

An Examination of a Structural Modeling Risk Probe Technique

William Anderson
Software Engineering Institute
Phillip Boxer
Boxer Research Limited
Lisa Brownsword
Software Engineering Institute

October 2006

SPECIAL REPORT
CMU/SEI-2006-SR-017

**Using Structural and Dynamic Modeling to Support Acquisition, Development, and Sustainable
Deployment of Software-Intensive Systems, an Independent Research and Development Project**
Unlimited distribution subject to the copyright and permission granted by sponsoring organization



This report was prepared for the

SEI Administrative Agent
ESC/XPK
5 Eglin Street
Hanscom AFB, MA 01731-2100

The ideas and findings in this report should not be construed as an official DoD position. It is published in the interest of scientific and technical information exchange.

This work is sponsored by the NATO Maintenance and Supply Agency, which granted permission for its release on January 22, 2007. The Software Engineering Institute is a federally funded research and development center sponsored by the U.S. Department of Defense.

Copyright 2007 Carnegie Mellon University.

Special permission to use PAN in this Technical Probe is granted by Boxer Research Limited.

NO WARRANTY

THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN "AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

Use of any trademarks in this report is not intended in any way to infringe on the rights of the trademark holder.

Internal use. Permission to reproduce this document and to prepare derivative works from this document for internal use is granted, provided the copyright and "No Warranty" statements are included with all reproductions and derivative works.

External use. Requests for permission to reproduce this document or prepare derivative works of this document for external and commercial use should be addressed to the SEI Licensing Agent.

This work was created in the performance of Federal Government Contract Number FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 252.227-7013.

For information about purchasing paper copies of SEI reports, please visit the publications portion of our Web site (<http://www.sei.cmu.edu/publications/pubweb.html>).

Table of Contents

| | |
|---|-----------|
| Abstract | v |
| 1 Introduction | 1 |
| 2 Technical Probe Study Approach | 3 |
| 2.1 Visual PAN Models | 4 |
| 2.1.1 Model Views | 4 |
| 2.1.2 Visual PAN Syntax | 8 |
| 2.2 PAN Matrices | 14 |
| 2.3 Interoperability Landscapes | 16 |
| 3 Outcomes from the Approach | 19 |
| 3.1 Type III Mission Risks | 19 |
| 3.2 Type II Composition Risks | 21 |
| 3.3 Type I Performance Risks | 23 |
| 3.4 Type 0 Constructive Risks | 25 |
| 3.5 Summary of Probe Outcomes | 25 |
| 4 Observer Summary Conclusions | 27 |
| References | 29 |

List of Figures

| | |
|--|----|
| Figure 1: A Top-Down PAN Model of Aspects of the Program (Stage One) | 4 |
| Figure 2: Physical World View | 5 |
| Figure 3: Cognitive World View | 6 |
| Figure 4: Effects World View | 6 |
| Figure 5: Dynamic Relationships—the Four Colors | 8 |
| Figure 6: Entity and Connector Symbols | 9 |
| Figure 7: Hierarchy and Connector Symbols | 10 |
| Figure 8: Process Entity and Connector Symbols | 10 |
| Figure 9: Symbols Pertaining to Demand | 11 |
| Figure 10: Symbols Depicting Synchronization and Coordination | 12 |
| Figure 11: Entity Connection Matrix | 13 |
| Figure 12: Matrices Based on Six-Level Stratification | 15 |
| Figure 13: Matrix Stratification with Exemplar Entities | 16 |
| Figure 14: Projecting Through the Matrix | 17 |
| Figure 15: An Example Interoperability Landscape | 18 |
| Figure 16: Mission Demand Projection | 20 |
| Figure 17: Mission Awareness Landscape | 21 |
| Figure 18: Projecting for Orchestration Risk | 22 |
| Figure 19: Orchestration Landscape | 23 |
| Figure 20: Performance Risk Projection | 23 |
| Figure 21: Performance Risk Landscape | 24 |
| Figure 22: Indirect Management Control | 24 |

Abstract

The integration of demand dynamics into a structural model is a key conceptual shift for software engineering. This report examines the utility and transition characteristics of a structural dynamic analysis modeling technique called Projective ANalysis (PAN) that was used on an interoperability technical probe of a North Atlantic Treaty Organization (NATO) modernization program. The report focuses on the process, rather than the findings, of the probe. Organizational entities are referred to generically and, in some instances, aggregated.

The probe involved workshops and interviews, conducted over a two-week period with more than 25 people, followed by analysis of the data gathered. PAN was used to model the NATO program as a system of systems. The model is a rapid assessment based on the subjective understanding of the interviewed subject matter experts. It is a snapshot in time; while dynamic stocks and feedback loops are represented, their temporal characteristics are not. From the model, five perspectives were analyzed for different forms of interoperability risk. These analyses produced three-dimensional projections that depict clusters of shared interfaces. The separation between these clusters identifies the interoperability risks.

The report notes that the PAN technique starts from a client-driven context and builds visual representations that are easily understood by, and bring immediate value to, the client. Further, the report observes that the modeler is critical to this technique and must possess expert skills in the Microsoft Visio application as well as an ability to quickly grasp and characterize the constructs and objects revealed through dialog-based inquiry.

The report concludes that PAN appears to offer a fresh approach, new insights, and appropriate mechanisms to study complexity in systems of systems. The potential for applying and amplifying this technique appears to be significant. The report also determines that an experienced process modeler would have little difficulty adapting to this modeling paradigm.

1 Introduction

This report examines the technique used in an interoperability technical probe of the system of systems involved in the sustainment of a NATO modernization (aircraft retrofit) program. The NATO sustainment organizations are challenged to put into operation a new set of aircraft and support-service capabilities and duplicate the initial aircraft's upgrade into the balance of the NATO fleet.

For the purpose of the probe, we interpreted system-of-systems interoperability in a broad sense. We examined the hardware and software in the context of its operational and sustainment environments. Therefore, the system of systems examined includes the many ground and airborne systems and the diverse organizations (represented as virtual systems) required to operate and sustain the upgraded NATO fleet.

At a preparatory briefing, a number of critical operational issues were identified that would have to be addressed by the modernized aircraft during the course of its sustainment phase. These issues were grouped into five broad categories:

1. surveillance
2. battle management
3. flight deck
4. maintenance
5. software

From the perspective of these critical operational issues, we approached the question of interoperability. Interoperability risks provide the link between the operational requirements of the capability and the way the capability is sustained.

In this report, we describe the probe technique in the context of the NATO engagement. We intersperse observer notes that highlight adoption or transition issues for the tools and technique described. The modeling technique is somewhat complex and our presentation here is targeted at researchers interested in modeling (and simulation) as a complex-system exploratory technique. We do not present the internal algorithms of the transformations performed, but we do endeavor to provide substantive details of the externally evident objects, constructs, and concepts employed in the technique.

We have abstracted up to the general classes of risks found. We are concerned here with presenting the technique employed and not the specific details of the case as they are confidential to NATO. (Thus, some of the figures are purposely scaled to make details illegible.)

2 Technical Probe Study Approach

Our approach assumed that interoperability was an issue at six different and successively broader levels: (1) services, systems, and know-how; (2) activity chains involved in integrating components; (3) activities supporting the operational capability; (4) orchestration of capabilities by crew and operators; (5) operational performance of the capability; and (6) mission environments. At the broadest level, we sought interoperability risks in the way different command authorities were able to work together collaboratively. At narrower levels, we looked at the way different Command and Control Information Systems (C2IS) assets and capabilities could effectively produce combined effects. At the narrowest levels, we examined the ability of hardware, software, and firmware to work together as effective subsystems within larger systems.

Observer Note: This stratification is not unique to this probe or the situation being examined, but it is fundamental to the modeling technique. The six layers form a framework against which the client's people, processes, and technical structures are analyzed in relation to the demands being placed upon them.

We used a method called Projective ANalysis (PAN) in the probe to build models of the way these levels interoperate in terms of the relationships between people, processes, technologies, and demands. The models were built during workshop sessions attended by knowledgeable staff from NATO. The study team gained an understanding, objectively and rapidly, of the problems and issues from this approach. Also, as models emerged, all participants seemed to show appreciation for one another's perspectives.

Observer Note: As is common with many modeling techniques, the model development process itself has a value-adding component because it builds shared understanding and promotes communication.

The models were produced in stages:

1. **Visual PAN Models**—a layered, graphical representation that conforms to a specified syntax with symbols and interconnection rules
This stage is interactive; subject matter experts are brought together in workshops. (See Section 2.1.)
2. **PAN Matrices**—a set of stratified spreadsheets that juxtapose activities with events
This stage is generated offline by the study team from the Visual PAN Models. (See Section 2.2.)
3. **Interoperability Landscapes**—the interrelationships specified in the matrices
This stage is generated offline and becomes the primary reasoning representation back to the stakeholder community. (See Section 2.3.)

PAN builds the models top-down (in the subjective opinion of interviewed stakeholders) to give an account of the problems and issues identified by workshop participants and individuals interviewed. Although advertised to be capable of considerable refinement, the models emerging from the workshops reflected the main characteristics of the program and its sustainment challenges.

Observer Note: This refinement potential is evident. NATO is currently exploring this refinement to help technically characterize change requests and quantify sustainment costs.

