Markov chain*.

Thus, as long as we keep the system closed by not allowing any atoms (the components of the system) in or out, then we have a *homogeneous Markov chain*. And, this represents a “wire” of the network that is an Organodynamic web. More correctly, it represents a *segment* of a whole wire of an Organodynamic web.

Our next step was to open up the system and allow components (atoms) to enter and leave the system at certain time steps. This meant that, at those time steps, the population changes. This has the consequence of the sample space of possible system organizations also changing, along with their probability distributions.

The result of this ramifies so that it affects the Markov chain that was in effect before this time step no longer correctly models the process for time steps after the population was “opened”. The result is that we have a new, and (slightly) different, Markov chain starting with the time step where we opened the population.

As long as we leave the system closed, we shall again have another homogeneous Markov chain – another *segment*. However, it will be operating under a different set of probability distributions than was the first segment.

If you view all of the time steps of both of these segments together as a single Markov chain, then it will be nonhomogeneous, because it is not the case that the same set of probability distributions works for all of its time steps.

However, if you break this Markov chain into its two *segments*, then you have a Markov chain that is homogeneous in pieces (the segments) but not overall. Each of these *segments*, then, is modeled as a *homogeneous Markov chain*.

And, we can connect several of these *segments* together to model a longer process, which we shall call an *edge*. We shall use the term “edge” from graph theory. (Later we shall connect multiples of these *edges* by using connectors. These connectors will form the *nodes* of a graph while the *edges* will form the *edges* of the graph.)

Thus, an *edge* of an Organodynamic web is a *piecewise-homogeneous Markov chain*. And, a *segment* of an edge is one of the homogeneous Markov subchains that comprise the edge.

We shall leave it to later chapters to add the other machinery necessary to “connect these wires” into a “network” structure using some kind of “node” structure.

Therefore, we shall use structures and concepts from probability theory, functional analysis and algebra to construct larger mathematical entities that then behave as the elements of a graph, or network. So, at the lower levels of detail, we shall be able to speak in terms of mathematical structures such as probability matrices and algebraic entities and operations, while at the larger level of abstraction we can treat the entire apparatus as a directed network, or graph.

Conclusions

In this section, we have introduced a canonical biological example system. Any biological system that can be modeled using the Organodynamics framework can be seen as a special case, or at least analogy, of this very general example.

Hopefully this chapter has shed light on how a target system must be viewed in order to be able to be modeled with this framework – and generally how to go about it.

Incremental Example Model Development

<>

Progress Check

At this point, we know how to construct the “wires” of an Organodynamic web. Specifically, the “wires” are instances of *piecewise-homogeneous finite state finite step stochastic processes* – specifically *piecewise-homogeneous finite state finite step Markov chains*.

The Uncertainty Model Gradient

The Issue Addressed

We have already seen in the previous chapter that there is a modeling need for two different types of stochastic process in Organodynamics – depending upon how much “information” the modeler currently has about the underlying system (organic entity) being modeled.

If one doesn't currently have very much information, the *conditional stochastic process* may be appropriate. However, if one has little more information, then the *unconditional stochastic process* may be more appropriate.

Also, a conditional probability distribution establishes dependency relationships between the one time step and the next. Therefore, whenever the outcome of a specific time step of a conditional stochastic process becomes known (realized), then these dependencies result in providing additional certainty about the which probability distribution to use for predicting the outcome of the next time step. Essentially the realization of a specific time step promotes the model of that time step to an unconditional distribution.

This fact sets up a situation where there are different types of models that can be used depending on how much information one currently has about the subject system.

It also suggests the possibility that a modeling methodology that encourages the initial construction of one type of modeling system that could then be improved, or promoted to another more preferable type of modeling system whenever additional information is gained during experimentation.

Such a methodology would present a gradient of model types that could be progressed through as additional information is gained regarding the subject system being modeled.

Such a gradient of model types is very useful in the application of modeling, since it is frequent that information is constantly gained about a system during the live of model development, and it is appropriate to have a mechanism to develop initial models based upon scant information and then to improve the model as more information is gained.

This chapter presents such methodology, which it calls the Uncertainty Model Gradient.

OCS Organizing Principle Supported
Organizing Principle # 6: Uncertainty.

Biological Example of the Issue

Consider any semi-permeable biological entity. This could be a single cell or the nucleus of a prokaryotic cell. For multicellular organisms, it could be a tissue, an organ or the organism itself.

Because of their semi-permeability, each of these systems can be modeled as having a time step during which atomic components enter or leave; and having a period of time comprising one or more time steps in which nothing enters or leaves, but during which constant chemical reactions (metabolism) also constantly change the molecular constitution of the stable atomic population.

Consequently, as discussed in the previous chapter, each of these biological systems (entities) can be modeled as a piecewise-homogeneous stochastic process. Moreover, since we are making the simplifying Markov assumption, then these systems can also be modeled as piecewise-homogeneous Markov chain.

However, in time, we observe the actual outcomes of the earlier time steps of such a process, and would like to be able to apply that new information so as to refine our prediction model of the time steps that follow. As discussed in the previous chapter, this means that we are dealing with two types of probability model: *conditional* and *unconditional*.

And, the unconditional model inherently embodies *more certainty* than does the conditional version of the model. We would like to be able to start out with the *conditional* version of the model and then *promote it* to the unconditional version as new information comes in.

Both of these versions of the model, conditional and unconditional, are probability models. But when we actually know the outcome that is realized for a time step, then all uncertainty has been removed. This is a *deterministic model* of the same time step.

Thus each time step can have three different model versions, ordered according to how much uncertainty each embodies: the conditional model, the unconditional model and the realized model.

Specific Challenges

To determine what type of additional information can be obtained that will enable the promotion of a conditional stochastic model to an unconditional one. Additionally, procedures must be developed to make use of such information to perform the promotion.

We also need for this methodology to show how to establish these three model versions for each time step of an organodynamic process, and for promoting the models for each time step to the next (improved) model version as information comes in – and for promoting that information across future time steps.

The Organodynamic Approach to Modeling the Issue

A “gradient” of three types of models will be developed. Each of these will represent a distinct level of uncertainty. A methodology will be developed that enables one to initially select the model type (of these three) that exhibits the most level of certainty but that fits the current level of information about the system being modeled that the modeler currently enjoys.

Subsequently, as more information is obtained, a procedure is developed for applying that information in order to promote the current model type to the next model type.

It is expected that the modeler will generally begin with the model type of the three that exhibits the most uncertainty, and then incrementally progresses through the other two model types as additional information is gathered.

In addition, it is also expected that the modeler will apply this process at all six approximation levels.

Three Model Types of Decreasing Uncertainty

Thus far, we have identified two categories of stochastic process for use in the Organodynamic modeling framework: the *unconditional stochastic process* and the *conditional stochastic process*.

We saw that the unconditional version is ideal for representing uncertain dynamical systems because it provides us with a single probability distribution for each time step that predicts the outcome of the next time step.

However, sometimes we don't even have enough information to develop one of these unconditional processes because we have insufficient information to develop these probability distributions for each time step of the process. However, we may have enough information to develop a preliminary model that – although not as informational, will tell us something.

This preliminary model – called a *conditional stochastic process* - is able to provide a partial stochastic model. Thus, it is appropriate to – when useful – develop a conditional stochastic model first; and then when sufficient additional information is obtained, to then convert the partial model into an *unconditional stochastic process*.

This situation often occurs when distinct researchers each have a partial view of the same system. The view of each must be able to develop a conditional probability distribution for at least one of the possible states of the system. That is, for some particular state, they must be able to determine the conditional probabilities for that state. In addition, collectively, conditional distribution must be developed for all such states. If all of this is the case, then a conditional stochastic model can be developed.

So, this accounts for two types of stochastic models: conditional and unconditional. The unconditional model is usually preferable, because it embodies less uncertainty than with the conditional model.

In addition to these two stochastic model types however, there is one more model type. This is the *realized* model – which has *no* uncertainty at all. This is the model that is recorded when each of the outcomes has been determined whenever each step of an unconditional stochastic process is *realized* (the die is cast or the coin is tossed).

At that point, the specific system organization is known, and all of the others have not occurred for that time step. One can restate this *realization* in probability terms by saying that “one of the possible states now has a probability of 1 (the state that was manifested), and all the other now have a probability of 0.

In other words, one can describe the transition from the unconditional model to the realized model by saying that the single probability distribution used in the conditional model has changed to a very simple probability distribution: the “deterministic distribution”.

This *realized* situation is also a model, but it is now deterministic. We call this model the *realized deterministic model*.

Thus, we now have three types of models, each representing the same underlying process but with gradually decreased degrees of uncertainty - and therefore decreasing entropy. In fact, as a modeling practice, it generally makes sense to develop these three types of model in the order of decreasing uncertainty. In fact, each can be gradually modified into the next whenever additional information is obtained.

So, we have now come to making recommendations about modeling practice using when using the Organodynamic framework, and thus have the beginnings of a *modeling methodology*. This chapter shall introduce such a practice, which is named the *uncertainty model gradient*.

The Uncertainty Model Gradient

What we have just described is a sequence of three models of the same organic process, each gradually decreasing in its *degree of uncertainty*. These three model types are, in decreasing order of uncertainty:

- The Conditional Stochastic Model (the conditional model)
- The Unconditional Stochastic Model (the unconditional model)
- The Realized Deterministic Model (the realized model)

We shall define this set of three related models for an organic process the *uncertainty model gradient*.

It is expected that the modeler using the Organodynamic framework will proceed through all three of these model types in the above prescribed order. Of course, it is permissible, to skip the first if the modeler has sufficient information from the beginning to proceed directly to the second or the third.

Most often, in modeling organic processes, it is more efficient to begin with the first. Then when sufficient information is obtained, the conditional stochastic model can be promoted to an unconditional one. When even more is obtained, the unconditional stochastic model can be promoted to the realized deterministic model.

In practice, each time step is promoted to the next level of model one at a time. Thus, one may at times have a mixed stochastic process where some of the earlier time steps have been converted all the way to the *realized deterministic models*, some of the intermediate steps have been converted to unconditional stochastic models, and the remaining time steps remain as *conditional stochastic models*.

In the subsection below where we detail the Markov chain, we shall discuss in more detail the process of promoting a conditional stochastic model to an unconditional one.

Levels of Approximation and the Uncertainty Model Gradient

From the preview chapter near the beginning of Part II, it was explained that we are developing the Organodynamic framework as an incremental and cumulative modeling exercise. For manageability purposes, we are building up a complex model step by step.

In fact, we are in the process of developing six (6) levels of fidelity, or accuracy, that we have called “levels of approximation”. Each of these levels develops additional mathematical structure that adds on more complexity, and more fidelity, to the mathematical structure of the previous level of approximation.

Eventually, one of these levels finally provides us with a structure that we have named the *Organodynamic web*. However, this incremental and cumulative process does not end there. We shall continue on afterward for a few more steps enriching further this network structure.

All of these structures are given names (that were introduced in the Preview chapter), but they shall also be known by their *level of approximation*: “first approximation”, “second approximation”, and so on all the way to the final level, the “sixth approximation”.

It is not intended that all modeling exercises would pursue all six levels of approximation. It is intended, however, that one begin with the first approximation and then stop at some desired level – and to usually go as far as the fourth approximation – the basic Organodynamic web.

But, for however many of these approximation levels one pursues, the three models of the *uncertainty model gradient* will advance through all of the approximation levels that are pursued.

In other words, the answer to the question “How does the *uncertainty model gradient* fit within the *six levels of approximation*?” is:

Within each of the six levels of approximation, the modeler will conduct all three of the *uncertainty model gradient* models.

Thus, if one were to take the modeling exercise all the way to the sixth approximation, then in total one would have developed eighteen (18) models in the process.

However, it is more accurate to say that for the modeling of any organic system with the Organodynamic framework, there are the three models, all of which begin at “the first approximation” and then advance through the other levels of approximation until the modeler is satisfied with the fidelity of the model.

Elements of the Three Uncertainty Gradients Models

This chapter defines more details of the three uncertainty model gradient types. We shall visit the in their order of uncertainty:

- The Conditional Stochastic Model
- The Unconditional Stochastic Model
- The Realized Deterministic Model

First, lets recall a few ground rules. In Organodynamics, we are modeling living and lifelike systems. As such, everything we are dealing with is *finite*. This means that the systems that we are modeling have a finite number of components. Those components can be organized into a finite number of organizations, each of which has a finite number of duples (ordered pairs).

In addition, the time steps in our processes are discrete – not continuous. Moreover, since all living systems have a limited lifetime, our process shall have a finite number of time steps.

We can summarize all of this by saying that will be modeling these organodynamic processes with *finite state, finite step stochastic processes*.

The Conditional Stochastic or Markov Model

As we have already indicated, In Organodynamics, we shall use the simplest type of *conditional stochastic model* – the Markov chain. This will simplify our descriptions and operations significantly.

A conditional stochastic model is useful whenever the system being modeled exhibits *dependencies between one time step and the next*. More specifically, in these types of systems, there are *certain conditions* with respect to which the outcomes of any time step are *dependent upon which condition* was realized at the previous time step.

In such a case, it is important to be able to capture these dependencies and represent them in the model. The reason for this is that these dependencies have the overall affect of *reducing the uncertainty* of the process. Stochastic

processes whose time steps are interdependent have a reduced degree of uncertainty over corresponding processes whose time steps are completely independent.

Conditional stochastic process models can often be established with incomplete information and then eventually promoted to *unconditional stochastic processes* when more information is gained during the modeling process.

In this early stage, only the interdependencies between time steps are initially modeled. The overall probabilities at each time step are not yet known.

For example, in our “atomic space” example, we have described each *state* in our probability space as a *system organization*, each of which describes a particular arrangement of all of the atoms in our population in a set of molecules, plus perhaps some free atoms.

We may be in the following position regarding this sample space of organizations: We may not currently know the probabilities of each individual arrangement, or organization. However, we may be in a position to predict what the organization of the next step will be *given* that we know what the organization of the current step is.

Since the outcomes of these adjacent time steps are interdependent, then we know that the outcomes of the next time step are *conditioned on* the outcome at the current time step. This can be stated as: Each possible organization that can be realized at the current time step *has its own probability distribution* that predicts which organization of these atoms will be realized at the next time step.

Thus, a *conditional stochastic model* of a single time step will be a set of conditional probability distributions – each being conditioned on the fact that a specific organization of atoms was realized at that time step.

This is a powerful probability model of the time step. However, it is still not enough information to actually determine the probability of what will occur at the next time step. In order to do that, we would have to know what actually occurred at the current time step – and at this point we do not know that.

Nevertheless, at some point in the future, we expect to find out which condition (organization) is actually manifested at the current time step. Then at that time we will better know which probability distribution to use to predict the next time step than if we did not know the conditional interdependencies among the time steps.

The Unconditional Stochastic Model

The unconditional stochastic model is a stochastic process that provides exactly one non-conditional probability distribution for each step. This distribution describes the probability of realizing each condition as the *next step outcome* – regardless of the outcome at the current step.

One can promote a Markov model (or conditional stochastic model) to an unconditional stochastic model by the proper application of additional information.

The promotion of a Markov model to an unconditional stochastic model amounts to the calculation of a single unconditional probability distribution from the set of conditional distributions in the Markov model plus some additional information that assists in making that calculation.

For example, the modeler may receive additional information that specifies which of the set of conditional probability distributions to use for a specific the time step. Then, such specified conditional distribution is promoted to an unconditional distribution.

Another example could be that the “additional information” is in the form of some values that specify some “weighting factors” for each of the conditional distributions of a Markov chain time step. These weighting factors could then be used to calculate a single distribution as a linear combination of the conditional distributions for that time step.

From a modeling perspective, any of this “additional information” may represent some *generating conditions*. Thus, a variety of conditions within the complex environment of the organic process being modeled could exert a variety of influence on any step of the Markov chain. It would behoove the modeler to identify these generating conditions as well as any mathematical relationship they may have on the various steps of the Markov chain.

When we see below how the state of a step of a Markov chain is represented via matrix algebra, and how various matrix operations are used to model these influences and co-influences, then it may be easier to imagine how these generating conditions can be identified and modeled mathematically.

The Realized Deterministic Model

The realized deterministic model is simply the sequence of manifest outcomes of all the time steps of the process.

When the outcomes of each of the steps of a dynamical system process are known – determined, then the non-conditional probability distributions of each determined step changes so that the probability of the realized state is one and the probabilities of the other states are zero.

Of course in Organodynamics a *state* is a specific *organization*, or set of dupes, of an underlying system’s components. Thus, a *realized* Organodynamic model is a sequence of system organizations of the same underlying system.

Example Markov and Conditional Stochastic Models

Of the three types of models in the uncertainty model gradient, the Markov models are the most complex – and the ones that modelers using the Organodynamic modeling framework will spend most effort on.

The other two types of models, the unconditional stochastic and the realized deterministic, are promoted in relatively straightforward ways from the Markov model.

Thus, it will be very helpful to provide some examples of Markov models. In fact, the ability to discern when a conditional stochastic process is Markov and when it is non-Markov is also an important detail.

When a process is non-Markov then our simplifying assumption in this text is not met, then our modeling system will lose accuracy. Even though we have decided not to cover the non-Markov case in this text – due to its complexity, it is sometimes reasonable to use the Markov approach as an approximation. This decision is left to the modeler.

Regardless of whether the modeler decides to use the Markov assumption, it is nevertheless essential to ascertain whether the assumption is warranted. Therefore, this subsection will present examples of both Markov and non-Markov *conditional stochastic processes*.

The second set of examples presented below is biological. As such they are quite complex, as is generally the case with biological examples. These are presented because of their pertinence to this text. Unfortunately, there are more complicated than desired for a clear and lucid set of first examples. Therefore, as set of simpler non-biological examples are presented first.

Simple Non-biological Examples of Markov Models

Readers who are familiar with Markov processes, the Markov condition, independent and dependent events, can skip either or both of the this and the next subsection.

The examples in this subsection are relatively simple Markov chain models. Some non-Markov, but conditional stochastic, models are also presented – mostly for the purpose of contrasting them with Markov models so that the Markov principle can be better appreciated.

All of the examples in this subsection pertain to “our friend Jack” and the types of evening meals that he regularly eats.

Jack has been sent on a two-year research mission to Antarctica. Unfortunately, his choice of evening meal is very limited. For each dinner he is given the choice of chicken, fish or meat. Jack is asked to choose the next evening’s main course after dinner on the current evening.

Of course, the kitchen keeps records of Jack’s meals, and after several months it emerges that Jack has exhibited consistent behavior. The kitchen has put

together a set of conditional probability distributions that describe Jack's dinner selection behavior.

Notice that this example supports slightly different research than suggested for conditional stochastic models above, where the suggested application was when the overall unconditional distribution cannot be known. In this case, the kitchen has the information to be able to provide the single unconditional distribution. However, providing the conditional distribution allows for a more specific prediction for a following meal because it represents the interdependencies.

So, the kitchen is providing us with three conditional probability distributions that describe Jack choosing behavior. These are for predicting his dinner selection for tomorrow...

- *Given* that he selected fish for today
- *Given* that he selected chicken for today
- *Given* that he selected meat for today

The three conditional probability distributions for what Jack will select for tomorrow, given what he selected for today, are represented in the table below.

It is customary to represent the set of conditional probability distributions for a Markov chain as a matrix – called the *transition matrix*. The rows of this matrix each represent one of the conditions – a member of the sample space – a *current step condition*.

The contents of the cells of the row contain the probabilities of each of the *next step possible outcomes*, for the conditional probability distribution for the condition represented by that row.

In other words, the rows of the transition matrix represent the *current step conditions* and the columns of the matrix represent the *next step possible outcomes*. But in a Markov chain step, both of these are the same set of conditions. Therefore, the transition matrix for a Markov chain is a square matrix.

We shall exemplify this now. The following table represents the probabilities in each row describe the probability distribution for the case mentioned in the first cell of the row. The body of this table is the transition matrix for this Markov chain.

Notice that the three entries of the first row represent the conditional probability distribution for the condition that the current step meal was chicken. The second row is for fish; and the third if for meat.

Conditional Probability Distribution Set Example

What Jack selected for today's dinner	Probability of Jack's selecting <u>Chicken</u> for tomorrow	Probability of Jack's selecting <u>Fish</u> for tomorrow	Probability of Jack's selecting <u>Meat</u> for tomorrow
<u>Chicken</u>	.1	.4	.5
<u>Fish</u>	.6	.1	.3
<u>Meat</u>	.4	.5	.1

In this example transition matrix, each row of the table is a conditional probability distribution. The first row is conditioned on Jack having chicken at the current step of the process (which is "last night"). The three probabilities in the rows of the table are *conditional probabilities*. They are conditioned on last night's entrée being chicken. The third entry of the first row (.5) is the conditional probability that Jack will have meat tomorrow night, conditioned on the event (or "given") that Jack fish last night.

Note that each time step of a Markov chain needs its own transition matrix – unless the Markov chain is homogeneous. By definition, a homogeneous Markov chain is one whose set of conditional probability distributions is the same for all time steps in the chain. In such a case, obviously the same transition matrix applies to all steps in the chain.

For a nonhomogeneous Markov chain, provisions must be made to define the transition matrix for each step individually. Most Markov chain modeling applications are homogeneous. Therefore it usually suffices to define a single transition matrix for the entire chain.

However, as we have seen, this is not generally the case in Organodynamics, where most Markov chains will be piecewise homogeneous. Therefore, In Organodynamics, provisions must be made to define several transition matrices for a Markov chain, and to associate each of them with the set of time steps to which they apply.

Independent Probabilities

At this point, we shall examine a special case. This is a Markov chain that has the same conditional probability distribution for all if its conditions. In other words, all of the conditional probabilities for that step are exactly the same distribution for each of the cases involved.

For example, suppose that, in the above example, that Jack essentially ignores (or does not remember) which meal he had for today's dinner when he makes his selection for tomorrow. A good model to represent this situation is to have all three of the joint probability distribution to be the same as each other.

For example, the set of three conditional distributions might be the following.
(For uniformity, let's assume that this set applies for all steps of this process.)

Independent Conditional Probability Distribution Set Example

What Jack selected for today's dinner	Probability of Jack's selecting <u>Chicken</u> for tomorrow	Probability of Jack's selecting <u>Fish</u> for tomorrow	Probability of Jack's selecting <u>Meat</u> for tomorrow
<u>Chicken</u>	.2	.4	.4
<u>Fish</u>	.2	.4	.4
<u>Meat</u>	.2	.4	.4

This table reflects the situation where Jack's choice of tomorrow's meal is not conditioned upon – or doesn't not depend upon – which meal he chose today. Not only does Jack "not remember" what he ate on previous days – he doesn't even remember (or chooses to ignore) what he ate today.

We can say that his choice for tomorrow's meal is *independent* of what he ate today. But we can also say that the situation is *unconditional*. It is not conditioned on what he ate today.

In fact, in this case, we don't really have three conditional distributions. It just looks like we do because we articulated them above in transition matrix format. In actuality, we just have one probability distribution – and it is unconditional.

Thus, what we really have is an *unconditional stochastic model* – even though it appears to be a *conditional stochastic model* (Markov chain).

The Markov Principle

Our first example of Jack's process was actually a very special case of a conditional stochastic process. It was the case when the state upon which the next outcome depended was the current state – *and only the current state*. That is, which of the three joint probability distributions we should use was completely determined by looking at what meal he selected today. We did not need to look back any earlier than today to see what he had selected then.

In fact - and this is significant - even if we *had* consulted earlier days than just today to see what Jack had selected, such information would not have helped us! That is, we would have ended up using exactly the same three conditional probability distributions whether we only looked at today's selection by Jack, or whether we would have looked at earlier days also.

Such a situation is called the Markov condition – and stochastic processes whose conditional probability distributions are conditioned only on the current step are called Markov processes.

So, a Markov process is a conditional stochastic process that depends *only* on the outcomes of the current step. Information about the outcomes of previous steps adds no useful information. In a Markov chain, even if the information about the outcomes prior to the current step is used, it yields the same answers (the same probabilities) as if it had not been used.

This property is sometimes labeled as “memory-less”. It is somewhat a misnomer, because the actual situation about Markov processes is that they *do* need any memory – but their need for memory does not go any further back than the current step.

The point is that, for a Markov chain, even if one used the information about the outcomes of steps prior to the current step it would not produce a different set of probabilities as compared with only using the current step outcomes.

This is the difference between a Markov chain and a conditional stochastic model that is not a Markov chain. To get a better feel of the difference between the two, let’s look at an example of each using our “Jack’s dinner entrée” example.

The first will be Markov and the second will not.

As our Markov example, we shall reuse the first example here already used above. So that the reader will not have to refer back several pages, we shall reprint that example here.

Markov Conditional Probability Distribution Set Example

What Jack selected for today's dinner	Probability of Jack's selecting <u>Chicken</u> for tomorrow	Probability of Jack's selecting <u>Fish</u> for tomorrow	Probability of Jack's selecting <u>Meat</u> for tomorrow
<u>Chicken</u>	.1	.4	.5
<u>Fish</u>	.6	.1	.3
<u>Meat</u>	.4	.5	.1

Thus, we have already exemplified a Markov process with our first example of Jack's dinner habits. So, let us now consider an example of a conditional, but non-Markov, stochastic process – still using Jack's eating habits.

This time, let's assume that only if Jack has had the same meal for two consecutive days does his disdain for that meal kick in. Otherwise, he is just as likely to pick any meal as he is on any other day. In such a case, the process could look like this:

Non-Markov Conditional Probability Distribution Set Example

What Jack selected for yesterday's dinner	What Jack selected for today's dinner	Probability of Jack's selecting Chicken for tomorrow	Probability of Jack's selecting Fish for tomorrow	Probability of Jack's selecting Meat for tomorrow
<u>Chicken</u>	<u>Chicken</u>	.1	.4	.5
<u>Fish</u>	<u>Chicken</u>	.2	.4	.4
<u>Meat</u>	<u>Chicken</u>	.2	.4	.4
<u>Chicken</u>	<u>Fish</u>	.2	.4	.4
<u>Fish</u>	<u>Fish</u>	.1	.4	.5
<u>Meat</u>	<u>Fish</u>	.2	.4	.4
<u>Chicken</u>	<u>Meat</u>	.2	.4	.4
<u>Fish</u>	<u>Meat</u>	.2	.4	.4
<u>Meat</u>	<u>Meat</u>	.1	.4	.5

We can immediately see that the stochastic process generated by these distributions is not *Markov*, because it *does* matter what dinner was eaten prior to the current day.

For example, look at the first three rows. They tell us that it is *not sufficient* to only consult the current day in order to know what the conditional probability distribution is for predicting tomorrow's meal. For example, the fact that the probabilities in the first row are different from those in the second and third rows – even though Chicken was selected on the current day in all three rows – tells us that this process has a memory extending prior to the current day.

Thus, we can see that this process is not Markov. If it were Markov, then the probability distribution for any case that chicken was selected for today would be the same for every other case that chicken was selected for today – and not dependent on any other condition. Moreover, if the process were Markov, this would also have to be true of fish and meat as well as chicken.

Examples of Markov Chains in Biological Systems

In this subsection, we shall provide several examples of both a Markov and a non-Markov process in biology.

A classical example of biological processes that biochemists model with Markov Chains is that of various chemical reactions under conditions of adequate supply of the reactants. Specifically, Michaelis-Menton kinetics is a probabilistic model of enzyme activity involving substrates and enzymes. This theory concerns the dynamics of enzyme activity as they bind to substrates and produce products. These products then bind with more enzymes and produce still more products. These chemical kinetic processes are classically represented as Markov chains. This works as follows: at each time step these reactions move among a fixed set of states – as described by substrate and product concentrations – where from each state there is a probability of moving to each of the other states.

We have already presented an example – in the chapter on Uncertainty in Part I - of a non-Markov, conditional stochastic process. The example being referred to is the spontaneously growing RNA polymer that loops back and attaches itself to one of the existing nucleotides that was already in the chain. Specifically, the newest nucleotide in the chain is the one that attached itself to an earlier nucleotide in the chain. And, you will recall that once this nucleotide bonded to the earlier one, then the probability of which type of nucleotide (A, G, C or T) would be added next to the end of the growing chain began to be influenced by the type of nucleotide that preceded the “partner” of the previous nucleotide. For example, if nucleotide number 147 in the chain happened to loop back and connect to nucleotide number 64, then the selection of nucleotide number 148 would be influenced nucleotide number 63. And that of nucleotide number 149 by the type of nucleotide number 62.

Thus, while the probabilities of which type of nucleotide will be added next are conditioned on which nucleotides appear at certain places in the chain, they are nevertheless not Markov. In order to Markov, the dependency would have to be on the exact previous nucleotide to the one being added – not on the state of some nucleotide further back in the chain. Thus, this process is a non-Markov, conditional stochastic process.

More precisely, the above process begins to behave as described - a non-Markov, conditional stochastic process – only at the point where the growing end loops back and attaches to one of the previous nucleotides in the growing chain. (This was where nucleotide number 147, in the above example, loops back and connects to nucleotide number 64.) Thereafter, from nucleotide number 147 for as long as the chain grows by attaching to previous nucleotides, this process is a non-Markov, conditional stochastic process. However, prior to the point where the loop back attachment occurs, the process is an unconditional stochastic process. Consequently, it is more precise to say that this process is of mixed type.

A more complex example of a discrete non-Markov, conditional stochastic process is that of Mendelian genetics. Mendelian genetics deals with the transmission of genetic traits involving sexual reproduction. It is an organic

process because it describes a sequence of genetic states that change in time. It is a stochastic process because probabilities are involved at every step. Moreover, it is a conditional stochastic process because the probabilities of the outcomes of each step are conditioned on the specific state of the current step. For example, if both “mother” and “father” are smooth Mendelian peas, then the probability distribution describing the outcomes at the next step (generation) is one distribution; but if the mother is smooth and the father is wrinkled, then the probability distribution is a different one. In addition, we only need to consult the current generation to determine the correct probability distribution. So then why is this not Markov? The reason is that the state space of the current step is a pair of peas (mother and father), whereas the state space of the next is single pea. The entire probability spaces are more complicated than what is considered by Markov chain theory; and thus it is not correct to identify this as a Markov process.

And, of course, some stochastic processes in biology are not dependent at all – but rather are independent. That is, they are conditional, not unconditional. These are the kind that the “Fred Hoyle and his Boeing 747 spontaneously-self-assembling” example assumed [Hoyle and Wickramasinghe 2001]. As we have shown, these unconditional stochastic processes also occur in living systems. In fact a “spontaneously-growing nucleotide polymer”, as in the example above, that never loops back to attach to an earlier nucleotide in the chain is such an example. This is true because, at least according to our assumptions, the probability of the “next” nucleotide is never influenced by the outcome of the present one – or any previous one, or of a known condition whatsoever.

Moreover, our “growing RNA chain” example is actually mixed regarding whether its steps are unconditional, conditional Markov or conditional non-Markov. Until the point where it loops back upon itself, it is an unconditional stochastic process. But, at the step after it loops it loops back and reattaches to itself, it becomes a conditional non-Markov stochastic process. Thus, it is mixed.

Thus, unconditional, conditional and mixed processes constitute stochastic processes in biology – and in Organodynamics. And, the conditional stochastic processes are represented by both Markov and non-Markov processes. The fact that these processes are not all independent, is the reason why [Hoyle and Wickramasinghe 2001] makes an unwarranted assumption of stochastic independence everywhere; and what provides Organodynamics with its dance of “ordered chaos”.

Beyond this, we can say that living and non-living systems even have mixed stochastic process – with some of the steps being any of the three types, as we exemplified above.

However, in Organodynamics we shall concentrate on pure conditional stochastic processes, since they provide much of the opportunities to move from high to low Shannon entropy through the life cycle of such a process, and are of primary interest. In fact, since Markov Chains are the simplest and most tractable of the discrete conditional stochastic processes, we shall use them often in what follows to exemplify the representation of organic processes.

Promoting a Markov Model to an Unconditional Model

In this chapter, we shall address the first stage of the *uncertainty model gradient* – how to promote a Markov chain model to an unconditional stochastic model. Recall that the reason one wants to do this is to utilize additional information to decrease the degree of uncertainty represented by a dynamical model. In the uncertainty model gradient, we start out with a Markov chain and we hope to promote it to an unconditional stochastic model whenever we gather enough additional information.

A conditional stochastic process – such as a Markov chain – gives us a way of constructing a model of a dynamical system at a stage when we have quite limited certainty about the process.

Even though, at this stage, we don't yet know enough about a dynamical system to be able to describe a probability distribution for each of its step, as long as we can 1) identify a partitioning of the conditions that constitute the outcomes, and 2) know a conditional probability distribution for each of those conditions; then we are in a position to develop a Markov model.

Later, when we gain more information, we can promote this Markov model – with multiple conditional distributions - to an unconditional stochastic model, where we have a single unconditional probability distribution to model each time step.

Of course, to do this, we must have some sensible way to “convert” this set of distributions to a single distribution.

Representing Probability Distributions as Vectors and Matrices

It turns out that we can use matrix algebra to facilitate the promotion of a Markov model to the next stage of the uncertainty model gradient – the unconditional stochastic model.

The general approach is to use matrix algebra to combine the “additional information” received with the set of unconditional probability distributions to produce a single new unconditional probability distribution. Of course, this is done for each time step of the Markov chain to produce a new time step of the unconditional stochastic model.

We already have a matrix representation of the conditional probability distributions of a time step of a Markov chain. It is called the *transition matrix* of the time step.

All we need then is matrix representations of the “additional information” and of the new unconditional probability distribution. Then, hopefully, we can use matrix algebra to “combine” the “additional information” matrix with the transition matrix to produce the “unconditional probability distribution” matrix.

Such an approach will let us promote each step of a Markov chain to a corresponding step of an unconditional stochastic model – one step at a time.

Initial Conditions

One way that “additional information” about a dynamical process can be articulated is in the form of some “initial conditions” for the first step of the process.

These “initial conditions” specify how to use the set of conditional probability distributions of the first step of the Markov chain to determine a single unconditional probability distribution for that first time step.

Deterministic Initial Conditions

A simple approach to this is that the additional information actually selects one of the conditional probability distributions to be used as the unconditional distribution. Thus, where there were initially multiple conditional distributions, now there is only one unconditional one. Obviously, this act of selection “transforms” an entire set of conditional distributions to a single unconditional distribution that is then used to model the step.

In this way, the Markov model of the first time step has been promoted – in a very simple manner – to an unconditional stochastic model.

Of course, it would be highly desirable to use mathematical structures and operations to represent this action. As indicated above, we shall show how to use matrix algebra to accomplish this *promotion*.

We have already shown how to represent the entire set of conditional probability distributions for a step of a Markov chain as a matrix – the transition matrix. Thus, it would be convenient if we could also represent a set of *initial conditions* in matrix notation. If so, then we could then use matrix algebra to produce an “answer” that is also a matrix. Recall that, in this case, the “answer” would represent the “selected” conditional probability distribution.

This can easily be done as follows.

To represent the *initial conditions*, define a matrix with one row that we shall call an *initial conditions row vector*. (A *vector* is a one-row or one-column matrix.) Moreover, this vector will have the same number of columns as the transition matrix has rows. Now, define the entries of this vector such that exactly one of them is one (1) and the remainders are zero (0).

Next, perform a matrix multiplication of the initial conditions row vector (C_i) and the transition matrix (A_i) for the time step (1). The result, shown below, is another row vector.

$$\begin{array}{r}
 A_1 = \begin{array}{cccc}
 .5 & .2 & .2 & .1 \\
 .3 & .2 & .1 & .4 \\
 .4 & .4 & .2 & .0 \\
 .3 & .4 & .2 & .1
 \end{array} \\
 \\
 C_1 = \begin{array}{cccc}
 .0 & 1.0 & .0 & .0
 \end{array}
 \end{array}$$

Now, let's multiply C_i and A_i .

$$C_1 * A_i = \begin{matrix} .3 & .2 & .1 & .4 \end{matrix}$$

Notice that the product $C_1 * A_i$ is the second row of A_1 .

Thus, the position of the "1.0" value within the initial conditions row vector (C_1) "selects" which of the conditional probability distributions to use as the unconditional probability distribution for the time step when *promoting* a Markov chain step to an unconditional stochastic model within the *uncertainty model gradient*.

Stochastic Initial Conditions

Notice that we are treating our set of *conditional probability distributions* as, itself, a sample space whose members are probability distributions. Moreover, our *initial conditions row vector* is actually a *probability vector* – meaning that its entries are non-negative and sum to one (1). Therefore, an initial conditions row vector represents a *probability distribution* of probability distributions!

In this sense, our initial conditions row vector is a "meta-probability distribution". However, since we have so far made a "rule" that says that exactly one of its entries is 1 then it must represent the deterministic probability distribution (of probability distributions).

However, there is no reason why we can't relax this rule – as long as we insist that an initial conditions row vector be a probability row vector! In fact, if we allow this relaxation, then we can see that a probability row vector can act as a set of "weighting factors" on the conditional probability distributions in the transition matrix for the time step.

In other words, if we relax the rule that insists that an initial condition row vector have one value of "1" and other values of "0", and merely require that it be a probability vector, then we have moved to a much more general and powerful situation in which it can be used to calculate a single resulting unconditional probability distribution that is a linear combination of the conditional distributions represented by the transition matrix.

Let's look at an example of this. We shall use the same example transition matrix A_i as above, but shall use a different initial conditions row vector.

$$A_1 = \begin{matrix} .5 & .2 & .2 & .1 \\ .3 & .2 & .1 & .4 \\ .4 & .4 & .2 & .0 \\ .3 & .4 & .2 & .1 \end{matrix}$$

$$D_1 = \begin{matrix} .1 & .1 & .7 & .1 \end{matrix}$$

Now, let's multiply C_i and A_i .

$$D_1 * A_i = \begin{matrix} .39 & .36 & .19 & .06 \end{matrix}$$

Notice that the product $D_1 * A_1$ is a linear combination of the rows of A_1 . And also notice that is a probability vector – that is all entries are non-negative and sum to 1.

Now, this row vector $D_1 * A_1$ represents our “answer” – which is the unconditional probability distribution for step 1. Therefore, we shall name this new vector the *unconditional probability row vector*.

At this point, then, we have converted the first step (step 1) of a Markov chain from a Markov model (containing an entire set of conditional probability distributions) to an *unconditional stochastic model* (containing exactly one unconditional probability distribution).

Propagation of Initial Conditions throughout a Markov Chain

Notice that the *initial conditions row vector* served as a set of initial conditions for the entire Markov chain. However, it was only applied to the first time step of the chain – to produce an *unconditional probability row vector*.

This *unconditional probability row vector* then was substituted (promoted) for the entire first time step of the Markov chain to become an the *unconditional stochastic model* for the first step of the chain. However, all of the remaining time steps (2+) remain as Markov models.

But notice that the *unconditional probability row vector* also happens to serve the same role to time step #2 that the *initial conditions row vector* serves for step #2.

That is, the *unconditional probability row vector* of step #1 works as an *initial conditions row vector* for step #2. Of course, this process continues recursively for every time step of our finite step stochastic process.

Thus, the first stage of the *uncertainty model gradient* – promoting a Markov chain to an unconditional stochastic model - can be recursively calculated by 1) obtaining a set of initial conditions, 2) applying these against the transition matrix of the first time step via the matrix algebra discussed above, and 3) interpreting the *unconditional probability row vector* of time step 1 produced by this operation as an *initial conditions row vector* for step 2 and repeating 1) of this paragraph.

As well, this procedure can be recursively applied through each time step of the stochastic process until the entire process has been promoted from a finite step Markov chain to a finite step unconditional stochastic model.

This promotion is simplified as whenever conditions of homogeneity apply – because the same transition matrix applies to multiple time steps. For Organodynamics, piecewise homogeneity is assumed. Therefore, there will be contiguous time steps for which the same transition matrix applies. This will simplify the situation as compared to having to use a unique transition matrix for every time step.

Stochastic General Conditions

Initial conditions have the convenient property that they can apply to the first time step of a Markov chain, and then have the results recursively applied to each succeeding step in the chain until the entire chain is promoted to an unconditional stochastic model – as demonstrated in the previous subsection.

However, in general, any time step may be “influenced” by conditions that can occur anywhere – maybe external to the Markov chain, or may being “felt” from multiple previous steps of the same chain.

There is the expectation that these influences may be able to affect the promotion of a time step of the chain – just as the results of the promotion of the previous time step influenced the results for initial conditions.

This text shall not attempt to identify any of these types of circumstances. We shall only point out the possibility of their occurrences – and also suggest the modeler consider ways to represent such influences as matrices, and ways to apply matrix algebra to perform such promotions.

More generally, it is also expected that less linear approaches may be more appropriate for the modeling of some dynamical systems.

Useful Analogy

In this way, by using an entire family of generating conditions, a conditional finite stochastic process – such as a Markov chain - can be seen as having a whole collection of specific finite unconditional stochastic processes “inherent within it”. And these conditional finite stochastic processes are “just waiting for the right *generating conditions* (or *dependencies*)” to be come along in order to transform the Markov chain into some particular unconditional stochastic processes.

In other words, a Markov chain can be analogized as a “family of potential unconditional stochastic processes”, one of which will “be realized” when the right set of conditions (initial or otherwise) come along to establish it as “the one”.

Of course, within the framework of the *uncertainty model gradient*, this means that the relationship between a *Markov model* and an *unconditional stochastic model* is one of “a set of potential things” waiting to be realized whenever the right dependency conditions manifest.

Of course, this also describes the relationship between an *unconditional stochastic model* and its subsequent *realized deterministic model*.

Of course, the difference is that, inside of the *Markov model* is a set of probability distributions, only one of which will be made manifest. On the other hand, inside of an *unconditional stochastic model* is a set of possible outcomes (members of the probability distribution), only one of which will be made manifest.

A useful visual analogy of both situations is that of a “conduit” that contains many “wires”. Each “wire” can be “realized” by the proper application of certain generating conditions.

Model Development: Building the First Two Approximations

We have already built up enough intellectual equipment for the modeler to be able to construct the first two levels of Organodynamic approximation models of the organic system being modeled.

As already discussed, an Organodynamic model is eventually a structure named an Organodynamic web. This structure first appears whole in the fourth approximation. The fifth and sixth approximations add additional capabilities to make the structure more robust.

An Organodynamic web consists of many *edges*, each of which consists of many *segments* chained together. In addition, these many edges are connected together at joints called *nodes*. Taken together, these edges and nodes form a network structure – the Organodynamic web structure.

The first three approximations gradually build toward this structure, even though they don't quite get to a full-fledged Organodynamic web. This honor is reserved for the fourth approximation.

The first approximation builds a single *segment*. The second approximation builds a single *edge* by building another couple of *segments* and chaining them together with the first segment. The result is an *edge* with three segments. The third approximation builds additional edges and connects them together by joining them with a *node* – thus producing a partial network called a *graph*.

As such, these early approximations are designed to have modeling value of their own; however none of them is expected to represent a whole and comprehensive “organism”. That capability is reserved for fourth approximation, where the Organodynamic is introduced.

Let us proceed now to develop the first two approximations.

Model Development: Building the First Approximation

Description of the Approximation

The *segment* is a contiguous set of homogeneous Markov time steps.

Because it is a linear sequence of time steps, we analogize it with the name *segment*, and speak of it as though it were a line segment in a graph. This is a useful analogy, but the modeler should not forget that in actuality a segment is a sequence of analytic and probabilistic entities and that are associates with a time sequence.

Nevertheless, within this graphical analogy, a segment can form a “connection” between two nodes in an Organodynamic web. Most often however multiple segments “strung together” form “connection”. So, a segment is the simplest “connection” between nodes in an Organodynamic web.

(As described above, multiple segments “strung together” form an *edge*. The *edge* constitutes the second approximation of our modeling methodology.)

As with any of the six Organodynamic approximations, all three models from the *uncertainty model gradient* must be developed for this approximation. These three model types are discussed below or the first approximation. Remember that these three model types form an uncertainty gradient, and each of the three should be developed in the proper order.

Actually, it is more correct to say that the Markov model type is initially developed for an approximation level, and then it is *promoted* to the *unconditional stochastic model* type when certain additional information is made available to the modeler. Then, subsequently, the *unconditional stochastic model* is once again *promoted* to the *realized deterministic model* type as each time step is realized.

Let us look in more detail at each of these three model types for the first approximation – and at the promotion process.

The Markov Model

Once the modeler has developed a Markov transition matrix that applies for some number of contiguous steps, then a first approximation – a *segment* – has been constructed for that number of contiguous time steps.

However, in order to develop the transition matrix, several things must be in order. First, the sample space of conditions (system organizations) must have been well defined, and this sample space must apply to each of the contiguous steps of the segment being built.

Next, a conditional probability distribution must be developed for each of these conditions (system organizations). This distribution describes the probabilities that each of the organizations in the sample space will be realized at the next time step - given that the condition is manifest at the current time step.

Of course, there will be one of these conditional probability distributions conditioned on each of the organizations in the sample space.

Once these conditional probability distributions are known, then the transition matrix is developed. This transition matrix must apply to all of the time steps in the segment. This is what it means for the contiguous set of time steps to be homogeneous, and to thus be a segment.

At this point the Markov model if the segment is completed.

The Unconditional Stochastic Model

The Markov model of the approximation is *promoted* to the unconditional stochastic model whenever enough additional information is obtained to be able to resolve which of the conditional probability distributions of the Markov model – or which combination of them – is to be used as the single unconditional probability distribution.

This promotion is accomplished on a time step by time step basis. Therefore, for much of its life, the process will consist of a mixture of models, with the first several contiguous steps being in the *realized* state, the second several contiguous steps being in the *unconditional* state, and the remainder of the steps being in the *Markov* state. In the beginning, all steps are in the Markov state, but are gradually promoted to the other states until all are in the realized state.

One way to promote a time step that is in the Markov state to a time step that is in the unconditional state is by use of a set of initial conditions. This process is described above, and will not be repeated here. However, the reader will recall that it consisted of representing the initial conditions as a vector that is then multiplied by the transition matrix to produce another vector. The second vector defines the single probability distribution that is the unconditional model for the time step.

The Deterministic Realized Model

Once the outcome of a time step has been realized, then the selected possibility is known. At this point, the model of the time step ceases to be a probability distribution, and becomes exactly one of its sample space members – the one that is realized.

Development the Markov Model for the Approximation

Let's apply the previous remarks to the development of the *canonical* biological system example already presented. We shall first discuss developing a *Markov model* for this system. Then we shall discuss promoting the Markov model to an *unconditional stochastic model*; then finally to promoting the *unconditional stochastic model* to a *realized deterministic model*.

It will be helpful to recall at this point that for the first approximation, the two stochastic models will be homogeneous. In this approximation, we are building a single *segment* of our (eventual) Organodynamic web; and a segment is time-homogeneous – meaning that the same probability model is at work for all time steps in the segment.

Selecting which Model Aspects will be Components

The first step is to determine which aspect of the system being modeled constitutes the *components* of the population. This will determine what our system population is. In our canonical biological model, we selected the *atoms* of the system as our *system components*. As modelers, we could have started

at a higher level of abstractions (organization) if we had so desired. For example, we could have decided that a specific list of simple molecules would be our components. However, let's stay with our canonical system and choose atoms as components.

It is also necessary to recall that our first approximation will confine itself to a closed system of the components – where no components enter or leave the population during the segment. In fact, we are assuming that since the population does not change during the segment that it is reasonable to assume that the probabilities will stay the same throughout the segment – thus justifying the *homogeneity assumption* above.

Selecting which Model Aspects will be Duples and what an Organization Represents

The second step is to determine how the notion of *organization* applies to our canonical model. One might intuit that a *molecule* in our canonical system will be represented by an *organization*. This is close to being correct – but it is not! Let's look into the matter of what aspect of our canonical system really is represented by an *organization* in the Organodynamic framework.

Recall that an *organization* is a set of sets of *duples* of components. We used the term *cluster* to name these sets of duples. Therefore, we can also say that an *organization* is a set of *clusters*.

It is going to turn out that it will work better if a cluster in our modeling machinery represents a molecule, and that an *organization* represents a specific collection of molecules.

We must also explain the duples that are members of the cluster. These duples have the purpose of logically pairing two components that enjoy some *relationship* that is of modeling significance to the modeler. Thus, the first question we must ask about the second step is “What does it mean in our canonical system for two components to be paired in relationship?”

We have a lot of possible choices for that answer within our canonical system, but the one that jumps out at us is “chemical bonding between atoms”. Of course, there are many kinds of chemical bonding between atoms – some strong, some weak and some of intermediate strength.

It is reasonable to want to model all of these types of these chemical bonds in the model that we are constructing. For example, it would be highly desirable to represent each of these types of bonds distinctly: covalent bonds, ionic bonds, hydrogen bonds and Van der Waals forces.

However, to keep matters as simple as possible in this text, we are going to limit our types of relationships between system components to one. One reasonable choice would be to decide that two atoms are “related” if any of these types of bonds exists between them. However, we shall instead decide that two atoms are related if there is currently a covalent bond between them. (Either choice is legitimate and is left to the discretion of the modeler.)

(The reader will recall that we suggested mechanisms for extending this nomenclature to accommodate these requirements. But, in the interest of pedagogy, we shall not make these accommodations in this text, and leave the issue of these extensions to further research.)

Remember, what we are trying to achieve in this subsection is to make a proper modeling choice as to what a *system* organization is in our conical biological system of molecules.

Lets summarize the modeling decisions that we have made for this above:

Framework Mechanism	Canonical Example
Component	Atom
Population	The set of all atoms in the closed set
Duple	Covalent bonds between two atoms
Cluster	A molecule
Organization	A partitioning of all of the covalently-bonded atoms in the population into a set of molecules.

Identifying the Sample Space

For our model of the canonical biological system, we have now mapped various elements of our subject system to the mechanisms of the Organodynamics framework.

But, an Organodynamic model is a probability model. Therefore, we somehow need to form a probability distribution from these elements. The first step in identifying a probability distribution is to ask the question "What type of things will have probabilities?"

Next, we must enumerate *the set of all possible things like that*. Such a set is called a *sample space*, and its members are called *events* and a number of other things.

Thirdly, we must assign probabilities to each of these events. Once we have done these three steps, then the result is a probability distribution.

In Organodynamics, we routinely select a *system organization* to be a *sample point*, or *event*. Therefore, in this example, an *organization of atoms*, as we defined it in the previous subsection, is an event.

And, the set of *all possible organizations* of our closed population of atoms forms our sample space. This is a very large sample space, indeed – even when the closed population of atoms is small. The number of such organizations is the same as the number of ways that this set of atoms can form pairs such that any number of these atoms can be unpaired, and no atom is paired with itself. But, as we have already said: "Welcome to the world of comprehensive complex system modeling!"

Of course, it is not actually our problem to count this number of *organizations*, but to *list them*.

Lets summarize where we are at this point by extending the table of modeling elements that we presented in the previous subsection and adding a couple of new elements to it. The new elements are depicted in bold style.

Framework Mechanism	Canonical Example
Component	Atom
Population	The set of all atoms in the closed set
Duple	Covalent bonds between two atoms
Cluster	A molecule
Organization	A partitioning of all of the covalently-bonded atoms in the population into a set of molecules.
State space	The set of all such organizations - exactly one of which can occur at any point in time.
Sample space	Another name for the state space. This name carries with it the implication that probabilities will be assigned to each of the <i>organizations</i> that comprise this state space, and that a probability distribution on this state space will result.

Developing the Probability Distribution

In order to have a probability distribution, we need a sample space, and we need to have probability assignment to all of the members (organizations) of the sample space.

Thus, this step involves assigning probabilities to each of the organizations in the sample space. This is not easy. There are a number of ways that this can be approached. Empirically, they can be measured or estimated. A second approach is to reason about the nature of the sample space and the canonical system, and then select a theoretical distribution and fit it to the sample space.

None of these are easy. But again, we are developing a comprehensive model of a complex system.

So, at the end of this step, we have a probability distribution.

More specifically, what we really have is an *unconditional stochastic model*. This is because we do have a single probability distribution.

But this is the *second* model in the uncertainty model gradient. If the modeler has enough information to develop this unconditional probability distribution,

then the *first* step of the uncertainty model gradient (the Markov model) will have been skipped.

However, in most cases there is initially not enough information as yet available to be able to immediately develop the *unconditional stochastic model* first. And it is more tractable to develop the Markov model first, and then to subsequently promote it to the *unconditional stochastic model* later.

In any event, if the Markov model is skipped, and the *unconditional stochastic model* is developed first, then we have added one more element to our table – the probability distribution. (We shall wait until we have added more elements before presenting this completed table again.)

Developing the Markov Transition Matrix

In many, if not most, cases, it is unrealistic if not impossible to develop the single probability distribution that unconditionally models all organization in our sample space. We just may not be able to empirically develop all of these probabilities.

More often, there will be a number of project participants all of whom have a partial view of this sample space. It may even be the case that, taken all together, each possible organization is represented in one of these views – so that all organizations are “covered”.

If this is true, then each of the participants can conduct an experiment using the subset of system organizations that they can “see”, and ignoring the ones they cannot see. Such an experiment can isolate on one of their “visible” organization at a time. The participant could observe to see “what happens” at the next step whenever the specific target *organization* occurs at the current step.

The outcome of such an experiment would be a conditional probability distribution based upon the specifically observed organization.

If every participant conducted the same type of experiment and developed conditional probability distributions for the system organization that they can see then the overall result would be a set of conditional probability distributions for all possible conditions (system organizations).

And this is precisely what is needed to have a Markov model for a single time step.

Of course, what we have at this point is a complete set of conditional probability distributions. We have discussed that it is more convenient to represent this set of distributions as a *probability matrix* call a *Markov transition matrix*. Making this representation is the final act of this time step.

At this point, then, we have added an additional element to our table: the Markov transition matrix.

Determining if a Markov Model is warranted

The previous step produced a *conditional probability model* for a single time step because it predicted the outcome of the *next* time step *conditioned* on so the outcome of some *previous* time steps. In fact, the model produced was specifically a Markov model, because it only utilized the current time step, and ignored any time steps prior to the current one.

This *ignoring* may have been warranted – or it may not have. If taking the outcomes of any previous time steps into consideration would have made a difference to the prediction of the *next* time step, then ignoring these previous time steps would have been a mistake. When this happens, we do not have a Markov model. If taking them into consideration would not have changed the answer – would not have changed the conditional probability distributions, then we *do* have the Markov condition, and we can ignore the previous time steps.

It is up to the modeler to determine if the Markov condition holds. In this text, for the sake of simplicity, we are only dealing with the case that the Markov condition holds, and leaving the more general case for further research.

In any event, at this point we have actually have a single Markov transition matrix that models the first time step of the Markov chain that models the *segment* – which is the first approximation.

This step adds no new elements to our table. It merely justifies that using a Markov transition matrix is warranted.

Determining the Transition Matrix for the Remaining Time Steps

The final step of developing the Markov model is to develop transition matrices for all of the other time steps of the Markov chain that we are building to represent a single *segment*.

However, a *segment* is, by definition, homogeneous. This means that the same Markov transition matrix applies to all of its time steps.

Thus, we shall reuse the Markov transition matrix developed for the first time step for all of the other time steps.

And we are done with the development of the Markov model for the first approximation – the *segment*.

Summarizing the Model Mechanisms for the First Approximation

We shall now complete the table of mechanisms for this example that we have been building by adding any elements that we have discussed above but not yet recorded in the table.

Framework Mechanism	Canonical Example
Component	Atom
Population	The set of all atoms in the closed set
Duple	Covalent bonds between two atoms
Cluster	A molecule
Organization	A partitioning of all of the covalently-bonded atoms in the population into a set of molecules.
State space	The set of all such organizations - exactly one of which can occur at any point in time.
Sample space	Another name for the state space. This name carries with it the implication that probabilities will be assigned to each of the <i>organizations</i> that comprise this state space, and that a probability distribution on this state space will result.
Unconditional probability distribution	Assign probabilities to each of the <i>organizations</i> in the sample space in such a way that they sum to 1. The result will be an unconditional probability distribution for the sample space. This will comprise the second model (the unconditional stochastic model), and the first model (the Markov model) will have been skipped. If there is not enough information to do this at this time, then proceed to the next mechanism where the Markov model is developed first.
Set of conditional probability distributions	For each <i>organization</i> in the sample space, develop a conditional probability distribution for the outcome of the <i>next time step</i> , given that the organization in question was realized at the current time step. The result is a set of conditional probability distributions.
Markov transition matrix	The conditional probability distribution set can be represented as a probability matrix – the Markov transition matrix. While this matrix describes the stochastic behavior of only a single time step, it is all we need for the entire <i>segment</i> of time steps, since each time step has the same stochastic behavior (due to the homogeneity property of a segment.)

Promoting the Markov Model for the Approximation

At this point, we have developed the first model for the first approximation – the Markov model. We now need to promote this model to the *Unconditional Stochastic Model*, and then promote that to the *Realized Deterministic Model*.

The Unconditional Stochastic Model

Our Markov model is very useful – but it also still contains a great deal of uncertainty.

For example, for any time step we have a whole set of conditional probability distributions – but we don't even know which one to use to predict the outcome of the next time step.

However, it may come to pass that we receive specific additional information that tells us exactly what the outcome is for the current time step. In that case, we know exactly which of the several conditional probability distributions for that time step.

This type of information is called an *initial condition*. We have demonstrated above how to use an initial condition to select the correct conditional probability distribution from the Markov transition matrix for the step. Such a selection effectively promotes this step from a Markov model to an unconditional stochastic model.

Another type of initial condition can also occur. This is when we ascertain the probabilities that each of the conditional probability distributions for the step will be selected. This is a more general case of the above.

In both of these cases of initial conditions, these conditions can be represented as a vector and that vector can be multiplied by the Markov transition matrix for the step to produce another vector. The final vector represents the new unconditional probability distribution for the time step.

The Realized Deterministic Model

The unconditional stochastic model is a sequence of unconditional probability distributions, one for each time step of the model.

However, eventually, the outcome for each time step becomes known. This is modeled by changing the probabilities of the distribution for the time step so that the probability of the realized sample point (organization) has a value of one (1), and all of the other sample points have probability of zero.

This is the realized deterministic model for that time step.

When this is repeated for all time steps, then the model is completely deterministic.

At this point in time, we developed all three models of the uncertainty model gradient (unless we were able to skip the first model). Thus, we shall update our table with these additional two elements. We shall also abbreviate some of the existing entries to make the table shorter.)

Framework Mechanism	Canonical Example
Component	Atom
Population	The set of all atoms in the closed set
Duple	Covalent bonds between two atoms
Cluster	A molecule
Organization	A partitioning of all of the covalently-bonded atoms in the population into a set of molecules.
State space	The set of all such organizations - exactly one of which can occur at any point in time.
Sample space	Another name for the state space. This name carries with it the implication that probabilities will be assigned to each of the <i>organizations</i> that comprise this state space, and that a probability distribution on this state space will result.
Unconditional probability distribution	Assume that this step is done below at the row named "Unconditional Stochastic Model".
Set of conditional probability distributions	For each <i>organization</i> in the sample space, develop a conditional probability distribution for the outcome of the <i>next time step</i> , given that the organization in question was realized at the current time step.
Markov transition matrix	The conditional probability distribution set can be represented as a probability matrix – the Markov transition matrix.
Unconditional Stochastic Model	Assign probabilities to each of the <i>organizations</i> in the sample space in such a way that they sum to 1. The result will be an unconditional probability distribution for the sample space. This will comprise the second model of the uncertainty model gradient.
Realized Deterministic Model	The time step has been realized and all uncertainty regarding its outcome has been removed. Therefore, exactly one organization in the sample space has a probability of 1, and all the others have probabilities of 0. Thus, the state of this time step can be simply represented by this realized system organization.

Mixed Models

When an Organodynamic model for a canonical system is first defined, all of its time steps are modeled by a Markov transition matrix. And, since we are dealing with the first approximation, the *segment* – that is homogeneous, then the same transition matrix applies to all time segments.

However, as time passes, the first time step is promoted to the unconditional stochastic model. Subsequently, other time steps are sequentially promoted to the unconditional stochastic model. Thus, the process will become mixed, with some number of early and contiguous time steps being of the unconditional stochastic model type and the remaining time steps being of the Markov type.

Subsequently, some of the early time steps will be promoted to the realized deterministic model type. At this point, the mixture will be of three types. Some early contiguous set will be of the realized type, followed by some other contiguous set of time steps that will be of the unconditional stochastic type. Then final, a last set of contiguous time steps will be of the Markov type.

Eventually, all time steps will be promoted to the realized type.

Model Development: Building the Second Approximation

Description of the Approximation

The second approximation emerges whenever it is discovered there are some time steps within a Markov model of the dynamical system being modeled that exhibit different probability characteristics from the previous time steps. In other words, the Markov transition matrix for the previous time step no longer represents the correct probabilities for new time steps.

In or canonical biological example of the closed system of atoms, this eventuality generally occurs whenever the population of atoms is “opened up” and new atoms are allowed to flow into the population – and existing ones are allowed to flow out.

When this happens, there is at least one time step in the process that requires a distinct transition matrix. The general case is that the process can be partitioned into a finite set of contiguous time steps, each group of which is correctly modeled by a shared transition matrix.

Such a Markov chain is described as *piecewise homogenous* and is called an *edge* in Organodynamics.

An *edge* is the second level of approximation in the Organodynamics modeling methodology.

For this example, we shall assume that our second approximation will consist of a single *edge* that is comprised of three *segments*.

Of course, this second approximation must also proceed through the three stages of the *uncertainty model gradient*. The procedure for this is articulated

exactly as it was in the immediately preceding subsection for the *segment* approximation.

Since our Organodynamic methodology develops the six approximations incrementally, we shall describe how we need to change or enhance the *first approximation* (the segment) presented above in order to achieve its second approximation (the *edge*).

The Markov Model

The Markov model of the first approximation resulted in a single Markov transition matrix for the segment. Since we are assuming that our *edge* in this second approximation is comprised of three *segment*, then our Markov model will result in three Markov transition matrices.

The Unconditional Stochastic Model

The Markov model of the approximation is *promoted* to the unconditional stochastic model whenever enough additional information is obtained to be able to resolve which of the conditional probability distributions of the Markov model – or which combination of them – is to be used as the single unconditional probability distribution.

Since our example process has three segments, the probability distribution will remain the same for all steps of a single segment of the edge, but it will change on the first time step of each new segment.

The challenge will occur when developing the probability distribution for the initial time step of each segment of the edge. This challenge is that constraints are at work between the two time steps owing to the dependencies between two consecutive steps.

The Deterministic Realized Model

Once the outcome of a time step has been realized, then the selected possibility is known. At this point, the model of the time step ceases to be a probability distribution, and becomes exactly one of its sample space members – the one that is realized.

Development Guidelines for the Approximation

Selecting which Model Aspects will be Components

This issue will already have been decided during the construction of the first approximation, but must be continued in the second and all remaining approximations. Obviously, this continuity will continue across segment boundaries within the edge being modeled by the second approximation.

Selecting which Model Aspects will be Duples and what an Organization Represents

This issue will already have been decided during the construction of the first approximation, but must be continued in the second and all remaining approximations. Obviously, this continuity will continue across segment boundaries within the edge being modeled by the second approximation.

Identifying the Sample Space

This issue will already have been decided during the construction of the first approximation, but must be continued in the second and all remaining approximations. Obviously, this continuity will continue across segment boundaries within the edge being modeled by the second approximation.

Developing the Probability Distribution

As with the first approximation, this step is performed at this point if and only if the modeler has enough information about the state space to be able to assign probabilities to each state (condition, organization) in it.

If so, then the second model type in the uncertainty model gradient (the unconditional stochastic model) is being developed by this action; and the first model type (the Markov model) will have been skipped.

More often, however, the modeler will not have enough information at the beginning of the modeling exercise to be able to skip the Markov model, and will not be able to perform this action at this time.

Assuming that the modeler is able to skip the Markov model and develop this second model at this time, then there are several ways that she can proceed. Two of them are 1) empirical observation of probabilities, and 2) estimation of these probabilities by theoretical distributions.

If the first approach is taken, then the observations should reveal that the distributions change at segment boundaries within the edge. If the second approach is taken, then the application should suggest good reasons why the theoretical distributions change on the segment boundaries.

Developing the Markov Transition Matrix

Assuming that the first model of the uncertainty gradient model – the Markov model – is not skipped, then we shall proceed with this action.

You will want to reuse, or update, the transition matrix developed for the first segment in the first approximation. The problem now is to develop two additional transition matrices – one for each new segment in our example application.

These additional transition matrices will be developed in the same manner as the first one was during the first approximation.

However, in our canonical biological example, remember that what has changed between the first segment of our edge and the second is fundamentally that some of the atoms in the population have left and others have entered the system.

Of course, this has affected the set of possible *organizations* of these atoms into clusters of molecules. Nevertheless, some of the new *organizations* will still be around in this new second segment. And, this fact will have an affect on the conditional probability distributions – because some of the transitions from one *organization to another* from the current time step to the next time step will *still be defined*.

Another way of saying this is that, just because a few new atoms have entered the space and others have left, nevertheless some of the exact same chemical reactions can still occur!

Of course, the probabilities will most likely have changed. But the interdependencies still remain for many of these “organizations” from one time step to the next.

Thus, the challenge is to ensure that any dependencies that need to be preserved across segments within this *edge* are preserved.

We shall not delve into this issue here, but it is appropriate to suggest it as an avenue as an avenue that deserves further research.

Determining if a Markov Model is warranted

This issue will already have been decided during the construction of the first approximation, but must be continued in the second and all remaining approximations. Obviously, this continuity will continue across segment boundaries within the edge being modeled by the second approximation.

Determining the Transition Matrix for the Remaining Time Steps

This section, overall, has addressed this issue.

The Unconditional Stochastic Model

This issue will already have been decided during the construction of the first approximation, but must be continued in the second and all remaining approximations. Obviously, this continuity will continue across segment boundaries within the edge being modeled by the second approximation.

The Realized Deterministic Model

This issue will already have been decided during the construction of the first approximation, but must be continued in the second and all remaining approximations. Obviously, this continuity will continue across segment boundaries within the edge being modeled by the second approximation.

Mixed Models

This issue will already have been decided during the construction of the first approximation, but must be continued in the second and all remaining approximations. Obviously, this continuity will continue across segment boundaries within the edge being modeled by the second approximation.

Uncertainty versus Organization in Organodynamics

It is important in Organodynamics, and in OCS, to make a distinction between *organization* and *uncertainty*. While this distinction may be insignificant or even blurred, conflated or equated in certain other disciplines, it is fundamental to Organodynamics.

Since making this distinction may be unfamiliar or unexpected to some readers, we shall spend some time establishing it and explaining its importance in Organodynamics.

The Need for the Distinction in Organodynamics

The reason that the distinction between uncertainty and organization is significant in Organodynamics is that there is a need for each of these to vary against the other. That is, both uncertainty and organization can increase, decrease or stay the same as the other increases, decreases or stays the same.

The Meanings of Organization and Uncertainty in Organodynamics

In order to exemplify this distinction, let me first give a brief description of each. Below I shall define these distinctions more thoroughly.

In Organodynamics, the concept of *organization* represents relationships between components. As such, *organization* is a static concept. It captures relationships among components that occur within a single moment in time.

Thus, in Organodynamics, *organization* means the structure, or arrangement, of a collection of components at a specific time. In Organodynamics, we only have a vague undefined notion of “highly organized” and “highly disorganized”. We need to define a measuring function for this concept, but at this time we have not done so. Thus, we must be content at this time to assert that an organization is either “highly organized” or “highly disorganized”.

Of course, the organization of a single system can change across multiple time steps. However, in Organodynamics, and in OCS, such a change is not defined as *organization*. Rather, it is regarded as a *change in organization* across time. This change is captured by a different mechanism – the stochastic process – that is discussed later in this text.

It is true that some investigators [Morowitz 1992] define their notion of “organization” dynamically. In their parlance, *organization* is process-oriented and is appropriately measured across time. This is entirely acceptable since

they are defining the term *organization* in the manner of their own needs. However, Organodynamics has the need to define *organization* as above. The reader must be advised of this usage.

By *uncertainty* is meant the degree to which the outcome of an event at some future time *cannot* be predicted. Fortunately, we do have a measuring function for uncertainty. It is Shannon's notion of *entropy*. As we have seen, entropy is technically defined on a finite probability distribution. Therefore, in order to measure the uncertainty of a phenomenon, we must have modeled it with a probability distribution.

Organodynamics has chosen to use the notion of *uncertainty* to describe an attribute of the change between one time step and the next. Such a usage, then, is *dynamical* in nature. Organodynamics does not claim that the concept of *uncertainty* is necessarily a dynamical, as opposed to static, concept. However, Organodynamics has chosen to apply it in such a manner that it results in describing a particular dynamical property – namely the uncertainty involved in transitioning from one time step to the next.

Examples of the Orthogonality of Uncertainty and Organization

Let me now exemplify how these two ideas can vary independently of each other.

First, let's see how they can vary monotonically with both increasing. For example 1, suppose, at time t , we have a set of components. At time $t+1$, we know that these components can be in any of several highly structured arrangements. We also know that the probabilities of each of these arrangements are equally likely. Since the arrangements are equally likely, then we have a uniform distribution. But we also know that the uniform distributions have the highest possible entropy value for their sample space. Therefore, we have here an example that exhibits high degrees of both organization and uncertainty. A biological example of this is meiosis, where the combinations of chromosomes that end up in the zygote are relatively equiprobable. However, each possible arrangement is highly organized.

Now, let's consider example 2, which will exhibit a high degree of organization, but a low degree of uncertainty. This time, we will have the potential of the same collection of organizations manifesting at time $t+1$. However, we shall assume that the probability of one of those arrangements, or organizations, has an extremely high probability of manifesting at $t+1$, while all the others do not. In this case, then, as advertised, we do have a high degree of organization, since all possible arrangements involved are highly structured. However, we have low entropy and thus a low degree of uncertainty because of the nature of the probability distribution. A biological example of this is DNA replication. The probability of copy errors occurring is very unlikely. Thus the probability of a correct replication is highly likely. Such a distribution has very low entropy, and has very low uncertainty. However, any copy that is produced is very well defined and highly organized – even if it is a “mistake”.

The reader should be able to conjure up the other two examples: 1) low organization (highly disorganized) but high uncertainty, and 2) low organization but low uncertainty (highly deterministic).

Nevertheless, it is hoped that the case has been made that Organodynamics needs to make the distinction between *organization* and *uncertainty*, because of the fact that, in organic systems, the two concepts can and do vary independently.

Organization and Uncertainty in other Disciplines

As already indicated elsewhere in this text, the literature pertaining to discussions of *entropy* most often refer to it as measure of “disorder”. But does the use of this term in those contexts mean “disorganization” or does it mean “uncertainty”? Does it mean “unstructured”, “unarranged” or “of weakly-defined relationships”? Or does it mean “unpredictable”?

Even the experts seem to give mixed messages about this issue. For example, the great cyberneticist Heinz von Foerster, in his book *Understanding Understanding*, comments this:

Order is usually considered as a ... loss of uncertainty. [von Foerster 2002]

On the other hand, in the same text, von Foerster also says the following about order:

We may wish to account for apparent relationships between elements of a set. [von Foerster 2002].

But “relationships between elements of a set” is precisely the definition of *organization* in Organodynamics.

It is compelling that Shannon’s formula for *entropy* – which he specifies as a measure of *uncertainty* – is a special case of the Gibbs formula for *entropy*. But the physicists seem united in their agreement that Gibbs’ (and Boltzmann’s) *entropy* is a measure of “order”.

Does this fact imply that, in statistical mechanics, “disorder” means “uncertainty” – and it is simply widely misunderstood to mean “disorganization”?

It does appear that in statistical mechanics, there is conflation, if not confusion, around the meaning of “disorder” and of entropy.

I make no argument with what statistical mechanics means or should mean by “disorder” or by *entropy*. My concern is that the same confusion be brought to Organodynamics. It must be understood unambiguously that by *entropy*, Organodynamics means a measure of *uncertainty* – and *not* a measure of disorganization.

Incremental Example Model Development

<>

Progress Check

In Part II, Organodynamics, we are gradually and incrementally developing a mathematical framework for modeling living and lifelike systems according to the seven organizing principles of OCS. Once completed, this modeling framework will achieve the form of an abstract but highly flexible mathematical structure that we are calling an *Organodynamic web*.

We are using the example of a hobbyist's toolkit for building electronics as an analogy to Organodynamics. Both are systems for "building something", and in both cases the resulting "thing to be built" has a network structure and has been given a general name.

In the case of the electronics circuit toolkit, the general name of the "thing to be built" is "circuit". The circuits built by this kit are highly customizable to specific applications. In the case of Organodynamics, the general name of the "thing to be built" is "Organodynamic web". The models built by this mathematical framework are also highly customizable to specific applications.

Because of the complexity of organic systems, any model that represents them needs to be built in cumulative and increasingly accurate stages. Each stage is generally referred to as an "approximation". Organodynamics as a model builder's toolkit provides for six of these approximations. These approximation levels are well defined and occur at precise places in the methodology set forth by Organodynamics.

In this chapter on *Uncertainty* – the sixth OCS organizing principle, we have provided the first two approximations. The first Organodynamic approximation is called a *segment* and the second an *edge*.

Together the edge and the segment form the "connectors" that span the *nodes* of the network structure that is an Organodynamic web. The *edges* are like the *wires* of the electronic circuit hobbyist's toolkit. The *segments* are pieces of the wires.

We are using graphical analogies for the elements of an Organodynamic web – such as "segments" and "edges", and comparing them to "wires". Nevertheless, when you inspect them closely, they are actually comprised of mathematical entities such as matrices and probability distributions. So, at a low level of abstraction, the elements of this structure are mathematical entities, at a higher level it is sensible to speak of them as graphical entities.

Specifically, a *segment* within an Organodynamic web is a homogeneous finite step Markov chain. Being homogeneous, there is a lot of theory and "intellectual equipment" to make it a tractable unit of "connection". A *segment* is the first level of approximation provided by the Organodynamics model building methodology.

Unfortunately, organic systems do not usually have very many contiguous homogeneous Markov time steps. Normally, there is some time step in an organic Markov chain that changes probability distributions (transition matrices), thus rendering the chain-between-nodes as nonhomogeneous.

Therefore, Organodynamics defines a sequence of contiguous segments to be an *edge*. The second approximation provided by the Organodynamics model building methodology is the edge. An edge can be thought of as a sequence of *segments*.

