BASIC STRUCTURAL MODELING PROJECT

Joseph J. Simpson

Mary J. Simpson

01-28-2014

PROBLEM DEFINITION REPORT

Version 1.01

Table of Contents

Preface

In the preface to *Structuring Complex Systems, Battelle Monograph Number 4,* (1974) on pages i and ii, John N. Warfield stated the following:

> *"In developing this monograph, it was useful to think of structural models of two generic types. The first type, the basic structural models, are those whose theory has evolved out of mathematics. They are the graphs and digraphs which carry no empirical or substantive information. Much is known about their properties. Methods exist for performing operations upon them that permit extensive manipulation and structural insight. The second type, the interpretive structural models, are those developed to help organize and understand empirical, substantive knowledge about complex systems or issues. Intent structures, DELTA charts, and decision trees, illustrated in earlier monographs, are examples of interpretive structural models. Other examples include interaction graphs, PERT diagrams, signal-flow graphs, organization charts, relevance trees, state diagrams, and preference charts.*
>
> *If the full knowledge of basic structural models could be brought to bear upon the development of interpretive structural models, a significant advance could be made in the rational analysis and synthesis of complex systems. Yet, it seems impractical to expect that those who are engaged in day-to-day interaction with complexity in human affairs would take the time to learn to apply such abstract concepts as mathematical logic, matrix theory, and the theory of graphs in their work. It also seems unlikely that mathematicians would take the time to become highly knowledgeable of complex real-world systems and issues. The dilemma of how to wed substantive issues and knowledge of complex systems to the mathematics seems significant. But even if people had all the mathematics and understood the complex system or issue, still another problem would be present. That is the extreme tyranny of working systematically to establish relations among many elements in the form of an interpretive structural model, and the long time period required to do this by manual methods.*
>
> *One approach shows promise of a way out of the mentioned difficulties. This approach is to introduce the digital computer to aid in problem definition. If the necessary mathematical knowledge as well as the logistical tyranny can be transferred to the computer, leaving to the developer of the interpretive structural model only the minimum, but critical, core of effort – providing the substantive knowledge of the system or issue – then the developer would not need to learn the associated mathematics, nor would he have to absorb the tyranny associated with the extensive manipulation of ideas on paper that would otherwise be required. The computer could be a major factor in compressing the time scale for development of an interpretive structural model.*
>
> *This monograph presents a method whereby the computer can carry out the necessary operations for those interpretive structural models that can be put in correspondence with digraphs. Since the monograph is largely limited to such models, it does not encompass all possible kinds of interpretive structural models."*

It is clear that structural modeling has two distinct components: basic structural modeling and interpretive structural modeling. To create an effective, computer-based tool to support various kinds of interpretive structural modeling applications, the relevant elements associated with basic structural modeling must be placed "in proper correspondence" with the interpretive structural model of interest; that is, the specific mathematical relations and constructs must be paired with an appropriate organizing relationship that defines the interpretive structural model being sought. The task of identifying, evaluating and selecting the correct, basic structural modeling elements for a given interpretive structural model is a nontrivial task. The authors developed a third component of structural modeling that addresses the selection, organization, and adaptation tasks associated with transforming basic structural modeling components into interpretive structural models. This third component of structural modeling is called 'structural integration modeling'. The primary artifact that is developed using structural integration modeling is called an abstract relation type (ART).

Introduction

The basic structural modeling project (BSMP) Problem Definition Report provides a definition of the issues, concepts, tools and procedures used when structural modeling techniques developed by John N. Warfield are engaged to develop solutions to complex problems. The problem definition report will organize problems and issues into three general categories. These categories are:

1. Model type selection;
2. Contextual structural relationship representation; and
3. Mathematical tools, techniques and solution approaches.

A well-defined and communicated problem definition context will facilitate presentation and understanding of the structural modeling material developed by Warfield. For these structural modeling techniques, a key element is the identification of the contextual structural relationship. The modeling of an unknown system structure is organized around a contextual relationship. Warfield presented six major categories of interpretive relationships found in English (Science of Generic Design, p60). These six categories are definitive, comparative, influence, temporal, spatial and mathematical. Examples of these categories are:

1. *Definitive* relationship – is necessary for, is implied by, includes, is assigned to
2. *Comparative* relationship – is more important than, is more useful than, is preferred to
3. *Influence* relationship – supports, affects, confirms, causes, enhances, strengthens
4. *Temporal* relationship – precedes, follows, coincides with, concurrent with,
5. *Spatial* relationship – is north of, is south of, is east of, is west of, is above, is below
6. *Mathematical* relationship – is equal to, is less than, is greater than, is a function of

For the examples presented in this problem definition report, spatial relationships will be used.

A binary relation defines how one thing is related to another thing. The natural language relationships listed in the above six categories are carefully transformed to mathematical relations using the techniques of basic structural modeling. Key aspects that guide the system scientist in this transformation process are the properties associated with binary relations. The properties of relations are important in mathematics and logic. There are specific properties of relations that are of special interest to logicians. These properties include: transitivity, intransitivity, nontransitivity, reflexivity, nonreflexivity, irreflexivity, total reflexivity, symmetry and asymmetry.

In *Scientific Method: Optimizing Applied Research Decisions* (1962), Russell L. Ackoff presented a list of properties of relations that are significant from a measurement point of view. These properties are: reflexive, irreflexive, nonreflexive, symmetric, asymmetric, nonsymmetric, antisymmetric, transitive, intransitive, nontransitive, quasi-transitive, connected, and nonconnected.

In *Introduction to Mathematical Philosophy* (1919), Bertrand Russell outlined the properties necessary to produce an ordering relation. An ordering relation will arrange items in a series. Russell presented three properties of an ordering relation: asymmetry, transitivity, and connectivity. Russell's description of the asymmetric property clearly states that the property holds between two different terms in the class that is to be ordered. The description of the transitive property also clearly indicates that three distinct terms of the class to be ordered are required to exhibit the property. The description of the connected property is given as a property of a relation that holds between any two terms in the class to be ordered. A relation is serial when it is asymmetrical, transitive, and connected.

> "Although a transitive asymmetrical connected relation always exists wherever there is a series, it is not always the relation which would most naturally be regarded as generating the series."[1]

A serial relation, and a series, are identical.

Russell continues his discussion of the properties of relations by addressing the relative utility and importance of these properties. The asymmetric property is assigned the highest utility. The construction of a relation with a symmetric property may be based on two distinct relations, each of which has asymmetric properties. Relations with a transitive

[1] Russell, Bertrand, *Introduction to Mathematical Philosophy*, p31.

property, and relations with a connected property, may be constructed from relations that did not originally have these properties.

A firm understanding of the impact of these types of relations in the interpretation of natural language speech is a very important tool used to enhance the precision of technical communication. Also, the proper application of these foundational concepts will facilitate the encoding of natural language speech into mathematical and graphical models that are necessary to support Warfield's model exchange strategy. The model exchange strategy provides a mechanism to precisely communicate concepts and information in three different formats: prose, graphics and mathematics. Some informal, contextual definitions and example of these properties of relations are presented next.

The first property of a relation to be considered is '**transitivity**.' This property of a binary relation is informally defined as:

> If object one has a transitive relation with object two
> and object two has the same transitive relation with object three,
> then object one has this same transitive relation with object three.

The natural language relationship 'is-north-of' is transitive. For example:

> If city one is north of city two
> and city two is north of city three,
> then city one is north of city three.

The second property of a relation to be considered is '**intransitivity**.' This property of a binary relation is informally defined as:

> If object one has a intransitive relation with object two
> and object two has the same intransitive relation with object three,
> then object one does not have this same transitive relation with object three.

The natural language relationship 'in-the-neighborhood of' is intransitive. For example:

> If city one is in-the-neighborhood of city two,
> and city two is in-the-neighborhood of city three,
> then city one is not necessarily in-the-neighborhood of city three.

The third property of a relation to be considered is '**nontransitivity**.' This property of a binary relation is informally defined as

> If the relation is neither transitive nor intransitive, then the relation is nontransitive.

The natural language relationship "is-two-miles-from" is nontransitive.

Problem Definition Context

The term structural model is given an expansive definition by Warfield, and models of all types many be associated, aligned and integrated using the framework established by structural models. Prose models, graphical models and mathematical models are all representations supported by the structural modeling techniques. Structural modeling has two basic components: basic structural models and interpretive structural models. There have been a number of approaches developed by a range of authors that are identified as ISM approaches. However, each of these ISM approaches, developed by authors other than Warfield, do not detail the specific role of the structural relationship and the basic structural modeling techniques used to support the ISM approach.

Model Type Selection

ISM techniques were developed by Warfield, Sage, Steward and Hitchins, among others. Each of these ISM approaches presents a significantly different approach to the basic structural modeling techniques and methods. Sage makes no distinction between the basic structural modeling approaches, and the ISM applications. Steward creates an ISM approach based on a natural language relationship that is not consistently transformed into a mathematical relation. Hitchins uses the Automated N-Squared Chart approach as the foundation for his ISM approach. Each of these approaches provides interesting insights into the use of prose, graphics and mathematics for large-scale problem solving. A key insight is the fact that the system structural relationship, and the matched mathematical relation, must have the same properties and attributes to properly apply the selected ISM technique.