2.1 VISUAL PAN MODELS

Five models were built during the course of the workshops for the first stage. Figure 1 illustrates one of them.¹

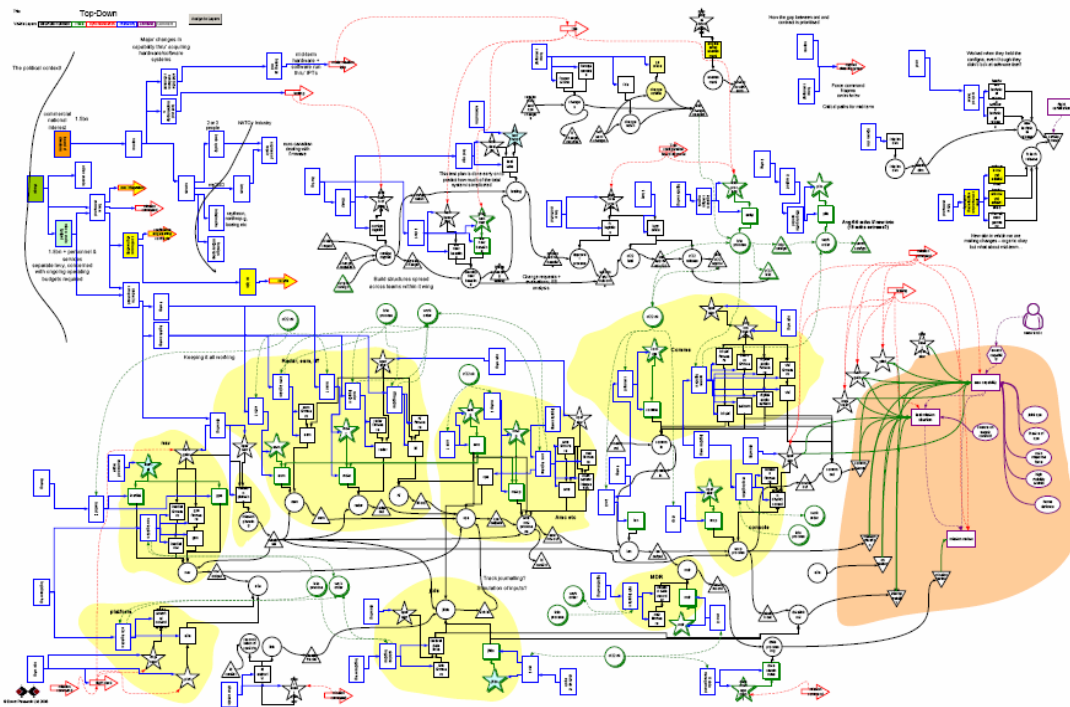


Figure 1: A Top-Down PAN Model of Aspects of the Program (Stage One)

Observer Note: These stage one models are analogous to other process or entity relationship diagrams and suffer from the same rapid increase in complexity. Before long, the models can become “eye charts” that convey the global complexity of the situation but do little to indicate specific risks.

2.1.1 Model Views

Preliminary interviews and background document reviews were used to familiarize the researchers with the problem space and linguistics of the organization. From this preliminary work, three “world views” were postulated that provided a framework for the discovery process. Exemplar artifacts from the background documents were used to illustrate the views, providing familiar touch points into the workshop participants’ environment.

¹ To protect NATO’s confidentiality, selected figures are purposely scaled to be unreadable in detail.

2.1.1.1 Physical World View

A physical world view explores the technology, processes, and people that are required to satisfy the systems' data requirements. The photograph in Figure 2 illustrates the physical world view for the NATO project.



Figure 2: Physical World View

2.1.1.2 Cognitive World View

A cognitive world view explores the technology, processes, and people that are required to translate the physical world view's data into information that can be presented to the appropriate decision makers at the appropriate time. Figure 3 shows the artifact that was selected to represent the cognitive world view in this technical probe.

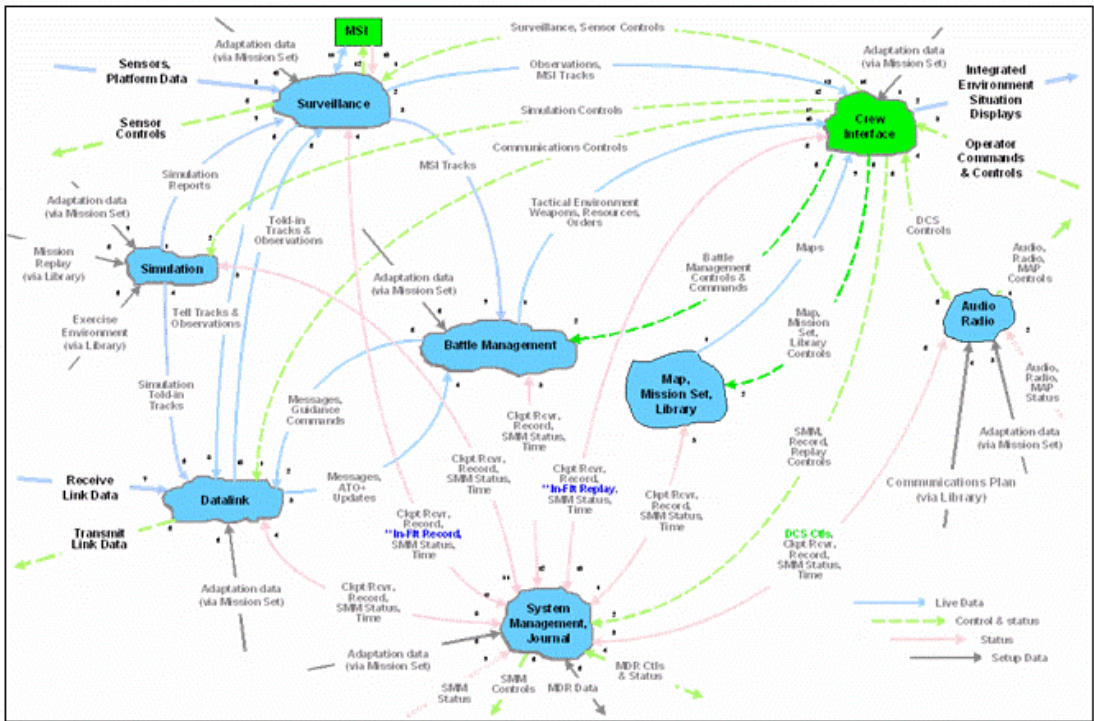


Figure 3: Cognitive World View

2.1.1.3 Effects World View

An effects world view explores the technology, processes, and people that are required to align the physical and cognitive world views with desired operational effects. Figure 4 is the effects world view artifact used for the NATO technical probe.

- | |
|---|
| <p>Different demands on force command being supported</p> <ul style="list-style-type: none"> 1 air surveillance 2 air operations <ul style="list-style-type: none"> 2.1 close air support 2.2 reconnaissance 2.3 airlift 3 tactical missions <ul style="list-style-type: none"> 3.1 air-to-air control 3.2 air-to-ground control 3.3 airspace management 3.4 air policing 3.5 search and rescue 3.6 force marshalling 3.7 threat warning |
|---|

Figure 4: Effects World View

In fact, the effects that we explored encompassed a broader demand impact than shown in Figure 4. We probed not only the military missions but also civil aviation restrictions and business drivers (such as budgets and national interests).

Observer Note: This technique worked well and should be quite reproducible. The three world views appear to cross domains in most cases but could be reconstituted as appropriate to fit the client's environment. These world views (1) help to find smaller groups of stakeholders that could work as a team to build the visual models, (2) serve to bring the analysts up to speed with the client's domain (i.e., let the analyst speak in the client's language with some artifacts that were familiar to the client), and (3) provide a comfortable starting point for the visual modeling (i.e., a context derived from client artifacts).

If a different client's artifacts lend themselves to different world views—development, test, and sustainment for example—the process could easily commence from those artifacts. One mitigating factor is the analyst's desire to explore market-demand influences; in that event, something analogous to the Effects World View would be desirable but not mandatory.

2.1.1.4 Dynamic Relationships

The modeling of the dynamic characteristics (the models are a structural snapshot of these characteristics) is very challenging and involves a blending of technical, cognitive, process, and organizational perceptions. We used a brainstorming aide referred to as the **four colors**, which has origins in war gaming, to facilitate discussion and initial expression of these dynamic characteristics. In war games, blue represents friendly forces; red, the enemy forces; white, the referees; and black, intelligence. We modified this rubric to fit the client context.

In NATO's case, we applied the colors to describe the program's capabilities (blue) in relation to the particular demands being placed upon them (red), within the context of what is driving the mission environment (black). White was used to represent the management of the interoperability among all of these constituents. A significant study team hypothesis was that NATO's focus was biased toward managing the capabilities (white/blue) in a way that was divisive to the ever-changing demand versus mission driver (red/black) relationship.

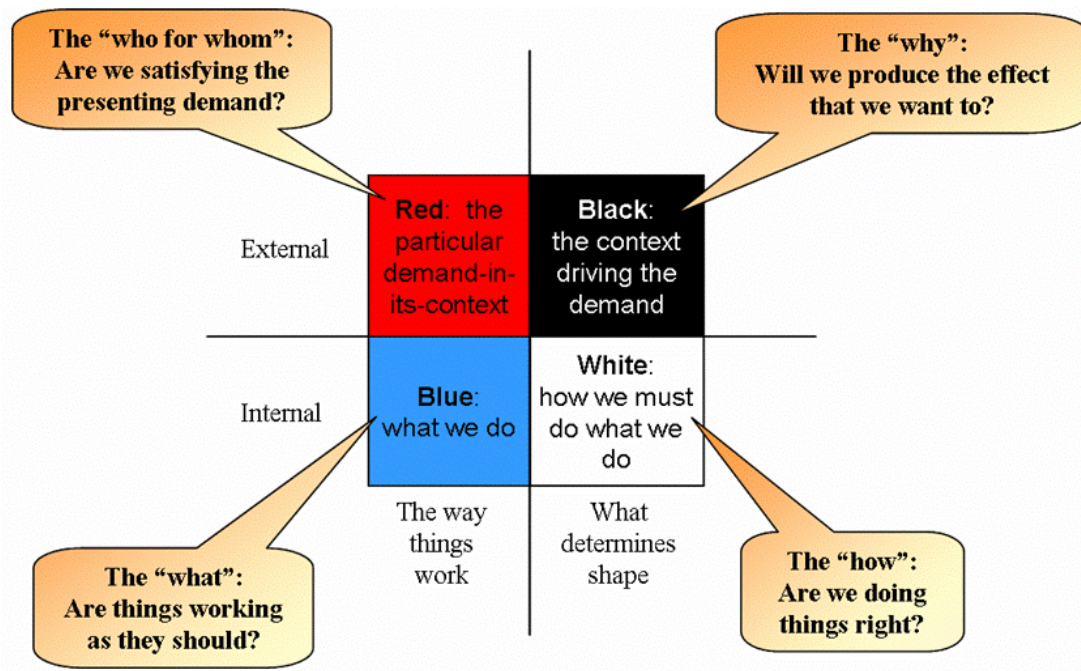


Figure 5: Dynamic Relationships—the Four Colors

*Observer Note: An emphasis on the importance of demand is a strong theme in PAN, at least when system-of-systems interoperability is desired. An assumption is that the client’s interoperability issues are and should be strongly influenced by the need/desire to be reactive to **changing** demands. If demand is stable and pre-identified (e.g., large nation-state military threat scenarios, huge stable demand for sport utility vehicles, or the best healthcare that money can buy), traditional hierarchical structures and monolithic systems work well. However, terrorism has changed the threat, gas prices fluctuate wildly, and healthcare costs have skyrocketed—forcing market-driven demand change. PAN helps find the gaps in an organization’s ability to react to these changes.*