Autocogeneration: How Organic Systems Reorganize

The Issue Addressed

The fourth OCS organizing principle, *reorganizational*, tells us that the mechanism of change in OCS takes entities that are organized and reorganizes them. However, we want to know a great deal more about that type of change. What is its general architecture and functioning? These questions are answered by the fifth organizing principle: *autocogeneration*.

Organic systems change in special ways. The *organizationality* does not describe enough about how these changes differ from other kinds of dynamical systems. It is the fifth organizing principle, *autocogeneration*, which introduces the additional intellectual equipment that provides the essentials of this difference between ordinary dynamical systems and organic systems.

Autocogeneration is a specific style of reorganization. It involves re-generation of the self by the self. But, even more specifically, it is re-generation through co-generation – the generation of the components and their interrelationships by each other. How such a phenomenon is even possible must be addressed. And how it can be implemented deserves special consideration.

But systems must do more than generate (or co-generate) themselves. In addition, they must repair, control (or promote “regular behavior”) and organize themselves. The inclusive term for all of this selected by Organodynamics is *self-management*. These issues and more are engaged in this chapter.

OCS Organizing Principle Supported

Organizing Principle # 5: Autocogeneration.

Biological Example of the Issue

Within the protein synthesis process, the nucleotide structures of the genes of DNA macromolecules are used as templates by certain enzymes – especially RNA synthetase, itself a protein – to construct a likeness of the gene. This likeness is an RNA molecule – also an organization of nucleotides – that is sent to a location within the cell where another active molecule, a Ribosome, uses it as a template to create the protein molecule – the molecule that is interpreted by molecular biologists to be the “target” of this process.

This ribosome molecule is, in fact, an organization of other RNA and protein molecules. In addition, some of these “target” protein molecules go on to play the role of “controllers” or “regulators” by interacting with the very same genes of the aforementioned DNA macromolecule in a manner that determines whether and when the gene is permitted to be “copied” by the RNA synthetase.

Thus, we have a network situation containing a conditional “looping”, or “feedback”, mechanism that can promote the persistent operation of this

dynamical process in a manner that exhibits “regular” behavior – at least within certain limits wherein it exhibits fluctuation.

A second example – at the level of multicellular organism rather than cell – is the endocrine and the neurological systems within multicellular organisms. These two systems interoperate to promote a vast array of “regular behavior” on the part of their organisms – at least within some boundary conditions around which they still exhibit fluctuations. In any event, the result is that the organism is “self-managed” in a manner that promotes persistence as a result of this regularity.

Specific Challenges

To provide mathematical mechanisms that implement these dynamics and that also utilize the existing mathematical equipment already developed for the Organodynamics framework where possible.

The Organodynamic Approach to Modeling the Issue

Organodynamics will use the asymptotic nature of specific types of stochastic processes from the probability theory literature in order to exhibit highly probable and predictable limiting – and therefore regular – behaviors.

Generally, these stochastic processes leverage conditional dependencies that exist between and among their time steps to severely limit the long run entropy of such processes.

Foundations of Autocogeneration

The essential aspect of autocogeneration is self-generation. But, an entity must pre-exist itself in order to be able create itself. Clearly, this makes self-generation a paradox. The concept of self-generation must offer a way out of this dilemma.

This *apparent* paradox can be circumvented whenever the entity in question is a *system*. A system has components. But the system itself comes into being when some of its components form relationships. Collectively, these relationships constitute a *system organization* of the components. The system is the emergence of this organization.

It is also possible for the components to create each other, and then to form relationships. This occurs to a high degree in biological systems. Actually, the fact that one creates another is already a relationship. This co-creation of component and the subsequent formation of relationships is the creation of the system.

In this sense, it can be said that a system can generate itself because its components generated each as well as their organization. So we can say the system has generated itself by virtue of its components generating each other and then generating relationships between them.

We already gave a biological example of this above by describing the interplay between the protein synthesis and the gene regulation processes. This example occurs at the cellular level, as do many other such cycles. But these kinds of network occur at all levels of organization of biological systems and conduct many functions at each of these levels. For example, at the physiological level in mammals, the endocrine system functions in this fashion. So does the nervous system – as do these systems with each other.

So, we have here a cyclic network of components that co-create, co-transform and co-regulate each other in an interdependent fashion. We shall refer to these interactions generally as “co-management”. The Mahayana Buddhists have a phrase that this cycle reminds me of: “interdependent co-origination”.

As well, these relationships are evidently the essential ideas of the theory put forth by Humberto Maturana [Maturana 1974] and his colleague Francisco Varela - named *Autopoiesis* [Maturana and Varela 1998]. Autopoiesis is the motivating idea behind autocogeneration. But OCS defines these ideas in its own manner, a manner that distinguishes it from these others in a way that integrates them with the other six organizing principles. And we anoint it with our own self-describing name: *autocogeneration*.

General Definition of Autocogeneration

Autocogeneration is system self-management. Autocogeneration includes self-generation as well as self-maintenance.

But self-management includes many forms of behavior. An advanced form of self-management is regularity: the continuity of a specific behavior on a regular basis. It is the existence and maintenance of such regular behavior that gives what identity is reasonable to an individual and to a species. In fact, the semblance of persistent identity depends on regularity. Without the persistence of some specific regularity, we cease to have justification for continuing to identify an individual or species as “the same one”. So, there are conditions under which regularity breeds persistence – continued survival.

On the other hand, an inflexible regularity can lead to annihilation if the behavior ceases to conform to certain demands of the changing environment. So, there are also conditions under which irregularity must occur - a change away from regular behavior – in order to persist.

So, we see that both regularity and irregularity can lead to continued persistence, each in its own time under its own conditions. But also both regularity and irregularity can also lead to annihilation.

A very high form of self-management, and therefore autocogeneration, is the propensity to exhibit regular behavior when regular behavior promotes persistence and to exhibit irregular behavior when irregular behavior promotes persistence. This kind of autocogenerative behavior is called *regularity selection*.

In addition, maintenance also implies some degree of quality control. This includes efficiency, resilience, and other aspects of quality.

Thus, it is sufficient to define autocogeneration as self-management. It is interesting to note that these ideas are at work in the field of systems engineering and large-scale project management. The difference between those and these, of course, is that in engineering, an external agent (the engineer or project manager) is managing the system. In OCS, the system is managing itself. This difference is far from trivial, and lies at the heart of why organic systems are complex.

Autocogeneration as Regularity and Irregularity

Within autocogeneration, self-management is a general behavior that includes self-creation, self-repair, self-regularity and self-irregularity. Perhaps the highest form of autocogeneration is *regularity selection*, by which we mean the propensity to exhibit regular behavior when regular behavior promotes persistence and to exhibit irregular behavior when irregular behavior promotes persistence.

A principle indicator of *regularity* is *conditional dependency*. If two phenomena are interdependent then we can have an expectation that they occur together, or that one follows the other, more often than if they are not conditionally dependent. Thus, *conditionality*, or *interdependency*, or *conditional dependency* is an indicator of regularity.

This fact tells us how we can model regularity using mathematics. Probability theory supports the notions of statistical independence and statistical dependence, as well as conditional probability. We shall find these are important tools that Organodynamics will exploit for representing and modeling autocogeneration.

Information theory strongly leverages these ideas of conditional probability and statistical dependence to discover limits and boundaries on irregular behavior. In particular, information theory is concerned with the uncertainty associated with stochastic processes, and the conditions under which that uncertainty is reduced or bounded. Explains Richard Kleeman of NYU, "The central idea of information theory is to measure the uncertainty associated with random variables [probability distributions]." [Kleeman 2009].

Perhaps the central inequality of information theory states that the uncertainty of one space will decrease if it is known which event occurred in some other space whenever the events of the two spaces are statistically dependent [Shannon 1963; Khinchin 1957].

If the events in question are the time steps of a stochastic process, and all of them are mutually dependent, then the additive degree of uncertainty (entropy) of the process can be severely diminished as compared to the same process if the time steps are statistically independent. We see that in biochemical systems a great percentage of the processes have strong dependency relationships (often causal), which give to them highly interdependent conditions. Thus, due

to the reduced uncertainty in many of these systems, it is reasonable to expect a preponderance of regular behavior.

Notice that I have avoided the term “regulation” here, and have kept with the terms *regularity* and *irregularity* instead - with good reason. “Regulation” carries with it the connotation, if not denotation, of *imposed will* – and generally the imposed will of a third party, rather than autonomous imposed will. And imposed will implies *intent*, *purposefulness* and other teleological phenomena. But, in Organodynamics, while the idea of *autocogeneration* permits intentionality, but does not require it. Generally, we shall not assume that any notion of purposefulness is at work in these processes.

Indeed, teleological phenomena are held by Organodynamics as a very sophisticated form of autocogeneration that occurs as an emergent systemic property at high levels of organization of deeply nested organic systems – such as the human brain. However, at lower levels of organization, such as molecular evolution, proto-life and likely even single-celled organisms, there is no need to assume the existence of intention or purpose. Thus, while Organodynamics includes purposefulness at higher levels of organization, it does not require it, and generally eschews it at lower levels of organization.

Examples of Autocogeneration in Science, Engineering and Mathematics

Autocogeneration is defined to provide an inclusive concept that can embrace many similar kinds of behavior that are encountered in various disciplines. All of these concepts capture some notion of what is typically referred to as “autonomous process regulation”.

However, “process regulation” typically implies some notion of the imposition of imposed will, often by an external entity. But in autocogeneration, “regulation” generally does not result from the imposition of any rules; but is rather the consequences of the operation of internal system dynamics. More specifically, *regularity* is treated by Organodynamics as an *emergent* property of the specific probabilistic conditions selected or assumed.

For this reason, Organodynamics eschews the term “regulation” with its connotation of rules and external imposition, and uses the term *regularity* instead.

However, “autocogeneration” and “regularity” is sufficiently general as to include the case of “the external imposition of rules” as well. Autocogeneration does not exclude these cases of “the intention of imposed will”, as long as the imposition originates from an internal system component. But it does not require it.

In particular, organic systems – especially primitive ones – do not appear to exhibit intention, purpose, will or any other form of teleological behavior. On the other hand, higher forms of biological organization – especially animals – do exhibit these traits. Thus, Organodynamics seeks to account for both.

Some specific forms of autocogeneration encountered in various disciplines appear in the following list.

- Homeostasis
- Cybernetic Feedback and Control
- Self-Regulating Behavior
- Self-Irregulating Behavior
- Regularity Selection
- Natural Selection and Adaptation
- Catalysis and Autocatalysis

Homeostasis

Homeostasis literally means “same state”. The idea, first expressed in the west by French physiologist Claude Bernard in the 1860s, is that in living systems, there exist certain internal states that need to be maintained in order for an organism to exist and survive. However, there are often forces at work that have a tendency to dynamically drive the organism away from these states. However, Bernard noticed that there are mechanisms within living systems that bring these systems back to these “same states” whenever the system is driven away from that state. We now use the term *homeostasis* to identify these mechanisms.

Bernard, and others since, identified this type of behavior as an essential characteristic of living systems, since living systems exhibit many forces that drive their dynamical processes away from any particular state that they may be in at any particular time. (Stochastic processes exhibiting certain probability distributions model this propensity in Organodynamics.)

Clearly, a tendency to move away from these essential states represents irregularity, while a homeostatic mechanism would drive the system back to regular behavior. Thus, living systems must also exhibit some kind of countervailing tendency that induces regularity. Homeostasis is the mechanism identified by Claude Bernard to account for this countervailing tendency toward regularity in living systems.

However, Bernard named this tendency *milieu interieur*. It was left to William Bradford Cannon, an American physiologist at Harvard, to supply the name *homeostasis*, and to bring additional understanding and clarity to the mechanism of homeostatic behavior [ECB 2004].

Cannon observed that certain conditions (states) of living systems required constancy (“regularity”). And mechanisms must exist within these systems to maintain that constancy. That is, for tendencies that mediate away from that constancy, there must be mechanisms that resist the change away from that constancy. Cannon also pointed out that the mechanisms involved in this “resistance to change away from constancy” consisted of a set of cooperating (co-operating) components, and was therefore the result of organized self-government [Cannon 1932].

Cannon also concluded that since this “constancy” is the result of organized self-government that, therefore, it could not be the result of chance [Cannon

1932]. However, with this last statement, Organodynamics takes issue. It is assumed by most that “chance” can only lead to chaos, and never drive toward determinism. However, it is a thesis of OCS that chance, under the right conditions, can lead to high degrees of certainty, and concomitantly to low entropy – and that it very often does so in organic systems. (See the subsection below entitled “Ergodics and Absorption” for more on this issue.)

Although some contemporary renditions of homeostatic theory have attempted to account for positive feedback in various ways, homeostasis, as envisioned by Bernard and Cannon, at its foundations, essentially concerns the processing of negative feedback and emphasizes “constancy” and regularity.

Clearly, while *homeostasis* was named and characterized within the context of biological systems, the concepts and principles involved are clearly extendable to general systems.

Cybernetic Feedback and Control

Cybernetics is a theory of autonomous system control in which regular behavior is achieved via a network of information that monitors the dynamical behavior of a system with respect to a certain properties of that system; and that, based upon that information, alters the behavior of that system, when necessary, in order to ensure that it persists in exhibiting regularity with respect to that behavior [Wiener 1961].

The information is called “feedback”. A Cybernetic system is a network of components, some of which process this feedback and alter the behavior of the remaining components so as to ensure the regularity of the system. The component(s) of the system that report that information is called the *control center* or *monitor*. Whenever the information reveals that the system is behaving irregularly – or contrary to the behavior to be repeated, then a specific component of the system, the *controller*, takes action to insure that the system’s behavior becomes regular.

Such feedback is called “negative feedback”, because the controller’s response is to take opposite action to what is revealed by the feedback pertaining to the departure from the current state from a “controlling quantity”, also called a *set point*. According to Wiener, “The information sent back to the control center tends to oppose the departure of the controlled from the controlling quantity, but it may depend in widely different ways on this departure.” [Wiener 1988]. The initial concepts of Cybernetics as introduced by Wiener and others emphasized negative feedback, and did not articulate an accommodation for its opposite, called positive feedback.

The developers of Cybernetics were aware that this type of behavior is exhibited by biological systems, where it is named *homeostasis*. In a sense, Cybernetics can be seen as an engineering discipline that enables man-made systems to be designed and constructed in a manner that implements many of the concepts evident in homeostasis.

In general, we can say that Cybernetics is characterized by negative feedback loops (networks), monitors and controllers for the purpose designing self-regulating systems. Although some contemporary renditions of Cybernetic theory have attempted to account for positive feedback in various ways, Cybernetics is at its foundations, essentially about the processing of negative feedback, and emphasizes “constancy” and regularity.

Characteristic of Cybernetics is the assignment to a system component (or subsystem of system components) the roles of *monitor* and *controller*. Thus, a division of responsibility occurs in Cybernetics in this sense.

Historically, Cybernetics, which emerged in the 1940s, was very instrumental in the legitimization of systems theory and systems science, and is credited with ushering in the computer revolution – if for no other reason than its principle advocates were mathematicians and scientists who participated in the invention of computers. Cybernetics was named by mathematician Norbert Wiener and was originated by a number of prominent mathematicians of the time, including John von Neumann, and Alan Turing. Also, the Cyberneticists were a global community of scientists, humanitarians and artists who sought to create a science of mind. As, well, Cybernetics had immediate application to a number of technologies developed by the western nations during World War II, and is generally credited with giving the west a decisive advantage in that global conflict. Accordingly, it was held in academic, industrial and governmental circles as an influential and successful program. [Capra 1996; Gleick 2010].

In addition, Cybernetics is usually associated with the engineering of control systems. In such a case, it is difficult to escape the notion that, since engineering is involved, that *intentionality*, *purpose* and other teleological aspects are also involved in Cybernetics. Moreover, this intentionality is not present in the system itself, but rather in the designers of those systems. And those designers (the engineers) are not components of the system. Rather, they are external agents who provide many of the cogenerative system properties, including replication and construction. Certainly, other autocogenerative properties, such as self-regulation, are engineered into these Cybernetic systems.

Consequently, even though Cybernetic systems do exhibit some autocogenerative properties, they are generally not *organic systems* in the sense used by Organodynamics, because they are not self-creating.

Self-Regulating Behavior

Any dynamical system that, for some reason, tends to revisit the same state repeatedly is said to be stable. This is an example of regular behavior. If this stability is the result of some aspect of the dynamical system itself, then the system is said to be *self-stabilizing*, and we have an example of self-regulating behavior.

Stabilizing behavior is a special case of regular behavior in which the one state is revisited at the expense of the others. However, we can relax this requirement and consider more general cases in which a subset of all of the

states is repeatedly revisited. Such dynamical systems are said to exhibit cyclical, or periodic behavior. Trigonometric series and Fourier analysis provide general mathematical models for these types of processes. The decimal expansions of the rational numbers also present this type of regularity.

When a dynamical system exhibits this property of regularity as a result of its own properties, then the system is said to be *self-regulating*. Such properties may be an integrated systemic property, or it may be a component property, as in Cybernetics and homeostasis.

The Cybernetics model provide “dedicated” system components whose role is to implement this regularity in the form of monitors and controllers. However, There are systems that, by their very nature (systemic properties) induce this regularity upon themselves without the necessity of dedicated subcomponents. We shall see an example of this below in the predator-prey system studied in ecology.

Another thing that is often said about self-regulating dynamical systems is that they are *self-correcting*. Of course, this articulation assumes that one of the states is “correct”, that the other states are “incorrect”, and that the process returns often to the “correct” state.

However, in general, we would like to be able to account for “self-correcting”, more accurately called “self-regulating”, behavior in a dynamical system even if it does not have the specialized components that are defined by Cybernetics.

A good example of such a system from Ecology is the predator-prey model as presented by Lotka and Volterra [Hazen 2002]. In this model, there are two species, one being the predator of the other. According to this model, the two species interact in such a way that the size of both populations goes through cycles. The two population size cycles exhibit synchronization, with one population being at a high point when the other is at a low point, and vice versa.

For example, the predators tend to over-graze the prey. But when that occurs the predator population starts dwindling for lack of enough food. When the predator population reaches a low point, the prey population begins to increase, and it heads toward a high point. Subsequently, the prey has enough to eat and begins to flourish and heads toward its own high point, as the prey diminishes due to its depletion by the prey. This brings the cycle back to where we started.

This cycle can continue for a lengthy duration until one of the species goes beyond its lower limit and meets its demise. (More often, it is the prey population that meets its demise first – contrary to Wall Street metaphors that represent “the strong” as out-surviving the meek.)

It is tempting to say that both the predator and the prey species regulate each other. If we could say that, then we would clearly have identified the regulating components, and then we could say that this system is indeed Cybernetic in nature.

But hold on a minute. The implication of both of the species being regulators is that all of the components of this system are regulators. Moreover, one must admit that the only reason that both of this system's components are regulators is because of the inter-relationships between them. Therefore, all of the systems components, plus their relationships is actually the "regulator". But, this is, in fact, the definition of the system – all of its components and their relationships (organization).

So, in the case of the Lotka-Volterra predator-prey model, the regulation occurs as a result of the nature of (the systemic properties) the system itself – rather than because of some of its components whose role is regulation.

It would be stretching credulity to classify this model as "Cybernetic", since there are no components of the system whose role it is to "regulate" any aspect of the system. Rather, it is the various systemic properties of the system that interrelate so as to result in regular behavior – at least for a possibly large number of dynamical process steps.

I claim that this self-regulating (self-correcting) behavior goes beyond the scope of what can be reasonably called Cybernetics.

Nevertheless, this complex, self-referencing, behavior falls within the purview of autocogeneration.

Self-Irregular Behavior

There are common phenomena in dynamical systems that are often referred to as the "snowball effect", because they are characteristic of a snowball that is rolling down a hill.

When a snowball rolls down a hill, it attracts the snow beneath its path to stick to it. This snow becomes a continuous layer that is added to the volume of the snowball itself. Consequently, the bigger the snowball becomes, the bigger it becomes. The snowball effect is a *self-reinforcing* phenomenon. The direction of its change in behavior as the process proceeds increases with the process.

As such, the *snowball effect* has become an informal name for self-reinforcing behaviors in dynamical systems.

Notice that the snowball effect moves the snowball away from its initial state with respect to mass. In fact, at each moment, the snowball effect moves the snowball away from its mass at any previous moment. This behavior is the opposite of *regularity*. Thus, in Organodynamics we shall call it *irregularity*. When this irregularity is a result of the nature of the system itself, then we call it *self-irregularity*.

As such, it shares *positive feedback* in homeostasis is a special case of *self-irregularity*. They are both examples of self-reinforcing phenomena. However, the snowball effect is both more primitive and more fundamental than positive feedback in homeostasis. And, it requires a lot less "equipment". (E.g. a snowball has no "parts" that move with respect to each other.) Whereas

homeostasis supplies a number system components whose role is to cooperate in order to implement positive feedback

Homeostasis is a special case of autocogeneration. It involves a number of specialized components whose role is to implement either the regularity (negative feedback) or the irregularity (positive feedback) mechanisms.

However, with the snowball effect, as with the self-regulating phenomena discussed in the previous chapter, the mechanism that induces either regularity or irregularity is bound up in the systemic properties of the system itself. The irregularity exhibited by the snowball rolling down the hill is a consequence of the interaction of the current surface atoms of the snowball with the atoms of the snow of the environment with which the snowball comes into contact dynamically.

In this sense, while homeostasis and Cybernetics are quite sophisticated, they are not sufficiently general to capture all self-reinforcing and all self-correcting phenomena. In fact, it does not describe either snowball effects or the general self-regulating effects that we have described.

However, autocogeneration includes all of these phenomena as special cases: homeostasis, Cybernetics, cyclic phenomena and snowball effects.

Regularity Selection

For any dynamical system, there are conditions for which *regularity* of behavior serves to promote persistence, and there are other conditions under which *irregularity* promotes persistence. It would further promote persistence if a dynamical system exhibited a property that selected either regular or irregular behavior depending on which promoted persistence.

We shall call such a behavior *regularity selection*. Many living systems seem to exhibit this behavior. For example, the immune system of multicellular biological systems can do this. The original manifestations of Cybernetics and of homeostasis did not identify such a capability.

However, we shall stipulate that *regularity selection* is an autocogenerative property.

Mechanisms of Autocogeneration in Biological Systems

Mechanisms of Regularity and Irregularity

A number of specific mechanisms by which living systems implement regularity and irregularity can be identified by observation. This section will discuss a number of these. All of these, then, are examples of autocogeneration – and therefore answer the questions “Precisely what are some of the reorganizational behaviors that are implemented by living systems?”

Division

One of the mechanisms by which living systems reorganize is that they partition themselves into two or more portions. In particular, living cells divide into two portions, with the result of doubling their numbers.

But even more primitive forms of protolife divide. Also, the molecular components of living cells also do this. For example, metabolic cycles, such as the citric acid cycle, involve steps in their cycle in which small molecules divide into two smaller molecules.

Often, division is symmetrical. That is, the partitioning is such that the two parts are identical. And, often the dividends are scaled-down version of the original undivided cell.

In the latter case, a replica of the original undivided cell can be reconstructed simply by replicating each of the components of either one of the dividends. This, without question, is an example of regularity. Thus, regularity can be implemented via division plus component replication.

On the other hand, the division of the original molecule often produces a nonsymmetrical partitioning. That is, neither of the dividends is “like” the original – and both are different. This, without question, is an example of irregularity. The reverse citric acid cycle has step where the six-carbon molecule of the cycle divides into two molecules – one having four carbon atoms and the other having two [ECB 2004].

Growth

Growth can lead to either regularity or irregularity, depending in its implementation. Some of these implementations are discussed below.

Replication

Growth can take the form of replication. The reader will recall that *replicate* is one of the operations defined on systems in the chapter of this text in *Reorganization*.

Replication is the creation of a current step of some system process via the construction of an identical system as some existing step of an existing system process.

Clearly, then, replication promotes regularity. Specifically, when replication is applied to a dividend of a division, then replication promotes regular growth.

Non-replication

Growth can also take forms other than replication. This occurs whenever a new step of a system process is constructed in such a way that is a non-replica of a specific state of a system process. For example, if at least one component of the constructed system state is different from the specific state, then non-

replication has occurred. Non-replication includes a number of determined system transformations, as well as in reendow transformations.

Clearly, non-replication can support varying degrees of regularity or irregularity, depending upon the amount of variance involved. Generally, however, non-replication is an engine of irregularity.

Unpredictable Growth

Some transformations produce unpredictable results. The degree of uncertainty may range from none to much. In the chapter on Uncertainty, we presented Shannon entropy as a measure of this degree of uncertainty.

An example of this is almost any metabolic process. Most of these processes (metabolic pathways) contain steps whose chemical compounds can be acted upon so as to produce more than one possible result, depending upon which compounds they encounter next. [Watson 1970]

These logical possibilities have a random nature and are described by various probability distributions. Depending upon which of these occurs, the process will become either one metabolic pathway or some other. The number of possibilities is normally small. However, the probabilities involved are sufficiently diverse to enable us to say that the degrees of uncertainty represented range from small to large [Watson 1970].

Thus, there are certainly cases when the outcomes of these metabolic processes are highly unpredictable. As, well, some of them are relatively predictable; and some exhibit intermediate degrees of uncertainty.

On the other hand, there are unpredictable growth processes whose degree of uncertainty is extremely low, although greater than zero. The quintessential example of this is DNA replication. The uncertainty occurs in DNA replication primarily through “copy errors”, but also because of mutations. In DNA replication of some species, the rate of error is extremely low – on the order of one in a billion.

Nevertheless, it is highly significant that these copy errors do occur. If they did not occur, then evolution would be impossible. This is because these copy errors are the major source of species adaptation. Thus, even though it is extremely important that DNA replication be almost perfect, it is equally important that it not be absolutely perfect!

Thus, DNA replication is a wonderful example of how the dance between regularity and irregularity is essential for living systems.

Like non-replication, these unpredictable growth transformations are generally an engine for irregularity.

[The careful reader will notice how I skillfully placed the entire subject of genetics under the *autocogeneration* organizing principle! Unfortunately, upon

reflection, this may seem heretical to many contemporary biologists, including many molecular biologists, geneticists and evolutionary biologists.

This means that I have categorized genetics under the same rubric as homeostasis! I have categorized the role of genetics as just one more mechanism - for managing regularity and irregularity.

I know that this disregards the penchant to treat the subject of genetics as though it were the very definition of livingness that is attributed to it by many, if not most, contemporary biologists.

My only defense is that I am developing a general theory of the lifelike – and have not confined my study to that of biological systems on earth. It very likely is the case that genetics is an indispensable implementation of regularity in earth biology. However, my position is that such implementation is simply one of many possible implementations, or styles, of regularity and irregularity strategies – and that genetics is not a mandatory architecture for lifelike systems.

However, what is essential is some form of autocogeneration.]

Multifarious Proliferation

This mechanism is otherwise known as the “shotgun strategy”. It consists of the generation of a large number of instances of a very large number of types, or species, of organic entity.

For example, certain species of insect lay a very large number of eggs, any number of which are the result of either replication, non-replication or unpredictable replication.

Of course, most of these copies will not survive; mostly do to resource constraints but also due to death for other reasons. Certainly, most of the non-replicas and unpredictable replicas will not survive. However, whenever the environmental conditions have changed appreciably, some small number of the non-replicas and unpredictable replicas will adapt better than the replicas.

So, this haphazard system actually works – by brute force alone. Most humans, especially engineers, would not be willing to describe such a “design” as very “intelligent”, or a least not very elegant. In fact, such a brute-force approach is generally anathema to engineers.

Engineers require highly intentioned, deterministic and well-considered organization based upon reasoned principles of engineering. Any brute-force, random and haphazard approaches such as this would be eschewed by any respectable engineer as “inelegant”.

Yet, here it is. Biology has leveraged such an approach to system synthesis to produce the kind of systems that exhibit systemic properties only dreamed of by the most astute systems engineers.

Natural Selection

Natural selection is a specific style of *regularity selection*.

Recall how we defined *regularity selection*. We said above that it would further promote persistence if a dynamical system exhibited a property that selected either regular or irregular behavior depending upon whether regularity or irregularity promoted persistence at that particular time. And we named such a property *regularity selection*.

But this is precisely what natural selection accomplishes for life on earth. Of all autocogenerative processes that are operational in a particular ecosystem, the ones that are currently promoting persistence – whether they are regular or irregular - are the ones that are persisting – that is, that are “selected” by natural selection.

Information theorist Vlatko Vedral states that “Natural selection is the process by which you are correlated to your environment...”. But, we also know [Feller 1968, p. 236] that two stochastic processes are *correlated* only if they are stochastically dependent. In this case, the two processes are “you” (an organic process) and the environment in which “you” as a system process exist. (In Organodynamics, the “environment” of a system is the organizations of the composite system of which it is a component system.)

This means that *natural selection* is a process by which all of the component systems of a composite system, and their composite system, become more *stochastically dependent*. This implies that all of the component systems of a composite system become mutually dependent with each other and with their composite system (their environment.)

The notion of natural selection is brilliant because it is almost tautological while, at the same time, non-intuitive – at least for the denizens of Darwin’s day. In any event, it is applicable well beyond biology. For example, it is operational in any number of natural phenomena. For example, we speak of “molecular evolution”, meaning molecular natural selection, in protolife [de Duve 1991; Hazen 2005].

The simplistic interpretation - often heard in popular culture - that natural selection is the strong defeating the weak is debunked by the Lotka Volterra predator-prey model described above. As we already pointed out, in that model, it is just a likely, perhaps more so, that the prey population will outlive the predator population, because the predators will likely all die of starvation before all of the prey population is completely devoured. That is, there is generally a threshold of prey population, below which the predator population cannot survive. This holds when the prey population is the essential food source for the predator.

Adaptation

Adaptation is a change in the state of an organic process from one step to the next that discontinuously improves the prospects of the processes continued persistence.

We can define *adaptation* in Organodynamics in terms of the OCS uncertainty organizing principle and Markov chains. Recall that in Organodynamics, our conditional model of an organic process is a Markov chain. A Markov chain whose transition matrix has changed (the chain becomes non-homogeneous) in such a way that the number of steps to the termination of the chain is greater than the number of steps to termination if the transition matrix had stayed the same; then the change in transition matrix is said to be *adaptation*.

In the previous subsection, we quoted Vlatko Vedral as saying that “Natural selection is the process by which you are correlated to your environment...”. But, this “correlating to your environment” of which he speaks is another way of saying “adaptation”. Thus, it is more specifically *adaptation* that drives the increased stochastic dependence that we attributed to natural selection above.

Thus, adaptation works by increasing stochastic dependence, and concomitantly decreasing relative entropy in time [Khinchin 1957; Kleeman 2009].

Catalysis and Autocatalysis

Catalysis is an operation on two organic processes, the substrate process and the catalysts process, that produces two output processes - the substrate process and the catalysts process. The catalyst process remains the same after the cooperation. But the probability distribution of the substrate process is changed so that one of the states has a significantly higher probability than it does in the substrate process.

Another way to say this is that catalysis increases the correlation between a substrate system and its product system. That is, the two become more highly *conditional stochastically*, and the relative entropy is reduced.

Such an operation can promote either regularity or irregularity, depending upon whether the state of the substrate process whose probability is increased promotes regularity or not.

Autocatalysis occurs when there is a nested (composite) system whose component systems are mutual catalysts for each other.

The Mechanisms that Implement Regularity In Organodynamics

Fortunately, we can use a number of mechanisms from probability theory and two of its branches – stochastic processes and information theory – to implement autocogeneration in the Organodynamics framework.

Of course, this is very economical and synergistic since we are already using many of these same ideas to implement another of the seven OCS organization principles – *uncertainty*.

Stochastic Dependence

We are beginning to see a trend here wherein regularity and stochastic dependence are related. Stochastic dependence has figured into our descriptions of a number of mechanisms of life on earth that drive regularity, including natural selection, catalysis and adaptation.

This is not surprising due to the mathematical fact [Feller 1968] that variables that are highly correlated must be stochastically dependent. Of course, regularity is a case of a particular type of correlation, called *autocorrelation*, wherein the outcome of a process at one step is correlated to the outcomes later steps of the same process.

Thus, one important mechanism of probability theory and the theory of stochastic processes that models regularity is that of *stochastic dependence*. Thus, it is not surprising that stochastic dependence has been a constant visitor in or probabilistic description of these several mechanisms of regularity.

Conditional Probabilities

But, in the theory of stochastic processes, stochastic dependence is implemented via the mechanism of conditional probability.

Specifically, events A and B are *stochastically dependent* when the outcome of event B alters the probability distribution of event A. That is to say, the probability of events A *depends* upon the outcome of event B – as compared to the probability distribution of A if the outcome of event B is not known. (It is easily shown that if A depends on B, then B also depends on A. That is, stochastic dependence is symmetrical.)

But, the “probability of event A, given knowledge of the outcome of event B” is known as *conditional probability*. In fact, the language used is “the probability of A given B”, and it is symbolized as “Pr[A|B]”.

Thus, since *regularity* of stochastic processes is modeled through *stochastic dependence*, and stochastic dependence is defined in probability theory in terms of conditional probability and conditional probability distributions, then it is natural that Organodynamics will leverage the theory of conditional stochastic processes as its foundation.

Of course, the simplest case of conditional stochastic processes is that of Markov chains. Consequently, Organodynamics will lead with the application of Markov chains to model organic systems, and move on to more complex types of conditional stochastic processes when necessary.

Ergodics and Absorption

As we shall see in the next chapter of this text on the use of Markov chains to model autocogeneration in Organodynamics, the execution of a Markov chain must inevitably lead the process to enter a finite subset of the state space and to never leave that subset.

This inevitable aspect of Markov chains is named *ergodics*; and such an “inevitable” subset is called an *ergodic set*. It is possible that an ergodic set has only one member. In such a case, the ergodic set is called an *absorbing set*.

Ergodics is a particular example of conditional dependency leading to regularity. Markov chains shows how a particular type of consistent conditional chance behavior can lead inevitably to regularity – specifically, cyclic regularity. This cyclic regularity is the *ergodic sets* that all Markov homogeneous Markov chains possess.

Absorption is a special case of ergodics in which the number of states in the cycle is one. Thus, *absorption* leads to, not chaos, but certainty. In fact, in this case, uncertainty resolves to determinism.

The principle point being made here is that chance, or probability, need not lead to chaotic conditions, as generally supposed. Rather, under certain conditions, probability leads inevitably to well-defined patterns and order – and even to certainty.

Debunking the treatment of chance phenomena as inevitably leading to chaos is one of the principle tasks set out by this theory. In fact, the sixth organizing principle – uncertainty – is essential in resulting in the kind of persistent organization that is life, and the lifelike.

Strategy for Modeling Autocogeneration using Markov Chains

Let's now consider a strategic approach to using Markov chains for modeling autocogeneration as *regularity* and *irregularity* of dynamical processes. In the next chapter, we shall work out the details of this strategy, and develop it into a full-fledged stochastic dynamical theory of organic systems. Because of the complexity of the theory presented in the next chapter, then, the present chapter has served as an orientation, introduction and conversation of strategy.

The remainder of the present chapter will organize and summarize what we have so far said about this approach in the present chapter.

Modeling Regularity using Markov Chains

To model regularity in Markov Chains, we want to represent regularity as a propensity of a dynamical process to frequently return to the same state, or states, over time.

There are several special cases of regularity that we also want to single out, discern and name. These include having the return to certain particular states

to be more favorable than to other states that may also entertain return visits by that process. The most special case would be to single out a particular state to which this return would be most favorable. This is the special case of regularity called *homeostasis*.

To achieve the idea of “frequently returning to the same state over time”, we shall leverage the notion of *ergodics*. That is, ergodics will provide us with a general strategy for achieving regularity. Within general ergodic behavior, we will be able to identify the special cases that we want as well.

An *ergodic set* is a set of states of a Markov Chain that, once entered, are never left – at least while the “overall conditions” stay the same [Kemeny and Snell 1969; Bharucha-Reid 1960]. That is, while the chain of steps is *homogeneous*.

If you find yourself in one of the ergodic states, then the state realized in the next time step must also be in that ergodic set, etc. Unless the overall conditions change, (non-homogeneity) during successive steps of the stochastic process, you will continue to cycle among the elements of the ergodic set, and never escape that set once entered.

Under the right the conditions, an example of an ergodic set in biochemistry is a metabolic cycle – such as the citric acid cycle. In such a process, a “state” is an organization of the atoms wherein one of the chemical compounds is represented by a cluster within that organization. As long as all of the right “food” molecules are present, then a metabolic cycle will continue indefinitely.

Note: The textbook presentations of metabolic cycles - in which the molecules involved can only cycle among the states defined in the cycle – form a set of ergodic states. This regularity can occur whenever the required food molecules are always available. In other words, time-homogeneity of the appropriate probabilities is maintained.

In actuality, though, the chemical conditions can change, in which case alternative transitions can occur that take the process out of the cycle. These eventualities are, in fact, examples of non-homogeneity, wherein the Markov transition matrix actually changes at the step in which the process exits from the cycle.

But in the “normal” case presented in textbooks on cell biology or biochemistry, the textbook generally provides a list of the compounds in the cycle. Classical examples of these cycles are the glycolytic cycle (glycolysis) and the citric acid cycle (i.e. the Krebs cycle) [ECB 2004].

Let’s pause for a second to comment on what all of this means to an organic process in Organodynamics. You will recall that in Organodynamics “system state” is a very specific entity: a *system organization* – which is defined to be a set of clusters of related duples of the system’s components. Therefore, for any given step in an organic process, the state space will be a set of such organizations.

What we are saying in this chapter is that for certain finite numbers of contiguous time steps, there will be a subset of these system organizations that form an ergodic set. A set of system organizations that contain clusters that represent all the molecules in the citric acid cycle would constitute such an ergodic set, for example.

In fact, there may be more than one such subset. For example, there may be two or three, etc. ergodic subsets of system organizations for a step of an organic process. But, there will be at least one ergodic subset of organizations for each step of an organic process.

What we are saying in this chapter is that, if the number of steps goes on long enough, then one of those ergodic sets will be entered – and never left. As we have said, this is regularity. In any event, it always has the propensity for regularity.

Note that in the above example, we used the phrase “Unless the overall conditions change” and “If the conditions are right”. In the terminology of stochastic processes, these phrases imply a time-homogeneous stochastic process. This means that the same set of conditional probability distributions (the same transition matrix) works for all of the steps of the process. Therefore, implicit in the notion of regularity is the idea of homogeneity.

Thus, to model regularity, we shall require homogeneity as well as ergodics. Below, we shall define the notion of “finite subchain” so that we can consider homogeneity to be limited to just a few contiguous steps at a time – which is more realistic.

We are getting pretty close to modeling regularity by putting these two ideas together: time-homogeneity and ergodics. And, Markov chain theory says that if you have time-homogeneity and enough time, you must have ergodics, and therefore, regularity.

But there is one thing still missing. So far, the notion of ergodics alone leaves us with too much “democracy” among the members of the ergodic set of system states to be able to cover all of the cases of regularity that we may want to model. This is because the examples we have seen so far have been strictly cyclic. We would like to allow ergodics to include the case where some of the members of the ergodic subset can get visited more often than others. This we shall do next.

Modeling Homeostasis with Markov Chains

What we know so far is that 1) all homogeneous Markov chains have at least one ergodic set, 2) after some time step in a Markov chain, an ergodic set gets entered and never left, and 3) all of the states in this ergodic set get re-visited periodically for the remainder of the steps of the chain.

However, in a specific case of regularity called *homeostasis*, we have the notion of a “set point” – which is a particular state of the ergodic set that gets emphasized over the others. Thus, in order to use ergodicity to model

homeostasis, we need to add some stipulation to what we already have in order to favor one of these ergodic states over the other ergodic states in the ergodic set.

We can achieve this special case of homeostasis by requiring that one of the states of the ergodic set have – in the long run - a higher probability than the other states of that ergodic set.

But, what does it mean to say that one of the states has a higher probability than any of the other states?

The problem here is that the transition matrix for such a Markov Chain has multiple conditional probability distributions for us to work with. And each of them may have a different state with the largest probability. How then do we select one distribution from among this set of distributions from which to make a determination as to which state has the highest probability?

Here is an example that illustrates this problem. In the “Jack’s dinner” transition matrix, which we repeat below, the state with the highest probability for tomorrow’s dinner in the first distribution (today’s entrée was “Chicken”) is the Meat state with a probability of .5. However, the second distribution (today’s chosen entrée was “Fish”) tells us that the Chicken state has the highest probability for tomorrow’s dinner with a .6. And the third conditional distribution in this matrix tells us that it is Fish with a probability of .5.

Conditional Probability Distribution Set Example

What Jack selected for today’s dinner	Probability of Jack’s selecting <u>Chicken</u> for tomorrow	Probability of Jack’s selecting <u>Fish</u> for tomorrow	Probability of Jack’s selecting <u>Meat</u> for tomorrow
<u>Chicken</u>	.1	.4	.5
<u>Fish</u>	.6	.1	.3
<u>Meat</u>	.4	.5	.1

The problem is, of course, which one of these distributions do we use to determine which of the three entrees will get served most often in the long run?

In fact, it is reasonable to ask whether this transition matrix contains enough information in and of itself in order to answer that question. Perhaps, surprisingly, it turns out that there is. In the case that our process is a homogeneous Markov chain, it can be shown that, after a large number of steps, this matrix begins to converge to another matrix that lets us answer this question.

And, even stronger, this approach allows us to get the number of distributions down to one! This is because, as the number of steps in our chain gets larger,

the compounded long-run transition matrix converges to a matrix whose rows are all the same as each other. And this one limiting row specifies the single probability distribution that we desire.

In Markov chain theory, a homogeneous chain whose state space consists only of a single ergodic set is called an *irreducible Markov* chain. It is also called an *ergodic Markov* chain. (Thus, we have two meanings for “ergodic”: an ergodic set and an ergodic chain.)

The approach that has been presented so far in this text is based upon a theorem that tells us that, given a homogeneous, irreducible Markov chain, then “in the long run” there is a limiting distribution that applies to all of its conditions [Kemeny and Snell 1976].

(Note that in classical Markov theory, a “regular” Markov chain is an irreducible, aperiodic chain. Certainly, this use of the word “regular” could be in conflict with our use here in Organodynamics. Fortunately, at least, an irreducible, aperiodic, homogeneous Markov chain would be a very special case of the Organodynamics notion of regularity – which only requires homogeneity. Moreover, we shall have no need to specify aperiodicity, since we never need to get that specific. Even if we do, we shall specify “aperiodic” so as to avoid using the word “regular” in the classical sense.)

In Markov chain theory, one can obtain a “long run transition matrix” by multiplying together the individual transition matrix of each of the steps involved. If the Markov chain is homogeneous, then all of the transition matrices are the same; and multiplying them is the same as raising them to a power.

It turns out that, if the transition matrix is raised to the n -th power, and n is allowed to infinity, it approaches asymptotically a limiting square probability matrix with the same number of rows and columns as the chain’s transition matrix. Moreover, that limiting matrix has the property that its rows are all equal to each other.

But this implies that, for a homogeneous irreducible Markov Chain (all of whose states are in a single ergodic set), in the long run it does not matter which condition you start in! (It does not matter what the initial conditions are.)

This means that no matter which condition you start in, the long-run conditional probabilities are all the same as each other – and get closer to each other as the distance of “long-run” increases.

In fact, these conditional distributions approach a limiting distribution as the time increases. This limiting distribution is called the *stationary distribution* of the irreducible homogeneous Markov Chain.

[Kemeny and Snell 1976] give a simple algorithm for calculating the transition matrix of such a chain on page 72 of their text. Any one of those rows, then – since they are all the same, defines the long run, or stationary, distribution for such a Chain. [Khinchin 1957] also presents a more concise proof of this.)

The probabilities defined by this *stationary distribution* are the ones we use in order to select the state that represents the homeostatic set point of our homeostatic Markov Chain - but, only if there is a state containing a maximum probability. Otherwise, we don't meet the homeostasis condition.

To summarize the significant points so far: All irreducible (all states form a single ergodic set) homogeneous Markov transition matrices converge to a single stationary distribution. Thus regularity is assured in such a case. Further, if one of the states has a probability that is the maximum, then, in addition, the process represented by this Matrix is homeostatic. And, the "set point" is this maximum state.

Let's now put all of this together and define what we shall mean by *regularity* and by *homeostasis for a Markov Chain*:

A Markov chain is said to *exhibit regularity* if there is a stationary distribution to which it converges ("asymptotically"). The following chapter on Markov chains and autocogeneration will identify the conditions under which a Markov chain exhibits regularity.

Moreover, a Markov chain is said to be *homeostatic* if

1. It is homogeneous,
2. It is irreducible, and
3. Its stationary distribution has a state whose probability is the maximum of the probabilities of the states of the ergodic set. Such a state is called the *set point* of the Markov Chain.

Thus, an irreducible Markov chain is guaranteed to have a stationary probability distribution. (The proof of this must be divided into two cases: the periodic and the aperiodic. The proof of the periodic case requires special treatment [Kemeny and Snell 1969].)

However, if any homogeneous Markov chain runs for enough steps – even if it is not irreducible, it can be shown that it will eventually enter some ergodic subset. For example, a Markov chain may have multiple ergodic subsets. Also it may have subsets that are not ergodic. Nevertheless, it is guaranteed that exactly one of these ergodic subsets will be entered and never left. Which of these possibly multiple ergodic subsets is entered will depend upon the initial conditions under which the chain is run. And whichever ergodic subset is eventually entered (and never left) will have its own stationary distribution.

It is true that – in the case of these Markov chains with multiple ergodic subsets - one cannot deduce which of these stationary distributions will hold without knowing the initial conditions of the chain. Nevertheless, one does know that any Markov chain will have some stationary distribution, and therefore that Organodynamic regularity is inevitable – even if the Markov chain is not irreducible. If the chain is irreducible, then one further knows with precision what the stationary distribution is. Otherwise, one only knows that some

stationary distribution will hold sway – but not which one it is, unless initial conditions are also known.

Thus, to summarize, we know that any homogeneous Markov chain *exhibits Organodynamic regularity*. Further, if the Markov chain's state space contains a single ergodic set, and if it is homogeneous, then we also know what the stationary distribution is.

If a homogeneous Markov chain's state space contains multiple ergodic sets, then we know that it has multiple stationary distributions. Thus, we know that it too eventually exhibits regularity. But we do not know which of its stationary distributions it moves toward.

Finally, for any homogeneous Markov chain, if we know the initial conditions, then we can determine the stationary distribution that it resolves to.

Modeling Irregularity using Markov Chains

We shall now describe the approach we shall use to model *irregularity using* Markov Chains.

A Markov Chain will be defined to be *irregular* if it is non-homogeneous.

According to condition 1 of the definition above, if a Markov Chain is non-homogeneous, it cannot exhibit regularity, and must therefore be irregular. So, changing transition matrices across steps of the process ensures irregularity of a Markov Chain.

Finite Subchains

So far, when defining our notions of regularity and irregularity, we have dealt with entire Markov Chains, which are stochastic processes with an infinite number of steps.

We also know that our notion of *regularity* obtains whenever we have a finite number of contiguous steps that are homogeneous – all of which share the same transition matrix. We also know that our notion of irregularity occurs whenever the transition matrix changes between two steps in a chain.

Therefore, it would be very manageable if we partitioned an entire Markov chain into a collection of *subchains*, each of which uses the same transition matrix throughout the subchain. And, whenever there is a change in transition matrix between steps, we shall start a new subchain at that point. Thus, we will have established a partitioning convention in which all subchains are homogeneous, and non-homogeneity is modeled via a change in subchains. This is the approach we shall adopt.

Moreover, in Organodynamics, all is finite. We shall limit our Markov chains to a finite number of steps. Thus, we can partition our finite Markov chains into a finite number of finite subchains – which we shall do.

In this way, we know that each subchain will be homogeneous and therefore will *exhibit regularity*. *Irregularity* will be modeled by virtue of a change in transition matrix – or through non-homogeneity. If one wanted to model complete irregularity, then one could provide a Markov chain that changed transition matrices at every step. Such a Markov chain would consist of a finite sequence of one-step subchains.

This does present a problem for us when we want to consider such notions as stationary distributions, which we have used above to describe our approach to defining regularity. The reason this is a problem is that the concept of *stationary distribution* was defined as the limit of an infinite sequence of steps. But, now we have limited the number of steps we are dealing with to a finite number.

Fortunately, there is an alternative way to define a stationary distribution for an ergodic subchain – even if it is finite – without requiring the notion of limits of infinite sequences. In the next chapter of this text, where we detailed the concepts that are being introduced here, we shall present this development. For now, suffice it to say that we can define the stationary distribution for a finite homogeneous subchain of our Organodynamic Markov chain.

Moreover, even though our finite subchains never “reach” their stationary distribution, it is reasonable for us to say that, at some point, the subchain will be “heading toward” it. (We shall refer to this as the *propensity* of a homogeneous finite Markov sub-chain to approach a stationary distribution.) We can think of such a stationary distribution as being a “gravity point” for the subchain, and therefore provides a “regularity force” over the steps of the subchain.

Incremental Example Model Development

<>

Progress Check

In Part II, Organodynamics, we are gradually and incrementally developing a mathematical framework for modeling living and lifelike systems according to the seven organizing principles of OCS. Once completed, this modeling framework will achieve the form of an abstract but highly flexible mathematical structure that we are calling an *Organodynamic web*.

We are using the example of a hobbyist’s toolkit for building electronics as an analogy to Organodynamics. Both are systems for “building something”, and in both cases the resulting “thing to be built” has a network structure and has been given a general name.

In the case of the electronics circuit toolkit, the general name of the “thing to be built” is “circuit”. The circuits built by this kit are highly customizable to specific applications. In the case of Organodynamics, the general name of the “thing to be built” is “Organodynamic web”. The models built by this mathematical framework are also highly customizable to specific applications.

Because of the complexity of organic systems, any model that represents them needs to be built in cumulative and increasingly accurate stages. Each stage is generally referred to as an “approximation”. Organodynamics as a model builder’s toolkit provides for six of these approximations. These approximation levels are well defined and occur at precise places in the methodology set forth by Organodynamics.

In this chapter, we have developed two important mechanisms for our Organodynamic web – although we have not as yet mentioned that fact. These two mechanisms are 1) the finite homogeneous Markov subchain, and 2) the finite non-homogenous Markov chain that is the concatenation of multiple finite homogeneous subchains.

In the next chapters we shall see that these are the same as two of the graphical elements of our upcoming Organodynamic web structure. The *finite homogenous Markov subchain* is, in fact, the *segment* that we have already introduced. Secondly, the finite non-homogeneous concatenation of the homogenous subchains is actually the graphical element that we have called the *edge*.

Thus, we have taken two existing elements, the *segment* and the *edge*, and demonstrated how they contribute to the regularity and the irregularity of an organodynamic process within the graphical constructs that we are developing in our drive to provide the equipment to construct the Organodynamic web..

Modeling Autocogeneration using Markov Chains

The Issue Addressed

In the previous chapter we introduced the notion of autocogeneration – a collection of concepts intended to specify the kinds of dynamical system behaviors that characterize the nature of *reorganization* in organic systems. In other words, *autocogeneration* gets specific about what forms *reorganization* takes in Organodynamics.

We described that *autocogeneration* is essentially self-management in its most general form. This includes self-creation and self-maintenance as well as the promotion of both continued regular behavior as well as continued irregular behavior – each whenever the persistence of organization is promoted.

In fact, we can pretty much capture much of autocogeneration by concentrating on the promotion of *regular behavior* whenever it promotes persistence of organization, and the promotion of *irregular behavior* whenever it promotes the persistence of organization. The capability to for a dynamical system to select between *regularity* and *irregularity* in order to promote persistence, we called *regularity selection*.

The previous chapter not only previewed these ideas, but also presented a strategy for using stochastic processes, and specifically Markov chains, to implement a general mathematical model of autocogeneration.

It is the purpose of the present chapter to go beyond the previous chapter, which introduced and motivated these concepts, and to present this model in a more formal and complete manner.

Any framework that models *autocogeneration* must provide a mechanism that represents *co-generation* of a system's components and their relationships as well as mechanisms for *self-regularity*, *self-irregularity* and *regularity selection*. We shall see that co-creation of the components of a system is already endemic to autocogeneration by virtue of the fact that the organodynamic transforms provide the essential equipment for its implementation. For example multiple instances of the Catalyze organodynamic transform can co-operate to support a co-creational model. These capabilities will be explored in a later chapter on simplex Organodynamic webs.

Rather, this chapter will focus on the portions of Organodynamic webs that are articulated via stochastic processes, and in particular, Markov chains. It happens that these portions have already been introduced in the form of *segments* and *edges*. Thus, one could describe this chapter as a detailing of the mathematics (Markov chains) that the Organodynamics framework will use to implement the backbone of Organodynamic webs – its segments and edges.

In order to maintain perspective, we should recall that the states that are involved in the Markov Chains of Organodynamics are actually *organizations* of an underlying population of components. That is, each *condition* of each step of

one of our Markov Chains is an arrangement – actually an *organization* – of the components of our underlying organic system.

This fact in itself is somewhat complicated. So, for the time being we are going to forget about this nature of the conditions, or states, of our Markov Chains. Instead, we will temporarily treat our Markov Chains as being simpler than that, and simply think of the conditions in our Markov Chains as simple states. We will return later, when we have more machinery developed, to remember that each condition, or state, is actually an *organization* of the components of an organic system.

We shall first characterize the general nature of Markov chains, and then show how we shall use them to model organic processes.

OCS Organizing Principle Supported

Organizing Principle # 5: Autocogeneration.

Biological Example of the Issue

We shall refer to the two examples that were mentioned in this same section for the previous chapter. These were: 1) the integration of the protein synthesis process with the gene regulation process in cell biology; and 2) the integration of the endocrine and the neurological systems in multicellular organisms.

In addition, the immune system of multicellular organisms deserves a mention; as do the dynamics of ontogenesis (especially the development of a fetus from a zygote) in sexually reproducing organisms.

Specific Challenges

There is a basic misconception on the part of some scientists [Hoyle and Wickramasinghe 2001] that any probability process is “maximally random” and must inevitably lead to chaos. This misconception must be overcome; and it must be demonstrated that stochastic processes can embody any degree of determinism or randomness on a spectrum whose polarities can include the two extremes of complete randomness all the way to complete determinism.

The Organodynamic Approach to Modeling the Issue

This chapter will leverage the foundations and fundamental results of information theory to demonstrate that stochastic processes, and in particular Markov chains, can lead to stable and non-chaotic long-run behaviors – even if they began in a maximally random state.

From these results, it will follow how to utilize these types of probabilistic mathematical processes to implement the *segments* and *edges* elements of the machinery that we are developing within the Organodynamics framework. Specifically, these segments and edges elements will form the basic graphical elements of Organodynamic webs.

The Role of Markov Chains in Organodynamics

We have introduced stochastic processes as the mechanism that we shall use in Organodynamics to model the unpredictable aspect of system processes. The *reorganizational* organizing principle motivates us to want to model *process*; and the *uncertainty* organizing principle motivates us to want to model process in a manner that supports uncertainty.

The mathematical notion of stochastic process provides a model of both of these organizing principles simultaneously. We have also discussed the need to model dependency conditions in Organodynamic processes. Fortunately, there is a class of stochastic processes called *conditional stochastic processes* that fit this bill nicely.

We have shown that we can use a specific family of conditional stochastic processes, named Markov chains, to model many of the organic system processes that we are interested in in OCS. The good thing about Markov chains is that they are the most tractable form of conditional stochastic processes. Thus, we have chosen to begin with them, and move onto more general conditional stochastic processes in Organodynamics when we need to.

However, process-orientation, stochasticism and stochastic dependency are not quite enough for our chosen modeling mechanism. It must also provide a reasonable model of autocogeneration. Specifically, Markov chains must also provide a model for the various aspects of autocogeneration that we have chronicled thus far. These include regularity, irregularity and co-generation.

This chapter will present the mathematical properties of Markov chains that provide the aspects of regularity and irregularity. Co-generation will be taken up in a later chapter. In order to see that Markov chains can implement regularity and irregularity, we must first investigate their nature and how they can be categorized.

Classification of Markov Chains and Their States

There are many criteria by which one may usefully categorize Markov Chains. It is necessary to make these discernments, because some of them lead to the regularity of process behavior, whereas others lead to irregularity of process behavior. We shall now consider a number of these that are pertinent to our development of Organodynamics.

Recall that a Markov chain is concerned with a process that is changing its state from step-to-step probabilistically. At each step, there is a “state space”, or sample space, of sample points – exactly one of which will be realized at that step.

In Organodynamics the elements of this sample space are complex – each one is a *system organization*. However, in this chapter we shall not be concerned with that – and will ignore it. We shall only be concerned here with what happens in Markov chains when transitions occur – especially long-run

transitions – from time step to time step. For these considerations, it is not important whether the sample space involves has complex or simple states.

We shall be interested in both *homogeneous* and *nonhomogeneous* finite sequences of time steps. Each of these exhibits its own behaviors regarding regularity and irregularity. And each will characterize the basic element of our framework (segments and edges) that we are developing.

Recall that for *homogeneous* Markov chains, the state space is the same for each time step and so are the probabilities. For a nonhomogeneous Markov chains, even if the sample space stays the same, the probability distribution may change. In Organodynamics, we shall even allow the state space itself to change within a nonhomogeneous process. (This is necessary to model “open systems”.) In such a case, of course, at least some of the probabilities must also change.

At each step we have an interest in which one of those sample points (or *states*, or *conditions*) will be realized at the next step. Specifically, Markov chains are interested in which sample point will be realized at the next step – given that one knows which sample point (condition, state) was realized at the current step.

Open Systems and Piecewise-homogeneity in Markov Chains

Unfortunately, living systems do not restrict themselves to homogeneous processes. They very often “change their transition matrices”. Fortunately, though, living systems very often “stay homogeneous” for several steps, and then change their transition matrix between two steps, and then go back to behaving homogeneously for another while.

This change occurs whenever the underlying population of components is allowed to change. When this happens, then their set of possible organizations almost inevitably changes too. This means that the sample space has changed and concomitantly so has the probability distribution.

If such a change is allowed in a dynamical system, it is called an *open* system. If not, it is called a *closed system*. Generally, closed systems are stochastically *homogeneous*; while open systems are stochastically *nonhomogeneous*.

As we introduced in the previous chapter, as a basic element of the Organodynamic framework, a *segment* is a finite step, finite state *homogeneous* Markov chain. A segment represents a closed dynamical system.

However, in Organodynamics, we want to model open systems as well. We shall provide this as follows. For a while, a system will be closed. (That is, no new components are allowed.) Then, at some time steps, new components will be allowed to enter the system and existing components will be allowed to leave. Subsequently, the system will remain closed – but with this changed population of components.

This scenario represents two homogeneous Markov chains that have been “concatenated” in such a way that the entire concatenation is nonhomogeneous, while the individual chains are homogeneous. In a previous chapter we create the segment and edge elements of our framework so that each of these individual chains are named *segments*, while the concatenated chain as a whole is called an *edge*. These are the foundational elements of our framework for modeling living and lifelike systems.

In other words, living systems are very often “piecewise homogeneous”. We shall then, attempt to develop Organodynamics as a theory of “piecewise-homogeneous” Markov chains so that we can leverage homogeneity as much as we can.

Classification of States in a Markov Chain

Consider a homogeneous Markov Chain. Intuition might argue that if one begins at the first step, and then transitions step by step for a large number of steps, that almost anything could happen – assuming of course that the chain under consideration is sufficiently long.

However, it turns out that such intuition is off base. Instead, there is a surprising amount of “order” inherent to all finite homogeneous Markov Chains as their steps are traversed. It turns out that the long run behavior of a homogeneous Markov chain is inherently constrained from “wandering off into the chaos of uncertainty”.

Rather, the degree of uncertainty of the behavior of a homogeneous Markov chain in the long run is limited by its very nature. In this chapter, we shall begin to unearth this matter.

To understand these dynamics, it will be necessary to identify the various states to which a Markov chain can obtain, and to differentiate among their constraints of behavior.

We shall now investigate the nature of the states of Markov Chains, and see how they are naturally categorized. We shall not develop the theory of finite Markov Chains here, but instead will refer the reader to sources such as [Kemeny and Snell 1969; Bharucha-Reid 1960; Iosifescu 1980] for more details. However, we shall quickly summarize results of this theory that are pertinent to this text.

We shall now proceed, without proof, to summarize some results of finite Markov Chain theory that are important to Organodynamics.

Suffice it to say that the states (conditions) of a Markov Chain can be partitioned into subsets, and these subsets can be characterized into two types: 1) *transient*, and 2) *ergodic*. These two categories pertain to whether or not certain states can “communicate” with each other. It turns out that, in a Markov chain, given that a certain state is realized at one step of the chain, it is either possible or impossible to get to from that state to another state after a certain

number of steps. If so, the first state is said to *communicate* with the second state in that chain.

If one state does communicate with a second state, and also the second state communicates with the first, then we say that the two states communicate with each other. Having described the notion of state communication, we are prepared to describe the two notions of transient and ergodic sets of states.

So, we interested in whether it is a useful idea to try to partition the states of a Markov chain into subsets in such a way that we can say some interesting things about those subsets. And we are interested in whether the states of each of those subset can communicate with any states in any others of those subsets.

It turns out that it is possible to partition the subsets of a Markov chain in this way. In fact, we shall see that it is possible to partition all of the states into subset in such a way that some of the subsets are “self-contained” with respect to inter-communication – while others of the subsets in the partition are not. This criterion will divide all of these subsets into two categories, as we shall shortly see. They are called *transient* and *ergodic*.

The states in *transient set* can communicate to other states within that set as well as with other states outside of that subset (and therefore in other subsets). On the other hand, the states of an *ergodic set* can only communicate among themselves, and cannot communicate to states outside of their set.

In summary [Kemeny and Snell 1969],

- Once a process enters an ergodic set, it can never leave.
- And once a process leaves a transient set, it can never reenter.

It may seem that there would be other possibilities for these subsets than just these two. But there are not. It is this, perhaps surprising, nature of Markov chains that, as we shall see, lends them a substantial amount of “order”, or *regularity*.

This unexpected regularity, which is a result of the stochastic dependency of conditional probabilities, brings with it the ability for Markov chains to start with a random and “chaotic” situation and inevitably resolve to a highly predictable, often deterministic, behavior over time. And we shall use this fact to countermand the misconception often held even among noted scientists that randomness must only lead to chaos.

But, I digress. Let's get back to *transience* and *ergodicity*. Either a subset of the sample space has the property that once entered, it is never left. Or it has the property that once left, it is never re-entered. There is no “middle case” and no combined case [Kemeny and Snell 1969]. This is true no matter what their conditional distributions are – no matter what their transition matrix is.

And, here are more facts about Markov chains. A Markov Chain has at least one ergodic set, and it may have more. A Markov Chain may have one or more

transient steps. But it may not have any. Even if a Markov Chain has more than one ergodic set, only one will be entered in a single execution.

So, in general, a Markov Chain can have several ergodic sets (it must have at least one), and it can have several transient sets (but may have none). Additionally, a partial ordering will exist among these sets with respect to the order in which they can be entered. All of the ergodic sets of a homogeneous chain are at the same minimal level of this partial ordering. On the other hand, some number of the transient sets is at the maximal of this partial ordering. There may also be some number of transient sets at intermediate positions in this partial ordering.

All of this means the following: a process may take its first step into a state that resides in any of these sets. This first step may be into either a transient set or into an ergodic set. If it enters at a state that is in an ergodic set, then the process will never leave that ergodic set. Of course, once an ergodic set is entered it is never left.

On the other hand, if the initial state is in a transient set, then it can eventually leave that set and enter another set. The next set may be either transient (if there is another transient set) or ergodic. If it is ergodic, then the process will never leave that set. If it is transient, then the process will eventually leave that set and go to another – and so on. Eventually, if the process executes long enough, it will enter exactly one of the ergodic sets, where it will stay until the last step of the process completes.

Thus, we can initially enter a Markov chain at any of these states. If the initial state is in a transient set, then we shall eventually “cascade” through all of the other transient sets that are below the initial transient set in the ordering until we eventually enter into exactly one of the ergodic sets. We shall then stay in that ergodic set for the remainder of the steps of the chain. None of the other ergodic sets will ever be entered. And none of the transient sets that were above the initially entered transient set will ever be entered.

On the other hand, if the initial state is in an ergodic set, then we shall never leave that ergodic set for the remainder of the chain.

Of course, since our Markov Chains in Organodynamics have been stipulated to have a finite number of steps, then the process may end without actually ever reaching one of the ergodic sets.

Nevertheless, because of the nature of these transient and ergodic sets, a Markov Chain is always “heading toward” one of its ergodic sets. We call this propensity. And if it arrives there, it will never leave that ergodic set.

Notice that this “heading toward” something is a kind of “tendency toward regularity”. We shall develop this idea further below.

Behavioral Dynamics of Homogeneous Markov Chains

However, before we discuss what all of this has to do with *regulation*, lets see how these ideas come to life when we represent our Markov Chains with a transition matrices that we discussed earlier.

Transition Matrix Canonical Form

It will be very convenient for all that follows if we place some consistency on how we organize transition matrices in such a way that both the ergodic and the transient sets of states are conspicuous and manageable. Markov chain theory has established a canonical form for the Markov transition matrix that organizes the rows and columns in such as way as to separate the important types of states and permits an intuitive understanding of the dynamics of long run behavior of Markov chains [Kemeny and Snell 1969; Bharucha-Reid 1960; Iosifescu 1980]. We shall explain this form in this subsection, since its understanding is essential to what follows.

Suppose we have a sample space of conditions for which we want to construct a Markov transition matrix. Each of these conditions is an alternative that can be manifested during the current time step, and that could also be manifested at the next time step.

We represent this fact by constructing the transition matrix so that 1) the possibility that each of these conditions can manifest in the current time step is represented by exactly one row in the matrix; and 2) the possibility that each of these conditions can manifest in the next time step is represented by exactly one column in the matrix.

Now, it does not matter from an application perspective – or from a matrix algebra one, what the order of the rows of the matrix are – or what order the columns are in either. Thus, we are free to rearrange the orders of these rows and columns any we a wish.

We shall therefore rearrange the order of the rows and columns to accommodate the fact that the conditions that represent those rows and columns are partitioned into a collection of subsets – some of which are *ergodic subsets* and the others of which are *transient subsets*.

We just saw in the previous subsection that these subsets enjoy a particular kind of ordering. This is the ordering that would necessarily be followed if one were to “jump into a Markov chain” at some arbitrary state, and then follow the time steps going forward. We saw that, eventually, one would end up in an ergodic subset of these states and never leave it.

The scheme is that one would go through all of the transient subsets in the uniquely prescribed order for that matrix until one enters one of the ergodic sets – which one would then never leave. Thus, only one of the ergodic sets would ever be visited.

All of this means that the this ordering of all of these subsets is such that:

1. All of the ergodic subsets are last in the ordering, and exactly one of them will be entered during the life of the chain. Moreover, the transient sets are not ordered, because only one of them is entered.
2. The transient sets entered firsts, but are placed in a specific order of visitation.

Because of these features, a *canonical* form has been established for transition matrices of homogeneous Markov chains. This form places all of the rows of a given subset (transient or ergodic) together; and moreover, places all of the rows of ergodic subsets at the top of the matrix. It does not matter which order the ergodic sets are placed in, since they are unordered.

After all of the rows of the ergodic sets have been assigned, then the rows for the transient subsets are assigned to the higher numbered rows. Again, the rows of any subset are placed together in the matrix.

Moreover, since the transient sets *do* have an order of visitation, then they respect that order. Specifically, transient subset appears in the *reverse* order of visitation. Thus, the later a transient subset appears in the order of visitation, the later its rows appear in the matrix. This puts the later transient subsets nearer the rows of the ergodic subsets.

In this way, a specific “execution” of a Markov chain can be seen as entering at some row in the matrix and then eventually working its way upward (through decreasing row subscripts) through all of the transient states until it finally leaves the last transient subset – at which point it enters a row of some ergodic subset. It then never leaves the rows of that ergodic subset of states.

Reapply these same ordering conventions to the columns as well, and we have the canonical format of a Markov transition matrix.

Another way to say this is that the general canonical transition matrix looks like this: The uppermost rows and the leftmost columns represent all of the ergodic sets. These sets can be in any order with respect to each other. The remaining rows and columns represent the transient sets in reverse order of their entry.

A detailed discussion of the canonical structure of the Markov transition matrix is beyond the scope of this text. Please see [Kemeny and Snell 1969; Bharucha-Reid 1960; Iosifescu 1980] for more.

Transition Matrix for Long Run Behavior

At this point, we have developed the use of Markov Chains to model the dynamics (behavior) of organic systems. We want to be able to use these models to quickly describe long run behavior within such a process.

We shall demonstrate this by developing a stochastic algorithm for modeling the long run behavior of a Markov Chain.

We shall do this in the form of calculating a transition matrix for the Nth step of a *homogeneous* Markov Chain.

The algorithm to calculate the long run Nth-step transition matrix: perform matrix multiplication of all of the transition matrices for steps 1 through N.

We can extrapolate this for any number of arguments to obtain the algorithm to calculate the long run Nth-step transition matrix for a homogeneous Markov Chain. Since the transition matrices for all of the steps of a homogeneous Markov Chain are the same matrix, then the algorithm becomes:

$$T^N$$

Where T is the transition matrix and N is the number of long run steps of the homogeneous chain. In this case, the Nth power means to perform matrix multiplication of the transition matrix on itself N times.

This is possible since all of the transition matrices of a finite homogeneous Markov chain (or segment) are square matrices of the same order. It can be shown that the product of any two transition matrices is in fact another transition matrix. (That is, that all of the rows are conditional probability distributions.) See [Kemeny and Snell 1969; Bharucha-Reid 1960; Iosifescu 1980] for more.

Calculating the Realized Probability Distribution for Long Run Behavior

We discussed earlier that a Markov model is a conditional stochastic model. This means that, for each step, the model provides multiple probability distributions (one for each possible current condition) – rather than just one probability distribution.

Clearly, this situation is even more unpredictable than if we only had one probability distribution to deal with. Unfortunately, a conditional stochastic model does not give us enough information to reduce these multiple distributions to just one. However, if we had the right kind of “extra information”, or conditions, then we could reduce our model to an unconditional model, in which there is only one distribution to deal with. At that point, then, we would know what the actual probabilities are for the next step.

There are a number of types of conditions (extra information) that provide sufficient information to be able to reduce the multiple conditional probability distributions to a single distribution. One type is called *initial conditions*.

Initial conditions can simply specify one of the multiple conditional distributions to use as the non-conditional distribution for the step. More generally, a set of initial conditions can provide a set of “weighting values” that can be used to calculate (average) a resulting non-conditional distribution from a linear combination of the conditional distributions. Of course, these “weights” can be provided in the form of an *initial conditions row vector*.

If one has an initial conditions row vector, then the realized probability distribution row vector that describes the probabilities of transition from initial

state to the Nth state can be found by multiplying the initial condition row vector by the long run N-step transition matrix.

Performing such an operation has the effect of moving from the Markov model with multiple conditional probability distributions to a model in which we have a single probability distribution. We described this in a previous chapter as moving from a conditional stochastic model to an unconditional stochastic model. This is a movement to more certainty from less certainty.

Now, for a homogeneous transition matrix, let's consider how to calculate the realized probability distribution row vector, given that you have an initial conditions row vector. The formula is:

$$i * T^N$$

Where, again, T is the homogeneous transition matrix, N is the number of long run steps and i is an initial conditions row vector.

Armed with this machinery, we are now in a position to discuss *regularity* in organic processes as modeled by conditional finite stochastic processes in the specific form of finite Markov Chains.

Regularity, Irregularity and Autocogeneration in Markov Chains

In order to pass a “believability test” for being “lifelike”, an organic process must not be “too regular”, and it must not be “too chaotic”. Rather. Its behavior should move around between these two extremes without staying too long on either one of them; and generally spend a great deal of time somewhere “in the middle” between chaos and certainty.

We want to show that our piecewise homogeneous Markov chain can model this type of behavior.

The Context for Regularity in Organic Systems

We are currently discussing how this behavior is addressed by the fifth OCS organizing principle, *autocogeneration*. This organizing principle states that organic systems don't just go wandering around chaotically. Rather they tend to exhibit some kind of *regularity* in their reorganizational behavior - particularly in homogeneous finite Markov Chains – and in therefore within the segments of our piecewise homogeneous chains. Certainly, the behavior of organic systems occasionally visits, or “flirts with” these extremes.

But as long as the organic system persists (“survives”), then it has a propensity to return to a middle degree of uncertainty between these extremes of certainty and chaos. It is this propensity toward returning to an intermediate degree of uncertainty that we wish to model via the autocogeneration organizing principle.

Of course, some piecewise homogeneous Markov chains *do* “wander off” into “chaos”, by which we mean tend to large uncertainty, and some “succumb” to very low uncertainty in the long run. But these “don't survive”. What is essential

Necessary, however, is that piecewise-homogeneous Markov can characteristically exhibit long run behavior that has exhibits “contained uncertainty” – intermediate degrees of uncertainty, regularity of behavior. We intend to demonstrate this in this chapter.

Regularity Defined

Regularity in organic processes may take many forms. For example, organic processes may exhibit cyclic behavior. An example of this is the various cycles found in cellular metabolism: the citric acid (Krebs) cycle comes to mind. In this situation, after a certain number of steps of the organic process have taken place, the set of states being visited has become reduced to a specific subset, and none of the other states outside of that subset is visited thereafter. Moreover, in a cycle, the visitation of these remaining states occurs in a strict order.

[Of course, this strict order is only observed if the conditions are right (enough food molecules). These networks are actually stochastic [Watson 1969]. But, within those conditions, these “cycles” do behave as described here.]

A weaker form of regularity is similar to cyclic behavior, except that we remove the restriction that the visitations occur in a certain order. In this case, after a certain number of steps have been taken in the process, the visitations become restricted to a certain set of states – but we no longer require that they be visited in any particular order. In fact, this situation is about as weak as we want to get in Organodynamics and still refer to the behavior as “regularity”.

For that reason, we shall use this description to define *regularity* in Organodynamics more formally below. But first, let us make an observation about something that these two examples (cyclic and noncyclic) of *regularity* have in common. Then we shall look at some other examples of types of regularity before providing our formal definition of it for Organodynamics.

Specifically, notice that the descriptions of what occurs in both of these examples fits our definition of an *ergodic set* of a homogeneous Markov Chain! That is, in both of these cases we have presented a subset of the entire sample space that, once entered, is never left. But this is our definition of *ergodicity*.