Any specific ISM approach and/or technique may be supported with a range of well-defined model types and operations. These mathematical model types are used in combination with the system's natural-language structuring relationship to create an effective representation of the system of interest. A significant portion of existing ISM mathematical models are based on a square matrix model, that is similar to an adjacency matrix. Real-word empirical data is needed to properly form the matrix model. This empirical data is in addition to, and may be different than, the natural-language contextual structural relationship that is used in developing the primary structuring mathematical relation. For example, the real-world, empirical, structural relationship used in the main examples in this paper is 'is-north-of.' However, additional empirical data is needed to properly construct and evaluate the matrix models. This additional empirical information relates to the number of objects (cities) that can occupy each level (latitude) of the model. In the model example provided in this paper, only one object (city) can occupy a given level (latitude). However, it may well be the case that more than one object (city) exists at a given latitude. If the real world situation allows more than one city, then the matrix model must also allow more than one city at each level. These two matrix models (one city at each level: more than one city at each level) create very different approaches to a problem solution.

Another key consideration in the evaluation of system matrix model approaches is the semantics associated with the structuring relationship in any given model. A standard N-Squared Chart (NSC) has flow directions associated with the upper and lower triangular sections of the matrix**.** Further, a standard N-Squared Chart focuses on the interface between two objects represented in the chart. These model constraints create a situation wherein all but one (1) row and one (1) column in a complete NSC must have an interface mark. A Design Structure Matrix (DSM), on the other hand, provides for the representation of a set of objects that have no relationship between or among the objects. As a result, it is possible to have a complete DSM matrix without any relationship marks – that is, a complete matrix of empty cells. Other matrix model types, developed by Warfield, have a range of structural marking methods depending on the real-world relationship that the matrix represents. It is clear that a matrix model must be selected to support and reflect the primary real-world structuring relationships as well as other controlling system structuring considerations.

Contextual Structural Relationship Representation

The selection of the structural relationship is a primary step in structural modeling. The alignment between the attributes of the natural-language relationship and the attributes of the selected mathematical relation are fundamental to the effective implementation of a structural modeling process. An established, well-understood set of natural-language relationship attributes provides the basis upon which a coherent structural modeling process may be developed. The bridge between the prose, natural-language relationship and the mathematical relation requires a deep understanding of the real-world problem space, and the inherent capabilities of the mathematical process that are applied in the exploration of the problem space.

The individual or group that is constructing a specific ISM approach must be aware of the three basic forms of ISM models. These three (3) forms are prose, graphics and mathematics. The model exchange isomorphism (MEI), a concept developed by Warfield, transforms a structural model in a prose form to a graphic or mathematical form without loss of information. The MEI is an important part of the basic structural modeling techniques that generate views of a given problem and/or solution in alternative formats, allowing individuals to interact with the presented information in a form they understand.

Mathematical Tools, Techniques and Solution Approaches

Properly applied mathematical tools, aligned with natural language speech and understanding, provide a mechanism to carefully analyze problems and systems of problems. The mathematics of Boolean algebra, binary relations, directed graphs, and matrices are the primary tools used in basic structural modeling. The selection of the structural relationship attributes has a strong impact on the specific types of mathematics selected. A relationship that has asymmetric properties will be addressed in one manner, while a relationship that has symmetrical properties will be addressed in a different manner. In cases where there are a matched set of asymmetric properties that can be combined into a well-understood, natural-language, symmetrical relationship presents a possibility for the use of a combination of techniques.

This report will focus on basic structural model components and processes as described in "Societal Systems: Planning, Policy, and Complexity", John N. Warfield, 1976.

Structural Modeling

Structural modeling provides a conceptual framework that supports the evaluation and analysis of a wide range of system types. If the system of interest is unstructured during the initial phases of evaluation, then structural modeling techniques may be applied to create a more structured system for evaluation. Interpretive structural models are used to organize, structure and support the understanding of empirical, substantive knowledge about vague, ill-defined and/or complex systems, issues and situations. PERT charts, project network diagrams, signal-flow graphs, relevance trees, intent structures, DELTA charts, decision trees, state diagrams, and preference charts are all examples of interpretive structural models. Basic structural models support the activity of interpretive structural modeling, and are the topic of this report.

Basic Structural Modeling Elements

The basic structural modeling elements that will be addressed in this report are:

- Boolean algebra
- Mathematical sets
- Binary relations
- Binary matrices
- Binary matrix models
- Directed graphs (Digraphs)
- Directed graph maps
- Directed graph models
- System structure
- System complexity
- Transitive embedding
- Graph cycles

Each of these components will be introduced, discussed and presented in a manner that highlights common application areas as well as any areas that may have a special or non-standard meaning. The first component to be addressed is Boolean algebra.

Boolean Algebra

Boolean algebra operates on two distinct constant values: zero (0) and one (1). There are a number of operators associated with these values in Boolean algebra. These operators are:

- complementation ' (monadic operation)
- addition + (binary operation)
- multiplication · (binary operation)

These operations conform to the following laws:

- commutative laws: $a + b = b + a$, $a \cdot b = b \cdot a$
- distributive laws: $a \cdot (b + c) = (a \cdot b) + (a \cdot c)$, $a + (b \cdot c) = (a + b) \cdot (a + c)$
- identity laws: $a + 0 = a$, $a \cdot 1 = a$
- complement laws: $a + {'a} = 1$, $a \cdot {'a} = 0$

Warfield augmented Boolean algebra by adding the concept of order to the two Boolean constants. The addition of order to Boolean algebra supports the addition of the following new Boolean operators:

- Less than operator: $0 < 1$
- Greater than operator: $1 > 0$
- Less than or equal to: $0 \leq 1$
- Greater than or equal to: $1 \leq 0$

- Boolean subtraction: 1 - 1

When this new concept of Boolean order is used with matrix operations, the following Boolean matrix operations are added:

- Matrix subtraction: subtract one matrix from another
- Matrix ordering: one matrix less than, greater than, or equal to, another Boolean matrix

Boolean recursion equations and Boolean inequalities are sets of Boolean equations that are used to analyze and evaluate systems. Solution techniques associated with these types of equation sets will be presented later in this document.

Mathematical Sets

A set is a well-defined collection of objects. Each object in a set is called a member or element of the set. Sets may be formed in two ways: **extension** and **intension**. The first way to create a set is to explicitly list all of the elements in the set, which is known as set formation by extension. The second way to create a set is to provide a rule, or set of rules, that describe the set members. This is called set formation by intension.

Given $X = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$
Set X is formed by **extension**.

Given $Y = \{10, 20, 30, 40, 50, 60, 70, 80, 90, 100\}$
Each indexed element of set Y is 10 times the corresponding indexed element of set X.
Set Y is formed by **intension**.

A set formed by intension is called well-defined if the rules and criteria for set formation provide the ability to determine if any given element is either included in the set or excluded from the set. If a set is not well-defined, then empirical procedures may be used to create a variable grade of membership that is assigned to each element in the set. The variable grade of membership is called the element weight. In any specific situation, an element-weight trigger value may be assigned to indicate when the element is considered part of the set. The element weights run from one (1), for full membership, to zero (0) for no membership. Fuzzy sets are sets that contain elements with unequal weights. The sets that will be considered in this report are mostly well-defined.

Well-defined sets may be indexed using an index set if the number of elements in the set is known. Sets may be interrelated in a number of ways, including**: subset, power set, proper subset, complement, union, and intersection**. Sets that are not interrelated are called disjoint, and they have no elements in common.

- Set Z is a **subset** of set W if every element in set Z is also a member of set W. Any set may be a subset of itself.
- The set of all subsets of Z is called the **power set** of Z.
- Given: sets Z and W,
  set Z is not equal to set W
  set Z is a subset of set W
  Then Z is a **proper subset** of W
- The **complement** of set X with respect to set Y, is the set of all elements in set Y that are not contained in set X.
- The **union** of set Z and set W consists of all elements that are members of set Z, or set W, or both set Z and set W.
- The **intersection** of set X and set Y consists of all elements that are members of both set X and set Y.

A vector set is ordered using the set indices as a structuring mechanism. When a set is ordered using the set indices, it is called a vector set, or just a vector. A Cartesian product (of sets A and B) consists of the set $A \cdot$ set B of ordered pairs (a, b), where a is an element of A and b is an element of B. The Cartesian product is also a set. Once a vector set has been defined, then the Cartesian product of any given vector set, with itself, may be defined. For example:

- $X = \{1, 2, 3\}$, set X is defined by extension.

- Cartesian product set $Y = X \cdot X$ = {(1, 1), (1, 2), (1, 3), (2, 1), (2, 2), (2, 3), (3, 1), (3, 2), (3, 3)}, set *Y* (the Cartesian product of Set *X* times Set *X*) is defined by intension.
- Set *W* = {(1, 1), (2, 2), (3, 3)}, is a subset of *Y*.