2.1.2 Visual PAN Syntax

Visual PAN has a specified syntax of symbols. These symbols and their relationship rules generate five interlocking layers in the visual model.

The layers are

- **Structure/Function**—the physical structure and functioning of resources and services
- **Hierarchy**—the formal hierarchies and standards under which both the nondigital and digital aspects of the whole are held accountable
- **Trace**—the digital processes and software that interact with the physical processes
- **Demand**—the organization of customers’ needs as demands on the way the enterprise is organized
- **Synchronization**—the lateral relations of synchronization and coordination within the enterprise and between the enterprise and its customers

Each of these layers has a well-defined set of colors, symbols, and relationship rules that are described in the following sections.

Observer Note: While they were not in the traditional software engineering dialect completely, these layers were easily assimilated by the client. They offer a decomposition that is useful to the subsequent analysis.

2.1.2.1 Structure/Function

The physical structure and function of resources and services are modeled using five entity symbols and two connectors, as shown in Figure 6.² A black square represents *capability* that determines the behavior of another capability or of a process. A black star represents *know-how* that can alter the way in which other know-how and capabilities determine behavior. Know-how can be a party to satisfying customer situations. A black circle represents a physical *process*. An up-pointing black triangle represents an *event* generated by a process. A down-pointing black triangle represents an *outcome* generated by a process. Outcomes may either satisfy or be contained by a customer situation (see Section 2.1.2.4). The two connectors are *determines* and *supplies*; they are represented by black curved and angular arrows, respectively.

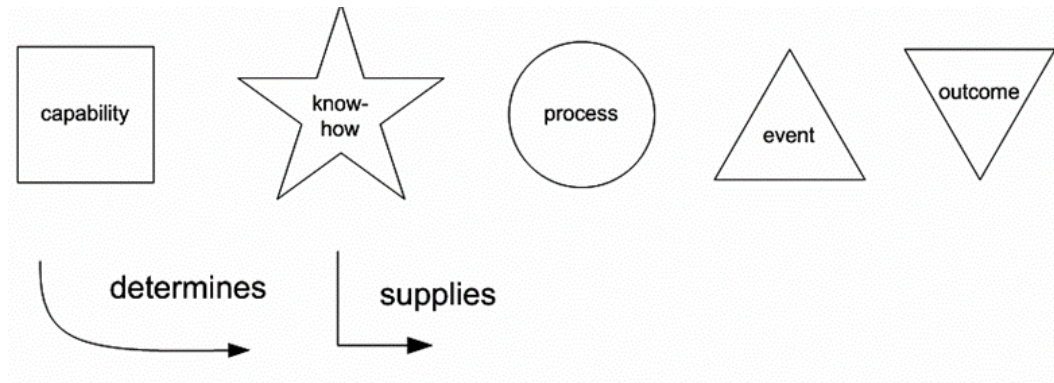


Figure 6: Entity and Connector Symbols

Observer Note: This symbology along with the other four layers of symbols (described in the following sections) presents some complexity that does require analyst mastery. We observed that the client participants quickly adopted this symbology and made suggestions for its use.

2.1.2.2 Hierarchy

The formal hierarchies and standards under which nondigital and digital aspects are held accountable were modeled using a blue rectangle called a *unit*. A unit is accountable for all the entities under its control. The unit entity is also used to represent the state of a set of entities. Blue angular arrows called *controls* are the connectors that designate the entities controlled by a given unit.

Figure 7 shows unit and control symbols.

² The connectors depict the bridge between layers. Their inclusion in a given layer description is somewhat arbitrary, but it is generally done to represent the primary use of the connector.

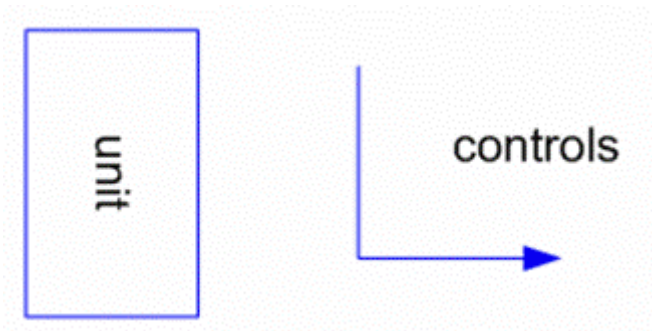


Figure 7: Hierarchy and Connector Symbols

Observer Note: These are simple constructs that were understood by all participants.

2.1.2.3 Trace

The digital processes and software that interact with the physical processes are modeled using four entities and two connectors. A *system*, symbolized by a green shadowed square, is a digital system that can determine the behavior of another system, a digital process, or a physical process. A green shadowed star represents *design* that can alter the way in which other designs and systems determine behavior. Design can be necessary to satisfy specific customer situations. A digital process (i.e., a software program) in the model is represented by a green shadowed circle called a *dprocess*. A digital event or artifact created by a physical or digital process is represented by a green shadowed triangle called a *trace*. The connectors in the trace layer are similar to those in the structure/function layer, but their names include a “d” (for digital) prefix. *Ddetermines* is a green, curved, dashed arrow; *dsupplies* is a green angular arrow.

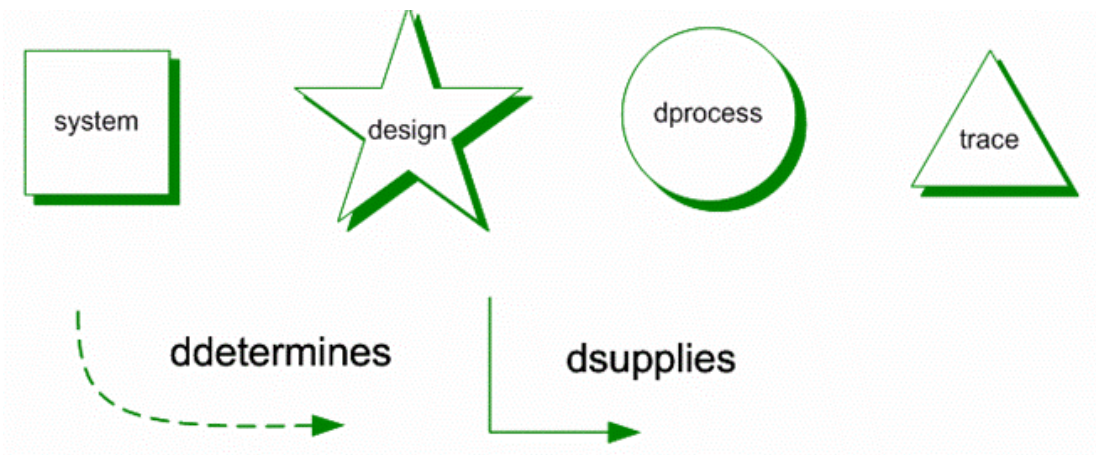


Figure 8: Process Entity and Connector Symbols

Observer Note: We observed that clients had little difficulty understanding and using the digital domain symbols. We also saw no desire to represent an object that did not easily fit in this template of symbols somewhere. Perhaps skillful facilitation was responsible for this ease-of-use.

2.1.2.4 Demand

Four entities and three connectors are used to model customer needs as demands on the way the enterprise is organized. The *problem domain* entity is represented by a purple bust. Problem domains represent the broadest classifications of demand sources placed upon the enterprise that is being modeled. For example, military aircraft must respond to demands from civilian and military aviation authorities; these demand sources could be considered two different problem domains. A purple hexagon called *demand situation* is used to represent a particular context. A particular formulation of a demand within that context—that is, a customer situation within a context-of-use—is represented by the *customer situation* symbol, a purple rectangle. A customer situation may also represent the state of the demand. The last demand entity is a purple oval, the *driver*. A driver is the motivation behind a given customer demand. The demand layer connectors are the purple curved arrow *satisfies*, the purple angled arrow *drives*, and the purple, curved dashed arrow *contains*.