Thus, we shall define *regularity of an organic process* in Organodynamics as follows:

Regularity: An organic process is said to exhibit *regularity* if it is homogeneous and ergodic.

Stationary Distributions and Regularity

The reader deserves more precision in our definition of regularity in Markov chains, and therefore in our organic processes. We shall provide some discussion toward that here, and refer the reader to a more detailed treatment in other references, particularly [Kemeny and Snell 1976], [Bharucha-Reid 1960], and [Iosifescu 1980].

Regularity in Irreducible Markov Chains

It turns out that for a certain simple type of Markov chain, it is easy to define regularity. This kind of Markov chain is one that contains only a single ergodic set of states – and no transient states. Fortunately, this type of Markov chain occurs often in living systems – including our canonical example based upon a closed set of atoms.

Such a Markov transition matrix is called *irreducible*.

The Stationary Distribution for an Irreducible Markov Chain

It turns out that any irreducible Markov matrix can be shown to converge in the long run to a single probability distribution for its states.

What this means is this: For the transition matrix that describes the long run behavior of an irreducible chain, all of its rows are the same!

(The proof of this fact usually divides the problem into the two types of irreducible chains – aperiodic and periodic. But both cases eventually lead to the same conclusion [Kemeny and Snell 1976, Chapters IV and V].)

But each row of a transition matrix represents the conditional probability distribution for that particular state. But such a matrix would say that the conditional probability distributions of all of the sample space state are the same probability distribution – in which case they are no longer conditional!

In other words, we can replace the multiple conditional probability distributions with a single probability distribution. This single (unconditional) probability distribution for an irreducible Markov chain is called its *stationary distribution*.

Thus we apparently have a reduction in uncertainty for irreducible Markov chains by virtue of the fact that we have gone from multiple probability distributions to one – at least as far as long run behavior is concerned.

But, in fact, we are concerned with long run behavior, so this apparent reduction in the degree of uncertainty is significant.

All Possible Initial Conditions Lead to the Same Stationary Distribution

Notice also that no assumptions have been made about the initial conditions of the irreducible Markov chain. They all asymptotically approach the same stationary transition matrix – regardless of what the initial condition row vector is!

For any Markov chain, if one begins with a set of initial conditions, one can always resolve any future step of the process. This can be seen by multiply it out. A $1 \times N$ row vector represents the initial conditions, while the transition matrix is $N \times N$. Suppose that we wanted to calculate the unconditional distribution for the k -th step of the chain. This is:

$$i * T^k$$

This is a $1 \times N$ row vector multiplied by an $N \times N$ matrix. The result is a $1 \times N$ row vector – which represents a single (unconditional) probability distribution for step k of the Markov chain.

However, the stationary distribution (at the “infinite step”) is not determined by any initial conditions, but is “unswayed” by them.

The Entropy of an Irreducible Markov Chain

We just mentioned the “degree of uncertainty” in an irreducible Markov chain. However, the astute reader will realize that we have given a definition to the idea of a single probability distribution. However, we have not yet given any formal meaning to the notion of “the degree of uncertainty of a *set of probability distributions*”, nor have we given a definition for the notion of “the degree of uncertainty of an entire Markov chain”.

Intuition suggests that we had in some sense “reduced uncertainty” because of the fact that we went from many to one probability distribution. However, we have yet to define any notion of the “degree of uncertainty of an entire Markov chain”, so we cannot say that – yet.

However, for irreducible Markov chains, we now have a way of reducing these multiple distributions to a single distribution – at least for long run (countably infinite) steps – in the form of the *stationary distribution*. (We call this *propensity behavior*.)

Moreover, this stationary distribution does not represent any single *time step* of a Markov chain. Rather, it is a model of the entire long run behavior of the chain, starting with the first time step and including all of them. This stationary distribution is the results of a limiting transition matrix, and thus represents the entire chain.

And, the fact that the stationary distribution is a single distribution means that we, in fact, *do* have a measure of uncertainty defined for it – Shannon entropy.

Thus, if we can determine the stationary distribution for an irreducible Markov chain, then we can calculate the degree of uncertainty of the entire chain. We shall examine the determination of this stationary distribution in the next subsection – given that we know the underlying transition matrix for the chain.

(Terminology note 1: It should be noted that aperiodic irreducible Markov chains are sometimes called “regular” in Markov theory [Kemeny and Snell 1976]. However, we are using the term *regularity* in Organodynamics in the more general sense already described. In our usage, all finite Markov chains, in our sense, exhibit regularity. However, we shall stay away from calling all finite Markov chains “regular” since that already has a more specific meaning. It should also be noted that periodic irreducible chains are called “cyclic”, and that another name for both kinds of irreducible Markov chains is the term *ergodic Markov Chain*. This usage has “ergodic” as an adjective of an entire chain. Whereas, our usage has it as an adjective to a subset of the state space of a homogeneous chain.)

(Terminology note 2: Homogeneous stochastic processes are often called *stationary*. This is the use of the term “stationary” to mean “time-homogeneous”. It is not to be confused with the notion of a stationary distribution of an asymptotic stochastic process that was presented above. Because of this confusion of terminology around this term, we have decided to reduce name conflicts and refer to time-homogeneous stochastic processes as *time-homogeneous* and not “stationary”. However, beware that sometimes the usage of the *stationary* in the literature refers to time-homogeneity rather than the stationary distribution that we have discussed.)

Calculating the Stationary Distribution of an Irreducible Markov Chain

The notion that a single probability distribution, the stationary distribution for the chain, can replace the entire set of conditional distributions to describe the long run regularity of the behavior of an irregular Markov chain is a powerful idea.

It would be very convenient to be able to calculate such a distribution, and use it to calculate its entropy – which we can then use to characterize the regularity of the finite homogeneous Markov chains that we are using in Organodynamics.

In fact, these stationary distributions could be used to individually characterize each segment of the piecewise-homogeneous Markov processes that we are using in Organodynamics to model organic processes – at least as long as the homogeneous chain consists of a single ergodic set (is irreducible). We will find that most of the organic processes that we want to model can be modeled as irreducible Markov chains. And if not, they can be decomposed into ones that are.

Thus, it seems that we have an approach to calculating the entropy of most any segment of our piecewise-homogeneous chains.

Actually there are two problems. The main problem is that our chains all have a finite number of steps, and therefore never proceed to a limit. The second problem is simply the complication of calculating the limits of a matrix. However, this is more an inconvenience than a problem.

It is the first problem that we must overcome. However, we can overcome both issues at once, because there is an alternative way to calculate the stationary distribution.

Fortunately, there is a theorem [Kemeny and Snell 1969; Bharucha-Reid 1960; Iosifescu 1980] that tells us that, for a given ergodic set, how we can calculate its stationary distribution without having to take the limit of the positive integral powers of its transition matrices.

Essentially, the theorem says that the stationary distribution of an irreducible homogeneous Markov chain is *omnipotent* with respect to matrix multiplication with the transition matrix of that Markov chain. And this gives a way to consider stationarity as an algebraic property of an irreducible homogeneous Markov chain – and not have to consider it as a limit of a function of an infinite sequence!

Of course, this also tells us how to calculate the stationary distribution. We need only find a row vector that can be multiplied by the transition matrix of the ergodic set that gives *itself* as the answer. (That is, find the row vector that is *omnipotent* with respect to the transition submatrix for the ergodic set.)

In other words, find a row vector x such that $x \cdot T = x$, where T is the transition matrix of the ergodic Markov chain!

Solving for this row vector x requires a simple system of linear equations, and is thus theoretically tractable to any reader. (Practical tractability is another question, due to the size of these sample spaces and their resulting transition matrices.)

The resultant row vector produced by this matrix algebra solution is a probability vector, and we shall refer to its content as the *stationary distribution* row vector for irreducible Markov chain.

Of course, we still have the problem that our segments are actually finite, and never progress to the limit that is the stationary distribution. However, we shall solve this problem by “defining it away”. Our approach will be to proclaim that these finite Markov chains have a “propensity” toward their stationary distributions, and that we shall assign the degree of uncertainty of the chain as its propensity – which we define as the Shannon entropy of the stationary distribution of the homogeneous chain (the segment).

Regularity in General Homogeneous Markov Chains

Thus, we expect that our approach of using a stationary distribution that we have presented above will work for most any organic system that we intend this framework to be able to model.

However, let see if we can generalize our approach to deal with more general Markov chains – chains that are not irreducible. IN order to do that, lets take stock of where we are at this point.

Let start with the fact that if we have access to a set of initial conditions, then we can calculate the unconditional probability distribution for any specific time step of any Markov chain – regardless of how many ergodic or transient states it has, and regardless of whether it is homogeneous or nonhomogeneous. Moreover, since all of our Markov chains are piecewise homogeneous and finite, we can calculate a single probability distribution for the entire chain – and therefore can calculate the Shannon entropy for that distribution and for the entire piecewise homogeneous Markov chain.

Therefore, if we have a set of initial conditions for any of segments or our edges, we can calculate an entropy value for it.

However, if we do not have an initial conditions row vector, then, so far, we can only calculate a stationary distribution for the chain – and entropy for the chain, if it is an irreducible chain.

To see if we can generalize our solution, let's review our inventory of types of Markov chains and see what we have yet to solve.

First, we have homogeneous and nonhomogeneous chains. We haven't been able to say much more about chains that are nonhomogeneous. However, we have isolated the problem of nonhomogeneous chains by partitioning them into homogeneous subchains, which we are calling *segments*. Right now, we are going to confine our concerns to developing entropy measures to segments. We shall return to calculating entropy for edges as a function of their segments in a later chapter.

So for the time being, we have confined our problem to calculating entropies for homogeneous Markov chains. And we currently have a solution for one particular type of homogeneous chain – the irreducible ones. We will now proceed to investigate other types of homogeneous chains.

Within homogeneous chains, we can partition our state space into transient and ergodic subsets. All homogeneous Markov chains must have at least one ergodic subset. If the entire state space is a single ergodic set, then we say that the chain is irreducible. We have just shown that all irreducible Markov chains have a stationary distribution that we can calculate.

Let's get more general. We also know that the state space of a homogeneous Markov chain may have more than one ergodic subset. However, if it does, then we also know that exactly one of them will be entered and never left. We also know that each of these multiple ergodic subsets also has its unique stationary distribution to which the long run behavior of the chain is destined – given that the chain enters its ergodic subset.

Thus, in the case of multiple ergodic subsets, and no transient sets, we know that the long run behavior of any execution of the chain will eventually be under the control of exactly one stationary distribution – we just don't know which one. (Unless we apply a set of initial conditions.)

We also know that the state space of a homogeneous Markov chain may have at least one subset that is transient. Even so, the chain will have at least one ergodic set. We also know that, if the chain runs long enough, it will eventually enter exactly one of the ergodic subsets and never leave. Thus, it will eventually be under the control of at least one stationary distribution.

In summary, any homogeneous Markov chain will have at least one stationary distribution that controls its long run behavior – one such distribution for each of its ergodic subsets.

1. If the chain has exactly one ergodic subset, then its stationary distribution governs the long run behavior of the chain – and we can calculate it. In this case the long run behavior of the process is guaranteed to *exhibit regularity*, and we can calculate the precise stationary distribution that describes that regularity.

2. If the chain has more than one ergodic subset, then it has one stationary distribution for each of those subsets. Exactly one of these stationary distributions will govern the long run behavior of any specific execution of the chain. Which of these so governs an execution can be determined by a set of initial conditions. Absent these initial conditions, the governing stationary distribution is uncertain. In this case the long run behavior of the process is guaranteed to *exhibit regularity*, but we cannot calculate the precise stationary distribution that describes that regularity.

Thus, in all cases of homogeneous Markov chains, we know that, if the process runs long enough, that *regularity is exhibited*.

And, if the process does not run long enough, then we have defined away the problem of calculating its entropy by calculating its *propensity*. The reader will recall that the propensity is defined as the idempotent row vector as discussed above.

Thus, for any homogeneous finite Markov chain, we have defined the notion of a limiting distribution, which we are in general designating as its *stationary distribution*. Thus we can calculate an entropy value for any Organodynamic *segment* as the entropy of its stationary distribution.

Specific Types of Regularity

Homeostasis

A special case of regularity is *homeostasis*. Homeostasis can be interpreted as “same state”. The implication is a behavior that allows for change of state across the steps of a process, but with a propensity for the process to revisit a certain state as the process proceeds.

Mechanisms for Implementing Homeostasis

Homeostasis in biological systems is seen as the chief mechanism by which dynamical systems are “regulated”. In Cybernetics, a specific component, or subset of components, have the responsibility of providing this regulation to the system overall. For example, working together, the endocrine and nervous systems keep the overall temperature of the human body at a particular temperature (for example 98.6 degrees F.)

In fact, cybernetics is based upon the principle that such a subset of components, called the *regulator*, exists and that it maintains a negative feedback loop of information that it uses to maintain the system at its set point.

In our parlance, the set point would be the statistical *mode* of the unimodal stationary distribution that exists in cases where we homeostasis is exhibited. But, homeostasis is more general than Cybernetic systems, because it does not require, but does allow, a regulator or a feedback loop. All that is required is that there is a unimodal stationary distribution.

Homeostasis can be seen as a special type of *noncyclic regularity* in which one of the states of the ergodic set is visited more often than the other states of that set. In other words, homeostasis exists when the stationary distribution of the Markov chain is such that the probability of one of the states in the ergodic set is higher than the probabilities of any of the other states in that ergodic set.

In the Jargon of Organodynamics, we would say, “the realized probability distribution row vector is *unimodal*”. Thus, we can define homeostasis as follows:

Homeostasis: An organic process that exhibits regularity is further said to exhibit homeostasis if its stationary distribution is unimodal.

We can determine whether regularity is, in fact, homeostasis by inspecting its stationary distribution to see if it is unimodal. If so, then we have heterostasis for the ergodic set in question. Or, it may be multi-modal, or not modal at all. In any event, it will represent an equilibrium distribution for the corresponding finite Markov Chain, or segment of our piecewise-homogeneous Markov chain.

Homeostasis and Random Variables

The *set point* of a homeostatic process is normally expected to be a real number. However, above, we have defined the set point of an Organodynamic process as the mode of a unimodal stationary distribution of an Organodynamics Markov chain.

But, a mode is a sample point, and sample point of Organodynamic processes have been defined to be *system organizations* – which are sets, not real numbers. This is not unusual. Many probability spaces – perhaps most in practice - have sample spaces whose states are not real numbers.

But many investigators prefer to have sample spaces whose states are real numbers so that a variety of statistics can be calculated on them. Some of these statistics are mean, median, mode, variance, standard deviation and other so-called *moments*. But these statistics cannot be calculated against sample points that are not real numbers, such as *system organizations*, coin flips, playing cards, soups or nuts.

The solution is to create a mapping from the initial sample space to a new sample space whose outcomes are, in fact, real numbers. However, this mapping must preserve the assignment of probabilities to the new sample space so that it can be used as a probabilistic representation of the initial probability space.

Such a mapping is called a *random variable*. Common usage actually has the new resulting probability space also referred to as the *random variable* [Doob 1953].

We described random variables in the early chapter on uncertainty above and will not repeat the explanation.

The idea of random variables surfaces in typical examples of homeostasis in biological systems because the sample spaces populated by complex assemblies of atoms and molecules (as are our system organizations in Organodynamics), but the practitioners are accustomed to grouping these assemblies together in categories, all of which are assigned the same real number.

For example, the normal homeostatic “set point” for the overall temperature of the human body is said to be 98.6 degrees F. In actuality, this number is associated with a collection of organizations of several levels of organization of the human body, all of which result (are mapped to) a temperature of 98.6.

Thus, where as the initial sample space is a space whose outcomes are actually configurations of atoms and molecules, it is preferable to medical practitioners if it is mapped to another probability distribution whose “conditions” are associates with real numbers that represent temperature. However, not any mapping will due, because the probabilities involved must be preserved between the two probability spaces.

Thus, homeostasis, with the use of set points, usually requires the use of random variables.

Other Special Cases of Regularity

Homeostasis is a stricter form of *regularity* than is the general form of regulation that we described above – wherein it was not required that one of the states of the ergodic set exhibit maximum probability.

Slightly more general forms of regulation than homeostasis occur when *the realized probability distribution row vector* is bi-modal. Regulation cases in which the realized probability distribution row vector is tri-modal or even quadra-modal offer increasingly general cases of regularity.

An even stricter form of *regularity* than homeostasis is the case that an ergodic set has exactly one state in it. In Markov Chain theory, such a state is called an *absorbing state*. In this case, there is certainty, whenever this ergodic set is entered, that this single state will eventually be visited and never be left – until the termination of the homogeneous Markov Chain (which is bound to occur since our chains in Organodynamics are all finite).

An absorbing state could be described as a strong form of homeostasis, wherein the realized probability distribution is unimodal because of the fact that it has only one state. An absorbing Markov chain is the stochastic process representation of determinism.

Of all these examples, the most inclusive form of *regularity* consists of simply stipulating that the chain in questions is homogeneous and that it contains at least one ergodic set. I

Of course, as we have discussed, all homogeneous Markov chains have at least one ergodic set. Thus, all homogeneous Markov chains do, in fact, exhibit

at least this weak form of regularity. This fact implies that all segments in the Organodynamic framework exhibit at least weak regularity.

This is precisely how we have defined *regularity* in Organodynamics: as ergodicity in homogeneous chains.

Complex Topics in Regularity

Multivariate Regularity and Joint Probability Distributions

Living systems must typically manage the homeostasis of several random variables at the same time.

These variables may or may not be stochastically interdependent. In fact, multiple variables may exhibit varying degrees of pairwise stochastic dependence. These degrees of pairwise dependence are addressed by the mathematical notion of covariance. We shall not define this measure in this text. Suffice it to say that the degree of stochastic dependence between two sample spaces (variables) is measurable.

In probability theory, when multiple variables may or may not be pairwise conditional or not (interdependent), we model them with the concept of a *joint distribution*.

A joint distribution describes the probabilities of the sample points in a new type of combined sample space called a joint sample space. A joint sample space among a set of variables is the Cartesian product of all of the sample points of each of the variables involved.

Thus, a sample point of a joint sample space from N variables will be a vector of N elements. The joint distribution of this joint sample space will assign a probability to each these vectors that are the elements of that Cartesian “product space”.

Once we have a joint distribution for a collection of sample spaces, then we can perform a number of statistical operations that assess the varying degrees of pairwise interdependence between its probability spaces.

The joint distribution is the framework that Organodynamics will use to address the notion of multivariate homeostasis. For example, in the human body, there are a number of individual state spaces, all of which must maintain homeostasis in order to manage the body in “good health”.

Most interesting, many pairs of these probability spaces are mutually dependent. If the autocogeneration of the body responds in such a way as to move the state of one of these variables toward its set point, the dependency between this state space and another may cause the other to move away from its set point – at least for certain values of the two sample spaces.

However, there may also be value ranges for both of these interdependent spaces where they can both be moved toward their set points simultaneously.

If we modeled these two sample spaces as a joint sample space and its joint distribution, then the sample points of such a joint sample space will be pairs of values – where the first point of a pair is from one sample space and the second is from the other.

If there are points in both spaces that can move together simultaneously toward their respective set points, as described above, then there will also be a set point in the joint sample space. It, in fact, will be the pair whose entries are the respective sample points from the two individual sample spaces. Moreover, the ranges of values from the separate sample spaces that could both move toward their respective set points simultaneously are represented by a set of corresponding sets of points in the joint sample space.

In realistic Organodynamic systems, it should be expected that a number of state spaces (sample spaces) will be identified, all of which will individually exhibit some kind of regularity behavior. Moreover, this collection of state spaces will generally be interdependent. It will be frequently necessary to model these as a joint sample space with its joint probability distribution in order to study the overall regularity of the organic system involved.

Such a model will permit the study of the interdependent relationships among the components of the sample space; and at the same time permit the study of the regularity of the system – and in particular the multivariate homeostasis of the system.

Stochastic Interdependence and Joint Probability Spaces

The previous section discussed how multiple probability spaces could interact so that their respective sample points can form joint sample points. However, such an interaction also represents a new “joint” sample space whose sample space is these new joint sample points. This new sample space is referred to as the *joint sample space*, or the product space, of the two initial probability spaces.

Moreover, one can investigate the degree to which the sample points of the two initial probability spaces are interdependent by comparing the probabilities of the joint sample points with the product of the probabilities of the individual sample points from their individual probability spaces. If the product of the two individuals is equal to the joint sample point, then the two are independent. Otherwise they are dependent.

For example, suppose we have two probability spaces, both pertaining to the selections from the menu of expensive restaurants in San Francisco. Probability space *W* is the selection of wine by world region; and it includes, France, Italy, Australia, and the US. Probability space *F* is the selection of entrees, and includes, red meat, white meat, seafood and vegetarian entrees.

The question is, are these two probability spaces statistically independent. If so, then the product of the probabilities of events from the two spaces individually equals the corresponding joint events from the joint space.

As we shall discuss in a later chapter that discusses information theory and stochastic processes, we shall see that stochastic dependence can have a severely limiting affect on the long run degree of uncertainty associated with a stochastic process.

For example, if two stochastic process have precisely the same probability spaces defined on a step by step basis; they can however exhibit strongly different long run entropies if their degree of stochastic dependence between time step dimensions is different.

Stochastic Processes and Joint Probability Distributions

The above discussion of joint probability distribution implied that the joint events involved in the probability spaces were pairwise occurring concurrently. While this is often the case, it is also possible to consider joint events, and joint probability distributions, where the events in the joint set are all occurring at different times.

The mathematics does not “care” when the multiple events occur.

Therefore, the use of joint probability distributions is widespread in the study of stochastic processes. Generally, each of the time steps of a stochastic process is treated as a distinct probability space (or random variable). Thus, the mathematics of stochastic processes is generally studied as a countably dimensioned joint (or product) space, where each time step is managed as a distinct dimension.

This is a little simpler for the finite step stochastic processes we deal with in the Organodynamic framework, because the joint probability distributions that we start with involve finite dimensional product spaces.

However, when we discuss nested systems below, we shall encounter reasons to model other conditional events that we may want to model as distinct dimensions outside of time step considerations. This will bring up the need to mix time and concurrent probability spaces together in a highly integrated joint probability distribution.

This can get complex. However, there is precedence for this in physics, economics, systems biology and many other disciplines.

Changes in the State Space of a Markov Chain

In Organodynamics, we need to account for the case when the state space in a dynamical system itself changes. That is, at one step of the process we have a particular set of states (organizations), and at the next step we have a different set of organizations (even if the two sets overlap).

In a Markov chain, this occurs whenever the transition matrix changes from one step to the next. Whenever this occurs in a Markov chain, we say that the chain is nonhomogeneous.

This can occur in organic systems in many situations. Of particular interest is the distinction between open and closed dynamical systems. In many cases it is reasonable to equate a change in probabilities with a change in population, and vice versa. This is not always the case, but it in many cases it is a reasonable model.

For example, our canonical example of a set of atoms that can arrange themselves into many organizations of molecules makes this assumption.

This is merely examples of how the state space of a Markov chain in Organodynamics can change from one step to the next in the chain. And, such a change to the state space will necessarily affect the transition matrix.

Technically, a stochastic process whose state space changes from step to step is not a Markov chain. It is not even a nonhomogeneous Markov chain. This is because Markov chain theory deals with the case that the state space stays the same from step to step – even if the conditional probability distributions are allowed to change. This means that the transition matrix for all of the steps in the Markov chain will have the same number of rows and columns; and the assignment of states to rows and columns stays the same – even if the cells of the matrix (representing the probabilities) change.

However, for modeling organic processes, this restriction will not do. We must be able to allow changes in state space, because this models what happens in an open system. And we are modeling organic systems as one that can be closed for a while, but then occasionally open to allow atoms to enter and leave.

That is, states (system organization) change constantly in living systems as time proceeds. Consequently, we shall extend Markov chain theory to provide for changes in the state space whenever necessary. We shall refer to this aspect of our Markov chains “dynamic state change”.

Nevertheless, this notion of dynamic state change fits in well with our notion of *piecewise-homogeneous Markov chain* as we have discussed. We accomplish this by allowing the partitioning points of our Markov chains to not only include a change to the cells of a transition matrix, but to also include a change of assignment of the columns to represent any changes in the state space at the next step.

Of course, this means that steps that involve a change in state space may have non-square (rectangular) transition matrices. Of course, such steps will be points of nonhomogeneity within the “chain”. And, of course, any notions of “stationary distribution” will not carry across such a break in the chain.

Thus, we shall extend the notion of *piecewise-homogeneous Markov chain* to include the possibility that our transition matrix may not only alter the contents of its cells, but also the semantics and number of its rows and columns to represent a change in state space.

This means that segment boundaries in our piecewise-homogeneous Markov chains are now demarcated by steps whose transition matrix has changed from that of the previous step in a number of ways: either its contents have changed, or its column assignment has changed, or both.

Moreover, our *piecewise-homogeneous Markov chain* can still exhibit the Markov property (“memorylessness”). In fact, this property is a principle reason for extending the notion of Markov chain to admit nonhomogeneity – and to contrive our notion of piecewise-homogeneity.

In this manner, our concerns regarding regularity can be generally confined to the segment level. Our chains can now be described as finite sequences of segments, each of which are homogeneous, and therefore exhibit *regularity* as we have defined it (homogeneous and admitting to at least one stationary distribution).

However, we have yet to address the concept of regularity across an entire sequence of concatenated *segments*, which we call an *edge*. We shall take up this issue in a later chapter where we shall need to address additional machinery from the discipline of Information Theory.

Stochastic Networks

Sometimes the behavior of a stochastic process is cyclic or periodic. That is, the probability models of particular sequences of time steps within the process begin to repeat after a while.

In addition, it may also happen that the outcome of a time step selects among multiple alternative segments. This effectively provides a conditional branch in the process, and therefore in the Organodynamic web.

Thus, Organodynamic webs exhibit both looping and conditional branching.

This looping and conditional branching provide additional regularity opportunities, and is therefore lies within the province of *autocogeneration*.

Space and time do not permit the investigation of these dynamics within this text. However, the discipline of *stochastic networks* provides considerable literature in these regards. Admittedly, much of the literature on this subject concerns computer network applications.

However, hopefully more theoretical research can be identified that can shed the considerable light on this subject that is needed by the present development of the Organodynamics framework in its study of the dynamics of autocogeneration.

In any event, further research into the area of stochastic networks is suggested in order to place the theory around autocogeneration on a stronger theoretical and practical foundation.

Incremental Example Model Development

<>

Progress Check

In Part II, Organodynamics, we are gradually and incrementally developing a mathematical framework for modeling living and lifelike systems according to the seven organizing principles of OCS. Once completed, this modeling framework will achieve the form of an abstract but highly flexible mathematical structure that we are calling an *Organodynamic web*.

We are using the example of a hobbyist's toolkit for building electronics as an analogy to Organodynamics. Both are systems for "building something", and in both cases the resulting "thing to be built" has a network structure and has been given a general name.

In the case of the electronics circuit toolkit, the general name of the "thing to be built" is "circuit". The circuits built by this kit are highly customizable to specific applications. In the case of Organodynamics, the general name of the "thing to be built" is "Organodynamic web". The models built by this mathematical framework are also highly customizable to specific applications.

Because of the complexity of organic systems, any model that represents them needs to be built in cumulative and increasingly accurate stages. Each stage is generally referred to as an "approximation". Organodynamics as a model builder's toolkit provides for six of these approximations. These approximation levels are well defined and occur at precise places in the methodology set forth by Organodynamics.

In this chapter we have added a number of elements to our modeling toolkit – the Organodynamics framework. We have added some specificity to the inner working of one of the elements introduced earlier: the *segment*. We now know a lot more about the fact that a *segment* is comprised of a finite step homogeneous Markov chain.

We are now better able to characterize a *segment* by how much *uncertainty* it "holds". The degree of uncertainty inherent in an organic system is a systemic property that is significant to OCS and to Organodynamics. So it is important that we now know how to measure uncertainty for one of the foundational elements of an Organodynamic web: the segment.

In subsequent chapters we shall continue this march toward being able to measure the degrees of uncertainty for all aspects of Organodynamic webs, including the entire structure. Of course, we shall continue to extend Shannon's entropy function to be able to measure these aspects. And we shall continue to look to information theory to provide the mathematics behind this exercise.

We also alluded to, but did little to develop, a foundation for, the mathematics to support looping and feed back mechanism for the Organodynamics web. However, such developments are essential in order to support the regularity

aspects of autocogeneration to the extent that we need using the mechanisms of stochastic processes, entropy and other information theory mathematics.

This is regarded as a serious deficiency, and it is suggest that the field of Stochastic Networks be investigated for supporting theory.

Nevertheless, we shall assume at this time that such a foundation has been laid at this point, and that we have enough intellectual apparatus now developed to be able to define the next and third *approximation level* of our Organodynamic modeling methodology – the *Organodynamic graph*, which we shall take up in the next chapter.

Organodynamic Graphs

In our quest to build the Organodynamics framework into a toolkit of elements that can be assembled in myriad ways to produce a network structure that is capable of modeling comprehensively “lifelike” organic systems, we have so far developed two element types: the *segment* and the *edge*.

One could argue that this is relatively meek output, and that we have quite a ways to go before we have a full-blown Organodynamic web structure. This is admittedly true.

But one must also take into account that each of these two “graphical network elements” (the *segment* and the *edge*) is merely a graphical artifact that wraps an internal construction of rich mathematical material – specifically, probability spaces that comprise stochastic processes of quite rich varieties.

It is the mathematics that is internal to segments and edges that provides the “real meat” of Organodynamics. The “network elements” play the role of integrating these underlying mathematics into a higher-level construction that, as we shall see, can exhibit network properties in its own right.

Of course, using just *segments* and *edges* by themselves, one can only make very trivial “networks”. We at least need to add some kind of *node* element to finally enable the construction of respectable networks structures.

In this chapter, we shall finally develop various types of *node* elements to our evolving toolkit. Of course, like *segments* and *edges*, *nodes* will be a high-level “wrappers” that cover some internal mathematics that provides the more exciting characteristics that we can attribute to these *nodes* and that enable the to be models of organic phenomena.

The Issue Addressed

The two previous chapters have introduced the fifth OCS organizing principle, *autocogeneration*, and defined how the notion of Markov chains is used to implement this principle in the Organodynamic framework.

It was also explained that the principle of autocogeneration can be described in terms of two interrelated aspects: *regularity* and *co-generation*. The Markov chain theory presented above addressed the first of these – regularity. However, we have yet to adequately address the second aspect – co-generation – which we shall do in the present chapter.

The idea of co-generation is that an organic system creates itself. This is possible because of the fact that, as a system, it has a population of components (as well as some relationships among them). The property that makes co-generation work is that in organic systems, these components can create each other. They can also create relationships with each other – and therefore *organizations*, which is the other major part of a system besides its

population. Therefore, the components can create the system by creating each other and relationships between each other. This is *co-generation*.

But, for OCS, this is just the beginning. OCS intends to include co-creation under the umbrella of *co-generation*, but to go beyond it. The most general notion intended in the term *co-generation* can be termed “co-management”, and includes creation, organization and maintenance. So, by *co-generation* is really meant *co-management*.

But *co-management* requires a way in which these components can “work with each other”. We need to add some mathematical mechanisms to our Organodynamic framework that permits the components of our system to “work with each other”. The question then becomes, what can those mechanisms look like.

For some inspiration to help us in answering that question, let's look at our canonical example from biochemistry. (Actually, this example starts out as plain ol' physical chemistry.) Recall that this canonical system is initially a close collection of atoms that are capable of “working together” to form arrangements that consist of some molecules, plus possibly some free atoms that did not “find any partners” to bond with in some of these “arrangements”.

The question that we ask at this point is “Exactly how did these atoms ‘work together’ to bond into molecules?”

Rather than giving the answer in “chem-speak”, we shall make a more general articulation that lends itself to being mathematicized:

This answer is: “These atoms combined together to form molecules, which then proceeded to further combine together and then to split apart. They then continue combining and splitting without surcease.”

How can this “combining together and splitting apart” be represented in mathematics? Moreover, can we use any of the mathematical constructs that we have already developed for the Organodynamic framework to represent these dynamics? These are the questions answered by this chapter.

Before we proceed, however, let's point out that “combining together and splitting apart” is a pretty good description of what happens in networks. In fact, it is one of the principle missing elements in our toolkit so far. We currently have *segments* and *edges*, but we have no way for them to “come together” or to “split apart”.

Thus, the very same mechanism that can enable our components to “work together” can also double as our mechanism for “coming together and splitting apart”.

Graphically, these elements will be referred to as *nodes*. Together, by using all three of *segments*, *edges* and *nodes*, we have the makings of primitive networks. We shall refer to these combined constructs as *Organodynamic graphs* – which is the title of this chapter.

Like *segments* and *edges*, we shall invent *nodes* so that they are a high-level wrapper around a deeper mathematical implementation. And it is this mathematical implementation that provides most of the systemic characteristics that will make Organodynamic webs the powerful modeling apparatus that it is emerging to be.

Organodynamics is always interested in how the dynamics of *uncertainty* – the sixth OCS organizing principle – come into play anytime we add new mechanisms to our modeling framework. So we must ask how the addition of *node* structures affects the entropy of these network structures.

We have already seen how the *regularity* aspect of autocogeneration via the use of Markov chains can have a damping or limiting effect on uncertainty in segments, and even edges, and can “prevent” them from “wandering off into chaos”.

We must naturally ask the same question regarding *nodes*. Intuitively a node “brings things together” coming into it – thus reducing choice, and “splits thing apart” moving away from it, thus increasing choice. These are examples of both increasing the amount of choice and of reducing it. Thus, intuitively, it would appear as though the introduction of nodes to apparatus should be able to both increase and decrease choice – and according to [Shannon 1963] – therefore entropy.

Much of this chapter is dedicated to presenting this mathematics.

OCS Organizing Principle Supported

Organizing Principle: 5, autocogeneration; 6, uncertainty.

Biological Example of the Issue

When cells divide, they split into two new cells. When atoms or molecules encounter and, perchance, subsequently bond, they join together. Organodynamics considers both of these examples of “entities working together” and requires a mathematical mechanism to represent them within the Organodynamic framework.

Specific Challenges

So far, the framework has utilized a number of mathematical disciplines, including probability theory, its branches information theory and theory of stochastic processes as well as basic set theory and some abstract algebra.

Whenever we add any new mathematical disciplines to our framework, we must integrate them in a way that results in real world of living systems can be modeled.

In this chapter, we must look to see if we already have some mechanism within the mathematical disciplines that we have already called forth that will enable

us to model the “working together” of existing model elements and that are amenable to represented nodes in a network. Fortunately, we do – as we shall see below.

Otherwise, we would have to have imported some new mathematical discipline that does, and integrate its elements into the framework.

The Organodynamic Approach to Modeling the Issue

In an earlier chapter on the *reorganizational* OCS organizing principle, we identified a number of *transformations* from one system organization to another. Formally, we called these *system transforms*. We categorized and named several of these. Each described a specific category of ways in which one type of system organization can be changed, or transformed, into another system organization.

Some of these named transformation categories were interesting because of the way that they worked with multiple system organizations at once (at the same time step), and described how they could collectively be combined into a single system organization at the next time steps. In abstract algebra, we call this type of transform an *operation*.

Certainly, this description fits the idea of “multiple entities working together to produce something else”. Perhaps an *operation* of two system organizations can be a candidate for the *node* that we are looking for.

But, “bringing things together” is only half of what nodes do. They also “split things apart”. Thus, for a *system transform* to qualify as a true *node*, it must also be able to split a single “thing” (system organization) into multiple parts. Moreover, nodes often do both things at once: “bring things together into one thing, and then split that one thing apart”.

We shall investigate the possibility of system transforms doing all of this in the present chapter.

However, this will not be quite enough – because in what we have developed so far in the previous chapter, we are no longer dealing with *system processes*, but with *stochastic processes*. And stochastic processes do not have simple *system organizations* as their time step. Rather, they have *probability spaces* of system organizations as their time steps.

Therefore, our idea an “operation on some system organizations” (e.g. *system transform*) is not “fancy enough”. We will have to re-invent the idea of *system transform* so that it can operate on the more complicated “*probability spaces* of system organizations” before we can use them as the *nodes* in our graph.

At that point we shall have two essential elements that a toolkit needs to create network structures: *edges* and *nodes*. In addition, we also have *segments*, which are a simpler form of *edge* of which edges are composed. (That is, their internal mathematics is simpler.)

And we can use these three constructs to construct a network pattern that we shall call a *graph* - or more specifically, an *Organodynamic graph*.

There is one essential network element that we shall not yet provide, and will hold back until the next chapter. This is a *loopback mechanism*.

System Transforms as Nodes that connected System Processes

In an earlier chapter of this text on the subject of the fourth OCS organizing principle (reorganization), we introduce the notion of a *system process* as a sequence of time steps, where each time step is modeled by the “state of a system”. The idea is that we can represent system dynamics as a sequence of *system states*.

Of course, we have decided that in Organodynamics, “system state” is represented by a particular arrangement, or *organization*, that a specific system can obtain at one time. In fact, we have a well-defined, set-theoretic articulation of how a *system organization* is constituted. So, a *system process* is defined in Organodynamics as a sequence of *system organizations*.

In the same chapter, we also discussed the idea of *system transforms*. A system transform is way to describe how to get from one time step in a system process to the next. The basic idea is that a system transform is a mathematical function that maps the *system organization* at one time step within a system process to a second *system organization* which constitutes the next time step.

However, we also saw that this idea of *system transform* can be made to be fancier than this. For example, we could define it so that it could take two or more *system organizations* that exist during the same time step, but within two or more distinct concurrent system processes. And – that system transform could also produce a resulting *system organization* that happened to be in multiple parts.

Thus, this fancier version of our *system transform* could provide many-to-one or many-to-many situations across time steps. In fact, if you think about it, it is able to also provide one-to-many and even one-to-one situations as special cases. We even provided example *system transform* categories, each of which could do some of these. For example, we had these categories of system transforms among others: Unite, Divide, Integrate, and Disintegrate.

One more thing: as discussed above, these system transforms are defined so as to “work on” (have inputs that are) *system organizations* at a single point in time.

But we could extend this idea so that a system transform can “work on” (have inputs that are) an entire *segment* – or even *edge* – instead. A very simple way that this could work is if we defined a *system transform* to “work on an edge” simply by “working on the last time step of the edge”. In fact, we could also define a *system transform* to “work on a set of edges” in the same manner – if all of the edges involved end at the same time step. (Of course, this same definition works for segments as well.)

And, we can extend the idea of a system transform operating on a single edge by operating on the system organization of its final time step and producing multiple system organizations as a result, all occurring at the next time step. And, if we defined these multiple system organizations as the first time step in each of several new edges. Having done, this, then we have define the notion of system transform operating on a single edge to produce multiple edges.

In general, then, we have extended the notion of a *system transform* that can operate on multiple *edges* (all of which terminate at the same time step) and produces multiple new *edges* (all of which start at the *next* time step).

Thus, we have just provided an algebraic definition of *system transform* that enables it to connect multiple system processes from the “past” to multiple new system processes that begin in the present and the extend into the future.

This situation is nothing short of a simple network of system processes that are connected via a node.

Contemplating Nodes to connect Stochastic Processes

We have just described the use of system transforms (an algebraic idea) as a *node* element in a “graph” to connect together multiple sequential *system processes* from the past with multiple other sequential *system processes* in the near future.

If we could do the same thing with *stochastic processes*, then we would have achieved the goals of this chapter.

In fact, we can be more precise in this statement. We have already been more specific about the stochastic processes involved. We have named them *segments* and *edges*. So, if we can find a way to define the notion of *node* for segments and edges – just as we did above – then we will have achieved our goal for the chapter. So we must find a way to do that.

For *system processes*, we defined *nodes* as *system transforms* that mapped (possibly) multiple *system processes* (all of which terminated at the same time step) to a set of *other system processes* (all if which initiated at the *next* time step).

So perhaps we can take the same approach with our *segments* and *edges*. Unfortunately, though, we do not have the concept of a *transform* for stochastic processes like we have for (deterministic) system processes.

But, maybe we can define it! Perhaps we can begin with the definition that we already have of a *system transform* and extend it so that it is a “stochastic transform”.

Let’s look briefly at what that would take. At its simplest (before we added the many-to-many capability and extended our system transform to work on whole processes), a *system transform* operates on an input *system organization* to produce an output that is another system organization.

How would we have to extend this idea so that it works for stochastic processes?

Now, a *system organization* is the *state* of (a single step of) a system process. So, we must determine what the *state* of a stochastic process is. And *that* is what our new “stochastic transform” will both operate on and produce.

So, what how do represent the *state* of (a single step of) a stochastic process? The answer is: as a *probability distribution* – whose state space is a whole set of system organizations.

Therefore, in order to define the idea of a “stochastic transform”, we must define a way to map a probability distribution at one time step to another probability distribution at the next time step. This imitates a *system transform* that maps a single *system organization* at one time step to another *system organization* at the next time step.)

And, we are going to also imitate, as much as it makes sense, the manner in which system transforms were defined when we define these new “stochastic transforms”. That is, we shall have the same categories of transforms, and their definitions will be extensions of the corresponding system transforms.

And, also, of reach category, we shall only concern ourselves with the definitions of the *sample spaces* involved with each category of transform - not the probabilities of the distributions involved. We shall save that determination for further research where specific transforms of each type are identified and defined.

One more thing, we shall not call these “stochastic transforms”, but rather *organodynamic transforms*.

And, it is these new *organodynamic transforms* – which are obviously algebraic mathematical entities – which we shall establish as the *nodes* of the modeling toolbox that we are gradually developing across these chapters.

Thus, when finish this chapter, we shall have three of the basic elements of our Organodynamic web toolkit defined: *segments*, *edges* and *nodes*.

To review, the *segments* are finite state, finite step homogeneous Markov chains; the *edges* are contiguous sequences of *segments*; and the *nodes* will be *organodynamic transforms* whose inputs and outputs are *edges*.

General Definitions of Organodynamic Transforms

To move closer to a formalization of the ideas just presented, we can state the following. The idea is to define a “simple stochastic transform” on a single time step as a *stochastic transform*; and then to define the extended idea of mapping entire edges as an *organodynamic transform* – because these are going to become the *nodes* of Organodynamic webs.

In the process, we shall make some minor adjustments to the terminology presented above.

In addition, we shall define *really* simple versions of both of these that operate on only one input source, and the more complex version that operates on multiple input sources and produces multiple output sources.

Thus, here are some definitions that we shall move forward with:

Definition: *simplex stochastic transform*; Maps a *probability space* representing a single time step of a stochastic process to another *probability space* representing the next time step.

Note: Depending upon which stage of the uncertainty model gradient is being represented, the *probability space* type will vary. It may be either a Markov transition matrix, a probability distribution of system organizations or a single system organization.

Definition: *stochastic transform*; Maps a collection of *probability spaces* representing a single time step of multiple concurrent stochastic processes to another collection of *probability spaces* representing the next time step.

Definition: *simplex organodynamic transform*; Maps an *edge* to another *edge* whose initial time step follows the terminal time step of the input edge.

Definition: *organodynamic transform*; Maps a collection of *edges* whose terminal time steps are concurrent to another collection of *edges* whose initial time step are concurrent and which immediately follows the terminal time step of the input edges.

The fourth definition, *organodynamic transform*, represents the node of an organodynamic graph.

Join and Split Transforms on Piecewise-Homogeneous Markov Chains

Biological processes are often described as complex webs of interactions. We want to model this type of dynamics in Organodynamics. In fact, it would be desirable to take the ideas that we have developed so far – all of seven OCS organizing principles as modeled in Organodynamics - and somehow weave them into such a structure that would inculcate these network dynamics. Such a structure would then be the “grand construct” of Organodynamics – a construct that could model mathematically any organic system.

We shall introduce such a dynamical structure in this chapter; and spend the remainder of this text enhancing it.

The present subsection presents the approach we shall take in order to accomplish this.

The essential idea of our approach is that any web, or network, can be viewed as a combination of single processes that occasionally split into multiple processes; and of multiple processes that occasionally come together, or join, into a single process. Taken together, all of the process *splits* and *joins* amount to the network.

Therefore, if we can find a mechanism to represent “process splits” and a mechanism to represent “process joins”, then this will form the basis of what we need. Below we shall show that we have already developed the basis for these mechanisms; and we shall extend them further to give us what we need.

But first, lets further motivate the need for such a networking capability by looking at some more biology.

Splits and Joins in Biological Processes

Biological processes often combine to form a single process. And other biological processes often split apart, to form multiple processes. At the molecular level, for example, simple molecules combine their components to make macromolecules, and complex molecules split to form multiple molecules from their components. In some cases, molecules even join in such a way that the result of the chemical reaction is multiple molecules at least some of which are distinct from the “input” molecules.

Of course, the same kinds of combinations of organic processes also occur at higher (and lower) levels of organization within organic systems. For example, in cell division of prokaryotes, we have splits but no joins. However, in sexual union and reproduction there are both splits and joins.

In evolution of the species, there are splits and no joins. While in ecological systems, it is possible for two ecologies to join (merge). This can occur if the water levels of two separated ponds increase enough to become one pond. Ecological splits also happen. For example, an earthquake separates an ecological system into two.

An Analysis of Splits and Joins in Biological Processes

Lets first look at a split operation in biology. Consider the case of a prokaryotic cell dividing. When we look at this process at the single-cell stage, we have more than a state – we have a process. That is we have a cell that is changing state from moment to moment. In Organodynamics, we model this using a finite-step stochastic process. Specifically, we model it with a finite step, finite state, piecewise homogeneous Markov chain. In any event, the point is that it is not just a “state”. Rather it is a process consisting of multiples steps, each of which has a state.

And this is the entity that we want to model: a process, not just a simple state.

However, just before it “splits” (divides), it has a final state (of its sequence of states) of its life before the split. Lets say that such state occurs at time $t-1$. At time t , however, the “split” has occurred.

Therefore, in order to model the “split of the process” it is sufficient to model a change that occurs between the state of the process at time $t-1$ and the state of the process at time t . And, at time t , there will be two (or more) concurrently exiting states – because this is the nature of a “split”.

So, we have produced two new *states*, but we still have not produced two new *processes*. But we can accomplish that by initiating two new processes and defining these two new states to be the states of their first steps (steps number 1), respectively.

The only remaining problem is that we now have three separate processes, rather than a single process that has a “split” in it. This is easily solved by renumbering the process steps of the two “output processes” so that they start at time t rather than at time “1”.

The point is that we can model the split of a process into multiple new concurrent processes by modeling a split of one of its states (at time $t-1$) into two or more concurrent new states at time t .

And, we can do the same with “joins”. Of course, “joins” have multiple concurrent processes coming together to produce a single process “going out”. But the same approach applies.

Approach to Modeling Process Splits and Joins in Organodynamics

The mechanism that we have constructed thus far to represent organic processes in Organodynamics is the piecewise-homogeneous Markov chain. Therefore, what we need to be able to do is provide some mechanism that enables us to introduce splits and joins into this mechanism.

It happens that we have already developed a mechanism in a previous chapter on system processes that implements splits and joins. This mechanism was the *system transform*. Some of these system transforms, such as Unite and Integrate, were *join* transforms. Other, like Divide and Disintegrate, were *split* transforms.

What we would like to do is to somehow embellish these *system transforms* that were defined on system processes into a new kind of transform that we can define for piecewise-homogeneous Markov chain. We shall do that in this chapter – and we shall name these new embellished transforms *organodynamic transforms*.

Thus, an *organodynamic split transforms* will enable a piecewise-homogeneous Markov chain to split into multiple segments. And an *organodynamic join transforms* will enable multiple piecewise-homogeneous Markov chains to join into a single segment.

Together, then, these split and join organodynamic transforms will enable us to expand our piecewise-homogeneous Markov chains into piecewise-homogeneous Markov networks. As we shall see below, this will give us the dynamical network structure that we want.

Types of Splits and Joins in Organodynamics

Our approach, therefore, is to define a set of mathematical transform categories that can be used as “split” and “join” operations in order to construct a network of “edges” – where these “edges” are actually piecewise-homogeneous Markov chains, each “edge” consisting of a finite sequence of time steps whose state is characterized by a Markov transition matrix.

Of course, a “network” or “web” is essentially a geometric or topological concept. But, “transforms” and “operations” can be thought of as “algebraic”. So in a sense, we are constructing a “geometric” structure from “algebraic” materials. In fact, this structure is already even more interesting because we have stochastic processes in the mix as well.

This subsection distinguishes between two types of splits and joins that are inherent in what we have already discussed. These types are referred to as *concurrent* and *conditional*, and will be explained below. It further points out that each can reside within the other – thus enriching the complexity capabilities of this structure as a device for modeling organic systems.

This complexity is good, because organic systems are very complex. Thus, if we can provide a complex modeling capability to model complex systems in a tractable manner, we shall have a very useful tool. Of course, real biological systems are also so complex as to be essentially unfathomable. Thus, our modeling system needs to be able to be tractable at the same time that it also borders on the unfathomable. That way, modelers will be able to trade of tractability and complexity to get the balance the want.

Concurrent Joins and Splits

The splits and joins that are the result of applying system transforms to Markov chains discussed above all involve multiple, concurrently existing, systems and Markov processes. We have talked about organodynamic transforms that join multiple concurrent processes into a single process; and we have talked organodynamic transforms that split a single process into multiple processes.

We have before us quite a bit of work in order to define these organodynamic split and join transforms. In order to accomplish this we have a number of tasks:

1. We must deal with mapping between one Markov chain and multiples Markov chains.
2. Ultimately, we must work – not with Markov chains – but with transition matrices, which represent the state of a single step of a Markov chain. Thus, we must take the last step(s) of the “input” chain at time $t-1$, and map it to the “first” step of a new chain(s). That is:
 - For *join* organodynamic transforms, we must define how to take multiple transition matrices from step $t-1$ of multiple concurrent Markov chain segments and combine them into a single transition matrix for step t . In

addition, the manner of combination must make sense for the specific transform (Unite, Integrate, etc.) being defined.

- For *split* organodynamic transforms, we must define how to take a single transition matrix from step t-1 of a Markov chain segment and split it into a multiple transition matrices for step t. In addition, the manner of splitting must make sense for the specific transform (Divide, Disintegrate, etc.) being defined.

3. We must finally “connect” the “input” chain(s) of this transform with the new “output” chain(s) so as to create an Organodynamic node. In this case, the “new” step becomes step t of an Organodynamic network, or web. Therefore, we now have something more general than a piecewise-homogeneous Markov chain. We now have, instead, a piecewise-homogeneous Markov network! This network is, in fact, what we are called the Organodynamic web.

So in all cases, we must take one or more transition matrices from a step or steps at time t-1 and somehow map that to one or more transition matrices for a step or steps at time t. But, recall that a transition matrix has three parts of information: 1) a set of possible current states, 2) a set of possible next states, and 3) the conditional probabilities for all pairs of 2) given 1).

We shall see that the general rule for calculating these output matrices is to 1) determine the *current state sample space* of the output matrix (or matrices) by applying the appropriate split or join algorithm, and then 2) to let the specific transform of the category determine the remaining aspects of the output matrix (or matrices). Recall that the text only describes *categories* of transforms and does not define any specific transforms. These remaining aspects include the *next step sample space* and *the probability values* of the matrix (or matrices).

A point that we have already started making is that the organodynamic transform category definitions below (Unite, Divide, Integrate, Disintegrate, etc.) only perform direct transformation on the underlying *populations* of the state spaces involved in these transition matrices, and leave the determination of the state spaces themselves to the specific transforms.

Conditional Splits and Joins

In addition to the *concurrent* splits and joins just discussed (which are unconditional), our stochastic model already embodies the notion of conditional (or dependent) splits in the form of the state spaces, or sample spaces, that are represented by the rows and columns of the transition matrix of a step of a Markov chain.

These sample spaces represent a split because any one of them can be selected when the step is realized. But only one of them can be selected. That is, they represent alternatives, rather than *multiple manifestations*. This is an important difference between these and the concurrent splits, because they represent *choice* – a subject of Shannon’s concept of entropy. In fact, we know that entropy increases whenever there is choice. Therefore, *conditional splits*, or *alternatives*, increase entropy.

Organodynamic also provides the notion of *conditional joins*. These occur in two situations that are described below. Both of them have the property that they reduce the number of alternatives from one time step to the next in a stochastic process. Thus, they reduce entropy.

Implementing Splits and Joins as Organodynamic Transforms

Lets now look at specifically how we shall implement these various types of organodynamic transforms using the mathematical discipline that we have so far incorporated into the Organodynamic framework.

Concurrent Joins

The *concurrent joins* are implemented by taking the Markovian structures – specifically the transition matrix – from one *step* of each of several concurrent piecewise-homogeneous Markov input chains. Each of these steps occur at the same time $t-1$ in each input chain. The concurrent join transform then maps this collection to a new combined transition matrix, which is the transition matrix for step t of the new the resulting organodynamic node.

Recall that all of the rows of the resulting matrix represent a sample space of possible outcomes for time t . Specifically; the states of this sample space are all system organizations of the same underlying system population.

It is this underlying population that the transform category determines. All other aspects of this resulting transition matrix for time t are determined either by specific transforms of the category (not define in this text), or by the application that is being modeled. (For example, the probabilities within the body of the matrix may be empirically determined.)

All of these *join* transform categories therefore specify how to combine the population of their operands system (at time $t-1$) into a new population (at time t).

Concurrent Splits

In biological systems, we see single processes split into multiple concurrent process often. An example of this is simple prokaryotic cell division. We begin with one cell process and end up with two cell processes. Of course, we want to model this phenomenon in Organodynamics.

The idea of a *split-able system* is that the organization of its components forms “separable sub-organizations”. This means that the set of duples in this system organization can be partitioned into a collection of sub-organizations whose underlying systems also happen to partition the underlying initial system population.

This is what happens, for example, in mitosis when the chromosomes line up about the spindle on two sides of the cell. Two new and “separable” sub-organizations of chromosomes have been formed. And these two new sub-

organizations will show up in the two new cells after the division has been completed.

We can model this phenomenon with the notion of *split-able system*.

This will require a number of stages. First, we must note that, before it divides, a single cell is a process that includes a sequence of steps, each of which has a state. In Organodynamics, we represent the process by a piecewise-homogeneous Markov chain. Second, we notice that there is a final step within this chain prior to the split occurring. Lets say that this is at time $t-1$. In Organodynamics, that time step, like any other, is represented by a Markov transition matrix.

Thirdly, at the next time step at time t , the Markov chain splits into two states – each of which we want to represent by a distinct transition matrix. To both of these steps we shall assign time t . Fourthly, the two divided cells now proceed with each in its own process. But each of these processes continues on as piecewise-homogeneous segments, with successive time steps at times $t+1$, $t+2$, etc.

Together, step $t-1$, and all steps numbered t form a *split node* in our new organodynamic network that we are now calling an Organodynamic web.

Notice that when we moved from the first stage to the second, our single *state* morphed into two distinct states. What was implied, but not said explicitly, is that the *components* of the initial state were *partitioned* into two new entities. This means that a *preservation of components* was exhibited in this “becoming two distinct entities”.

However, this partitioning of components provides an additional opportunity for randomness. This time, the potential randomness involves the population rather than the organization of the initial population at time $t-1$. (The randomness involving organization will follow.) In fact, this is the first occasion in Organodynamics where we have considered randomness with respect to the population.

We must add the mathematics here to account for this randomness involved in this population split. As indicated, this splitting involves partitioning, which is very well modeled by the notion from combinatorics of *combinations*. That is, the sample space of possibilities of the splitting of the population at time $t-1$ into multiple populations at time t is the set of all combinations of N things taken R at a time, where N is the size of the initial population at time $t-1$ and R is the number of populations that it is split into at time t .

The assignment of probabilities to this sample space will not be defined for categories of transforms, but will be reserved for the specific transforms of each category. The reader will recall that we are not identifying specific categories of organodynamic transforms in this text; but leaving that task to further research.

Once the split has occurred, then multiple systems have been established, and their individual populations have been defined. It is at this point that the

organizations of each of these new systems are established. Again, this is also a stochastic phenomenon. But, involving the possible organizations of a population, it is the same probabilistic event that we have been dealing with all along.

Now, we know that in some types of “biological splitting” some preservation of organization *is* preserved. But in other types of biological splitting, preservation of organization is not preserved. In general, different amounts of organization are preserved across different kinds of biological splitting. But, the preservation of the componentry is very broadly observed in this splitting in biological systems.

Therefore, we want to define our general model of the category of organodynamic concurrent splitting to preserve the components, but to say nothing about the organization of the components across the split. We shall leave any rules about what happens to the organizations across the split transforms to the individual transforms themselves.

And, since this text only discusses the categories (Unite, Integrate, etc.), not any specific transforms (such as mitosis, meiosis, evolutionary splits, etc.), then we shall not have any definitions of splits in this text that directly manipulate organizations across these transforms.

Thus, each category of the split transform that is defined below will discuss how it will operate on the population of system components that underlies the state space of the transition matrix at time $t-1$ and transform (map) it to a *split-able system*.

Lets summarize what we have discussed in this subsection. These split transforms first involve a partitioning of the population of the input system into multiple populations of the output systems. And, as indicated, this is a probabilistic split whose sample space is enumerated as multi-cell combinations. Thus, eventually when specific instances of these transform categories are identified, probabilities must be assigned to these combinations. Further, the reorganization of these separate populations after the split has occurred is modeled via the further probabilistic event. All of this is represented within the mitotic process of cell division in biology.

Conditional splits; Alternatives

Conditional splits are *alternatives* and are represented in stochastic processes as the sample spaces of probability distributions. In a Markov transition matrix, these alternatives are represented by the *next step* probability space, and therefore by the column headings of the transition matrix.

Given that conditional splits are choices, alternatives, then they fit Shannon’s definition of entropy – that is, they pertain to choice - the more choice, according to Shannon, the more entropy.

Therefore, conditional splits represent an increase in entropy – at least whenever the probabilities stay (in some sense) the same. For example, if all of

the probabilities are equally likely and we increase the number of choices (sample space members), then the entropy will increase.

In any event, all other thing being constant, an increase in the size of a sample space portends an increase in entropy.

Conditional joins; Manifestations

Conditional joins are provided in two ways.

One way is by resolving from a conditional version of the model to an unconditional version of the model; or by resolving from an unconditional version of the model to a realized version of the model. This takes place within the uncertainty model gradient.

Moving from a conditional version of the model to an unconditional version of the model can be accomplished in two ways. One way is, for every homogenous segment of the Markov chain or web, calculate the stationary probability distribution for the web. This gives the unconditional version of the model. This resolution “collapses” the row space of the segment to a single state – which represents a single composite system organization for the underlying current-step state space.

Moving from a unconditional to a realized version of the model collapses the column space to a single outcome – the realized outcome. Both of these “collapses” represent a conditional join action.

A second way that a conditional join can manifest whenever a change in time step is also a change in probability space, and the size of the sample space is reduced from the current time step to the next time step. In such a case, the amount of choice is reduced; there is an opportunity for the entropy to be reduced.

Of course, a change in probability space can also result in a decrease in sample space size. In either case, it is possible for entropy to either increase or decrease as a function of both sample size as well as the probability distribution.

In any event, this change in probability space occurs 1) between segments of an edge, or 2) across an organodynamic transform.

The Necessary Nonhomogeneity of an Organodynamic Transform

In general, the transition matrix does not survive an organodynamic transform. This means that the state space (or sample space) of an organodynamic web generally changes when we pass through an organodynamic transform.

But having the state space change from one step to the next is the very definition of non-homogeneity.

The only exception to this is the special case of the Reform transform where no “reformation” actually occurs. That is, both the population and the sample space of organizations of that population remain the same after the transform. A good name for this special instance of the Reform category would be the Identity transform.

In any event, the Identity transform is the only transform in which the state space stays the same, and the transition matrix stays the same. One could even say that whenever no transform is applied, the Identity transform is at work.

Consequently, except for the Identity transform, all organodynamic transforms introduce a point of non-homogeneity.

Patterns in Defining the Organodynamic Transforms

The definition of each of these transforms enjoys a pattern that we shall now discuss. There is quite a bit to say about this pattern, so we shall discuss it in three separate sections.

Organodynamic transform of all types fall into four categories of:

- Unary
- Joins
- Splits
- Both joins and splits.

A category’s inputs determine whether all transforms of that category are unary or joins. Categories with single inputs are unary, and categories with multiple inputs are joins.

However, whether a transform is a split is determined dynamically by whether its output transition matrix is split-able, as defined above.

Inputs and Outputs

Unary transforms have one input (domain entity), which is a transition matrix for time $t-1$ of its input process. The output (codomain entity) of a unary transform is a single transition matrix for the current step at time t .

The essential problem in defining the unary transforms is to determine the single output transition matrix for the current step at time t . An example of unary transforms is Reform.

Join transforms have two or more input arguments, all of which are transition matrices. These come from the final steps of a set of concurrent Markov chains, all of which occur at the same time $t-1$. The output of all join transforms is a single transition matrix for the current step at time t .

The essential problem in defining each of these join transforms is to determine the single output transition matrix for the current step at time t .

Split transforms have one input and two or more output arguments, all of which are transition matrices. The input transition matrix comes from the previous step at time $t-1$. Each split transform immediately maps the input transition matrices to a single, "split-able" output transition matrix.

However, in order for the transform to be "split-able", this single output matrix must conform to certain rules (as described above). If it does, then it can subsequently be mapped to multiple transition matrices - where each of these is the state of the first steps of a set of concurrent Markov chains at time t .

The split transform, is then formed by composing these two stages together – the mapping of the initial matrix at time $t-1$ to a "split-able" matrix, and then the subsequent mapping of this matrix to multiple matrices, followed by the creation of one Markov chain whose first steps are these output transition matrices. Finally, the input Markov chain and the output Markov chains are united via the node into a single Organodynamic node where the input matrix is at step $t-1$ and the outputs are all at steps t .

The essential problem in defining each of these split transforms is to determine the "split-able" transition matrix for the current step at time t . The "splitting" of this "split-able" matrix is then straightforward.

Some transforms may also be both joins and splits – if their outputs are separable.

Of course, if the transform is not a join operation, then it will have only one input transition matrix from the previous step. However, if it is a split-type transform, then the transition matrix for the current step will be a "split-able" or "separable" matrix, as defined above.

General Target

In any event, the problem to be solved in defining these transforms is to decide how to determine one or more transition matrices for the current step at time t . But any transition matrix can be described as having three major aspects. It is these three major aspects that must be defined for each organodynamic transition category type.

These three aspects are: 1) the input state space that is associated with the rows, 2) the output state space that is associated with the columns and 3) the probabilities of the matrix.

Specific Target

However, we have already said that aspects 2) and 3) shall be left to the specific transforms that make up these categories of transforms – and that those member transforms shall not be defined in this text. We shall only define the categories of transforms.

So, we only have item 1) to deal with: the input states of the current step Markov transition matrix.

Moreover, notice that the input *states* of the current transition matrix form a *set of system organizations* – all of which share the same underlying population! A great simplifier here is that the *transform categories* that this chapter deals with only address the behavior of this underlying system population. Any specifications regarding the system organizations in the set are imposed by any specific transforms that are members of these categories. The categories themselves do not impose any characteristics of these organizations; but rather leaves that to the specific transforms of those categories.

Therefore, these categories of transforms defined below will define only the underlying populations of the sample spaces the input states of the current step Markov transition matrix. This is best presented by example: which we do immediately using the Unite transform.

Unite as an Example Organodynamic Transform

Consider the Unite system transform. This transform takes two “input” systems and maps them to a single combined system. The algorithm for determining the single combined system works with the populations of the two input systems only; and it determines the population of the “output” system by use of a simple set-theoretic algorithm.

This simple, algorithm in the case of Unite is the *union* operation on sets. In other words, the population of the result of the Unite system transform is the union of the two underlying system populations of the input systems.

It is important to note that the organization of the output system is not defined by the Unite transform category definition – but is left to the specific application that is being modeled by our Organodynamic web! In other words, the organization of the output system can be any “legal” organization of the resulting population of the output system of this Unite transform.

In our definition of the Unite organodynamic transform, similar issues are resolved in similar ways. Let’s describe at a high level how we shall define this transform below.

The central problem we need to solve in defining the Unite organodynamic transform is “How do we define the shared population of the “current” state space for the transition matrix for the current step?” We already know that the definitions of the other two aspects are left to the application.

In order to solve this definition problem, we shall consult a real example from biology. We already said that an example of the system transform version of Unite is a simple chemical reaction that combines two molecules to produce a third molecule. Of course, there are many types of “combining”. But the kind of “combination” we defined for Unite involves:

- 1) Consider the two input molecules as systems, each of which have a population and an organization;
- 2) Take the two population parts and combining them with the set-theoretic *union* operation, ignoring the two organization parts;
- 3) Form the population of the resulting current state space of the output system as this *union*; and
- 4) Form the organization part of the current state space of the output system with any set of duples that is a “legal” organization of the population part of the output system. The precise collection in this state space is determined by the specific transform – not b the transform category.

For the organodynamic version of Unite, then, we shall form the set-theoretic union of the two underlying populations just described. This union will form the underlying population of all of the members of the current state space of the transition matrix that we are calculating for this step.

A Specific Example of Unite from Physical Chemistry

This was precisely the case with our molecules in the canonical molecular example. If two molecules react probabilistically to result in a single compound, we know that the component atoms (the populations) of both reactants will show up as a combined population. And by “combined population”, I mean set Union. We know this because of preservation of matter - at the atomic level in this case.

However, we cannot in general predict the arrangement of the atoms in the resulting single compound. The arrangement (organization) of the atoms in the output molecule is determined by the specific input molecules involved, and not by some general rule. However, the population of atoms in the output molecule is determined. Also, the arrangements (organizations) of the input molecules are generally not preserved.

Therefore, the organizations of the input molecules are generally ignored through the chemical reaction; and neither the populations nor the organizations of the inputs, nor of the population of the output determine the organization of the output molecule.

These facts about a simple chemical reaction involving two molecules are in accord with the way that we described our approach to defining the organodynamic transform version of Unite.

Organodynamic Transform Categories

In this section, we shall introduce a number of the organodynamic transform categories. Each of these organodynamic transform categories is defined in more detail in Appendix 2 at the end of this text. Please consult this appendix for more information on organodynamic transforms in general and for more specific definitions of each transform category.

The set of organodynamic transforms are central elements in the construction of Organodynamic webs – described above the grand scheme of Organodynamics. The primary role that they play within that scheme is generally that of the *nodes* of the network that is, in fact, an Organodynamic web – as we shall see later in this section.

Nevertheless, *organodynamic* transforms are best understood as stochastic embellishments of the *system* transform categories already defined. Therefore, we shall construct each of the defined *organodynamic transforms* by embellishing the original *system transform* categories that we identified in the above chapter entitled “System Dynamics as Reorganization” and defined further in Appendix 1.

In Appendix 2, we shall not take the time and space to “promote” all of the defined *system transforms* to *organodynamic transforms*. Rather, we shall develop a selected subset. The promotion of the remainder will be left as an exercise for further research.

Our strategy in doing this “promotion” will essentially be to convert the “state” of each of those system transform categories from a *system organization* to a Markov transition matrix. That is, whereas the *state* of a system transform is defined to be a *system organization*, the *state* of an organodynamic transform is a *Markov transition matrix*.

Of course, any such Markov transition matrix will have rows and columns that represent an entire *state space* of *possible system organizations* – any one of which will occur for the current step of the chain. Thus, we will not be leaving behind the notion of system organization as state. We shall just be dealing with entire state spaces of possibilities of them.

Each of these possible system organizations will be represented by a row of the transition matrix. Also, if the step in question is homogeneous, then the columns will represent the same set of possibilities.

Now, each of these system organizations in the sample space shares the same underlying system population. That is, the system organizations in the sample space are all different reorganizations of the same system population. It is these underlying populations that our system transform categories will be working with.

They will be taking these underlying populations, performing transformations on them, and yielding resulting system populations. It is these resulting transformed system populations that will then be used to develop a sample space of system organizations – possible organizations of that resulting population. This new set of possible system organizations will be the new sample space for the resulting transition matrix that we are calculating.

As we have said, the transform categories do not dictate what the new set of possible system organizations are. (That is left to specific transforms.) The transform categories only concern themselves with calculating their underlying shared population.

And, as discussed above, we have named these new transform categories *organodynamic transform categories*. And don't forget that they form the nodes of our Organodynamic webs.

We will approach these definitions by taking each of the system transform categories and embellish it into its corresponding organodynamic transform category. However, in the interest of space and time, we shall develop only a representative set and the organodynamic transform categories in this text.

The pattern of how to develop an organodynamic transform from its corresponding system transform is relatively straightforward – as described in the previous subsection. So, it is expected that the readers will be able to make the generalization and develop the remainder themselves.

Presently, we shall define six of the organodynamic transform categories for use in our Organodynamic webs, and leave the development of the remainder to the reader.

The six systems transform categories that we shall presently develop in this chapter are:

1. Reform
2. Identify
3. Catalyze
4. Unite
5. Divide
6. Integrate

Generally, these categories describe constraints on the system populations of the resulting output (codomain) systems. The definitions of what happens to the system organizations of the resulting output systems are left to the definitions of the individual transforms of those categories.

The reader is directed to Appendix 2 for the specific descriptions of the promotions of these six transforms to their probabilistic *organic transform* version.

Organodynamic Graphs

In this text, we are gradually building a “construction toolkit” for mathematical models of organic systems. We are doing this in six major steps, called approximations.

So far, we have developed two of these approximations: the *segment* and the *edge*. The present chapter has added another piece of equipment to our collection of elements: the *node*.

Thus, we now have three elements: the *segment*, the *edge* and the *node*.

Together there three form the third approximation – the Organodynamic graph.

But, each of these three network element types is actually comprised of some quite rich mathematical material. And, in order to understand how they apply to model living systems – and how they can be connected together to form “network structures” one must master the mathematics of their interiors. Hopefully we have enabled this mastery with the contents of these chapters.

Now, these three element types have been articulated at a very high level as “network elements”. Essentially, this articulation is an analogy, and the “real elements” are actually the mathematical constructs that we have been describing.

But, in actuality, an Organodynamic graph is both. It is the network analogy, and at the same time it is the mathematics that contrives it. The successful modeler will move back and forth between these two views.

There is one major capability as of yet missing from this Organodynamics graph – a capability that we shall need in order to have a full-fledged Organodynamic web. It is a mechanism to allow looping of edges. We shall add this in the next chapter.

Topology of an Organodynamic Graph

The *segment* is the first approximation of the object system defined in the Organodynamic methodology. The *edge* is the second approximation. The Organodynamic graph is the third. This section will discuss how to build an Organodynamic graph, given that one has already constructed a single edge from three segments.

An *edge* represents a single organic process. In general, however, organic systems involve several concurrent processes, operating together as a single process at a higher level of organization.

For example, we can understand a biological cell – an organic process – as consisting of thousands (or more) macromolecules, each of which is its own organic process. These thousands of macromolecules co-exist concurrently. In Organodynamics, we represent each of these as an *edge*; and together to operate across the same set of time steps as *concurrent edges*.

Moreover, each of these edges is piecewise homogeneous. This means that they each consist of one or more homogeneous sequences called *segments*.

This third approximation consists, then, simply of multiple second approximations that are joined by nodes, which can then split and produce multiple more edges. These edge and node structures can be combined in myriad ways where edges are interconnected by nodes and nodes by edges to produce general graphs.

The idea is quite simple. However, treating it as a distinct approximation level provides the opportunity for the modeler to identify and develop a considerable amount of the model that is to become an Organodynamic web. (The fourth

approximation will begin to join some of these edges together in nodes, thus initiating the beginning of a network structure.)

Of course, each of these concurrent edges will progress through all three of the *uncertainty gradient* models, as before.

Simple Non-Biological System Examples of Organodynamic Processes

We shall continue with the use of the board game Scrabble® to provide simple non-biological examples of the concepts being presented.

Of course, Scrabble® is not and cannot be an organic process, and certainly not an organodynamic process because it is not *autocogenerative* for the simple fact that it cannot create, organize or generally manage itself. The players do all of that. And they are “outside agents” who are external to the game itself.

Nevertheless, both the network and the stochastic aspects of this game enable it to provide an illustrative example of an Organodynamic graph.

Scrabble®

The game begins when each player selects seven tiles from the collection of 100 tiles that have been arranged face down on a surface. As we indicated in the section on Scrabble® in the *Reorganizational* chapter above, the *Divide* transform models the selection of these initial tiles. This Divide transform transitions this organodynamic graph from step 0 to step 1.

Of course, we are now using the stochastic version of these transforms – the so-called *organodynamic transforms* wherein the inputs and outputs of each transform are probability distributions whose sample points are system organizations.

When we last used the Divide system transform, we applied it once to select all of the (three) player’s seven tiles at once. However, this time we shall use a slightly different treatment and apply it once for each player to select her seven tiles.

The *system* version of the Divide transform uses a single system as its input (the system of all facedown tiles) and produces multiple (in our case, two) output systems (one being the seven tiles and the other being the remaining facedown tiles). In the *system* version of Divide, the organization of the input system was taken to be the empty set.

However, in the *organodynamic* version, we shall take the sample space of the organization of this system to be the set of all possible arrangements (“linear organizations”) of these tiles. The use of permutations would be a better fit here than ordered pairs – in which case we would define the sample space to be the set of all permutations of the tiles.

Of course, the organization for an *organodynamic process* is a probability distribution. Thus, the organization of the input system for the Divide

organodynamic transform is the appropriate probability distribution whose sample space is the set of all possible arrangements of all of the facedown tiles. And this is the state of step zero of this organodynamic graph.

As outputs, the organodynamic version of the Divide transform produces two organodynamic systems. The first represents the seven tiles selected. The populations of these systems represent all possible combinations of 100 tiles taken seven at a time. For each of these populations, there is one organization that is the empty set – because these systems of seven tiles are initially unorganized.

The second output system represents the 93 remaining facedown tiles. However, all possible arrangements of these 93 tiles are possible. There are 93! of these.

This transform is repeated twice more for each of the two other players, which takes us through time steps 1 and two of this process.

The next steps involve each player rearranging her seven tiles via the Reform organodynamic transform. These transforms all map a system at time step “t” to another system at time step “t+1” whose population is remains invariant. However, the system organizations change with some probability. Therefore, the sample space of each system is a probability distribution whose sample space represents all possible permutations of the seven tiles in the player’s tray.

The third type of transform is the *Precipitate* organodynamic transform. This transform is used to select some number of tiles from a player’s tray and use them to make a word play on the game board. In a similar manner, the stochastic nature of this action is modeled by the use of probability distributions on sample spaces involving system organizations.