A partition (Π) on set *Z* is a collection of disjoint, nonempty subsets of *Z* whose set union is *Z*. For example, consider *Z* = {*A, B, C, D, E, F*}. Some possible partitions of set *Z* are:

- Π_1 = [A, B, C; D; E, F} (i.e., three partition blocks: A, B, C; D; and E, F)
- Π_2 = [A; B, C; D; E, F} (i.e., four partition blocks: A; B, C; D; and E, F)
- Π_3 = [A, B, C, D; E, F] (i.e., two partition blocks: A, B, C, D; and E, F)
- Π_4 = [A, B, C, D, E, F] (i.e., one partition block: A, B, C, D, E, F)
- Π_5 = [A, B; C, D; E, F] (i.e., three partition blocks: A, B; C, D; E, F)
- Π_6 = [A; B; C; D; E; F] (i.e., six partition blocks: A; B; C; D; E; F)
- Π_7 = [A, B; C, D; E, F] (i.e., three partition blocks – A, B; C, D; E, F)

The concepts of **equality, product**, **sum**, and **order** are included in partition algebra.

- If every block in one set partition on Z is also a block in another set partition on Z, then the set partitions are **equal**. As an example:
  Π_5 = [A, B; C, D; E, F]
  Π_7 = [A, B; C, D; E, F]
  Π_5 is **equal** to Π_7.

- The **product** of two partitions on a set is another partition on the set. The members, element 1 and element 2, of set Z, are in the same block of the product partition only if they are in the same block in both of the argument partitions. As examples:
  Π_1 = [A, B, C; D; E, F}
  Π_5 = [A, B; C, D; E, F]
  $\Pi_1\Pi_5$ = A, B; C; D; E, F
  Π_2 = [A; B, C; D; E, F}
  Π_6 = [A; B; C; D; E; F]
  $\Pi_2\Pi_6$ = A; B; C; D; E; F
  Π_4 = [A, B, C, D, E, F]
  Π_3 = [A, B, C, D; E, F]
  $\Pi_4\Pi_3$ = A, B, C, D; E, F

- The **sum** of two partitions on a set is another partition on the set. The members, element 1 and element 2, of set Z are in the same block of the sum partition only if there exists a sequence $x_1, x_2, \ldots, x_m$, such that element1 = x_1, element2 = x_m and all pairs of the form $x_1, x_2, x_3 \ldots, x_{m-1}$ are in a single block in either argument partition one or argument partition two. As an example, for:
  Π_1 = {a, b; c, e, g; d; f, h, j; i}
  Π_2 = {a, b, d; c, g; e, f, h, I; j}
  The first block of the sum is a, b, d. That block is composed of two blocks in Π_1 (a b, and d) and one block in Π_2 (a b d). The second block of the sum is c, e, f, g, h, i, and j. That block is composed of the other blocks from argument partitions Π_1 and Π_2 that are not in a single block.
  The sum of Π_1 and Π_2 is: $\Pi_1 + \Pi_2 = \Pi_3$ = {a, b, d; c, e, f, g, h, i, j}

- Partitions on a set may be **ordered** in value using the following definitions. A partition is less than or equal to another partition on the same set only if every block in the first partition is contained in some block of the second partition. Given two partitions Π_1 and Π_2, Π_1 is less than or equal to Π_2 only if the product of Π_1 and Π_2 is equal to Π_1 and the sum of Π_1 and Π_2 is equal to Π_2. The product of Π_1 and Π_2 is always less than or equal to the sum of Π_1 and Π_2.

If the partition of set A ($\Pi(A)$) consists of a single block, it contains all members of set A, and is called the **identity** partition (Π_I).

- The **product of identity partition** with any other partition Π is equal to Π.

The **zero** partition of a set (Π_Z) consists of a partition that has as many blocks as it has elements.

- The **sum of the zero partition**, Π_Z, and any other partition Π is equal to Π.

Binary Relations

A **binary relation** *R(A, B)* is a subset of the Cartesian product of vector set A and vector set B. A binary relation *R(A, A)* is a subset of the Cartesian product of vector set A with itself, or A X A. A binary relation *R(B, B)* is a subset of the Cartesian product of vector set B with itself, or B X B. For example:

Set $B_1 = \{1, 2, 3\}$
Set $B_2 = \{4, 5, 6\}$

Some binary relations on sets B_1 X B_2 are:

$R_1 (B_1, B_2) = \{(1, 4), (1, 6), (2, 4), (3, 6)\}$
$R_2 (B_1, B_2) = \{(1, 5), (2, 6), (3, 4)\}$
$R_3 (B_1, B_2) = \{(2, 5), (3, 5)\}$
$R_4 (B_1, B_2) = \{(3, 4)\}$

The **complement** (') of the binary relation, R_1, is composed of all of the elements that are not part of R_1.

The complement of R_1 = $'R_1 = \{(1, 5), (2, 5), (2, 6), (3, 4), (3, 5)\}$

The **transpose** (R^T) of R_1 is produced by exchanging the order in every element pair.

The transpose of $R_1 = R^T{}_1 = \{(4, 1), (6, 1), (4, 2), (6, 3)\}$

A **binary relation,** *R(W, W)* on set W may have the following properties (where w_1 and w_2 are elements of W):

reflexivity: w_1Rw_1
irreflexivity: $w_1'Rw_1$
symmetry: if w_1Rw_2, then w_2Rw_1
asymmetry: if w_1Rw_2, then $w_2'Rw_1$
antisymmetry: if w_1Rw_2 and w_2Rw_1, then $w_1 = w_2$

Note that the notation w_iRw_j means that (w_i, w_j)

If a binary relation, R, is reflexive and transitive, and R and the complement of R are antisymmetric, then R is called an **order**.

If the complement of R is not antisymmetric and all other conditions are met, then R is called a **partial order**.

If a binary relation is a partial order, and also identifies a greatest lower bound and a least upper bound, then this partial order is a **lattice**. For example : (page 218)

Let A consist of the binary vector set $\{a_i\} = \{(u_i, v_i, w_i)\}$, where:

$a_0 = (0, 0, 0)$
$a_1 = (0, 0, 1)$
$a_2 = (0, 1, 0)$
$a_3 = (0, 1, 1)$
$a_4 = (1, 0, 0)$
$a_5 = (1, 0, 1)$
$a_6 = (1, 1, 0)$
$a_7 = (1, 1, 1)$

Define binary relation, R, on set A X A using the following two conditions:

(1) a_iRa_j if and only if $u_i \leq u_j$, $v_i \leq v_j$, $w_i \leq w_j$, where, by definition $0 < 1$, $0 = 0$ and $1 = 1$

(2) a_i can participate in R if and only if a_i is a solution of the Boolean recursion equation set:

$$\begin{aligned} u &= u \\ v + u &= v \\ w + v + u &= w \end{aligned}$$

The only elements that can participate in R are the elements of subset $A_1 = \{a_0, a_1, a_3$ and $a_7\}$. Applying condition (1), it is seen that the binary relation defined by the two conditions is

$$R = \{(a_0, a_0), (a_0, a_1), (a_0, a_3), (a_0, a_7), (a_1, a_1), (a_1, a_3), (a_1, a_7), (a_3, a_3), (a_3, a_7), (a_7, a_7)\}$$

This relation is not a partial order for the set A, since it is not reflexive. However, if it is reinterpreted as a binary relation on A_1 X A_1, it is both a partial order and a lattice.

In this way, a set of Boolean recursive equations can constrain a binary relation to a subspace that is both a partial order, and a lattice. The theory of matrices and digraphs are developed next to provide an appropriate tool set for managing system structures that have a large number of elements.

The primary objective of encoding expert information, obtained in natural language, into formal mathematical constructs is strongly supported by the concepts of natural language relationships and their attributes. These natural language relationships are encoded into mathematical relations in a manner that supports the objective of creating a system structure model of an unknown, or poorly defined, system. As indicated, the example real-world relationships used in this document are spatial relationships.

For example, consider a situation in which there are 19 cities, and you need to know which city and/or cities are north of a specific city. This type of problem can be encoded into a 19 by 19 matrix that represents a binary relation 'is north of.' Using the Warfield approach of assigning the value of one (1) to true, zero (0) to false and zero (0) to unknown, a matrix containing 361 zeros would be generated. The first task is to determine what each zero (0) represents. Since we have no empirical information about the cities, no city names, no city locations – it would seem that all of the zeros would indicate an unknown state. However, this is not the case. From the structure of the matrix, each entry on the diagonal of the matrix will ask the question "is city x north of city x?" This question can be answered by reviewing the properties of the 'is north of' binary relation. From a real-world perspective, letting a city occupy space north of itself is nonsensical, and the answer should be that the city is not north of itself. From an analysis of the binary relation properties, each zero (0) on the diagonal can be correctly determined to be a false value.

The transition from real-world relationships to mathematical models presents a few conceptual hazards and traps. Design structure matrices (DSM) are one example of this type of conceptual hazard. Many DSM techniques use the natural language relationship 'precedes' as the system structural relationship. However, Steward – the originator of DSM, provides an analysis of the 'precedes' natural language relationship that assigns relational properties and mathematical operations to this relation. In this case Steward indicates: "If $x \leq x$ we say that x 'preceded' x. Note by definition that each x precedes itself." Most individuals would find the concept of an event preceding itself to be nonsensical. However, this is allowed in mathematical relational constructs of Stewards DSM from 1981, *Systems Analysis and Management: Structure, Strategy and Design.*

Another area of caution in the transition from natural language relationships to mathematical relations is associated with the transitive property. Most people assume that a transitive relation must be among at least three distinct, individual objects. As Warfield pointed out, the mathematical relation does not require that the objects be distinct; you could have a transitive relation on one object. In fact, that is one outcome of the manner in which Steward assigned the 'precedes' relation. Steward allows event x to precede event x which can precede event x and so on. While these arrangements will work out mathematically, these concepts become nonsense when they are translated to real-world events and systems.