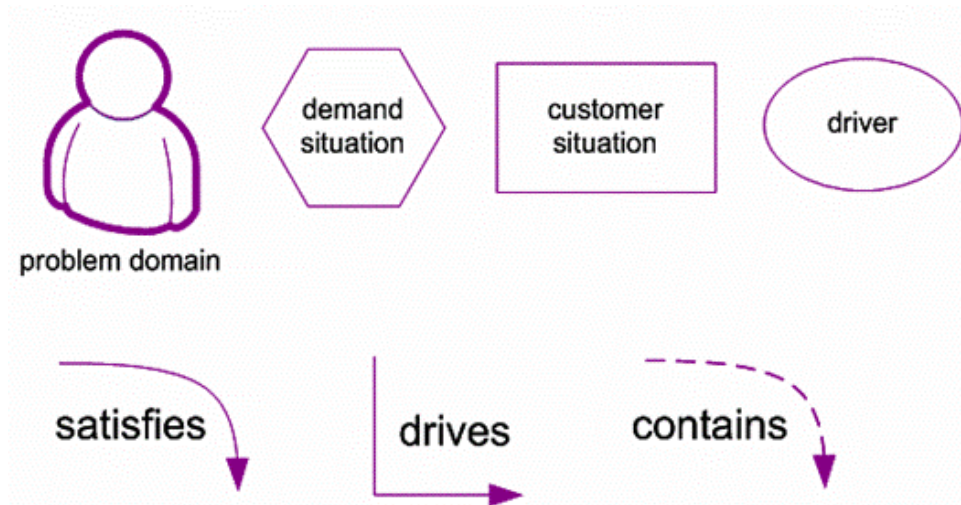


Figure 9: Symbols Pertaining to Demand

Observer Note: These symbols depict the model layer that is least familiar to software engineers. The analyst needed to coax the recognition of their subtle distinctions, particularly from demand-agnostic or demand-uninformed stakeholders.

2.1.2.5 Synchronization

A lateral relation (synchronization and coordination) within and between enterprises or between enterprises and their customers is represented by a red arrowhead called *order*. Order connects to other entities by a red, curved, solid arrow called *frames*. A red shadowed arrow called *dorder* represents the synchronization and coordination of the associated data. Dorder connects by a red, curved, dashed arrow called *dframes*.



Figure 10: Symbols Depicting Synchronization and Coordination

Observer Note: This more abstract notion of synchronization required a bit of analyst facilitation by relevant examples, but overall it was accepted and appreciated by the client stakeholders.

2.1.2.6 Connector Rules

The modeling syntax imposes a set of connector rules to define simple object relations. Figure 11 shows the connector rules in a matrix. The names of the entities are listed at the beginning of the rows (to indicate sources) and across the columns (to show destinations). The name of a connector in a cell shows how the entities are connected.

Read the table in this way:

1. Start from a sourcing entity on the left edge.
2. Follow the row headed by that entity name to a cell containing a connector.
3. Move up in the column to the appropriate destination entity.

For instance, the source entity *event* is connected by *supplies* to the destination entity *process*.

These simple object relations can be analyzed for patterns (complex or emergent³) that facilitate the structuring of entities according to a six-layer stratification. These six layers (from physical processes to problem domains, as introduced in the beginning of Section 2) are augmented by nine matrices of aligning structures (described in Section 2.2). These aligning structures model both the mechanisms that determine the organization’s ability to react and manage itself (e.g., governance, actors, design authority—the **determining** structures) and those that carry out the directives of the determining structures (e.g., systems, processes, agents—the **determined** structures) within the enterprise being modeled.

Observer Note: Herein lays the beauty of this model. The infinitely complex visual (some would call them spaghetti) diagrams (which due to the layering, selective display, and zoom capabilities of the tool are still reasonably tractable) become highly structured and analyzable with automated tooling.

³ The patterns are represented by model-generated entities that emerge from more complex interactions than are represented by the simple entity-connector-entity constructs represented in Figure 11 (e.g., markets, orchestrations, and super-channels).

| Layer | destination source | Know | Capability | Process | Event | Outcome | Design | System | Digital | Trace | Dorder | Order | Unit | Problem | Demand | Customer | |
|--------------------|-----------------------|------------|------------|------------|----------|----------|-------------|-------------|-------------|-----------|----------|----------|-----------|---------|-----------|-----------|-----------|
| | | How | | | | | | | Process | | | | | Domain | Situation | Situation | Driver |
| Structure-Function | ☆ Know How | determines | determines | | | | | | | | | | | | | | satisfies |
| | □ Capability | | determines | determines | | | | | | | | | | | | | |
| | ○ Process | supplies | supplies | | supplies | supplies | supplies | supplies | | | | | | | | | |
| | △ Event | | | supplies | | | | | dsupplies | | | | | | | | |
| | ▽ Outcome | | | supplies | | | | | dsupplies | | | | | | | | satisfies |
| Trace | ☆ Design | | | | | | ddetermines | ddetermines | | | | | | | | | satisfies |
| | □ System | | | determines | | | | ddetermines | ddetermines | | | | | | | | |
| | ○ Digital Process | | | | | | | | | dsupplies | | | dsupplies | | | | dsupplies |
| | △ Trace | | | supplies | | | | | | dsupplies | | | | | | | |
| Synchronization | ⇒ Dorder | | | | dframes | dframes | | | | dframes | dframes | | | | | | dframes |
| | ⇒ Order | frames | frames | | | | frames | frames | | | | frames | | | | | frames |
| Hierarchy | ▮ Unit | controls | controls | | | | controls | controls | dsupplies | | controls | controls | controls | | | | |
| Demand | 🔑 Problem Domain | | | | | | | | | | | | | | | contains | |
| | ⊖ Demand Situation | | | | | | | | | | | | controls | | | | contains |
| | 📄 Customer Situation | | | | | contains | | | dsupplies | | | | | | | | contains |
| | ○ Driver | | | | | | | | | | | | | | | | drives |

Figure 11: Entity Connection Matrix

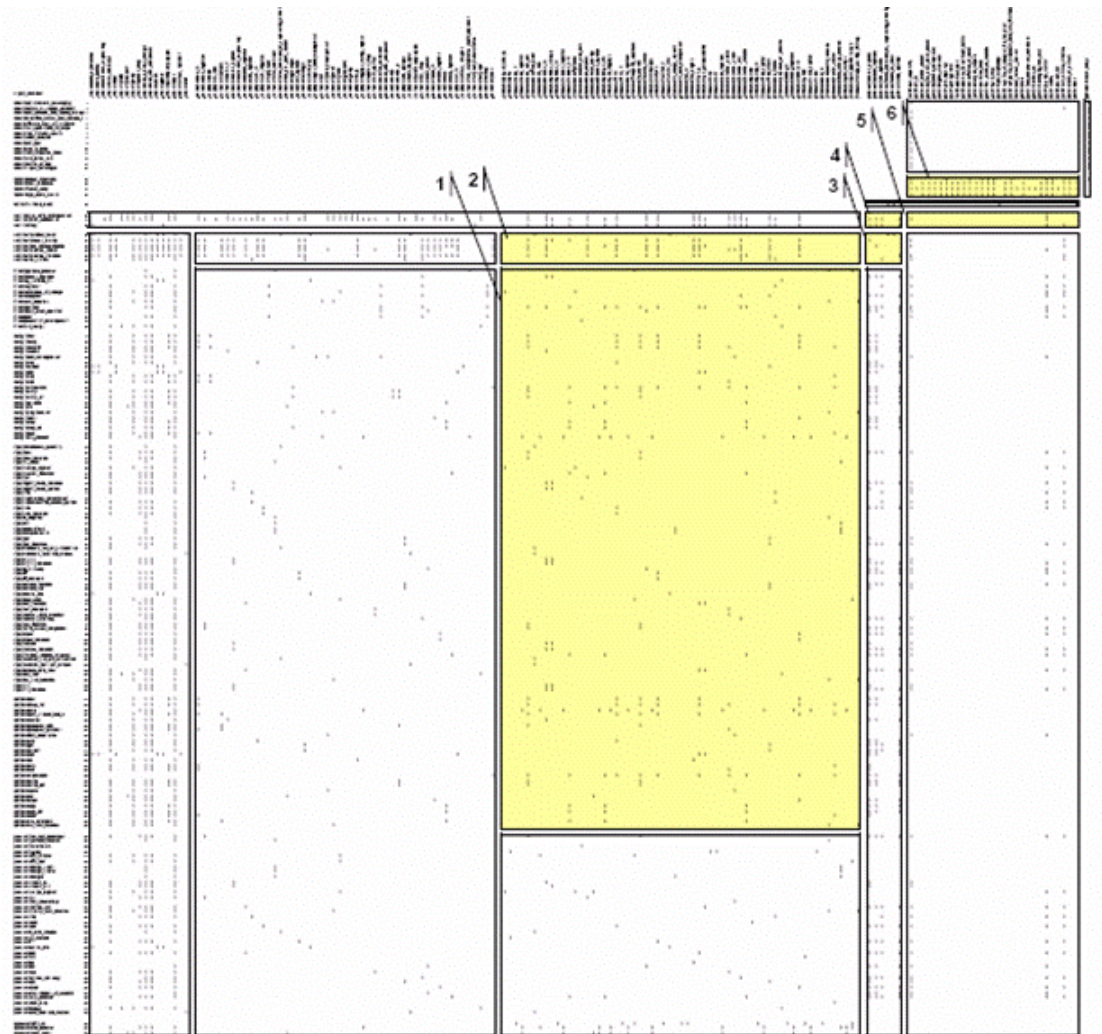
2.2 PAN MATRICES

Each stage one workshop aimed to represent the dynamic relationships between the different aspects (colors) of the world view being examined. In stage two of the probe, we combined the entity relationship diagrams and converted them to a stratified matrix.

We analyzed the models from the point of view of the different demands being placed on the interoperations they described. From this analysis, we identified *gaps* in the way these levels were able to interoperate. To identify these gaps, we converted the models into stratified matrices using an automated utility written in Prolog—called Prolog PAN—that leverages the entity/connector relationships and a set of recognizable patterns (e.g., a hierarchical unit that determines another hierarchical unit produces a derived source entity called an orchestration). The original entities and the derived entities became the labels for the stratified matrix. The matrix was populated as a binary representation of the connections between the entities. The physical processes were at the bottom of this stratification; the demands driving the operational use of the capability were at the top. The strata in between aligned the physical processes to the demands.

