Proceedings of the System of Systems Interoperability Workshop (February 2003)

Linda Levine
B. Craig Meyers
Ed Morris
Patrick R. H. Place
Daniel Plakosh

June 2003

System of Systems Interoperability Internal Research and Development Project

Technical Note
CMU/SEI-2003-TN-016

The Software Engineering Institute is a federally funded research and development center sponsored by the U.S. Department of Defense.

This work was created in the performance of Federal Government Contract Number F19628-00-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.

Contents

List of Figures

Abstract

The Software Engineering Institute has initiated an internal research and development effort to investigate interoperability between systems. As part of the research, a workshop was held in February 2003 with an advisory board of Department of Defense experts. A preliminary model of interoperability was presented and feedback on the model was requested. This technical note documents the model of interoperability presented and the findings from the workshop.

1 Introduction

The Software Engineering Institute (SEI[SM]) has initiated an internal research and development (IR&D) effort to investigate interoperability between systems. As part of the research, a workshop was held in February 2003 with an advisory board of Department of Defense (DoD) experts. A preliminary model of interoperability was presented and feedback on the model was requested in the following areas:

- critical interoperability issues
- insight into programs that are solving critical interoperability problems
- best approaches for conducting research on the current state of the practice

This technical note documents the model of interoperability that was presented and the findings from the workshop.[1]

1.1 Background

The importance of interoperability cannot be denied. Interoperability to achieve information superiority is the keystone on which future combat systems (e.g., Air Operations Center, Future Combat Systems), logistic systems (e.g., Global Combat Support System), and other government systems (e.g., interoperability between organizations for homeland security) will be constructed. *Joint Vision 2020* [Joint 00], which guides the continuing transformation of America's armed forces, states, "Interoperability is the foundation of effective joint, multinational, and interagency operations."

Currently, there is a tendency to concentrate on the mechanisms used by the various systems to interoperate. However, concentrating on mechanisms misses a larger problem. As *Joint Vision 2020* suggests, to create and maintain interoperable systems of systems, there is a need for interoperation not only at the operational level, but also at the level of their construction and program management. Improved interoperation will not happen by accident and will require changes at many levels.

While many systems produced by Department of Defense (DoD) programs can, in fact, interoperate with varying degrees of success, the manner in which this interoperation is achieved is piecemeal. In the worst case, interoperability is achieved by manually entering

[SM] SEI is a service mark of Carnegie Mellon University.

[1] The SEI Acquisition Support Program contributed to the capture of the knowledge from the workshop.

data produced by one system into another–a time consuming and error-prone process. Clearly, if America's armed forces are to achieve *Joint Vision 2020*, and if cross-organizational homeland security capabilities are to be developed, a better way forward must be found:

> Although technical interoperability is essential, it is not sufficient to ensure effective operations. There must be a suitable focus on procedural and organizational elements, and decision makers at all levels must understand each other's capabilities and constraints. Training and education, experience and exercises, cooperative planning, and skilled liaison at all levels of the joint force will not only overcome the barriers of organizational culture and differing priorities, but will teach members of the joint team to appreciate the full range of Service capabilities available to them [Joint 00].

1.2 Approach

It is important to focus on current practice to determine the interoperability problems being encountered. It is also vitally important to consider promising acquisition and management approaches as well as technologies that can be applied to the problems. Thus, the IR&D will investigate current practice and identify promising research and innovative solutions.

As part of our vision, we foresee a future state where a theory of interoperability is used to inform decisions made during the entire acquisition life cycle. As a step toward that vision, our study will develop an initial informal model of interoperability. The model will help to identify gaps where current solutions and research do not address existing problems, thus suggesting areas of work to be initiated within the research community. The model will also help the team to determine where current research may replace existing solutions, leading to more efficient creation and evolution of interoperable systems of systems. The gaps in the solution space will help determine the plan for future SEI work in interoperability.

2 A Model of Interoperability

In many cases where systems are interoperating with other systems, this interoperation is achieved through heroic effort and at great expense. Too often, the approaches used lead to interoperability that is specific to the targeted systems (sometimes called "point-to-point" integration) and that does not facilitate extension to other systems. Even then, the technical approaches employed, such as Defense Information Initiative Common Operating Environment (DII/COE) and Extensible Markup Language (XML), offer only partial interoperability.