As can be seen, even the analysis and modeling of a relatively simple board game can be bewilderingly complex. Organodynamics provides as framework that is capable of modeling this extent of complexity, and hopefully even that of living and lifelike systems. Obviously, the modeling process itself can be extremely complex. Even so, the Organodynamics framework provides an approach to understanding the dynamics of these complex systems.

Incremental Example Model Development

<>

Progress Check

In Part II, Organodynamics, we are gradually and incrementally developing a mathematical framework for modeling living and lifelike systems according to the seven organizing principles of OCS. Once completed, this modeling framework will achieve the form of an abstract but highly flexible mathematical structure that we are calling an *Organodynamic web*.

We are using the example of a hobbyist's toolkit for building electronics as an analogy to Organodynamics. Both are systems for "building something", and in both cases the resulting "thing to be built" has a network structure and has been given a general name.

In the case of the electronics circuit toolkit, the general name of the "thing to be built" is "circuit". The circuits built by this kit are highly customizable to specific applications. In the case of Organodynamics, the general name of the "thing to be built" is "Organodynamic web". The models built by this mathematical framework are also highly customizable to specific applications.

Because of the complexity of organic systems, any model that represents them needs to be built in cumulative and increasingly accurate stages. Each stage is generally referred to as an "approximation". Organodynamics as a model builder's toolkit provides for six of these approximations. These approximation levels are well defined and occur at precise places in the methodology set forth by Organodynamics.

In this chapter, we developed the third of these approximations: the Organodynamic graph.

This approximation consists of three toolkit element types: the segment, the edge and the node. We already had the segment and the edge coming into this chapter, where we added the node.

The Organodynamic web is almost – but not quite – a full-fledged Organodynamic web. Only one more capability is needed – a loopback mechanism – which we shall add in the next chapter.

Simplex Organodynamic Webs

In this chapter we present the first rendition of the much-promised *Organodynamic web* – the *grand scheme* of our modeling framework. The initial basic version presented here is called the *Simplex Organodynamic web*.

The Issue Addressed

We have advertised this grand structure, the Organodynamic web, as the analogical equivalent to the concept of a “circuit” in our electronics hobbyist’s toolkit analogy.

Whereas any “comprehensive thing” that one can build using our electronics toolkit can be reasonably described as a “circuit”; within the world of complex systems modeling, any “near-comprehensive” organic model can be built can be reasonably described as a *simplex Organodynamic web*.

We say “near-comprehensive” because this simplex version intentionally omits support for the *nested* OCS organizing principle. Support for nested generally adds significant complexity to any organic model. It is more manageable to develop a near-comprehensive model first, and subsequently to incrementally enhance it with elements that provide system nestedness. Thus, we shall add the nested capabilities in the next version – the *composite Organodynamic web* - introduced in the following chapter.

So, the simplex Organodynamic web is the first milestone in our journey of incrementally building our modeling construction kit where we can finally say that it is representative of a *near-comprehensive* organic system.

This comprehensiveness is the reason that the simplex Organodynamic web structure is anointed as the *fourth approximation* in the set of six approximation levels that constitute our model building methodology.

But what do we mean by “comprehensive”. Our use of this term means that we have structure that can exhibit all seven of the OCS organizing principles at once. This is the importance of an Organodynamic web. As indicated, the simplex Organodynamic web supports six of these seven principles.

Any comprehensive construction built using the electronics hobbyist’s toolkit is a circuit and contains all of the construction elements of a circuit – including wires, logic gates, etc. Just so, any comprehensive model built using the Organodynamics framework contains all of the element types including segments, edges and nodes – plus the additional elements added in this chapter.

In addition, there is a specific type of *regular behavior* that has not been so far addressed that we can observe in biological systems that we want to provide a mechanism for at this stage of our framework development. This is what we shall call *routine behavior*. By this mean the repetition of the same actions, but

with “minor” variation permitted within each repetition. We shall work out how to implement this idea and provide the machinery for *routine behavior* as well.

By adding this single new mechanism for *routine* behavior to our framework, we will have all of the elements needed to comprehensively model living and lifelike systems.

Of course, you can build “less than comprehensive things” with the electronics hobbyist’s toolkit as well – just as you can with the Organodynamics framework. For example, you can string several copper wires together using relays, but this is not a complete circuit. In Organodynamics, this would be equivalent to developing a few *segments* and connecting them together into an *edge*. But such a model would not exhibit very many of the seven OCS organization principles. So it is not a comprehensive model of an organic system.

We have already presented three *approximations*, none of which are sufficient to develop comprehensive organic models: the *segment*, the *edge* and the *organodynamic graph*. And our modeling methodology intends that the organic systems modeler develop several of each of these first.

Then, one can assemble all of them together into a comprehensive Organodynamic web – or course utilizing a additional element that we shall add in this chapter.

OCS Organizing Principle Supported

Organizing Principles: All seven are collectively addressed.

Biological Example of the Issue

Consider a single prokaryotic cell. It exhibits all seven OCS organizing principles. (This idea is developed more thoroughly in a section later in this chapter.

Specific Challenges

We must include a new mechanism that represents the *routine behavior* mentioned in the introduction to this chapter. This mechanism will be a further elaboration of the *autocogeneration* idea of *regularity*. It will be most desirable if it can be defined in terms of the existing mathematical disciplines already utilized by the framework.

The Organodynamic Approach to Modeling the Issue

Provide a mechanism to join together multiple Organodynamic graphs. To do this, add a new mechanism to implement the repetition of Organodynamic graphs, and do so within a single integrated structure.

Organodynamic Networks

In the previous chapter, we defined the notion of an *Organodynamic graph* that consists of multiple *edges* or *segments* from “the past” (time step $t-1$) all of which “terminate” at a single *node* in “the present” (time step t). In addition, multiple new edges “originate” at the node (time t) and have concurrent time steps (time $t+1$).

In this node structure, one can say that edges from “the past” are connected to edges in “the future” via a node.

However, it is possible that an edge from the past originates at time $t-p$ where there is another node, and terminates at a second node in the present at time t . If so, then we could also say that we have two nodes that are “connected via an edge”. This is also true of one of the edges into future terminates at time $t+f$.

A more general way to view this construction is to say that we have two Organodynamic graphs, called A and B, where the node of A is at time $t-p$, while the node of B is at time t . Moreover, let's specify that we have an edge that originates at node A and terminates at node B.

Then, we have just combined two Organodynamic graphs into a single structure. We shall name this new structure an Organodynamic network.

More generally, let's define an Organodynamic network as:

Definition: Organodynamic network: A collection of Organodynamic graphs whose nodes are connected pairwise via edges.

We shall not demark this structure as an *approximation* in our methodology, because after adding one more network element mechanism we shall have arrived at “bigger game” – the simplex Organodynamic web. Thus, the reader should regard the Organodynamic network as a stepping-stone to the Organodynamic web.

Think of an Organodynamic network as network structure that is comprised of multiple interconnected Organodynamic graphs.

The next section will discuss the final mechanism that we shall add in order to arrive at the Organodynamic web structure.

Routine

To support the *autocogeneration* organizing principle, we have developed a number of mathematical mechanisms for the Organodynamics framework that, in some way, represent the notion of *regularity*.

However, there is a particular kind of *regular behavior* that we observe in virtually every kind of living system – and that we expect in any system that we should be willing to call “lifelike”. This is the idea of “routine behavior”.

By *routine*, we mean a repetition of a behavior sequence. In Organodynamics, we specifically mean that there is a contiguous sequence of time steps – each step having its own probability space – and that this sequence of time steps is repeated.

For example, suppose you 1) flip a coin, then 2) roll a die, and then 3) flip a coin again. This is a sequence of three time steps, each of which has its own probability space. (In fact, the first and third have the same probability space.

If you were to repeat this sequence, say, five times, then you would have five repetitions of the same three time steps. This is what we mean by *routine behavior* – a sequence of probability spaces that is repeated.

There is one point to make about a routine. Even though there is a sequence of *probability spaces* that are repeated, their *realizations* may not be repeating.

For example, of the five times that this example routine is repeated, its outcomes may be as follows:

Heads, 1, Tails, Tails, 6, Tails, Heads, 5, Heads, Heads, 1, Tails, Tails,
3, Heads.

Thus, even though the probability spaces are repeating, the *realizations* are not.

Thus, even though we have *regularity* at the level of probability distribution, we do not have it at the level of manifested outcomes. Thus, we have *irregularity* within *regularity*.

One interesting question is “How can we articulate this in the language of the *uncertainty model gradient*?”

In this example, the “thing” that was repeating five times is, in fact, an *edge*. That is it is a nonhomogeneous sequence of time steps – with no *nodes* involved anywhere.

But we would like for a *routine behavior* repeatable sequence to be more inclusive than merely an *edge* – to include contiguous time steps that have *segments*, *edges* and *nodes*.

Routine Behavior in Unconditional Stochastic Models

In this subsection, we shall look closer at *routine behavior*. We shall define a *routine* to be more inclusive than merely an edge. In fact, we shall allow it to be as big as an *Organodynamic network*.

Routines of Organodynamic Networks

And we shall also answer the question we posed in the previous subsection concerning “irregularity within regularity” and what that has to do with the *uncertainty model gradient*.

Suppose you have an Organodynamic network model of an organic process. Perhaps this model has two nodes and some edges coming into and out of both of these nodes, and perhaps an edge that connects the two nodes.

For this example, let's also assume that – from the perspective of the *uncertainty model gradient* – that we are dealing with the second type of model: the *unconditional stochastic model*. In other words, each time step is modeled by a single probability distribution. (We'll discuss Markov models later.)

Furthermore, suppose that when you “dig down” into this model all the way to the mathematics, you see that – at a certain time step – you begin to see a repetition of a contiguous set of time steps. For example, let's say that starting at time step 156, you observe that there are ten contiguous time steps (156 – 165) that “start repeating”. This means that their probability spaces (distributions) start to repeat. So, the sequence formed by steps 166 -175 look just like the sequence of steps from 156 – 165.

We shall call such a repeatable sequence of time steps a *routine*.

Moreover, we are going to allow these time steps to include time steps that are simple probability distributions as well as time steps that are *organodynamic transforms* in any sensible mixture. In other words, we are not confining our *routines* to include segments or edges. They can also include nodes.

If one considers all of the time steps in a routine, the elements involved can form an Organodynamic network. Mathematically, we can say that the closure of the time steps of the routine is an *Organodynamic network*.

To carry the example further, suppose that there are 25 contiguous repetitions of this 10 time step “routine”. This is what we mean by *routine behavior*.

Thus, a *routine* in an unconditional stochastic model is an Organodynamic network that is contiguously repeated.

Irregularity within Regularity

But, in living systems, we observe change within constancy, irregularity within regularity.

So, we want to allow *something* to change for every repetition. If it is not the probability distributions nor the organodynamic transforms, then what is it that we can allow to change?

The answer is: the *realizations* (outcomes) of each of these probability distributions can change for each repetition of the routine. The realizations can change, but the probability distributions must stay the same – if we want to call this “routine behavior”.

Putting this all together then, we have “routine behavior” whenever sequences of time steps within an *unconditional model* repeat, even though the corresponding sequences within the *realized model* do not repeat.

That is, we have *routine behavior* whenever the *second type of model* in the uncertainty model gradient has *routines*, even though the *third type of model* in the uncertainty model gradient *does not have routines*.

Routine Behavior in All Uncertainty Model Gradient Models

If we step back and look at what we resolved in the previous subsection, we can articulate it from the perspective of the *uncertainty model gradient*.

What we did was to describe the notion of *routine* for the *second* model of the uncertainty model gradient – the *unconditional stochastic model* – while allowing the *third* model to still change.

But we could also take the same view starting with the *first* model of the gradient. That is, we could describe a second kind of *routine* from the perspective of the *first* model of the gradient – the Markov model.

This means that we could describe a second kind of *routine behavior* in which the Markov model of an Organodynamic network sequence is repeated exactly, but in which the *second* model type is allowed to change – for those same time steps.

Thus, we have identified two types of *routine behavior*. They both involved an Organodynamic network of a particular model type “repeating” (remaining invariant) over successive contiguous time step sequences. While at the same time, the model type that was one lower in the *uncertainty model gradient* was allowed to be variant over those same iterations.

We shall name these two types of *routine* as follows: *Markov-routine* and *unconditional-routine*.

Or course, we can generalize this concept to apply to the third model type of the gradient. This would mean that an Organodynamic network sequence repeated exactly the same outcome – the exact same system organizations.

But this is precisely *periodic behavior*. Thus the third kind of *routine behavior* as applied to the *realized deterministic model* type, is periodic.

Formal Definition of Routine Behavior in Organodynamic Networks

Let us now present more formal definitions for these ideas.

Before we provide a formal definition of the notion of *routine* in our framework, lets define the notion of an *instance* of an Organodynamic model. Specifically, at this time we shall define an *instance of an Organodynamic graph*:

Definition: *instance of an Organodynamic graph*: An *unspecified* organodynamic graph is defined as beginning at a constant but undefined time step. If one assigns a specific time step index to an Organodynamic graph, then the graph is called an *instance* of the unspecified Organodynamic graph.

The idea of *routine* has a specific interpretation in terms of the uncertainty model gradient. It is this:

Definition: *Markov-routine*: an Organodynamic graph whose *Markov model* is invariant for each instance of the graph, but whose *unconditional stochastic model* may vary for each instance of the model.

Definition: *Unconditional-routine*: an Organodynamic graph whose *unconditional stochastic model* is invariant for each instance of the graph, but whose *realized deterministic model* may vary for each instance of the model.

Definition: *Deterministic-routine*: an Organodynamic graph whose *realized deterministic model* is invariant for each instance of the graph. This model is necessarily periodic.

Definition: *routine*: an Organodynamic graph that is either *Markov-routine*, *unconditional-routine* or *deterministic-routine*.

We have developed the notion of *routine behavior* so that we can add the single additional mechanism to our framework that we need in order to be able to define the near-comprehensive modeling construct that is our goal – the Organodynamic web.

Simplex Organodynamic Webs

If *routine behavior*, as described in the previous subsection, is observed within the behavior of an Organodynamic web, then a structural mechanism is needed to manage the repetitive articulation of the structure.

A simple device can accomplish this: a looping mechanism.

No new mechanism is required. All that is necessary is a simple graphing technique that involves the redirection, or reconnection, of the *edge* that occurs after the last time step of the first repetition of the *routine*.

Notice that a *routine* is an Organodynamic network that repeats. Notice also that any construct that we can consider that consists of segment, edges or nodes is an Organodynamic network. A *routine* exists whenever a “smaller” Organodynamic network repeats contiguously. However, the routine itself – the set of repetitions – constitutes a larger Organodynamic network. And if there are any non-repetitious time steps that lie outside of the routine, then the routine itself lies within a larger network structure that also constitutes a “containing” Organodynamic network.

Therefore, in general, a routine is a sequence of “smaller” Organodynamic networks that is a sub-network of a more inclusive network – which itself is an Organodynamic network.

This “more inclusive” Organodynamic network can be more efficiently described by replacing its *routine* by the looping mechanism described earlier in this section. The result is a simplex Organodynamic web.

This is accomplished by having the last time segment of the first repetition of the routine connect back to the first repetition of the routine. Once that is accomplished, then all other repetitions of the routine after the first can be eliminated.

What follows is a procedure for accomplishing this.

Reconnecting Some Edges

Identify a routine within an Organodynamic network. This will be a repeating Organodynamic sub-network within the larger Organodynamic network.

Next, locate the first repetition of the routine. Disconnect this edge so that it no longer connects to the first time step of the next iteration, and reconnect it to the first time step of the first iteration. This creates a looping mechanism.

Note that this looping mechanism is simply an abbreviation nomenclature for economically reducing the articulation of the network. As a stochastic process, it is still a sequence of time steps.

Deleting the Repeated Organodynamic Graphs

The remaining time steps of the repetitions of the routine after the first can now be deleted. They, of course, are represented within the looping structure.

Generalizing the Time Step Indices

However, within the looping structure, the time step structure no longer represent fixed indices. Rather these indices vary each time through the loop.

This idea of “times through the loop” is defined in terms of the model with its corresponding model at the next level down in the uncertainty model gradient. We shall leave this to further research to formalize these time step relationships.

Exit Conditions

Any looping mechanism needs a condition under which the loop terminates, and the repeating behavior changes.

In an Organodynamics network, this is provided by some sample point having a positive probability of lying outside of the sample space of the next time step.

Such a condition requires that some time step of the routine has a probability distribution that differs from that of a previous time step. This implies that the Organodynamic web over which the routine is defined is nonhomogeneous.

The Organodynamic Web Structure

We can summarize the Organodynamic web structure by defining it as a composition of connected Organodynamic network structures that can include the looping structure.

In other words, an Organodynamic web consists of the following elements: segments, edges, nodes, where the nodes can be interconnected by edges in such a way that originating and terminating time steps of edges or segments can be in positive or negative time order. This last statement enables the loopback mechanism.

Organodynamic Webs and the Seven OCS Organizing Principles

We have stated that the *simplex Organodynamic web* structure has been defined in such a way as to be able to support the exhibition of all-but-one seven OCS organizing principles. The principle not required is the *nestedness* principle.

This means that the network topology described in the previous section, together with the mathematical internal structures that comprise the elements of this network structure, work together to support the exhibition of these six principles within this single Organodynamic web structure.

In order to support such a claim, we shall look at an example of a basic biological system to consider how this organism might be modeled as an instance of an Organodynamic web, and how such a model might exhibit these seven organizing principles.

In the introduction to this chapter, we offered up the prokaryotic cell as an example of a system that could be modeled as an Organodynamic web.

We shall not at this point discuss further the steps involved in building such a model, but save that issue for the next section. Rather, we would like to now address the issue of the comprehensiveness of Organodynamic webs. We have suggested that what we mean by “comprehensive” is the ability within Organodynamics to model all seven of the OCS organizing principles.

Thus, at this time, we shall take up each of these six principles and discuss how and the extent to which the prokaryotic cell exhibits the principle.

Principle 1: Organized: We can model this cell as a system whose population is a collection of atoms, and whose organization at any moment in time is a collection of molecules plus some number of un-bonded atoms of the population. Thus, we can model this organization as we do in this framework: as a set of sets of clusters of duples, where each cluster models one of the molecules.

Principle 2: Emergent: the cell as-a-whole exhibits a number of systemic properties that none of its components alone do. For example, any property that cell has merely because it is a prokaryotic cell will not be exhibited by any of its

components – which are atoms. (The ability to undergo meiosis would be one such property.)

Principle 3: Nested: The first level of organization – molecules – can be further organized into small molecules (e.g. amino acids, nucleotides, etc.). At this level, each small molecule is an organization of atoms. A third level of organization could organize these small molecules as components whose populations are polymers.

Principle 4: Reorganizational: At each quantum time change, the organization of this closed set of atoms can change. This closed set of atoms can remain the same for a while. (A segment models this.) In addition, periodically the set of atoms changes. (An edge models this.) Sometimes some of the molecules combine chemically to form molecules. Sometimes these molecules further combine chemically, or separate chemically. (The system transforms model this).

Principle 6: Uncertain: These reorganizations are the result of chance encounters. Which organization will manifest at the next time quantum is probabilistic.

Principle 5: Autocogenerative: Even though these encounters are probabilistic, there are limitations on their behavior. These limitations are the consequences of stochastic dependencies from one time step to another. Specifically, the occurrence of any particular *organization*, or arrangement, of atoms at any particular time depends on the organization at the previous time step. These dependencies severely limit the possibilities from one time step to the next and lend various degrees of regularity to the behavior of the ways that the atoms and molecules of the prokaryotic cell can evolve in time. Specifically, in time, the prokaryotic cell can be seen to exhibit *routine behavior*. Each repetition of the routine can vary to some extent with each repetition. However, enough regularity is preserved in order to for the term *routine* to reasonably apply.

Principle 7: Persistent: These features combine to result in a stochastic expectation (as defined in information theory) that the prokaryotic cell will have a life expectancy of some reasonable finite length.

Principle 3: Nested: The prokaryotic cell does, in fact, exhibit this principle as well. But we shall ignore this fact for this chapter.

Building Models with Organodynamic Webs

To build a model of an organic system utilizing the Organodynamic web structure as a guiding template, one must first identify the elements of the target system and see how they can each be modeled in terms of the elements of an Organodynamic web.

These modeling elements include:

- Segments
- Edges

- Nodes
- Routines articulated as loops

Each of the segments is internally a homogeneous Markov chain of time steps. Each of these time steps is represented by a Markov transition matrix. Each of these transition matrices has a sample space whose sample points is a system organization of an underlying population of components.

Each of the edges is a sequence of segments. Both the underlying population of components and the Markov transition matrix is allowed to change between one segment and the next. The last time step of each segment must be contiguous with the first time step of the next segment in the edge.

Each node represents a connecting point where multiple edges from the past connect with multiple edges in the future. Each node is associated with a single time step. Each node connects possibly multiple time segments from the past. Each of these past time segments must terminate at the preceding time step to that of the node. Each of the future time steps must originate at the successive time step of the node.

Routines are represented within the Organodynamic web via *loopbacks*. A loopback is an edge whose terminating times step precedes its originating time step.

This topology of the high-level networking elements of an Organodynamic web together with the mathematics defined for each of these elements is intended to provide a modeling structure that can operate as a template for developing models of organic systems that exhibit all seven of the OCS organizing principles.

Incremental Example Model Development

<>

Progress Check

In Part II, Organodynamics, we are gradually and incrementally developing a mathematical framework for modeling living and lifelike systems according to the seven organizing principles of OCS. Once completed, this modeling framework will achieve the form of an abstract but highly flexible mathematical structure that we are calling an *Organodynamic web*.

We are using the example of a hobbyist's toolkit for building electronics as an analogy to Organodynamics. Both are systems for "building something", and in both cases the resulting "thing to be built" has a network structure and has been given a general name.

In the case of the electronics circuit toolkit, the general name of the "thing to be built" is "circuit". The circuits built by this kit are highly customizable to specific applications. In the case of Organodynamics, the general name of the "thing to

be built” is “Organodynamic web”. The models built by this mathematical framework are also highly customizable to specific applications.

Because of the complexity of organic systems, any model that represents them needs to be built in cumulative and increasingly accurate stages. Each stage is generally referred to as an “approximation”. Organodynamics as a model builder’s toolkit provides for six of these approximations. These approximation levels are well defined and occur at precise places in the methodology set forth by Organodynamics.

This chapter has presented *simplex Organodynamic web* structure of the Organodynamic modeling framework.

The simplex Organodynamic web structure is the fourth approximation level of the Organodynamics modeling methodology, and is the first version (of three) of the *grand scheme* of the framework, the Organodynamic web.

It is the first of the three approximation levels that are intended to be capable of building comprehensive models of organic systems. By “comprehensive” is meant that such a model can exhibit all seven of the OCS organizing principles.

However, despite our claims of comprehensiveness, there is one of the seven OCS principles that is under-represented in the version of the Organodynamic web presented in this chapter. While, the version of the Organodynamic web presented in this chapter is near-comprehensive, it nevertheless fails to sufficiently support one of the seven OCS principles – the *nestedness* principle.

Owing to the complexity introduced by the nestedness principle, we have left its inclusion into the Organodynamic web structure for a later exercise. For this reason, we have named the version of the Organodynamic web presented in the present chapter the *simplex Organodynamic web*.

The amended structure that is capable of supporting the *nestedness* principle is named the *composite Organodynamic web*. The concepts required to support nested systems in Organodynamic webs is introduced in the next chapter, and the *composite Organodynamic web* structure in a subsequent chapter.

Nested Systems

The third organizing principle of OCS, *nested systems*, states, “Any organic system has at least one component that is another system.”

This is perhaps true in many senses, and is a tantalizing concept. But our task is to define it rigorously, with clarity and in the language of set theory that we have been using for developing the Organodynamics framework.

The Issue Addressed

Organic systems are nested systems and usually involve several levels of organization. We need to make sure that our representation of these nested systems is able to capture, represent and articulate all of the various interrelationships that are involved.

The simplest of biological entities, the prokaryotic cells, are reasonably modeled as having at least three levels of nested organization: the cell itself, the macromolecules (polymers) and the small molecules that constitute the polymers. In such a case, the small molecules would constitute the components of the population of the system.

Our canonical biological example takes this a level deeper and defines *atoms* as the components of the population. However, this is a modeling decision, and a modeler is operating within her purview by making either decision.

Thus far, we have developed a way to represent a system and its state as the ordered pair (P; O), where P is the population of the system's components, and O is an *organization* (structure or arrangement) of those components – represented by a set of clusters of related pairs (called duples) of the systems population.

Whenever a system is not nested, because it only has one level of organization, we shall refer to it as a *simplex system*. Otherwise, we shall call it a *nested system*. Thus far in this text, we have only studied simplex systems – which are not actually candidates for being “organic”, since they are not “nested”. However, the complexities of system nesting are such that we have reserved the topic of system nesting until now.

A *nested system* contains a nested hierarchy of systems. We shall refer to any system within the nesting hierarchy of a nested system, at any level of organization, as a *subsumed system*. Subsumed systems will include the top-level system of the nesting hierarchy as well as the others all the way down the hierarchy.

With the (P; O) symbolism as defined so far, we can only provide a revealing descriptive articulation of simplex systems. Nested systems cannot be adequately described directly by this simple symbolism.

We need a mechanism for articulating nested systems that is capable of revealing all levels of organization of their nesting hierarchy within a single structure. That is, we need an approach to articulating a nested system that simultaneously expresses the contents of both the sub-populations and sub-organizations of all subsumed systems of all levels of organization within the organizational hierarchy of a nested system.

This chapter shall develop such a representation of nested systems. We shall name such representation the *composite representation* of a nested system. And, we shall show how this view can also be placed as a time step within an Organodynamic web.

We actually already have another way to describe a nested system within an Organodynamic web. We already described this view in the previous chapter in our discussion of the *Integrate* transform. We called this the *simplex representation*.

In order to represent a nested system using the simplex representation, a separate simplex representation must be provided for each subsumed system in the nesting hierarchy of a nested system. You will recall that we used the name *process view* to represent the use of multiple simplex representations to describe a nested system.

In fact, a single node type in an Organodynamic web – an *Integrate* node - connects all of the subsumed systems of a nested system. This is because, as described above in the subsection on the *Integrate transform*, it is the Integrate node that creates a nested system. Thus, anytime that there is an Integrate node, then the “edges” that follow from it represent the systems of the nesting hierarchy of a nested system. This description is, in fact, the *process view* of a nested system.

In other words, we already have one way to represent a nested system within an Organodynamic web: the *process view*. This view is good for representing the concurrent Markov chains that are involved after a nested system comes into being as the result of an Integrate node.

But the process view does not reveal the complex structure and interrelationships of the multiple levels of organization among the subsumed systems of a nesting hierarchy. For this we need a second view – the *composite view*, which we shall develop in the present chapter.

OCS Organizing Principle Supported

Organizing Principle # 3:Nested Systems.

Biological Example of the Issue

A prokaryotic cell can be modeled as a nested system. There are many ways to do this. One is to model a cell as its macromolecules (polymers) as its components.

But, each polymer is already a system of molecules that are related, or organized, in a specific manner. It is these organizations that make the polymer what it is. Thus, when representing the polymers as components of the cell's population, it would be desirable to use a representation of them that reveals their inner nature and their inner organization.

The same goes for the representation of the *organization* of the cell. When representing the polymers within the duples of the cell's organization, it would be desirable to use a representation of them that reveals their interrelationships and their internal organization.

In other words, if we could represent the organization of a cell as the organization of organizations of small molecules, then much of the true complexity of the organization of the cell would be evident in this representation.

Specific Challenges

We must develop a model of a composite system that contains within it all of the nested populations and their organizations for each subsumed system at every level of organization of the nested system hierarchy for that system.

This composite view should represent the state of a nested system at any time step within a Markov chains, and within an Organodynamic web. Thus, this composite system structure contains the state information of a nested system hierarchy.

The Organodynamic Approach to Modeling the Issue

We have already developed our view of a simplex system state. It is the ordered pair (P; O) – it population and organization. This pair represents *simplex system state* for a single step of an *organic process*.

In order to define a nomenclature for representing composite systems, we desire to preserve this nomenclature for simplex system, but extend it so that it can adequately represent composite systems in a manner that describes their component interrelationships at all levels of organization that are subsumed within the composite system.

Recall that we are modeling an organodynamic system as a probabilistic time series called a *stochastic process*. Each time step in the series is represented by a *sample space* whose logical possibilities are each quite complex in their own right. In fact, each such logical possibility is a *system organization* that the underlying organic system can be *realized* as in that time step.

For the simplex systems that we have discussed so far, each such *organization* is only “one level deep”. That is these *organizations* are not nested. What is new in this chapter is that each of these *organizations* may be nested – even deeply nested.

However, with *nested systems*, we are still have a sample space whose “events” are *system organizations*. The difference now will be that they are *composite organizations* – which means that they will be articulated using a new nomenclature that is capable of representing all of these nested levels of organization at once – including their internal interrelationships to all depths of organization.

Essentially, this nomenclature will represent the *nesting hierarchy* that describes a composite system and all of its subsumed system. Moreover, all of this information will be represented by a nomenclature that defines a single set (set-theoretic set) that includes the entire nesting hierarchy.

The principle challenge of this nomenclature will be to retain both the population nesting as well as the organizational nesting simultaneously within this hierarchical structure.

Our modeling methodology will still have the same six approximation levels, and each of these levels will still have the same three modeling types of the uncertainty model gradient. So, the methodology does not change.

The only complication, then, is that the sample points (events) in the sample space each represent a composite system, rather than a simplex system. Thus, the sample points are more complex than with simplex systems. One would expect that often the sample space would have more sample points (system organizations) than a corresponding simplex sample space. Thus there is a difference in degree and complexity.

Nevertheless, both simplex and composite time steps are both represented by a sample space of *system organizations* and an associate *probability distribution*.

Concomitantly, simplex and composite time steps each have the three model type representations that are promoted sequentially from one to the next for each level of approximation. This is all the same. The only difference between simplex and complex models is the complexity of each sample point (system organization) within the sample space for the process.

System Nesting Considerations

The nesting of systems within systems is a complex and fascinating concept. Also called *system composition*, or *composite systems*, this kind of system structure or organization occurs whenever the components of a system are themselves – in some sense – “other systems”.

One could make a valid argument that some nested systems have a mixture of simple and complex components. However, a theory of composite systems is complicated enough without having to contend with non-system components as well.

Therefore, hopefully without loss of generality, we shall define *nested systems* as systems whose components are (in some sense) other systems. If there is a

requirement for allowing non-systems to also be components of composite systems, then we shall leave this to further research. In this chapter, we must decide “in what sense” can one system be “nested inside another”, and provide a consistent definition of that concept that captures the properties exhibited by these kinds of systems.

The concept of system nesting is integral to the modeling of living and lifelike systems. Organodynamic systems models are generally deeply nested. Even the simplest biological system can be modeled as a deeply nested system. A single cell, for instance, can be reasonably modeled, as composed of macromolecules, which are composed of simple molecules, which are composed of atoms, and so on.

But deeply nested composite systems are obviously very complex. Therefore it behooves us to spend a considerable effort investigating their organization and dynamics – which we shall do in this chapter.

Examining the Limitations of the Simplex System Nomenclature

In order to example how we need to articulate and define a notion of *composite system* that will characterize organic systems in a way that we desire, let's revisit our canonical biological example.

Recall that it is a collection of molecules that can organize itself into various arrangements of atoms. We call these arrangements *organizations*. Each of these organizations is described as a particular “configuration” of molecules, plus possibly some “free atoms” that are currently “un-bonded”.

At any particular time, exactly one of these *organizations* is realized. Moreover, at any point in time, each of these *organizations* of atoms has some probability of being realized at the next point in time. We may even have conditional information that lets us predict more precisely what these probabilities are for the next time step if we know which of these organizations is realized at the current time step. If so, we have a Markov model.

However, so far, we have not distinguished between simple molecules, complex molecules and very large molecules such as polymers. Our current model treats all sizes of molecules as the same.

Another way to say this is that our current model is a *simplex model* of a system that is really not simplex! Rather, it is really composite. But we do not currently have a *representation mechanism* for capturing that “compositeness”.

Even more “embarrassing” is the fact that we cannot currently account for all of these different “levels of organization” of molecules simultaneously and in a manner that captures the fact that they are “contained within each other” as multiple layers of abstraction.

For example, suppose we want to represent a single nucleotide with our current nomenclature. What we would like to do is to represent the nucleotide as a system whose components are three molecules – a phosphate, a sugar and a

base. Then, in turn, we would like to represent the phosphate as a nested system whose components are *its* constituent molecules, the sugar as a nested system whose components are *its* constituent molecules, and the base as a nested system whose components are *its* constituent molecules.

But our current nomenclature cannot represent this nucleotide that way – as a system that contains three nested systems that contain constituent molecules. Rather, what we currently have to do is to represent this nucleotide as all of these constituent molecules from all three of the larger molecules *all thrown together* – with no demarcation of the fact that some of them come from a larger phosphate, sugar or base. Our current *simplex* nomenclature must omit any demarcation of the phosphate, the sugar and the base.

We need to correct this and provide a nomenclature that is able to express the fact that a nucleotide is composed of a sugar, a phosphate and a base, and the these three are molecules (nested systems) in their own right that also have their own components. Moreover, this new nomenclature must represent the fact that this nesting is occurring – that there are “three levels of organization” within this structure.

This ability is what we mean to provide in the present chapter.

A Composite Representation for our Canonical Biological System

To elucidate the issues at play here, let's look at a couple of examples of our canonical biological system that beg for a nested systems representation in order to see if they can shed more light in the nature of the nomenclature challenge and perhaps point to a solution approach.

A Nucleotide Molecule as a Nested System

As our first canonical example, let's suppose that our system is a collection of atoms that have assembled together into an organization of molecules that form a pure set of nucleotides. To keep it simple, let's also assume that these nucleotides share no atoms.

To a naïve chemistry student, it is reasonable to assume that each nucleotide molecule is “just a bunch of bonded atoms”, and not to appreciate the specific fact that three particular smaller molecules (a phosphate, a sugar and a base) historically “came together” to bond into the nucleotide. In fact, we note that there is apparently no historical information present in the nucleotide to know express that history. This student has a good argument when she insists, “There are no small molecules hidden inside of the nucleotide”.

On the other hand, it is only reasonable for a seasoned chemist who knows that there was a time when there were three separate molecules that subsequently bonded to become this nucleotide. The chemist would typically have a desire to model a nucleotide as a composite system with three small molecules as its components. And also to model each of these three small molecules as nested systems that are subsumed within the nucleotide system.

We have already accommodated the naïve student with the simplex model, which ignores the possibility that the phosphate, the sugar and the base are anywhere in the picture.

But, suppose we also wish to accommodate this chemist. To do this, we would want to model the nucleotide as a *composite system* using our “populations”, “organizations”, “clusters”, and “duples” nomenclature in such a way that some of them are “nested inside of each other”.

At the same time, we shall also want to be faithful to the fact that the nucleotide has a phosphate, a sugar and a base as components, and that each of these is a system in its own right with its constituent atoms as its components. And of course, all of these – the nucleotide, the phosphate, the sugar and the base – are systems with population and organizations of the form (P; O).

We need to somehow develop a nomenclature scheme that accommodates all of these simultaneously!

Lets tackle this problem systematically. Consider first the structure of the most complicated construct mentioned above – the *organization*. Recall that an *organization* is defined as a set of clusters. In our nucleotide example, we must ask “What entity do we need to convert into a composite representation. The answer is “the nucleotide molecule”.

Secondly, we must ask, “What portion of an *organization* represents a single nucleotide molecule?” The answer is “a single cluster” of an organization. We can generalize from this answer that: It is not an entire *organization* of a system that we want to “convert to a composite representation. Rather, it is merely a single cluster.

More precisely, we need to convert entities to composites on a cluster bases – not on a “whole organization” basis. So we are not desiring to “make composites” out of the entire *organization* of our population of atoms – but only of individual *clusters* within such an organization.

This is because it is the clusters that represent individual molecules; and it is these that we want to “break down” into composite representations. Therefore, we shall concentrate our embellishment of the nomenclature to represent compositeness on the clusters.

What we desire to do in our nucleotide example is to “isolate” the portions of each nucleotide cluster that represent the phosphates, sugars and bases as distinct systems in their own right. Then, secondly we want to represent the nucleotides molecules (the clusters) as systems whose components are the phosphates, sugars and bases. Then, thirdly, we also want to represent the phosphates, sugars and bases molecules as systems in their own rights, each with its own population and organization.

But, in our current simplex representation, the components of a nucleotide cluster are individual atoms. And there is no information contained within this

articulation that the phosphate, sugar and base molecules exist – even though our seasoned chemist knows better.

What we need is a mechanism by which we can represent the phosphates, sugars and bases as systems whose components are their constituent atoms; and then to change the representation of the nucleotide clusters so that their the phosphates, sugars and bases are its components – rather than the atoms. The challenge is to find a way to do that.

Here is a way that will work...

Step 1: Define a phosphate molecule as a system whose population P_P is its constituent atoms, and whose organization O_P is the set whose only cluster is the set of duples that represent the covalent bonds of its component atoms.

Step 2: Repeat step 1 for a sugar molecule and a base molecule.

Step 3: Redefine a nucleotide molecule so that its population consists of the *organizations* of the three smaller molecules – the phosphate, the sugar and the base. That is, the population of a nucleotide consists of three components:

1. The organization of a phosphate system
2. The organization of a sugar system
3. The organization of a base system

This step is the actual “trick” that gives us what we want. The trick is that we selected the *organizations* of the subsumed systems to be the *components* of their composite system. In this way, we have not lost the *relationships* among the components of the subsumed systems as we move up the hierarchy to their composite system.

Of course, what we have lost as we go up the nesting hierarchy is the *populations* of the subsumed systems. However, we shall have to assume, given any *system organization*, that we can locate its associated *population*.

[The astute reader will notice that we could have decided a different way and defined the components of the composite system to be the *populations* of its subsumed systems – rather than their organizations. We could then further assume, given a population, that we could always locate its *organization*.

However, there is a difficulty here. For each population, there may be many organizations, but not conversely. Therefore, by choosing the *organizations* of the subsumed systems to be the *components* of their composite – rather than choosing the populations – we can uniquely locate their populations.

However, the main advantage of choosing *organizations* for this purpose is that the interrelationships between related components at all levels of the nesting hierarchy are all represented within the composite system nomenclature.]

Step 4: Redefine the *organization* of the nucleotide molecule so that it contains as its only member a cluster whose duples are pairs of *organizations* of the phosphate, the sugar and the base systems.

Thus, the *organization* of the nucleotide system is a set that contains as its only member a cluster whose duples are pairs of organizations of the three small molecules.

To recap, the *population* of a composite system consists of the *organizations* of its components systems. And the *organization* of the composite system consists of a cluster of pairs of organizations of the three smaller molecular systems.

Admittedly, this is very complex. But it contains less information than the actual nucleotide molecule that it models. Moreover, it contains vital information about a nucleotide molecule at multiple levels of organization – including multiple sets of essential relationship, or structural, information at several levels of organization.

Importantly, these complex interrelationships can be recursively represented through multiple levels of a nested hierarchy to provide a nomenclature that is “flat” enough to be represented in the simplex “(P; O)” nomenclature, but yet contains all of this very rich information for all of its levels of nested organization.

DNA as a Nested System

The nucleotide example of the previous subsection contains several cases where an *organization* possesses only one *cluster*. This means that there were many cases where a set of sets had only one set in it. Admittedly this seems inefficient and begs the question of whether all of these levels of sets within sets were necessary in these definitions.

There are also other questions raised by the above nucleotide example. Specifically, the only *organization* in the composite nucleotide system that had multiple clusters in it was the first and lowest level of organization – the initial organization of the “atoms” population into an organization of some molecules plus some “free atoms”. In this structure, the molecules were modeled as clusters inside of the *organization*, and the free molecules went unmentioned in the *organization*. At least, this particular organization had several clusters – one for each molecule in the first level of organization. Some of these molecules were nucleotides while others were whatever other molecules happened to be formed when this space of atoms *organized*.

But, when we “reorganized” these nucleotide clusters so that they are composite systems whose components are a phosphate, a sugar and a base, none of their organizations had more than one cluster inside of them.

The question raised by all of this is “Is it always true that only the lowest level of organization of a composite system will ever have multiple clusters – and have no “free components” that are not represented by duples?”

The example model provided in this subsection will show that the answer is No. We shall provide an example “higher level of organization” whose *organizations* have multiple clusters – *and* whose populations have some “unattached” components (components with no relationships with other components).

This example is another model of DNA. It will be a different model that will concentrate on higher levels of organization within a DNA molecule, and show that they have these same complexities as discussed above. We shall not delve into the details of this example, but merely describe it prosaically.

Specifically, we shall deal with a higher level of organization within a strand of a DNA molecule which requires multiple *clusters* within its *organization*, and having “un-related” components. The example system is a gene within a DNA molecule as a system.

With apologies to molecular biologists, we shall –for the sake of pedagogy - take extreme liberty with these examples and pretend like these processes are much simpler, and more consistent, than they actually are.

We shall model a *gene* as a system whose components are nucleotides. The protein synthesis process will ultimately use some of these nucleotides to determine which amino acid molecules are needed to build a specific protein molecule.

However, the remainders of these nucleotides are essentially “junk” and are eventually “ignored” by the protein synthesis process and never formed into triplets to be mapped to amino acids. These nucleotides are referred to as *introns*.

To make matters worse, the set of nucleotides that constitute a gene can be interspersed with a contiguous sequence of introns followed by a contiguous sequence of nucleotides that are *not* introns (called *exons*). This pattern can continue so that a gene can be seen as a pattern of alternating intron subsequences and exon subsequences.

In the face of this, how can we model a *gene* using our nomenclature? Here’s how. We can define a gene as a system (P; O) whose *population* P is a collection of nucleotide systems, and whose *organization* O is defined as follows. As an *organization*, O must be a set whose elements are clusters.

In this model, we are going to let the intermittent sequences of exons be represented a clusters (one exon sequence to one cluster). And, we are going to ignore all of the intron sequences of nucleotides and not represent them within the *organization* of the gene. (This is similar to our canonical atomic model where we ignored the “free atoms” in the *organization* and represented the atoms that became parts of molecules as duples within clusters.)

Thus, the *organization* O in our gene system has clusters, each of which represents an exon. But, what are the *duples* of these clusters? Recall that a duple represents *relationships* among components – nucleotide molecules. The relationships that we want to represent here are the chemical bonds between

adjacent nucleotide molecules in the DNA chain. (We are ignoring temporarily for the sake of simplicity that DNA is a *double* helix.)

Thus, each cluster within our *organization* is a pair of adjacent molecules within an exon of a gene.

Thus, we have demonstrated that there is a relatively high level of organization (a gene) within a highly nested system (a strand of a DNA molecule) whose representation in our nomenclature requires that its *organization* 1) required multiple *clusters* and 2) has no representation of some of the components (nucleotides) of its underlying population.

More than Merely a Pandora's Box

System nesting certainly includes the notion of “components inside of components” and “collections inside of collections”. This is the idea celebrated by the “Pandora’s box” idea – “components inside of components”.

However, system nesting goes considerably beyond a “Pandora’s box” “containment concept”. We touched upon this earlier, but we should look more closely at the issue now.

More significantly, it must also include a very strong notion of *interrelationships*. There are relationships among sibling systems. And, there are also relationships between component systems and their composite system. It is this latter concept whose implementation requires considerable intellectual machinery within Organodynamics – and that brings considerable power and application scope to this framework.

Thus, in order to construct a modeling system for representing system nesting, we are going to have to dig very deeply into the mechanism of nested systems and how they work – perhaps deeper than has yet been done by any investigation of contemporary systems theory.

In so doing, we shall unearth a surprisingly rich set of relationships between the components of multiple levels of organization within nesting hierarchies. We shall also discover that there are many possible ways of modeling these intricacies. And we shall select from among them one such approach.

Our approach is intended to be very general – and to provide a first approximation that should be generally suitable to multiple levels of organization within such a hierarchy.

Certainly, more specific modeling approaches could represent more faithfully specific applications and specific levels of any such hierarchy. While the development of such specific OCS models are encouraged, Organodynamics shall present a general model that is intended to be broadly applicable – and to form an example that can be used to develop more specific OCS modeling frameworks.

Composite and Component Systems

Having looked at some examples of how we would like to define the notions involved in composite systems, lets now attempt to put some order to these ideas.

It is evident that we shall need to distinguish between *two different roles* that a system can play in the context of *nested* systems.

One is role is that of a system whose components are other systems. We shall call a system operating in this role a *composite system*.

The other role is that of a system that is a component of another system. We shall call a system operating in this role a *component system*. Of course, a single system can play both roles simultaneously. This occurs whenever a system has components that are systems, and is also a component of another composite system.

Lets now look at these two roles individually to understand some of their general properties and relationships within the domain of composite systems.

Composite Systems

Like all systems in Organodynamics, systems that are operating in the role of a composite system are defined as an ordered pair $S = (P_S, O_S)$, where P_S is the population of S (the collection of its components), and O_S is an *organization* of S .

But, composite systems are different from simplex systems because their components, the members of P_S , are systems in their own right.

We would like to have a way to represent these component systems as members of P_S in a way that reveals that they are systems in their own right at the same time that they are acting in the role of components of P_S .

In other words, our task is to develop a nomenclature for representing composite systems. The whole idea of this nomenclature is to carry along the description of any component systems within the description of their composite systems so that the entire structure of the composite system – at every level of subsumed organization – is captured.

If we can achieve this, one can look at a fully articulated rendering of the nomenclature for a composite system and locate all of its subsumed structures, to identify their components and their interrelationships (organizations) at every level of the component hierarchy.

Developing such a nomenclature is a problem that we need to solve presently.

Component System Representation

For convenience, lets work with an example composite system named

$$S = (P_S; O_S).$$

Our nomenclature has the task of representing both the *population* P_S and the organization O_S of a composite system in such a way that

1. The population P_S and organization O_S are clearly distinguishable and identifiable.
2. The components of P_S are articulated as systems in their own right with their own populations and organizations.
3. Any organization O_S of S is a set of clusters whose duples contain components of the population P_S of S .

Item 2 above concerns itself with the problem of representing the components of the composite system S – which are systems in their own right – in such a way that both their population and their organization is also articulated.

Admittedly, both of these “bits of information” are needed in order to fully articulate a composite system. However – for the sake of brevity – we are going to take a shortcut and *only use the organization* of a subsumed system (a *component system*) to represent it within this nomenclature.

(We will assume, if we know a component system’s organization, that we have access to its entire (population; organization) definition, and can therefore gain access to its population definition if needed. Certainly we can achieve this if these data structure were implemented in software. We shall assume that we can also do so in set theory.)

Representing Component Systems by using their Organizations

So, we have decided the following for our nomenclature:

Within the composite system nomenclature, component systems will be represented by their organization only. It will be assumed that their population descriptions can also be accessed.

This decision results in a more concise nomenclature, while at the same time capturing the portion of the system definition that is emphasized by Organodynamics: their *organizations*.

Example Component Systems Representation

Lets now look at a simple example of this nomenclature.

Let $S_A = (P_A; O_A)$ be the composite system of our example. Assume that S_A has two component systems $S_{A1} = (P_{A1}; O_{A1})$ and $S_{A2} = (P_{A2}; O_{A2})$. Then, according to our naming scheme, the population P_A of S_A has two members: O_{A1} and O_{A2} , which are the *organizations* of P_{A1} and P_{A2} .

Lets specifically define the contents of the populations and organizations of the two component systems S_{A1} and S_{A2} as follows:

Let $P_{A1} = \{11, 12, 13\}$
 Let $O_{A1} = \{ \{(11, 13)\} \}$
 Let $P_{A2} = \{21, 22, 23\}$
 Let $O_{A2} = \{ \{(21, 22)\}, \{(21, 23), (23, 22)\} \}$.

Our task is to represent the composite system $S_A = (P_A; O_A)$ using our new nomenclature. To achieve this, we will start with $S_A = (P_A; O_A)$ and expand via substitution all entities until they are fully articulated.

We shall first tackle the expansion of the P_A portion of the pair $(P_A; O_A)$, and work on expanding the O_A portion later.

Since in our scheme the *components of a population* of a composite system is the set of all the *organizations* of its component systems, then

$$P_A = \{ O_{A1}, O_{A2} \}.$$

Substituting for the members of O_A, O_{A2} , We have

$$P_A = \{ \{ \{(11, 13)\} \}, \{ \{(21, 22)\}, \{(21, 23), (23, 22)\} \} \}$$

Having described P_A in our nomenclature, lets now describe O_A .

We have yet to define O_A , so lets do that now – and then expand it.

Now, O_A can be any set of clusters whose duples are pairs of elements of O_A .

As we just showed, O_A has two elements:

$$\{ \{(11, 13)\} \} \text{ and } \{ \{(21, 22)\}, \{(21, 23), (23, 22)\} \}$$

Lets give them names, say:

$$X = \{ \{(11, 13)\} \}, \text{ and } \\ Y = \{ \{(21, 22)\}, \{(21, 23), (23, 22)\} \}$$

Thus, the duples of the cluster of O_A must be pairs of these two – allowing for the first and the second entry of any pair to be the same or different.

Considering this, lets arbitrarily invent such an organization:

$$O_A = \{ \{(X, X), (X, Y), (Y, X)\}, \{(Y, Y)\} \}$$

Notice that this organization has two clusters. The first consists of the three duples $(X, X), (X, Y), (Y, X)$ and the second consists of the single duple (Y, Y) .

In preparation for expanding O_A to a fully-expanded articulation in our nomenclature, let's rewrite our expression for O_A so that there is a single ordered pair on each line of text:

$$O_A = \left\{ \left\{ \begin{array}{l} (X, X), \\ (X, Y), \\ (Y, X) \end{array} \right\}, \left\{ (Y, Y) \right\} \right\}$$

Fully expanding O_A , we obtain:

$$O_A = \left\{ \left\{ \left\{ \{(11, 13)\} \}, \{(11, 13)\} \right\}, \left\{ \{(11, 13)\} \}, \{(21, 22)\}, \{(21, 23)\}, \{(23, 22)\} \right\}, \left\{ \{(21, 22)\}, \{(21, 23)\}, \{(23, 22)\} \right\}, \{(11, 13)\} \right\}, \left\{ \left\{ \{(21, 22)\}, \{(21, 23)\}, \{(23, 22)\} \right\}, \{(21, 22)\}, \{(21, 23)\}, \{(23, 22)\} \right\} \right\}$$

This expression essentially fulfills our needs. Every organization of every system – composite and subsumed – is listed somewhere in this articulation. Moreover, these are in the correct relation to each other.

The only things that are missing are the populations of the subsumed systems. But these can be retrieved for know relationships between an organization and its unique population.

It will be noted that this articulation – although it contains or references the complete population and organizations of all subsumed component systems as well as the same for the composite system itself, it is hardly a human-readable expression.

We shall rectify this problem by providing a software library that manages (synthesizes, parses, etc.) the nomenclature. The software library shall accompany the Organodynamic framework.

Eschewed Alternative Approaches

Of course, we could have defined the population of a composite system so that its components were not merely articulated as the organizations of their components but as the complete ordered pairs (P; O). However, such an approach would be needlessly complicated.

We could also, alternatively, have defined the population of a composite system as simply the populations of its component systems. This may be reasonably called the “Pandora’s box approach”. But such approach would have omitted all relationship description of the components within the nomenclature for the composite. Obviously, this is insufficient for the purposes of Organodynamics.

Deeply-nested Composite Systems

Biological systems are often “deeply nested”. That is, there are systems within systems within systems to many levels of depth. We need to model this concept of deeply nested systems in Organodynamics. That is, composite systems should be able to be “nested” to any finite level of organization.

The title of this text includes the word “comprehensive”. One of the implications of this term is that a single instance of an Organodynamic model is capable of modeling the entirety of an organism (to some degree of fidelity). It is this capacity to model deeply nested composite systems that is responsible for much of Organodynamics capability to claim “comprehensiveness”.

Deep nesting is supported by the nomenclature that we developed above. It is easy to see this if one contemplates building a nested system “from the bottom up”.

This concept results in a *nested systems* hierarchy, in which – in the middle of the hierarchy there are systems that are component systems of composite systems “above them” in the hierarchy. But they are simultaneously composite systems of component system “below them” in the hierarchy. Thus, “composite” and “component systems” can be relative terms.

Algorithm for Constructing a Composite System

A composite system is constructed “on top of” one or more existing systems – which then become its components.

That is, given some finite number n of systems $\{A_1, A_2, \dots, A_n\}$, a composite system $B = (P_B; O_B)$ can be constructed with this set as its component systems. Each of these systems A_{jk} has a population P_{jk} and an organization O_{jk} , so that $A_i = (P_{Ai}; O_{Ai})$.

We need an algorithm for constructing composite system $B = (P_B; O_B)$, given systems $\{A_{11}, A_{12}, \dots, A_{1n}\}$, which will be the component systems of system B.

In fact, this procedure is precisely what is accomplished by the *Integrate* organodynamic transform that was discussed in the chapter on *autocogeneration* above.

The procedure for creating a composite system is this:

1. First, define the *population* P_B of composite system B as

$$P_B = \{ O_{A1}, O_{A2}, \dots, O_{An} \}.$$

In words, the population of the composite system is the collection of the *organizations* of its component systems.

2. Second, define the *organization* O_B of composite system B. O_B can be any set of clusters whose duples contain elements of the population P_B of B.

Composite Systems Representation in the Uncertainty Model Gradient

In this chapter we have described how to extend the set-theoretic nomenclature that we have been using to represent the *states* of simplex systems to a format that is capable of representing the states composite systems in a manner that includes all of the organizational interrelationships among the components of all of its subsumed, or nested, systems.

Essentially, this nomenclature represents the *organization* of a composite system (which is defined in OCS to be its *state*) as the organization of the organizations of the organizations...of the organizations of its component systems to as many levels of depth as there are nesting levels within the composite system.

In this section, we shall describe how this notation is used within each of the three model types of the *uncertainty model gradient*. Recall that these three are:

1. The Markov model
2. The unconditional stochastic model
3. The realized deterministic model

Each of these three model types is a stochastic process - a sequence of mathematical entities that are associated with time steps, also called a time series.

Being stochastic processes, the mathematical entities of each time step of each model type involve, in some way, one or two probability distributions, depending on model type. In other words, each of these model types can be seen, in some way, as a sequence of probability distributions.

Of course, any probability distribution has a *sample space* of *sample points* (or *events*), each of which is assigned a probability value. In all three of these model types, the *sample points* are defined in Organodynamics as *system*

states. Specifically, system states in Organodynamics are defined as *system organizations*.

Thus, prior to the present chapter, the sample points of all of these probability distributions have been instances of the simplex system nomenclature.

In order to accommodate composite system, the only change to any of these probability distributions – in any of the time steps of any of the three model types – is that the nomenclature of the *sample points* is now the extended notation for representing composite systems, rather than the limited, single level-of-abstraction special case of the simplex systems.

The only other difference is one of degree – owing to the added complexity of composite system representation within our nomenclature, there are vastly more possible system organizations for each sample space. Consequently, the size of the sample space for composite systems is likely be far vaster than for simple system.

Lets now briefly visit this issue for each of the three model types.

The Markov Model

The Markov model represents the probabilities of the outcomes (system organizations) of the next time step, given that the outcome of the current time step is known. Thus, for each possible outcome of the current time step, the Markov model provides an associated probability distribution for the outcomes at the next time step.

A Markov transition matrix represents this set of probability distributions. This situation is thought of as two distinct probability distributions – or more precisely a *joint probability distribution*). The first of these represents the probabilities of the possible outcomes of the *current step*. The second of these represents the probabilities of the possible outcomes of the *next step*.

Moreover, each of the outcomes for both the current step and the next step are, In Organodynamics, a *system organization* that is articulated using our simplex system nomenclature.

The only change to this model for composite systems is that the *composite system nomenclature* is used. Since the simplex system nomenclature is a special case of the composite system nomenclature, then this change is essentially a generalization of the Markov model for simplex systems.

Of course, we can expect Markov transition matrices to generally be considerably larger than those for simplex systems – but only because it is expected that they will often have many more possibilities.

The Unconditional Stochastic Model

This model is represented by a single probability distribution – describing the probabilities for the *next* time step. Again, the sample space is the set of all possible *system organizations* that can be manifested at the next time step.

As with the Markov model, the only change to this model for composite systems is that the *composite system nomenclature* is used. Since the simplex system nomenclature is a special case of the composite system nomenclature, then this change is essentially a generalization of the Markov model for simplex systems.

Of course, we can expect Markov transition matrices to generally be considerably larger than those for simplex systems – but only because it is expected that they will often have many more possibilities.

The Realized Deterministic Model

This model is deterministic, and does not actually need to be articulated as a sequence of probability distributions. However, we do so for constituency with the other two models – which are stochastic, and because we can, since determinism is a special case of stochasticism in which the probability of exactly one of the sample points has a value of one and all the others have a probability value of zero.

To represent this, we begin with the previous model (the unconditional stochastic model) and promote it to this one preserving the same sample space of system organizations and changing their probabilities so that the system organization that was actually realized is assigned a probability value of one.

As with the other two model types, the only change to this model for composite systems is that the *composite system nomenclature* is used. Since the simplex system nomenclature is a special case of the composite system nomenclature, then this change is essentially a generalization of the Markov model for simplex systems.

The Inherent Complexity of Modeling Nested Systems

System nesting is a very complex subject. Even the modeling of the simple real-life examples listed above can become too lengthy and complicated for an text of this type. For example, a reasonably complete Organodynamic model of the mycoplasma bacteria would be considerably beyond the scope of this text, and would constitute a lengthy modeling project.

The essential challenge is that complex systems, by their very nature, defy tractability. Thus, any model of complex systems will necessarily flirt with intractability in order to be verifiable. But, at the same time, it must attempt tractability in order to be a useful model. Any mathematical framework capable of modeling these systems, in order to be useful, must make simplifying assumption so that it can be more tractable than the systems that it intends to model.

Nevertheless, any such modeling system, to be useful, will still result in complex useful models – but hopefully more tractable than the systems they model. Thus, a good organic modeling system will provide models of intermediate complexity. They can be tractably more complex than models produced by less rich modeling systems, but less complex than the organic domains that they model. Herein lies their value.

Incremental Example Model Development

<>

Progress Check

In Part II, Organodynamics, we are gradually and incrementally developing a mathematical framework for modeling living and lifelike systems according to the seven organizing principles of OCS. Once completed, this modeling framework will achieve the form of an abstract but highly flexible mathematical structure that we are calling an *Organodynamic web*.

We are using the example of a hobbyist's toolkit for building electronics as an analogy to Organodynamics. Both are systems for "building something", and in both cases the resulting "thing to be built" has a network structure and has been given a general name.

In the case of the electronics circuit toolkit, the general name of the "thing to be built" is "circuit". The circuits built by this kit are highly customizable to specific applications. In the case of Organodynamics, the general name of the "thing to be built" is "Organodynamic web". The models built by this mathematical framework are also highly customizable to specific applications.

Because of the complexity of organic systems, any model that represents them needs to be built in cumulative and increasingly accurate stages. Each stage is generally referred to as an "approximation". Organodynamics as a model builder's toolkit provides for six of these approximations. These approximation levels are well defined and occur at precise places in the methodology set forth by Organodynamics.

In this chapter, we have extended our notion of *system*. Prior to this chapter, a system has been defined as an ordered pair (P; O), where P is a *population* of components and O is an *organization* of the components or that population.

In this chapter, we have extended this definition to accommodate the idea that the components of a system can be, in their own right, other systems. Since the notion of *system organization* is of supreme importance in OCS, it is essential that we extend our articulation of a system so that the nomenclature used is capable of representing the inter-organizational relationships at all levels of nesting.

We were able to develop such an extended notation by representing the components of a composite system as the organizations of their component systems.

This required that we build a composite system from the “bottom up” – which imitates biological systems. We provided an algorithm for performing this act, and noted that the algorithm is precisely that implemented by the *Integrate* organodynamic transform.

Conditional Dependency Relationships in Composite Systems

You may have noticed from the example composite systems presented above that component systems must be “somewhat well behaved” before they can assemble into a higher level of organization. That is, of all of the possible organizations that each component system can assume, only some of them have the property that they can assemble together with some of their sibling component systems with the result of forming an organization at the next higher level.

In other words, in order to form into a composite system, the component systems are generally constrained to a small subset of all of their possible organizations.

Another way to say this is that the formation of a composite system *depends upon* each of the component systems being organized in one of certain specific ways. Not all possible *organizations* of each of the components systems will enable them to join together (to *integrate*) to form a larger organization with their sibling component systems in order to form a composite system.

A third way of saying this is that the formation of a composite system *depends upon* the component systems each organizing itself into certain “arrangements” that permits them to *organize with*, or *relate to* each other.

These conditional dependencies between the levels or organization are not evident in our definition of composite systems. However, they were pointed out in all of the examples of actual composite systems presented earlier in this chapter, such as molecular systems, corporate institutions and software applications.

In this chapter we shall add some formality regarding these organizational layer interdependencies into the Organodynamic web model that we are currently constructing.

The Issue Addressed

Suppose we have several systems. Lets look at what it takes for such a set of systems to form a composite system. In other words, we are asking the question “Generally, in real live systems, what are the necessary conditions for a collection of systems (lets call them *component* systems) to be able to integrate into a single *composite system*?”

The answer we are looking for in this chapter is that: Each of the component systems “has to be ready” before they can integrate into a composite system.

But what does “have to be ready” mean? It means that each of the component systems must be *organized* in a manner that permits it to *join together with the others* – whenever they are also “properly organized”. Thus, when all of them are “individually properly organized” then it is possible for them to “join together”, or *integrate*.

There are several more points to be made here. One is that there may be many ways that each of these component systems can be individually organized but that still allows them to integrate into a composite system.

Of course, keep in mind that – and this is the second point - *integration into a composite system* also means that a new, larger organization is formed – an organization that relates those component systems to each other.

A third point is that, in all the many ways that these systems may be individually organized and still permit them to integrate into a composite system, each of these different ways that they can be individually organized may result in distinct organizations of the resulting composite system.

And there is one final point. For all the ways that these components systems can individually be organized in such a way that they can integrate into some composite system, there are generally many more ways that these components system can be individually organized that does not lead them being able to integrate into a composite system at all.

All of this can be summarized by saying that component systems *depend on each other being organized (arranged) in certain way in order for them to be able to integrate into some composite system*. And, we can say that these individual organizations that permit integration are, then, the conditions upon which this integration depends. In other words, the component systems depend on each other in order to be able to integrate.

Moreover, in order for a particular composite system to come into existence, and to maintain its existence, it depends upon its component systems maintaining their individual organizations.

And, even beyond that, it even occurs that the maintenance of a relationship within an individual components system (its organization) depends upon this component system being related to another sibling component system. Of course the relationship between these two components systems is one that is defined within the *organization* of the composite system.

There is an expression that is encountered in Vedanta literature - “interdependent co-origination” that is brought to mind by the above description.

In any event, this chapter aims to provide the necessary intellectual equipment to the Organodynamics framework in order to be able to articulate and investigate these issues pertaining to composite system, their component systems, their origination, maintenance, annihilation, composition and decomposition.

OCS Organizing Principle Supported

Organizing Principle # 3:Nested Systems.

Biological Example of the Issue

Consider a prokaryotic cell.

Our canonical biological example would treat this cell as a semi-permeable space, and model it as a closed space that is intermittently opened to allow an exchange of atoms and molecules, and then that is closed again for some time period.

Within these closed time periods, this cell can be modeled as a system of atoms whose state space of possible arrangements into molecules plus free atoms (its *organization*) describes all of the ways that the collective atoms of the cell can arrange themselves into molecules.

Of course, of all of the *logically possible organizations* that his cell can be arranged into, only a small percentage (still vast in number) are actually possible in biology, since the integrity of the cell must be maintained in order for it to persist as a cell.

Thus, given that at a single time step, this cell is arranged into exactly one of that state space of organization, there are only a limited number of those organizations that it can transition to in the next time step. This is because any particular arrangement of molecules only permits a relatively small number of arrangements to be transitioned to at the next time step.

In other words, a particular organization of these atoms into molecules is *dependent upon* certain *conditions* – these conditions being the immediately prior organization of those same atoms.

Specific Challenges

We must make a deep observation of the manner in which deeply nested system exhibit interdependencies within, between and among their systems, components and layers.

The challenge to such a study is to determine how we can enhance the Organodynamic framework in a way that preserves our current approach but also adds the required intellectual equipment so as to be able to account for these interdependencies and conditions within the framework's mathematical mechanisms.

The Organodynamic Approach to Modeling the Issue

The mathematics of probability theory provides the mechanisms that we need. This is because these mathematics account for both the existence and absence of dependencies and interdependencies, but also their degree of dependency. This is accomplished through the twin devices of conditional probability coupled with the ideas of dependent and independent conditions through the use of joint probability distributions.

This chapter will introduce some of these ideas and leave other for further research.

In addition, the nomenclature constructs that have been defined so far in this text also needs an update in order to promote the inter-relationships and interdependencies involved in this chapter.

In order to best understand how to do that, we investigate the field of software engineering. It seems that software engineering is a discipline through which human engineers have had to delve into the notion of deeply nested systems to the point that they have created a rigorously articulated discipline around these ideas.

While it can be argued that biological systems are more complex composite systems than those designed and constructed by software engineers, nevertheless the discipline of software engineering has had to become self-conscious about the organization of these systems. It has therefore created a rich discipline and methodology *about* the organizations of deeply nested systems that biological systems (having constructed themselves) never needed.

This “meta-systemics” product of the software engineering discipline is most instructive in this regard, and we shall leverage its findings and artifices in our study of the interdependencies of deeply nested systems.

Organizational Dependencies in Nested Systems

We saw in the previous chapter that the ability to define nested systems affords the opportunity for the number of possible system organizations for an underlying population of components to explode.

However, even though these sometimes vast number of logically possible system organizations are theoretically potentialities, in actual practice, many of them can never be realized. This is often due to the fact that most applications, many sibling inner systems (component systems) never form meaningful relationships with each other – and therefore should never be represented by *duples* within any clusters of any *system organizations*.

In fact, in most applications, the number of pairs of subsumed component systems of a composite system that are “actually related” is relatively small. This fact has a tendency to reduce the sizes of sample spaces from “vast” to more manageable.

This fact encourages us to look at what it means for two components systems (as some level of organization with the composite systems hierarchy) to be related. There are many possibilities, but some of them stand out as often repeated, and deserve to be codified within our composite systems framework.

Specifically, we must look at the fact that when two components systems are related at one level of the hierarchy, at the immediate lower level these components are represented as system organizations (set of clusters of

duples). It is often the case that these two components systems are “related” at the higher level in the context of the application because of the fact that their subsumed organizations have components that are “cross related” – again in the context of the application.

An example if this would be this: Suppose there are two atoms that are covalently bonded to each other. However, suppose further that one of the atoms is considered to be a “component system” of a larger molecule, and the other atom is considered to be a “component system” of a second larger molecule. Then, we can also consider to two larger molecules to be “related” to each other by virtue of the fact that their “component systems” (the two atoms) are already related.

This kind of “propagated” relationship is rampant in living systems, and deserves to be supported within our modeling framework. It should be further observed that these kinds of relationships represent *conditional dependencies* between and among components systems within a nested systems hierarchy.

Moreover, these types of conditions can be modeled by attributing increased probabilities to their occurrences, and decreased probabilities to the occurrences of duples between component systems between which these conditions are not obtained. In other words, the presence of these conditional dependencies can be modeled with probability distributions – a reverberating theme within the Organodynamics framework.

In this chapter, we shall visit more examples of this in order to abstract the general mechanism that is behind these kinds of component relationships across sibling component systems within composite systems, and even across levels of organization. From these abstractions, we shall develop additional intellectual equipment to add to the Organodynamics framework.

Examples of Dependency Conditions in Nested Systems

In order to motivate the approach to modeling system nesting described above, we shall first review a number “real life” examples of nested system hierarchies – both in and out of the life sciences. Some of these examples may not be organic systems because they may not exhibit all seven OCS organizing principles. Nevertheless, they are examples of nested systems.

These examples can give us key insights to guide our further development of the modeling of composite systems in Organodynamics.

To assist in this process, we shall articulate each example in the language of composite systems that we have begun to develop so far in this chapter.

In order to set the stage for our discussion of nested systems, we shall identify here a brief list of examples of nested systems. The following list will be developed further in the subsection below.

- A Simple System of Molecules
- A Software Application

- A Corporate Institution
- A DNA Molecule

A Simple System of Molecules

Consider first a system of two simple molecules that have bonded to form a larger molecule. From a modeling perspective, we shall consider this to be a nested hierarchy that exhibits three levels of organization.

The lowest level is that of the atom. The second level - whose components are the atoms - are the two simple molecules – type unspecified. The third and highest level of organization in this example is the “small macromolecule” formed by the bonding of the two small molecules.

(It is evident through this example that the choice of the number of levels of organization to use in this model is an arbitrary choice on the part of the modeler, as is generally the case with the modeling of any natural phenomenon. As usual, it is tough to “do modeling” without a modeler-observer.)

In this example, relationships occur at two levels of organization. First, within each atom, the electro-magnetic and strong forces bond the components of each atom together. This, of course, is occurring at the sub-atomic particle level – that is, at the level of the *components* of the atom. (We could have modeled this in other ways. But this approach is reasonable to some level of approximation.)

However, at the second level of organization, the atoms form each molecule via an interrelationship called the electromagnetic force. Specifically – and this is key to the Organodynamic approach to modeling – certain components of the atoms, namely the electrons (subatomic particles), relate to each other through the electromagnetic force – through which they subsequently bond the two atoms together into a molecule.

Note, however, that this bonding of atoms was “implemented” through the bonding of their components, subatomic particles. This is a significant point.

That is, a relationship between the components – the electromagnetic forces between certain subatomic particles – became “propagated upward” to the atom level. And at the atom level, the atoms also became related – bonded – through this same electromagnetic force that was “inherited” from the subatomic particles.

That is, the relationship that holds the atoms together is actually “inherited” from the subatomic level - a lower level of the nested hierarchy.

This is a kind of “relationship inheritance” mechanism. However, in this case, inheritance is propagated from the bottom up. This phenomenon turns out to be much more than an isolated effect for this particular example. Rather, we shall see below in the other examples as well that this mechanism reappears again and again to ramify relationships at one level of a nesting hierarchy throughout higher levels of the nesting hierarchy.

Another point to make is that, at any of these levels of organization, there is the concept of “availability to bond”, or “availability to pair.” For example, one atom can “expose to another atom its availability to pair” with that other atom through a covalent bond by having an electron that is can share with the other atom.

Of course, even if one atom has an electron that it can share with another atom, that doesn’t necessarily mean the sharing will actually occur. For that to manifest, certain conditions, dependencies, would have to arise.

Thus, there are three possible stages with respect to a pairing of components via some relationship: 1) There exists two components (atoms). 2) One becomes available for pairing with another components. 3) The pairing actually occurs.

In general, then, in order for one nested entity to be able to pair with a second entity in relationship, there must be some condition in the first that is capable (via some force) of being bonded with the second entity. In Organodynamics we shall represent such pairings, or relationships, as a pair of the components of the two nested entities.

A Software Application

As our second example, lets consider the development of a software application using a contemporary class-based object-oriented programming platform. (For example, Java and C++ are class-based object-oriented languages.)

The presentation of this example will be rather extensive. This is because the invention and evolution of software has required the very deep investigation of the issues that are being investigated in this chapter – the mechanism for the interrelationships among the highly organized components of deeply nested hierarchies. In the case of software, the components of these “deeply nested hierarchies” are the various executable module types: functions, subroutines, objects, etc.

The software engineering world has been forced to articulate and solve the complex problems around these issues already. Biology – which is even more complex – has not been forced into being as insightful or as inventive as has software engineering around these nested systems issues. This may be because biologists do not *design* or *build* the systems that they study, and have not been forced to the same depth of understanding of nested systems as have software engineers.

However, as we look below at composite biological systems, it should become evident that the view of the software engineering discipline of the nested systems world sheds valuable insight about the mechanisms involved.

To follow this example, it will not be necessary to know object-oriented programming. However, it *will* be necessary to understand programming. The essential requirement is to understand that programming languages have the ability to nest modules inside of modules for some number of levels of

organization. One essential difference between object-oriented languages and their procedural predecessors is the support for deep levels of nesting.

Since we are dealing with an imaginary programming language in this example, we can invent an imaginary set of nesting structures for the language. In order to keep this example organized, we shall use different named nesting structures for each level of organization of the software being developed.

The nesting structures that we shall invent for this example are named, from the bottom level up, “statements”, “functions”, “objects”, “packages” and “subsystems”. So, statements are the components of functions; functions are the components of objects, objects are the components of packages and packages are the components of subsystems.

We’ll concentrate on program behavior (execution) rather than data in this example. And, we shall be borrowing various programming mechanisms and constructs from a number of programming languages.

If you are familiar with computer programming, it should be easy to see that a set of statements can, under certain conditions, form a smallest executable unit. In computer programming, these units are referred to by a number of specific names. Without dwelling on their distinctions, we shall use the term *function* to refer to these minimal executable units. Among the required conditions is that they must be organized into a specific set of relationships that define a linear order in which these statements will be executed.

Thus, a *function* has a population and an organization. Its population is its set of statements. Its organization is imposed by the order in which they are executed. This order is determined by the order in which the statements occur in the program as well as the conditional and transfer-of-control statements. True to our nomenclature, this order can be articulated using our “set of sets of clusters” definition of a *system organization*.

Thus, the statements (the population) involved in the *function* as well as their *organization* together constitute a *system* named a “function”. So, in this example “function” is the lowest level of system in our nesting hierarchy. And its components – which are not systems - are statements.

However, functions can be treated as components that can be organized into “classes”, at design time, which can be “instantiated” as multiple “objects” at run time. Since we are talking about program execution, we shall use “objects” for this level of organizational structure, and ignore classes. In this case the relationships between objects is defined by their component functions cross-references, or their “calls to” one another. That is, whenever a function of an object invokes, or “calls” a function of another object, then we say that the two objects are related.

Of course, one function within an object can call another function in the same object. In that case, the two functions become related. And, the set of all of these pairs of related functions within the same object are, in fact, the *organization* of that object.

Thus, “function invocation” becomes the relationship among these components (functions) that constitutes their *organizing principle*. That is, the population of an object is its set of functions. (The functions are the components of the object.) Moreover, these invocation relationships between these functions form the *organization* of the object.