The matrices produced are shown in Figure 12. Even the extremely shrunken perspective of the matrices in Figure 12 points out an interesting view of the situation. The higher levels (such as mission environments, level 6) of the matrix are very small, indicating little system-of-systems-wide recognition of demand pull.⁴ The densely populated first block on the left of the matrix is the organizational hierarchy above the first level of management and Integrated Product Team (IPT) processes. This density indicates a significant reliance on hierarchical structure to facilitate interoperability.

⁴ Demand includes such elements as military missions, civil aviation restrictions, and business drivers (e.g., budgets and national interests).



Six-Level Stratification

1. Services, systems, and know-how
2. Activity chains involved in integrating components
3. Activities supporting the operational capability
4. Orchestration of capabilities by crew and operators
5. Operational performance of the capability
6. Mission environments

Figure 12: Matrices Based on Six-Level Stratification

Observer Note: The translation from the visual models to the matrix representation is automated by a tool written in Prolog. We did not examine or explore this tool. It would certainly be an area for further investigation, particularly if one is concerned about adaptability.

Figure 13 is a general representation of the stratified matrix structure. The six key layers are highlighted in gray (the darker shading); they correspond to the list of levels provided with Figure 12. The other sections of the matrix illustrate the structures that facilitate the connection of the underlying infrastructures to the top level contexts-of-use region. This figure also includes some exemplars of the entity types that populate the various sections of the matrix.

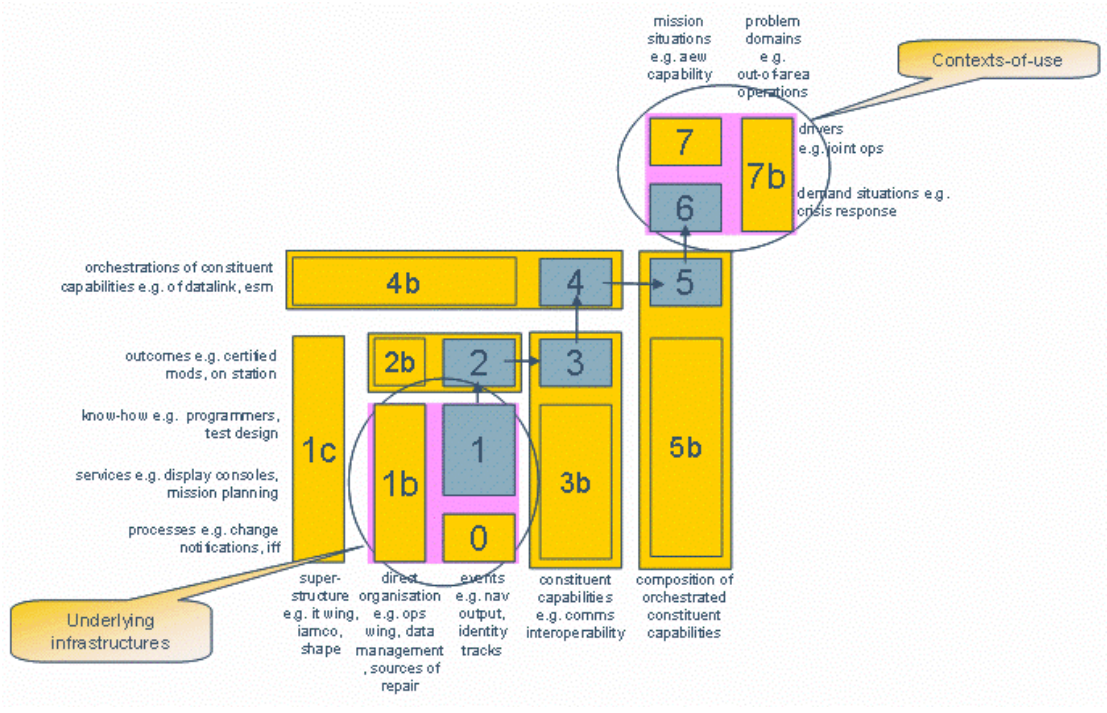


Figure 13: Matrix Stratification with Exemplar Entities

2.3 INTEROPERABILITY LANDSCAPES

In the third stage of the probe, we produced different slices of the matrices and projected them in the form of interoperability landscapes. These landscapes were graphic chart depictions. They provided a means to describe the characteristics of each matrix slice in terms of the way that events on one axis in the chart were related to each other through the activities on the other axis.

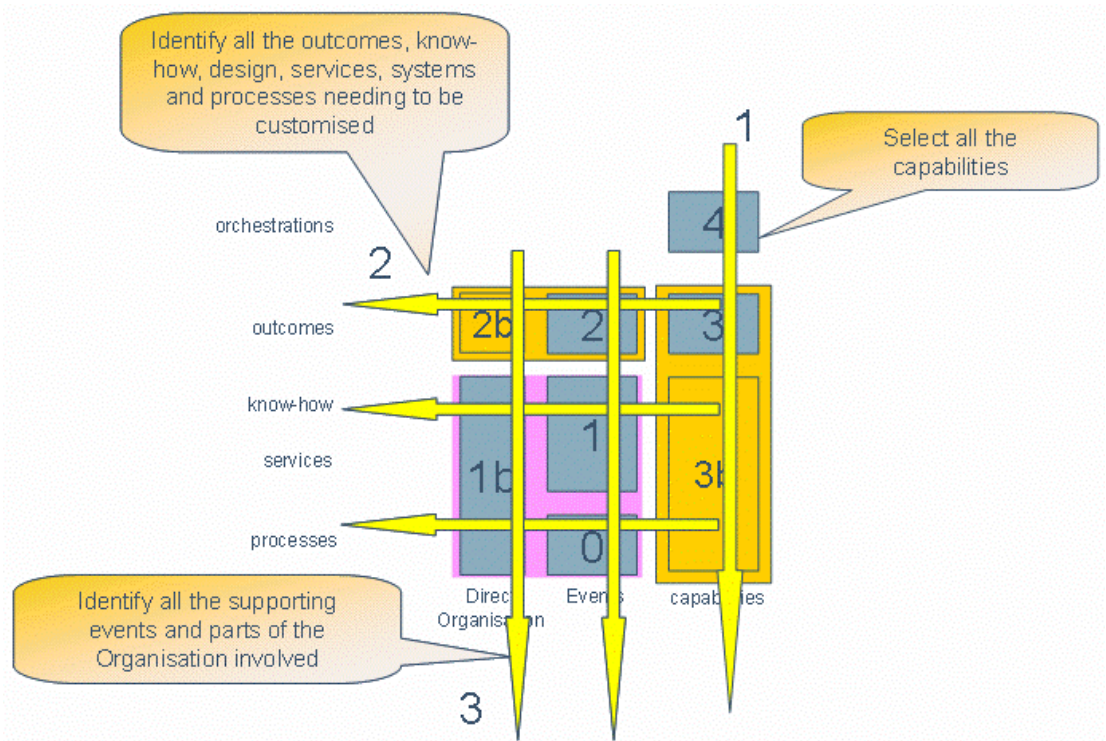


Figure 14: *Projecting Through the Matrix*

Figure 14 conceptually illustrates the general projection process.⁵ One “enters” the matrices from a point of investigation. In Figure 14, we would follow these steps to investigate the capabilities that provide orchestration support (see the callouts numbered 1, 2, and 3):

1. In Step 1, we would select all the capabilities named at Level 4 and “project” them downward, finding rows with entities that are associated with the capabilities. They would be in matrix sections labeled 3 and 3b.
2. Then, in Step 2, we would project the entities discovered in Step 1 left (across the rows) to identify columns with a secondary association.
3. Finally, in Step 3, we would project those columns downward to identify all the entities that have a tertiary association.

The entities thus identified would then be sorted by counting the various associations (i.e., connections) and rank-ordered to identify commonalities and differences in their levels of connectedness.

Figure 15 shows a landscape resulting from a projection. The columns in the landscape are organized so that entities connected to the same number of entities are next to each other in terms of their height and depth dimensions.

- The height dimension (q in the landscape) describes the number of shared underlying activities; the higher the q between columns, the more related they are.

⁵ The analysis and composition technique applied can be described in terms of the properties of hyperbolic quaternions. For more information on hyperbolic quaternion, see http://en.wikipedia.org/wiki/hyperbolic_quaternion [Wikimedia 2006].

- The depth dimension (**k**) describes the number of other related columns there are at that level of **q**.

The resulting landscape shows peaks separated by valleys. These valleys illustrate the gaps between the different levels of shared activity. (Note: The axis definitions would be changed depending on the particular projection that is rendered.)

Interoperability landscapes enabled us to visualize the relationships and gaps within the models, viewed from different perspectives that are codified by the matrices. The matrices used to generate a given landscape detail what is being grouped together. From Figure 15 and its underlying matrices, for instance, you can look up the 17 relationships that generate the high peaks or the 10 events that share services in the broad plateau on the left of the figure.