Achieving large-scale and consistent interoperation among systems will require a consistently applied set of management, technical, and operational practices that support the addition of new and upgraded systems to a growing interoperability web. Improvements in technology alone (whether XML or any other) will not be sufficient. There must be parallel improvements in the ways that current and future interoperability needs are identified, and how organizations pursue interoperability.

A simple model depicts the broad range of activities that are necessary to achieve interoperability (Figure 1).

Figure 1: System Activities Model

As shown in Figure 1, *Program Management* defines the activities that manage the acquisition of a system. *System Construction* defines the activities that create or evolve a system (e.g., use of standards and COTS products, architecture). *Operational System* defines the activities within the executing system and between the executing system and its environment, including the interoperation with other systems. The end user is considered part of the operational system.

The simple model represents the activities within a single acquisition program. As illustrated in Figure 2, interoperability between multiple systems requires the development of interoperability between Program Management Offices (PMOs) and the systems they control. This interoperability will be achieved most effectively through interoperation at the program management, system construction, and operational levels. Each dimension represents a type of interoperability that is discussed in greater detail below. In each case, an example is provided.

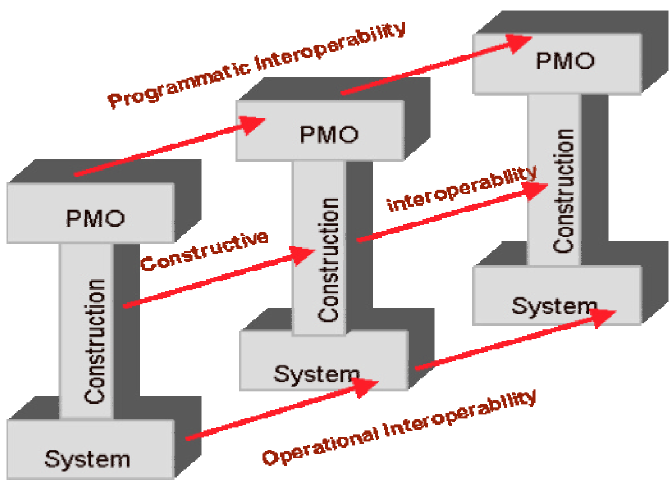


Figure 2: Different Types of Interoperability

2.1 Programmatic Interoperability

Programmatic interoperability encompasses the activities related to the management of one program in the context of another program. Typically, programs are managed in comfortable isolation, with little need to consider the management functions of other programs that are producing "interoperable" systems. As a result, interoperability is typically defined by a common specification and is accomplished through the efforts of technical personnel. Because of the limitations of specifications and the lack of incentives for programs to probe beyond them, the resulting interoperability is often less than what is desired.

To remedy this situation, management approaches and techniques that bridge the gaps between the isolated programs and perspectives are needed. Some candidate strategies for achieving programmatic interoperability include

- synchronization of schedules and budgets
- joint risk management

- coupled award fee boards
- linked promotions

Achieving programmatic interoperability in a system of systems context demands consistent, joint risk management (for both hardware and software) among the PMOs. Risk will be shared, and distributed, across programs, and failure to understand risk in the larger context will only perpetuate a stovepipe management perspective.

Example: A program is building a large distributed combat system. They are integrating many subsystems, most provided by other programs. As part of their system test, they require a simulator, which is developed by another program. The simulator is delivered late, and it does not implement the interface as expected. The simulator provider identified problems with the interoperability approach, made reasonable technical decisions to resolve these problems, but did not communicate these changes to others. No mechanisms were in place to support joint risk management, and poor communication between programs exacerbated the problem.

2.2 Constructive Interoperability

Constructive interoperability addresses those activities related to construction and maintenance of one system in the context of another system. Constructive interoperability includes the common use of architecture, standards, data specifications, communication protocols, languages, and COTS products to build interoperable systems. Two factors limit current conceptions of constructive interoperability. First, a common assumption holds that attending to such mechanisms (architecture, standards, etc.) alone will result in interoperability between systems. Second, these same mechanisms capture only part of the rich semantics that must be shared for the interaction of sophisticated systems.

New mechanisms (e.g., XML) are being developed that provide richer semantics than current approaches. Without doubt they will be useful, but they are also likely to prove insufficient, because the vision for interoperability will always exceed what is technically possible.

Achieving constructive interoperability demands new perspectives on the use of standards and software architectures. It is naïve to assume that simply using a standard will guarantee system interoperability. Joint definition of the standards and system of systems architecture will provide a critical aspect for each system's construction.