At this point, our nested hierarchy has two levels of organization: at level 1 we have “functions” and at level 2 we have “objects”. Let go another level higher. Our third level will be a nesting of objects. That is, it will be a collection of objects that are somehow related. Lets call this entity at the third level of organization a “package”, as is done in the Java programming language. Thus a package – at run time – consists of a set of objects.

The question is, what is the organization relationship among “objects” in a package. The answer is – for a second time – function invocation. That is, at level 3 of our organization hierarchy, a level 1 pairing mechanism (function invocation) is again used.

That is, we have not defined a separate mechanism at the package level of organization for the components (objects) to use to form relationships (pairs). Rather, we have re-used, or inherited, the relationship mechanism that was defined at level 2.

Notice, “Objects do not really invoke objects”. Rather, it is a *function* of an object that invokes *functions* of other objects. However, whenever that occurs, we permit ourselves to say, “an object invoked an object” – even though what really occurred is that “a function of an object invoked a function of another object”. In fact, we shall define our organizing relationship for objects by saying that “Objects are related (invoke each other) whenever a function of one invokes a function of the other”.

Thus, we have defined the relationships at one level of organization indirectly by reusing the relationship that already exists at the next lower level of organization.

Now, lets exemplify another aspect that we shall provide in our general modeling apparatus of nested systems: *interfaces*. Another fact about an object-oriented programming language like Java or C++ is that, at the object level of organization (level 2 in our example), some of the functions can be disallowed from being invoked from the functions of other objects, while others of its functions can be “exposed” to (invoked from) the functions of other objects.

This fact portrays the notion of “public” and “private” functions. In object-oriented programming, for any specific object, we can refer to the set of all of the public functions of an object as the “interface” of that object. (This is also referred to as the “application-programming interface” to the object and its class, or API.)

We shall generalize this notion of “interface” for Organodynamics. We shall define the “interface” of a component system to be a subset of its components

that are available for being related to (pairing with) components of other component systems.

Lets take our example to one more level of organization of nested systems by forming a system whose components are packages. There is no standard name for this level of organization in programming (except that, for example, Java provides for packages within packages.) However, for the sake of our example, we'll use the term "subsystem" for our new (fourth) level of organization. So, the components of a subsystem are packages in our example. For a subsystem to actually be an Organodynamic system, however, its components must be able to organize through some kind of relationship into an *organization*.

We have two choices, as always. We can either define a new way that these components (packages) can relate to each other. Or we can do what we have done with objects and packages earlier: we can reuse, or inherit, the type of relationship that was defined at a lower level of organization to indirectly define relationships between the packages in our subsystem.

We'll take the later course by reusing the "function invocation" relationship between two objects. So far we have defined two functions to be related if either invokes the other. We have also defined two objects to be related if any of the functions of one invokes the functions of another. Thus, we shall define any two packages as being related if any of functions of any of their objects invoke any of the functions of the objects of the other.

However, for subsystems, we shall define an addition organizing principle: *author*. If the same independent software developer authors two packages of a subsystem, then we will pronounce the packages related by authorship. Thus, subsystems have one organizing relationship that is inherited from a lower level of organization as well as another organization relationship that is defined its own level.

So, in this example of a software application, we have four levels of organization. The lowest level, function, defines its on organizing principle: statement order. The second level of organization, object, also defines its own organization principle: function invocation. However, the remaining level of organization, package and subsystem, inherit their organizing principle from the object level. The package level defines no further organizing principle. But the subsystem level does define its own separate organizing principle as well as inheriting the function-oriented one.

A Corporate Institution

Corporations can be modeled as a hierarchy of nested systems. For example, a corporation may have companies that have divisions that have departments that have employees.

The notion of "component relationship" must apply at all levels of organization. That is, the components at any level of organization must have a way of relating to each other in order to form an organization – which then becomes a component at the next higher level of organization.

For example, a department has employee components. Their working relationships form the organization that is the department. Departments of a division have well-defined relationships. These are defined by engagement protocols between individual employees of the two departments. Companies have divisions that have well-defined relationships. These are defined by engagement protocols between individual employees of departments of the companies. Thus, at each level above the department level, the nesting system reuses the native organizing relationship of the department level to indirectly define its organizing relationship – in the same fashion as we saw in the software application example above.

The notion of “interfaces” applies in this example as well – and it can apply at multiples of these levels of organization. For example, at the division level, an interface can establish a protocol whereby the VP of one division initially calls (relates to) the VP of another division. Their interaction can subsequently delegate an interaction (engagement) between two departments, one from each division. These types of interfaces can be defined at any of the levels of organization of the corporation.

Notice that the three stages of relating apply. Firstly, organizations must exist and have components. Secondly, as an enabler for other components to relate to a component, certain of its components must be exposed and be relatable via a organizing relationship. Finally, the event of another components becoming related to the first component through its exposed component must occur.

A DNA Molecule

So far in this chapter, we have presented examples of nested systems with various numbers of organizing relationships, which were defined at various levels of their organizations.

Typically, a particular level of organization defines its own organizing relationship, and then that relationship is reused indirectly at higher levels. A good example of this was how we defined packages being related and subsystems being related if they contained, at some level below, functions that were related via function invocation.

We saw an example – the software application - with only one organizing relationship (the function call) that was defined at its lowest level of organization and then reused indirectly at its other level. We also saw other examples with two definitions of component relationship, where one of them was only applied at its native level, and the other was reused at higher levels.

We are now going to see an example wherein there is a level of organization within the hierarchy that simultaneously utilizes two relationships to bind (relate, pair) its components. This example involves a DNA molecule (for any species).

There are many ways that one can use the Organodynamic approach to nested system modeling to develop a DNA model. This depends on which parts of the DNA macromolecule are decided by the modeler to be nested systems within

which other parts of the molecule. Many of these choices are reasonable. The “best choice” depends on what aspects of the molecule that the modeler wants to emphasize.

In our case, we shall decide that a DNA molecule, as a system, contains two components – each of which is one of the strands of the double helix. Each strand, then, has some large number of nucleotides as its components. And, each nucleotide has three components – its base, the sugar and the phosphorous molecules.

Thus, our DNA model has three levels of organization: DNA, strand, and nucleotide. This time, we stopping our modeling of subsumed systems at just three levels or organization, and are not considering the three components of a nucleotide (the phosphate, the sugar and the base molecules) to be subsumed systems – even though we could because each is a molecule. We could take this down to many more levels, but shall not at this time.

The organization of each of these levels of organization will not be defined, right now – mainly because we want to emphasize the relationships involved between the two strands here. However, each such organization will represent the way that the constituent molecules are bonded. For example, the nucleotide layer will relate the three constituent molecules in a linear fashion, with the base related to the sugar, which is related to the phosphorous.

Lets return to the highest level – where the system is the DNA molecule, and its components are the two DNA strands. We shall describe the bonding relationship that occurs between these two strands as two distinct mechanisms – both of which are operational, and need to be accounted for. One of them is the hydrogen bonding that occurs between the bases of the individual nucleotides. The second is the fact that the two strands are physically intertwined, and cannot be separated without being “unwound”. Lets look at each of these in turn.

First, we'll consider the hydrogen bonds between the bases. These are weak bonds that must be temporarily broken by the enzyme molecule RNA polymerase when it is decoding a DNA sequence.

Right away we have an apparent modeling difficulty with these bonds between the bases. Because of the way we have decided to nest these “systems” within each other (and that was our modeling decision), the bases are very remote from each other within our model, even though they are physically near each other in reality, and are chemically bonded via a hydrogen bond.

However, we really do not have a modeling difficulty after all, because of the following. These two bases can be directly related to each other because a molecule of one of the bases can be exposed as part of the *interface* of that base. This allows a molecular component of the other base to form a relationship to the exposed component. In fact, the two bases can be bi-directionally related via the same mechanism in the other direction. Thus the two bases are indirectly related.

We can define the two nucleotides of which these two bases are components to be indirectly related through the relationships of their bases – as we did in all the other examples above. And, taking this one more step, we can define the two strands to be indirectly related as well. Thus, we have managed to define a relationship between the two strands at the DNA level of organization. And, of course, we can repeat this thinking, as many times as there are nucleotides in a strand – to form millions of these relationships between the two strands. And, all of these relationships are indirect – being reused from multiple lower levels of organization.

But, there is yet a second way that we can define a relationship between these two strands at the DNA level of organization. As we said, these two strands are “physically related” by virtue of the fact that they are wound together, at the DNA level of organization, as a double helix configuration. This makes them bound together at this level. They even have to be separated at this level (“unwound”) by some mechanism in order to gain access to a lower level of organization, where the genetic information resides. So, this physical winding of the two strands of the double helix is clearly a second, and direct, bonding mechanism of the DNA level.

Thus, clearly the DNA system containing these two strands is “highly organized” since its two components are so strongly related – by both a direct and multiple indirect relationships.

Constraints and Liberties on Organizations in Composite Systems

Analyzing the examples above, we can abstract some common patterns, which we shall discuss in this section.

Two Opposing Organizational Forces

In the examples above, we have seen two opposing forces at work: 1) a tendency to proliferate the number of possible system organizations as systems become more deeply nested, and 2) the tendency of only a restricted number of the organizations that are possible to actually be available for realization.

The first tendency is due to the large number of ways that a nested set of underlying populations can be organized; while the second tendency is due to the fact that the number of ways that sibling component systems relate to each other in real applications depends upon the “cross relationships” that their inner components reach out across systems to relate to each other.

In the end, this second tendency often limits the number of actual ways that components organize into composite systems to a much more limited set of organizations than might be expected.

Inter- and Intra- System Constraints

As we have already pointed out, any composite system comes into existence via the *Integrate* organodynamic transform. This occurs when an Integrate

transform takes a set of existing systems as its “inputs” and creates (“outputs”) a single composite system, with the “input” systems as its components, as a result.

This means that the set of component systems pre-exist the composite system. What the examples above are highlighting is that, in real composite system applications, there are usually “forces at work” – actually *dependency relationships* at work – that create a *propensity* (with some probability) for this *Integrate* transform to occur.

In other words, from the perspective of the component systems, the composite system does not exist at the time that it is being created via an Integrate transform. Only the component systems “exist” at that time. Prior to the performance of the Integrate transform, only (what are to become) the component systems and their organizations, the relationships among them, exists.

However, in real applications, once the composite system has been created, it is capable of projecting its own forces upon its components systems. And the composite system may impose constraints on the changes of organization that it allows its components to make.

For example, in our molecular example, a molecule exerts forces on its individual atoms, just as the forces of the atoms influence the stability of the molecule. The two levels of organization are interdependent.

Thus, the constraints in relationships between sibling component systems are initially imposed by the components systems themselves. However, once a set of sibling systems are integrated via an Integrate transform, then the resulting composite system can further impose constraints on the organization of its components systems.

Stochastic Dependencies and Composite System Organization

Consider two sibling system organizations that “enjoy a relationship” with each other – a relationship that promotes the expectation that they will be inter-related.

In Organodynamic terminology, this means that it is very likely that there will exist an *organization* of the underlying population of component systems in which the duple containing these two components will appear within that organization.

Thus, these dependency relationships between sibling component systems can be articulated in terms of the probabilities of *system organizations* that contain duples of those systems. In other words, these dependency relationships can be modeled by probability distributions whose sample space is the set of all possible system organizations of the underlying population.

More specifically, we can develop these probability distributions as conditional probability distributions, where the probabilities that certain system

organizations will occur at the next time step is conditioned on the defined relatedness of the component systems.

There are many ways that these conditional probability distributions can be developed. For example, they can be as described above, or they can be joined in joint probability distributions with other conditions.

The point being made is that conditional probability can form the backbone of a general approach for model the conditional relationships among sibling component systems and their relational behavior. This direction of inquiry has many possibilities, but lies beyond the scope of the present text. Therefore we leave this issue for further research.

The Semantics of Component Relationships

In the previous chapter on the construction of composite systems and nested system hierarchies, much was made of the fact that before components can be paired into duples in order to form organizations, the components must be “related”.

It was further described that these relationships initially come from the application domain. But it was further explained that the Organodynamic framework provides mechanisms for propagating these relationships up composite system hierarchies.

The semantics of these component relationships are the essence of composite systems because they determine how composite system organizations can be formed and express the nature of the systems that they model. Plus, as can be seen by looking at our definition of composite system, these relationships permeate every element of a nested system hierarchy.

Thus, there is a lot to say about component relationships in Organodynamics, what they mean and how they are created or reused. And we have reserved this chapter to contemplate their depths.

The Construction of Component Relationships

We have defined a procedure for developing a composite system from a collection of existing system that will become the component systems the composite system being created. In fact, the procedure we defined is precisely the algorithm used by the *Integrate* organodynamic transform to create a composite system.

One can study that procedure and see that it describes two levels of organization – the composite system level and the component system level. The composite system level has exactly one system – the composite system. While the component system level has one or more systems. It generally has multiple systems.

Each system at each of the two levels has two parts: 1) the population and 2) the organization. Altogether, we have four types of *parts* to consider across these two levels:

- The composite system's *population*
- The composite system's *organization*
- The component systems' *populations* and
- The component systems' *organizations*.

However, left out of this description are the details of how the various relationships among these four types of entities shall be formed in Organodynamics.

But, for the reasons described so far in this chapter, these relationships are the heart of the matter. And how Organodynamics has chosen to define and represent these relationships is a key characteristic of the framework.

Choices about these relationships in Organodynamics and how they play into the construction of the nesting and component systems, and their populations and organizations must be defined. The following sections develop a framework for these interrelationships, which are the essence of system nesting in Organodynamics.

System Integration

The essence of what is going on within composite, or nested, systems is to observe what is going on when a composite system is created from two or more other systems via an Integrate transform.

The first thing to notice is that more has occurred than simply “combining two or more systems into a new one”. The most significant occurrence is that a new system at a new level of organization – or level of abstraction – has been created. A second thing to notice is that the initial systems that were combined continue to exist.

A phrase that comes to mind that describes this kind of occurrence is *system integration* – and thus the choice of the name of the *Integrate* organodynamic transform.

Our usage here of the phrase *system integration* is also consistent with the connotations within the computer industry, where the phrase implies much more than simply “combining two computer systems”.

Implicit in this usage is, for example, that the original systems remain untouched and can be used as before the integration. Also implicit is that a higher-order system has been created as well – the integrated system.

Also implicit in this connotation is that the measure of certain qualities of the integrated system is greater than the sum of the measures of the qualities of the systems from which it was integrated.

Consequently, we find that the *Integrate* transform lies at the heart of the formation of system nesting hierarchies.

The Mechanisms of System Integration

The formation of a composite system via system integration is a very rich affair that involves far more than simply “combining” two systems. Rather, system integration involves the creation of a new composite system from a collection of component systems. Moreover, it involves the construction of a set of relationships – a system organization - that is defined between the components of each of the composite systems.

In order for two or more component systems to integrate into a composite one, they must be able to form into an *organization* of the composite systems. And, for this to occur, as we have already seen, certain pairs of these component systems must be related to each other. In other words, there must be a definition of a *relationship* between the component systems.

We are going to introduce three ways that relationships between components systems tend to form within typical application systems that are reasonably seen as nested systems. We shall give names to these three ways, and provide procedures for forming them with the Organodynamic framework.

Domain-defined Component Relationships

The first of these ways for relationships among the composite systems to come into existence is for them to be defined directly by the application domain at the level of the composite systems. By “application domain” we are referring to the fact that Organodynamics is a modeling system that is intended to be applied as a model of certain “real situations”, which we shall refer to as the *modeling domain*, or simply the *domain*.

When using Organodynamics to model a domain, the primitive notion of relationships among system components is the responsibility of the modeling domain. That is, the domain dictates component relationships. As a theoretical framework, Organodynamics enforces these relationship stipulations. It also adds value to them by providing a framework mechanism to propagate these domain-defined relationships up a nested systems hierarchy.

For example, physical chemistry is an “application domain”. This domain dictates that molecules can become related electromagnetically by a covalent bond. Thus, covalent bonds are a type of domain-defined relationship in the physical chemistry domain. Ionic bonds are also a domain-defined relationship, as are Van der Waals forces.

Propagated Component Relationships

A second way that components, or component systems, can be related to each other is through “inheritance”. That is, two systems can be come “related” just because a component of one happens to be related to a component of the

other. Perhaps these components are related via some domain-defined relationship.

An example of this is found in our simple molecular system above. We had two atoms which each had components that were attracted to each other through the electromagnetic force. Specifically, this was an electron of one atom and the nucleus of the other. This relationship caused the two atoms to inherit this relationship in what we express as a covalent bond.

Moreover, this relationship was even inherited the next level up as follows. Each of these two atoms is a component of a molecule. And, these two molecules also became “bonded” as components of a larger molecule. Thus the same relationship is “inherited” up another level of organization.

Organodynamics refers to this second type of relationship as a *propagated* relationship.

We also saw a propagated relationship at work in the software example. A function of one object invokes a function of another object. Ultimately, the relationship originates between the two functions. It is the two functions that have the capability of being related via function invocation. But we propagate that relationship that is defined between functions upward to the object level – and say, also, that the objects are related.

The framework is not in a position to provide domain-defined relationships. That is the province of the application domain. In fact, the application domains *must* define domain-defined relationships. Otherwise, the framework has nothing to work with. It all begins with domain-defined relationships. The other two types of component relationships are merely defined in terms of domain-defined relationships – or of each other.

However, the framework *is* in a position to support propagated relationships. It can offer the service of propagating a domain-defined relationship upward within a nested system hierarchy. In fact, a propagated relationship can propagate any of the three kinds of relationships up a nested systems hierarchy.

Accessibility-enhance Component Relationships

As we have seen by looking at propagated relationships, higher-level components can “inherit” relationships that exist between components that are embedded within each of the higher-level components.

Sometimes this situation brings up the question “How do two embedded components ‘find out about’ each other, given that they are embedded in different higher level systems?”

That is, sometimes it is not possible for two composite systems to inherit relationships from their embedded members, because they cannot gain access to them. In these cases, some addition mechanisms are required to “expose”

those embedded members to these composite systems to enable them to “inherit” them.

This problem arises in many, if not most, nested systems – and therefore has been solved by them. However, the software industry was forced to recognize this problem, articulate it and engineer a rigorous solution. In doing so, this industry provided a set of ideas and terminology to articulate it.

Specifically, these ideas were initially identified, defined and implemented in the idea of the *remote procedure call* in the mid 1980s. Since then, the technology has ramified into a mature and complex solution that is capable of supporting the idea of distributed application programs and forms the backbone of what today is referred to as “enterprise computing”.

The basic problem to be solved is twofold: 1) generally, there are too many systems that may want to be related to each other through one of their embedded components, and 2) generally, an embedded component of one system, while “known to” its sibling components, is not “known to” embedded components of other system.

The framework offers support for two component systems to become related to each other through their own embedded components via mechanism – developed below – called *accessibility-enhanced component relationships*.

Summary of Component Relationship Types

Three Types of Component Relationships

<u>TYPE</u>	<u>DESCRIPTION</u>	<u>The Software Application Example</u>
Domain-defined	These relationships are defined by the application domain between the components of a specific system at a specific level of organization of the nested system hierarchy.	Function invocation.
Propagated	These relationships are defined between composite systems by “inheriting” them from their component systems.	Defining “package systems” as “being related” whenever an object of one of the packages is related to an object of the other package.
Accessibility-enhanced	<p>A propagated relationship between two systems wherein an embedded component of the second system requires “extra equipment” in order to be made known to – or “exposed” – to an embedded component of the first system.</p> <p>At this point, then, the two components can be “related” and form a duple within an organization of the first system.</p> <p>The rules of composite relationships then allow the two systems to become related at a higher level of organization.</p>	<p>Two “object” systems within the same package “system” can be defined as “being related” if a function of one invokes a function of the other. This is a case of “relationship by function call” being inherited up the hierarchy from the “object” level to the “package” level.</p> <p>However, for this to occur, any functions that are to be called from outside of their own object must</p>

		be especially enhanced, or marked, so as to be accessible, or “exposed”, to such calls.
--	--	---

Each of these types of component relationships will be detailed in the sections below.

Defining Domain-defined Relationships

All component relationship types originate with a relationship between components that is defined by the target domain application of a model. In other words, component relationships ultimately have the semantics of the modeling target domain.

Our software example used “function invocation” as its domain-defined relationship. Our molecular examples used a variety of relationships between molecular, atomic or subatomic entities as their domain-defined component relationships.

Domain-defined relationships can be defined for any components at any level of a nested hierarchy. They often, however, are defined for lower levels; and then the other two types of relationships are defined at higher levels by “inheriting” from these domain-defined relationships at a lower level.

In fact, the other two types of relationships defined below must build upon a domain-defined relationship in order to “get started”. Both of them initially “inherit” from a domain-defined relationship at a level below them in a nesting hierarchy.

Each of the other two types (categorized as *framework-defined relationships*) inherits from one of the three types of relationships at the level of organization immediately below them in the hierarchy. Thus, the bottom level of one of these “inheritance chains” must be a domain-defined relationship. The levels above it in such a chain can be a mixture of the other two types.

Defining Propagated Component Relationships

The Conditions Required for Propagated Relationships

For two systems to be related via a propagated relationship means that each has an embedded component such that these two components are related to each other – even though they are components of two different systems.

This framework will assume that such a relationship is already defined among the components of each of the individual systems.

For example, in the software application example, there exists a *function call* relationship between the functions of an object. That is, if there are two or more functions that are components of an object system, then either one of them can call the other. At the object level, it is desirable to permit a function of one object to call a function of the other object. This idea entails taking a relationship that exists *within* the functions of one object, and that also exists *within* the functions of a second object, and extending it so that it also applies to a pair of functions where one is a component in the first object and the other is a component in the second object.

Therefore, the essential requirement to be able to define propagated relationships is that the component systems to be related through it share a common relationship among their own components.

Formally Defining Propagated Relationships

If two systems have the same relationship defined on their components, and if one of those systems has a component that is related via this relationship to a component of the other system, then this is sufficient to define a relationship between the two systems. Such a relationship is called a *propagated component relationship*.

Defining Accessibility-enhanced Component Relationships

Suppose we have two component systems whose embedded components share the same relationships internally. For example, suppose that we have a composite system B that has two component systems A_1 and A_2 . Suppose further that the components of A_1 have a specific relationship defined between them, and also that the components of A_2 have the same relationship defined between them.

Then it is semantically sensible for a component of A_1 to enjoy this same relationship with a component of A_2 .

This situation is precisely what we defined above a *propagated component relationship*. That is, composite system B *inherited* the relationship that was shared by both A_1 internally and by A_2 internally. Thus, the relationship that was shared by both A_1 and A_2 was *propagated* upwards to B.

However, sometimes more intellectual equipment is required to make this happen. Specifically, even though these relationships already exist in A_1 and A_2 , it happens that the components of A_1 and A_2 are not capable of becoming related via this relationship – of *inheriting* this relationship, perhaps because they are not, in some application sense, *accessible to each other*. (To use an anthropomorphic analogy, we would say that they do not *know about* each other.)

The purpose of the *accessibility-enhanced component relationship* is to provide this “extra equipment” to propagated relationships that need it in order to gain the accessibility that they need but do not otherwise have.

Observing a System with an Accessibility-enhanced Relationship

In our software application example, we have a quintessential example of an accessibility-enhanced relationship. Lets study this example a bit to see what light it can shed on what this “extra equipment “ is that provided the accessibility needed.

In the simplest case of our software example, we have *package systems* whose component systems are *object systems* whose component systems are *function systems*.

All of the *functions* of the same object are automatically *accessible* to each other – that is, they are capable of *calling* each other. This fact is a *domain-defined relationship*. In the software application domain, it is the case that all function with the same object are able to *call* each other.

We would like to propagate this *function call* relationship upward to the next level of the composite systems hierarchy and allow *object systems* to inherit it. This would enable the *functions* of one *object system* to call a *function* of another object system. When this capability occurs, we say that the two object systems inherit the *function call* relationship.

However, in the software application domain, it is not automatic that a function from one object can call a function from another object – even though it *is* automatic that two functions within the same object can call each other.

In fact, the software application domain requires that some *extra equipment* be established that enables this capability.

The reasons for requiring this extra equipment are several – including 1) that there are so many objects systems in the software domain that automatically maintaining knowledge of all of them to all of them would be prohibitive, and 2) security.

The software domain provides two specific “pieces of equipment” to enable the functions of some objects to call some of the functions of other objects. (There are more, but for the sake of simplicity, we shall keep it to two.)

The first of these mechanisms provides for an object to *expose* a limited set of its functions. Thus, the functions of other objects will be able to call at most the functions in the limited set. For a given object, limited set is referred to as its *interface* – to borrow the term used for this concept in object-oriented programming.

The second of these mechanism is a *registry* that can be used by a “calling object” to look up the interface of the object whose functions are to be called. It is the responsibility of any object that wants to expose its functions to deposit its interface into this registry.

The Organodynamics framework takes a particular approach to the use of the interface of a target composite system, the *related system*, by a *relating system* that wants to be related to it. In this approach, a *proxy* system is created to

represent the *relating system*. The population of this proxy system contains all of the components of the *relating system*, plus the exposed components of the *related system*. It is able to obtain the exposed components of the target system from its interface, which it obtains from the registry.

By imitating these ideas from the software application domain, let's now define these ideas more formally; and then provide a simple set-theoretic example of such a relationship. The example will not be as intuitive as our software application, but it will be simpler so that we can develop it using our nomenclature that is based on set theory notation.

Introducing the Enhancement Proxy

In the previous subsection we used our familiar software application to introduce the strategy we shall use to define the accessibility-enhanced component relationship.

We showed how we would enhance one system – the “relating system” – in order for it to be able to establish an accessibility-enhanced relationship with a second system – the related system.

This approach creates a new system that is to be used as a proxy for the relating system whenever organizations are formed between the relating and the related systems. This new system is called the *enhancement proxy* for the relating system.

This enhancement proxy is a system whose population contains all of the components of the relating system, plus any components of the related system with which some components of the relating system will form relationships. Of course these latter components must be in the interface set of the related system.

Then, the organization of the new system will contain all of the duples of the clusters of the organization of the relating system, plus any new duples that relate the components for the relating system to components of the related system.

It can be seen that this new organization is a well-formed organization of the components of the population of this proxy. In fact, the reason for all of these elaborate steps is to ensure that this resulting proxy organization is well formed – meaning that all of the elements of its duples are members of its population, and that they are related.

Thus, this newly created enhancement proxy has a population that includes all of the components of the relating system plus any components of the related system that are needed. And the enhancement proxy also has a “legal” organization that relates the appropriate components, regardless of whether they are from the relating system only, or from both the relating and the related systems when appropriate.

We shall now more formally define this enhancement proxy by defining both its population and its organization.

We shall construct the *enhancement proxy* $A(B)$ of system A (the relating system) with respect to system B (related system),

$$A(B) = (P_A(B); O_A(B))$$

Where

$P_A(B)$ is the population of the *enhancement proxy* of system A with respect to system B , and

$O_A(B)$ is the population of the *enhancement proxy* of system A with respect to system B .

Let us now define these this population and organization of the enhancement proxy.

The Population of an Enhancement Proxy

The idea when creating the population of an enhancement proxy for a component system is to add the interface set of the related system to the population of the relating system.

Definition: Population of an Enhancement Proxy: Let $A = (C_A; O_A)$ and $B = (C_B; O_B)$ be two systems. Let B_i be the interface of system B . Then define the *population of the enhancement proxy of system A with respect to system B* as

$$C_A(B) = C_A \cup B_i .$$

The Organization of an Enhancement Proxy

The idea when creating the organization of an enhancement proxy for a component system is to add new duples to the organization of the relating system. These new duples will relate certain components of the relating system to components of the relating system.

This enhanced organization will be well formed, because all the components of all of its duples are components of the population of this proxy.

Definition: Organization of an Enhancement Proxy: Let $A = (C_A; O_A)$ and $B = (C_B; O_B)$ be two systems. Let $C_A(B)$ be the population of an Enhancement Proxy C_A with respect to a system B . Then define

$O_A(B) =$ Any organization consisting of clusters of duples whose components are elements of $C_A(B)$.

$O_A(B)$ is called the *organization of the enhancement proxy of system A with respect to system B* .

Thus, the *organization of the enhancement proxy* of a system is a set of clusters whose wherein some of the duples are those of the organization of the initial population of the system, and the rest of the duples have one member that is a component of the initial population, and the other member is an interface member of system B.

The Enhancement Proxy of a Component System

Thus far, we have started with a system named $A = (C_A; O_A)$. We then took its population C_A and created from it a new population named *the population of the enhancement proxy of system A with respect to system B*. We symbolized it with $C_A(B)$. This proximation consists of all of the components of C_A plus all of the interface components of B.

Secondly, we took this newly created population - $C_A(B)$, *the population of the enhancement proxy of system A with respect to system B*, and we constructed a new organization from it. We named it *the organization of the enhancement proxy of system A with respect to system B*. We symbolized it with $O_A(B)$.

Thus, we now have both the population and an organization for this new proxy system. We shall name it *the proximation of system A with respect to system B*. We symbolized it with $A(B)$.

Thus, the *enhancement proxy of system A with respect to system B*,

$$A(B) = (O_A(B); O_A(B))$$

Example Enhancement Proxy

This is getting pretty complex, so we had better look at an example of $A(B)$, the enhancement proxy of a Component System.

Again, I apologize for the complexity. But it is necessary, since we are modeling complex systems. And our abstractions here are considerably less complex than the real systems that they will model! So, we need Organodynamics to be as simple as possible – but still not lose the essential complexity of complex systems.

Lets return again to our software system example to provide an example of the *enhancement proxy*, and its attendant *population* and *organization*.

Recall that, within the nesting hierarchy of our example software application there are systems of type *object* whose components are systems of type *function*.

Lets say that there are three object systems B_1 , B_2 and B_3 . Each of these have three functions, which are named C_{11} , C_{12} and so on – where in the name C_{nm} , n specifies which object the function is a component of, and m signifies a unique index for the functions within its object system.

Accordingly, these three object systems are defined thusly:

$$\begin{aligned}
 B_1 &= (C_{B1}; O_{B1}) = (\{ C_{11}, C_{12}, C_{13} \}; \\
 &\quad \{ \{ [C_{11}, C_{12}], [C_{11}, C_{13}] \}, \{ [C_{12}, C_{13}] \} \}) \\
 B_2 &= (C_{B2}; O_{B2}) = (\{ C_{21}, C_{22}, C_{23} \}; \\
 &\quad \{ \{ [C_{21}, C_{23}] \}, \{ [C_{22}, C_{23}] \} \}) \\
 B_3 &= (C_{B3}; O_{B3}) = (\{ C_{31}, C_{32}, C_{33} \}; \{ \{ \} \})
 \end{aligned}$$

The interfaces of these three systems are:

$$\begin{aligned}
 \text{Interface of } B_1 &= \{ C_{11} \} \\
 \text{Interface of } B_2 &= \{ C_{21}, C_{22} \} \\
 \text{Interface of } B_3 &= \{ C_{31}, C_{32}, C_{33} \}
 \end{aligned}$$

From this information, we can develop an example *population proxy*, say, of system B_1 with respect to system B_2 .

We shall first develop the *population proxy* C_{B1} with respect to system B_2 , - symbolized " $C_{B1}(B_2)$ ". After that, we shall develop the *organization proxy* O_{B1} with respect to system B_2 - symbolized " $O_{B1}(B_2)$ ". Then, having both the population and the organization, we have the system proximation $B_1(B_2)$.

First, $C_{B1}(B_2)$ will consist of the union of C_{B1} (the components of system B_1) with the interface of B_2 .

$$\text{Thus, } C_{B1}(B_2) = \{ C_{11}, C_{12}, C_{13} \} \cup \{ C_{21}, C_{22} \} = \{ C_{11}, C_{12}, C_{13}, C_{21}, C_{22} \}.$$

Next we must construct $O_{B1}(B_2)$, an *organization proxy* O_{B1} with respect to system B_2 . $O_{B1}(B_2)$ can be any organization of the components of $C_{B1}(B_2)$.

Thus, for example, $O_{B1}(B_2)$ might be

$$\{ \{ (C_{11}, C_{12}), (C_{12}, C_{13}) \}, \{ (C_{11}, C_{13}) \} \}$$

Further Component Relationships Characterizations

For all of these three types of component relationship types, there are further characterizations we can make about individual relationships. Specifically, the relationships between two components can be symmetric or non-symmetric.

Additionally, it is possible to characterize relative strengths of various component relationships.

These characterizations are explored further in this section.

Weak system component relationship

Whenever one system is related to another system by a specific relationship, but the other system is not related to the first through that relationship, then the relationship is said to be a *weak component relationship*.

For example, one function of an object invokes a function of another object, but the second function does not invoke the first. This is a weak component relationship.

Weak relationship is non-symmetric.

Inspection of the duples of the organization of a composite system can reveal a weak relationship whenever a duple is found whose inverse is not also a member of a cluster of the same organization.

Doubly weak component relationship:

Whenever one system is related to another system by a specific relationship, and the other system is also related to the first through that relationship, but not through the inverse component duple, then the relationship is said to be a *doubly weak component relationship*.

For example, one function of an object invokes a function of another object, and one function of the second object invokes a function of the first object, but the function of the first object that is invoked by the second object is not the same function of the first object that did the initial invoking.

The doubly weak relationship is symmetric.

Inspection of the duples of the organization of a composite system can reveal a doubly weak relationship whenever a duple is found that pairs a component of the first system with one of the second, and another duple is found that pairs a component of the second system with a component of the first; but the two duples are not inverses of each other.

Strong system component relationship

Whenever one system is related to another system by a specific relationship, and the other system is also related to the first through that relationship, and the relationship is through the inverse of the first component pair, then the relationship is said to be a *strong component relationship*.

For example, one function of an object invokes a function of another object, and same function of the second object invokes the first function of the first object, then the two objects are strongly related. (In this software example, care must be taken that an endless loop is not developed.)

The strong relationship is symmetric.

Inspection of the duples of the organization of a composite system can reveal a strong relationship whenever a duple is found that pairs a component of the first system with one of the second, and another duple is found that is the inverse of the first duple.

Related systems components

If two system components are either strongly, weakly or doubly weakly related, then we say they are *related*. System component relationship is symmetric.

Unrelated components

If two systems are not related via any of the above three relationship types, then we say that they are *unrelated*.

Component Relationships in the Uncertainty Model Gradient

In this chapter we have described how membership in the sample space of realizable system organizations can be constrained by the interrelationships between sibling components (of the same population) and between a composite system and its component systems.

These constraints have led us to consider how the nomenclature for composite systems that we developed in the previous chapter reveals and articulates these constraints.

We now turn to the question of how these considerations affect the three model types of the uncertainty model gradient. That is, how do these considerations show up in the three model types? We shall describe these effect in this section.

Recall that the three model types of the uncertainty model gradient are:

1. The Markov model
2. The unconditional stochastic model
3. The realized deterministic model

Each of these three model types is a stochastic process - a sequence of mathematical entities that are associated with time steps, also called a time series.

Being stochastic processes, the mathematical entities of each time step of each model type involve, in some way, one or two probability distributions, depending on model type. In other words, each of these model types can be seen, in some way, as a sequence of probability distributions.

Of course, any probability distribution has a *sample space* of *sample points* (or *events*), each of which is assigned a probability value. In all three of these model types, the *sample points* are defined in Organodynamics as *system states*. Specifically, system states in Organodynamics are defined as *system organizations*.

The considerations of this chapter only affect the sample spaces of each of the three model types that we discussed in the previous chapter by 1) possibly eliminating some of them, and by 2) altering some of their probability values.

Some of the sample points of the three model types that we described in the previous chapter may have been eliminated by the considerations of the present chapter. This can occur for sample points (composite system organizations) that the composite relationship restraints in question may have ruled out. Another way this “elimination” can be handled, however, is simply by changing the probability values for these *organizations* to zero (and raising the values of others so that the probability values for the entire sample space sums to one).

In general, the effect that the component dependency conditions have on the existing probability distribution as we left it in the last chapter is that it alters the probabilities of the constrained outcomes (system organizations) so that they are lowered, and raise the probability values of some of the other.

Lets now briefly visit this issue for each of the three model types.

Incremental Example Model Development

<>

Progress Check

In Part II, Organodynamics, we are gradually and incrementally developing a mathematical framework for modeling living and lifelike systems according to the seven organizing principles of OCS. Once completed, this modeling framework will achieve the form of an abstract but highly flexible mathematical structure that we are calling an *Organodynamic web*.

We are using the example of a hobbyist’s toolkit for building electronics as an analogy to Organodynamics. Both are systems for “building something”, and in both cases the resulting “thing to be built” has a network structure and has been given a general name.

In the case of the electronics circuit toolkit, the general name of the “thing to be built” is “circuit”. The circuits built by this kit are highly customizable to specific applications. In the case of Organodynamics, the general name of the “thing to be built” is “Organodynamic web”. The models built by this mathematical framework are also highly customizable to specific applications.

Because of the complexity of organic systems, any model that represents them needs to be built in cumulative and increasingly accurate stages. Each stage is generally referred to as an “approximation”. Organodynamics as a model builder’s toolkit provides for six of these approximations. These approximation levels are well defined and occur at precise places in the methodology set forth by Organodynamics.

In this chapter, we have investigated the ways in which the individual subsumed systems within a composite system can be related and can be interdependent. In the process, we identified three particular ways in which component systems can be related to each other and inter-related with their composite systems.

We did not have to add any structures to our existing composite system nomenclature to accomplish this, or to any other aspect of the Organodynamics framework as we have developed it so far. However, we did identify certain new relationships than can be observed between the nomenclatures for the organizations of any two systems that are so related.

Composite Organodynamic Webs

The Issue Addressed

Thus far we have developed the Organodynamic web structure to a point that we named it the Simplex Organodynamic Web, which was defined a few chapters above. This structure is the fourth approximation of the six to be developed.

This Simplex Organodynamic Web was identified, in fact, as the fourth approximation (of six) of the Organodynamic modeling methodology, making it a landmark in our quest to develop this theory and its practices.

But the Simplex Organodynamic Web lacks a mechanism that is fundamental to OCS – a mechanism to comprehensively represent deeply nested systems. However, since we define the Simplex Organodynamic Web, we have delved deeply into the concepts of nested and composite systems, and now have the intellectual constructs that can be added in order to inject these capabilities into the Organodynamic web.

In this chapter, then, we shall add the requisite features and produce the result, which we shall call the *Composite Organodynamic Web*. And, importantly, this structure qualifies as the fifth approximation of the Organodynamic model development methodology.

OCS Organizing Principle Supported

Organizing Principle # 3: Nested Systems.

Biological Example of the Issue

Multicellular organisms are deeply composite entities, containing organs that are comprised of tissues that are comprised of cells with organelles that are constituted by polymers that are made up of small molecules whose components are atoms.

Specific Challenges

The nomenclature that has been developed in the previous two chapters, called the *composite systems view* of nested systems, will be introduced into the Organodynamic web structure in this chapter. This nomenclature reveals that the components of a nested system are comprised of the organizations of its component systems. The challenge is to insert this composite systems view and its constructs into the structures already defined for the Simplex Organodynamic Web.

Composite Systems in Organodynamic Webs

It is time now to return to the notion of Organodynamics web, and see where and how the composite view of a nested system fits within that structure. Recall that our process of constructing Organodynamics as a theory has been to gradually build up the *Organodynamic web* as the single construct that provides a comprehensive model of living and lifelike system processes.

In fact, all seven of the OCS organizing principles will eventually be inculcated within this single dynamical structure. Each of the last several sections of this text have existed for the purpose of adding some significant machinery to our definition of Organodynamic web. The present chapter adds an important concept and capability to our nascent organodynamic web – nested systems. It does this through the introduction of the notion *composite system view*.

Before we make new introductions, though, let's first summarize where we are currently with our development of this Organodynamic web structure. The last chapter in which we left this structure was the one entitled *Simplex Organodynamic Webs*.

This structure modeled each time step of a stochastic process as a probability distribution each of whose sample points is a system, replete with a population and an organization. The simplex organodynamic web structure fashioned these time steps into sequential structures named *segments* and *edges*. In addition, multiple edges can be interconnected via *node* structures which are the graph theory representation of *organodynamic transforms*. In addition, a looping mechanism was added to support the notion of *routines* in order to model repetitious, or *regular*, behavior.

In this chapter, we shall preserve all of these structures – but add *nested systems* to this web. Nested system can add a lot of complexity to the framework. For this reason we have delayed its introduction. While this is a challenge to tractability, it also enables significant and essential modeling capability, since nestedness is a required organizing principle for organic systems.

The Process View of a Nested System

The *Integrate transform* takes k systems as inputs and produces $k+1$ system as output. “ k ” of those $k+1$ output systems are the same as the k input systems. The remaining output system is the new *nested system* whose components are the other k systems, and whose organization can be any set of duples whose members are constituted by those k systems.

This approach is good, because it lets us treat each of the subsumed systems as having its own distinct “life”. And we need such a view of a nested system. Because this view of a nested system lets us see the individual subsumed processes as it changes in time, we shall refer to this as the *process view*, or the, or the *horizontal view*.

However, this process view, while true, does not articulate either the rich nature of either the population or the organization of the created nested system. This is because in the *process view*, the multiple levels of organization of a nested system are each separately represented as its own distinct system. And there is no single construct that represents the resulting complexities of a nested system in such a way that each of its levels of organization are positioned “inside of” its higher levels of organization – to how ever many levels of organization it embodies.

We need a second articulation of this resulting nested system that represents the fact that the population of the new nested system is, in some sense, the other k systems, and that its organization is constituted by duples whose members are these other organized systems. Such is the purpose of the *composite view* of a nested system.

The Composite View of a Nested System

This horizontal view is important, but it is not enough. We also need a “vertical” view of a nested system that lets take a single nested system at a single step in time and look down its organizational hierarchy to behold its entire organizational and subsumed organizational structure at once within a single construct. This structure in fact is a static view of the state of a nested system at a single step within an Organodynamic web.

Fortunately, we have already developed this view and its associated nomenclature and structure in the previous two chapters. There, we named it the *composite view*.

This *composite view* of a nested system provides us with an articulation of a nested system state that reveals both the interrelationships between these levels of system organization as well as the interrelationship of “sibling” systems within each level of organization – all occurring at the same time step of a single nested system.

In the present chapter, we shall insert this view as a member of the Organodynamic web as it now stands. The result will be the Composite Organodynamic Web.

Combining the Two Representations in an Organodynamic Web

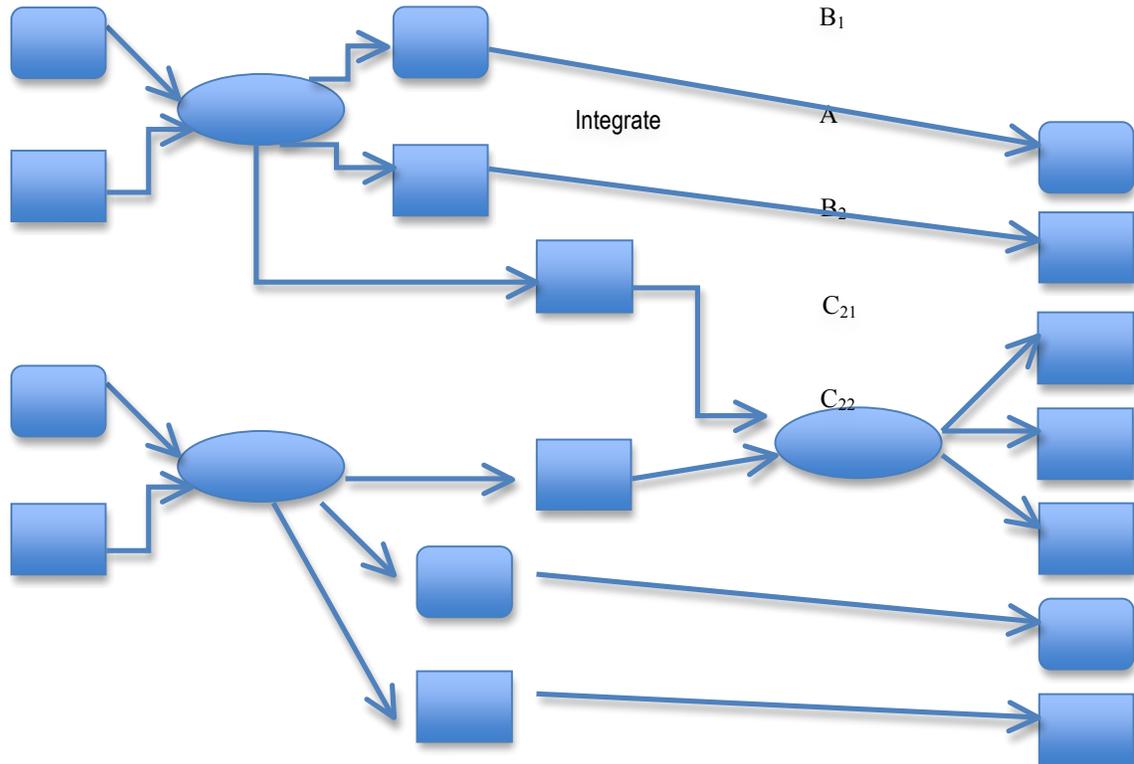
In this subsection we are going to demonstrate that we can “mix and match” the simplex and the composite representations within the same Organodynamic web. And we have much freedom as to how we wish to represent any nested system within that web in a number of ways.

Moreover, we do not have to invent any more machinery in order to do this; because we shall reuse the *composite view* that we developed in the previous two chapters.

We shall approach this subject by using a simple example. Suppose we have a nested system “A” with three levels of organization. Suppose that the first level

has two component systems "B₁" and "B₂" and that each of those has two component systems at the third level. Lets say that "B₁" has component systems "C₁₁" and "C₁₂"; and that "B₂" has component systems "C₂₁" and "C₂₂".

If we using the *simplex view* to represent this system, then the portion of the Organodynamic web that represents it looks like this:

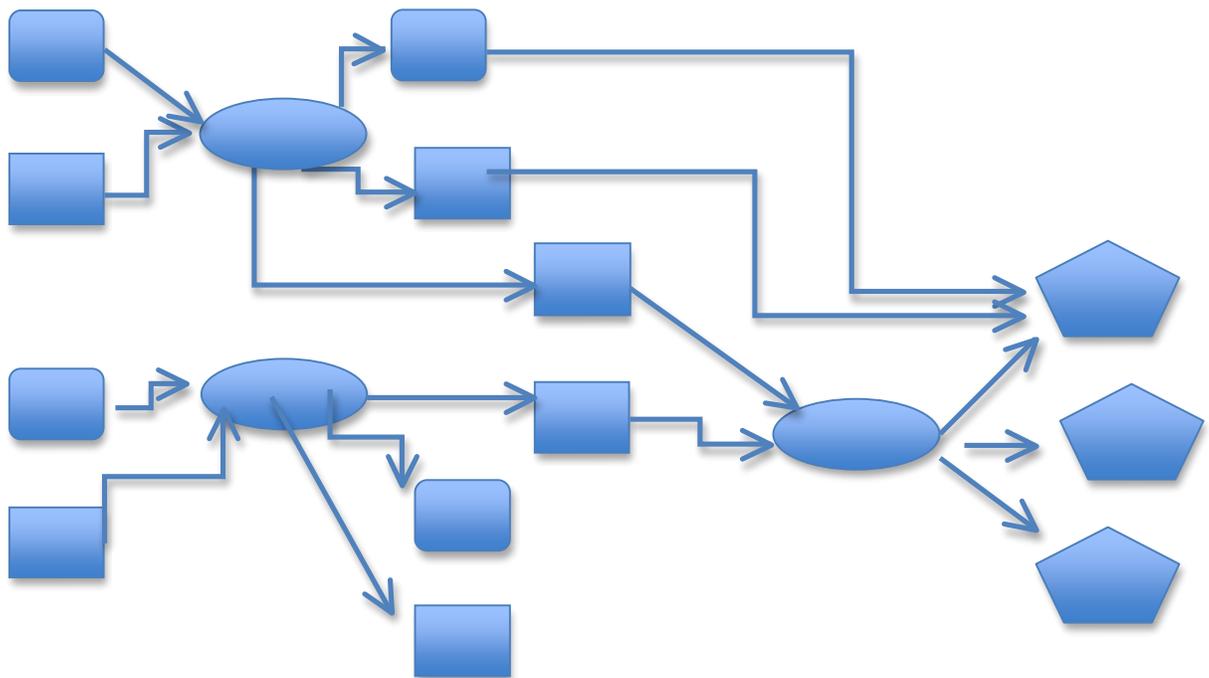


Note that there are three Integrate transforms in this figure. The first and second one (both on the left side of the diagram) create the second level of organization of the composite system. The first Integrate creates system B₁ from systems C₁₁ and C₁₂. Notice that systems C₁₁ and C₁₂ continue to exist after the Integrate transform because Integrate is non-destructive. The same can be said when the second Integrate transform creates system B₂ from systems C₂₁ and C₂₂. The third Integrate transform then creates system A using systems B₁ and B₂, which also persist after the transform.

However, an equivalent way to represent the same web is to represent some of the nested systems in the diagram in the *composite view*. For example, after the third Integrate, we could choose to represent B₁ and B₂ as composite systems – systems that contain the C's as their component systems. This would eliminate the necessity to provide the boxes for the C's in the diagram at that point, because they would already be accounted for in the B's.

But, to improve the readability, we need a diagramming symbol that indicates that a "box" is of the composite view. We shall use a five-sided box for that. The result will be a diagram with fewer symbols, because we have consolidated the C's under the B's. The resulting diagram is below.

For example, in the diagram below, we have decided to represent systems B₁ and B₂ using the composite view. This means that we no longer need to "carry along" their components, the C's, because their existence is implicit within the composite views of the B's. However, to account for the continued existence of the C's, we draw a line from the C's into these B's.



The resulting Organodynamic web view contains a mixture of the simplex views and composite views. There are a number of other places in this diagram that we could repeat the same behavior and consolidate simplex views into composite views, but will leave this as exercise for the reader.

Composite Organodynamic Webs

Thus we have shown how we can mix the *simplex* representation and the *composite* representation of various systems within the same Organodynamic web structure.

Before we began the present chapter, we had already developed the *simplex view* in the previous chapter. The use of multiple concurrent simplex views

coming from an Integrate transform node was a natural outcome of defining the Integrate Organodynamic transform.

Our challenge in this chapter has been to reuse the *composite representation* of nested systems as a mathematical structure and insert it into the Organodynamic web structure that we have so far developed prior to this chapter.

It has turned out that this insertion has been trivial – requiring only that the *composite systems view* structure be utilized to represent the output system of the Integrate transform instead of rearticulating the input systems as outputs, and then substituting the composite system structure as the integrated output for the simplex structure that was utilized previously. This refinement results in the renaming of the Simplex Organodynamic Web as the Composite Organodynamic Web.

In addition, we allow either articulation of the output of the Integrate transform in the expression of the *Composite Organodynamic Web*. This allows for greater flexibility.

And, the addition of system nesting to the Organodynamic web structure that we are incrementally developing throughout this text is significant enough to anoint the Composite Organodynamic Web structure as the *fifth approximation* of the *Organodynamic model construction methodology*.

The Uncertainty Model Gradient for a Nested System Step

We have at this point developed enough intellectual equipment to be able to develop all three of our models for the Uncertainty model gradient. All three of these models will be easy to define at this point, because we have already done most of the work.

Recall that our chapter above on the sixth OCS organizing principle (Uncertainty) and stochastic processes identified three such models that form a sequence from the most uncertain to the most deterministic. These are:

1. The Conditional Stochastic Model (most uncertain)
2. The Unconditional Stochastic Model
3. The Realized Deterministic Model (most deterministic)

It is important to understand that both the Simplex Organodynamic Web and the Composite Organodynamic Web can exist in all three forms of these models. The Organodynamic methodology expects that an Organodynamic web model will transition through all three of these versions during its life within each approximation level.

Of course, the implementation of each of these models is accomplished within the individual *time steps* that are distributed throughout the edges and the nodes of the Organodynamic web structure. The each edge consists of a sequence of time steps. It is within each of these times steps that these three

models are articulated – as well as within the intermediate single time step of each of the *organodynamic transforms* that constitute each node of any Organodynamic web.

Of course, a Composite Organodynamic Web is constructed from the same kind of time steps, segments, edges and nodes as the Simplex Organodynamic Web – with the exception that some of these can be *composite system view* constructs rather than *simplex system view* constructs.

However, we have already discussed how to model both simplex system constructs and composite system constructs for all three cases of the *uncertainty model gradient*.

Thus, there no new concepts to develop for the implementation of a Composite Organodynamic Web, except for the fact that the implementations of the entities of this web can now be both *simplex system view* and *composite system view* constructs.

Incremental Example Model Development

<>

Progress Check

In Part II, Organodynamics, we are gradually and incrementally developing a mathematical framework for modeling living and lifelike systems according to the seven organizing principles of OCS. Once completed, this modeling framework will achieve the form of an abstract but highly flexible mathematical structure that we are calling an *Organodynamic web*.

We are using the example of a hobbyist's toolkit for building electronics as an analogy to Organodynamics. Both are systems for "building something", and in both cases the resulting "thing to be built" has a network structure and has been given a general name.

In the case of the electronics circuit toolkit, the general name of the "thing to be built" is "circuit". The circuits built by this kit are highly customizable to specific applications. In the case of Organodynamics, the general name of the "thing to be built" is "Organodynamic web". The models built by this mathematical framework are also highly customizable to specific applications.

Because of the complexity of organic systems, any model that represents them needs to be built in cumulative and increasingly accurate stages. Each stage is generally referred to as an "approximation". Organodynamics as a model builder's toolkit provides for six of these approximations. These approximation levels are well defined and occur at precise places in the methodology set forth by Organodynamics.

In this chapter, we have added the ability to represent nested systems in a format that expresses this nesting within a single construct where subsumed

systems and their organizations are expressed with a nomenclature that includes subsumed systems inside of their composites.

This structure, called the *composite system view* was developed in an earlier chapter and is brought into service in the present one in order to account for rich system nesting with an Organodynamic web structure. The resulting web structure is called the *Composite Organodynamic Web*, and it constitutes the fifth approximation within the Organodynamic model development methodology.

Joint Composite Organodynamic Webs

There is an oversight with our model of organodynamic transforms that we must now repair.

This oversight was intentional. It was a simplification introduced to ease the presentation of Organodynamic webs and to permit a more incremental development of the structure. This approach made for a good approximation of organic processes. But the time has now come to add some machinery to the model that will make it more realistic.

The Issue Addressed

The Organodynamic web structure that we have been building up incrementally over the last several chapters, is the showpiece of the Organodynamic framework, and encompasses comprehensively in a single construct the whole of the Organodynamic modeling apparatus.

Because of the complexities involved in the development and presentation of the Organodynamic web structure and the necessity to develop and present it incrementally, we have deliberately allowed this oversight, and left until now its revelation and resolution.

First we must explain what the oversight is and then how to repair it.

The oversight has to do with knowing, for any time t , which if any organodynamic transforms occur and what systems states are involved with each of those transforms. In other words, we have been “pretending” to know when (at which time steps) these nodes occur – and when they do not. That is, we have been pretending to know when these organodynamic transforms occur.

We have been pretending that the occurrence within the web of these nodes, these organodynamic transforms, is deterministic, when in actuality, it too is stochastic.

Just because a collection of systems in an organodynamic web is available to come together and be operated on by a transform, doesn't necessarily mean that they will. And even if they do, *which* transform will be used?

Moreover, there are many ways that the total of all the systems at time t can partition into groupings – groupings that may or may not be operated on by some transform. Out of all the possible ways that these systems can partition themselves, which of these partitioning of groupings will they form?

Until now, we have proceeded as though we know in advance when (which time step) a specific transform will occur. Of course, if this were true, then the occurrence of transforms would be a deterministic phenomenon. But in actuality, it is not known when an organodynamic transform will occur. In fact, its occurrence is itself a probabilistic phenomenon.

So we must introduce into our Organodynamic web model a treatment of transforms that is not deterministic – that treats the occurrence of these transforms stochastically. In fact, we must remove the current aspects of our model that treats organodynamic transforms deterministically, and replace it with a stochastic treatment.

And we must also contend with the fact that these systems – even before they have the opportunity to be acted upon by some transform – must partition themselves into some number of groups, each of which is available to be potentially acted upon (or not) by some one of our transforms.

OCS Organizing Principle Supported

Organizing Principle # 6: uncertainty.

Biological Example of the Issue

Consider for example, a collection of small molecules that coexist at time t . It is a possibility that all of the conditions are right for some subset of them to chemically bond at this time to form a bigger molecule at time $t+1$.

According to specifically how this subset subsequently forms their bond, the resulting molecule may be best modeled specifically by one of the organodynamic transforms so far defined. For example, it may be that this subset of molecules could be best modeled as a *Union* transform or as an *Integrate* transform, or by some other transform in our arsenal.

Consider for example, a collection of small molecules that coexist at time t . It is a possibility that all of the conditions are right for some subset of them to chemically bond at this time to form a bigger molecule at time $t+1$.

On the other hand, just because this subset of molecules happens to coexist at time t does not mean that they will bond at all. In other words, another possibility is that no transform occurs for this subset at time t into time $t+1$. But saying that “no transform occurs” is equivalent to saying that the *Identity* transform has occurred for each of them, because the identity transform at time t leave a system unchanged at time $t+1$.

In summary, because we are including the organodynamic transforms in our model; and because these transforms can each be associated with multiples of the systems in our model (as inputs to the transform), we can no longer constrain our probability model to be associated with the single systems at time t , as we are now doing.

Specific Challenges

In our incremental development of the Organodynamic web structure, we have so far associated probability distributions with individual systems states. These states may have been simplex or they may have been composite. But it was single system states that we associated probability distribution. In other words,

the nature of our stochastic modeling thus far has been to restrict it to individual systems at time t .

The next step was to introduce dynamics by considering how any one of these systems could change state over a sequence of time steps. Since our state models were already stochastic (they were Markov transition matrices), then we could sequence the multiple states of a system over multiple time steps into a piecewise-homogeneous Markov chain. Therefore, our web could be represented at that point as a set of concurrent piecewise-homogeneous Markov chains.

A complication was the introduction of the *organodynamic transforms* that had the effect of enabling these concurrent (“parallel”) chains to sometime come together to form nodes, where each node is one of the transforms.

This introduction of transforms as nodes enabled the birth of the Organodynamic web structure from a set of parallel piecewise homogeneous processes whose sole connection was time steps.

But this was also when “the oversight” was introduced. The oversight was that we treated the occurrence of a transform at some time t as though it were determinable, predictable.

But, as pointed out in the biological example above, just because two or more systems can be grouped together does not necessarily mean that they will “bond” or “combine” or “be transformed” into a confluent output. There is some probability that they will be so transformed - but not a certainty.

However, the assumption at the time that the introduction of the transforms at time t is deterministic – while not true – was however a reasonable first approximation. And it was an approximation that didn’t burden the explanation at the time with more complexity than many readers could bear at the time.

In this chapter we shall address this oversight and change our model of the Organodynamic web to account for it.

The essential problem is that we have been viewing both state and dynamics from the perspective of individual systems. But, the introduction of the organodynamic transforms into the model requires a perspective of the entire collection of systems at time t . And our dynamical model also requires such a whole system perspective.

Thus, our essential challenge is to add a mechanism to represent the whole system perspective.

However, it is desirable to maintain the current individual system perspective as well – one for each system.

Most importantly, we must somehow integrate all of these perspectives into a single unified model where all the pieces are interrelated. This chapter will restructure the existing evolved version of the Organodynamic web (the

Composite Organodynamic Web) into the new evolution that repairs our oversight and creates a version with a whole network perspective. For now, we shall refer to this near-complete version of the Organodynamic web structure as the *Joint Composite Organodynamic web*.

In addition, this resulting comprehensive model of an Organodynamic web must continue to be probability-based – and also admit to the three articulations of the Uncertainty model gradient.

That is, we must have all three of 1) a conditional stochastic, 2) an unconditional stochastic and 3) a realized determinist model of the *Joint Composite Organodynamic web*. These are our challenges in the present chapter.

The Organodynamic Approach to Modeling the Issue

We want to

1. Preserve the individual Markov chain perspectives that we currently have of each system in an Organodynamic web at time t . There will be as many of these as there are systems at time t . This portion of the model already exists.
2. Add a perspective from the point of view of all of the systems of the web collectively. This perspective will model the participation of the organodynamic transforms. Moreover, it will treat the occurrence of these transforms and their system arguments stochastically. There will be exactly one of these perspectives.
3. Integrate all of the perspectives mentioned in 1) and 2) into a single structure. This structure must be articulated as a sample space whose sample points can be assigned probabilities.

To accomplish all of this we shall preserve the sample spaces of the individual systems in our model. These will be the Markov chains and their transition matrices for each system in the web.

We shall then add a new probability space that will model the stochastic nature of the occurrence of the transforms in the model. This new space will model all of the ways that the collection of systems at time t can form groups (sets) of systems. (That is, we shall identify all of the possible ways that a set of systems can be partitioned into “groups”.)

Each of the groups will then have the opportunity to be operated upon by a transform type that can handle the number of systems in the group as its input arguments. Each of the subgroups will have some number of transform whose “signature” fits that number of systems as inputs. Thus, we will have a number of ways that the systems can be partitioned, and within those ways we shall have a number of ways that each of those subgroups can be transformed by the available transforms in our arsenal.

Taken together, all of these possibilities form a sample space of transform invocations. It is this sample space that will represent the second perspective listed above – the whole-system perspective. We shall refer to it as the *transform probability space*, because it describes the stochastic behavior of the organodynamic transform within an Organodynamic web.

In fact, this *transform space* will effectively replace the “transform nodes” that we have in the Composite Organodynamic web structure. The “node” aspect will still be there, but they will operate stochastically – which means that they “may or may not” be there at various times, and therefore be too elusive to represent in a static graphical network.

Unfortunately, then the resulting joint composite Organodynamic web structure is beginning to be too dynamic to effectively image in the mind of the modeler.

Analysis of the Issue

Let's think about how to model this.

What we are trying to accomplish in this chapter is to replace our “individual system” view of our Organodynamic web at time t with a “joint system” view at time t . To do this, we must provide a joint sample space to replace all of the individual sample spaces that we have for each system.

Our problem in this chapter, then, is to find a joint sample space. What we have inherited from our composite Organodynamic web structure is several sample spaces for individual systems. What we want to accomplish is to form a joint sample space from these (and perhaps some extra information as well).

The probabilities of the joint distribution for time t are calculated from the conjoined events of the underlying Markov matrices for time t for each of the systems at time t .

But we also need to add a new probability space to the mix. This one will do two jobs at once. First, it will reintroduce the organodynamic transforms into the picture in a manner that permits them to be treated stochastically. After all, in real organic system processes, we can't predict when they will occur; and we can't predict which of the available systems of the organodynamic web will be used as input arguments to the ones that do occur at a time t .

So, we must introduce a probability space – the transform space - to represent this. And, we must then incorporate this transform probability space into our joint distribution. Thus, the joint distribution will have one component distribution for each system process in the Organodynamic web, plus one component for the transform distribution.

Finally, we have another complication: we must produce all three of models of the Uncertainty model gradient for this joint view: a *conditional stochastic model*, an *unconditional stochastic model* and a *realized deterministic model*. The problems for these three are all slightly different. Lets look at each one.

Considerations for the Conditional Stochastic Model

In the *conditional stochastic model*, the Markov chain model, a Markov transition matrix represents each system state. But a Markov transition matrix has two sample spaces: one for the current time step and one for the next time step. These are named, respectively, the *current time step sample space* and the *next time step sample space*.

In the Markov model, the “outcome” of a “trial” occurs within the *next step sample space*. The outcome of the *Current time step sample space* establishes which conditional probability distribution to use to calculate the probability of the outcome of the trial. Thus, the focus of the “trial” at time t is still the *next time step sample space*. Nevertheless, we must also calculate the *current time step sample space* as well as part of the Markov model for the system for the time step.

Therefore, for the conditional stochastic model, we must ascertain both sample spaces. In fact, the transition matrix is essentially a joint probability distribution whose product space is derived from these two sample spaces.

In the homogeneous case, these two sample spaces are the same. In this model we also have a transition matrix that represents a set of probability distributions of getting from the current step to the next step.

But, again, we do not have to worry right now about those conditional probability distributions that are represented in the transition matrix. All that matters to us is determining these two joint sample spaces: one for the current time step and one for the next time step.

We have already said that the general problem for this chapter is to determine a joint probability space to represent the system as a whole, rather than only representing each of the individual systems, as does our composite Organodynamic web model.

However, for the *conditional stochastic model* of the joint composite Organodynamic web structure, we must define two such joint probability spaces: one for the *current time step sample space* and another for the *next time step sample space*.

We shall call these, respectively, the *current step joint sample space* and the *next step joint sample space* for the conditional stochastic model.

Considerations for the Unconditional Stochastic Model

In the *unconditional stochastic model*, each system state is represented by a single probability distribution that describes the probabilities of each state of the sample space being the outcome of the current event.

In other words, unlike the Markov model, the unconditional stochastic model only has a *current step sample space*. Thus, the “outcome” of a “trial” occurs within the *current step sample space* in the unconditional stochastic model.

Therefore, for the *unconditional stochastic model*, we only have one joint distribution to determine. The probabilities of the joint distribution for time t are calculated from the conjoined events of the underlying Markov matrices for time t for each of the systems at time t .

To determine this joint sample space, we have the same consideration that we had for the joint sample space for the *next time step* in the case of the *conditional stochastic model*.

We shall call this the *next step joint sample space* for the unconditional stochastic model.

Considerations for the Realized Deterministic Model

In the *realized deterministic model*, the same sample space is used as for the associated *unconditional stochastic model* – the *current time step sample space*.

However, one of the sample points of this space is assigned a probability of one, while all the other now have a probability of zero.

Thus, no new problem of finding a joint sample space exists here. We simply use the same sample space we did in the *unconditional stochastic model*, and reassign the probabilities as described.

We shall call this the *next step joint sample space* for the realized deterministic model.

Conclusion for calculating all three Models

In reviewing the analysis of all three models above, we can see that two sample spaces are involved across all three: the *current step joint sample space* and the *next step sample space*.

The *current step sample space* is necessary for the *conditional stochastic model*. The *next step sample space* is required for all three of the models.

Thus, to be able to generate all three stratified joint models, we need to be able to determine both of these joint sample spaces. The *Strategy* and the *Resolution* subsections below, then, will be divided into two subsections – one for determining the *current step joint sample space*, and one for determining the *next step joint sample space*.

Strategy

Organodynamics is not the first theory to run into this problem of not being able to confine its probability model to individual pieces. Statistical Mechanics also had to solve this type of problem. It turned to the concept of *joint probability distributions* to solve its problem [Penrose, Oliver 2005]. As we shall shortly

see, we shall do the same. However, our choice of probability spaces to use to form the joint distribution will be different from those of statistical mechanics.

So, our approach will be to identify several probability spaces within our Organodynamic web model that can be treated orthogonally. We shall then form the product spaces of these orthogonal spaces. And this product space will be the joint sample space of the probability model that we seek.

The probabilities of the joint distribution for time t are calculated from the conjoined events of the underlying Markov matrices for time t for each of the systems at time t .

This concludes our central strategy.

However, we have the three models of the Uncertainty model gradient to define. And, together they utilize two probability spaces. These are:

1. The current step joint sample space, and
2. The next step joint sample space

The first one is used by *conditional stochastic model*, and the second one is used in all three models.

Thus, we have two distinct joint probability spaces, and thus two distinct joint sample spaces to develop.

The Resolution

We shall address separately the strategies that we shall pursue for each of the two joint sample spaces.

The *current step joint sample space* is the product space of only those individual *system current step sample spaces*.

On the other hand, the *next step joint sample space* product spaces include an additional sample space as well – the *transform sample space*.

These two facts will inform our strategies for developing these two joint sample spaces.

Let's now address these two joint sample spaces separately, and provide the specifics for developing each one.

Strategy for Developing the Current Step Joint Sample Space

We have already pointed out above that the *current step joint sample space* is the product space of only those individual *system current step sample spaces*.

This completely determines the *current step joint sample space*. This space is one of the two joint sample spaces that we must develop for the *conditional stochastic model*. In fact, this joint sample space is used only in the *conditional*

stochastic model. This joint sample space is not a part of either of the other two models: the unconditional stochastic *model* or of the *realized deterministic model*.

Thus, the *current step joint sample space* is the product space of the following spaces:

- *Current step sample space for system 1*
- *Current step sample space for system 2*
- ...
- *Current step sample space for system n*

Strategy for Developing the Next Step Joint Sample Space

The *next step joint sample space* is the product space formed from two sources.

The first source is also the set of individual system *current step sample spaces* that exist at time t in an Organodynamic web. These are each called the *next step sample spaces*.

The second source is a new probability space, mentioned above, that we must develop. It represents the possible ways that organodynamic transforms can operate upon the systems in the Organodynamic web at time t. This is called the *transform probability space*. Its sample space is called the *transform sample space*. We shall develop this space below.

Thus, the *next step joint sample space* is the product space of the following spaces:

- *Next step sample space for system 1*
- *Next step sample space for system 2*
- ...
- *Next step sample space for system n*
- *Transform sample space*

These will now be described in more detail.

The System Next Step Sample Spaces

For each system at time t, the next step sample space is the set of possible system organizations that are permissible to be realized in the next step of the Markov chain. These are the sample spaces that are assigned to the columns of the transition matrix in the *conditional stochastic model*.

The Transform Sample Space

As mentioned above, the *transform probability space* describes all of the ways that the existing system states within an Organodynamic web at time t can be operated upon by organodynamic transforms.

As before, we do not have to ascertain the probabilities for this space at this time. Our problem is to ascertain the related sample space – the *transform sample space* – or the *transform space* for short.

It is the *transform space* that will provide the holistic system point of view to our joint space. This is because any one of its sample points will describe a possible configuration of all of the systems in the web at time t and how they will be grouped and assigned to organodynamic transforms.

At this point we need to characterize more exactly what a sample point of the *transform space* looks like. The sample space itself will then be the set of all of these configurations.

This is a little complicated, because we are going to combine two aspects into the same sample space. These two aspects are: 1) the ways in which the set of all systems at time t can be partitioned into cells, or “groups” we shall call them; and 2) the number of system transforms whose input argument list matches each of these groups.

To discern more specifically what a sample point of this *transform space* looks like, recall that it will be “a way” in which the set of all systems at time t can be grouped together (partitioned) and then have some of the organodynamic transforms in our arsenal applied against those grouping.

It is important to recall that each of the transform types defined take an argument list of one or more systems. That is, all arguments to all transforms are systems. Also, some transforms take a specific finite number of arguments, while other take a varying number of arguments (“ n arguments”).

And, the number of systems in each grouping will determine which transforms can be performed against them. This is because the transforms in a grouping will provide the arguments for the transform. Thus, if a particular grouping has, for example, 2 systems in it, then the grouping can be transformed by any of our transforms that take exactly 2 input arguments – or any of the transforms that take “ n ” of arguments, because they can take 2 arguments also.

Therefore, each sample point of the *transform space* will represent a grouping – actually, a *partitioning* – of the set of systems that exist at time t . But for each of these partitionings, each of its groups can be associated with any transform that takes the number of systems in the group as its arguments.

That is to say, the content of any group of any partitioning of the systems at time t will be a specific transform that takes the arguments in the group. And there will be as many instances of that group as there are organodynamic transforms that can take that many arguments.

Before looking at an example of such a partitioning, it will be helpful if we build a table that contains the numbers of arguments that each transforms takes. We shall limit ourselves to the organodynamic transform types defined in the chapter on Organodynamic webs. (“ n ” means any positive number of arguments):

<u>Transform type</u>	<u>Number of Arguments</u>
Reform	n
Identity	n
Catalyze	2
Unite	n
Divide	1
Integrate	n

For example, suppose we have an Organodynamic web with six systems at time t : $S_1 - S_6$. There exists a partitioning of these six systems such that there are three groups, with S_1 in the first group, S_2 and S_3 in the second group, and $S_4 - S_6$ in the third group.

Now, the first group of our example partitioning only has one system, so we must ask what is the set of all transforms that can take exactly one parameter. The answer will be any transform type that takes either 1 or n arguments. This set is {Reform, Identity, Unite, Divide and Integrate}. So, this set has 5 transforms that can take the first group's members as arguments.

The second group of our example has 2 systems. The Transforms that can take 2 systems are members of the set {Reform, Identity, Unite, Catalyze and Integrate}. So, this set also has 5 transforms that can take the second group's members as arguments.

The third group of our example has 3 systems. The Transforms that can take 3 systems are members of the set {Reform, Identity, Unite and Integrate}. So, this set also has 4 transforms that can take the third group's members as arguments.

This for our example partition of three groups with, respectively 1, 2 and 3 members in each group, there are $5 \cdot 5 \cdot 4$ configurations of transforms that can take the arguments of the three groups in this one partitioning.

To find the total number of configurations of organodynamic transforms that can be applied to any partitioning of the set of systems in our Organodynamic web at time t , we musts enumerate all of the possible partitionings of this set of systems, and then identify all of the ways that transforms can be applied to the resulting groupings of each partitioning.

This collection of configurations will form the sample space of this last probability space – the space that we have named the *transform space*.

So, we have defined a collection of probability spaces that can vary interdependently, and that together can describe the state of an entire Organodynamic web at a specific time step t .

Our purpose in the present subsection is to define the *joint next time step sample space*, which we are now prepared to do, as follows:

The *joint next time step sample space* is the product space of the following sample spaces:

- *Next step sample space for system 1*
- *Next step sample space for system 2*
- ...
- *Next step sample space for system n*
- *Transform sample space*

Assembling the Uncertainty Model Gradient

Collectively, the three models of the Uncertainty model gradient require the *joint current next time step sample space* and the *joint next time step sample space*. Each of which we have now developed.

Thus, we shall at this time define all three of these models.

The Conditional Stochastic Model

The conditional stochastic model of an Organodynamic web is a stochastic process whose time steps are each modeled as a Markov transition matrix.

The *current time sample space* for such matrix for time step t is the *joint current next time step sample space* for time t .

The *next time sample space* for such matrix for time step t is the *joint next time step sample space* for time t .

The probabilities of the Markov matrix for time t are calculated from the conjoined events of the underlying Markov matrices for time t for each of the systems at time t .

Thus, we have defined the state of the joint space for each time t as a Markov transition matrix whose state spaces and probabilities we have also defined. This matrix defines the state of the entire Organodynamic web at time t .