The process of identifying, defining, and communicating the structure of a system is a very demanding task in which all available types of information must be used to reduce the uncertainty associated with the system of interest. The properties of the organizing system relations are a rich source of information, that seem to be ignored by most system scientists and engineers in their evaluation of an ill-defined and/or unknown system.

For example, using one of the binary spatial relations provided in the introductory text – ‘is-two-miles-from’ - city A can be two miles from city B, and city B can be two miles from city C. This binary relation, ‘is-two-miles-from,’ was assigned a nontransitive property. If this relation is nontransitive, then the spatial configuration of the cities has a very high degree of uncertainty. However, if the relation is transitive, then the spatial configuration of the cities is precisely known to be an equilateral triangle.

The choice of the correct relational property can greatly reduce uncertainty by increasing the information content of the communication. Abstract relation types were created and developed by the authors to provide a structured means of system description that focuses on the system-defining relation, and the relation characteristics and properties.

Binary Matrices

A matrix (*M*) is defined as the collection of four sets:
The first set is an **ordered vertical index** set.
The second set is an **ordered horizontal index** set.
The third set is the **Cartesian product of the first and second sets**.
The fourth set is the **entry set of the matrix**.

A binary matrix is a matrix that has the entry set restricted to the symbols 0 and 1. The paper, published in 1973, titled "Binary Matrices in System Modeling", by John Warfield details the use of binary matrices and digraphs in system modeling. This paper **introduces the concept of model-exchange isomorphisms**, wherein a binary system model may be replaced with a directed graph (digraph) without any loss of structural information.

As Warfield points out in the 1973 paper:
"Binary matrices are useful because they represent the presence or absence of a specific kind of relation between pairs of elements in a system, thereby opening up opportunities for structuring the system."

Binary Matrix Models

A binary matrix model is composed of four elements: 1) a binary matrix, 2) contextual elements associated in a one-to-one manner with the vertical index set, 3) contextual elements associated in a one-to-one manner with the horizontal index set, and 4) a contextual relation R called the model relation. A reachability matrix is a square, transitive, reflexive, binary matrix which serves as a model matrix for a matrix model organized around the "is antecedent to" binary model relation.

The binary matrix is constructed from **the contextual binary relation that forms the system**. If the relation holds between a pair of elements, then a one (1), representing true, is entered in the appropriate matrix cell. If the relation does not hold between a pair of elements, then a zero (0), representing false, is entered in the appropriate matrix cell. While this approach may appear to be straight forward, in practice there can be a number of issues.

Some of the application issues may be traced back to the properties associated with the binary model relation. If we refer to Warfield's natural language relationships and his six major categories provided at the beginning of the paper, the task then becomes one of creating the proper mathematical relation to support the structuring of the system of interest. For example, the temporal natural language relationships are given as: ‘precedes,’ ‘follows,’ ‘coincides with,’ and ‘concurrent with.’ The ‘precedes’ natural language relationship may be translated into a mathematical relation that has asymmetric, transitive and connected properties. The ‘precedes’ natural language relationship may be paired with the ‘succeeds’ natural language relationship to create a balanced set of asymmetric relations. The same can be done with the ‘follows’ natural language relationship and the ‘leads’ natural language relationship. These balanced sets of asymmetric relations are not immediately identifiable with a single symmetric relation. The natural language relationship, ‘coincides with,’ may be transformed into a mathematical relation that is symmetric, transitive and connected. The natural language relationship, ‘concurrent with,’ may be treated in the same manner. *The **relations that have asymmetric relation properties will support structuring a sequential type of system**, while the relations that have symmetric relations will be of much less value in structuring sequential systems.*

One example of the application of basic structural modeling techniques that highlights some of these issues is presented next.

> In Warfield's 1973 paper, John presented clear and detailed connections between systems modeling and mathematical structural modeling, and established the foundations for his systems science work. This foundational work was applied in an interesting manner to the development of DSM methods for managing concurrent engineering tasks. In 1991, Eppinger referenced Warfield's 1973 "Binary Matrices in System Modeling" paper. As shown in Figure 1, Eppinger presented a graphic with three panels, and referred to these graphic representations as directed graphs, or digraphs. The graphics forms presented by Eppinger had little similarity to the referenced work by Warfield, or to standard directed graphs.

Directed Graphs (Digraphs) and Digraph Maps

A digraph consists of two sets, and a partition on these sets. The organizing, binary relation, and its complement, are mapped on the directed graph vertex set. A binary relation between two members of the vertex set, create an edge. A digraph map is a graphical representation of a digraph with each vertex represented as a point, and each edge represented with a directed line.

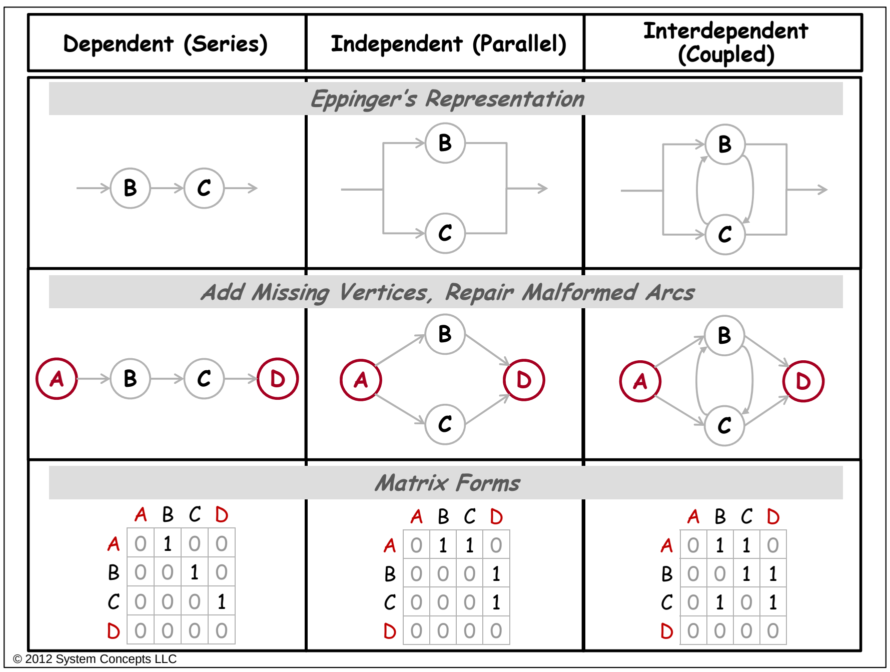

Figure 1. Digraphs

Further analysis of Eppinger's graphic indicates that the figures do not meet the fundamental definition of a directed graph. A definition cited from the Handbook of Discrete and Combinatorial Mathematics is:

> "A directed graph (digraph) consists of:
> - a set *V*, whose elements are called vertices
> - a set *E*, whose elements are called directed edges or arcs, and
> - an incidence function that assigns to each edge a tail and a head."

Two directed graph vertices (elements) are missing from the presented graphic representation. The figure shows these two vertices added, and the applicable matrix forms that correspond to those connections. The addition of the two missing elements, and the proper forming of the arcs, starts to reduce the ambiguity associated with the graphs presented by Eppinger. The analysis of the matrix forms in this figure indicates that all three system configurations are irreflexive.

Directed Graph Models

A directed graph model is composed of three elements: 1) a digraph map, 2) a set of elements associated with the vertices of the digraph, and 3) the binary relation associated with the digraph map. Digraph maps, models, and binary matrices are all tightly bound around the system structuring relation.