Example: Two systems are being built using industry standard object request brokers provided by different COTS vendors. The two program offices assumed that conformance to an industry standard would ensure interoperability. Unfortunately, one vendor added unique features to the product that extended the standard. These unique features were used during construction of one of the systems, making it impossible for the two systems to interoperate without rework to one or both systems.

2.3 Operational Interoperability

Operational interoperability refers to the activities related to the operation of a system in the context of other systems. These activities include

- doctrine governing the way the system is used
- conventions for how the user interprets information derived from interoperating systems (i.e., the semantics of interoperation)
- strategies for training personnel in the use of interoperating systems

Achieving operational interoperability demands imposition of requirements in the larger system of systems context. The software associated with each individual system must satisfy many of these requirements. For example, if there is a requirement to distribute doctrine among autonomous vehicles, then the software in those vehicles must be capable of satisfying the larger context of requirements. In this case, the distributed doctrine is now *shared* among individual systems, but it must be shared in such a manner that interoperability can be achieved.

Example: Multiple combat platforms are exchanging data over different communication links. Each type of link places different restrictions on the data. As a result, different users are receiving different views of the battle environment. This problem at the constructive level of interoperability also has implications at the operational level for establishing conventions for *how* the user interprets the data.

2.4 Interoperability Backplanes

The vision of interoperability between PMOs is extensible to a system of systems that crosses Program Executive Office (PEO) boundaries through the use of programmatic, constructive, and operational backplanes. These backplanes define a consistent set of practices and techniques leading to successful interoperation that can be employed for any number of systems and PEOs (Figure 3).

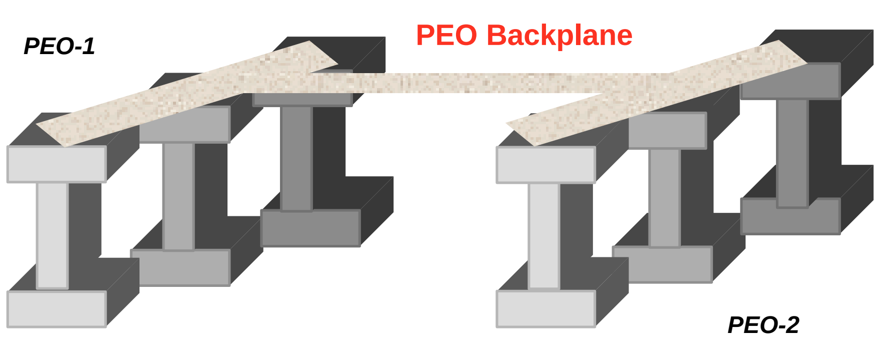


Figure 3: Interoperability Backplanes

To achieve these interoperability backplanes, each system of systems must be viewed as a unit. This view is in contrast to current practice where the program manager's incentive is tied to a particular system, with minimal attention paid to the wider context in which the system will reside. Activities that we take for granted as part of system development (e.g., requirements management, architecture definition, development and management processes, and definition of the operational semantics) must also be performed at the system-of-systems level.

3 Workshop Findings

The workshop was held on a snowy day in February in the Washington, D.C. area. The heavy blanket of snow reduced attendance at the workshop, but the brave souls who ventured out engaged in a lively and informative discussion of the SEI model and related topics. The major themes of the workshop are captured in the following sections.

3.1 Definition and Interpretation of Terms

Workshop attendees agreed that individuals and groups have different interpretations of the meanings of terms such as *system of systems* and *interoperability*, based on their divergent needs. Regarding the former, we were advised to make sure we define our use of terms and recognize a need for education, because, "What someone considers to be a system of systems, someone else considers a system."

There is a similar need for precise definition of interoperability, because the term can have various meanings in different contexts. For example, interoperability with a weather system may be defined as hourly updates. This could conceivably be handled by a phone call. In contrast, radar reports of objects in the environment need to be updated far more frequently. Thus, interoperability between interacting radar systems may require automated updates as frequently as every few seconds.

The workshop attendees recognized that we may never have precise definitions, partly because our expectations for interoperability are constantly changing. New capabilities and functions offer new opportunities for interactions between systems. This continual change is one of the reasons it is so difficult to define interoperability precisely. Effectively communicating what is meant by interoperability will demand that the context, operations, and requirements be specified whenever the term is used.