Thus, for all time steps, the Organodynamic web is defined as the Markov chain whose transition matrices is these matrices at each time step.

It is to be noted that generally, this is a nonhomogeneous Markov chain. If for no other reason this occurs because of the changing nature of the organodynamic transforms at each time step. This has the effect of changing the *joint next time step sample space* for many time steps, resulting in a typical frequent occurrence of matrices with differing numbers of rows and columns.

The Unconditional Stochastic Model

The *unconditional stochastic model* of an Organodynamic web is a stochastic process whose time steps are each modeled as a single probability space whose sample space is the *joint next time step sample space* for time t .

The probabilities of the *joint next time step sample space* for time t are calculated from the conjoined events of the underlying *next time step sample space* for time t .

Thus, we have defined the state of the joint space for each time t .

For all time steps, the Organodynamic web is defined as the stochastic process whose probability distributions are *joint next time step sample spaces* enumerated over time t .

The Realized Deterministic Model

The *realized deterministic model* of an Organodynamic web is the unconditional model with the sample spaces of each step unchanged, but with the probabilities of the distributions at each step changed.

This change is such that, for each time step, the probability of exactly one of the members of the product space is one and the probabilities of the remainder of the members is zero.

Incremental Example Model Development

<>

Progress Check

In Part II, Organodynamics, we are gradually and incrementally developing a mathematical framework for modeling living and lifelike systems according to the seven organizing principles of OCS. Once completed, this modeling framework will achieve the form of an abstract but highly flexible mathematical structure that we are calling an *Organodynamic web*.

We are using the example of a hobbyist's toolkit for building electronics as an analogy to Organodynamics. Both are systems for "building something", and in both cases the resulting "thing to be built" has a network structure and has been given a general name.

In the case of the electronics circuit toolkit, the general name of the "thing to be built" is "circuit". The circuits built by this kit are highly customizable to specific applications. In the case of Organodynamics, the general name of the "thing to be built" is "Organodynamic web". The models built by this mathematical framework are also highly customizable to specific applications.

Because of the complexity of organic systems, any model that represents them needs to be built in cumulative and increasingly accurate stages. Each stage is generally referred to as an "approximation". Organodynamics as a model builder's toolkit provides for six of these approximations. These approximation levels are well defined and occur at precise places in the methodology set forth by Organodynamics.

In this chapter we have presented the sixth and final approximation of our uncertainty model gradient model development methodology – the *Joint Composite Organodynamic web*. This model effectively takes a similar approach to that of statistical mechanics when faced with the problem of

massive unpredictability of unpredictable joint events – it resorts to a joint probability distribution.

In the case of Organodynamics, the unpredictable “straw that broke the camel’s back” was the observation that “encounter events” (that involved phenomena that were already themselves probabilistic) were also probabilistic. These encounter events are modeled in Organodynamics as nodes (from a network perspective) or as *organodynamic* transforms (from an algebraic perspective).

Furthermore, it was observed that heretofore in the earlier approximations of the Organodynamic web, these “encounter events” were being treated as though they are deterministic, when in fact they are not.

The treatment that this chapter provided to solve this problem was to take all of the individual systems that were involved in an Organodynamic web, plus all of the nodes of that web, and to treat them each as an individual “dimension” in a joint probability space. Then, give each of the joint events in that joint sample space its own probability. The result is the joint probability space that we have named the *Joint Composite Organodynamic web*.

This approach of using a joint probability distribution to model the fact that “everything in the space is turning out to be stochastic” is a time-honored approach in dynamical systems. The statistical mechanists used it, as did the Cyberneticists, for example.

The downside of this approach, unfortunately, is the loss of intuitiveness. But, as science has shown many times in the last century [Gibbs 1902; Penrose, Oliver 2005; Wiener 1961], sometimes intuitiveness is the casualty of deep understanding.

Measuring Uncertainty in Organodynamic Webs

The sixth organizing principle of OCS asserts:

All living processes depend upon both chance events as well as deterministic ones. Living processes operate along a spectrum between complete randomness and complete determinism, uncertainty and certainty. Where along that spectrum an organic system lies at any time is unpredictable. Thus, the degree of certainty or uncertainty of an organic system is itself unpredictable. ("Life is unpredictably unpredictable.")

This chapter defines with precision how this organizing principle is implemented within the Organodynamic framework – specifically for all (six) approximation levels within the three models of the Uncertainty model gradient.

<grant - As we shall see in a later chapter, the approach taken will be to treat this collection of entropies of the probability distributions involved in a single Organodynamic web as a random variable, and then characterize that random variable by its moments. >

The Issue Addressed

The principle issue is the determination of how *uncertainty* will be measured in an Organodynamic web.

This issue is complicated by the fact that there are three distinct models required by Uncertainty model gradient for an Organodynamic web. Moreover, for each of these three models, there are six levels of approximation through which each of these three models may advance.

As discussed, a given modeling exercise may choose to stop at a certain level of approximation and not go through all six levels. Nevertheless, for each of the three models of the uncertainty gradient, a measure of uncertainty must be provided for each approximation level through which the modeling exercise advances.

Thus, for the modeling of any organic system using the Organodynamics framework, we must provide a specific algorithm for calculating the degree of uncertainty for each of these distinct model/approximation-level combinations.

OCS Organizing Principle Supported

Organizing Principle # 6: Uncertainty.

Specific Challenges

The sixth organizing principle of OCS contains the statements

“Thus, the degree of certainty or uncertainty of an organic system is itself unpredictable. (“Life is unpredictably unpredictable.”)”

We must articulate the meaning of this statement in precise mathematical language, and then proceed to substantiate it.

The Organodynamic Approach to Modeling the Issue

We shall define the notion of “uncertainty of an organic system” from established concepts in Information Theory.

To accomplish this, we identify these concepts and apply them to the task of measuring of the *degree of uncertainty* for each of the 18 models referred to above. Principle among these concepts is that of *Shannon entropy*, which we shall appropriate as our uncertainty (degree of uncertainty) measure.

We shall emphasize that Shannon defined *entropy* as a measure of both *uncertainty* and of *information* – which he equated. We shall also spend some time explaining this, perhaps counterintuitive, equation in Information Theory.

Finally, consider any organic process. All 18 of our models represent such a process – at varying approximation levels – as a sequence of time steps, each of which is represented by one or more probability spaces.

Thus, as we traverse the three models of the uncertainty model gradient through all six approximations, our measure of uncertainty of the model is constantly changing. The reason for this change is because our knowledge of the system is changing as we progress. Sometimes this change increases our degree of choice (and therefore of uncertainty) and sometimes we gain certainty.

Thus, it is to be expected that our measures of uncertainty will change. One reason for this is that uncertainty is a relationship between an observer and a system that is being observed – rather than strictly a measure of a system in isolation from observers. After all, “doing science” is an empirical exercise. It involves an observer. For the same reason, so is “doing modeling”

Once we have introduced probability into Organodynamic models then each of the approximation levels involves a network of probability spaces. To each of these probability spaces, we can apply our uncertainty measure (Shannon entropy).

Uncertainty and Information

Uncertainty is a significant concept in Organodynamics because it is the sixth organizing principle of OCS. This organizing principle makes assertions about the degree of uncertainty of organic processes – namely that the degree of uncertainty undergoes variation within the “life” of a single organic process, and, furthermore, that such variation is itself uncertain. In other words, unless a system exhibits “unpredictable uncertainty”, then OCS is not willing to say that it is “lifelike”.

In order to put meaning to these assertions, and to the concepts underlying them, it is evident that we must establish some way of measuring this notion of uncertainty. This we mean to do in this chapter.

Uncertainty and certainty are certainly opposites. As the sixth organizing principle of OCS points out, organic systems are constantly changing from certain to uncertain and back again.

At any point in time, various aspects of an organic system will exhibit a degree of uncertainty – with respect to that aspect – somewhere along a spectrum between complete certainty and complete uncertainty. The sixth organization principle says that this is a “fact of life”.

When uncertainty changes to certainty – it is called *information*. This dialectic is the subject of a branch of probability theory named Information Theory.

The founder of Information Theory, Claude Shannon, did not make a distinction between *uncertainty and information*. Shannon's interest in the two was that they are mathematically equivalent. That is, the same formula that measures the degree of one also measures the degree of the other [Shannon 1963]. We shall derive the measuring function for both below.

However, an early interpreter of Shannon's work, the renowned mathematician A. I. Khinchin of the Russian school of probability theory, in his *Mathematical Foundations of Information Theory*, provides a plausible explanation. Khinchin explains:

Thus, we can say that the information given us by carrying out some experiment consists in removing the uncertainty which existed before the experiment. The larger this uncertainty, the larger we consider to be the amount of information obtained by removing it.

So, the degree of uncertainty and the resulting degree of information are equivalent. Essentially, the *information* lies latent within the uncertainty. And this explains why the two are measured by the same function (information entropy).

Essentially, one could say that *uncertainty and information* are, indeed, semantically distinct. But - since one essentially derives from the other - they are both measured by the same formula, and are thus mathematically equivalent.

Since the notion of uncertainty/information is so central to Organodynamics, we need to contrive a way to measure it. We shall turn our attention to that below.

The Degree of Uncertainty Inherent in an Event

The idea of attempting to measure the degree of uncertainty inherent in some phenomenon has received considerable attention with the established field of Information Theory. We therefore shall defer to that august discipline to form the basis of our measuring function.

We have already described the mathematical measuring function of the amount of information inherent in a “situation” as modeled by a probability distribution. That function, Shannon’s entropy, has been discussed throughout this text in one chapter or another.

However, it may not be intuitively obvious why Shannon’s entropy formula actually works as a measure of uncertainty. It turns out, however, that there is an intuitive explanation. This explanation is based upon the fact that there is a simpler and more fundamental concept than the “uncertainty of an entire probability distribution”. This simpler concept is the “uncertainty of a single event”.

This section intends to make plain and intuitively clear the meaning the “uncertainty of a single event” – and how it can be measured in an obvious manner. Once that is done, then we shall demonstrate that it is a simple step from there to the measure of uncertainty of an entire probability distribution.

In Information Theory, the root of uncertainty is *unlikelihood*, or *improbability*. And any measure of uncertainty must take account of that fact. This leads us directly into the theory of probability. Thus, the concept of uncertainty will be articulated in terms of probabilities.

But, before we define how to *measure* uncertainty, lets ask, “What is it that can be uncertain?”

In probability theory, the answer to this question is “anything that can possibly happen” – which we shall call “an event”. Without delving into the details of probability theory, let’s define that our measure of uncertainty will be applied to the notion of an *event*. It is the *event* that can have varying *degrees of uncertainty*.

Before we are done, we shall extend our measure of uncertainty to apply to whole collections of events, as well as to single ones.

The generally accepted idea is that “the more unlikely, or improbable, an event is to occur, the more should be its measure of uncertainty.” So, the *degree of uncertainty* of an event should vary directly with its improbability.

We can reword this by saying that: *The uncertainty of an event should vary inversely to its probability.*

Let’s state this symbolically. Let E be an event, and Pr(E) be the probability of event E. Then, we want for any measure of uncertainty of event E to be a function of the inverse of its probability.

A refined way of saying this is that, we want the uncertainty of an event E to be a function of $1/\text{Pr}(E)$.

So, a “candidate” formula for the uncertainty measure of an event could be:

$$\text{Candidate measure of uncertainty: } U'(E) = 1/\text{Pr}(E)$$

We are naming this measuring function U' because – as we shall soon see - it is not quite good enough. And we are going to change it slightly. The revised one will be called U .

What is good about U' is that it is “monotonic”. That is, as the improbability of E increases, U' also increases. And we “certainly” want any measuring function to be monotonic.

So, what is wrong with U ? It is missing another property that we value in measuring functions – additivity.

We also want our measuring functions to be “additive” under certain conditions. And, since we are free to invent all of the measuring functions that we wish, we might as well invent one that has all of the properties we desire. U' had one property that we desire – monotonicity. But it may not have other properties that we also want.

For probability spaces, we want our measure of uncertainty to be *additive* whenever two statistically independent events are involved. That is, if events A and B are stochastically independent, we would like for the uncertainty of their joint occurrence to be equal to the sum of the uncertainty of their individual occurrences.

Without getting into the probability theory and proving it, we shall point out that our candidate measure of uncertainty above, U' , is not additive! That is why it is not good enough.

However, it is fairly easy to make a small revision to U' that will render the resulting measuring function as additive. Importantly, this revision retains the monotonically increasing property.

Here is the revised measuring function of the uncertainty of an event E :

$$U(E) = \log(1/\text{Pr}(E))$$

It matters not what the base of the logarithm is. Any base will do. Of course, the choice of different bases yields different answers. In information Theory it is customary to choose a base of 2.

The Degree of Information Inherent in an Event

As we have discussed, in Information Theory, the notion of *uncertainty* and that of *information* are mathematically equivalent. And a measuring function of one is the measuring function of the other.

This fact is evident in Shannon’s seminal work [Shannon 1963] by virtue of the fact that he says that they are and provides a function to measure them both (along with “choice”).

This argument is also made plain in [Vedral 2010], where an excellent discussion is had regarding the equivalence of *uncertainty* and *information* in Information Theory. Likewise, Vedral presents the formula in terms of measuring information rather than measuring uncertainty. His articulation of the formula for the degree of information in a single event E is:

$$I(E) = \log(1/\text{Pr}(E))$$

Simplifying the Uncertainty Measure

Hopefully you noticed that the two formulas above are exactly the same except for the choice of terminology: “U” for uncertainty and “I” for information.

Could it be that we have two distinct concepts that happen coincidentally to have the same formula? (Like the formulas for *distance* and *area*.) Or is it that the two concepts are equivalent – and therefore *must* have the same formula?

Information Theory asserts that the latter is the case. *Information* and *uncertainty* are – if not the same phenomenon – strongly related. Since this idea is counterintuitive to many, we shall delve more deeply into it in the next two subsections.

For now, though, we need to provide an alternate expression of the above formula for our uncertainty formula: $U(E) = -\log(\text{Pr}(E))$.

It is a brief proof that $\log(1/\text{Pr}(E)) = -\log(\text{Pr}(E))$, as follows:

$$I(E) = \log(1/\text{Pr}(E)) = \log(1) - \log(\text{Pr}(E)) = 0 - \log(\text{Pr}(E)) = -\log(\text{Pr}(E))$$

Thus,

$$I(E) = -\log(\text{Pr}(E))$$

Admittedly, “ $-\log(\text{Pr}(E))$ ” is not that much simpler than “ $\log(1/\text{Pr}(E))$ ”. However, “ $-\log(\text{Pr}(E))$ ” is the form more often cited.

Shannon Entropy: The Mean Degree of Uncertainty/Information

Probability theory is very useful for modeling phenomena that exhibit some amount of randomness in their behavior.

In order to model this, probability theory represents a phenomenon as an “experiment” of repeated trials. We shall refer to each possible single outcome of a repeated trial as an *alternative*.

Being alternatives, these possible outcomes are mutually exclusive. And, each of these alternatives has a specific probability of occurring when a trial is run. Probability theory assigns a non-negative real number value to these probabilities. Moreover, for normalization purposes, it stipulates that the sum of the probabilities of these alternatives is 1.

As well, the occurrence of one of these alternatives is also called an *event*. These events have the same probabilities as assigned to their associated alternatives. And the set of all alternatives is referred to as the *sample space*.

Probability theory is also interested in defining sets of these alternatives, extending the notion of *event* to them as well, and also assigning probabilities to these “compound events” as well. Probability theory assigns the value of the sum of all of the alternatives of a compound event to the compound event.

To summarize, we have discussed three phenomena that are essential to probability theory: 1) a set of alternatives, 2) a set of events on these alternatives and 3) a set of probability assignments to these events.

After adding some rules on the set of events (so that they qualify as a *sigma – algebra* – also called a *sigma-field*), these three entities together are referred to as a *probability space* [Feller 1968].

In addition, in the case of a finite set of alternatives, the probability function as applied to the set of alternatives is referred to as a *probability distribution*.

Lets get back to the notion of $I(E)$, our measuring function of the probability of a single event. Now that we have a probability space, we know that all events in a probability space have probability assignments and therefore can be measured via $I(E)$. And, since alternatives are also events, then we can apply $I(E)$ to the alternatives in the sample space as well.

However, what is also desired is to extend this measure of uncertainty from single sample space *members* (i.e. alternatives) to the entire sample space of alternatives as a compound entity. In other words, we would like to generalize our measure of uncertainty of a single alternative so that we can apply it to an entire probability distribution.

This is easily accomplished by taking the usual approach when a measure of individual instances is extended to a group: take the average, or mean. For example, if we have measured the heights of a group of men, then we can develop a meaningful measure of height of the entire group by taking the average of the heights of each member of the group.

So, we shall take the mean of $I(E)$ across all sample points of the distribution. Of course, the mean is the sum of the products of the values and their associated probabilities taken over the sample space.

However, for each sample point E , its value is

$$U(E) = \log(1/\Pr(E)) = - \log(\Pr(x));$$

and its probability is

$$\Pr(x).$$

Thus, for all of the E 's in the sample space of alternatives, this product is

$$\Pr(x)*(- \log(\Pr(x))) = -\Pr(x)*(\log(\Pr(x))).$$

And, we need to sum this value for all members in the sample space.

Thus, for the entire sample space of N members, this is:

$$\sum_{i=1,N} \Pr(E_n) * (-\log(\Pr(E_N))) = -\sum_{i=1,N} \Pr(E_n) * \log(\Pr(E_N))$$

But, this is precisely the formula for Shannon's entropy of a probability distribution – which he named in reference to the similar concept from Statistical Mechanics. He also uses the same symbol H.

Thus the *Shannon entropy* for a discrete random variable E with a population of N sample points is:

$$H(x) = -\sum_{i=1,N} \Pr(E_n) * \log(\Pr(E_N))$$

The Equivalence of Uncertainty and Information

We have already asserted above the equivalence of the concepts of *uncertainty* and *information* in Information Theory.

Admittedly, this equivalence will be counterintuitive to many. So it behooves us to address this briefly.

It is widely held that, in order to dissipate uncertainty, the remedy is to increase the amount of information. According to this viewpoint, the degree of uncertainty should decrease as the degree of information increases – which is contrary to the position taken by Information Theory. Thus, how can Information Theory assert the equivalence of these two ideas, when they appear intuitively to be opposites (vary inversely)?

The explanation of this conundrum is, not surprisingly, that common usage means one thing by the term *information*, while Information Theory means another.

In common usage, “information” is the information that YOU HAVE, or that one has, pertaining to a phenomenon. Clearly, the less information YOU HAVE, the more uncertain YOU are likely to be.

But Information Theory is not concerned with the amount of information that YOU HAVE concerning a phenomenon. Rather it is interested in how much information THERE IS – that exists – inherent in the phenomenon itself.

(Of course, YOU the observer is brought back into the issue when it is realized that the formulas for *uncertainty*, *information* and *entropy* have as their only parameter a *probability* – which at its heart is a measure of a relationship between an observer and the observed. To wit, two observers can have differing probabilities regarding the same event.)

The point of view of Information Theory is that the more unlikely an event becomes, the more information it has. Moreover, the amount of information inherent in a phenomenon (its entropy) also increases as the mean amount of the uncertainty, I(E), of its alternatives increases. And this has nothing to do

with how much information you or anyone else happens to possess, or not possess, about that phenomenon. It is in this sense that *information is uncertainty* and *uncertainty is information*.

It will be helpful in studying Information Theory and anything related to Shannon entropy if this is kept in mind. It is also instructive to note that Shannon, who invented Information Theory in 1948 [Shannon 1963], took this approach right from the very beginning.

Here is a quote from Shannon's original paper (published as a book in 1963) that shows how he considered the two concepts as equivalent (in addition to the concept of *choice*):

Quantities of the form $-\sum p_i \log p_i \dots$ play a central role in information theory as measures of information, choice and uncertainty [Shannon 1963].

Shannon Entropy and the Uncertainty Model Gradient

Since *uncertainty* is an important organizing principle in Organodynamics, we must show how it can be applied to each of the three models of the *uncertainty model gradient*.

Moreover, you will recall that each of these three models advances through six levels of approximation afforded by the incremental and cumulative approach to modeling an organic process.

Thus, the estimation of the degree of uncertainty must be allowed to change at each level of approximation. This is because, since each level of approximation adds more fidelity, then it may also be expected to decrease the degree of uncertainty.

In this chapter we shall discuss the general approach to applying Shannon's entropy to each of the three models of the uncertainty model gradient. After that, we shall discuss how each of these three can be refined as the modeling effort advances through the six approximation levels of the Organodynamic web structure.

The transition through the three models of the uncertainty model gradient assumes that the modeler has resolved some uncertainties and needs a new model to reflect that. This generally translates into the need for a revised model in which the Shannon entropies of the model are changed.

This generally calls for a different set of measures of the Shannon entropies for these three levels of model. We shall visit each of these in turn below and identify the approach to measuring the Shannon entropy, or uncertainty for each of the three cases.

Some might expect that all 18 of these measures that we have to develop for a specific organic process should provide the same measurement (Shannon entropy), since they are all models – at varying levels of approximation – of the same underlying organic system.

However, this is not true. Each of the 18 measures of uncertainty will generally be unique – even though their structures are models of the same underlying organic system. One must realize that Shannon's entropy is a measure of uncertainty (uncertainty) of a *model*; not of its underlying domain. The model is not the domain.

We know this is true because Shannon's entropy is a function of a probability distribution. And probability distributions are mathematical abstractions - not real live goo. As well, we know that each of these 18 models provides its own approach to using probability distributions for representing the same underlying organic system. This is why they provide differing levels of approximation or fidelity.

Of course we shall use the same basic formula for calculating each of the 18 values for a given organic process. However, each of these 18 valuations (formulas) of uncertainty is expected to, in general, provide unique parameters to that formula, and thus to generally provide unique measures. Thus, it would be surprising if all of these 18 valuations of the same underlying organic process did, indeed, exhibit the same uncertainty (Shannon's entropy).

Uncertainty of the Conditional Stochastic Models

The first model of the uncertainty model gradient is the *conditional stochastic* mode. We are using a specific type of this – the Markov chain. The essential problem with developing a measure of uncertainty for a Markov chain is that – even for a single time step - there is no one probability distribution to use for the calculation of entropy.

Rather, there is a *set* of probability distributions: the conditional distributions that are associated with the time step and that are represented in the transition matrix. This puts us into a position to calculate *several* entropy values for a single time step of a Markov chain.

No Uncertainty Measure for the Conditional Stochastic Model

It would be convenient if there were some way inherent in a single time step of a Markov chain to force, or even suggest, some way to “combine” all of these conditional distributions into one. But there is not.

In fact, it must be realized that forcing such a combining is equivalent to forcing a conversion to the second type of model – the unconditional stochastic model.

Organodynamics takes the position that the Markov model (and its more general conditional stochastic model) are *too uncertain* to provide an actual *measure* of uncertainty.

Equivocation

The impossibility of developing an entropy measure for a single time step – due to there being multiple unresolvable probability distributions – does not stop some from trying to force one.

A popular approach exists for such a calculation. It is called *equivocation*.

Equivocation provides a mechanism for combining all of the conditional distributions into one. It does this by making the assumption that all of them are equally likely to be selected as *the* distribution to be used. (Of course, in the view of Organodynamics, this assumption effectively promotes the model to an *unconditional stochastic model* – and therefore only changes the problem rather than solving it.)

This assumption is easily implemented by treating it as an “initial condition” – the initial condition where by the initial conditions vector has equiprobable values for all of its entries.

When one multiplies the transition matrix by this vector then the set of conditional probability distributions are effectively averaged into a single distribution – called a *marginal distribution*. This marginal distribution can then be used to calculate an entropy value for the time step in question of the Markov chain model.

Classical Probability Structures Involved

Information Theory provides more formal mathematical structures from probability theory than were described above. These structures support the algorithm described; and it should be useful to identify them here.

First there is the notion of *conditional entropy of a conditional probability distribution*. This is calculation by applying Shannon’s formula to the probabilities identified in each of the conditional probability distributions of a time step. This statistic could, then be calculated for each of these distributions.

Next, there is the notion of the *conditional entropy of one random variable, given the probabilities of the other*. In our case, we do not have “the probabilities of the other”, so we assumed that they were equiprobable.

In fact, our calculation used this second notion, the conditional probability distribution – which is also known as *equivocation*, for the reasons we discussed above.

The Wikipedia site at http://en.wikipedia.org/wiki/Quantities_of_information contains a useful summary of these concepts.

Uncertainty of the Unconditional Stochastic Models

In all unconditional stochastic models, we have a single probability distribution that describes the stochastic behavior of the current step of them organic process being modeled.

This being the case, it is easily seen how the uncertainty value for the time step is ascertained: simply calculate the Shannon entropy for the probability distribution represented.

Uncertainty of the Realized Deterministic Models

The probability distribution for a single step of all *realized deterministic models* is the constant distribution. This means that the probability of one of its population members is one (1).

But $-\log(1) = 0$. Thus the Shannon entropy of this distribution is zero (0); and the entropy of any step is zero.

Any of the six approximation levels of our Organodynamics models will define uncertainty as some linear combination of the entropy of certain other time steps that precede it in their approximation level. Consequently the uncertainty of all steps of any of our six approximation levels is zero.

Uncertainty in Organodynamic Basic Structures

At this point, we have established some general observations regarding the uncertainty values for each of the three models of the uncertainty model gradient. The conclusions are:

1. There is no measure of uncertainty applied to conditional stochastic models of a time step, because there is too much uncertainty to measure. This fact is a result of there being no way to resolve the multiple probability distributions to a single distribution without forcing the model to be promoted to an unconditional stochastic model.
2. The time steps of an unconditional stochastic model are single probability distributions. Therefore their entropy values can be calculated. These are the entropies of the time steps of this model.
3. The time steps of a realized deterministic model all have constant distributions, whose entropy is zero (0). Thus, the entropies of these models is zero everywhere.

The conclusion of these three observations is that entropies are interesting only for the unconditional stochastic models of all six approximations of the uncertainty model gradient.

What we need to do now is to see how the uncertainty calculations are refined for the *unconditional stochastic models* as they advance through the six levels of approximation in the incremental and cumulative model development of the Organodynamics framework.

First, however, we must visit the calculation of uncertainty for the basic structure for *system state* in Organodynamics. This is true because the system is the fundamental entity of Organodynamics is present in every model at every level of approximation in the framework.

In the discussions above, we have already been describing the uncertainty value for a system, because the sample spaces of the probability distribution discussed are always, in Organodynamics, systems. Specifically, the

organizations of systems constitute the sample space members. But in actuality their underlying populations are implied. And thus, the sample space members involved in the discussion are essentially systems.

What is left to discuss in this subsection is the distinction between the process and the composite views of nested systems, and how the stochastic representation of the two varies – as well as their uncertainty values.

Uncertainty of a Simplex System

The population of a simplex system is a single-level (non-nested) population of any components that reasonable lifelike system could exhibit. In Organodynamics, these populations are abstract.

The sample space is a set of possible system organizations all of whose population is the abstract one mentioned above. These organizations can be any set of related duples whose members are components of the underlying abstract population.

The Conditional Stochastic Model

Like all conditional stochastic models, the simplex system has no entropy defined, because there is no single probability distribution associated with any of its time steps to permit such a calculation.

The Conditional Stochastic Model

A simplex system will have a probability distribution associated with it. The calculation of uncertainty will be the Shannon entropy of this distribution.

The Realized Deterministic Model

The entropy of all time steps of a realized deterministic model is zero (0). The same is true for a simplex system.

Uncertainty of Nested Systems

Organodynamics provides two distinct views of nested systems: the process view and the composite view.

However, being different views they each present a different picture of the degrees of uncertainty within a nested system. Thus, each of these must be accounted for separately. (Remember, entropy is a measure of the uncertainty inherent in a model – not of the object system.) The uncertainties associated with each are discussed in this subsection.

Uncertainty of The Process View of a Nested System

Within an Organodynamic web, the component systems involved at each of the multiple levels of nesting are modeled as individual simplex systems. Thus, each of these simplex models provides its own distinct measure of its entropy.

Uncertainty of The Composite View of a Nested System

The *composite view* of nested system constructs an elaborate sample space for all of its levels of organization collectively.

In this view, each sample space member is a complex representation of a vertical path from the bottom of the nesting hierarchy to the top, and is generally a large and complex sample space. But, while the articulation of sample members is complex and large, there is only one probability distribution for the entire sample space.

The entropy calculation for such a view of the nested system is the entropy calculation against this single probability distribution.

In fact, we can say that the use of the composite view, as opposed to the process view, of a nested system actually “tames” the discussion around its entropy – if only because it permits the calculation of a single entropy value for the entire nested system. This is due to the fact that the composite view presents a single probability distribution for the entire nested system, rather than distinct probability distributions for each of its subsumed systems.

But one must expect this since the composite view takes into account that there are internal dependencies between composite systems and its component systems for all of the levels of organization involved in the nested system. And, since these dependencies are taken into account, one might expect the improved predictability to the model.

Uncertainty in Organodynamic Approximation Levels

We have already discussed the idea that entropy is interesting only for the second model of the uncertainty model gradient – the unconditional stochastic model. Thus, this section will allude only to that model type throughout the discussion of all six approximation levels.

Characterizing The Uncertainty of the Six Approximations

In this subsection, we shall visit each of these six approximation levels one at a time, and describe how the use of uncertainty will characterize the *unconditional stochastic model* at that level.

Each of these approximations consists of some a collection of contiguous time steps, each of which represents one or more of the system views just discussed. Some of these approximations involve multiple concurrent (parallel) contiguous sequences of time steps of system states.

We shall not attempt to develop a single entropy value for any the six approximations. Rather, we shall consider the collection of all of the entropy values for each of the system states – across all time steps - represented in the models for that approximation.

Therefore, our approach is to represent the uncertainty inherent in any of the six approximations as a probability distribution of entropy values.

In the case that a nested system is represented by the process view, each of its subsumes systems will count as a separate system whose uncertainty is represented as a sample point in the distribution of uncertainty values. This will result in process views of nested systems having a different distribution of entropy values than the composite view of the same nested system.

<grant – Revisit the decision in the above paragraph once the Khinchin book is received. Also discuss this decision with Dean. >

Suggestion for Further Research: the Entropy Random Variable

Thus, in order to characterize the uncertainty involved in any of the six approximation levels, we shall define a new set whose members are the individual probability distributions of each time step or node of the structure involved. Let's call this set the *distributions of the approximation*. In the case that the approximation is an Organodynamic web (which the last three are), lets call it the *distributions of the Organodynamic web*.

For nodes, the time step involved will be the output systems. For split nodes, each of the output systems will be included. In this section, we shall use the term "time step" to include nodes.

We can define a function on this set - the *distributions of the approximation* - that maps each distribution to its entropy. At this point, then, we have a set whose members are all of the entropies of all of the distributions of the Organodynamic web structure involved.

Lets name this set the *entropies of the approximation*. In the case that the approximation is an Organodynamic web (which the last three are), lets call it the *entropies of the Organodynamic web*.

Ultimately, it is desirable to be able to associate probabilities to each of the distributions in the *distributions of the approximation*, and thus to the *entropies of the approximation*.

If this were possible, then such a function would qualify as a *random variable*, because it maps the members of a sample space (which in this case are, themselves, probability distributions) to real number values (which are the entropies of these distributions).

However, finding such a random variable would amount to associating probabilities to each of the times steps, and nodes, of an Organodynamic web. Such a distribution would pertain to the relative frequency of "visits" to each

time step or node in the Organodynamic web structure. Such an investigation would involve the theory of *stochastic networks*. We shall not enter such an investigation here, but rather suggest it for further research.

In any event, if such probabilities could be developed, then we would have a new random variable that associated each of these distributions with its own entropy. At which point, we would name this the *entropy random variable*.

First Approximation: Uncertainty of a Segment of a Markov Chain

Recall that a segment of a Markov chains is defined to be a maximal homogeneous contiguous sequence of time steps.

This will result in a constant distribution of entropies for this approximation.

<grant – an alternative to this is to use the “entropy rate”, which is based upon the stationary distribution of a homogeneous Markov chain. That is, the entropy of the stationary distribution is used as the entropy of the chain. This approach is VERY different from the one proposed involving the joint distribution of all states involved. Discuss this with Dean.

Also, pursue research on “entropy rates” of stochastic processes. In fact, there is a paper on this in the “research” folder. >

Second Approximation: Uncertainty of an Edge of a Markov Chain

Recall that an *edge* is a piecewise-homogeneous finite Markov Chain.

The distribution of entropies will have as many values as there are time steps in the edge, and as many distinct values as there are segments in the edge.

The Conditional Stochastic Model

Use equivocation (conditional entropy) to resolve the conditional distributions of each step involved to a single distribution – the “marginal distribution”. (This is equivalent to assuming an equiprobable set of initial conditions.)

Then, use these resulting marginal distributions as to represent its system in the joint distribution.

Third Approximation: Uncertainty of an Organodynamic Graph

Recall that this approximation consists of multiple concurrent edges, each of which may consist of multiple consecutive segments. In addition, however, we add *nodes*. Each of these nodes adds one or more systems to our structure its output system. The number of output systems added by a node depends upon whether the node is a split node. A node may be either a *join node*, a *split node*, or both.

Split nodes have multiple output systems, and therefore provide multiple probability distributions to an unconditional stochastic model. Of course, each of these has an entropy value to add to the structure.

Fourth Approximation: Simplex Organodynamic Web

The only new mechanism added to this approximation is the loopback mechanism. However, this makes no change to the number of output systems or the number of entropy values. Therefore, the calculation of entropies of the structure is the same as for the third approximation.

Fifth Approximation: Uncertainty of a Composite Organodynamic Web

The only new element added to this approximation is the composite view of nested systems. However, this does not add any new output systems. In fact can reduce the number of output systems by replacing multiple simplex systems that were output systems in the fourth approximation with single instances of composite view time steps.

Thus, in general, Composite Organodynamic Webs will have less entropies to deal with than will the fourth approximation.

the joint distribution.

Sixth Approximation: Uncertainty of a Joint Composite Organodynamic Web

It turns out that – contrary to earlier pronouncements of this text – we ultimately do represent an entire Organodynamic web by a single entropy value. It is in this sixth approximation that this is achieved.

This is because the sixth approximation provides only one single probability distribution for its entire Organodynamic web!

This is true because this approximation “wraps” all segments, edges, nodes and time steps into a single complex probability space – the joint distribution, whose multiple dimensions encompasses all of the ways that these can vary.

In fact a single member of the sample space of this distribution is a snapshot that selects a single state for all possible edges, nodes and time steps – and assigns a probability to it.

It turns out that this approach is the only way that we have managed to treat the occurrence of nodes as probabilistic – which is a requirement for a truly stochastic dynamical system. This fact makes this sixth approximation essential.

The downside to this approximation, however, is the difficulty mere mortals have in intuitively visualizing this probability space – a difficulty that it shares with a number of other mathematical models from science and engineering, including Cybernetics and statistical mechanics.

There is a positive and simplifying side to this sixth approximation, however. It only presents a single probability space with a single probability distribution. While the sample space is, itself, very complex, there is only a single distribution.

But, a unique advantage provided by this sixth approximation is the fact that, since there is only one probability distribution involved, then there is only one entropy value. Thus, the sixth approximation finally offers us a single measure of uncertainty for an entire Organodynamic web. We shall call this the *joint entropy* of the Joint Composite Organodynamic web approximation.

Uncertainty, Organization and Shannon Entropy

Shannon Entropy is Not a Measure of Disorganization

Shannon entropy is a measure of uncertainty/certainty. But it is not a measure of disorganization/organization.

It is easily seen that entropy is a measure of uncertainty of a space, because its *only parameter* is the set of *probabilities* of the sample points of the space of which it is a measure.

It is equally seen that entropy *is not* a measure of *organization* – or of *disorganization* – of a space, because there is no reference to interrelationships between the components of the space of which it is a measure.

Simply take another look at Shannon's formula for entropy:

$$H(x) = -\sum_{i=1,N} p_i \log p_i$$

The only input parameters to this formula are “ p_i ”, the probabilities of the components of the space. There is simply no mention – no accommodation – for anything having to do with any relationships that may or may not exist among or between those components.

In other words, even if the space upon which entropy is defined happens to have a relationship defined between its components, the entropy formula completely ignores those relationships!

And, in Organodynamics, we define the notions of *organization* and *disorganization* of a space in terms of the relationships between the components of that space.

Therefore, entropy does not measure organization or disorganization, because it has no accommodation for interrelationships among the components of a space – whether or not they exist.

Therefore, if we want to measure the *degree of disorganization* of the *degree of disorganization* of a space, *entropy* will not be it.

Sometimes the components of a space will be assigned probabilities – *and* there will be relationships defined on those components. Then, in this case it is appropriate to have *two* distinct measures for that space – a measure of *uncertainty/certainty* (using the probabilities and based upon Shannon's formula), *and* as second measure of *disorganization/organization*.

And, sometimes the components of a space utilize their *interrelationships* to form *arrangements* or *organizations*. Not only that, but sometimes it may even happen that these *organizations* are assigned probabilities. In this case, we happen to have a new space whose components are *organizations* – and one can measure the *uncertainty/certainty* of that new space using Shannon's formula. However, Shannon's formula *still does not* measure any relationships between members of that new space either.

So, given a space of components, there is nothing about the mathematics of their measure of *uncertainty* (Shannon entropy) that takes into account the fact that they may also be *organized*.

The Orthogonality of Organization and Uncertainty

As one can surmise, the degrees of uncertainty/certainty, and the degrees of disorganization/organization of a particular space can vary independently.

In the first place, there is the question of whether or not either of these measures can be defined on a space.

In order for a space to have a measure of uncertainty/certainty defined on it, it must be a probability space.

On the other hand, in order for a space to have a measure of disorganization/organization defined upon it (as defined in Organodynamics), its components must have a relationship defined between its components; and there must be some function defined on the product of that space that constitutes a measure of that relationship.

Clearly, these two capabilities can be defined for a space in a manner that is logically independent of each other. They are therefore orthogonal concepts.

In fact, Organodynamics has a requirement for defining these two concepts in an orthogonal manner so that they can vary independently of each other.

Specifically, Organodynamics uses defined relationships among the components of the systems that it deals with for the purpose of defining entities called *organizations*. Subsequently, Organodynamics assigns probabilities to those organizations.

It is observed that, in biological systems, there are biological entities that are 1) highly organized and highly uncertain, 2) highly organized and highly certain, 3) highly disorganized and highly uncertain, and 4) highly disorganized and highly certain.

Thus, Organodynamics has the need to define the concepts of disorganization/organization in a manner that logically independent from its definition of the concepts of uncertainty/certainty so that they can be allowed to vary independently.

Comment Regarding Entropy in Thermodynamics

I have just made the case that Shannon's entropy – the entropy of Information Theory – is a measure of uncertainty. This is clear because its only input arguments are the probabilities of the components of the system whose uncertainty it is measuring. And clearly, probability is inversely related to uncertainty. That is, the higher the uncertainty of an event, the lower its probability.

And, I have also made the case that Shannon's entropy (or information theory entropy) is *not* a measure of disorganization, of "arrangement", of "unstructuredness", or of "the dissipation of anything" because it contains no mathematical expression for interrelationships among spatial components.

An interesting question is "Can the same be said for *entropy* in thermodynamics.

My cautious guess is: "No" for the classical entropy of Clausius; and "Yes" for the entropy of statistical mechanics. "Yes" for the entropy of Boltzmann and Gibbs.

Thus, admittedly, the entropy of classical thermodynamic is certainly about dissipation of heat.

However, Gibbs equation for entropy is pretty much precisely the same as Shannon's. It is as though Shannon took Gibbs' formula and set the constant "k" to the value of 1.

So, Gibbs formula only contains probabilities as well. There is no mathematical accommodation for the interrelationships between molecules of an ideal gas in Gibbs' formula.

Certainly Gibbs knew those relationships are there. He invented his "ensembles" to capture those relationships. And those ensembles evidently inspired his formula for entropy.

And yet, there is no accommodation for those relationships in his entropy formula. It does not measure relationships. It does not measure the dispersion of anything – even if those things *do* have relationships, and *do* disperse. All of that is simply ignored by his formula.

If I am correct in these observations, then Gibbs formula *only* measures uncertainty.

But, what about the term “disorder”? Does Gibbs’ formula measure *disorder*? Well, if one *defines* the term disorder to mean *uncertainty*, then of course the answer is yes.

However, if one defines *disorder* to mean *degree of absence of interrelationship*, or *degree of absence of arrangement or organization*, then I must insist that the answer is “no”. Gibbs formula does not measure disorganization.

Thus, my first suspicion is that the entropy of statistical mechanics, like information theory entropy, is a measure of uncertainty – and that it is *not* a measure of disorganization (unless statistical mechanics *defines* disorganization to be uncertainty, as suggested to me by my friend Dave Steinkraus).

My second suspicion is that the popular usage of the term “disorder”, by scientists and others, most often carries the meaning “disorganization”, but occasionally carries the connotation of “uncertainty”. If true, then this has led to confusion.

Can some thermodynamicists please check me out on this??

Incremental Example Model Development

<>

Progress Check

In Part II, Organodynamics, we are gradually and incrementally developing a mathematical framework for modeling living and lifelike systems according to the seven organizing principles of OCS. Once completed, this modeling framework will achieve the form of an abstract but highly flexible mathematical structure that we are calling an *Organodynamic web*.

We are using the example of a hobbyist’s toolkit for building electronics as an analogy to Organodynamics. Both are systems for “building something”, and in both cases the resulting “thing to be built” has a network structure and has been given a general name.

In the case of the electronics circuit toolkit, the general name of the “thing to be built” is “circuit”. The circuits built by this kit are highly customizable to specific applications. In the case of Organodynamics, the general name of the “thing to be built” is “Organodynamic web”. The models built by this mathematical framework are also highly customizable to specific applications.

Because of the complexity of organic systems, any model that represents them needs to be built in cumulative and increasingly accurate stages. Each stage is generally referred to as an “approximation”. Organodynamics as a model builder’s toolkit provides for six of these approximations. These approximation levels are well defined and occur at precise places in the methodology set forth by Organodynamics.

In this chapter we have discussed how we characterize uncertainty within each of the six approximations of Organodynamics. In the first five approximations, we observed that a number of probability distributions exist – one for each time step of each edge, and at least one for each node (with multiple distributions for split nodes). Moreover, each of these distributions provides a single entropy calculation.

Thus, for each conditional stochastic model of each of the first five approximations, the uncertainty of the approximation is represented by a set of entropy values.

However, the sixth approximation provides only one single probability distribution and therefore only a single entropy value – the *joint entropy* of the sixth approximation.

Thus, in the sixth approximation, we have finally arrived at a single characterization of uncertainty for the entire Organodynamic web – at least for the conditional stochastic model.

None of the unconditional stochastic models have entropies, because they cannot be resolved to single distributions. And all of the realized deterministic models have entropies of zero – rendering them as trivial with respect to uncertainty.

Entropy Foundations and Constraints in Stochastic Processes

The Issue Addressed

The idea of *entropy* is more than simply an ancillary statistic of probability distributions. It actually goes to the foundation of probability and information theory. How this truth needs more embellishment than has so far been presented.

In addition, there are a number of fundamental misconceptions that are widely accepted within the research and practice cultures surrounding the notions of randomness, determinism, entropy and the long range behavior of probabilistic, or stochastic processes that need to be addressed in light of the mathematics of entropy and information theory.

OCS Organizing Principle Supported

Organizing Principle # 6: Uncertainty.

Biological Example of the Issue

The very nature of biological systems and their propensity toward persistence deserves an explanation in terms of the information theoretic concepts presented in this text.

Specific Challenges

To link the concepts of joint probability spaces, stochastic independence and dependence, asymptotically limiting distributions, relative entropy and mutual information in order to refute a number of misconceptions surrounding probability and stochastic processes.

The Organodynamic Approach to Modeling the Issue

We shall explain and apply the foundational relationships and results of information theory (as a branch of probability theory) to dispel a number of these misconceptions.

The Information Random Variable

We have been dealing with the fact that the amount of uncertainty inherent in a “situation” as modeled by a probability space before the situation has been “realized” is measured by the *entropy* as originally defined by [Shannon 1963]. As well, information theory has decided that the amount of information inherent in the same situation once it has been realized is equivalent, and is therefore also measured by the *entropy* calculation.

However, it can be shown that *entropy* is the average of a more primitive concept. We shall investigate this circumstance and articulate it in term of mathematical statistics in this section.

Random Variables

Throughout this text, we have been modeling organic systems with probability spaces. To develop such spaces, we began with some population of components (such as a set of atoms) and then considered the set of all possible ways that the population could be organized. This set of *organizations* then became the state space of the underlying population. We then assigned probabilities to each organization in the state space and the result was a probability space as characterized by its probability distribution.

Let's now ask a reasonable question about such a distribution: "What is its *mean*?" In other words, what is the average of this set of *organizations*?

Of course, it doesn't have an average – because organizations are not numbers. More precisely, our sample space of organizations does not have any arithmetic operations defined on it to enable use to perform the calculations necessary to compute a mean value.

For the same reason, we also cannot calculate its median, mode, variance, standard deviation or any of its *moments*. (These values are called *statistics*.)

This is not a new problem to experimentalist. Most of their sample spaces do not initially have arithmetic operations defined on them either: students, molecules, beers, pizzas, football players, etc.

However, usually there is some set of numbers that the researcher is very interested in and that can be associated with subsets of a sample space. These might be grams, meters, feet per second, height, IQ or other measurable attribute. And these *do* have arithmetic operations defined on them – because they are *numbers*. Because they are numbers, we can calculate *moments* and other statistics.

My point is that, unless one can map an experimental or theoretical sample space into the real numbers, then one cannot calculate any *statistics* on that sample space.

Keep in mind that one already has probabilities for the sample space; but *statistics* cannot be calculated unless there is a mapping from the sample space into the real numbers. The calculation of statistics requires both probabilities and number assignments to its sample points. If these real numbers are substituted for the sample points, then these various statistics, including the *moments*, can then be calculated. The whole discipline of statistics is centered around this.

The *moments* are important because, as an infinite sequence, they can completely characterize a probability distribution. Much of the theory of modern

probability theory is based upon the ability to develop the moments of a distribution.

Thus, mapping a sample space to a set of real (or complex) numbers is essential if we want to develop any statistics for the distribution.

Such a mapping from a sample space into the real number is called a *random variable*.

Generally, an early step in processing a probability distribution is to develop a random variable on a sample space, resulting in a set of numbers. The probabilities already associated with the sample space can then be projected onto these numbers – often after grouping the sample points and applying the numbers to these groups. This assignment results in a new probability space whose sample points are the real numbers that were generated by the random variable mapping, and whose probabilities have been derived from the mapping also.

The term *random variable* is formally used for this mapping; but it is also often used as the probability space, including the distribution, whose population is the codomain of the mapping.

Once one has developed a random variable in a probability space, it is then possible to develop statistics on it – particularly the *moments of the random variable*.

One can apply a good deal of *probability theory* with just a sample space and without a random variable. But one cannot “do statistics” on a probability space unless it has a random variable defined.

The Moments of a Random Variable

We have mentioned the notion of *moments* above as being an important set of statistics to be able to develop for a probability distribution.

And, again, moments cannot be developed for a distribution unless the distribution is against a random variable. For example, I cannot take the mean (average) of population of football players. However, I can take the mean of the heights of a population of football players, because *height* is a random variable on football players. I can also take the mean of the point scores of a set of football players, because *point scores* is another random variable on football players.

Because of the inherent complexity of probability distributions, it would be convenient to use simplifying mechanisms to characterize them. The principle mechanism used is called *moments of a random variable*.

Moments are an infinite sequence of real numbers, each of which characterizes a particular aspect of the distribution in question. Collectively the moments can be used to characterize the distribution as a whole or to essentially reconstruct the distribution.

These moments can be calculated easily by a simple sequence, $\text{Exp}(X_i^n)$. “Exp” is the expectation, or *expected value* operator. Thus, to calculate the n th moment of a random variable, raise each value of the random variable’s sample space to the n th power, multiply it by its probability, and then sum those products. Notice that the first moment reduces to the mean of the X_i s.

Of course, this calculation is only possible if the values of the sample space are real (or complex) numbers. The problem with *moments* is that they are not defined for all distributions because the sample spaces of many distributions are not real numbers; but rather entities of actual application domain spaces such as people, donuts, bacteria, etc., and do not have “averages”.

But *moments* are a very powerful “hammer”, and it would be desirable if we could turn all of these distributions into “nails” – if we could convert non-numeric sample points to numeric ones.

This is the purpose of the *random variable*.

Domain-based Random Variables

Thus, in experimental statistics, one of the first steps taken is to identify one or more random variables on an underlying population.

The field of sports statistics is a case in point. It is a veritable study in random variables. Often, the underlying population in sports statistics is a team of athletes. There is no shortage of random variables – all of which make sense within the application domain. That is, these random variables are semantically significant to the domain. For example, here are some reasonable random variables often applied in sports statistics: points per game, runs batted in, weight, height, etc.

We call this type of random variable “domain-based”, because the semantics of the application domain dictates them. Domain-based random variables do not “travel” well from domain-to-domain. For example, “runs batted in” (RBIs) is a random variable that works fine for American baseball, but not for the IQs of school children.

Semantically Insignificant Random Variables

Thus, for any sample space, there is always a search to find one or more random variables for the underlying population.

Sometimes, the population itself is already a set of numbers – and is already a random variable. This is very convenient.

However, often there is no meaningful domain-based random variable for an underlying population. This occurs quite often in experimental statistics – especially when measures of dependency and correlation are being attempted between two variables.

Many of these dependency tests require that both of the variables involved be placed into some kind of linear order to be able to perform these tests of dependency.

This kind of practice results in the arbitrary mapping of sample points to the real numbers when there is no semantically significant mapping like that in the domain of the experiment.

However, this kind of practice is encouraged in experimental statistics in order to enable the application to avail itself to the random variable orientation of contemporary statistical tools.

The Information Random Variable

So, there are some probability spaces for which there is no semantically significant domain random variable. And to invent one merely for the sake of being able to apply the concept of *moments of a random variable* is specious, if not intellectually dishonest.

However, this predicament could be avoided if there were some random variable that actually *made sense for all probability spaces* – no matter what their domain of application.

Of course, such a random variable would not be domain-based. Lets categorize such a random variable – if one can be found – as *inherent*. It would be a random variable that is inherent in every probability space, no matter what the domain of application is.

But, what property of a probability space is inherent in all probability spaces? The answer is:

Uncertainty!

Therefore, we could find what we need if...

We could find a *random variable* that maps any event of any sample point of any probability space to a real number in such a way that it represents the *uncertainty* of that event.

The Uncertainty Random Variable

But we already have such a random variable!

We developed it in the previous chapter. It is our measuring function of the degree of uncertainty of a single event. We called it the *uncertainty measure* of an event, and we defined it as:

$$U(E) = \log (1/\text{Pr}(E)) = -\log(\text{Pr}(E))$$

That is, the uncertainty of an event E is $-\log(\text{Pr}(E))$.

This is “certainly” a random variable because, given any probability distribution, it maps each sample point to a real number. We know this is true because any sample point is an event (E) and $-\log(\text{Pr}(E))$ is a real number.

Thus, we have identified an inherent random variable – the *uncertainty random variable* U(E).

U(E) applies to all probability spaces. And, it pertains to the uncertainty associated with each of the sample points of a probability distribution. And all probability distributions concern uncertainty.

The Information Random Variable

But, recall that, in Information Theory, the amount of information in an event is the same as the amount of uncertainty inherent in that event.

That is, the *operational definition of information* in Information Theory is

$$I(E) = U(E) = -\log(\text{Pr}(E)).$$

The more uncertainty there is, the more information there is.

Since, in Information Theory, *uncertainty* and information are measured by the same function, then we can name our new inherent random variable either the *uncertainty random variable* or the *information random variable*.

Let's choose the name *information random variable* or IRV.

Thus, the IRV is the random variable that applies to all probability distributions – because all probability distributions have some measurable degree of uncertainty. As a random variable, the IRV maps each sample point of a probability space to its degree of uncertainty.

But also, being a random variable, the IRV allows us to apply the notion of *moments*. And since the IRV applies to all probability spaces – even the ones that do not have any meaningful domain-based random variables, then the IRV allows us to apply the notion of *moments of a random variable* to all probability spaces.

Moments of the Information Random Variable

The question arises “What is the IRV telling us about a probability space to which we apply it?”

The answer is “uncertainty”. It is telling us how uncertainty (or *information*) is held and distributed about that probability space. The IRV and its moments characterize the degree of uncertainty that is inherent in a probability space – any probability space.

And, the mechanism that the IRV uses to characterize the nature of the uncertainty inherent in a particular distribution is via its *moments*. The moments

of any random variable are used to characterize the distribution that those moments are calculated against.

So, our purpose in developing the IRV is to be able to use its moments with respect to any probability distributions in order to characterize how it holds and distributes *uncertainty*.

Moreover, the IRV, by its very nature, has a normalizing affect on all probability spaces. So we can use the IRV and its moments – not only to characterize a probability distribution – but also to compare and contrast the degrees of uncertainty of any two distributions – without any further need to normalize them in some other manner.

We say that the first moment (the mean) is most important only because it is used to derive a second version of all the others. This second version is called the *central moments*. Typically, experimenters use the first moment, and then use the central moments of all of the others. For example, the variance is the second central moment. It is calculated as the second moment of the random variable minus the mean. The remaining central moments are also calculated against the random variable minus the mean.

In any event, the first moment and the remaining central moments should be calculated as a general expression that can then be readily applied to any probability space.

In order to use the IRV to do all of this, general expressions for the calculation of its moments for any probability distribution must be derived. We shall not go into this derivation at this time. However, we shall however, derive the most important of these moments – the first moment, also called the *mean* of the random variable. We shall leave the derivation of others as an exercise.

However, we shall now briefly look at the derivation of a general expression of the first moment – the *mean of the IRV*.

The Mean of the Information Random Variable

Let's calculate the first moment (the mean μ) of the IRV to see what we get.

From above, the formula for the first moment is:

$$\mu = M_1(X) = \text{Exp}[x_i^1]$$

Recall that we calculate the mean by multiplying each *value* of a random variable by its probability, and then summing those products across all of the values of the random variable.

We shall express this as:

$$(2) \quad \mu = M_1(X) = \sum_{i=1}^n ([x_i^1 * \text{Pr}[x_i]])$$

$$= \sum_{i=1}^n (x_i \cdot \text{Pr}[x_i])$$

Since IRV is a random variable, then, given any event E in the underlying sample space, the IRV maps it to a new value in the new sample space. This new value, x_i , is $-\log(E_i)$. That is, in the above expression (2),

$$x_i = -\log(\text{Pr}[E_i]), \text{ and}$$

$$\text{Pr}[x_i] = \text{Pr}[E_i],$$

since the random variable inherits its probabilities from the original distribution.

So, to calculate the first moment of the IRV, we substitute for x_i and $\text{Pr}[x_i]$ in the formula at (2). We get the following result:

$$M_1(\text{IRV}) = \sum_{i=1}^n (-\log(E_i)) \cdot \text{Pr}[x_i], \text{ So}$$

$$M_1(\text{IRV}) = -\sum_{i=1}^n (\log(E_i)) \cdot \text{Pr}[E_i]$$

So, the first moment of the IRV is $= -\sum_{i=1}^n (\log(E_i)) \cdot \text{Pr}[E_i]$.

This expression should look very familiar.

It is the definition of the Shannon's *entropy* of the underlying probability distribution of which the IRV is a random variable!

In other words, the *first moment* of the IRV is the *entropy* of the underlying probability distribution!

This tells us a lot about what entropy really means:

The entropy of a probability distribution is the average amount of uncertainty (or information) of each of its sample points!

This is why we say that *entropy* is defined in terms of something more primitive: the uncertainty measure of its individual sample points. In fact, entropy is the average of those individual uncertainty functions.

The next step in this investigation would be to calculate the second, third, fourth and other central moments of the IRV (the variance, skewness, kurtosis, etc.) of the IRV. These would further characterize distributions pertaining to how they hold and distribute uncertainty and information. Again, we leave this as an exercise for further research.

The Information Random Variable (IRV) in Organodynamics

The information Random Variable (IRV) is an excellent tool for completely analyzing, characterizing and comparing the uncertainty and information characteristics of general probability distributions.

The sixth organizing principle of OCS, *uncertainty*, states that lifelike systems and processes are characterized by uncertainty. Therefore, the use of the IRV is presented by Organodynamics as the mechanism of choice for characterizing uncertainty – and information – in organic system processes.

The Applicability of Uncertainty and Information to Organic Systems

There is considerable folklore within the community of science philosophy and practice that has led to some untrue conclusions regarding chance and determinism.

One of these untrue conclusions is that *chance phenomena cannot lead to the origin of life*. The belief is that the origin of life is too improbable to have occurred by chance.

A second untrue conclusion is that *random processes cannot produce stable long-run behavior*, and that they must devolve into chaos. A corollary to this piece of misinformation is that *any theory of the origins of life that involves chance cannot explain the orderliness of biology*, because chance piled upon chance inevitably leads to chaos, and biology is not chaos.

A third untrue conclusion is entropy *can never* decrease because that would violate the second law of thermodynamics. A corollary to this is that *biological life can never have occurred, because it violates the second law of thermodynamics*. Since this is clearly untrue, many scientific explanations of why have been offered.

Below, we have provided explanations as to why all three of these conclusions are untrue. These explanations are all strictly grounded in mathematics. In fact, we shall rely exclusively on tenets from *information theory* as a sub-discipline of *probability theory*.

In this section, we shall add a few principles from information theory that we have not already discussed in order to impress upon the reader the futility of the above three conclusions.

Chance Phenomena Can Lead to the Origin of Life

[Hoyle and Wickramasinghe 2001] claim that the probability of a protein molecule forming as a product of "chance" is so infinitesimally small that it could never have occurred within the life of the universe.

However, their method of calculation exposes their unwitting assumption of stochastic independence – an assumption that is neither grounded nor plausible. Their assumption of stochastic independence is revealed by their *use*

of multiplication of the probabilities of the joint events involved in growing a polypeptide chain!

If adding a new amino acid to a growing chain of a polypeptide is truly statistically independent of the amino acids that preceded it in the chain, then Hoyle and Wickramasinghe are correct. However, if they are incorrect – if the additions of new amino acids to a growing polypeptide chain are dependent events - then their calculation is unreasonably large.

However, it is clear that the addition of an amino acid to a growing polypeptide chain depends upon the chain being co-located with the amino acid molecule. The conditional probability that the required peptide bond occurs, given that the polypeptide chain is co-located with the amino acid molecule is different from the probability of the amino acid forming a peptide bond without that assumption. This situation is the very definition of conditional probability.

And, when the conditional probability of an event is different from the unconditioned probability of that event, then the two events are stochastically dependent. In fact, this difference is the definition of stochastic independence.

Therefore, our conclusion is that the (perhaps unconscious) assumption on the part of Hoyle and Wickramasinghe that these events are stochastically independent is not warranted. Thus, their calculation using multiplication is incorrect.

The real problem with this assumption of statistical independence on the part of Hoyle and Wickramasinghe is not what it says about the probability of any specific event (such as the spontaneous formation of a protein), but rather about the probabilistic characteristics of the entire distribution of events (including the protein). The real issue is what this assumption says about the *entropy* of that distribution – about its inherent degree of uncertainty.

In information theory, the notion of statistical independence (or statistical dependence) is cast as a relationship between two variables. (In this case it involves the bonding of a polypeptide chain and an amino acid molecule.) To do this, information theory creates a new probability distribution that is a combination of the two whose independence is being discussed. This is called the *joint distribution* of the two variables.

Information theory says that if the two variables are stochastically independent, then the entropy of their joint distributions (their *joint entropy*) is at its maximum value. If, on the other hand, they are stochastically *dependent*, then their joint entropy must be less than that.

This is stated by the well-known and fundamental relationship in information theory that is expressed as follows [Kleeman2 2009]:

$$H(X_1, \dots, X_n) \leq \sum_{i=1, n} H(X_i), \text{ where } H \text{ is the entropy calculation.}$$

The fundamental relationships and concepts in information theory are primarily focused on a single phenomenon: that *stochastic dependence of two variables reduces entropy as compared stochastic independent*.

This propensity of stochastic dependence to reduce the degree of uncertainty shows up in a special entropy measure - named *mutual information* - that is applied to *pairs* of random processes. Mutual information is defined so that it is a reverse entropy measure. Whenever the two random processes are stochastically independent, their mutual is at the minimum. The more they are stochastically dependent, the larger becomes their *mutual information* measure. Mutual information is defined on a more primitive entropy measure named *relative entropy* [Kleeman 2009].

We want to be able to argue that the probability of a protein forming by chance can be higher than expected by Hoyle and Wickramasinghe. Their argument is effectively that the probability of spontaneous protein formation is the same as the probability of all other possible events in the same probability space – because the joint distribution is uniform. This follows from their assumption of statistical independence. However, since this space is so astronomically large, then this probability must be astronomically small.

By arguing against stochastic independence in this space, we are arguing that the probabilities of the joint distribution are not equiprobable, and that therefore, some of the probabilities are larger than the others. Some are smaller also. But we have to show that spontaneous protein formation is larger than the mean.

Actually, we do not actually have to prove that protein formation is larger than the mean, because we have already proved our point – that Hoyle and Wickramasinghe have not proved theirs. We have shown that Hoyle and Wickramasinghe have not proved that spontaneous protein formation is not as improbable as they claim, and that their argument can be disregarded as out-of-hand.

So this fact of statistical dependence can reduce, even dramatically, the degrees of uncertainty associated with this growing chain - as compared with the case that they are statistically independent.

Random Processes Can Produce Stable Long-run Behavior

There is a widely accepted misconception that one cannot begin a process with randomness and expect it to do anything but increase over time. In other words, initial randomness must lead to eventual chaos.

This idea is refuted by information theory.

The information theory articulation of “random process” is the stochastic process, which we have worked with at length in this text.

In information theory, stochastic processes can exhibit varying degrees of stability over time. In many ways, information theory is the study of the conditions under which a stochastic process behaves chaotically or with

stability. Many of its fundamental relationships and theorems are concerned with this. We shall look at a couple of situations to demonstrate these relationships and behaviors.

Markov Chains

Lets first look at the simplest stochastic processes – irreducible Markov chains. We have already demonstrated the points to be made here in previous chapters, so we will merely summarize them here.

We have shown that irreducible Markov chains have a *stationary distribution*. This is a probability distribution that is a limiting distribution. That is, in the long run, the Markov chain approaches asymptotically a single distribution. That is, its long-run behavior is characterized by this distribution. In fact, no matter what the initial conditions are, the same stationary distribution is the limiting long run behavior for a Markov chain.

And, since the stationary distribution has finite entropy, then long run behavior of an irreducible Markov chain exhibits limited long run uncertainty. Therefore, all irreducible Markov chains exhibit long run stability.

But so do all homogeneous Markov chains. If they are not irreducible, then we cannot calculate their asymptotic distributions, and therefore we cannot calculate their entropies. But nevertheless they exist and they are finite.

Therefore, all homogeneous Markov chains exhibit long run stability.

Other Stochastic Processes

In information theory, the concept of *entropy rate* is used to characterize the change in long run uncertainty of a stochastic process as it grows time step by time step.

In statistics, we are interested in how the average statistic of a group is altered by the addition of more members to the group – members who express the same statistic on an individual basis.

The idea of the *entropy rate* is to do a similar thing for the statistic for *uncertainty* (i.e. entropy) as we add more terms (time steps) to a growing stochastic process. In other words, as a stochastic process gets longer and acquires additional times steps, does the “average entropy” increase? If not then we can say that the process exhibits certain stability.

More precisely, the entropy rate is the starts by considering each time step of the stochastic process as a distinct probability space and then forms the joint probability distribution of that space. Next, compute the entropy of that joint distribution. Finally, to calculate the *entropy rate* of the stochastic process, take the limit of these computations as the number of terms goes to infinity.

In symbols,

$$\text{Entropy rate} = \lim_{n \rightarrow \infty} 1/n * H(X_1, \dots, X_n)$$

Says Richard Kleeman of the Courant Institute at New York University:

The entropy rate is the additional uncertainty to the whole chain introduced by adding another random variable and in general this will be less than the entropy of this particular variable by itself since there is dependency between the random variable then and earlier random variables in the chain [Kleeman3 2009].

It is possible then, contrary to the expectations of many who are unfamiliar with the workings of information theory, that chance phenomena can lead from uncertainty to stability over time.

Entropy Can Decrease

Before we begin this explanation, it will be helpful to remind the reader of what entropy measures - and what it doesn't. The reader should be reminded that an increase in entropy *does not define* an increase in *disorganization*. Rather it is an increase in *uncertainty*, or unpredictability.

For example, a system may continue to be highly organized while its entropy increases. This would be the case if it were highly uncertain as to which of its possible organizations were going to manifest at the next moment – even if all of its possible arrangements are highly organized.

Let us now turn our discussion to the main point of this subsection. Which is that:

Entropy can be decreasing in some stochastic processes - as opposed to the fact that it must be non-decreasing in statistical mechanics. Indeed, entropy can decrease in organic processes.

The reason that entropy can decrease in some stochastic processes is explained eloquently by Richard Kleeman in the third lecture of his lecture series on Information Theory and Predictability [Kleeman 2009]. I will attempt to summarize it here.

We have seen above that certain stochastic process move asymptotically to a limiting probability distribution. The example we presented above was the simplest case: an irreducible Markov chain. In these processes, the limiting, or asymptotic distribution, is called the *stationary distribution*.

It turns out in the theory of stochastic processes that there are more general cases of this phenomenon of limiting asymptotic distributions than just the irreducible Markov chains. Here, we shall also refer to these asymptotically limiting distributions as *stationary distributions*.

Suppose we have one of these distributions and further assume that the stationary distribution is something other than the uniform distribution. Suppose further that the initial conditions vector of the stochastic process in question *is* the uniform distribution. Clearly, both of these assumptions are possible.

But, the entropy of the uniform (initial conditions) distribution is the maximum value, and therefore has a higher entropy value than the stationary distribution in this case. Consequently, we have demonstrated that it is possible for a stochastic process to exhibit decreasing entropy.

However, this does not disprove the second law of thermodynamics. This is because, in that domain, the stationary distribution is the uniform distribution (because of the assumption of equilibrium dynamics). Thus, in such a case, the limiting distribution has maximum entropy. Thus, under those assumptions, the entropy must be at least as high as at any time step in the process.

General Misconceptions

At the root of all three of above untrue conclusions lay three false assumptions regarding uncertainty and information.

These assumptions are usually unconscious on the part of a proponent. If the proponent is called out on the use of any of these misconceptions, he usually denies subscription to it at that point. However, when his underlying arguments are inspected closely, the misconception is often spotted forthwith.

Here are the three misconceptions.

1. Randomness and determinism are mutually exclusive explanations.
2. Randomness implies a situation where all alternatives are necessarily equally likely.
3. Any two random phenomena are statistically independent.

Information theory refutes all three of these assumptions. In fact, considerable space has already been allotted in this text to the refutation of them.

The entire notion of entropy refutes 1) and 2). As can be seen by inspecting any text on Information Theory, the bulk of that discipline concentrates on what occurs when joint probability spaces are mutually statistically dependent. Essentially this result is that uncertainty is reduced as compared with when statistical independence holds sway. Thus, much of information theory refutes 3).

Incremental Example Model Development

<>

Progress Check

In Part II, Organodynamics, we are gradually and incrementally developing a mathematical framework for modeling living and lifelike systems according to the seven organizing principles of OCS. Once completed, this modeling framework will achieve the form of an abstract but highly flexible mathematical structure that we are calling an *Organodynamic web*.

We are using the example of a hobbyist's toolkit for building electronics as an analogy to Organodynamics. Both are systems for "building something", and in both cases the resulting "thing to be built" has a network structure and has been given a general name.

In the case of the electronics circuit toolkit, the general name of the "thing to be built" is "circuit". The circuits built by this kit are highly customizable to specific applications. In the case of Organodynamics, the general name of the "thing to be built" is "Organodynamic web". The models built by this mathematical framework are also highly customizable to specific applications.

Because of the complexity of organic systems, any model that represents them needs to be built in cumulative and increasingly accurate stages. Each stage is generally referred to as an "approximation". Organodynamics as a model builder's toolkit provides for six of these approximations. These approximation levels are well defined and occur at precise places in the methodology set forth by Organodynamics.

In this chapter, we have looked more closely at how information theory and its concept of *entropy* can bring a strong and rich theoretical foundation to Organodynamic framework. Moreover we saw how the concept of the *information random variable*, or IRV, can bring broad foundation to the discipline of statistics by providing a random variable pertaining to the notion of uncertainty and its concomitant *moments series* that is applicable to all probability spaces in a manner that allows comparison across the entire space.

In addition, we saw how the foundational relationships and theorems of information theory provide an approach to dependency relationships between and among probability spaces that characterizes the reduction of uncertainty that is at work as dependency increases. These theoretical dynamics also explain the conditions under which stochastic processes either eventually settle down to stability or wander off into chaos.

Furthermore, using these relationships, we were able to show how the second law of thermodynamics is only guaranteed if the limiting asymptotic distribution is uniform – as is the case in statistical thermodynamics. However, in certain other cases – as often presented by biological systems – relative entropy can actually decrease over the life of the process.

Emergence

This chapter discusses the incidence of emergence in Organodynamics.

The Issue Addressed

In Part I, *Organizing Principles*, two types of emergence are distinguished: *process emergence* and *systemic emergence*. It was also noted that OCS emphasizes systemic emergence.

OCS Organizing Principle Supported

Organizing Principle # 3: Emergence.

Biological Example of the Issue

Any biological system exhibits properties that are not exhibited by any of its components.

For example, a cell has a metabolism while none of its macromolecules do. Virtually any biological system, no matter how simple or complex, can perform functions or exhibit other systemic properties that none of its components do or can.

This fact is the essence of the popular phrase “the whole is greater than the sum of its parts”.

It is true that there are non-organic systems for which this is also true. However, the second OCS organization principle, *emergence*, does not say that organic systems are the only ones that are emergent. It simply states that they are.

Specific Challenges

We must define the distinction and relevance of *systemic* versus *process* emergence. We must address the possibility that emergence is, in some sense, inevitable, or nearly inevitable, and consider the circumstance under which it occurs.

The Organodynamic Approach to Modeling the Issue

Distinguish between two types of emergence, and define them as properties of systems.

Systemic Versus Process Emergence

OCS is interested in both types of emergence, but emphasizes *systemic emergence*. The two often occur together. That is, it is often that an emergent property is both systemically emergent and at the same time process emergent. These two are explained and distinguished in this section.

Process Emergence

Process emergence, which we shall also call *historical emergence*, probably enjoys more frequent usage in scientific literature. Process emergence is the appearance of a property – the exhibition of a property – by a system at some point in time, in which the property had not been exhibited by that system before that time.

More than likely, most scientists use the term *emergence* to mean what we are calling *process emergence*.

Systemic Emergence

OCS identifies, and names, a second kind of emergence – *systemic emergence*. Systemic emergence is the exhibition of a property by a system when none of its components also exhibit the same property.

Systemic emergence can be credited with being responsible for the phrase “the whole is greater than the sum of its parts” as applied to systems in general. This is certainly one of the reasons that OCS emphasizes systemic emergence.

Another interesting distinction between systemic and process emergence is that systemic emergence can be observed within a single time step of a dynamical process. All one needs to do is to determine whether the system – within that time step – exhibits a property that none of its components exhibit.

However, to ascertain whether a system is *process emergent*, one must find two time steps within its system process such that the later of those two steps exhibits the property in question, whereas the earlier does not.

In this sense, *systemic emergence* is a static property of a system; whereas, process emergence is a dynamic property of a process.

The Practical Inevitability of Systemic Emergence

The very notion of *system* seems to hold a promise of “the whole being greater than the some of its parts”. That is, systems seem to inherently have more to them than their collections of components.

The notion of systemic emergence speaks to this by discussing a property that the system (“the whole”) can exhibit that none of its parts (components) do.

So, the question arises, “Must a system necessarily exhibit a property that none of its components do?” If so, then by definition all systems must be emergent systemically (even if only in some trivial way).

The notion of *systemic emergence* is tantalizing because it holds out the prospect that all systems may be systemically emergent – or nearly so.

Are All Systems Systemically Emergent?

So, the question we want to ask is “Does a system necessarily exhibit some property that none of its component systems exhibits?”

Without loss of generality, we shall work with the case of nested systems. If we can prove them to be systemically emergent, then all systems are.

We shall show that under “normal circumstances”, a composite system is an emergent system. However, there is a “strange circumstance” under which it is not. We shall identify it now.

In the first place, in order for a composite system to be emergent (systemically), there must be at least one *true statement* (property) that one can make about it that cannot be made about any of its component systems.

Consider two systems, A and B, where B is a component of A. These two systems are different systems if and only if there is some true statement that can be made about one that cannot be made about the other.

Obviously, if A and B are the same system, then A will not be systemically emergent, because it will then have a component that exhibits all of its properties. The converse is also true. If A has a component system B that exhibits all of the properties of A, then $B = A$. To see that this is true, suppose B has some property that A does not. Then the negation of that property would be a property. And therefore, A would have a property that B does not – which violates our assumption.)

Thus, a composite system can be emergent systemically if and only if it cannot be a component system of itself.

Thus, we must find out if there is some reason why no composite system can ever be a component of itself. And, if it can, then we must find out the circumstances under which it can be a components system of itself.

Recall, that – by definition - for system $B = (P_B; O_B)$ to be a *component system* of composite system $A = (P_A; O_A)$, O_B must be an element of P_A . That is, the *organization* of a component system is the *component* of a composite system.

Lets assume that a composite system can be one of its own components – and then see if this assumption leads to any contradiction. On the other hand, we may ascertain instead that it is possible to construct a composite system that has itself as a component system.

It turns out, in fact, that we can actually do the latter. Yes, it is possible to construct a particular composite system that can be its own component system. However, it is a very unusual system – one that shall not soon be seen in biology.

Furthermore, other systems that are not like this one (with respect to a particular circumstance) cannot be their own component systems - and thus do not exhibit precisely the same set of systemic properties.

Thus, there must be at least one systemic property that the system and its component systems exhibit that the other does not. If the composite system exhibits it, then it is a systemically emergent property. If at least one of the components exhibits it, then its negation is exhibited by the composite system. In either case, the composite system exhibits a property that its component systems do not.