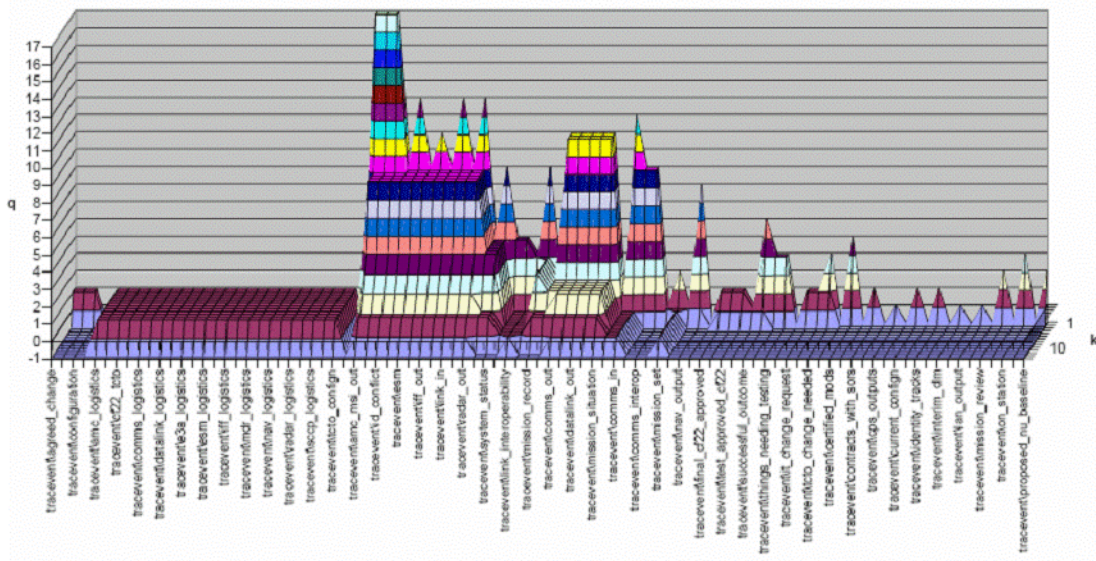


Figure 15: An Example Interoperability Landscape

Observer Note: The projection of the matrices into profiles is facilitated by an automated tool that also was not examined during this engagement. As with Prolog PAN, this tool would be an area for further investigation.

The probe in general was well received by the client. It is difficult to specify value per process stage. Stage one enabled the needed data collection; stages two and three facilitated the analysis of the data; and the stage three landscapes helped to convey the message. The landscapes stimulated conversation and gave empirical evidence for the conclusions drawn.

3 Outcomes from the Approach

Using the approach described in Section 2, we constructed models of the people, processes, and technologies that make up the program and represent the way demands are placed on their use. Using those models and representations, we developed an objective view from which we derived matrices that reflected the major interoperability challenges faced by the program. We categorized those challenges as follows:

- Type III Mission Risks (Section 3.1)
- Type II Composition Risks (Section 3.2)
- Type I Performance Risks (Section 3.3)
- Type 0 Constructive Risks (Section 3.4)

In the following sections, we describe each of these categories, posing and discussing its primary question.

3.1 TYPE III MISSION RISKS

Would the system of systems function within its operational context-of-use in the ways demanded of it?

To answer the Type III risk question, we selected the missions identified in the level 6 (the mission environments level) matrix and projected them downward to identify the mission synchronization needs. (See the callout numbered 1 in Figure 16.) We then projected those needs left across the other matrices (callout 2) and finally downward to identify those parts of the organization and infrastructure that are needed to support the mission demands (callout 3). We ordered the entities identified by the projection steps and ranked them according to counts of their mutual connectedness or shared interfaces.

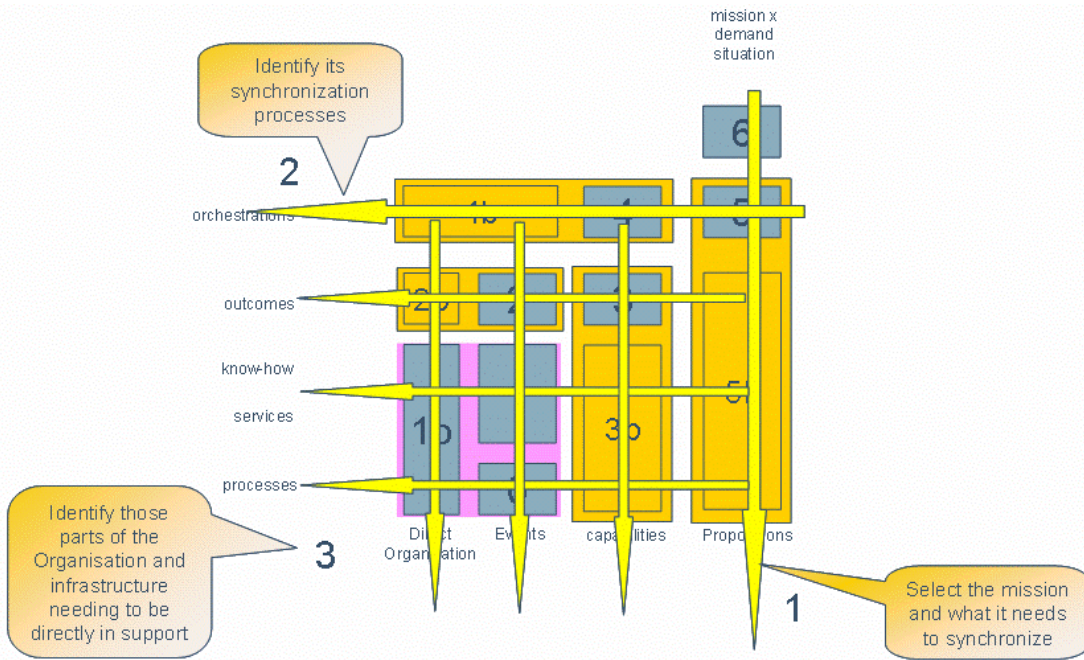


Figure 16: Mission Demand Projection

A three-dimensional depiction (from Excel) was our Mission Awareness Landscape, Figure 17. This particular example shows that the predominant mission awareness integration point is the system operator and the operator’s display console. The rest of the systems for areas such as development, support, and acquisition are virtually unaware of mission-demand complexity. (That is, these systems do not interoperate in response to demand situations. If they should, type III risk is high.⁶)

⁶ If the consequence of the detected condition is not serious (i.e., benign), the risk may be considered low [INCOSE 2004, Figure 6-1].

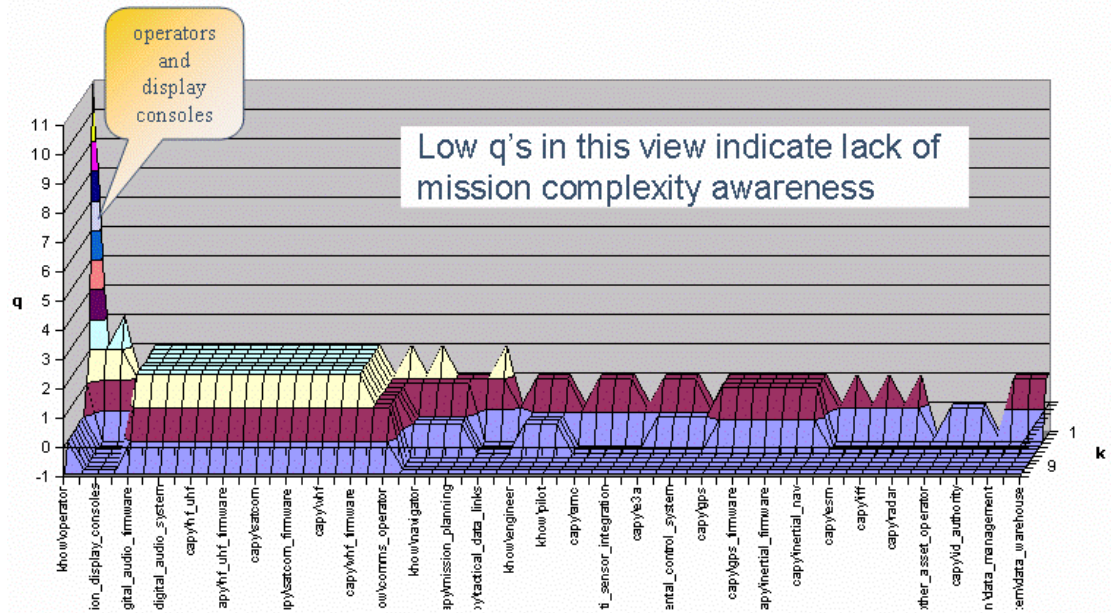


Figure 17: Mission Awareness Landscape

3.2 TYPE II COMPOSITION RISKS

Would the set of systems that need to interoperate within a given system of systems interoperate in the ways being demanded of them?

To answer the Type II risk question, we entered the matrices at level 4, our orchestration level. We projected capabilities downward (callout 1 in Figure 18), across to identify entities connected to the capabilities (callout 2), and then downward to identify the supporting entities needed (callout 3).