3.2 Alternative Views of Interoperability

3.2.1 People-Centered View of Interoperability

There was a clear recommendation that the SEI present its message from the standpoint of the end users of interoperable systems. One workshop attendee stated, "Try something bottom up that represents the end users (aircraft, soldiers, subs, tanks). Look at what needs to be there to make the required interoperability happen." Other attendees stated, "What capability do you want for the armed forces? Put the customer first.” “Build a model of interoperability, then derive how to do it. The problem is communication among people.”

The workshop attendees wanted to avoid the following problem: “I have seen programs stopped. [The system] passed testing but wasn’t what the user needed. They spent seven years doing this. How dumb is that?”

Even though our current model has an operational dimension, it was regarded as “top down” because it presented information from the standpoint of an acquisition program constructing the system. While the intention of the model was to include end-user protocols and procedures, the attendees believed that the model would be better received if it was presented from the perspective of the end user.

3.2.2 Life-Cycle View of Interoperability

Attendees also suggested that we consider the specific activities that must occur in each life-cycle phase to achieve interoperability. In keeping with a people-centered perspective, the life cycle must be extended to include “training, fielding, and the people in your view of the whole system.” This perspective focuses on designing for interoperability at an early stage and maintaining that focus throughout.

This perspective also takes a broad view of the system. “What is the whole? Is it the sum of [the individual system] parts? It is more than the sum of the [individual system] parts.” “. . . maneuver and artillery requirements may be separate in development, but in the field they interact.”

The current acquisition model is based on an interaction between end users who provide requirements to the acquisition community; the acquisition community then contracts for a solution that meets the end-user needs. There is little ongoing interaction between the contractor and the customer. In contrast, an approach that provides ongoing interaction with the customer runs the risk that the customer will continuously change requirements, thereby affecting acquisition cost and schedule. There are advantages and disadvantages associated with both approaches.

3.3 Complexity

The attendees agreed that interoperability is a hard problem to solve. It is even more difficult than people think because any solution will require addressing organizational and technical issues. For example, achieving interoperability between two distinct systems will require changes to management planning and system implementation.

Attendees stated that very little is known about interoperability requirements at the start of a program. In some cases, the interoperating systems are not yet conceived. Thus, new strategies must be developed to anticipate future needs and cope with current uncertainty.

Some of the specific issues related to complexity are described below:

- Backwards compatibility: Maintaining compatibility with older systems sometimes conflicts with achieving greater levels of interoperability between newer systems. This conflict can lead to decisions to accept reduced interoperability between old and new systems. Unfunded mandates that force resources away from upgrading old systems, and making patches to old systems, can exacerbate the problem if interoperability is not considered.
- Angular compatibility: Interoperability between systems is sometimes specified in the following form:
  A is interoperable with B
  B is interoperable with C
  This does not imply that A will be interoperable with C, as sometimes inferred.
- Inconsistent standards: A number of attempts have been made to increase interoperability by developing standards and models for architecture (Joint Technical Architecture–JTA) and system components (Defense Information Initiative Common Operating Environment–DII/COE). These models and standards have been developed by different organizations with no consistent approach to achieving interoperability. Using the same standard can give a false sense of security and the models alone are insufficient for achieving interoperability.
- Contractors' processes: Interoperability is hindered by the size and diversity of the systems built and the number of contractors necessary to build those systems. Processes have not been established between contractors to guarantee the required level of interoperability. Further, "even with one contractor, we must [define] some processes – you will still have this need."
- Ambiguous terminology: Differences in the use of terms across organizations can be troublesome. The terms used are sometimes mutually exclusive or conflicting. This ambiguity extends down to the operational level. For example, American armed forces use different terms for cease fire (e.g., hold fire, weapons hold).
- Rules of engagement and doctrine: Consideration of the operational context must also address the way that a system is used. This is often described in terms of "rules of engagement" or doctrine. In development controlled by one acquisition organization, all information–including doctrine–is controlled in the user-acquisition context. However, when we address interoperability among multiple systems, doctrine also must be interoperable.