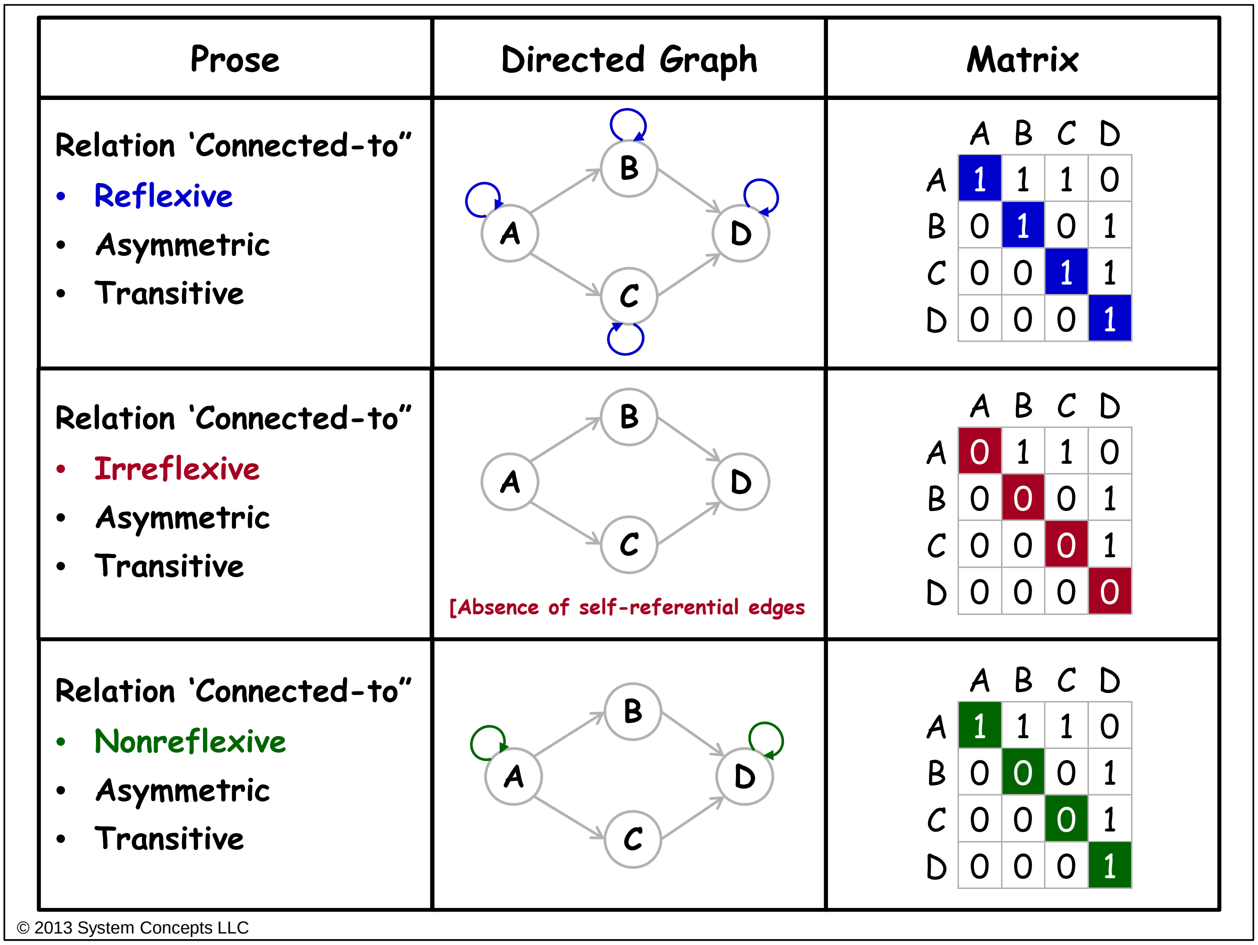

Figure 2. Directed Graph Model Exchange Isomorphisms.

Further analysis of the three graphic node configurations presented by Eppinger was performed to determine the common relationship among the graphic elements.

A detailed review of the first panel, representing a series relationship between the elements of the graph, produces two primary candidate organizing, binary relationships.

- The first candidate relationship between the elements is "connected-to." One element is connected to the other element.
- The second candidate relationship is "precedes-succeeds." The first element precedes the second element and the second element succeeds the first element.

A detailed review of the second panel, representing two independent elements with no interaction between the elements, produces no candidate organizing, binary relationship.

A detailed review of the third panel, representing two interdependent elements with multiple interactions between the elements, produces one candidate organizing, binary relationship. This candidate relationship is 'is-connected-to.'

Warfield does present a detailed process for developing a directed graph in the work referenced by Eppinger. The basis for this process is the identification of a binary relation between the elements represented by the directed graph. It is problematic that no specific organizing binary relation was identified in Eppinger's paper.

Warfield's process for developing a binary matrix sheds more light on the malformed directed graphs presented by Eppinger. Eppinger proposed that all three types of directed graph configurations are allowable into any type of product development DSM analysis. A key issue is the lack of any common, organizing, binary relationship among these three types of node configurations. Once the missing initial and final directed graph elements are added, there appears to be an approach that may create a unified binary relationship that will support the construction and use of binary matrices in DSM system modeling. The essence of this approach is the compression of the disconnected system elements into a single, connected, system element. After the compression is complete, the 'is-connected-to' binary relationship applies to all the types of system element configurations.

When the 'is-connected-to' natural language binary relationship is evaluated for transformation to a mathematical relation, a few domain-specific issues must be addressed. The first issue is associated with the fact that two fundamental types of relations are allowed in a DSM matrix at the same time. The first relation type is a series relation. The second type is a concurrent relation. As noted earlier in the note about Russell's work, these types of mathematical relations (a series relation and a concurrent relation) are mutually exclusive. When Steward outlined the DSM approach he indicated that the 'precedes' relation had a symmetric property that allowed an event to precede itself. This is a very non-standard application of the natural language 'precedes' relationship, and generates much confusion about the DSM approach. Given the work of Russell and Warfield, there appears to be no mathematical relation that supports both a series system structure and a concurrent system structure.

It appears that there may be an approach that uses a balanced set of asymmetrical relations that can be directly mapped to, and transformed to, a symmetrical relation. However the approach presented by Warfield, that used a single binary relation, does not appear to have a direct application to the classical DSM configuration. An approach that uses a series of transforms, and/or multiple concurrent matrix formulations, may be able to effectively address the issues surrounding the current DSM methods and processes.

As an example of one possible solution approach, consider the process of ordering fifteen (15) tasks. In this first simple example, all of the information about the system tasks is known before the ordering process is applied. Table 1 presents the information about each task's predecessor and successor.

Table 1. Natural Language Asymmetrical Relationship Ordering

Task	Predecessor	Successor
a	none	b, c, d, e, f, g, h, i, j, k, l, m, n, o
b	a	c, d, e, f, g, h, i, j, k, l, m, n, o
c	a, b	g, h, i, j, k, l, m, n, o
d	a, b	g, h, i, j, k, l, m, n, o
e	a, b	g, h, i, j, k, l, m, n, o
f	a, b	g, h, i, j, k, l, m, n, o
g	a, b, c, d, e, f	h, i, j, k, l, m, n, o
h	a, b, c ,d, e, f, g	i, j, k, l, m, n, o
i	a, b, c, d, e, f, g, h	m, n, o
j	a, b, c, d, e, f, g, h	m, n, o
k	a, b, c, d, e, f, g, h	m, n, o
l	a, b, c, d, e, f, g, h	m, n, o
m	a, b, c, d, e, f, g, h, i, j, k, l	n, o
n	a, b, c, d, e, f, g, h, i, j, k, l, m	o
o	a, b, c, d, e, f, g, h, i, j, k, l, m, n	none

Two asymmetric natural language ordering relationships, 'precedes' and 'succeeds,' are used to arrange the fifteen tasks into a series of nine (9) events. Of these nine events, seven events have a single task and two events have more than one task. During the transition from a natural language relationship to a mathematical relation, the events with more than one task need to be transformed into a single item or entity. The events that have more than one task can now be treated in a number of different ways, as the transition from a natural language relationship to a mathematical relation continues.

If the structural relationship for the multiple task events is selected from Warfield's temporal category of relationships, then a relationship that has symmetric relationship properties is required. However, if a natural language relationship is selected that has focused on other substantive, empirical data associated with (1) the total event sequence, (2) local task execution sequence, and/or (3) task logic, then that relationship may be used to order the tasks that populate the events that have multiple tasks. This creates a clear, structured approach that can be used to apply a series of ordering relations, one after the other, to organize, evaluate and process specific types of unknown system structures.

The AN2C approach uses only the 'is-connected-to' natural language relationship, that is converted into a mathematical relation with symmetric properties. The AN2C method is used to identify subsystems in a larger known system, and does not necessarily create any specific object order. The application of a single natural language relationship that has been correctly transformed into a mathematical relation, is usually all that is required to complete a AN2C subsystem identification process. The classic DSM approach, on the other hand, allows the combination of both asymmetric and symmetric relationships to be included into the analysis process. A set of global contextual values is needed to support a process that evaluates the total system structural configuration, and assigns a value to each candidate configuration. The configuration with the best value is selected.

System Structure

The basic structural components and concepts that form the foundation of structural modeling are all associated with a set of objects O, and a contextual relation R, that is mapped over this object set. The relation R is a binary relation, and the objects may be of various types. This approach can be expanded to include other types of relations.

The basic structural modeling project is designed to establish a standard method for the application of the system modeling components associated with system structuring. While there are many ways to view this activity, the lack of a standard approach greatly increases the effort needed to evaluate different approaches.

Another example of the application of BSM techniques to support the implementation of interpretive structural modeling (ISM) practices is provided by Hitchins in *Advanced Systems Thinking, Engineering and Management* on pages 148 and 149. Hitchins states:

> "Interpretive structural modeling (ISM), is a powerful graphical method for finding and presenting purposeful relationships between entities. Given a set of entities, the relationships between them can be identified using Saaty's pair-wise comparison technique. Simply, this technique examines a list of entities taken two at a time and asks questions of the pair."

Saaty, in *Decision Making with Dependence and Feedback: The Analytic Network Process* (1996), describes the analytic hierarchy process (AHP) as:

> "The Analytic Hierarchy Process is a general theory of measurement. It is used to derive ratio scales from both discrete and continuous paired comparisons in multilevel hierarchic structures. These comparisons may be taken from actual measurements or from a fundamental scale that reflects the relative strength of preference and feelings."

Warfield indicated that Intent Structures – the specific type of ISM that Hitchins referred to – are constructed based on logical values (zero and one), and are combined using the logical operators 'and,' 'or,' and the 'exclusive or.' This appears to be another instance where a clear definition of the BSM mathematical components and processes would greatly reduce complexity and confusion.