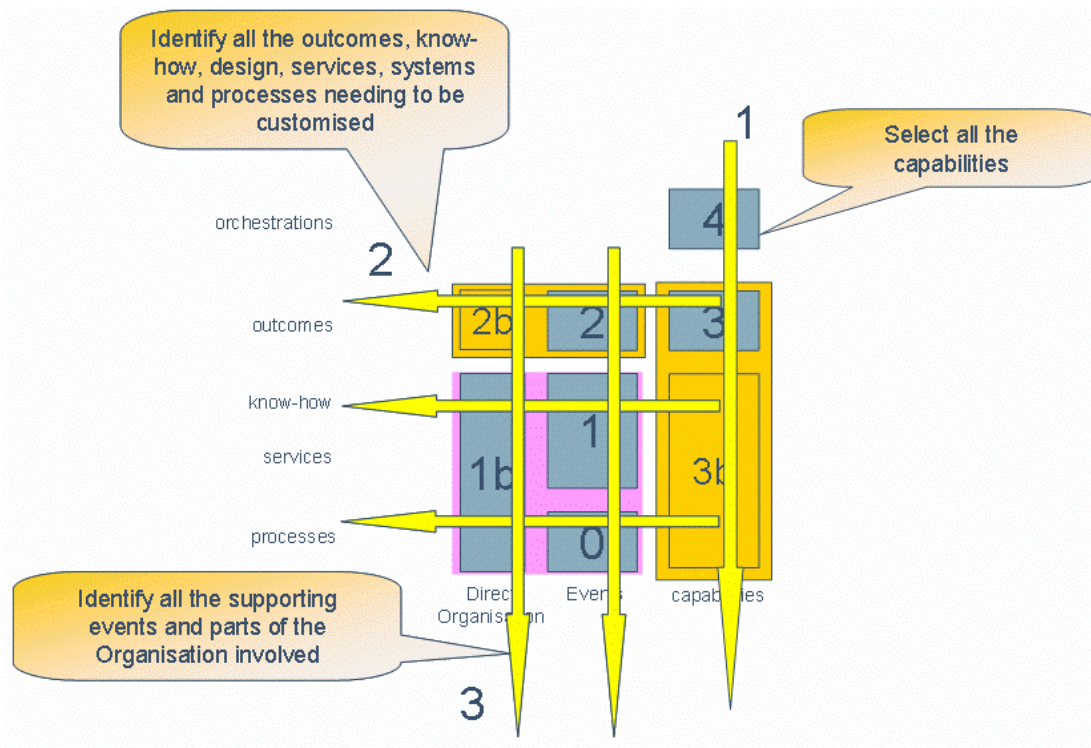


Figure 18: Projecting for Orchestration Risk

After ordering and ranking, the resulting Orchestration Landscape (see Figure 19) revealed obvious islands of high connectivity with broad regions of separation. In practice, the specific entity groupings would be examined to determine if the separation were warranted. For example, hardware configuration management was quite separate and poorly “orchestrated” with software version management, as the gaps show. The depth of the valleys indicates that the baseline “connective tissue” of aspects such as change management and revision control is far from seamless in this system of systems. The model (at the modeled fidelity) is good at indicating missing connections; it conversely indicates presences of connections (peaks) but does not speak to the sufficiency of those connections. Therefore, gaps tend to be truer signs of risks (it’s hard to interoperate when one has no connection) than peaks are guarantees of interoperability (because high connectivity does not necessarily mean interoperability). Both gaps and peaks are good pointers for further investigation, however.

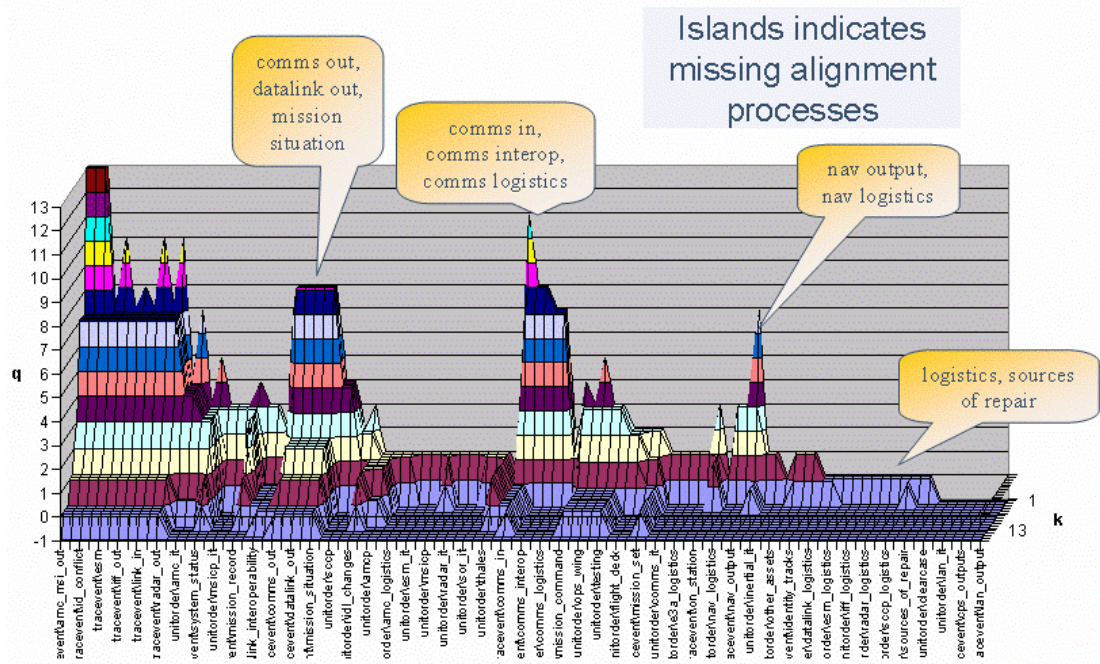


Figure 19: Orchestration Landscape

3.3 TYPE I PERFORMANCE RISKS

Would the subsystems within system elements interoperate to respond to demand?

We entered the matrices at a lower point, level 3, to answer the Type 1 risk question. At level 3, we are identifying the activities supporting the operational capability. We followed the same selection and projection steps (see Figure 20, callouts 1, 2, and 3) to produce risk entities and associations.

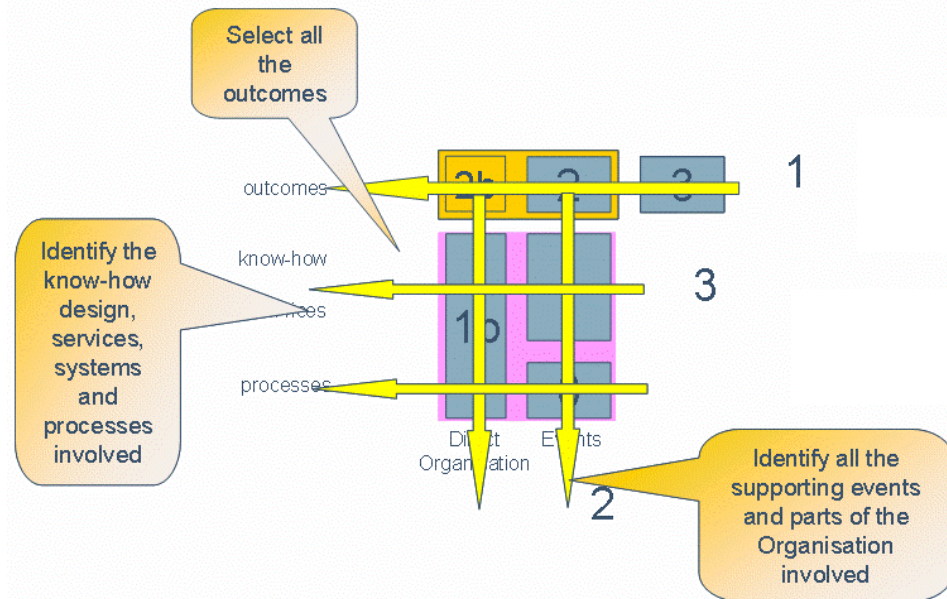


Figure 20: Performance Risk Projection

Our Performance Risk Landscape, shown in Figure 21, reveals the degree of isolation between the many structural entities in this system of systems. Once again we can predict high likelihood of connectivity gaps (call them potential risks if you like); these gaps require further examination to determine the severity of consequences before declaring specific high risks.

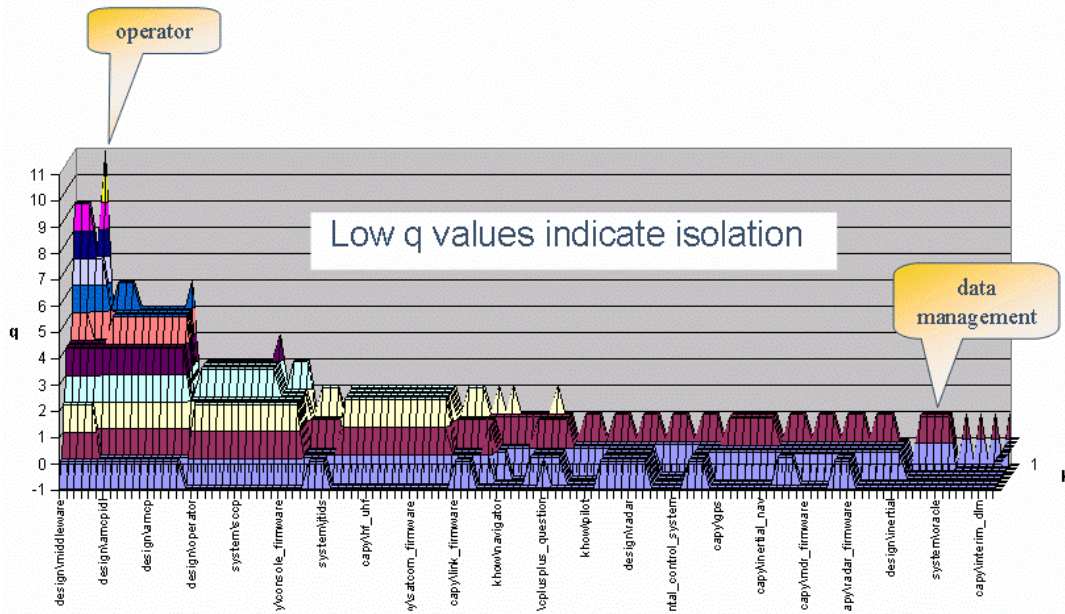


Figure 21: Performance Risk Landscape

The projection process can be customized and creatively applied. For example, when the first-line management structures are suppressed, the impact of indirect management control jumps out. Figure 22 vividly shows the vertical command dependencies in this system of systems. It also reveals the significant separation between the acquisition organization and the line command structures. (The dark area to the left of the landscape is the list of events; it is obscured here because of the size of our figure.)