Therefore, unless this “strange circumstance” holds, then composite systems are emergent systemically.

Strange Circumstance Under Which Composite Systems are not Emergent

We have made the case that composite systems are emergent systemically – except in a certain “strange circumstance” that virtually never holds for composite systems that we are generally interested in.

We need to know what the circumstance is, so that we can avoid those types of composite systems. Or, at least, if we can't avoid them, we will know that they cannot be emergent systemically.

So, let's now construct a composite system that actually is its own component system. This will show us the circumstances under which a composite system is not an emergent system. All other composite systems are emergent.

First, notice that if a composite system is its own component system, then (as a component system) its organization is an element of its population (as a composite system).

If we can find some conditions under which the organization of a composite system can be a component of its population, then we will have discovered the conditions under which a composite system can be its own component – in which case it is not emergent because a component system (itself) has all the same properties as the composite (itself). Moreover, will have found the conditions under which a composite system is not emergent. All other composite systems will of necessity be systemically emergent.

So, let's construct a general counter example. We want to find a system whose organization is one of its components. By definition, this will make it a composite system of itself. Consider the following system. (The symbol Φ represents the empty set.):

$$\begin{aligned} \text{Let } S &= (P; O), \text{ where} \\ P &= \{ \Phi \} \\ O &= \Phi. \end{aligned}$$

Notice that we have used the fact that the organization of a system may contain no duples (is Φ), representing the case that the system is completely unorganized. This of course is a degenerate case, but for completeness, we include in Organodynamics.

$$\text{Thus, } S = (P; O) = (\{ \Phi \}; \Phi).$$

But, we can use the Integrate transform to create a composite system $S' = (P'; O')$ using S as its only input (component) system. The population of S' by definition is the collection of all of the organizations of the input (component) systems. But there is only one such organization involved and that happens to be the set $O = \Phi$. Thus, the population P' of the composite system being created must be the set whose only element is Φ . Therefore, $P' = \{ \Phi \}$.

Now we must construct O' . We can use any collection of ordered pairs of P' that we wish to construct O' . For example, we could use the pair (Φ, Φ) as the only pair involved, in which case O' would be the set $\{(\Phi, \Phi)\}$. But we can use any other set of duples, including the set with no duples, Φ . In fact, we are going to choose Φ as our choice of O' – just because we can, and because we want to.

Thus, our newly constructed composite system is

$$S' = (P'; O') = (\{ \Phi \}; \Phi).$$

But this is precisely our initial component system S that was the only input to our Integrate transform.

Thus, $S' = S$. And we have provided an example of a system being a component of itself. And, such a system is not an emergent system, because it has no property that none of its components do.

However, any system that does not have itself as a component *is* an emergent system, because there is some statement that can be made about it that cannot be made about any of its components systems.

Certainly, most, if not all, biological systems will never have themselves as components, and will therefore be emergent composite systems. Moreover, notice that the counter example we provided above had the property that its organization is empty. In other words, it is completely disorganized.

One could argue that we do not want to allow such a set to be counted as a “system”, because the whole idea is that a system is not a system unless it is organized. Thus, we could disallow a system’s organization to be Φ . If we were to make this stipulation, it would result in all systems being emergent. And this entire subsection would be unnecessary, because we would have ruled out, by definition, the bothersome and unusual circumstance that we have been discussing.

However, there may be some abstract space in which it may be desirable to permit composite systems that can be their own component system. For these, then, we can say that those particular composite systems will never be emergent systemically, while all the rest must be.

The General Inevitability of Emergence in Nested Systems

For most “mortal” organic systems, however, we can assert that no composite system can be a component system of itself, and will therefore be systemically emergent.

Opportunities for and Likelihood of Systemic Emergence

In Organodynamics, new systems are formed during the performance of Organodynamic transforms. During this formation, the two parts of a system – tis *population* and its *organization* come into being.

The populations of these newly formed systems are generally determined by the specific transform involved. However, the formation of the *organizations* of these newly formed systems is defined in Organodynamics as a stochastic event.

This means, that, for each organodynamic transform, a collection of possible *organizations* of the determined underlying population is provided. Generally, in organic systems, this collection of organizations is expected to be large. Certainty it is in biological systems.

This generally large collection becomes the underlying sample space and has an associated probability distribution that describes the probabilities of which of these organizations will be realized.

And it is this large set of system organizations that enables the system to differentiate itself from its components.

These organizational differences, then, provide a field of opportunities and likelihood for systems to exhibit systemic properties that are distinct from their component systems.

Counter-emergence

Non-emergence can provide interesting systemic properties as well as emergent conditions. In this section, we shall look at one such circumstance.

Fractals Patterns

It can occur that a subsumed system of a nested system can have an organization that is “very much like”, or “exactly like” the organization of its composite system – or of one of its component systems.

Such a circumstance can be well defined by a notion of *isomorphism* between system organizations. For example, it would be useful to define the following two system organizations to be *isomorphic*:

$$\begin{aligned} A &= \{ [a, b], [a, c], [b, d], [d, c] \} \\ B &= \{ [w, x], [w, y], [x, z], [d, z] \} \end{aligned}$$

The mapping $f:A \rightarrow B$ where $f(a) = w$; $f(b) = x$; $f(c) = y$ and $f(d) = z$ would hopefully be an “isomorphism” under such circumstances.

We would like to point out that such a notion of “isomorphism” could be used to define a “fractal-like” relationship among subsumed systems of a nested system hierarchy.

This type of approach can be developed along many lines of relationship in order to characterize nested system hierarchies within Organodynamics. This line of investigation is suggested as topics of further research.

Process Emergence

Recall that Organodynamics distinguishes between *systemic emergence* and *process emergence*. We have dealt extensively with systemic emergence above; and now we shall turn to the treatment of *process emergence*, also called *historical emergence*, in Organodynamics.

Recall that *process emergence* is the occurrence of a systemic property at some time step of an organic process whereas the same systemic property was not exhibited by a previous time step of the same process.

The point of this subsection is this:

The continual transitions between involving various degrees of certainty and uncertainty afforded by the sixth organizing principle of OCS promotes the incidence of novelty, and thus of process emergence.

Let's elaborate on some of the factors that promote this novelty and eventual process emergence. First, the essential nonhomogeneity (as exhibited by piecewise-homogeneity) of Organodynamic webs results in continued change in the underlying organic system populations as well as their sample spaces of system organizations. And, along with these changing sample spaces is the concomitant change in probability distributions.

All of this sets the stage so that the probability of the occurrence of novel system organizations for at least some time steps becomes highly likely. It is these novel system organizations that necessarily exhibit novel systemic properties – and thus incidences of *process emergence*.

Consequently, we conclude that historic emergence is expected in Organodynamics.

Incremental Example Model Development

<>

Progress Check

In Part II, Organodynamics, we are gradually and incrementally developing a mathematical framework for modeling living and lifelike systems according to the seven organizing principles of OCS. Once completed, this modeling framework will achieve the form of an abstract but highly flexible mathematical structure that we are calling an *Organodynamic web*.

We are using the example of a hobbyist's toolkit for building electronics as an analogy to Organodynamics. Both are systems for "building something", and in both cases the resulting "thing to be built" has a network structure and has been given a general name.

In the case of the electronics circuit toolkit, the general name of the "thing to be built" is "circuit". The circuits built by this kit are highly customizable to specific applications. In the case of Organodynamics, the general name of the "thing to be built" is "Organodynamic web". The models built by this mathematical framework are also highly customizable to specific applications.

Because of the complexity of organic systems, any model that represents them needs to be built in cumulative and increasingly accurate stages. Each stage is generally referred to as an "approximation". Organodynamics as a model builder's toolkit provides for six of these approximations. These approximation levels are well defined and occur at precise places in the methodology set forth by Organodynamics.

This chapter has explained how Organodynamic webs exhibit both systemic and process emergences. The implications of this chapter are that both of these forms of emergence are essentially inevitable in Organodynamic webs.

Persistence

The Issue Addressed

Persistence is the seventh of the OCS organizing principles. This chapter contemplates how this principle arises in organic systems and its relationship to the other six organizing principles.

OCS Organizing Principle Supported

Organizing Principle # 7: Persistence.

Biological Example of the Issue

If a biological system were to come into existence and then expire immediately, no biologist would regard it as ever having lived. It would have been “born dead”. DOA.

On the other hand, no biological organism has ever been known to live forever – or to have had the opportunity. It is presumed in the biological sciences that all organisms that live die. The same is presumed for all species, and for life itself.

Thus, it is presumed that biological life persists for some time – but not forever.

This is also true of non-biological entities – such as rocks. Thus, persistence is not the sole criterion for being lifelike – but is a criterion.

What makes Persistence different is that it appears to be the *effect* – even the *consequence* – of the other six OCS organizing principles – or at least of some interplay of the others.

Specific Challenges

In order to study the interplay and dynamics among the seven OCS organization principles as they are implemented within the Organodynamics framework and its grand scheme the Organodynamic web, it will be necessary to isolate them from the Organodynamic web model itself – at least for while.

The challenge is to construct a meta-model so that it can advance an understanding of how the interplay of the OCS organizing principles can result in the Persistence of Organization.

The Organodynamic Approach to Modeling the Issue

In this chapter, we shall develop a meta-model of an idealized Organodynamic web whose “components” are all of the OCS organizing principles except for Persistence. The plan is to determine what conditions must obtain in order for Persistence to emerge as a property of the meta-model.

Specifically, we shall construct this model as a Markov chain. The reader shall be encouraged to play “what if” with the transition matrix of this chain in order to detect the condition of the chain that result in either persistent or nonpersistent long run behavior of the meta-model.

The risk with this approach is that the resulting meta-model will be so abstract and so far removed from actual organic systems – and from our Organodynamic web model, that it can be easily misinterpreted.

The intention, however, is that the meta-model can be profitably used as a thought experiment to accrue an enriched understanding of the interplay among the OCS organizing principles as they are implemented within the Organodynamic framework, and how they interoperate to result in the limited persistence that is characteristic of lifelike systems.

The Persistence of Organization

The essence of *livingness*, as understood by OCS, was described at the beginning of OCS Part I as:

Self-perpetuation of organization within change through change

Part I then went on to elaborate this as follows:

In other words, livingness is about the self-perpetuation of organization. But the continued existence of any organized entity is constantly threatened with various kinds of change, including growth and decay. In order to survive, a living entity continuously combats this deleterious change with change - by changing its organization in a manner that promotes the persistence of organization itself. While the identity of an organized entity itself may be lost through change, organization itself persists.

This says that in lifelike systems what gets persisted is organization itself. Consequently, persistence in Organodynamics concerns the persistence of organization.

The countervailing force against this persistence of organization is change. But also, the antidote against disorganization (the absence of the persistence of change) is also change. So change is both the problem and the solution.

In OCS, there is Organization, which can persist; or if Organization does not persist then it becomes Disorganization. But whether it does or not is a function of change.

In order to effect the antidotal change that Organization needs in order to persist, organic systems also exhibit a set of other organizing principles that interoperate to achieve this persistence. In fact, Persistence of Organization is a property that emerges through the interplay of five other organizing principles: Emergence, Nesting, Reorganization, Autocogeneration and Uncertainty.

As long as this interplay results in the continued exhibition of Organization, then life persists – even if Disorganization is also occasionally exhibited by an organic system.

At this point we have started to discuss these OCS organizing principles as though they are a higher-level abstraction of an Organodynamic web. We are actually treating these organizing principles as a meta-system *about* organic systems and *about* Organodynamic webs.

Such a meta-system may be too abstract to be of any real specific value. However, such a meta-model does provide us with a kind of “thought experiment” to think about the nature of Organodynamic webs, organic systems and the interplay of the OCS organizing principles within that milieu.

So there should be some probative value in such play, as long as it is not taken too literally. So, in this section, we shall pursue this line of thought. The essential idea that we shall contemplate is that, in lifelike systems, Persistence of Organization is an emergent property of the interplay of the other OCS organizing principles – at least for a while.

But, we also have to take into consideration that Disorganization is also a state that organic systems can reach. In fact, Persistence continues to emerge within a lifelike system as long as Organization is occasionally exhibited, even if Disorganization is also occasionally exhibited. Only when Disorganization is exhibited, while at the same time Organization is not, has Persistence ceased.

That is, only when Disorganization becomes a non-changing state (an absorbing state in Markov chain parlance) does the Persistence organizing principle cease to be exhibited. Moreover, in our meta-model, the interplay among the components consists of several interrelationships and transitions among them.

In other words, our meta-model will also take the form of a Marko chain. Throughout this section, keep in mind that we are entertaining a quite abstract meta-model that is one large level of abstraction removed from the grand scheme of Organodynamics – the Organodynamic web. But, as a “thought experiment”, this thought process should pull us into contemplating the interrelationships among the OCS organizing principles and how Persistence of organization, and or an organic system, emerges – at least until it no longer does.

Conspicuous Mechanisms of Persistence in Biological Systems

But first, in order to set the stage for a deeper conversation about the Persistence of Organization in lifelike system, we shall take a look at the conspicuous mechanisms of persistence in the biology of life on earth.

Persistence in biological systems is a complex, highly interdependent phenomenon that is modeled with Organodynamics through the interrelationships among the seven OCS organizing principles.

However, there are some specific mechanisms that can be identified and related to various organizing principles of OCS. We shall discuss two of these here. Both are various forms of *adaptation*.

Adaptation through Mass Ramification of Organizational Diversity

The first form of biological adaptation that we shall recount here can be described as *mass ramification of organizational diversity*.

This diversity includes systems with both high and low degrees of organization. Since *persistence* pertains to organization, low degrees of organization work against persistence and high degrees encourage it.

Mass ramification is conspicuous in biological systems as *adaptation* through mass replication and natural selection. This kind of adaptation works by enabling replication for organisms that survive long enough to replicate. This approach optimizes a very narrow kind of “fitness” – fitness to survive to replicate. In other words, highly organized systems have further to go to reach a state of total disorganization than do systems with low levels of organization.

Within biological systems, the mechanism of mass ramification is ubiquitous. For lower levels of organization – specifically protobiotic and cellular – this mechanism predominates, and is often exclusive there.

From an “engineering perspective”, it could be argued that this natural selection process is downright crude and generally suboptimal. Specifically, the “fitness to survive to replicate” is not necessarily the “most important kinds of fitness” according to many other criteria. Specifically, it singles out fitness for a specific event (replication) to the exclusion of others. For example, “fitness to survive to replicate” is not the same as “the strongest”, “the smartest”, etc. Natural selection would select “idiots” over “geniuses” if idiocy got one to the replication state more efficiently – in which case, “genius” would never have a chance to be selected.

Moreover, this *mass ramification* approach seems inelegant to modern engineering culture – if for no other reason than its apparent wastefulness. For example, massive amounts of energy and matter are expended just for the sake of removing all entities except for the ones which were the “most fit to survive to replication”.

Nevertheless, regardless of any “degree of engineering inelegance”, the *mass ramification* program of biological life on earth has proven vastly more successful than any engineering achievements to date. Mass ramification may not be a program to produce “the best”, but the process itself has proven persistent – whether or not its products have.

Mass ramification, by increasing the incidence of both organization and disorganization, therefore increases the absolute incidence of organization. Two OCS organizing principles are conspicuously involved in this mass ramification. They are Uncertainty and Autocogeneration.

Autocogeneration implements the ramification of organization and disorganization via its various mechanisms, such as organodynamic transforms and regularity mechanisms. And Uncertainty provides the diversity of organization and disorganization.

Adaptation through Memory

In biological systems, higher-level animals can adapt through “learning”. Learning requires a combination of memory and “feedback control”.

In Organodynamics, “feedback control” is a function of Autocogeneration and is modeled as conditional probability distributions located within stochastic processes. These are called *conditional stochastic processes* and these dependencies are modeled via conditional probability distributions. In fact, these combinations are the bases for Cybernetics and other Autocogenerative behavior.

Now, much of the “engineering inelegance” of the mass ramification approach can be obviated if Uncertainty and Autocogeneration can leverage memory. These memory phenomena can increase the value of stochastic dependency to be able to obviate the occurrence of disorganization and to increase that of organization.

The specific type of conditional stochastic processes that we have emphasized in Organodynamics is the Markov chain. Unfortunately, Markov chains are “memory-less”. That is, even though the outcome of step $t+1$ is dependent on the current time step t , it does not retain any “memory” of the past – of any time steps $t-1$ or earlier.

(Markov chains actually do have memory – but it is very short, being only for the current step. More elaborate unconditional distributions have longer memories – memories that can be of any length. On the other hand, unconditional probability distributions have absolutely no memory – not even for the current step.)

Therefore, as it currently stands, we cannot model *adaptation through memory* very effectively using Organodynamics because of our use of Markov chains. In order to effectively model adaptation through memory within Organodynamics, we need to enhance the Organodynamics framework to use more general forms of conditional stochastic processes than Markov chains. This is important and needs to be pursued. However, it is beyond the scope of this text, and we leave it for further research.

We chose to concentrate on Markov chains, rather than more general conditional stochastic processes, because of their accessibility and tractability. After all, the subject matter is complex enough for a first volume in the topic.

Other Organizing Principles Support of Persistence

OCS defines seven organization principles that interrelate to result in the persistence of a lifelike system – and of life itself, at least until some time at

which this persistence ceases. But, the point is that we don't seem to be willing to name something as lifelike unless it persists for at least some amount of time. Fleetingness of existence of a form that would otherwise be considered as "lifelike" is sufficient for the form to "fail the test".

According to the principles of OCS, while the essence of "livingness" is the "persistence of organization", that description is not enough. OCS insists that certain kinds of organization and certain kinds of change must also be involved if it is willing to grant an entity the appellation of being "lifelike". And these other "descriptions" are what OCS calls its organizing principles. A rock, for example, exhibits "persistence of organization", but OCS does not regard a rock a "lifelike".

In OCS, all of these organizing principles interrelate to result in the – at least temporary – persistence of organization.

The entire set of organizing principles forms a network that supports each other, and therefore the persistence of organization. What's most interesting is that the network itself supports persistence of the network, and therefore of persistence of organization as well.

Thus, in OCS, the persistence of organization amounts to persistence of this network of interactions among the seven organization principles.

Of course, the Organodynamics framework is, in fact, an implementation of this network of interactions among the seven organizing principles of OCS. And, the embodiment of the Organodynamic framework is its Grand Scheme, the Organodynamic web.

Therefore, what we want to do in this chapter is to support the claim that an Organodynamic web can be persistent. In so being, it can support the persistence of Organization.

In order to do this, we shall develop a high level, toy model of an Organodynamic web so that we can consider its persistence. Of course, this will be a model of a model, and thus a *meta-model*.

Therefore, in this chapter we are engaged in the development of a meta-model of Organodynamics for the purpose of considering its persistence – and therefore the persistence of Organization.

The elements of such a meta-model will essentially be the seven organizing principles of OCS. And the purpose will be to appraise their inter-relationships to see how this might result in the persistence and/or termination of Organization and its opposite, Disorganization.

Such a consideration is quite abstract and too high-level to be a proper model. However, ultimately, the notion of persistence in Organodynamics must pertain to the entire network of relationships among these organizing principles.

And, certainly, such a model is not a model of an organic system – as is Organodynamics itself. Rather it is model of a high-level abstraction of the Organodynamic framework. We must keep the degree of abstraction in focus while considering it. Nevertheless, a consideration of how these organizing principles interact even at this abstract level should be revealing as to what is at work in lifelike systems and their persistence.

Let us proceed now to construct such a meta-model. Essentially, we shall be developing a general meta-model of Organodynamic webs.

Directed Graph of OCS Organizing Principles

In order to reflect upon the interrelationships and interactions among the OCS organizing principles, let's construct a directed graph that considers how each of these organizing principles can "give rise" to the others.

Our process of discovering this will be to review our Organodynamic web model, and as we traverse the structure, notice when we transition from one of these organizing principles to others. It is expected that there will be occasions when some of these organizing principles can transition to multiple others – but only one at a time.

The result of this exercise will be a directed network model – but it would not be a model of an organic system. Rather it would be a very high level, toy, meta-model of an Organodynamic web – and therefore of Organodynamics itself.

Moreover, in order to register some sense of dependency dynamics among the organizing principles, whenever an organizing principle transitions to multiple others, we shall require that any specific transition will be to exactly one of the others for each instance of the transition. For subsequent instances of the same transition, other choices can be made. This assumption will enable the use of a transition matrix and of Markov chains in our meta-model.

We shall also register a subjective sense of relative frequency of each choice. To be consistent, let's identify our subjective references to the following: ALWAYS, ALMOST ALWAYS, OFTEN, OCCASIONALLY, SOMETIMES, SELDOM, ALMOST NEVER, and NEVER. OCCASIONALLY is considered to be more often than 50% of the time, and SOMETIMES is considered to be less than 50% of the time.

We can further associate probabilities with these subjective values, and in so doing promote our subjective model into a real Markov process. We shall also want to play "what if" with this model by changing the probabilities to various values so as to produce different categories of Markov chains (homogeneous, ergodic, absorbing, nonhomogeneous, etc.) and to contemplate the results of each.

While the modeling value of such a meta-model is probably questionable, it should nevertheless provoke some worthwhile thought regarding the relationships and dynamics among these seven organizing principles. Therefore, we shall embark on the exercise anyway.

Identifying Relationships between Organizing Principles

In order to engage in a somewhat empirical exercise of “observing” the OCS organizing principles “giving rise” to others, we shall engage in a kind of narrative. The narrative will imagine the formation of an Organodynamic web, starting from the appearance of a simplex system and continuing on with the transition through time steps, the occurrence of organodynamic transforms, the emergence of composite systems and other system dynamics within an Organodynamic web. During our “trip” though the “life” of this Organodynamic web, we shall observe and record these possible transitions, and their relative likelihoods, between OCS organizing principles.

Meta-model Strategy

Keep in mind that the purpose of all of these meta-models of Organodynamics is to consider the nature of its Persistence.

As we work through this narrative, we shall observe when new instances of the seven organizing principles arise. We shall also take note of which other organizing principle the new instance arose from. We shall take note of this by noting an ordered pair of the form [a, b], where “b” is the new organizing principle instance, and “a” is the other organizing principle that it “arose from”, or “transitioned from”.

This collection of ordered pairs will provide the raw information we need in order to construct meta-models – including the directed graph. Further below, however, we shall use this collection of ordered pairs to create two more meta-models of an Organodynamic web – and thus of Organodynamics.

The Narrative

Having explained our exercise, we are ready to begin our narrative.

Transitioning from Organization

Let’s begin in the beginning with the appearance of a system – a simplex system with a population and an organization. This, of itself, is the first organizing principle – Organization.

But organizations don’t “just sit there”. They change. And what changes them (in Organodynamics) is that they “undergo a reorganization” (the *reorganize* organizing principle). That reorganization may be a change in organization, or it may be a change in population or both. Often, this reorganization is a simple change of system state. But sometimes the change is more complex – it is the result of systems combining or dividing. . That is, the change is the result of splits or joins organodynamic transforms. The Autocogeneration organizing principle defines these transforms.

Now, we have mentioned a number of organizing principles so far, but have yet to record any ordered pairs of them, as we said we would. The reason we

haven't is because none of them have occurred in our narrative yet. We are simply setting up for what is about to happen.

So, at this point in our narrative, we have a simplex system, which constitutes the first organizing principle – *Organization*. What happens next is that “time takes its toll” on our system, and it gets *reorganized*. This is our fourth OCS organizing principle, *Reorganization*. So, NOW we have the occurrence of one organizing principle *transitioning*, or giving rise to, another. In fact, we have the ordered pair [Organization, Reorganization]. This pair OFTEN occurs whenever Organization occurs.

However, there is one other possibility. If an organization has an emergent property, then it transitions to Emergence. This gives us the ordered pair [Organization, Emergence]. This transition occurs relatively SELDOM.

Transitioning from Reorganization

But, the principle of *Reorganization* in Organodynamics always transitions (gives rise to) the fifth organizing principle, Autocogeneration. The reason for this is that Autocogeneration gives specificity to Reorganization *by* defining *how* reorganization is carried out.

However, there are a number of ways that Reorganization can be carried out in Organodynamics. And all of these ways are specified under the Organizing principle named Autocogeneration. We shall eventually transition to Autocogeneration, where we shall discuss these ways. However, the manner in which we transition to Autocogeneration from Reorganization is uncertain. Therefore, must transition through the organizing principle of Uncertainty on our way from Reorganization to Autocogeneration.

This gives us the ordered pair [Reorganization, Uncertainty]. This pair always occurs whenever Reorganization occurs. And we can say that it means that Uncertainty is the way that Reorganization uses to select its choice of Autocogeneration mechanisms.

Transitioning from Uncertainty

But, in Organodynamics, uncertainty always pertains to the way in which reorganization is carried out. This means that Uncertainty always “selects” a mechanism of the Autocogeneration organizing principle and then transitions to Autocogeneration in order to have it carried out. . This gives us the ordered pair [Uncertainty, Autocogeneration]. This pair always occurs whenever Uncertainty occurs.

Transitioning from Autocogeneration

There are a number of ways that Autocogeneration is implemented, including the state transitions within a segment or even within an edge of an Organodynamic web. A second way that Autocogeneration is implemented is via the organodynamic transforms.

Regardless of these two general mechanisms there are three types of outcomes that can occur from an Autocogenerative mechanism. What is important to us in our quest to construct a directed graph of the OCS organizing principles is that these three outcomes pertain to three other organizing principles: Organization, Nesting and Disorganization.

Disorganization? But Disorganization is not one of the seven OCS organizing principles! Why is it being included?

This observation is correct. Disorganization is not one of the seven organizing principles. However, it is a “state” of the Organization organizing principle. In fact, the principle of Organization is that the “degree of organization” of a system can be anywhere from “highly organized” to “highly disorganized” and anywhere in between.

It is true that we have not yet defined a measure of organization/disorganization in Organodynamics. However, we do have the concept – even though we have not defined how to measure it (yet). (We have left the job of defining such a measuring function to further research.)

Nevertheless, the concept of Disorganization as a degree of the organizing principle Organization is very important to Organodynamics. In fact, without it, the question of Persistence cannot arise. This is the case, because Persistence means “the persistence of Organization”. Without Disorganization, there is no issue of degree of Organization.

Therefore, in order for the directed graph that we are constructing to be able to address the concept of persistence, it must be able to distinguish between the two states of the Organization organizing principle: the *organized state* and the *disorganized state*.

Admittedly, we have not yet defined when a system organization (a set of related duples) is “relatively organized” or “relatively disorganized”. But we know that anytime that Reorganization occurs (and thus an Autocogenerative event) that the resulting organization will have some relative degree of *disorganization* or *organization*. And, each of these *must have some relative probability*.

In fact, the two states of *relatively organized* and *relatively disorganized* are two of the three possible outcomes of the Autocogeneration organizing principle in our directed graph. At this time, we must identify the other two – and then assign some relative likelihood values to each of the three.

In fact, we have already mentioned the three possible transitions from Autocogeneration, but got momentarily sidetracked explaining one of them: Disorganization. Again, the three OCS organizing principles that can be transitioned to from Autocogeneration are: Organization, Nesting and Disorganization.

We shall presently explain the circumstances under which each of these are transitioned to from Autocogeneration, and also assign a subjective relative frequency value to each.

First there is Nesting, giving us the ordered pair [Autocogeneration, Nesting]. Nesting occurs whenever the Autocogeneration is executed via the Integrate transform. This transform always creates a nested system from a collection of others. This transition occurs (relatively) ALMOST NEVER.

Next is Organization, giving us the ordered pair [Autocogeneration, Organization]. This occurs for all segment or edge reorganizations by virtue of the fact that a distinct system organization is selected for the next step in the organic process. However, a distinct system organization is also always selected for any of the organodynamic transforms that Uncertainty could have selected for this case. This transition occurs relatively SELDOM.

Of course, in all of these cases, the degree of organization of the selected system organization could be considered to be organized or disorganized. In this case, we mean the system organizations that are considered to be *organized* rather than *disorganized*.

Finally there is Disorganization, giving us the ordered pair [Autocogeneration, Disorganization]. This is the same case as the above, except that the system organizations are considered to be *disorganized*. This transition occurs OCASSIONALLY.

Notice that the previous three transitions are all from Autocogeneration and therefore must sum to 100% if the time.

Again, for the above two cases, it is necessary to have defined a measure function for the “degree of disorganization/organization” of a system organization – and we have admittedly not done that yet. Also, we would need to identify a specific degree of disorganization – according to that measuring function below which would be considered to be “organized”. Although we have not done these, we are assuming that they are possible.

Transitioning from Emergence

Emergence enables new component relationships, and therefore new system organizations. Consequently, Emergence always transitions to Organization. This provides the pair [Emergence, Organization]. This transition occurs ALWAYS.

Transitioning from Nesting

Nesting is the act of creating a composite system, which is a new system, which includes a new organization. Consequently, Nesting always transitions to Organization. This provides the pair [Nesting, Organization]. This transition occurs ALWAYS.

Transitioning from Disorganization

Disorganization is a state of Organization, and, like Organization, always transitions to Reorganization. Consequently, Disorganization always transitions to Reorganization. This provides the pair [Disorganization, Reorganization]. ALWAYS.

OCS Organizing Principles Relationship Pairs

Below, we shall use the ordered pairs that we identified above to construct a number of meta-models of an Organodynamic web. Recall that all of these meta-models are being constructed in order to consider the persistence of organization and of the life of an Organodynamic web.

In the service of constructing these models, we shall collect these ordered pairs in one place, which is here. In addition to the pairs of related OCS organizing principles, the subjective relative frequencies that these pairs are taken in the directed graph model presented below is also presented – as observed from the above narrative. These relative frequencies will also be helpful for the construction of some of the meta-models below.

[Organization, Reorganization] OFTEN

[Organization, Emergence] SELDOM

[Reorganization, Uncertainty] ALWAYS

[Uncertainty, Autocogeneration] ALWAYS

[Autocogeneration, Nesting] ALMOST NEVER

[Autocogeneration, Organization] SELDOM

[Autocogeneration, Disorganization] OCASSIONALLY

[Emergence, Organization] ALWAYS

[Nesting, Organization] ALWAYS

[Disorganization, Reorganization] ALWAYS

The Relationship Graph of OCS Organizing Principles

Constructing a directed graph from the relationships among the OCS organizing principles that we summarized above, we get the following meta-model of an Organodynamic web.

Summary of Organizing Principle Relationships

A number of aspects of the above graph deserve comment. The reader will notice that Persistence does not appear within any of these models as an organizing principle of organic systems. Rather, Persistence has been treated as an emergent property of the meta-model – or of Organodynamics itself.

Persistence occurs as long as the model is *visiting* the Organize node of the graph.

Also, Persistence ceases to occur if the model *visits* the Disorganize node of the graph and never visits the Organize node.

In order for an Organodynamic web to behave “lifelike”, it would need to continue to revisit Organize for a long time and then eventually revisit Disorganize in such a way that it never leaves it.

Whether or not this behavior occurs depends upon the relative frequencies of the visitations of these nodes – and also perhaps on whether these relative frequencies change over time.

In order to study these more thoroughly, we shall use this meta-model to construct a couple of other meta-models – meta-models that will permit the study of these eventualities more closely.

The Organizing Principles As a System

We have represented the relationships between the OCS organizing principles as ordered pairs. This means that they are of a format with which we can represent a system.

Simplex System of Organizing Principles

Clearly, the set of organizing principles, together with the set of pairs of those principles that we “harvested” from the Organodynamic web forms a simplex system with a population of components (the organizing principles) as well as an organization of those components (the ordered pairs).

Organic System of Organizing Principles?

Just as clearly, they do not form an organic system – if only because they are not nested.

The Organizing Principles As a Markov Model

What we have constructed so far is an extremely high-level abstraction of Organodynamic systems in general, but may be fun to contemplate anyway. It should be considered as a toy model, and “taken with a grain of salt”.

It is well known that a finite directed graph could be represented as a Markov transition matrix – if transition probabilities are known. We do not have

transition probabilities for this graph, but we do have subjectively described relative frequencies, from which we can estimate probabilities. Thus, we can construct a Markov transition matrix.

Then, we can use this transition matrix as a starting point to perform various speculations and what-ifs regarding the long-run behavior of this Organodynamic meta-process in order to speculate about its persistence as a family of stochastic processes.

The Initial Markov Transition Matrix Values

In this section we shall establish an initial Markov transition matrix for use in modeling the homogeneous and nonhomogeneous cases of our Markov meta-model of an abstract Organodynamic web.

The intention is for the reader to use and alter these probabilities to determined which probabilities result in long running Markov processes. Below is the initial transition matrix. It is consistent with the directed graph depicted above. The column headers have been abbreviated to the first letter in order to conserve space.

	O	E	N	R	A	U	D
Organized	.00	.35	.00	.65	.00	.00	.00
Emergent	1.00	.00	.00	.00	.00	.00	.00
Nested	1.00	.00	.00	.00	.00	.00	.00
Reorganizational	.00	.00	.00	.00	.00	1.00	.00
Autocogenerative	.30	.00	.15	.00	.00	.00	.55
Uncertain	.00	.00	.00	.00	1.00	.00	.00
Disorganized	.00	.00	.00	1.00	.00	.00	.00

The Homogeneous Case

For the homogeneous, we shall fix the probabilities of the transition matrix, and test the long run behavior of the chain. The idea is to try different transition probabilities to see which ones result in various lengths of persisting visitations of the Organization state.

As currently configured, the transition matrix will result in a chain that never resides in a single state – namely because the matrix represents states that are periodic and irreducible. It is expected that the reader will alter these probabilities to achieve other effects. These are discussed below.

When experimenting with changing these transition probabilities for the homogeneous case, there are two subcases in particular that one would want to experiment with, since these model two possible conditions that could reasonably result in the long run behavior of lifelike systems. These are: 1) an irreducible chain and 2) an absorbing chain. The first of these models the case where all seven states are visited forever, and neither Organization nor Disorganization prevails. (As discussed, the matrix above is a periodic version of this case.) The second plays with the case that either Organization or Disorganization could be an “attractor”, and at some point in time become the sole state in existence forever.

Irreducible Markov Chain

The use of an irreducible Markov chain models the case where the seven organizing principles would all be visited forever, with none of them ever gaining total control, and not allowing any of the others to be visited. Therefore, “total disorganization” would never be realized.

Rather, the Disorganization organizing principle, though always revisited, would always transition to some other organizing principle. The same goes for total Organization – and for all of the other organizing principles. All of them would be continuously revisited. Of course, this represents immortality, which is not what Organodynamics expects.

For those who believe that the second law of Thermodynamics assures “absolute disorganization” after some point in time then this model would not be for you either. For example, you may be interpreting the Clausius version of thermodynamic entropy. Or you may interpret statistical mechanics as talking about *disorganization* when it uses the term *disorder* (rather than *uncertainty*).

So, if your position is that thermodynamics specifies complete disorganization then you would need one of the *absorbing* Markov chain models presented in the next subsection. Or perhaps one of the nonhomogeneous Markov models discussed after that would do for you. In any event, the irreducible chain would not work for you in that case.

But, lets get back to our irreducible chain. Of course, the *percentages* of times visited could be weighted toward any of the seven organizing principles. This is achieved by judicious settings of the transition probabilities.

The transition matrix presented above is an irreducible matrix – meaning every state can be eventually visited from every other state. One can see this from the matrix itself because there is exactly one ergodic set and no transient sets. This arrangement conforms to the directed graph depicted above.

Of course, this particular transition matrix has been offered in the spirit that it is an initial approximation. The expectation is that the reader will experiment with trying other probabilities for this matrix.

Nevertheless, as listed above, this matrix is irreducible - also called an *ergodic* transition matrix because the set of all of its states is ergodic.

This matrix, as presented, is also aperiodic. This means that not all states are *directly* accessible from all other states. This means that you cannot get from every state to every state in one “hop”. Sometimes it takes several hops. You can tell that this is the case because some of the probability values are zero. If you could get to every state directly from every state, then none of the probabilities would all be positive.

Such a transition matrix is called *periodic*. If I had provided an initial transition matrix that had no zero probabilities, then our homogeneous Markov chain would have been periodic – and it would be ensured to have a *stationary distribution* - as we have previously discussed [Kemeny and Snell 1976].

If this transition matrix were aperiodic (no zeros anywhere), then its long run behavior would necessarily have a limiting distribution, called its *stationary distribution*. This means that if this matrix is multiplied by itself N times, and N is allowed to go to infinity then the limiting matrix of such a sequence has all of its rows the same. That is, all of its rows equal each other, and they all represent the same probability distribution - its stationary distribution. The stationary distribution represents the long run, limiting behavior of a homogeneous aperiodic irreducible Markov chain.

The reader will recall that it turns out that there is an easier way to find the stationary distribution of a finite step homogeneous aperiodic irreducible Markov chain than by calculating this limit. The technique [Kemeny and Snell 1976, pp. 71-72] involves finding a vector which, when multiplied (on the right) by the transition matrix (on the left), yields itself as a vector. Such a vector (being a probability vector) describes the stationary distribution of the chain.

Calculating the stationary distribution of such a transition matrix then become a matter of solving the simultaneous linear equations that result from such a multiplication.

The reader may want to consider whether to change the above transition matrix so that it is aperiodic – no zero probabilities. This would mean that every state (organizing principle) would be able to transition directly to every other – even if the probability were very small. Whether or not this would make sense to the model is something the reader must decide.

But if the reader decides to experiment with an aperiodic irreducible transition matrix for this meta-model, then a stationary distribution will exist for it, and can be calculated.

Absorbing Markov Chain

If one would like to construct an Organodynamic web meta-model in which “the universe can devolve to total disorganization”, then changing the initial Markov matrix provided above so that it is an absorbing homogeneous Markov chain could very well be what you want.

In fact, changing it so that “the universe can evolve to total Organization” can also be modeled with an absorbing homogeneous Markov chain. It all depends

on how one establishes the probabilities. (More on that in a few sentences below.)

For that matter, one can use the same approach so that “the universe can evolve to exactly one of Disorganization forever or Organization forever – and you don’t know which” when you start the model.

In fact, in the spirit of honesty, one can use the same approach to model the case that ANY (but not all) of the seven organizing principles can dominate – and you don’t know which when you start the model. All of this can be controlled by the way you set up your homogeneous transition matrix.

I’m going to assume that you want to construct the transition matrix so that only Disorganization or Organization is a candidate for “total domination” forever. Here’s how to do that.

Select among Disorganization and Organization the one(s) you want to have the opportunity to EVENTUALLY “dominate forever”. If you only select one of these, then it will definitely dominate forever if you use the following procedure. If you select both, then exactly one will dominate – if the model is run long enough. But it could be either one on any run of the model. However, it will never be any of the others.

To construct a transition matrix for this, you want to ensure that whichever ones you have selected is represented by an *absorbing state* in the transition matrix, and that all of the remaining states are members of one or more *transient sets*.

Once your transition matrix is established to have these characteristics, then the selected states (Organization or Disorganization) is bound to eventually take over and never lose control to any other state. One can then change the probabilities within the matrix to see how long it takes for “domination” to occur, which dominates and which paths are traversed prior to “domination”.

Nonhomogeneous Case

Homogeneous stochastic processes, in which the sample space points and their probabilities never change step after step are the most tractable of stochastic processes. Also they provide a very rich theory, simply because there is a lot of simplified relationships that can be stated about them.

Nonhomogeneous Markov Chains in Organodynamics

But, in “real life”, stochastic processes (Markov chains in particular) are not often homogeneous – at least, they are not homogeneous for long. Actually, homogeneous Markov chains occur very often – in “real lifelike” systems; they just can’t be depended on for lasting very long. While their transition matrices may stay the same for a few steps, they inevitably change their probabilities sooner or later. Thus, in living and lifelike system, stochastic processes are generally nonhomogeneous.

Take the weather, for example. It not biological, but is often assumed to exhibit “lifelike” behavior. (Whether it actually satisfies all seven OCS organizing principles deserves consideration – but not right now.) In late summer, where I live, intermittent rains punctuate the weather – especially in the late afternoon. If one were to identify the states of rain, clear, cloudy, etc. and recorded their probabilities in a matrix, these probabilities would hold relatively the same for late summer into autumn. In other words, during this period, the weather is well modeled by a homogeneous Markov chain.

But in the middle of October the weather changes – and so would the probabilities. But these new probabilities would hold pretty well until late December, when they would change again.

What we are describing here is what we have called a *piecewise homogeneous Markov chain*. That is, the same transition matrix holds for some number of contiguous steps, and then changes and holds for some other number of contiguous steps.

Overall, such a chain is nonhomogeneous, because the same transition matrix *does not* hold throughout. In fact, if the transition matrix were to change at every step, then the process would be a completely nonhomogeneous Markov chain. In fact, such would be a special case of a piecewise homogenous Markov chain, all of whose homogenous segments were of length one.

Thus, in general, we can say that Organodynamic systems are piecewise homogeneous, and also that they are nonhomogeneous. Thus, the Markov chains of Organodynamics are ultimately nonhomogeneous. The fact that they are nonhomogeneous makes them realistic. The fact that they are piecewise homogeneous makes them partitionable into segments that are tractable.

The Organizing Principles Meta-model as a Nonhomogeneous Markov Chain

Above, we considered what would happen if the Markov chain used for our “organizing principles” meta-model were homogeneous. It is true that an Organodynamic web itself it *not* homogeneous. (In fact, it is a special kind of nonhomogeneity called piecewise homogeneous.)

Nevertheless, above we played around with setting up our meta-model as being homogeneous. Such an assumption is pretty limiting (nonhomogeneous chains are far more flexible), but we were able to get some desirable behaviors anyway. One of these behaviors was a chain that necessarily becomes Disorganized forever. Another was a chain that could either become Organized forever or Disorganized forever, depending on initial conditions. (These were both categorized as Absorbing Markov chains [Kemeny and Snell 1969].)

It may have been surprising that we could achieve these behaviors with homogeneous Markov chains.

The point of this subsection is that *we can achieve almost any result that we want if our meta-model uses nonhomogeneous Markov chains.*

From a modeling perspective, this means that, from time to time, the probabilities in the Markov transition matrix will change. This can be motivated by a change in some initial or external conditions in the environment; and can be modeled mathematically via matrix multiplication. For example, a matrix representing a certain external condition can be multiplied by the existing transition matrix to produce a changed transition matrix. This new transition matrix can then be in effect for several contiguous time steps until the conditions again change.

Thus, it is likely that the best fit for our Markov meta-model is a nonhomogeneous one – particularly a piecewise homogeneous model. One can achieve pretty much any long run results desired by a suitable sequence of transition matrices.

The challenge in this case is to find such sequences of transition matrices that generate various results. The second challenge is to then justify that there are reasonable circumstances (conditions) under which these sequences of transition matrices could obtain.

Long Run Disorganization, Uncertainty and Entropy In Organodynamics

Readers who equate large entropy with large degrees of disorganization will be in for a surprise with the results of Organodynamics.

In the first place, as has been pointed out already, entropy in Information Theory is not a measure of disorganization. Rather it is a measure of uncertainty.

And in Organodynamics – and probably in general – uncertainty and organization are orthogonal, because they can vary independently. One can have high degrees of uncertainty and, at the same time, have either high or low degrees of organization. The same goes for high degrees of certainty.

(This orthogonality is essential to Organodynamics. There can be a lot of uncertainty of what the system organizations of the next step will be – even if all the possibilities are highly organized, or whether they are all highly disorganized.)

Therefore, knowing that a stochastic process exhibits a high degree of certainty tells one absolutely nothing about whether the states of the process are highly organized (or highly disorganized).

Let's apply these facts to our organizing principles meta-model. The same considerations will hold whether we have a homogeneous or nonhomogeneous meta-model. Therefore, for the sake of simplicity, let's assume a homogeneous meta-model.

Let's first take the case that absolutely goes to Disorganization in the long run. (This is the case that models the popular conception of the second law of thermodynamics.) This case is an absorbing Markov chain with one absorbing subset – the subset that contains only the Disorganization principle.

For such a case, the long run probability distribution has probability of 1 for the Disorganization state, and probabilities of zero for the other cases. But, if we calculate the Shannon entropy for such a distribution, we find that its value is zero! That's right – *minimum* entropy.

But this is the opposite result from that expected by those who interpret entropy as a measure of *disorganization* – where the expectation is that the second law demands that the entropy is at its *maximum*.

The quandary is solved, however, by realizing that entropy is not a measure of disorganization – but rather a measure of uncertainty. Of course, most interpretations of the second law insist that entropy is a measure of *disorder*. But this implies that the word *disorder* must actually mean *uncertainty* – not *disorganization*.

Another point to make is that *any* absorbing Markov chain with exactly one absorbing set will have an entropy of zero – regardless of whether the absorbing state is Disorganization, Organization or any of the other states!

Another interesting case is the one in which all of the states are equiprobable. Such a distribution has maximum entropy. This is always the case for any finite probability distribution [Shannon 1953]. We can conclude from this that the distribution with the maximum entropy is *not* the case where Disorganization is the inevitable case – or where any state is the inevitable case. This fact is also counter-intuitive to those who interpret the second law to ensure that maximum disorganization must obtain when entropy is maximized.

As stated, all of these surprises are explained by recognizing that Shannon entropy is not a measure of disorganization. It may be a measure of *disorder* – but only if “disorder” means *uncertainty*, and not if it means *disorganization*.

Conclusions

This chapter has entertained a meta-model of Organodynamics for the purpose of considering the interplay among the seven organizing principle of OCS within the Organodynamics framework – in particular within its *grand scheme*, the Organodynamic web.

The meta-model was constructed using the seven OCS organizing principles as its “components”, with the following alterations: 1) Disorganization was added as a component, and 2) Persistence was removed as a component of the meta-model and treated as an emerging property of the meta-model.

The meta-model was expanded into a dynamical model by associating transitions among its components (the OCS organizing principles). This was accomplished by developing and analyzing a narrative of the construction of an idea Organodynamic web. This dynamic meta-model was further embellished by associating subjective likelihood terms to each transition, and then quantized by associating probabilities to these likelihood terms.

At this point, the resulting meta-model was amenable to articulation as a Markov chain. The aforementioned probabilities are easily assembled into a Markov transition matrix.

The intention is for the reader to play “what-if” with this meta-model by altering the probabilities in the transition matrix to various scenarios, and then seeing how the resulting Markov chain plays out in long run scenarios.

Several tests were suggested, including the altering of the transition matrix to achieve homogeneous, irreducible, periodic, aperiodic or nonhomogeneous chains in order to test various scenarios. The idea is to see which of these scenarios resulted in which long term behaviors, including persistent Disorganization, persistent Organization, either persistent Disorganization or persistent Organization and nonpersistent behaviors.

In all cases, it is admonished to understand that these meta-modeling exercises are to be considered as extremely high-level models of a modeling framework – rather than as models of organic systems. Specifically, the meta-model we developed here is not an Organodynamic web. It is the Organodynamic web that is intended to directly model organic systems – not the meta-model discussed in this section.

Nevertheless, contemplation of this toy meta-model should be able to shed some light on the interplay of the seven OCS organizing principles as they operate within the Organodynamic framework and their relationship to persistence in Organodynamics – if interpreted in the proper light.

Incremental Example Model Development

<>

Progress Check

In Part II, Organodynamics, we are gradually and incrementally developing a mathematical framework for modeling living and lifelike systems according to the seven organizing principles of OCS. Once completed, this modeling framework will achieve the form of an abstract but highly flexible mathematical structure that we are calling an *Organodynamic web*.

We are using the example of a hobbyist’s toolkit for building electronics as an analogy to Organodynamics. Both are systems for “building something”, and in both cases the resulting “thing to be built” has a network structure and has been given a general name.

In the case of the electronics circuit toolkit, the general name of the “thing to be built” is “circuit”. The circuits built by this kit are highly customizable to specific applications. In the case of Organodynamics, the general name of the “thing to be built” is “Organodynamic web”. The models built by this mathematical framework are also highly customizable to specific applications.

Because of the complexity of organic systems, any model that represents them needs to be built in cumulative and increasingly accurate stages. Each stage is generally referred to as an “approximation”. Organodynamics as a model builder’s toolkit provides for six of these approximations. These approximation levels are well defined and occur at precise places in the methodology set forth by Organodynamics.

This chapter has addressed how the Organodynamics framework addresses the seventh OCS organizing principle, persistence. The answer is that persistence is an emergent property of interrelationships exhibited by an Organodynamic web as a dynamical system that exhibits the other six OCS organizing principles.

Review of Organodynamics

We've finally arrived at the end of this recitation of our mathematical framework, Organodynamics, which embodies the seven OCS organizing principles. Herein, we've attempted to provide a system for constructing comprehensive mathematical models of living and lifelike systems, or - as we have labeled them - *organic systems*.

This review will summarize all of the modeling structures provided by the Organodynamics framework, their interrelationships, and how to use them to construct dynamical models of specific organic systems. It will do so by constructing a narrative that describes how these constructs fit together to incrementally build up a model of an organic system.

The challenge of Organodynamics has been that organic systems are massively complex. So much so that attempts to construct comprehensive models of them have been rare. Mathematical models of organic systems are almost always of isolated subcomponents – virtually never of entire organisms.

But, Organodynamics takes the position that constructing comprehensive models of whole organisms is important. For myriad reasons, models of whole organisms are useful – even essential - for some applications. Another motivator is that summing models of components does not often produce verifiable models of whole systems due to the nonlinearities involved.

What is needed is a model construction system that is designed to produce comprehensive models from the start. Such is the intention of Organodynamics.

But, if living systems are bewilderingly complex, then models of them must preserve a good deal of that complexity in order to be verifiable. On the other hand, any model must be intellectually accessible to scientific observers for it to be usable.

Therefore the complexity of the models produced by the Organodynamics framework must “walk a line” between being too complex, yet not complex enough. These models should not be quite as complex as the organic systems that they model, because we do not want to bewilder the human modelers. On the other hand, these models must be capable of embodying tremendous complexity in order to be valid.

These required levels of complexity also have something to say about the mathematical framework that is capable of producing them: Organodynamics must – itself – embody generous complexity. Its purpose is to preserve this complexity - but at the same time to bring order to it.

And, in fact, the Organodynamic framework is admittedly a complex apparatus. This makes writing this Review chapter somewhat challenging. I must ascertain an approach that covers the major elements of the theory in a lucid and perhaps intuitive way.

Actually, I do see an obvious approach. I shall appeal to Greek philosophy – and to the science methodology work of system methodologist Arne Collen [Collen 2003] - for a categorical approach to be used by this review chapter.

We shall follow Collen’s lead and utilize these three organizing categories, colloquially termed: *being*, *doing* and *knowing*. These three are conspicuously present in OCS to the extent that the seven *organizing principles* can be grouped according to these three categories. Even so, these three principles are highly interconnected and interdependent within the Organodynamics framework.

The next three sections, then, will review the elements of the Organodynamics framework in accordance with these categories:

The Beings of Organodynamics
The Doings of Organodynamics
The Knowing of Organodynamics

In addition, we shall also have some comments regarding *system complexity*, and offer some concluding remarks.

The Beings of Organodynamics

Organodynamics is a theory about a particular class of things, of *entities*. Specifically, this class consists of entities that exhibit the seven organizing principles of OCS, the *organic systems*. But, as a theory, Organodynamics needs to define what these entities look like – their constitution or form.

Organodynamics goes about specifying the form that it will give to these entities by giving specificity to the *first three* OCS organizing principles. Recall that these are:

1. Organized
2. Emergent
3. Nested

The manner that Organodynamics chooses to put specificity to these three principles defines Organodynamics as a framework. Other mathematical frameworks are possible that would also satisfy the seven OCS principles. So, it is the manner in which a framework specifies these principles that defines it.

This aspect of Organodynamics is the *ontological* one. In ancient Greek philosophy, *ontology* is the study of *being*. More precisely, we want to talk about its *beings* (plural) - to look specifically at the various forms that are provided by Organodynamics to represent the beings of organic systems.

Another consideration is that Organodynamics needs to be a *dynamical systems theory* – that is, a theory that models *change*. This raises the question “Change in what?” In order for there to be change, there must be “something that is changing”. Thus, in order to be able to subsequently discuss *change* we must initially discuss a *static entity* that can subsequently be changed and

define its *form*. Later, when we want to discuss change, then we can discuss it in terms of change-of-form.

Another term for the form of an entity is its *state*. Thus, another way to describe the *being* of an entity is that its *state* is being defined.

One way to characterize *system state* is that it represents a system at a single point in time – when the system is unchanging. In other words, system state is a static phenomenon versus a dynamical phenomenon.

The First OCS Organizing Principle: Organized

Organodynamics has decided that it wants to use the idea of *system* – in the sense of General Systems Theory [Von Bertalanffy1968], also known as *systemics*, to represent *organic systems*. While systemics is quite clear by its intention regarding the idea of *system*, more mathematical specificity is required for this idea in Organodynamics.

Organodynamics desires a mathematical definition of the term *system* that has the broadest possible application, yet still embodies the principles of *system* from systemics. This desire for broad application leads to the choice of *set theory* for the articulation of the meaning of the term *system* in Organodynamics.

An essential idea regarding *systems* in systemics is the notion that a system is more than merely a set of components. What is most important is that a system embodies *interrelationships* among those components, as well as embodying the components themselves.

There are many approaches utilizing a set theoretic articulation that could model these ideas. Many of these were discussed in this text. However, it was decided that the framework would use a simple form:

$$S = (P; O)$$

Where

S is a system being defined.

P is the set of components of the system, called its *population*.

O is a set of ordered pairs ("duples") of the elements of P. This set is called the *organization* of the system. P is more completely named the *system organization* of the system S.

This form defines the Organodynamics specification for the first OCS organizing principle: *Organized*.

The Second OCS Organizing Principle: Emergent

Organodynamics defines two kinds of emergence: *systemic emergence* and *process emergence*. Both of these are *systemic properties*. This means that they are *true statements that can be made about a system*.

Emergence is a *property* of a being, rather than a being itself. For example, systemic emergence is a property of a *system*. Thus, there is a type of entity defined in Organodynamics (the *system*) that can have the *property* that it is *emergent systemically*.

(If a system exhibits *systemic emergence*, we sometimes say that it is *emergent systemically*.)

Systemic emergence is emphasized by Organodynamics. It is defined as a *property of a system that is not also a property of a component of that system*.

It is worth noting that *systemic emergence* is a *static property* because it can be detected entirely within a single moment in time. For example, in order to determine whether a *system* has the systemic emergent property, one merely inspects all properties of the system (as a whole) and makes sure that none of its components also has that property. Obviously, inspecting the state of a system within a single moment in time can make this determination. No comparison of the states of a system at two different moments in time is necessary in order to detect a systemically emergent property

We showed in a previous chapter that there is only one system (of all possible finite systems) that is *not* emergent systemically. It is the system $(\{ \Phi \}; \Phi)$. But, this is a trivial, if not degenerate, system. We can happily exclude such a system from those of our interests – organic systems.

However, any system that does not have itself as a component *is* an emergent system, because there is some statement that can be made about it that cannot be made about any of its components systems. We therefore say that all organic systems are emergent systemically.

On the other hand, *process emergence* is a dynamical property because more than one moment in time is required to ascertain its exhibition. To show this, we must understand that process emergence is defined as *the exhibition of a systemic property by a system at one moment in time when it had not been so exhibited by the system at a prior moment in time*.

Obviously, at least two time steps must be inspected – including the present one – in order to ascertain whether a systemic property at the current moment is, in fact, process emergent.

In truth, most investigators of complexity science use the term *emergence* to mean *process emergence*. However, OCS is more interested in *systemic emergence* because it is here that there is the appearance of “something (the emergent property) coming from nothing”. This is appearance only, because the emergent property comes from the *interrelationships* that are captured by the *organization* of the system.

Thus, the *systemic emergence* comes from the *organization* rather than from the *population*. But, since most people forget that the *organization* is a part of the system also (or never knew), then it appears to them that the *emergent*

systemic property came from nowhere. This is what gives systemic emergence its apparent “magic”.

In practice, most instances of emergence are both *systemic emergence* and *process emergence* simultaneously. However, there is still the need to differentiate them for a number of applications.

The Third OCS Organizing Principle: Nested

The idea of the *system* that we discussed above is the foundational static entity of Organodynamics. It is its fundamental *being*.

However, this notion of *system* is too simple to represent accurately all organic entities. This fact is revealed through the third OCS organizing principle: nestedness. Organic systems, according to OCS, are nested system – all of them. This means that their components are also systems in their own right.

What is needed is to create a generalization of *system* that does embody nestedness – the *composite system*. In order to unambiguously differentiate between the two, we shall rename the term *system* as defined to be *simplex system*.

Thus, a *simplex system* is a special case of a *composite system* that has exactly one level of organization. A *composite system*, though, can have any integral number of levels of organization. We shall now use the term *system* inclusively for either.

Composite systems have multiple levels of organization, wherein the components of a system can be other systems. We need a way to express this using set-theoretic notation as we did with the definition of the form of a simplex system.

In actuality, we would like to express that the components of a composite system are other *systems* – which would include both its *population* and its *organization*.

The challenge is to define the form of a *composite system* that embodies these ideas. To do this, we defined a nomenclature for composite systems. However, we decided that capturing both the *organization* and the *population* of the component system inside of the composite system becomes unmanageable for this nomenclature – especially for deeply nested systems.

Not only is including both the population and the organization of a component system within the nomenclature unmanageable, but it is also unnecessary. Rather, we can abbreviate and use only the *organization* of the components (nested) system to represent it, and intend that its *population* is implied. This will result in a readable nomenclature. If the corresponding population is needed, it can be accessed by reference.

Therefore, it was decided that, when describing a component system as a subsumed system within a composite system, the nomenclature would express

only the *organization* of the subsumed component system. And, even though the population is also part of the component system, this fact would be implied rather than expressed by the nomenclature.

Thus, the nomenclature of composite systems specifies that *the population of a composite system* is the set of *system organizations* of the component systems that are subsumed as the components of that composite system. Also, the *organization* of the composite system can be any set of duples whose members are those organizations. The precise declaration of the organization of a composite system will be defined on a case –by-case basis by the actions of the model.

Thus, when defining a composite system, its components are specified by this nomenclature definition, but its components are left unspecified. However, its components are constrained. They must be duples of the components of the population of that composite system. Moreover, the organization of the composite system *is specified* – but not by the definition of composite system. Rather, they are specified by a set of system transforms that will be discussed in the next section on the *Doings* of Organodynamics.

Summary of the Beings of Organodynamics

We have summarized two *beings* – forms, system states – in the Organodynamics framework.

The first is the *simplex system*, which is comprised of a *population* and an *organization*. The population is a set of components. The organization is a set of duples (ordered pairs) of those components that describe the interrelationships between the components.

The second form, or being, is the *composite system*. This is a general case of the *simplex system* that models system nestedness. A composite system has *components* that are also systems, and that are referred to as *component systems* of the composite system – or as its *subsumed systems*. These structures can be nested to any integral numbers of levels – referred to as *levels of organization*.

A nomenclature has been developed to express this form. In this nomenclature, the *organizations* of components systems are the components of the population of composite systems. The *organizations* of composite systems are undefined, except for the stipulation that they be comprised of duples whose members are components of the composite system.

The Doings of Organodynamics

OCS, and Organodynamics, is a dynamical systems theory. This means that it is a theory that concerns actions, doings, and behaviors of the target system across multiple points in time.

Like most dynamical system theories, Organodynamics models the *change* that is emphasized by the dynamical aspects of the theory the change of state of its

static entities. Therefore, Organodynamics models system dynamics by expressing changes that occur over time in its simplex systems and in its composite systems.

Thus, the fundamental entity of the dynamical aspect of Organodynamics is the notion of *system process*. System process is defined as a set of *time steps*, each of which has a *state*. This state is specified by either a *simplex system* or by a *composite system*, each of which has both a *population* and an *organization*.

However, we shall use the *organization* aspect of a system to represent its state. Its *population* aspect will be implied.

These are the fundamentals of the system dynamics of Organodynamics. Below we shall describe how these ideas have been ramified into a rich set of constructs as elements and properties are added to them in order to be able to support certain of the seven OCS organizing principles, and to result in a rich and expressive dynamical modeling framework.

Organodynamics goes about specifying the manner in which change can occur to simplex and composite system state via its mathematical implementation of the OCS organizing principles number four through seven. Recall that these are:

4. Reorganizational
5. Autocogenerative
6. Uncertainty
7. Persistent

We shall now developed further the ideas regarding the dynamical aspects of Organodynamic systems that we introduced above by visiting each of these three OCS organizing principals and reviewing the mechanisms provided by the Organodynamics framework to implement those principles.

All of the mechanisms that comprise the Organodynamic framework use simplex and composite systems as their fundamental building blocks. However, for the initial development of each of these mechanisms of the Organodynamic framework, *we shall intentionally omit the use of the composite system*. The reason for this temporary omission is the initial reduction of complexity.

We shall occasionally include the composite system along with the simplex one when describing these mechanism and their constructs. However, we shall reserve the actual usage of the simplex system building block in these constructs until we reach the advanced construct named the Composite Organodynamic Web.

The Fourth OCS Organizing Principle: Reorganization

As discussed, the dynamical aspects of a systems theory concern actions, behaviors or doings of systems. Organodynamics chooses to address these

doings by describing the change of state of its ontological entities (simplex and composite systems) from one point in time to another.

Organodynamics refers to this change of system state as *reorganization*. It most often involves a change in the *system organization* portion of a system. It can also involve the change in both the population of the system as well as its organization. And, it can also be a change in population without any change in the system organization (although not usually). Any of these cases, however, is referred to as *reorganization*.

Change of state is also the mechanism used by classical Newtonian dynamics to model behavior. In this discipline the *state* of a particle or small body is represented by some parameters that specify its *position* and *momentum*. Three parameters define its position (for the three spatial dimensions) and three other parameters define its momentum (also for the three spatial dimensions).

But, Newtonian dynamics is also a dynamical systems theory because it is primarily interested – not just in the state of a particle – but in the change of state of a particle. In fact, Newtonian dynamics gives change of state of a particle a name: *trajectory*. One could say that the primary dynamical entity of Newtonian dynamics is the trajectory.

Organodynamics, also being a theory of dynamical system, is primarily interested in the change of state of its static entities: simplex and composite systems. It is tempting to also give the name *trajectory* to this change of state also.

But to do so would be misleading – mainly because the state of an Organodynamic system is so much more complex than the state of a particle that the analogy begins to break down.

Whereas a particle in Newtonian dynamics is simple enough so that its *trajectory* can be viewed as a simple *line, or path*, traced through six-dimensional Euclidean space. Unfortunately, the *state of an Organodynamic simplex or composite system* is not so simple as to lend itself to a graphical representation.

Rather, both the population and the organization of an Organodynamic system define its state. If either changes, then the state of the system has changed. Moreover, while the population is a simple set of components, the organization is a set of pairs (duples) of those components, and can be of varying length.

These configurations make tracking the changes in these systems complicated. And it does not lend itself to graphical representation. In particular, the nomenclature used to represent the *composite version* of Organodynamic systems is particularly complex and obviously resists graphical representation.

In any event, the description of change of state of Organodynamic systems becomes too complex right away to be able to describe it in terms of a “graphical pathway”.

Rather, it is more manageable to represent the change of state of an Organodynamic system as a *process* in which there are a number time steps. Within each of these times steps, the system is represented by its changed state for that time step. Such a process is referred to as a *system process*.

We defined a *system process* as a finite sequence of *time steps* where each time step is expressed as the *state* of an Organodynamic system. The representation of this state is abbreviated to its *organization*, although its *population* is also implied.

Thus, in Organodynamics we abbreviate the articulation of a system's state by using its *system organization*. This means that a *system process* then is articulated as a sequence of system organizations.

Thus the *system process* is a fundamental construct of the Organodynamics framework. Whereas, the *simplex system* and the *composite system* are the fundamental static entities of Organodynamics; the *system process* is the first fundamental dynamical entity of Organodynamics.

The Sixth OCS Organizing Principle: Uncertainty

But OCS – according to its sixth organizing principle - regards the notion of *uncertainty* as imbuing all active elements of its systems theory. Thus, it is time in our review narrative to introduce uncertainty.

In mathematics, *uncertainty* is embodied by probability spaces (or their concomitant probability distributions). And uncertainty can be introduced into dynamical models (processes) via a mechanism called the *stochastic process*. A stochastic process is simply a sequence of time steps wherein the state expression of each time step is replaced by a probability distribution describing the likelihoods of several possible states.

In other words, a stochastic process introduces *choice* (or *alternatives*) into the notion of *process*. This means that each time step of a stochastic process generally has several choices, or alternatives, each of which is expressed as a *system organization*. This is to be contrasted with the time steps of a *system process*, each of which has exactly one *system organization*.

Moreover, in a stochastic process, to each one of the set of choices that are associated with a single time step, a probability value is assigned. So, for each time step we actually have an entire probability distribution. And the sample points of this probability distribution are *system organizations* – all of the possible system organizations of an underlying population.

This is contrasted to the case of the *system process*, where each time step contained a single system organization, and no probabilities were involved.

This can be made clear if we discussed how to take a *system process* and convert to a corresponding stochastic process.

Recall that a *system process*, as defined above, is a *finite sequence of system states*. In Organodynamics, we abbreviate the articulation of a system's state by using its *system organization*. This means that a *system process* then is articulated as a *sequence of system organizations*, one for each time step.

In order to convert, or promote, this system process to a stochastic one, we have to change each of its time steps. As a system process, each of its time steps is articulated as a single *system organization* of some underlying system (either simplex or composite). To promote it to a time step of a stochastic process, we must replace this single *system organization* with a *probability distribution* whose sample space is an entire *collection* of system organizations – one of which is the initial system organization of the corresponding system process.

This is how we “add uncertainty” to a system process: by replacing its time

But, what are the semantics of this probability distribution? What does it mean? Organodynamics wants to characterize the change from one time step to the next within a process as being an *uncertain* phenomenon. In other words, *uncertainty* is used by Organodynamics for the purpose of characterizing the *dynamics* of organic systems.

This being the case, it is natural to specify that the probability distribution for a “current time step” determine the probabilities that the various system organizations in its sample space for the “next time step”.

Thus, for each time step of the *system process*, we have replaced its state (a single system organization) with a probability distribution whose sample space is a set of system organizations (of some underlying system) along with associated probabilities that each will be realized at the *next time step*.

What is special about this stochastic process, however, is that the sample spaces of each of its time steps contain *system organizations of an underlying Organodynamic system*. In Organodynamics, we call such a stochastic process an *organodynamic process*.

Notice that the *state* of a time step for an organodynamic process is a *probability space* that is described by its *probability distribution*. We shall call such a probability distribution – one whose sample space is the set of all possible system organizations of an underlying population – an *organodynamic distribution*.

The *organodynamic process* is also a fundamental construct of the Organodynamics framework. In fact, it is the second such fundamental process that we have encountered in this section on the dynamics of Organodynamics. The other one is the *system process*.

But, our introduction of the *system process* was merely as a stepping-stone to build up to the *organodynamic process*. After this, all of our attention is on the *organodynamic process*.

In fact, we shall use the *organodynamic process* as the basic building block to develop the elements of the remainder of the framework, including: Segments, Edges, Nodes, Organodynamic Graphs, Simplex Organodynamic webs, Composite Organodynamic Webs and Joint Composite Organodynamic Webs.

For example, a *segment* is a sequence of contiguous time steps whose probability distributions remain the same.

The Fifth OCS Organizing Principle: Autocogeneration

The kind of system change that we have been discussing is pretty simple because at any time step, only one system (as represented by its *organization*) gets realized at the *next step* of the organodynamic process.

That is, at one time step we have a single system organization, and at the next time step we still have exactly one system organization. It is true that a probability distribution is involved that makes it uncertain as to which one of several alternatives will end up being realized, or manifested, at the next time step. Nevertheless, after some selection is done, called *realization*, exactly one of these system organizations will be realized at the next time step.

It is true that living system processes go along for a while as a single system that changes states for a number of contiguous time steps. For example, a eukaryotic cell will go along as a single system for a while, changing its organization very rapidly, but remaining a single cell. The organodynamic process that we described above easily represents this.

In fact, if the same probability distribution were adequate for describing the changes that were undergone at every one of these time steps, then this organodynamic process would be a *segment* as defined above.

However, in real life, various punctuated events occur to these processes that involve multiple other systems. For example, at some point our eukaryotic cell will undergo *cell division*. Coming into the time step where the cell actually divides, the cell was one system in each of those time steps. However, after the division is complete, then two cells are now involved.

The opposite also occurs in real life where multiple systems come together to form a single system. For example, two molecules can join to create a compound molecule.

Thus, we had to introduce a mechanism into Organodynamics to that provided for these *split* and *join* events. We accomplished this via the concept of *organodynamic transforms*. These transforms are defined as mathematical functions that accept multiple system organizations as input and/or produce multiple system organizations as outputs. We defined a number of categories of these split and join transforms, but left the specification of individual transforms to further research.

(In fact, the “normal” transition between two time steps (say, within a segment) that only involved one system at each of these two time steps is regarded as a

simple special case of a transform. It is a transform that is neither a split nor a join. This generalization allows us to consider all transitions between time steps to be *transforms*. However, we were forced to invent the idea of transform in the first place because of the need to support these splits and join events.)

These transforms, then, enabled the idea of organodynamic process to be generalized into a collection of concurrent processes that can combine into single “threads” of “execution” and then split again into multiple other threads. In fact, we are now beginning to describe our organodynamic processes in a graphical manner that includes linear structures (segments) as well as “nodes”, where the split or join (or both) transforms occur.

In addition to *segments* and *nodes* we also defined a third construct, the *edge*. Essentially an *edge* is a contiguous set of time steps within an organodynamic process that is initiated and terminated by nodes. An edge could consist of exactly one segment, or of several contiguous segments.

Thus, we are now able to talk about a mathematical construct (stochastic process, organodynamic process) - that is essentially algebraic and analytical in nature - as though it were graphical in nature. The graphical element types involved are the *segment*, the *edge* and the *node*. However, these are all implemented as algebraic and analytical constructs that include probability distributions, stochastic processes and transforms of probability spaces.

Nevertheless, inspired by the graphical aspect of these constructs, we have given a name to any arrangement that involves segments, edges and nodes. We call it an *Organodynamic Graph*.

We have developed these *nodes* on top of the *organodynamic process* construct in order to implement the fourth OCS organizing principle, autocogeneration, into the Organodynamics framework. These nodes, representing the organodynamic transforms that they do, enable us to represent to self-creation, self-organization and self-management aspects of autocogeneration.

However, there is a second aspect of autocogeneration – intertwined with the first – that is not supported by the Organodynamic Graph structure. This aspect is the *regularity* endemic in the autocogeneration principle.

However, we can induce regularity into the framework by the injection of a *looping mechanism* into the Organodynamic Graph. The result is the *Simplex Organodynamic Web* – which is an Organodynamic Graph equipped with a loopback mechanism.

This injection is done as follows. First identify the repetition of steps within an Organodynamic Graph. Such a repetition can include segments, edges and nodes. And, we call such repetitions by the name of *routines*.

Next, add two new capabilities to the Graph. One is the ability of a routine to carry with it a time step identity – rather than “belonging to” specific time steps. Second add a reference to time steps that specifies which time step the

traverse to as its “next time step”. With these two additions, we have the capabilities required to extend the notion of Organodynamic Graph to that of the Simplex Organodynamic Web.

The Simplex Organodynamic Web is a very significant milestone in the development of the framework. This construct finally has enough “equipment” to be able to model a “whole organism” in a near-comprehensive manner. By this we mean that a significant subset of the seven OCS organizing principles can be represented to make a near-lifelike model of an object system.

In fact, many projects that seek to model organic systems will be content with such a model. For these projects, this approximation is good enough. It is not worth it for these projects to implement the remaining mechanism that will provide the other two OCS organizing principles.

For modeling projects that seek improved fidelity, the Simplex Organodynamic Web will be an incremental milestone. Those will press beyond this to implement the other two constructs that we review below.

The Seventh OCS Organizing Principle: Persistence

The end result of the first six OCS organizing principles is that the likely persistence of a specific Organodynamic web structure instance. This likelihood results in an expected duration (from the probability and stochastic process perspective) for an instance of such a web, and for the occasion of specific instances of the web to be long-lived. In addition, it has resulted in the expectation that web instances continue to be created for a quite lengthy expected set of time steps.

However, these expectations cannot be supported by the mechanism described so far alone. There are two more addition constructs with which we have equipped the Organodynamics framework in order to support this kind of persistence. These are: the Composite Organodynamic Web, and the Joint Composite Organodynamic web.

The Organodynamic Composite Web finally adds *composite systems* to the Simplex Organodynamic web. We mentioned earlier that we have been withholding the use of composite systems as building blocks of the more advanced construct that we have been building, such as segments, edges, nodes, Organodynamic Graphs and Simplex Organodynamic webs. The reason for this deliberate omission has been to temporarily reduce the complexity of constructing the structures.

However, now is the time to reintroduce the composite system to these constructs. This is the purpose of the Composite Organodynamic web. It is the Simplex Organodynamic Web that is repaired with the use of *composite systems*.

We provided an alternative articulation of nested systems, called the *process view* that permitted the representation of nested systems within these earlier

constructs. But it represented each level of organization of a nested system as though it were a distinct simplex system.

This approach using the *process view* of nested systems at least accounted for the existence of these subsumed nested systems. However, it provided a poor representation of the interrelationships involved. Thus the composite system representation is preferred. It is within the Composite Organodynamic Web that the composite system view is reintroduced and supported.

Since converting the process views of nested systems within a Simplex Organodynamic Web can be a lot of work, the Composite Organodynamic Web is defined as one of the six *approximations* of the Organodynamic modeling methodology.

The Joint Composite Organodynamic Web was introduced in order to correct an oversight – an oversight that infects both of the Organodynamic webs that have so far discussed: the Simplex Organodynamic Web and the Composite Organodynamic web.

The oversight is this: We have assumed that we already know *when* (at what time steps) system transforms (the nodes) occur. But we actually do not know this, because their occurrence is also probabilistic.

An example of this is the encounter of two small molecules in such a manner that they form a covalent bond. This event is also probabilistic – involving as much uncertainty as any of the simple time step changes that occurs within a segment or edge.

Yet we have been assuming that the time step in which this encounter of molecules occurs is known in advance. In order to be a purely stochastic model, we must submit the occurrences of these encounters, these nodes, to stochastic treatment also.

However, there is no easy way to do this within the approach to architecting Organodynamic webs that we have been pursuing. We must take a completely new approach, and disrupt our network-oriented view.

The approach that has been decided upon to handle this correctly is a time-honored one that is taken by many other fields of application of stochastic processes. Statistical mechanics is a case in point; and we shall follow their lead in this regard.