The Hitchins' example was the initial example that motivated the authors to create the Abstract Relation Type (ART) construct. While Hitchins' description of the ISM process conflicts with many of the other descriptions of the ISM process, the adaptation was creative and very useful. The ART construct is designed to document, describe and present the details associated with any method that uses a relationship to structure a system.

The key to the application of the ART technique to the Automated N-Squared Chart (AN2C), is the clear understanding of the properties of the organizing relation. In the AN2C approach, an 'N by N' matrix is used to list each possible interface between each of the pairs of objects in the system. The AN2C is constructed from two asymmetrical relationships: the feed-forward relation and the feed-backward relation. These two asymmetrical relationships may be used to construct a single symmetrical system relation. The graphical display of fundamental mathematical logic provides the foundation for precise, detailed technical communication using prose, graphics and mathematics.

The basic structural modeling techniques, outlined by Warfield, support the application of any one of these relations. Warfield's techniques do not present a process for the structured transformation of two asymmetric relations into one symmetric relation. Hitchins' application of basic structural modeling and ISM to the AN2C domain was limited on theoretical details and detailed process steps. The authors were able to reproduce and verify the work produced by Hitchins. During this verification process, areas for process improvement were identified and documented. The use of the multi-spaced ART construct coupled with Russell's relation classes and transforms may provide a mechanism to create a standard, reproducible ART AN2C approach.

System Complexity

Given the preceding technical definitions and system example, it is clear that a system's complexity is strongly associated with the system structure. When basic system modeling concepts are used in the process of discovering a previously unknown system, it is clear that the identification and communication of the system structure transforms the system from unknown to known, and greatly reduces the system complexity. A system is defined as a relationship mapped over a set of objects. Information and knowledge associated with the system object set and the structural relationship can be substantive, empirical and/or structural. As is the case with most complex system discovery activities, only partial information is available. Complete knowledge of the system is not achievable due to a number of pragmatic constraints. Under these operational circumstances, information about the system is always approximate. Improvement in the state of system knowledge becomes a cost benefit trade-off that must be evaluated as the system discovery process is executed.

The next example of structuring the configuration of 19 cities shows the interleaving of structural information, empirical data and basic structural modeling analysis techniques to structure and configure a previously unknown system.

System complexity is often found in the form of a hierarchy. The concept of hierarchy assumes the existence of elements and relationships. The concept of theory is associated with the concept of hierarchy through these basic elements. A field theory focuses on the system of relations among the elements. A monadic theory focuses on the system elements. System theories can be a combination of these types, or one type alone.

As shown earlier in this document, a hierarchy is directly represented in a number of matrix forms. A binary matrix is a primary structure that represents hierarchies very well. Basic structural modeling techniques are used to support the presentation and communication of five kinds of models: mental models, prose models, matrix models, structural models and dynamic models. As demonstrated in the previous example, the goal of complexity reduction can be achieved using basic structural modeling techniques to create a structured transition from a mental model to a matrix model and to other structural and dynamic models. A key feature of basic structural modeling is the identification and verification of the structural relation properties. In most cases, the structural relation will have a transitive property. In cases where the structural relation does not have a transitive property, then the approximate nature of the modeling process must be considered. As a first approximation, the relation can be modeled with a transitive property. If the first approximation model does not fit the empirical data, then other values for the structural relation may be modeled and evaluated – governed by the cost benefit constraints of a specific activity.

Transitive Embedding

Transitive embedding is a component of basic structural modeling that mediates the exchange of logical information and empirical data during the system structuring process. The transitive embedding process context includes a group of human experts that provide empirical data, a set of human system operators that control the process, and a unique computational system that is used to encode, structure and evaluate the current state of the system structuring process.

Transitive embedding is an iterative process, and creates three general process states.
- State one, no prior structuring has been accomplished.
- State two, a significant amount of structuring has been completed on at least on aspect of the system.

- State three, the system structuring process has been completed, but an acceptable system structure was not created, and corrective action must be taken.

State one is explored in the next example. State two requires the application of the matrix coupling method and/or the matrix inference methods.

Graph Cycles

The structural models associated with basic structural modeling techniques may contain feedback loops and cycles. Any specific cycle may be replaced by a proxy node that reduces the overall matrix structure to a hierarchy. The cycle groups may be evaluated using weights and maximum cycle length techniques. The specific problem set will determine the best way to address the analysis of feedback cycles. Effective engagement of feedback loops in system structure evaluation is one of the more challenging areas of basic structural modeling.

Basic Structural Modeling Example

The example presented here is based on the identification of the ordering of 19 cities based on the 'is-north-of' binary relationship. Example graphic representations of each step of the identification and configuration process are presented to motivate the discussion of graphical representation of binary relationships using Warfield's techniques.

The problem space consists of 19 cities that have an unknown global configuration. Each city has three data items associated with the city's relative position. In Figure 3, shown below, the binary matrix is a 19 by 19 matrix populated with 361 zeros. At this time the names and locations of the cities is not known. However, based on the irreflexive property of the 'is-north-of relation,' each zero on the matrix diagonal is interpreted as false, which is indicated using the red background color in the matrix cell. The other 342 zeros indicate an unknown state as indicated by the yellow background color.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 3.

This sequence of matrix transforms can be automated or accomplished manually by an individual or in a group. The graphic display during the manual operation provides graphic configuration feedback to guide the operations. The partially-ordered system can now be used to determine the specific city that will provide the most information for a given configuration. The detailed steps in ordering the cities is given in Appendix A. Figure 4 shows the matrix connections with both the connections known from real data (in green), and the connection inferred from the logic of the basic structural modeling techniques (in light blue). Figure 4 shows the completed matrix configuration.

	3	4	6	2	8	10	1	12	7	14	17	9	5	13	16	11	18	19	15
3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	1	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0
12	1	1	1	1	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0
7	1	1	1	1	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0
14	1	1	1	1	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0
17	1	1	1	1	1	1	1	1	0	1	1	0	0	0	0	0	0	0	0
9	1	1	1	1	1	1	1	1	0	1	1	1	0	0	0	0	0	0	0
5	1	1	1	1	1	1	1	1	0	1	1	0	1	0	0	0	0	0	0
13	1	1	1	1	1	1	1	1	0	1	1	0	1	1	0	0	0	0	0
16	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0
11	1	1	1	1	1	1	1	1	0	1	1	0	1	0	0	1	0	0	0
18	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	1	1	0	0
19	1	1	1	1	1	1	1	1	1	1	1	0	1	0	0	1	0	1	0
15	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Figure 4. Final Matrix Configuration.

The current basic structural modeling example demonstrates some of the fundamental techniques used to combine empirical data and logical inference to guide the configuration and structuring of a previously unknown system. These techniques are applied in an iterative fashion until the system configuration is developed to a point where it is useful to the current project team. A set of tests for the basic structural modeling techniques will have to be developed. In order to test these techniques, a known global structure will be identified, and the information encoded into a test system. The matrix development and test process will have access to randomly selected data points from the known global structure. Using these techniques, the basic structural modeling techniques may be tested and validated.

Basic Structural Modeling Problem Definition Summary

The foundation provided by the basic structural modeling components support the generation of system structure descriptions presented in three forms. These system structural description forms are: (1) prose, (2) graphics, and (3) mathematics. The model exchange isomorphism (MEI) developed by Warfield provides an example of the utility associated with a set of language mechanisms (prose, graphics and mathematics) that all have equivalent semantics.

The authors have evaluated portions of the DSM and Automated N-Squared literature. Processes and techniques, focused on decreasing the confusion and complexity associated with the application of basic structural modeling and MEI methods, have been identified, defined and documented. Although the MEI has demonstrated substantial value, it appears that the systems science and systems engineering literature holds a significant body of work that displays a fundamental misunderstanding of the basic MEI mathematics of relations. As a consequence, it creates confusion and complexity, rather than reducing them.

The primary technique that has been used by the authors, is the application of the Abstract Relation Type (ART). The ART approach is designed to standardize and focus the presentation of information associated with basic structural modeling, MEI, ISM, ART N-Squared Charts and ART Design Structure Matrices. The ART technique provides an area to document the system structure design and analysis approach for each of the three language forms: prose, graphics and mathematics.

The prose section of the ART form contains the documentation and explanation for the system structuring, natural language relationship(s). The prose section provides details of the manner in which the natural language relationship is used in the system context, as well as how the system structure is impacted by the chosen relationship. The prose section contains the attributes and properties associated with the natural language relationship. It also provides a concise discussion of the bases used to transform the natural language relationship into a mathematical relation.

The graphic section of the ART form contains the graphic representation of the system structure as well as a written description of how the graphic form was generated from the ART mathematical component. In most cases, the graphic will be generated based on some type of matrix information. The mathematical relation, along with the relation attributes and properties, are documented in the graphic section.

The mathematics section of the ART form contains the mathematical representation of the system structure. The mathematical properties, relations, functions and overall operations are documented in the mathematical section. The mechanisms used to develop, verify and validate the equivalency of the prose, graphical and mathematical forms are detailed in the ART form mathematics section. This MEI pattern can be established by simply referencing a known MEI pattern.