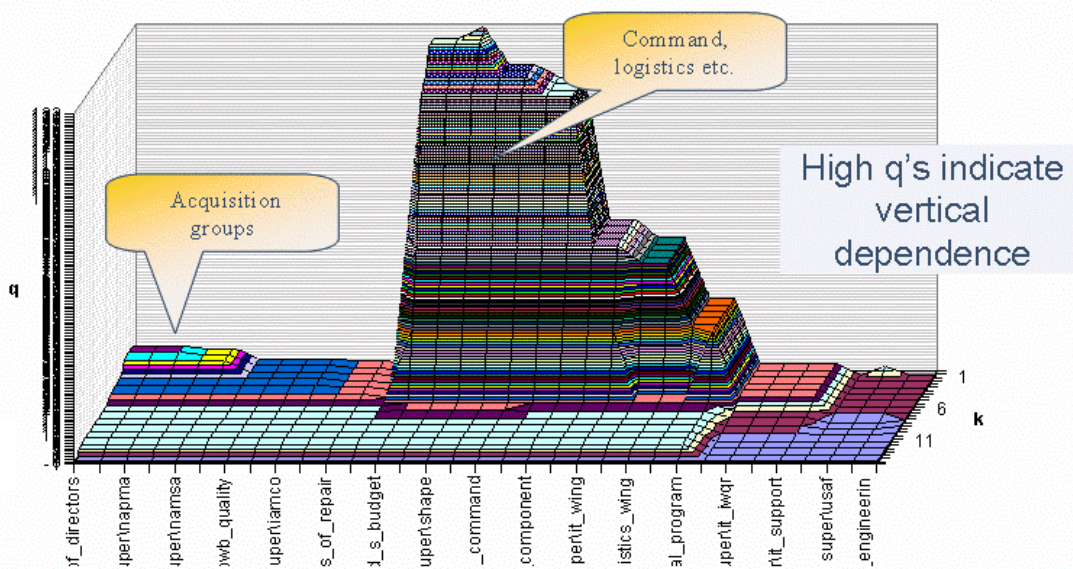


Figure 22: Indirect Management Control

3.4 TYPE 0 CONSTRUCTIVE RISKS

Would the constructed capability be able to perform according to its original design specification?

The Type 0 risks are typically identified by the program's internal risk management procedures. The risks that those procedures reveal might include some interoperability factors; more often, they focus on the "grassroots" issues of making the baseline products function properly. These risks were not illustrated through the model in this probe. But they were collected as issues raised during the interactive model development process and subject matter expert interviews.

3.5 SUMMARY OF PROBE OUTCOMES

In our four categories, we identified interoperability risks and visually reinforced their presence by the landscape topologies. The objects and relationships depicted in the landscapes were familiar to the client and served to facilitate constructive dialog about mitigation strategy.

Observer Note: The examination of interoperability is a challenge in understanding complexity. The structural models produced by PAN bring a welcome engineering viewpoint to the process. While a significant level of subjective interpretation is still brought to bear when examining the landscapes, one can easily imagine that, given sufficient case samples, repetitive patterns will be discernable and identified. These patterns or archetypes will carry specific interpretations in terms of risks, costs, and mitigations.

The techniques as observed in this engagement produced a rapid (nominally two days per model) snapshot of the interoperability risks from the perspective of the interviewed stakeholders. The models helped to justify and prioritize follow-on activities, such as detailed impact analysis, model refinement and validation (through detailed, bottom-up fact finding), and cost analysis in targeted areas.

4 Observer Summary Conclusions

The PAN technique starts with a client-driven context and builds visual models that are very easily understood by, and bring immediate value to, the client. The technique stresses the need to speak in the client's language. The client artifact-based starting point was useful; it gave familiar ground for the client participants and facilitated the analysts' learning curve. Similarly, the four-color rubric was well accepted, perhaps due to this client's familiarity with war gaming. (Most of the stakeholders were active or retired military personnel.)

However, as we have noted in this report, these rather classic entity relationship style diagrams quickly became "eye charts" that were too complex to convey anything other than the global impression of complexity. The rules of object-connector relationships brought some structural rigor to this problem, facilitating a transformation of the data into matrices that support empirical analysis.

The integration of demand dynamics into the structural model not only was a key conceptual shift but also gave the client a face-saving externality on which to deflect accountability. The operational environment had changed; the challenge is to adapt to this new environment.

Of greater merit in the probe was the explicit attention paid to understanding the relationship of the operational context and the supplied technologies, capabilities, and governance mechanisms.⁷ By identifying gaps in their alignment, we identified critical risks that are often overlooked.

The palette of objects and connectors was adequate to represent the structures investigated. The model was a structural snapshot. Although dynamic stocks, sinks, or source constructs⁸ were represented, the dynamic characteristics of their behaviors were not.

The modeler is currently a key component of this technique. The modeler's task is simpler than that of a systems-thinking or system-dynamic modeler.⁹ The PAN modeler is less concerned with temporal aspects—while reinforcing or negative feedback loops are distinguished structurally, no attempt is made to dynamically represent their behavior over time. The modeler must possess expert Visio operating skills and an ability to grasp quickly and characterize the constructs and object relationships during the interactive building of the model.

It is difficult to judge how well these skills can be transitioned. We do know that the practitioner observed has many years of experience with the tools, constructs, and objects used. However, these models are somewhat analogous to process flow diagrams in the level of abstraction required by the client. (They are more complex in the aggregate due to the multiple layers of structure. But using the layering capabilities of Visio mitigates that difference.) Therefore, we infer

⁷ PAN models the *structure-determining* processes of the organization-in-context as well as the *structure-determined* processes of the systems the organization uses.

⁸ These are constructs often used in other dynamic modeling techniques.

⁹ A forthcoming report, CMU/SEI-2006-SR-018, explores the distinctions among Systems Thinking, System Dynamics, and PAN.

that an experienced process modeler would have little difficulty adapting to this modeling paradigm.¹⁰

The niche character of this technique is good and bad. PAN offers a fresh, unique, approach—as well as new insights and mechanisms—to study complexity. Its use could uniquely position the Carnegie Mellon[®] Software Engineering Institute to analyze systems of systems in new ways. However, it does not yet enjoy a broad user base, cross-domain application, or commercial grade tool set. It is not a “whole product” as yet and is therefore not suitable for end-user self help. It is at a stage of development that is appropriate for application as an internal, custom tool suite for specially trained complex-system analysts.

We recommend continued exploration of this technique. The potential for application and amplification of this body of knowledge appears to be significant in the system-of-systems area and possibly beyond.

¹⁰ By contrast, the significantly different conceptual content in System Dynamic modeling makes its adoption more problematic.

[®] Carnegie Mellon is registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.

References

[INCOSE 2004, Figure 6-1]

International Council on Systems Engineering. *INCOSE Systems Engineering Handbook: A “What to” Guide for All SE Practitioners* (INCOSE-TP-2003-016-02, Version 2a, June 2004).
http://www.incose.org/delvalley/data/se_hdbk_v2a.pdf.

[Wikimedia 2006]

Wikimedia Foundation. “Hyperbolic Quaternion.” *Wikipedia, The Free Encyclopedia*.
http://en.wikipedia.org/wiki/Hyperbolic_quaternion (2006).

| REPORT DOCUMENTATION PAGE | | | <i>Form Approved OMB No. 0704-0188</i> | |
|--|--|---|---|--|
| Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503. | | | | |
| 1. AGENCY USE ONLY (Leave Blank) | 2. REPORT DATE October 2006 | 3. REPORT TYPE AND DATES COVERED Final | | |
| 4. TITLE AND SUBTITLE An Examination of a Structural Modeling Risk Probe Technique | | 5. FUNDING NUMBERS FA8721-05-C-0003 | | |
| 6. AUTHOR(S) William Anderson, Phillip Boxer, Lisa Brownsword | | | | |
| 7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Software Engineering Institute Carnegie Mellon University Pittsburgh, PA 15213 | | | 8. PERFORMING ORGANIZATION REPORT NUMBER CMU/SEI-2006-SR-017 | |
| 9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) HQ ESC/XPK 5 Eglin Street Hanscom AFB, MA 01731-2116 | | | 10. SPONSORING/MONITORING AGENCY REPORT NUMBER | |
| 11. SUPPLEMENTARY NOTES | | | | |
| 12A DISTRIBUTION/AVAILABILITY STATEMENT Unclassified/Unlimited, DTIC, NTIS | | | 12B DISTRIBUTION CODE | |
| 13. ABSTRACT (MAXIMUM 200 WORDS) The integration of demand dynamics into a structural model is a key conceptual shift for software engineering. This report examines the utility and transition characteristics of a structural dynamic analysis modeling technique called Projective Analysis (PAN) that was used on an interoperability technical probe of a North Atlantic Treaty Organization (NATO) modernization program. The report focuses on the process, rather than the findings, of the probe. Organizational entities are referred to generically and, in some instances, aggregated. The probe involved workshops and interviews, conducted over a two-week period with more than 25 people, followed by analysis of the data gathered. PAN was used to model the NATO program as a system of systems. The model is a rapid assessment based on the subjective understanding of the interviewed subject matter experts. It is a snapshot in time; while dynamic stocks and feedback loops are represented, their temporal characteristics are not. From the model, five perspectives were analyzed for different forms of interoperability risk. These analyses produced three-dimensional projections that depict clusters of shared interfaces. The separation between these clusters identifies the interoperability risks. The report notes that the PAN technique starts from a client-driven context and builds visual representations that are easily understood by, and bring immediate value to, the client. Further, the report observes that the modeler is critical to this technique and must possess expert skills in the Microsoft Visio application as well as an ability to quickly grasp and characterize the constructs and objects revealed through dialog-based inquiry. The report concludes that PAN appears to offer a fresh approach, new insights, and appropriate mechanisms to study complexity in systems of systems. The potential for applying and amplifying this technique appears to be significant. The report also determines that an experienced process modeler would have little difficulty adapting to this modeling paradigm. | | | | |
| 14. SUBJECT TERMS Projective analysis, PAN, system of systems, interoperability, risk driver, SoS | | | 15. NUMBER OF PAGES 38 | |
| 16. PRICE CODE | | | | |
| 17. SECURITY CLASSIFICATION OF REPORT Unclassified | 18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified | 19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified | 20. LIMITATION OF ABSTRACT UL | |