Unfortunately, this approach sacrifices intuitiveness for intellectual completeness. It also happens to lose its “time sequence” orientation and network analogy as the principle organizing mechanism of these graphical representations. Instead, it treats a change in time as just another dimension of a highly multi-dimensional space.

Also, each possible node type is a separate dimension of this space. This approach essentially identifies every varying domain in the model space as a

distinct dimension, and constructs a large multidimensional space where each varying domain is represented.

Thus, this space is the joint space that has dimensions for every possible node type as well as the separate dimension that includes each time steps.

This space is called the *joint state space* for the entire Organodynamic web. Every possible system state for each possible simplex or composite system for each possible times step has a “cell” in this large multi-dimensional state space.

Each of the cells of this state space represents possible *joint system organization at a specific time step for all time steps*. The set of all such cells represents the sample space for the entire Organodynamic web. Thus, this single joint state space is the sole state space for this Organodynamic web and it includes all time steps in the web.

As a sample space, each of its members is assigned a probability. The result is a single probability distribution that models the entire Organodynamic web. Such an Organodynamic web is called a Joint Composite Organodynamic Web, and is the *grand scheme* of the Organodynamics framework.

In Organodynamics, this construct, the Joint Composite Organodynamic Web, is the template that can model any organic system. It is equivalent to the idea of a “circuit” in our mythical “electronic circuit hobbyists toolkit” that we have been using as an analogy to the Organodynamics framework.

The Knowings of Organodynamics

We have talked a lot about *uncertainty* and postured it as a distinctive organizing principle of OCS and Organodynamics.

And we have used the mathematics of Information Theory extensively throughout – thus indicating the centrality of *uncertainty* to the foundations of Organodynamics. (As explained by Richard Kleeman in his NYU course on Information Theory and Prediction, “The central idea of information theory is to measure the uncertainty associated with random variables”.)

Of course, it is difficult to deny that the notion of *uncertainty* pertains to *knowledge*. For if it is possible to be certain or uncertain of anything, it would “certainly” have to be knowledge!

But this puts us squarely within the province of the branch of philosophy that the ancient Greeks called *Epistemology*. We aren’t doing philosophy here – only organizing this review chapter by three of its concepts.

However, if *uncertainty* is so central to Organodynamics – as it also is to Information Theory, then the obvious question that is begged here is:

Who is it that exhibits this uncertainty?

But, there is also a second question implied. This question stems from the fact that this “Who” that is *uncertain* is uncertain *about* something – some object. In other words, the “Who” is the subject of this uncertainty, and there is some other entity that is the *object* of the uncertainty.

Thus, the second question that is begged by all of this Information Theory stuff is:

What is the *object* of this uncertainty that is exhibited by the subject?

But, if there is a subject and an object aspect to this uncertainty, then it is reasonable to say that there is a relationship between the subject and the object and furthermore that the uncertainty pertains to that relationship.

Therefore, we shall conclude that, in Organodynamics (and in Information Theory?) that:

Uncertainty is a property of a relationship between a subject, called the *observer*, and an object, called the *observed*.

But, specifically, who is this subject and who is this object in this observational relationship?

In the practice of “doing science”, the answer is quite obvious and widely adopted, because “science is empirical and experimental”. The *subject* is the experimental scientist and the *object* is the entity being empirically observed through the experiment.

Theoretical science also concerns modeling in one form or another. The specification of theories and the construction of mathematical models are both speculative activities in which hypotheses are stated and models (either intellectual or computer) are constructed. Whereas the experimental scientist is the observer of experiments, the theoretical scientist is the observer of models.

In both cases, uncertainty is evident in the relationship between the observer and the observed. The observer is observing a probability space – and the intention is to resolve this uncertainty into a realization of one of its sample points.

But, what about *organic systems*? They are autocogenerative. They are their own creator. They are their own repairer. They are their own manager.

The *subject* has become the *object* within organic systems – and with Organodynamic Webs.

Could it be that “Organic systems are their own observer?” Could it be that they are their own knower?

The quantum physicists say that particles “wink in and out of existence” [Greene 2003]. They say that these particles exhibit uncertainty all by

themselves – whether there is an observer or not. This may be so – and it may be an example of uncertainty outside of an observational relationship.

Nevertheless - and this is something that is worthy of consideration - could it be that the *uncertainty* that we attribute to organic systems is uncertainty pertaining to their own self-organization, self-management?

I had better end before I get caught at waxing philosophic. But, hey, I didn't bring up the question of *uncertainty*. Information theory does. And - let's face it - I'm not the only investigator [Gleick 2011] who finds information theory entirely apropos to an increasing number of domains of application in western science – and just about everywhere else!

Complexity

Since *complexity* appears in the title of this text – *Organic Complex Systems*, it is reasonable to expect that, at the very least, a definition of the term be proffered somewhere in the theory.

Furthermore, it is also reasonable that a measuring function for *complexity* be put forth somewhere within the mathematical model presented.

However, none of this has occurred, and an explanation for this oversight is in order.

Complexity as Degree of Organization

Mention was made – in the chapters that introduced the *organodynamic system* - of a measuring function for the *degree of organization* of an Organodynamic system. However, none was defined. The definition of such a measure was suggested for further research. This was done mainly due to the perceived complexity of addressing the possible definitions of such a concept.

In fact, it is reasonable to suggest that a measure for the degree of organization of a system could serve as a definition of the degree of its *complexity*. Such a suggestion would be paramount to saying that the *complexity of a system* is limited to its *state*; and that *system complexity* is strictly an ontological concern.

This would imply that the complexity of a system could be ascertained at a single moment in time by sizing up the state of the system – its *degree of organization* - at that moment. This static approach to defining system complexity is reasonable, and is often assumed.

Complexity as Degree of Dynamism

But, such a static definition of system complexity would be a disappointment to many complexity science theorists and practitioners. They might hold that the true complexity of a dynamical system doesn't even get started until one beholds its dynamical aspects. This is certainly a reasonable observation.

But, if Organodynamics is having difficulty figuring out how to devise a definition of degree of complexity of a *static system* – because of the complexity of such an exercise, then one can imagine the difficulty in do so for both the dynamical and the static aspects of a complex dynamical system.

Even, so this is not the end of the conversation – or the difficulty. We just discussed the difficulties involved in attempting to define *system complexity* for the first two of the three philosophical aspects of our discussion in this chapter: 1) the ontological and 2) the dynamical. But we have yet to visit the third: the epistemological. This is the information theory aspect of Organodynamic systems - the concerns with uncertainty.

Complexity as Degree of Uncertainty

Suppose we were to start with a deterministic system. Then each aspect of the system is represented by a single state. Suppose that to one or more of these aspects we add multiple choices of states – including the original deterministic state for that aspect – to select from. And, we have also introduced uncertainty into the system.

Clearly adding more states to select from (uncertainty) to the initial one adds complexity to the system.

However, suppose that we further specify a probability for each of these multiple choices for this aspect of the system. Then, since we have added more choice for this aspect of the system and, as well, assigned probabilities to these choices, then we have imposed a probability distribution to represent this single aspect of the system – rather than the single “choice” that initially represented that aspect of the system.

Now, instead of imposing this probability (uncertainty) treatment to only one aspect of the system suppose we impose it onto many aspects of interest. Then, we have converted a deterministic system to a stochastic one. And, we have certainly added considerable complexity to that of the deterministic one.

What we have done is to bring to bear several mechanisms from probability theory and information theory to this system. These include the fact that information theory treats determinism as a special case of stochasticism. In addition, information theory treats *uncertainty* as a continuously varying measure (entropy) of a probability distribution that ranges from zero (completely deterministic) as a minimum all the way to $\log m$ (completely random) as maximum (where m is the number of sample points). On this scale the possibilities for the degree of uncertainty range continuously along this spectrum.

And our appeal to and utilization of these aspects of information theory has added a formal approach to *uncertainty*, being that the rather vague ideas of “randomness” and “determinism” have been rigorously defined by a mathematical function, named *entropy*, whose domain is the set of finite state probability spaces and whose range is the non-negative real numbers.

The conclusion we draw here is that introducing the notion of randomness as supported by the formalities of information theory have provided a third source of complexity into our system theory – in addition to the ontological and dynamical complexities already discussed.

We already admitted that we have not yet determined how to develop a measure of complexity within the static (ontological) domain – a measure of the degree of organization of a system. And, it is certainly beyond our reach at this time to push any measure of complexity into the dynamic realm.

We do have a measure of uncertainty – Shannon's entropy. However, we are resistant to defining it as our measure of system complexity because it is important to Organodynamics, and to OCS, to allow *organization* to vary independently of *uncertainty*, as we have illustrated on several occasions.

As for our meager attempts at defining a notion of static complexity, the task has proven too complex at this time. The same goes for dynamical complexity, and, of course, for any combination of the two.

Most likely, it would be most desirable to Organodynamics to define a notion of system complexity that combines the static and the dynamic into a single measure; but that treats uncertainty as orthogonal to these two, and continues to measure it according to its *entropy* – rather than as “complexity”.

In summary, we find it reasonable to define the notion of *system complexity* within the systemic domains of 1) *system organization* (static) and 2) *system dynamics*. The third realm, uncertainty, would then continue to be measured by Shannon's entropy.

Thus, regrettably, we leave the pursuit of a satisfactory definition of *system complexity* to further research.

Complexity-induced Modeling Methodology

The complexity of organic systems and of the Organodynamics framework has induced the necessity of providing guidelines for the modeler to follow in order to successfully manage a modeling project that uses the framework. These guidelines are best implemented by a well-considered modeling methodology.

But, more than that, this need for a methodology forces the framework to define and organize the construct that it defines so that they lend themselves to optimal use by the methodology.

For example, the complexity of both organic systems and the Organodynamic framework suggest that small pieces of the model be built and that they then be integrated into larger models. Such an approach necessitates that the framework define small structures that can be integrated through assembly into larger structures.

And, of course, there is always the constraint that the results have to exhibit, eventually, all seven of the OCS organizing principles.

The conclusion is that the complexity of both organic systems and of the Organodynamics framework results in the choice of structures and processes that ultimately have been defined by Organodynamics.

Summary of Organodynamics Structures and Methodology

Appendix 3 of this Part II provides a summary of the Organodynamics framework structures that have been reviewed in this chapter.

Appendix 4 provides a summary of the steps proposed by the Organodynamics modeling methodology.

Conclusions

The Organodynamics framework is a mathematical implementation of the seven OCS organizing principles. It provides a platform for modeling living and lifelike systems – referred to as *organic systems*.

The organic systems targeted by this model construction framework are of massive, perhaps bewildering, complexity. Consequently, this framework must be capable of developing models that are almost as complex as the organic systems that they represent.

Concomitantly, the Organodynamic framework must, itself, involve enormous complexity. After all, Organodynamics strives to comprehensively model living or lifelike organisms – admittedly a daunting task. Most current modeling technologies attempt detailed functional models only for focused subprocesses – subprocesses that cannot exhibit all seven of the OCS organizing principles. Organodynamics seeks to go beyond this to comprehensive model systems that exhibit all seven of these principles.

However, in order to be attractive and useful to system modelers, the framework must be as accessible and tractable as possible. The task of developing such a modeling system is obviously challenging. As credited to Einstein, “Everything should be made as simple as possible, but no simpler.”

We have attempted to imbue Organodynamics with enough complexity to be able to model its target organic systems; but at the same time to provide sufficient manageability so as to make it an attractive modeling framework.

The approach to the framework development has been to construct a mathematical theory that is constituted by elements that, collectively, embody the seven OCS organizing principals; and then to provide a modeling methodology that enables system modelers to incrementally and cumulatively develop addressable aspects of their system models and then to gradually integrate them into whole comprehensive system models.

Appendix 1: System Transforms

This appendix is an extension to the chapter above entitled “System Dynamics as Reorganization”. In that chapter, the notion of *system process* was introduced. It was defined as a set of contiguous system states, each of which represents the state of a specific system at increasing point in time.

Introduction to System Transforms

Review of the System Transform Concept

The notion of *system transform* was also introduced as a way in which the state of an organic system at one point in time can be changed to a subsequent state at a later point in time.

In other words, the *system transforms* describe how to get from one step in a *system process* to the next. Specifically, a *system transform* provides a mathematical *function* that maps the state of a system at one step of a system process to the next.

The chapter “System Dynamics as Reorganization” introduced twelve such transforms. In actuality it introduced twelve *categories* of transforms, and left the definitions of specific transforms that populated each category to future research.

However, the chapter did not describe these system transform categories in any depth, and left that task to this appendix.

System Transforms as Edges in a Graph

As described above, a single transform creates a “connection” from one system state at a certain point in time to the next system state at the next point in time. Obviously then a sequence of transforms is an alternate description of a sequence of system states – or, on other words, of a system process.

Such a sequence can be thought of as a “linear structure” – a sequence of “things” (states) arranged in a straight line. Thinking in a graphical metaphor, one could describe such as sequence as a “line”, and “arc” or an “edge” – to use graph-theoretic terminology.

Split and Join Transforms as Nodes in a Graph

However, most of these 12 transform categories – viewed as functions – either take multiple inputs or produce multiple outputs – or both. Such system transform categories are referred to as *split transforms* or *join transforms* or both simultaneously.

Thinking in graphical terms, these split and join types of transforms can be described as *nodes* of a graph.

The Makings of Networks

Thinking again in a graphical metaphor, then, these transforms – comprising both *edges* and *nodes* are the basic equipment we need to construct networks.

Of course, such “networks” actually have mathematical (set-theoretic) components such as sets, relations and functions. Nevertheless, they can be articulated as *networks* as well. In fact, we call these networks *system process networks*. And, this text has used them as the foundations of which the Organodynamic Web structure is initially based.

However, these networks will require additional machinery before they “grow into” Organodynamic webs. This machinery is developed in several chapters above.

Thus, below, you will find a more detailed description of each of these categories of system transform.

System Transform Categories

We shall use the remainder of this chapter to define a number of categories of these system transforms. Each of these categories has a descriptive name that attempts to characterize the nature of the transformations being defined. Specifically, we shall define the following transform categories:

- Reform
- Replicate
- Catalyze
- Consume
- Expel
- Unite
- Divide
- Acquire
- Divest
- Compose, Integrate
- Decompose, Disintegrate
- Precipitate
- Dissolve

Generally, any of these transforms can technically be either join transforms, split transforms or both. This is true because all of these transforms admit to a product space as their domain space, given that a product space of dimension one is an ordinary system space. This is also true of split transformations since a single system output can be considered as a system partitioning of cardinality one.

Reform

Sometimes the most specific statement that can be made regarding a change to a system is simply that it has changed. Either its population or its organization has changed, or both. This is captured by the *Reform transform*.

If for a given instance of Reform, the population is preserved, then the instance is said to be *homopopulous*. If for a given instance of Reform, the organization is preserved, then the instance is said to be *homo-organizational*. It is possible for system processes to be either, both or neither. If neither, then the special case can be called the *Identity transform*.

Specification

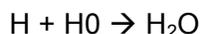
Reform is a category of *system transform* in which the codomain system has the same population as its domain counterpart; but the organization may be different. Formally:

Let $S_A = (P_A; O_A)$ be a system. Let $S_B = (P_B; O_B)$ be a system.

Then any system transform $T(S_A) = S_B$ is said to be a *Reform transform*.

Example

Consider the chemical formula:



On the left, we have a system whose population consists of three atoms - two hydrogen atoms and an oxygen atom. The organization of this population is $\{[H_2, O_1]\}$. Thus the system on the left, call it system A is:

System A = ({H1, H2, O1}; {[H2, O1]}).

However, the system on the right, system B, is:

System B = ({H1, H2, O1}; {[H1, O1], [H2, O1]}).

That is, systems A and B have the same population but different organizations. Thus, this formula is a manifestation of the Reform transform. Moreover, this instance of the transform is homo-populous, because the population is preserved by the instance of the transform.

Comments

Reform is not a *join* transform.

Replicate

Specification

Replicate is a category of *system transform* in which a new process for a new instance of a system is created. *Replicate* preserves both the population and the organization of the domain step of the domain process and creates the first step of a new process – which “operates concurrently with” the original process.

Example

Both DNA replication and creation of messenger RNA via their respective enzymes are examples of the Replicate transform.

Comments

Replicate is a *split* transform.

In previous sections of this text, we have admitted the possibility of copy errors in DNA and RNA replication. If we make that assumption, these transforms no longer produce the same deterministic result all the time. In such a case, we need a more complex apparatus than our system transform. Rather, we need a stochastic transform to model this.

We shall provide such a transform in a later chapter of this text. For the time being, however, let's proceed as though these replications are perfect every time, and therefore deterministic.

This same observation can be made for most, if not all, of the system transforms presented in this chapter. Because of this fact, we shall be providing stochastic versions of all of them in a later chapter – a chapter that discusses a construct that we call an Organodynamic web.

Catalyze

In biochemistry, catalysis is an important, even ubiquitous, transformation. This transformation permits a chemical reaction to be vastly sped up by involving a special molecule, called a catalyst, to participate in the reaction. The result of the reaction when the catalyst is involved is the same as it would be without its involvement – except that the reaction times is vastly more efficient.

Another interesting aspect of catalysis is that the catalyst remains unchanged after the reaction.

We shall define the Catalyze system transform to model this later behavior of catalysts – their preservation both as a population and as an organization after the reaction is completed.

The (perhaps) more significant aspect of catalysis, wherein it speeds up the reaction will not be defined at this time – simply because system transforms lack the machinery to accomplish that speed-up. This aspect of catalysis will have to wait until the chapter of this text in Organodynamic webs – where we shall embellish all of these systems transforms to handle stochastic processes. This embellishment will also permit us to model the “speed-up” aspect of the Catalyze transform.

Specification

Catalyze is a category of *system transform* in which two component systems interact in such a way that one is reorganized while the other remains unchanged. Formally:

Let $S_1 = (P_1; O_1)$, $S_2 = (P_2; O_2)$ and $S_3 = (P_3; O_3)$ be systems. Suppose T is a system transform such that $T(S_1, S_2) = \{S_1; S_3\}$.

System S_1 is called the *catalysts system*, or *catalyst*. System S_2 is called the substrate; and system S_3 is called the product of the transformation.

Then any system transform T is said a *Catalyze transform*.

The application of this category to cellular chemistry is obvious.

Example

The enzymatic reactions of metabolism are catalytic reactions. A large percentage of genes in all living species specify the production of catalytic enzymes that are essential to timely chemical reaction in living systems.

Comments

What is missing, at this point, is the notion of the enhanced efficiency of catalysis as compared with a more simple *reform* transformation. Unfortunately, we must wait until we have discussed the sixth organizing principle, *uncertainty*, before we can define this aspect of *catalyze*.

This is because the Organodynamics approach will be that catalysis changes the probability distributions associated with the stochastic process of the transform in such a way that the Shannon entropy is decreased.

This will be presented later in our discussion of the *autocogeneration* organizing principle – at which time we shall enhance all of these categories of transforms.

Consume

We need a transform to model the consumption of “food” by living entities. When a food component is taken in, it is added to the population. The consuming system is reorganized so that the newly added component can form essential relationships with the existing components of the initial system.

Specification

Consume is a category of *system transform* in which a component is added to the population of the system to result in a new system. The organization of the new system may also be changed. Biological systems often *grow* by consuming. Formally:

Let $S = (P; O)$ be a system, and let A be a set that is disjoint from P . Define new entities

$$\begin{aligned} P_A &= P \cup \{a\}, \text{ where } a \text{ is an element of } A. \\ O_A &, \text{ an organization of } P_A. \\ S_A &= (P_A; O_A). \end{aligned}$$

Then any system transform $T(S) = S_A$ is said a Consume transform.

Example

Food intake of living systems is modeled by the Consume systems transform.

Comments

Consumption constitutes system growth in Organodynamics.

A more elaborate form of the Consume transform can be defined. Such a transform would intake an entire system – not merely as component. Some subset of the population of the consumed system would then be “unioned” with the existing system population, which would then be reorganized.

Expel

We need a transform to model the elimination of “waste” by living entities. The Expel transform is the inverse of the Consume transform. When “waste” is eliminated, it is removed to the population. The eliminating system is then reorganized so that the newly removed component is no longer represented in any relationships (duples) in the organization of the resulting system.

Specification

Expel is a category of *system transform* in which a component is removed from the population of the system to result in two new systems. The first is the initial system, without a subsumed system. The other is the subsumed system.

In order for this to occur, the organization of the first new system must be changed to remove any permutation that contains the removed component. Biological systems often *defecate* by expelling. Formally:

Let $S = (P; O)$ be a system.

Define a new system without component “a”, called the *resident system*:

$P_A = C - \{a\}$, where a is an element of P.

O_A , an organization of P_A .

$S_A = (P_A; O_A)$.

Also define another new system constituted by element “a”, called the *effluent system*:

$P_B = \{a\}$.

O_B , an organization of P_B .

$S_B = (P_B; O_B)$.

Then any system transform $T(S) = \{S_A; S_B\}$ is said a Expel transform.

Example

Defecation or waste elimination in living systems are examples of the Expel transform.

Comments

<>

Unite

The Unite transform takes two systems from the domain space and creates a new system in the codomain whose population is the union of the other two populations.

This occurs very often in biochemical reactions between two or more molecules. They combine into a new molecule whose set of atoms consists of all of the atoms from the other two molecules. However, the new molecule is a reorganization of those molecules. The exact organization is peculiar to each such Unite transform.

Specification

Two systems combine into a single new system. The populations of the first and second systems combine by set-theoretic union to become the population of the new system. The organization of the new system can consist of any allowable organization (any set of related duples) of the new population.

Formally:

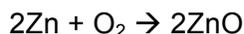
Let $S_A = (P_A; O_A)$ and $S_B = (P_B; O_B)$ be systems.
Then system $S_{AB} = (P_A \cup P_B; O_{AB})$, where O_{AB} is any allowable organization of $P_A \cup P_B$.

Then any system transform $T(S_A, S_B) = S_{AB}$ is said a Unite transform.

Special cases occur when the organization of either or both systems are subsets of the organization of the new system.

Example

An example of such a union from biochemistry is the oxidation of metals. For example, the oxidation of lithium by molecular oxygen yields zinc oxide. The chemical formula is:



To model this chemical reaction as a system transform, we analyze it as follows. In the domain of the transform, there are two systems, systems A and B. One has a population of 2 Zn atoms and an organization containing no duples (the empty set). The second system has a population of two oxygen

atoms. Their organization contains one duple of both of these atoms. The *Unite* transform specified in this example creates a new system in the codomain from these two domain systems. The population of this codomain system is the set-theoretic union of the two domain populations. The organization of the codomain system consists of two like duples. Each duple pairs a Zn atom and an oxygen atom.

Symbolically:

$$\begin{aligned} \text{System A} &= (\{ \text{Zn1}, \text{Zn2} \}; \{ \}) \\ \text{System B} &= (\{ \text{O1}, \text{O2} \}; \{ [\text{O1}, \text{O2}] \}) \\ \text{Unite(A, B)} &= (\{ \text{Zn1}, \text{Zn2}, \text{O1}, \text{O2} \}; \{ [\text{Zn1}, \text{O1}], [\text{Zn2}, \text{O2}] \}) \end{aligned}$$

Comments

Depending upon the characterization of the codomain of the *Unite* transform, a specific *Unite* transform may or may not be a *split* transform.

Divide

We need a model of cell division and other types of division in biochemical processes. The idea is that the components of a system are partitioned into a set of mutually exclusive subsets whose union is the initial population. Then, new systems are formed whose populations are these subsets. Then any legal organizations are formed on these populations to result in a new set of systems.

Specification

A single system divides into two new systems. The populations of the two new systems form a partitioning of the population of the original system. The organizations of the two new systems can be any allowable organizations of those systems.

Special cases occur when either or both of the organizations of the two new systems are subsets (sub-organizations) of the organization of the original system.

Another special case of division is *symmetric* division. This occurs whenever the cells of the partitioning share the same type and number of components as each other. Cell division in biology is a special case of symmetric division. Mitosis is a special case of symmetric division.

Formally:

Let $S = (P; O)$ be a system.
 Let $\{ P_A, P_B \}$ be a partitioning of P .
 Let O_A be and set of related duples of P_A .
 Let O_B be and set of related duples of P_B .

Furthermore, let $S_A = (P_A; O_A)$ and $S_B = (P_B; O_B)$ be systems.

Then any system transform $T(S) = \{S_A, S_B\}$ is said to be a *Divide transform*.

Example

As mentioned, cell division, specifically mitosis, is an example of the Divide transform.

Comments

Divide is a split transform.

Acquire

Acquire is a more sophisticated form of Consume, wherein the “consumed” entity is a system – not merely a “standalone” component.

Specification

Two systems combine into a single new system. The population of the second system becomes a component of the first system. The organization of the new system can consist of any complete set of duples of the new population.

Formally:

Let $S_A = (P_A; O_A)$ and $S_B = (P_B; O_B)$ be systems.

Define $P_N = P_A \cup \{P_B\}$

Define $S_N = (P_N; O_N)$, where O_N is set of duples of components of P_N .

Then any T such that $T(S_A, S_B) = S_N$ is said to be an Acquire system transform.

Acquire is a *join* transform.

Example

Most food intake and digesting in living systems is better modeled with Acquire than with Consume, particularly if the food itself begs to be modeled as a system in its own right.

Comments

Acquire is a join transform. If its result (codomain) is “separable”, then it may also be a split transform.

Divest

Divest is a more sophisticated form of Expel, wherein the “expelled” entity is a system – not merely a “standalone” component.

Specification

One system removes one of its components, which is also a system, resulting in two separate systems. The population of the first of these resulting systems is absent the removed nested system component. The second system is the removed nested system component. The organization of the first of the resulting systems is any complete set of pairs of the new population of this system. Formally:

Let $S_A = (P_A; O_A)$ and $S_B = (P_B; O_B)$ be systems, such that P_B is an element of P_A .

Define $P_N = P_A - \{P_B\}$

Define $S_N = (P_N; O_N)$, where O_N is complete set of duples of components of P_N .

Then any T such that $T(S_A) = \{S_N, S_B\}$ is said to be a Divest system transform.

Divest is a *split* transform.

Example

Most waste excretion in living systems is better modeled with Divest than with Expel, particularly if the expelled waste itself begs to be modeled as a system in its own right.

Comments

Divest is a split transform.

Compose, Integrate

In biology, molecules can combine to form macromolecules, macromolecules can form cells, cells can form tissues, etc. These formations create hierarchies of systems within systems with multiple levels of organization – all continuing to exist at their original levels of organization at the same time that they form the composite hierarchy.

This is precisely what occurs when two systems “integrate”. Two systems come together to create a new “super-system”. The two original systems still exist as systems in their own right. But now they also participate in the new integrated system as though they are components of that new system. And, as components, they organize into relationships with each other within the new system. Thus, a new level of organization is also created – the new integrated system.

Specification

The *integrate*, or *compose* operation describes how smaller systems combine by becoming the components of a larger system. Thus, the arguments to *compose* are already systems in their own right. The result of the operation is

another system, called the *composite system*, whose components are a certain representation of its argument systems.

Specifically, the components of the composite system are the *organizations* of the argument (input) systems. Any organization of the composite system must of course be a set of duples whose members are the components of the composite system – which are themselves duples. Thus, an organization of a composite system is a set of duples of duples.

Formally:

Let $S_A = (P_A; O_A)$, $S_B = (P_B; O_B)$, ..., $S_N = (P_N; O_N)$ be systems.

Define a new population $P_{Comp} = \{ O_A, O_A, \dots, O_A \}$.

Define any new organization on P_{Comp} . Call it O_{Comp} . So, O_{Comp} is a set of duples of components of P_{Comp} . In other words, O_{Comp} is a set of duples of duples.

Now that we have a population P_{Comp} and an organization on it O_{Comp} , then we have a new system $S_{Comp} = (P_{Comp}, O_{Comp})$.

Suppose T is a system transform such that $T(S_A, S_B, \dots, S_N) = S_{Comp}$. Then T is called a *Compose operation*, or alternatively, an *Integrate operation*.

Compose is an N-ary operation that takes as its arguments a set of systems. The result of this operation is a single system - the composite system.

It is often said that S_1, S_2, \dots, S_n are the “components of T ”. Of course, according to our definition, this statement is not actually correct. The true components of T are the members of its population, which are, in fact, the *organizations* of S_1, S_2, \dots, S_n – not the populations of S_1, S_2, \dots, S_n , and not the entire systems S_2, \dots, S_n themselves. However, as described above, we do use the phrase *composite systems* to describe the set of systems S_1, S_2, \dots, S_n .

This distinction is significant, so it is good to arrive at a understanding of it. The above definition and its distinction are actually motivated by common experience. So I shall provide an example, immediately following, that conveys that.

In the military, a *company* consists of some number of *platoons*. We are going to consider these organizations in marching formation. It is reasonable to model such a company as a composite system whose “components” are its platoons. And, it is reasonable to model each platoon as a system with a population and an organization.

The population of a platoon is a set of soldiers. And, an organization of a platoon is a description that places them within a platoon marching formation. Perhaps it is a 4 by 16 rectangle of soldiers, or some other such permissible

formation. It is possible to use an Organodynamic organization as a set of duples to define such a platoon marching formation. We shall omit doing that here; but point out that we can do that.

Now, lets look at how we would define a company as a population and an organization, as we need to do in Organodynamics. The population of the company needs to be some kind of representation of its platoons. We could say that the *company* is a set whose elements (components) are the populations of its platoons. But that is not quite what we want!

We do not want the soldiers of the platoons to be wandering around aimlessly! No. We want them to be organized into a platoon marching formation before we make them (integrate them, compose them) into a *company*. Therefore, we are better off insisting that the components (population) of a *company* consist of the *organizations* of the platoons – not the *populations* of the platoons. Thus, this is why we define the components of the population of a composite system to be the organizations of its components systems.

Integration (or composition) is the mechanism by which nested system hierarchies (also called *composite systems*) are constructed from smaller systems – which become their components. (Or, actually, whose *organizations* become their components.)

System nesting, or composition, is deeply important to OCS, and is, in fact the third OCS organizing principle. An in-depth chapter on Nested Systems is provided later in this paper.

Compose is a *join* transform.

Example

As mentioned, composite hierarchies that consist of multiple levels of organization within biological and ecological structures are the result if integration.

Even in the systems industry, for example when we “integrate two computer systems”, the integrate transform is occurring. When we truly integrate two computer systems, they continue to exist in their own right – even with their original populations and organizations. However, a larger system comes into being also. Its population can reasonably be modeled as the organizations of the two initial systems. And its organization is some set of duples of those organizations.

Comments

Integrate is a join operation. Depending upon whether the resulting joined system is “separable, it may also be a split transform.

Decompose, Disintegrate

When complex biological entities “die”, they decompose. This decomposition is a result of the fact that these systems were initially formed through the Compose transform, also called the Integrate transform.

This means that they are constituted by a hierarchy of systems within systems. When they decompose, they unwrap this hierarchy starting from the top.

The Decompose transform models this decomposition. It is the inverse of the Compose, or Integrate transform.

Specification

A nested - or composite - system “divides” into the multiple systems that are its component systems. Decompose is defined only for composite systems. And, this operation is defined in such a way that it is the inverse of the *compose* or *integrate* operation. Formally:

Let $S_{\text{Comp}} = (P_{\text{Comp}}, O_{\text{Comp}})$ be a composite system.

Since S_{Comp} is a composite system, then there are systems $S_A = (P_A; O_A)$, $S_B = (P_B; O_B)$, ..., $S_N = (P_N; O_N)$ that are the components of its population P_{Comp} .

Any transform $T(S_{\text{Comp}}) = \{S_A, S_B, \dots, S_N\}$ is said to be a *Decompose transform*, or a *Disintegrate transform*.

Suppose T is a system transform such that $T(S_A, S_B, \dots, S_N) = S_{\text{Comp}}$. Then T is called a *Compose operation*, or alternatively, an *Integrate operation*.

Decompose is a unary operation.

Example

The disintegration of any organic system is an example of the Disintegrate, or Decompose transform.

At the biochemical level, the breakdown of a macromolecule decomposes into its component molecules is also an example of Decomposition. Note that in this case, the resulting decomposed systems are, in fact, systems in their own right. Moreover, both their populations and their organizations are preserved from their level of organization of the organization within the initial composite system.

Comments

Decompose is a split transform.

Precipitate

Sometimes a small subset of the components of a system happen to form relationships with each other in such a way that they begin to behave as a single distinct system in their own right.

For example, imagine a “primordial soup” that contains atoms, but no molecules. Further imagine that at a certain point in time, the dynamics of these atoms brings some small number of them together. If the right conditions occur, enough of the covalently bonded pairs of these atoms come together to the extent that we would consider this collection to be a molecule.

At this point, it is reasonable to remove this collection of interrelationships – this molecule – from the milieu and to identify it as a distinct “small system”. To do this, we want to remove the duples involved in these interrelationships from their larger *system* organization and with them form a new *system organization*. And, of course, we would also want to remove the atoms involved in these duples from the larger *population* of the milieu system and with them form a new *population*.

So, we are discussing the idea of starting with a one system (the “milieu” of atoms) which has its own population and organization; and then removing some of the duples of that organization in order to create a smaller organization; and then removing some of the components of that population in order to create a smaller population. Finally, we are creating a smaller system from these whose population is the removed population and whose organization is the removed organization.

We call this newly created smaller system a *precipitate* of the first larger system. Moreover, we call the operation that produces this precipitate the *precipitate* transform.

More precisely, the *precipitate transform* takes as its only input system a single (larger) system. It produces two systems as outputs. The first of these outputs is the *precipitate* system. The second is the remainder of the input system after the duples and the components of the precipitate system have been removed. We shall call this second system the *residual* system of the transform.

Of course, to be able to do this, none of these atoms in the population of the precipitate can be members of any other duples in the organization of the residue system. This means that when selecting duples to be part of the organization of the precipitate, one must be careful that none of them share a component that is also shared by any duple of the residual.

This will become clearer with the following example. Within our milieu of atoms and their covalent bonds a small molecule can form from the covalent bonds represented by the duples in the system organization of this milieu. Such a molecule must not participate in any covalent bonds involving other atoms that are not in this same molecule.

An implication of this is that, if all covalent bonds are used to model the organizing relationships between the atoms of the milieu, then only the largest

macromolecules can be “precipitated out” using this transform. For example, the peptide bonds between two amino acids in a protein polymer are – at a lower level of organization – actually covalent bonds between atoms of the two amino acid involved.

The point is that these covalent bonds would prevent the amino acids from being “precipitated out” of the protein polymer. One would first have to “ignore” any covalent bonds involved in a peptide bond before precipitating its amino acid molecule from the protein. The *Reform* transform can be used to provide such “ignoring”.

In order to render the precipitate transform as more efficient, it would be desirable for it to be able to “precipitate out” multiple “small systems” in a single operation. We shall, then, define it so.

Thus, we shall define the *precipitate transform* so that it operates on a single input system and produces two types of outputs. The first is a collection of systems – the precipitates – that have been removed from the input system. The second is the residual system this is constituted by the remainder of the input system after the residual systems have been removed.

Of course, care must be taken so that the residual system actually forms a *system* in the sense that the duples of its organization contain no members that are not components of its population.

Specification

Suppose $S = (P; O)$ is a system, and that $O' = \{ (a_i, b_i) \mid \forall i = 1, k \}$ are duples in O . Further, assume that neither a_i nor b_i is a member of any duple in O that is not a member of O' .

Further, form $P' = \{ a \mid a \text{ is a member a duple of } O' \}$.

Define $S' = (P'; O')$. S' is called a *precipitate* of S .

Define $R' = (P - P'; O - O')$. R' is called the *residual of S with respect to S'*.

The transform $T(S, O') = (S', R')$ is called the *Precipitate transform*.

In words, the *Precipitate transform* separates a sub-relationship (O') of duples from an existing system (S) by removing it's duples from the existing system's organization, removing its components from the existing system's population and creating a new system (S') whose organization is the removed organization (O') and whose population is the removed components.

With repeated application of this transform to a system S , on can “precipitate out” multiple precipitate systems. In this case, then,

$$T(S) = (\{S_1', S_2', \dots, S_k'\}, R'), \text{ where } S_1', S_2', \dots, S_k' \text{ are precipitates of } S.$$

Example

An enzyme that separates an amino acid from a polypeptide or protein macromolecule performs an instance of this transform against the macromolecule.

Comment

A nested, or composite, system can be formed by first applying the *Precipitate transform* to a large system in order to isolate several small molecules. Subsequently, a composite system can be formed from these precipitates via the *Integrate* transform.

Precipitate and Divide are similar. The distinction is that *Divide* starts by removing components from a population and then removes duples from the associated organization to accommodate the component removal. On the other hand, *Precipitate* begins with the removal of duples, and subsequently removes components in compensation.

Precipitate is a split transform

Dissolve

Dissolve is the inverse of *Precipitate*. It takes one system and places it inside of another.

For this to occur, the populations of the two systems (and by implication their organizations) must be disjoint.

Specification

Suppose $S_1 = (P_1; O_1)$ and $S_2 = (P_2; O_2)$ are systems.

Define $S_1' = (P_1'; O_1') = (P_1 \cup P_2; O_1 \cup O_2)$

The transform $T(S_1, S_2) = (P_1 \cup P_2; O_1 \cup O_2)$ is called the *Dissolve transform*.

With repeated application of this transform to a system S , one can “dissolve” multiple systems into another system. In this case, then,

$$T(\{S_1, S_2, \dots, S_k\}, R) = (P_R \cup P_1 \cup P_2 \dots, P_k; O_R, O_1 \cup O_2 \dots, O_k), \text{ where } R, S_1, S_2, \dots, S_k \text{ are systems. } R = (P_R; O_R),$$

Example

The addition of an amino acid molecule to a growing polypeptide chain can be modeled via the Disintegrate transform. As it also can via the *Acquire* transform).

Comment

The use of the *Disintegrate* transform in sequence with the *Dissolve* transform can “break down” and then mix a composite system into a “solution” system.

Dissolve and Unite are similar. Both begin by defining the new population as the set-theoretic union of the population of two systems. The difference is that Unite allows the formation of the organization of the new system with any duples of the new components. On the other hand, *Dissolve* requires that the new organization be formed as the union of the organizations of the two input systems.

Appendix 2: Organodynamic Transforms

The content of this appendix defines the set of *organodynamic transforms* that were introduced and briefly described in the chapter above entitled “Simplex Organodynamic Graphs”.

This is accomplished by starting with the *system transforms* that were introduced and described in the chapter above entitled “System Dynamics as Reorganization”, and that were defined in Appendix 1 above. These *system transforms* are promoted in this appendix to *organodynamic transforms* by adding probability machinery to them, as explained below.

Introduction to Organodynamic Transforms

Review of System Transforms

Recall that the *system transforms* of Appendix 1 each define ways in which the state of an organic system at one point in time can be changed to a subsequent state at a later point in time.

In other words, the system transforms describe how to get from one step in a *system process* to the next. Specifically, a *system transform* provides a mathematical *function* that maps the state of a system at one step of a system process to the next.

But *system transforms* and their resulting *system processes* are deterministic in nature. But OCS requires its processes to be able to exhibit *uncertainty*. This allows for determinism, but cannot require it. OCS does require a systems representation that generally exhibits randomness and uncertainty. Determinism should be treated as a special case of uncertainty.

The mechanisms of probability theory provide for all of this. Thus, Organodynamics uses aspects of probability to represent system processes.

Approach Promoting System Transforms to Organodynamic Transforms

The approach the Organodynamics takes to implement this is to add probabilities to each step of its processes. For *system processes* this is accomplished making certain additions to each *system transform category* – since a system process can be characterized as a sequence of system transforms.

In order to achieve the desired uncertainty, there are two things that must be added to each system transform categories. These are:

1. Multiple choices
2. Probabilities to each of these choices

Recall that each system transform maps the system state at one point in time to another system state at the next point in time. It is these system states that we shall augment with the above two additions. Here's how.

First recall in Organodynamics that a system state is not simply the set of components of the system. Rather, it is something related but more complicated. System state is defined as an *organization* – essentially a configuration - of the underlying system's components.

Moreover, this *organization* at time step "t" represents the organization of the system being described at time "t+1". (This is a convention that we required in order to be consistent with the enhancement that we are about to add.)

Since we want to add more choices – "multiple choices" as mentioned above then instead of having the single *system organization* that we have now, we shall add more such system organizations!

The question is, though, what do all of these new organizations represent. The answer is:

These organizations represent all of the possible system organizations of the same underlying system that could *possibly be realized* at the next time step after the current one in the process.

So, we have introduced uncertainty. Instead of the state of the current system being the exactly determined *organization* of the underlying system that will be realized at the next time step, instead it is an *entire set* of organizations of the underlying system – all of which can possibly be realized then.

This takes care of the first thing we need to add: the multiple choices. But we also must add the second thing: the probabilities. Thus, we assign probabilities to each of the multiple choices that we have already identified. And, of course, these probabilities must sum to 1, since for any realization, exactly one of these choices will be realized.

The Resulting Organodynamic Transform

Of course, what we have now is a *probability space*, complete with a probability distribution.

In fact, this probability distribution is now the new *state* of our current time step.

But, we have added so much "machinery" to our *system transform* concept and to its corresponding *system process* concept that we need to give the new names.

We shall name our embellished system process an *organic process*. And the enhanced system transform we shall name an *Organodynamic transform*.

Let's summarize in a more complete manner what we have. The state of a times step of an *organic process* is no longer a single *system organization*.

Rather, it is a probability distribution whose sample space is a *set of system organizations* – all of which are organizations of the same underlying system, and all of which have some probability of being realized at the next time step.

Organodynamic Transform Categories

In this appendix, we shall define the organodynamic transform categories that we introduced in the chapter entitled “Simplex Organodynamic Webs”. We shall do this by embellishing some of the *system transform* categories that we identified in the chapter on the reorganizational organizing principle.

In the introduction above, we describe the organodynamic transform in terms of *unconditional stochastic processes*, where each system state per time step is represented by a single probability distribution. However, according to the *uncertainty model gradient* we must also be able to form organodynamic transforms for Markov chains – which are a special case of the *conditional stochastic processes*.

Thus, in the remainder of this appendix, we shall deal with the case that our organic processes are Markov chains. And, the state of a time step of a Markov chain is articulated as a probability matrix (the Markov transition matrix) rather than a single probability distribution.

Nevertheless, such a state is still a probabilistic state. It differs from what we described above merely by being a more complex probability state.

Recall that our task is to convert the *system transform categories* of Appendix 1 to corresponding *organodynamic transform categories* by (somehow) adding more choice and probabilities.

Our strategy will essentially be to convert the “state” of each of those system transform categories from a single *system organization* to a Markov transition matrix. That is, whereas the *state* of a system transform is defined to be a *system organization*, the *state* of an organodynamic transform is a *Markov transition matrix* that represents possible transitions from a *set of system organizations* at time “t” to that same set of *system organizations* at time “t+1”.

Of course, any such Markov transition matrix will have rows and columns that represent an entire *state space* of *possible system organizations* – any one of which will occur for the current step of the chain. Thus, we will not be leaving behind the notion of system organization as state. We shall just be dealing with entire state spaces of possibilities of them.

Each of these possible system organizations will be represented by a row of the transition matrix. Also, if the step in question is homogeneous, then the columns will represent the same set of possibilities.

Now, each of these system organizations in the sample space shares the same underlying system population. That is, the system organizations in the sample space are all different reorganizations of the same system population. It is

these underlying populations that our system transform categories will be working with.

They will be taking these underlying populations, performing transformations on them, and yielding resulting system populations. It is these resulting transformed system populations that will then be used to develop a sample space of system organizations – possible organizations of that resulting population. This new set of possible system organizations will be the new sample space for the resulting transition matrix that we are calculating.

As we have said, the transform categories do not dictate what the new set of possible system organizations are. (That is left to specific transforms.) The transform categories only concern themselves with calculating their underlying shared population.

And, as discussed above, we have named these new transform categories *organodynamic transform categories*. And don't forget that they form the nodes of our Organodynamic webs.

We will approach these definitions by taking each of the system transform categories and embellish it into its corresponding organodynamic transform category. However, in the interest of space and time, we shall develop only a representative set and the organodynamic transform categories in this text.

The pattern of how to develop an organodynamic transform from its corresponding system transform is relatively straightforward – as described in the previous subsection. So, it is expected that the readers will be able to make the generalization and develop the remainder themselves.

Presently, we shall define five of the organodynamic transform categories for use in our Organodynamic webs, and leave the development of the remainder to the reader.

The six systems transform categories that we shall presently develop in this chapter are:

1. Reform
2. Identify
3. Catalyze
4. Unite
5. Divide
6. Integrate

The Reform Organodynamic Transform

Recall that the Reform systems transform was loosely defined in the chapter on the fourth organizing principle as a function in which:

The codomain system has the same population as its domain counterpart; but the organization is different.

One thing to notice is that this transform is a *unary* transform. That is, it takes as a domain element an Organodynamic *system state* and maps it to a codomain system state in which the system is the same as the domain system and the organization is some organization on the same system population as the codomain system.

The organodynamic transform version of Reform is also a unary transform. It will take as its only input a Markov transition matrix for time t-1 and produce as its only output a new transition matrix at time t.

The focus of the Reform organodynamic transform category will be on calculating the underlying population of the row state space of the output transition matrix for time t.

In order to calculate this, the transform must ascertain the underlying population of the input transition matrix for time t-1. To ascertain this underlying population, one must identify the systems from which the row space organizations come. All of the states (organizations) in that row space share the same underlying population. This is the population upon which any organodynamic transform in this category will operate.

And here is how it will operate on that population: It will “leave that population alone”. That is, it will perform an Identity map on that population at time t-1 in order to ascertain the underlying population for the row space for time t.

This is the final action that the category is responsible for. Any actual transform of the Reform category must then proceed to generate some set of possible system organizations from this underlying population. This set will become the row state (sample) space for the transition matrix for time t.

The specific transform may impose rules, algorithms or restriction on what this state space is or can be. In addition, the specific application that is being modeled may also impose conditions on these outcomes. This fact is also determined by the specific Reform transform. (These shall be referred to as “the conditions”.)

In addition, the columns state space must also be ascertained. The conditions will impose the column state space. If the node is deemed homogenous by the conditions, then it will be the same as the row state space.

Also, the conditional probabilities in the body of the matrix must also be ascertained. The conditions will impose the conditional probabilities.

It might be expected that we would define Reform to take one system as its input and produce one system as its output. However, for reasons of operational convenience, we will define the Reform transform to operate on an entire set of systems at time t in such a way that each of the systems will be reformed at time t+1. Formally

Let $S_1 = (P_1; O_1)$, $S_2 = (P_2; O_2)$,... $S_n = (P_n; O_n)$ be systems.
Also, let $S'_1 = (P_1; O'_1)$, $S'_2 = (P_2; O'_2)$,... $S'_n = (P_n; O'_n)$ be systems.

In other words, the S' systems have the same population but possibly different organizations than their S counterparts.

Suppose T is a system transform such that

$$T(\{S_1, S_2, \dots, S_n\}) = \{S'_1, S'_2, \dots, S'_n\}.$$

T is called the *Reform* transform.

The Identity Organodynamic Transform

For reasons of operational convenience, we will define a transform that “does nothing”. That is, it maps a set of systems at time t to themselves at time $t+1$. It does not create a new process (Markov segment). Rather it continues with the same process and with the system in the same state at time $t+1$ as it was at time t . Formally:

Let $S_1 = (P_1; O_1)$, $S_2 = (P_2; O_2)$, ... $S_n = (P_n; O_n)$ be systems.
Suppose T is a system transform such that

$$T(\{S_1, S_2, \dots, S_n\}) = \{S_1, S_2, \dots, S_n\}.$$

T is called the *Identity* transform.

The Catalyze Organodynamic Transform

Recall that the Catalyze systems transform was loosely defined in the chapter on the fourth organizing principle as follows:

Catalyze is a category of *system transform* in which two component systems interact in such a way that one is reorganized while the other remains unchanged. Formally:

Let $S_1 = (P_1; O_1)$, $S_2 = (P_2; O_2)$ and $S_3 = (P_3; O_3)$ be systems.
Suppose T is a system transform such that $T(S_1, S_2) = \{S_1; S_3\}$.

System S_1 is called the *catalysts system*, or *catalyst*. System S_2 is called the substrate; and system S_3 is called the product of the transformation.

One thing to notice is that this transform is a *combination join and split* transform. That is, it takes as a domain element two Organodynamic *system states* and maps the pair to a pair of codomain system states. One of the resulting codomain systems is the same system as one of the input systems. The other of the resulting codomain systems is, however, different from the other codomain system. That is, the second system is modified, transformed; while the first system is preserved, as-is.

The organodynamic transform version of Catalyze is also a *combination join and split* transform. It will use as its only input two Markov transition matrices for time $t-1$ and produce as its output a pair of transition matrix at time t . One of

those pair will be the same as one of the input matrices, while the other will be transformed.

The input, and the output, transition matrix that is preserved is referred to as *the catalyst*. The input that is transformed is referred to as the substrate. And the output that has been transformed is referred to as the product of the transform.

All Catalyst-type transforms exhibit a preservation of population of the input populations. Whether the input organization is preserved is determined by the specific Catalyst transforms.

The focus of the Catalyst will be on calculating the underlying population of the row state spaces of the *product* output transition matrices for time t . This is straightforward, because, since we have preservation of population, then the product population must be the same as the substrate population.

In order to calculate this, a Catalyst-type transform must ascertain the underlying population of the substrate transition matrix for time $t-1$. To ascertain this underlying population, it must identify the systems from which the row space organizations of the substrate matrix come. All of the states (organizations) in that row space share the same underlying population. This is the population upon which any organodynamic transform in this category will operate.

And here is how it will operate on that population: It will “leave the underlying population of the substrate alone”. That is, it will perform an Identity map on that substrate row space population at time $t-1$ in order to ascertain the underlying population for the row space for the product for time t .

The catalyst transition matrix must be the same at time t as it is for time $t-1$.

Any actual transform of the Catalyst category must then proceed to generate some set of possible system organizations from this underlying product state space underlying population. This set will become the row state (sample) space for the transition matrix for the product population at time t .

There is one additional consideration that we were unable to model for the Catalyst system transform category. However, we can now model this for the Catalyst organodynamic transform category. This is the significant fact that a catalyst is supposed to, in some sense, “speed up” the chemical reaction.

The “sense” in which we shall model this “speedup” is to increase its probability of occurring. Whether this is adequate is a reasonable point to argue. However, this is the mechanism that we shall use to model catalysis in Organodynamics.

This is modeled by imposing the requirement that a specific system organization of the product sample space have an elevated probability over the other system organizations in that sample space – at least for the conditions for which the catalyst is to behave as a catalyst.

The specific Catalyst-type transform may impose rules, algorithms or restriction on what this state space is or can be. In addition, the specific application that is being modeled may also impose conditions on these outcomes. This fact is also determined by the specific Reform transform. (These shall be referred to as “the conditions”.)

In addition, the columns state space must also be ascertained. The conditions will impose the column state space. If the node is deemed homogenous by the conditions, then it will be the same as the row state space.

Also, the conditional probabilities in the body of the matrix must also be ascertained. The conditions will impose the conditional probabilities.

The Unite Organodynamic Transform

Recall that the Unite systems transform was loosely defined in the chapter on the fourth organizing principle as a function in which:

Two systems combine into a single new system. The populations of the first and second systems combine by set-theoretic union to become the population of the new system. The organization of the new system can consist of any allowable organization (any set of related duples) of the new population.

Formally:

Let $S_1 = (P_1; O_1)$, $S_2 = (P_2; O_2), \dots$, $S_n = (P_n; O_n)$ be systems.
Then system $S_U = (P_1 \cup P_2, \dots, \cup P_n; O_U)$, where O_U is any allowable organization of $P_1 \cup P_2, \dots, P_n$.

Suppose T is a system transform such that

$$T(P_1 \cup P_2, \dots, \cup P_n) = S_U \text{ is said a Unite transform.}$$

One thing to notice is that this transform category is a *join* transform. That is, it takes as its domain elements two Organodynamic *system states* and maps the pair to a codomain system state in which the population is the set-theoretic union of the two underlying populations of the domain systems. The organization of the codomain is some organization of this unioned population.

The organodynamic transform version of Unite is a *join* transform. It will take as its inputs a set of Markov transition matrices for time $t-1$ and produce as its only output a new transition matrix at time t .

The focus of the Unite organodynamic transform category will be on calculating the underlying population of the row state space of the output transition matrix for time t .

In order to calculate this, the transform must ascertain the underlying populations of the two input transition matrices for time $t-1$. To ascertain these underlying populations, one must identify the systems from which the row spaces organizations come. All of the states (organizations) in each those N

row spaces share the same underlying populations. These are the N populations upon which any organodynamic transform in this category will operate.

And here is how it will operate on those two populations: It will perform a set-theoretic union on them in order to produce a single population at time t.

This is the final action that the category is responsible for. Any actual transform of the Unite category must then proceed to generate some set of possible system organizations from this underlying population. This set will become the row state (sample) space for the transition matrix for time t.

Specific Unite-type transforms may impose rules, algorithms or restriction on what this state space is or can be. In addition, the specific application that is being modeled may also impose conditions on these outcomes. This fact is also determined by the specific Reform transform. (These shall be referred to as “the conditions”.)

In addition, the columns state space must also be ascertained. The conditions will impose the column state space.

Also, the conditional probabilities in the body of the matrix must also be ascertained. The conditions will impose the conditional probabilities.

The Divide Organodynamic Transform

Recall that the Divide systems transform was loosely defined in the chapter on the fourth organizing principle as a function in which:

A single system divides into two new systems. The populations of the two new systems form a partitioning of the population of the original system. The organizations of the two new systems can be any allowable organizations of those systems.

Formally:

Let $S = (P; O)$ be a system.
 Let $\{ P_A, P_B \}$ be a partitioning of P.
 Let O_A be and set of related duples of P_A .
 Let O_B be and set of related duples of P_B .

Furthermore, let $S_A = (P_A; O_A)$ and $S_B = (P_B; O_B)$ be systems.

Then any system transform $T(S) = \{S_A, S_B\}$ is said to be a *Divide transform*.

One thing to notice is that this transform category is a *split* transform. That is, it takes as its domain a single Organodynamic *system state* and maps it to a pair of system states in the codomain. In fact, this Divide is the only organodynamic transform exemplified in this appendix.

The populations of the resulting two system states form a partitioning of the population of the input system state. This partitioning results in the preservation of the population of the input system.

The organodynamic transform version of Divide is a *split* transform. It will take as its input a single Markov transition matrix for time $t-1$ and produce as its output a pair new transition matrices at time t .

Being a split transform, Divide must contend with the fact that there is an additional probabilistic step that takes place. The reader will recall that split transforms first involve a partitioning of the population of the input system into multiple populations of the output systems. And, as indicated, this is a probabilistic split whose sample space is enumerated as multi-cell combinations.

Thus, eventually when specific instances of the Divide transform category is identified, probabilities must be assigned to these combinations. Further, the reorganization of these separate populations after the split has occurred is modeled via the further probabilistic event. (All of this is represented within the mitotic process of cell division in biology.)

The focus of the Divide organodynamic transform category will be on calculating the underlying populations of the two row state spaces of the output transition matrix for time t – and calculating them in such a way that the from a partitioning of the row state space if the input transition matrix.

In order to calculate this, the transform must ascertain the underlying population of the input transition matrix for time $t-1$. To ascertain this underlying population, one must identify the system from which the row space organization comes. All of the states (organizations) in this row space share the same underlying population. This is the population upon which any organodynamic transform in this category will operate.

And here is how they will operate on those this population: They will first contrive a partitioning of this population. The parts, or cells, of this partitioning will then become the underlying population of the two row spaces for each of the transition matrices for time t that are the products of this Divide transform.

This is the final action that the category is responsible for. Any actual transform of the Divide category must then proceed to generate some set of possible system organizations from these two underlying populations. These two sets will become the row state (sample) spaces for the two transition matrices of this transform for time t .

Specific Divide-type transforms may impose rules, algorithms or restriction on what these state spaces are or can be. In addition, the specific application that is being modeled may also impose conditions on these outcomes. This fact is also determined by the specific Divide transform. (These shall be referred to as “the conditions”.)

In addition, the columns state spaces must also be ascertained. The conditions will impose the column state spaces.

Also, the conditional probabilities in the body of the matrix must also be ascertained. The conditions will impose the conditional probabilities.

The Integrate Organodynamic Transform

Recall that the Integrate systems transform – also called Compose - was loosely defined in the chapter on the fourth organizing principle. The basic idea is to take an initial set of systems and form a new system whose components are, in some sense, that initial set of systems.

The initial set of systems is called the *component systems*; while the new system is called the *composite system*.

A significant aspect of *Integrate* is that the new system that is formed by it is “at a higher level of organization” than the initial set of systems that become its population of components. In fact, to say that one system (or a set of systems) is a *component* of another system is what “levels of organization” mean – and also what is meant by the phrase “nested systems”. Thus, in this light, the Integrate (or Compose) transform is the mechanism by which nested systems are formed.

But, the specifics of how to define the *Integrate* transform unearth some interesting challenges that we must overcome in order to well-define this transform. In particular, there are two challenges that must be resolved.

The Population of the Composite System

The first is to determine exactly what the *population* of the new “output” composite system is; and the second is to determine the same for its *organization*.

Recall that we want the *population* new composite system to – in some sense – be the set of input systems. But each of these input systems – being systems – is a complicated entity – being a set of the form (P; O) as described above. This is perhaps more complicated than we need. We could simply take the populations “P” from this set of systems and define the population of the composite set to be this set of populations. Unfortunately, that approach would lose the important system organization information for each of these component systems.

That leaves us with using the *organization* part, “O”, of each of these component systems as the components of the population of the new composite system. And this would work, because these organizations contain the information we need.

Therefore, for the Integrate transform, we shall define the *population* of the (output) composite system as the set of organizations of the transform’s component systems.

The Organization of the Composite System

But that leaves us with this: What is the *organization* of the new *composite system*? We shall see that this Integrate category of transforms that we are defining here will only require that the organization of this new composite system be any “legal” organization of its population. That is, at this time we shall only require that the organization of any composite system be a set of related duples whose elements are the components of this composite system.

Of course, the components of this composite system are, themselves, sets of related duples of the component systems of the transform. Therefore, we must conclude that the *organization* portion of a composite system is a set of related duples whose elements are two sets of related duples.

Of course, this statement merely imposes restraints on the kind of entity that the organization portion of a composite system can be; and does not “narrow it down” very much. We shall leave this “narrowing down” to more specific sets of duples to the specific Integrate transforms of the Integrate category. Our definitions in this subsection will regard all specific transforms of this category.

Thus, we shall leave the imposition of any more specific constraints on these organizations to any of a number of specific Integrate-type transforms – all of which are beyond the scope of this text.

Formal Definition of the Integrate Transform

Now that we have completed these comments regarding the intention of the Integrate transform category of organodynamic transform, let's proceed to provide a more formal definition of it. In order to provide such a definition, let's briefly review what the chapter on system transforms above had to say about the Integrate system transform category. This is necessary because we shall inherit much of the system transform definition in our of the organodynamic version definition.

The system transform definition of the Integrate category takes one or more input systems (called *component systems*) and forms a new system (called the *composite system*) whose *population* is the set of organizations of the component systems, and whose *organization* is any set of related duples of those components.

This is consistent with the choices that we introduced above for the resulting *population* and *organization* of composite system – which is the “output” of the transform category.

Specifically, the components of the composite system are the *organizations* of the argument (input) component systems. Any organization of the composite system, therefore, must be a set of duples whose members are the components of the composite system – which are, themselves, duples. Thus, any organization of a composite system is a set of duples of organizations – which makes them a set of duples of sets of duples.

Formally:

Let $S_A = (P_A; O_A)$, $S_B = (P_B; O_B)$, ..., $S_N = (P_N; O_N)$ be systems.

Define a new population $P_{Comp} = \{ O_A, O_A, \dots, O_A \}$.

Define O_{Comp} to be any any organization on P_{Comp} . Therefore, O_{Comp} is any set of duples of components of P_{Comp} . This means that O_{Comp} is a set of duples whose elements are duples. More specifically, O_{Comp} is a set of duples whose elements are duples of members of some P_i .

Now that we have a population P_{Comp} and an organization on it O_{Comp} , then we have a new composite system $S_{Comp} = (P_{Comp}, O_{Comp})$.

Suppose T is a system transform such that $T(S_A, S_B, \dots, S_N) = S_{Comp}$. Then T is called an *Integrate operation*, or alternatively, a *Compose operation*.

Integrate (or *Compose*), then, is a category of N -ary operation that takes as its arguments a set of systems. The result of this operation is a single system - the composite system. A composite system is also called a *nested system*.

One thing to notice is that this transform category is a *join* transform. That is, it takes as its domain elements two or more Organodynamic *system states* and maps the pair to a codomain system state.

The organodynamic transform version of Integrate is also a join transform. It will take as its inputs two or more Markov transition matrices for time $t-1$ and produce as its only output a new transition matrix at time t .

Focus of the Integrate Transform Development

The focus of the Integrate organodynamic transform category will be on calculating the underlying population of the row state space of the output composite transition matrix for time t .

But this underlying population is, by definition, the set of *organizations* of the two (or more) input transition matrices for time $t-1$. (Without loss of generality, we shall assume exactly two input composite systems for this description.) These underlying organizations are evident, because they are specified as a part of these input component systems. The underlying population of the row state space is thus determined.

Any specific organodynamic transform of the Integrate category must further proceed to determine the row state space for the transition matrix of the output (composite) system. Of course the row state space must be some populated by some sets of duples of this population of duples.

Specific Integrate-type transforms may impose rules, algorithms or restriction on what this state space is or can be. In addition, the specific application that is being modeled may also impose conditions on these outcomes. This fact is also

determined by the specific Integrate-type transform. (These shall be referred to as “the conditions”.)

But, this determination is the responsibility of specific Integrate transforms. We do not have to make a determination as to what these state spaces of duples of duples look like for the Integrate category – except to say that it is in fact a state space whose entities are these duples of duples.

In addition, the column state space must also be ascertained. The conditions will impose the column state space. If the node is deemed homogenous by the conditions, then it will be the same as the row state space.

Also, the conditional probabilities in the body of the matrix must also be ascertained. The conditions will impose the conditional probabilities.

Two more things are significant about the Integrate organodynamic transform. The first is that it is non-destructive. By this we mean that the input systems remain as system at time t .

In addition, they continue as distinct organodynamic processes. That is, each continues to be modeled as piecewise-homogeneous Markov chains as “edges” within the Organodynamic web.

Creational Relationships in Nested System Hierarchies

It is interesting to note that a composite system is created via the Integrate transform from its components – or actually from systems that become its components.

Thus, it is not an accurate analogy to refer to a composite system as a “parent” and to its components as its “children”. This is not a parent-child relationship, because children do not create their parents!

It is more accurate to understand that a composite system is preceded by its components and is created by them through an Integrate transform.

In other words, nested system hierarchies are built from the bottom up. We certainly find this to be true of biological systems. Prebiotic systems emerged into unicellular systems, which emerged into multicellular systems – all via integration of their components.

As a result of these findings, the *Integrate* transform is the essential engine of emergence in Organodynamics.

The Impact of the Integrate Transform on the Organodynamic Web Structure

Being an organodynamic transform, *Integrate* is represented as a node in an Organodynamic web. As a node, an *Integrate* transform has “edges” coming into it as well as “edges” coming out from it. Let's discuss this picture in more detail to understand what these “edges” represent.

In the first place, notice that it is the *Integrate transform* node that creates a nested system. Thus, anytime that there is an Integrate node, then one of the “edges” that follow from it at time t must represent the composite system that it creates.

And of course the Integrate transform uses two or more *component systems* as its inputs in order to create this composite system as an output. This means that there should be a set of “edges”, each of which represents a component system chain that comes into the Integrate node from time $t-1$.

Thus far our picture has at least two components system edges coming into the Integrate node from time $t-1$ and one composite system edge going out of the node at time t .

But, recall that Integrate is non-destructive. This means that its input systems continue to exist after the transform node – along with the Markov chains that continue their “lives”.

Therefore, an Integrate node will have two or more (“ n ”) component system edges coming into it from time $t-1$; and it will also have the same n components system edges coming out of it at time t . And, it will also have exactly one system edge coming out of it at time t to represent the composite system.

This description represents a *view* of the creation of a composite system by the Integrate transform. And it is a representation that shows how and where a composite system, along with its components systems, fits within an Organodynamic web structure. We shall refer to this view of a composite system as the *process view*.

This process view highlights that 1) a nested system is created by an Integrate node, and 2) that an Integrate node is simultaneously both a split and a join transform with n systems coming into the node from time $t-1$, and $n+1$ system coming out of the node at time t . The “extra” system that comes out of the node at time t is the composite system. While the other systems that exit the node at times t are the same systems (the components of the composite system) that came into the Integrate node from time $t-1$.

The process view utilizes multiple “single-layer” component systems representations, each living concurrently with the others. Each of these single-layer system representations is referred to as a *simplex representation*. In order to represent the entire nesting hierarchy, the process view must use one each of these simplex representations for each for the subsumed systems of a nested system.

Later, we shall introduce a mechanism for representing multiple layers of a nested system hierarchy within a single structure. This will be called the *composite system representation* of a nested system. In fact, the all of the layers and all of the subsumed systems of a nested system hierarchy can be represented within such a composite system representation. The essential task of this chapter is to develop such a structure.

Extending Composite Systems to Deeper Levels of Organization

This subsection on the Integrate transform has represented a composite system as having two levels of organization: the composite system level as well as the component system level.

In the next chapter on *nested systems*, we shall show how these components systems can be treated as composite systems in their own right – and have their own component systems. Thus, a composite system can have component systems that are also composite systems, which have their own component systems. Thus, we can have a “top level” composite system that has three levels of organization.

Obviously, this concept can be extended to include any finite number of levels of organization. We call such a system a *nested system*. We take up this discussion in the next chapter.

Appendix 3: The Organodynamics Modeling Structures

Appendix 3 of this Part II provides a summary of the Organodynamics structures that comprise the modeling elements of the framework.

The system modeler initially builds small model elements, as defined by some of these constructs, and then subsequently combines them into larger compound constructs, also defined by the framework. Eventually, complex models are cumulatively built up through a number of incremental iterations.

This appendix summarizes all of the construct types provided by the framework for this purpose. The presentation of these structures is in order of complexity. Generally, each construct type is comprised of many instances of the construct types that appear prior to it in the table.

<u>Structure Name</u>	<u>Structure Function</u>
System	An ordered pair (P; O), where P is a set of elements, called <i>components</i> , and O is a set of ordered pairs, called <i>duples</i> , of those components. O is called the <i>organization</i> of the simplex system. A system can be either a simplex system or a composite system.
Simplex system	A system whose components are not other systems.
Composite system	A system whose components are other systems, called component systems of the composite system. The nomenclature used to represent these component systems specifies that the component systems are represented by their organizations only. (Their populations are implied.) Thus, the nomenclature for composite systems represents its components by their organizations.
Organodynamic Probability Distribution (of a System) Also called an "Organodynamic Distribution"	A probability distribution whose sample space is comprised of all possible system organizations that can be created from the population of an underlying system.
Organodynamic Process	A finite state, finite step stochastic process, each of whose time steps are

<u>Structure Name</u>	<u>Structure Function</u>
	populated by an organodynamic distribution.
Time step	This term is often used to refer to the state of a time step within an organodynamic process. Such a state is actually an organodynamic distribution. Thus, this term sometimes refers to a specific organodynamic distribution.
Segment	An organodynamic process all of whose time steps have the same organodynamic distribution. In other words, a segment is a homogeneous organodynamic process.
Organodynamic transform	<p>A function whose domain is a set of organodynamic processes whose termination time steps are synchronized to occur at the same time; and whose codomain is a set of organodynamic processes whose initiation time steps are synchronized to occur at the same time. Moreover, the termination times mentioned are two time steps prior to the initiation times mentioned.</p> <p>The intention of an organodynamic transform is to provide a “meeting event” where multiple concurrent processes can join into a single point and where a single system, and that single system can split into multiple concurrent processes.</p> <p>These “split” and “join” transforms are the basis for the formation of “nodes” – and thus enable the formation of networks from segments and nodes.</p> <p>However, a special case of a transform is one that has a single input and a single output. Such a transform can be used to transition from a single time step to the next – even within a segment.</p>

<u>Structure Name</u>	<u>Structure Function</u>
Node	<p>A join transform or a split transform. A node provides the mechanism for multiple concurrent segments (or edges) to come together at a “join” time step, or to split apart in a “split” time step. Of course a node can involve simultaneously both a split and a join.</p> <p>In any event, the existence of nodes in conjunction with segments (and edges) enable network structure to be formed.</p>
Edge	<p>A set of contiguous time steps formed by one or more segments. Whereas segments are internally homogeneous finite step stochastic processes, edges are internally nonhomogeneous finite step stochastic processes.</p>
Organodynamic Graph	<p>A simple network whose elements are either segments, edges or nodes, as defined above.</p>
Simplex Organodynamic Web	<p>An Organodynamic Graph with loops. These loops are implemented by adding two elements to its time steps. (These elements can reside within segments, edges or nodes). The first of these two elements is a reference to other times steps. This implements the looping mechanism. The second is a specification for which actual time the time step represents at any instant. This enables the time step identity to be a variable so that a time step specification can be reused with a routine.</p>
Composite Organodynamic Web	<p>Recall that – starting with the <i>segment</i>, all previous constructs defined in this table are constituted by one or more time steps. Also recall that each of these time steps consists of an organodynamic distribution. Moreover, each of these distributions has an underlying system – as defined</p>

<u>Structure Name</u>	<u>Structure Function</u>
	<p>above.</p> <p>Until now, none of these constructs were allowed to have underlying systems that are composite systems. The reason for this restriction is to reduce the complexity of initially developing these constructs.</p> <p>At this point, however, we are going to introduce composite systems into the mix. This means that we are going to take any Simplex Organodynamic Webs and convert them so that they now contain composite systems where appropriate.</p> <p>This conversion generally means converting any <i>process view</i> nested systems within Simplex Organodynamic Webs to composite systems.</p>
Joint Composite Organodynamic Web	<p>Both of the above organodynamic web structures exhibited network semantics. This articulation presented an intuitive analogical view that was friendly and useful to system modelers.</p> <p>Unfortunately, it misrepresented a random aspect of organic systems by masquerading it as though it were deterministic.</p> <p>This aspect is the occurrence of split and/or joint transforms. That is, of nodes.</p> <p>Although not obvious, the previous construct assumed that the time steps in which nodes occur are pre-determined. However, in organic systems, this is not generally the case.</p> <p>The solution is to drop the network semantics and to treat all possible variables as orthogonal dimensions. This approach allows each of these</p>

<u>Structure Name</u>	<u>Structure Function</u>
	<p>variables to behave with any varying degree of stochastic dependence or independence that may manifest for a specific organodynamic web.</p> <p>These multiple dimensions then join together to form a joint sample space, wherein each cell of this multidimensional space is treated as a sample point, and is assigned a probability.</p> <p>The result is a joint probability space. The sample space itself is very complex. However, there is only one sample space involved.</p> <p>Moreover, the degrees of dependence/independence among any collection of these dimensional variables are then completely determined by the probabilities of this joint distribution.</p> <p>Unfortunately, such a treatment loses its intuitiveness as compared with the previous Organodynamic web structures.</p> <p>However, this approach finally achieves the intellectual fidelity that is demanded by the modeling domain.</p> <p>Moreover, this approach of using a joint probability distribution to model this kind of stochastic flexibility is a traditional approach in complex stochastic systems, such as statistical mechanics.</p>

Appendix 4: The Organodynamics Modeling Methodology

Appendix 4 provides a summary of the modeling procedure proposed by the Organodynamics modeling methodology.

Because of the complexity of developing organodynamic models, a modeling methodology has been put forth in this text. This methodology identifies six milestones, each of which is associated with one of the constructs of the Organodynamics framework that were described in Appendix 3.

Each of these six milestones is referred to as an *approximation*. The idea is that each adds additional fidelity to the accumulating system model – but at the expense of also adding additional complexity. Thus, the first milestone is the “first approximation” of the object system, the second milestone is the “second approximation” and so on until a final approximation is built.

These six “milestone” structures that constitute these “approximations” are:

1. Segments
2. Edges
3. Organodynamic Graphs
4. Simplex Organodynamic Webs
5. Composite Organodynamic Webs
6. Joint Composite Organodynamic Webs

The procedure specifies a specific order in which these six structures are developed.

Moreover, these six structures are cumulative in nature. Specifically, each of them is comprised of multiple instances of the previous construct type. Thus, the development of any one instance of one of these construct types consists of developing multiple instances of the previous construct type, and then integrating these multiple instances into this single instance of the construct type in question.

This procedure is iterated recursively until either 1) all of the six milestones are constructed, or 2) the last desired milestone is constructed.

In addition, each of these approximation is subject to three “sub-models” or “model versions”. These three model versions take the modeler through a three-step progression of ever-decreasing uncertainty regarding the model. Thus, this progress is referred to as the *uncertainty model gradient*.

The first model version of this gradient is a Markov chain model of the approximation. Such a model embodies the most amount of uncertainty of the three versions, since a Markov model is a stochastic process that uses an entire set of conditional probability distributions for each time step. This situation is too complicated to provide a single entropy measure. However, it is an important type of stochastic model in science, engineering and mathematics.

The second model version is called the *conditional stochastic model*, because it is a stochastic process whose time steps are characterized by a single probability distribution. It is usually arrived at by the application of additional information to the first model version – the Markov chain model. Techniques are provided in the text for taking this additional information and applying matrix algebra to the Markov model in order to produce the conditional stochastic model. Since each time step of the conditional stochastic model has a single probability distribution, then entropy values can be calculated for each of them. Therefore, the conditional stochastic model is manageable enough to be characterized by a set of entropy values.

The third model of the uncertainty model gradient is the *realized deterministic model*. This model is obtained by observing the realization of the probability distributions of the *conditional stochastic model*. (There is also a matrix algebra explanation of these realizations.) The *realized deterministic model* is also characterized in stochastic terms by articulating each of its time steps as an organodynamic distribution with entropy of zero (0). This means that one of the sample points now has a probability of one (1), while the remaining ones have probabilities of zero (0).

The Organodynamic modeling procedure specifies that this three-step *uncertainty model gradient* be taken through each of the six approximation milestones in order. Thus, in all, the methodology specifies $6 \times 3 = 18$ model manifestations.