Fundamental natural language relationship properties must be identified and recorded. The process used to transform the natural language relationship into a mathematical relation must be documented. These fundamental activities will greatly reduce the confusion and complexity associated with the basic structuring of large-scale systems. For example, when an unknown and/or undefined process is evaluated, substantive, empirical data gathered from the people involved in the process provides the basis for selecting the natural language relationship used to structure the system. If a group of individuals is asked to list all tasks that come before a specific task, or to list all tasks that come after a specific task, then the group should be able to generally produce these task lists based on their experience. However, if a group is asked to list all tasks that preceded themselves, the individuals in the group will be confused and be unable to provide any substantive information.

The basic structural modeling methods are designed to support a wide range of system structuring techniques that are based on rational, real-world, natural language relationships. In the next phase of the project, a few simple examples will be demonstrated.

The general types of problems associated with the development and communication of basic structural modeling techniques are organized around the five application areas discussed by Warfield in the preface to this document. These five areas are:

1) Mathematics,
2) Conversion of mathematics to computer code,
3) Generation of BSM computer programs,
4) Transition of BSM computer code to ISM computer applications, and
5) Use of ISM techniques in Interactive Management and other group problem solving processes.

Each of these areas is discussed in more detail next.

- ***Mathematics* -** The first problem area, mathematics, contains a number of interesting issues and problems associated with the development and application of basic structural modeling methods and techniques. Most of these issues may be traced to the use of a binary matrix and Boolean operators by Warfield in his analytical work. His addition and augmentation of Boolean operators also provides a challenge to any individual that wishes to understand the mathematical foundations of structural modeling. A key element is the use of Boolean operators, and a matrix that allows only a zero (0) or a one (1). These were not described as Boolean matrices, but as binary matrices with Boolean operations.

  A review of Warfield's earlier work, *Introduction to Electronic Analog Computers*, Prentice-Hall (1959), and *Principles of Logic Design*, Ginn and Company (1963), show an integration of signal-flow logic, binary circuit design logic, Boolean minimization approaches, and Boolean reasoning approaches. These early works clearly show the roots of his later work on the mathematics of structure. A key observation is the use and interleaving of Boolean minimization methods and Boolean reasoning methods in the design of logic circuits and the solution of logic problems.

  Boolean reasoning is a little known, formal method of logic that can be used to great advantage in many logic design areas. Blake's development and application of syllogistic formulas in 1937, has many strong similarities to Warfield's development and use of syllogisms in the solution of structural matrix models. Boolean syllogistic reasoning differs from resolution-based, automated reasoning techniques used by predicate logic in significant ways. Boolean reasoning uses forward chaining. Forward chaining is not possible in predicate logic, because it is unguided and not guaranteed to terminate. In most Boolean problem solving, the problem is not formulated as a theorem to be proved. The primary activity of Boolean reasoning is the identification of a derived Boolean system from a given Boolean system. The derived system may be categorized as a combination of either a functional antecedent or functional consequent type of system. Most work in Boolean reasoning has been associated with the functional and antecedent approaches. Boolean reasoning that is based on functional and consequent approaches has received very little attention since Boole's work in the middle 1800's. A review of the Boolean operators and techniques used by Warfield, indicate a use of functional and consequent approaches.

  The primary root problem associated with the mathematics portion of basic structural modeling, is the use of esoteric Boolean reasoning that is different from predicate calculus and standard Boolean minimization techniques in significant ways.

- ***Conversion of mathematics to computer code*** – The second problem area, the conversion of mathematics to computer code, is associated with the conversion of the Boolean operations identified in problem area one to functional computer code in a manner that allows peer review and communication of the fundamental techniques. This step should create a general set of object-based, or function-based**,** computer code that can be used to explore, discover, verify, validate, and test specific Boolean reasoning operations associated with the specific classes of system structuring problems.

- ***Generation of BSM computer programs*** – The third problem area, generation of BSM computer programs, is associated with the creation of mathematical techniques that are specifically designed to be used in a given type of problematic situation. A library or set of libraries must be developed and tested to support the generation of ISM computer programs that address a specific class of problem.

- ***Transition of BSM computer code to ISM computer applications*** – The fourth problem area, transition of BSM computer code to ISM computer code, is associated with the identification of general types of ISM problems. Once the general types of problems are identified, then specific sets or subsets of the BSM code can be selected and incorporated into a viable ISM software program.

- ***Use of ISM techniques in Interactive Management and other group problem solving processes*** – The fifth problem area, the use of ISM techniques in Interactive Management and other group problem solving processes, is associated with the clear identification of the processes and functions that must be supported by the computer in these interactive group activities. A wide range of these techniques exist, which provide a well-documented starting point for the development and integration of computer-based systems needed to assist large groups of individuals in the solution of complex problems.

Clearly, a broad set of knowledge, skills and abilities is needed to address the complete range of problems specified in the specific problem areas enumerated above. The mathematical problems and transforms needed in the math area are well known to mathematicians, but most of these issues are considered trivial in a pure mathematical sense. The development and application of structured organizational engagement processes is well within the ability of group and organizational development professionals. However, the mathematical details that comprise the heart of the ISM programs are not well known to most people using the 'ISM-like' software. The skills and knowledge necessary to create distributed-network-enabled software programs is available in a number of computing professionals. However, the ability to span all three areas – math, computer system production, and organizational development – is a very rare talent. The Basic Structural Modeling Project was designed to provide a structured basis upon which individuals with expertise in one or more of the needed areas could collaboratively work together and share their knowledge, skills, and abilities to produce a structured set of processes as well as the necessary computer systems needed to effectively implement 'ISM-type' projects.

Appendix A – Detailed Modeling Example

The example presented here is based on the identification of the ordering of 19 cities based on the 'is-north-of' binary relationship. Example graphic representations of each step of the identification and configuration process are presented to motivate the discussion of graphical representation of binary relationships using Warfield's techniques.

The problem space consists of 19 cities that have an unknown global configuration. Each city has three data items associated with the city's relative position. In Figure A-1, shown below, the binary matrix is a 19 by 19 matrix populated with 361 zeros. At this time the names and locations of the cities is not known. However, based on the irreflexive property of the 'is-north-of relation,' each zero on the matrix diagonal is interpreted as false, which is indicated using the red background color in the matrix cell. The other 342 zeros indicate an unknown state as indicated by the yellow background color.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure A-1.

This sequence of matrix transforms can be automated or accomplished manually by an individual or in a group. The graphic display during the manual operation provides graphic configuration feedback to guide the operations. The partially-ordered system can now be used to determine the specific city that will provide the most information for a given configuration.

The initial set of empirical information for this example is associated with the cities numbered 1, 2, 3, 4, 5, 16, 17, 18, and 19. This initial set of data contains 33 data points. The initial example configuration is shown in Figure A-2. The example matrix will experience a series of configuration changes in the form of row and column exchanges. The first series of exchanges are as follows:

- Exchange row 1 and row 3, as well as column 1 and column 3, Figure A-3.
- Exchange row 2 and row 4, as well as column 2 and column 4, Figure A-4.
- Exchange row 1 and row 10, as well as column 1 and column 10, Figure A-5.
- Exchange row 5 and row 17, as well as column 5 and column 17, Figure A-6.
- Exchange row 17 and row 8, as well as column 17 and column 8, Figure A-7.
- Exchange row 17 and row 12, as well as column 17 and column 12, Figure A-8.
- Exchange row 9 and row 1, as well as column 9 and column 1, Figure A-9.
- Exchange row 9 and row 17, as well as column 9 and column 17, Figure A-10.
- Exchange row 11 and row 5, as well as column 11 and column 5, Figure A-11.
- Exchange row 15 and row 16, as well as column 15 and column 16, Figure A-12.
- Exchange row 15 and row 18, as well as column 15 and column 18, Figure A-13.
- Exchange row 18 and row 11, as well as column 18 and column 11, Figure A-14.
- Exchange row 15 and row 19 as well as column 15 and column 19, Figure A-15.

This set of exchanges creates the second basic configuration. As noted, Figure A-2 shows the initial configuration. The following figures (Figure A-3 through Figure A-15), display the matrix transformations resulting from the exchanges described above.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
2	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
6	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1
16	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0
17	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0
18	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0
19	1	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0

Figure A-2. Initial Data Set.

	3	2	1	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1
16	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0
17	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0
18	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0
19	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0

Figure A-3 - Exchange row 1 and row 3; column 1 and column 3.

	3	4	1	2	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1
16	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0
17	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0
18	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0
19	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0

Figure A-4 - Exchange row 2 and row 4; column 2 and column 4.

	3	4	10	2	5	6	7	8	9	1	11	12	13	14	15	16	17	18	19
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0
1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0
12	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1
16	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0
17	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0
18	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0

Figure A-5 - Exchange row 1 and row 10; column 1 and column 10.

	3	4	10	2	17	6	7	8	9	1	11	12	13	14	15	16	5	18	19
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0
6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0
12	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
14	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1
16	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0
5	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0

Figure A-6 - Exchange row 5 and row 17; column 5 and column 17.

	3	4	10	2	8	6	7	17	9	1	11	12	13	14	15	16	5	18	19
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	1	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0
12	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
14	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1
16	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0
5	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0

Figure A-7 - Exchange row 17 and row 8; column 17 and column 8.

	3	4	10	2	8	6	7	12	9	1	11	17	13	14	15	16	5	18	19
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0
17	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
14	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1
16	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
18	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0

Figure A-8 - Exchange row 17 and row 12; column 17 and column 12.