3.4 Communication

Interoperability depends on the quality of communication within and between organizations. This communication must occur at the level of systems of systems, but acquisitions are typically organized around individual systems. Intra- and inter-organizational communication is complicated by a lack of

- understanding who to communicate with
- methodology for how to communicate ("We need an easy method to get information in and out.")
- early specification of interoperation ("I built a great little system. I was told to do that piece. I wasn't told about the interface.")
- incentive to pursue interoperability when it makes the work more complex and creates internal and external dependencies that can affect the program

It has proven very difficult to improve communications flow and break down stovepipes. "What capabilities do we in the Air Force need between now and the future? We chose critical areas, we did the architecture work, and we are doing analysis of other programs doing upgrades. We are doing cost trades. We have taken money from one program and put it in another. However, we are still talking about stovepipes. We are trying as best we can to break down the stovepipes."

3.5 Funding and Control

Program staff is often inexperienced in estimating the costs associated with interoperability. We are not aware of any guidelines for estimating the level of effort necessary to achieve a given level of interoperability. One attendee reported industry estimates placing the costs to build interoperable systems at 140% of the costs to build similar, non-interoperable systems. This estimate is based on additional resources needed to integrate the outputs of the various teams.

Attendees pointed out that interoperability is almost never funded, and reaching agreement between programs is dependent on money: "A key tenet of interoperability is who controls the funds." "PEOs are reluctant to collaborate because then they will have to share or give up some of their funding." Attendees made the following suggestions:

- Interoperability (including overhead) must be planned for, funded, and resourced.
- Contractors will need to receive incentives to tie a program's success or profit to another program's successes.
- The current funding paradigm will need to change in order to achieve success. "We get [the money] religion real quick. [After all we wouldn't] have done SIAP [Single Integrated Air Picture] without money."

Beyond money issues, program staff is reluctant to relinquish control for technical reasons. One attendee remarked, “I will be forced to change my perfect implementation … for your imperfect [implementation].” Another added, “Don’t take away my control of my stuff.”

3.6 Leadership, Direction, and Policy

A barrier to interoperability is a lack of centralized or coordinated ownership of the problem. Attendees complained of short-sighted decisions to promote a single system’s view at the expense of other systems. They also expressed a number of concerns about interoperability policy making. Some felt that policies were drafted in a vacuum without a full understanding of the problems and the people affected. They observed that “policy for policy’s sake is bad,” “policy needs to be sensitive to the implementer’s controls and constraints,” and “policy needs some flexibility.” The following additional comments were made about policy:

- Policy decisions often reflect only a single domain, whereas interoperability may be different in different domains, and it may require consideration of special constraints (e.g., environment, safety).
- Timing of policy implementation is critical. “Just because you put out a policy on interoperability does not mean that all the preexisting system or systems under development are going to be interoperable.”
- “Writing policy is the easy step. Implementing it is hard.”
- “Policy is sometimes used as a hammer. Others work around policy to be successful. Both can get things done.”
- Contractors sometimes prefer standards/policies like DII/COE or JTA because they are easier to satisfy (check the box instead of having to solve the interoperability problem). Some people understand the real interoperability problem, but they prefer not to acknowledge it because then they will have to provide a solution. In this case, policy can work against doing the right thing.

Finally, attendees suggested that while there are many policies, no one is collecting the data to determine whether they are effective. “Where is the traceability back to policy to determine whether policy is working?”

4 Implications

The interoperability model presented by the SEI was largely corroborated. The dimensions of programmatic, constructive, and operational interoperability were thought to reflect an appropriate theoretical construct for understanding interoperability. However, two strong recommendations emerged. The first concerned the need to ensure that people who will be using and maintaining interoperable systems receive greater attention. The second argues for taking a life-cycle perspective on interoperability and considering the specific activities that must occur in each life-cycle phase to achieve interoperability. While these issues are currently addressed in the model, they are easily overlooked and require greater emphasis.

During preparation of this report, several topics were identified that require further consideration in light of the various aspects of interoperability. These include

- integration of acquisition models
- implications of the demise of DoD 5000 documents
- potential organization of a PEO with respect to components, rather than systems
- approaches to joint risk management

These topics will be addressed as part of the IR&D work.

Workshop attendees also recommended that the SEI provide a forum where representatives from all the services can share information on their respective views of interoperability and the steps they are taking to address the problem. A second workshop was proposed and was held in May 2003.

Appendix A Workshop Presentation

Carnegie Mellon
Software Engineering Institute
Pittsburgh, PA 15213-3890

System of Systems Interoperability (SOSI)

Sponsored by the U.S. Department of Defense

page 1

Carnegie Mellon
Software Engineering Institute

SEI Background

The Software Engineering Institute is a federally funded research and development center (FFRDC) operated under contract with DoD by Carnegie Mellon University in Pittsburgh.