	3	4	10	2	8	6	7	12	1	9	11	17	13	14	15	16	5	18	19
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0
17	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
14	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1
16	0	0	0	0	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
18	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0	0	0

Figure A-9 - Exchange row 9 and row 1; column 9 and column 1.

	3	4	10	2	8	6	7	12	1	17	11	9	13	14	15	16	5	18	19
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0
9	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
14	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1
16	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	1	0	0	0	1	1	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0	0	0

Figure A-10 - Exchange row 9 and row 17; column 9 and column 17.

	3	4	10	2	8	6	7	12	1	17	5	9	13	14	15	16	11	18	19
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
14	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1
16	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0
19	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	1	0	0

Figure A-11 - Exchange row 11 and row 5; column 11 and column 5.

	3	4	10	2	8	6	7	12	1	17	5	9	13	14	16	15	11	18	19
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
14	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1
11	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0
19	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	1	0	0

Figure A-12 - Exchange row 15 and row 16; column 15 and column 16.

	3	4	10	2	8	6	7	12	1	17	5	9	13	14	16	18	11	15	19
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
14	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0
18	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0
11	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1
19	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	1	0	0

Figure A-13 - Exchange row 15 and row 18; column 15 and column 18.

	3	4	10	2	8	6	7	12	1	17	5	9	13	14	16	11	18	15	19
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
14	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1
19	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0

Figure A-14 - Exchange row 18 and row 11; column 18 and column 11.

	3	4	10	2	8	6	7	12	1	17	5	9	13	14	16	11	18	19	15
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
14	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0
19	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0

Figure A-15 - Exchange row 15 and row 19; column 15 and column 19.

Figure A-16 presents the results of a reachability matrix created from the data matrix that contains the additional data. The data from the 33 data points were incorporated into the matrix. Then the data matrix was ordered according to the binary relation. Once the data matrix was ordered, then a reachability matrix was developed, and logical connection were determined and presented in blue.

	3	4	10	2	8	6	7	12	1	17	5	9	13	14	16	11	18	19	15
3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	1	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
12	1	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
17	1	1	1	1	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0
5	1	1	1	1	1	0	0	1	0	1	1	0	0	0	0	0	0	0	0
9	1	1	1	1	1	0	0	1	1	1	0	1	0	0	0	0	0	0	0
13	1	1	1	1	1	0	0	1	0	1	1	0	1	0	0	0	0	0	0
14	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
16	1	1	1	1	1	0	1	1	1	1	1	1	1	0	1	0	0	0	0
11	1	1	1	1	1	0	0	1	1	1	1	0	0	0	0	1	0	0	0
18	1	1	1	1	1	0	1	1	1	1	1	1	0	0	0	1	1	0	0
19	1	1	1	1	1	0	0	1	1	1	1	0	0	0	0	1	0	1	0
15	1	1	1	1	1	0	1	1	1	1	1	1	1	0	1	1	1	1	1

Figure A-16 – Blue 1's Inferred From Reachability Matrix

From the matrix presented in Figure A-16, it is clear that row/column 6 and row/column 14 will provide the best return on gathered data (since they have the greatest number of unknowns). So the data for row/column 6 and row/column 14 were collected and entered into the matrix shown in Figure A-17. At this point the matrix has 39 empirical data points, which represents less than 11 percent of the total matrix capacity. The rest of the data entries are logically inferred using the techniques of basic structural modeling.

	3	4	10	2	8	6	7	12	1	17	5	9	13	14	16	11	18	19	15
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	1	0	1	0	0	1	0	0	0	0	0	1	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
14	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0
19	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0

Figure A-17. Empirical Data re rows 6 and 14; columns 6 and 14 – added to Figure A-15.

The next set of matrix transforms (for figures A-18 through A-21) are listed below along with the associated figure number.

- Exchange row 10 and row 6, as well as column 10 and column 6, Figure A-18.
- Exchange row 17 and row 14, as well as column 17 and column 14, Figure A-19.
- Exchange row 5 and row 17, as well as column 5 and column 17, Figure A-20.
- Exchange row 13 and row 5, as well as column 13 and column 5, Figure A-21.

	3	4	6	2	8	10	7	12	1	17	5	9	13	14	16	11	18	19	15
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	1	1	0	1	0	0	0	0	0	1	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
14	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0
19	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0

Figure A-18 – Exchange row 10 and row 6; column 10 and column 6.

	3	4	6	2	8	10	7	12	1	14	5	9	13	17	16	11	18	19	15
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
14	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
9	0	0	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
17	0	0	0	0	1	1	0	1	0	1	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0
19	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0

Figure A-19 – Exchange row 17 and row 14; column 17 and column 14.

	3	4	6	2	8	10	7	12	1	14	17	9	13	5	16	11	18	19	15
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
14	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	1	1	0	1	0	1	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0
18	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0
19	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0

Figure A-20 – Exchange row 5 and row 17; column 5 and column 17.

	3	4	6	2	8	10	7	12	1	14	17	9	5	13	16	11	18	19	15
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
14	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	1	1	0	1	0	1	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
16	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0
11	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0
18	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0
19	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0

Figure A-21 – Exchange row 13 and row 5; column 13 and column 5.

Figure A-22 presents the results of a reachability matrix created from the data matrix that contains the additional data. Once the initial data set is properly prepared, the techniques associated with basic structural modeling are used to determine the total system configuration from limited system data. This technique greatly reduces the cost of collecting data and the complexity associated with creating a global system structure from local, limited data sets.

	3	4	6	2	8	10	7	12	1	14	17	9	5	13	16	11	18	19	15
3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	1	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
12	1	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
1	1	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0
14	1	1	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
17	1	1	1	1	1	1	0	1	0	1	1	0	0	0	0	0	0	0	0
9	1	1	1	1	1	1	0	1	1	1	1	1	0	0	0	0	0	0	0
5	1	1	1	1	1	1	0	1	0	1	1	0	1	0	0	0	0	0	0
13	1	1	1	1	1	1	0	1	0	1	1	0	1	1	0	0	0	0	0
16	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0
11	1	1	1	1	1	1	0	1	1	1	1	0	1	0	0	1	0	0	0
18	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	1	1	0	0
19	1	1	1	1	1	1	0	1	1	1	1	0	1	0	0	1	0	1	0
15	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0

Figure A-22– Additional **Inferred** Connection In Light Blue.

Data from 39 points are now incorporated into the matrix. Data for an additional 7 points is collected after the matrix configuration in Figure A-22 is evaluated. These data points are associated with rows 7 and 1. The additional data is presented in the matrix shown in Figure A-23, which provides the basis for the next set of matrix configuration changes.

	3	4	6	2	8	10	7	12	1	14	17	9	5	13	16	11	18	19	15
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
12	0	1	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
14	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	1	1	0	1	0	1	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
16	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0
11	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0
18	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0
19	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	1	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0

Figure A-23. More information collected associated with rows 7 and1.

The next set of matrix configuration changes (Figures A-24, A-25, and A-26) are as follows:

- Exchange row 7 and row 12, as well as column 7 and column 12, Figure A-24.
- Exchange row 12 and row 1, as well as column 12 and column 1, Figure A-25.
- Exchange row 7 and row 12, as well as column 7 and column 12, Figure A-26.

	3	4	6	2	8	10	12	7	1	14	17	9	5	13	16	11	18	19	15
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	1	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
7	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
14	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
16	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	0	0	0	0
11	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0
18	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0
19	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	1	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0

Figure A-24 – Exchange row 7 and row 12; column 7 and column 12.

	3	4	6	2	8	10	1	7	12	14	17	9	5	13	16	11	18	19	15
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
12	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
14	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	1	1	0	0	1	1	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
16	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	0	0	0	0
11	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0
18	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0
19	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	1	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0

Figure A-25 – Exchange row 12 and row 1; column 12 and column 1.

	3	4	6	2	8	10	1	12	7	14	17	9	5	13	16	11	18	19	15
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
7	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
14	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	1	1	0	1	0	1	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	1	0	0	1	0	1	0	0	0	0	0
11	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	1	0	0	0
19	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	1	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0

Figure A-26 – Exchange row 7 and row 12; column 7 and column 12.

Figure A-27 shows the matrix connections with both the connections known from real data (in green), and the connection inferred from the logic of the basic structural modeling techniques (in light blue).

	3	4	6	2	8	10	1	12	7	14	17	9	5	13	16	11	18	19	15
3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	1	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0
12	1	1	1	1	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0
7	1	1	1	1	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0
14	1	1	1	1	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0
17	1	1	1	1	1	1	1	1	0	1	1	0	0	0	0	0	0	0	0
9	1	1	1	1	1	1	1	1	0	1	1	1	0	0	0	0	0	0	0
5	1	1	1	1	1	1	1	1	0	1	1	0	1	0	0	0	0	0	0
13	1	1	1	1	1	1	1	1	0	1	1	0	1	1	0	0	0	0	0
16	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0
11	1	1	1	1	1	1	1	1	0	1	1	0	1	0	0	1	0	0	0
18	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	1	1	0	0
19	1	1	1	1	1	1	1	1	1	1	1	0	1	0	0	1	0	1	0
15	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Figure A-27 – More Inferred Connections in Light Blue.