Our mission is to transition technology to DoD and its contractors in order to build systems better.

 Version 1.0 page 2

Carnegie Mellon
Software Engineering Institute

SOSI Motivation

Interoperability between systems has become increasingly important – no modern system stands alone.

- Future Combat System
- Air Operations Center
- Navy battle groups
- Joint battle groups

Interoperation problems can seriously limit the ability to perform operational missions.

Improved interoperation:

- will not happen by accident
- will require changes at many levels

 Version 1.0 page 3

Carnegie Mellon
Software Engineering Institute

SOSI Background

An independent research project has been started in the area of *system of systems interoperability.*

We have asked you here to help us define our future.

 Version 1.0 page 4

Carnegie Mellon
Software Engineering Institute

Agenda

Approach

Model Overview and Discussion

State of Practice Assessment

Future Steps

Summary

 Version 1.0 page 5

Carnegie Mellon
Software Engineering Institute

Approach

The approach to the IR&D effort involves several aspects:

- Work with advisory board to set direction
- Assess state of the practice
- Identify promising areas for future work

 Version 1.0 page 6

Carnegie Mellon
Software Engineering Institute

Advisory Board

We want to interact with people from DoD to help us shape this work. We created an advisory board that includes the following

- Randy Blystone, NRO
- MGen. Thomas Brandt (Ret), SEI
- Col. Norris Connelly, AF-CIO/S
- Dr. Stan Levine, Technical Integration, G8
- Mr. Jim Linnehan, Army, ASALT
- Ms. Beth Lynch, Technical Integration G8
- Mr. Reuben Pitts, PEO IWS
- Dr. Jim Walbert, Army Program Objective Force
- CAPT. Jeff Wilson, SIAP

Carnegie Mellon
Software Engineering Institute

Goals and Expected Output of this Meeting

Today we would like to:

- pick your brains regarding critical interoperability issues
- get feedback on a model for system acquisition where interoperability is a key factor
- identify programs that are solving critical interoperability problems
- understand the right questions to ask in an assessment

Findings of the workshop will be documented in an SEI report.

Carnegie Mellon
Software Engineering Institute

Setting the Context

To help us understand the nature of the beast we are after, let's all try and complete the following metaphor:

Interoperability is like ________.

 Version 1.0 page 9

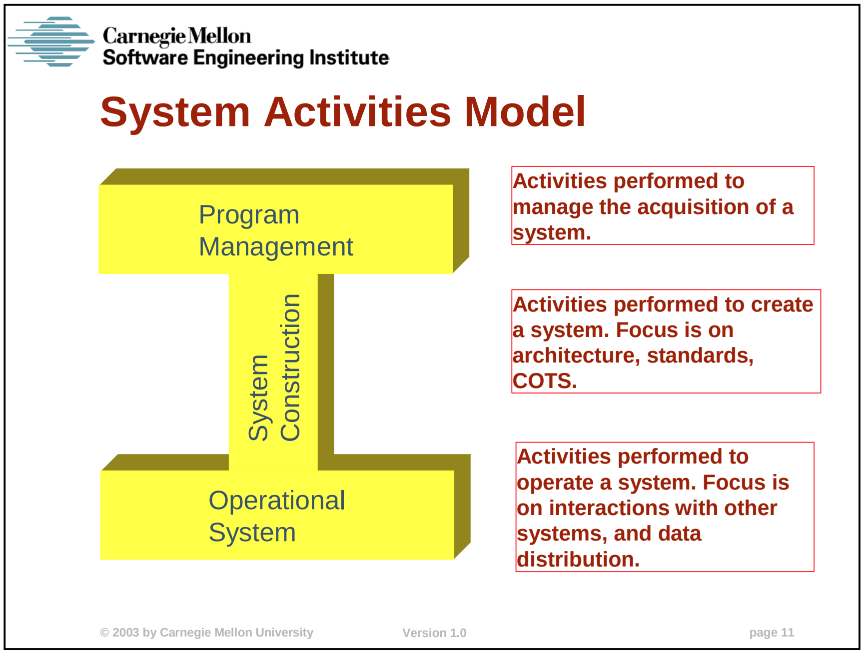
Carnegie Mellon
Software Engineering Institute

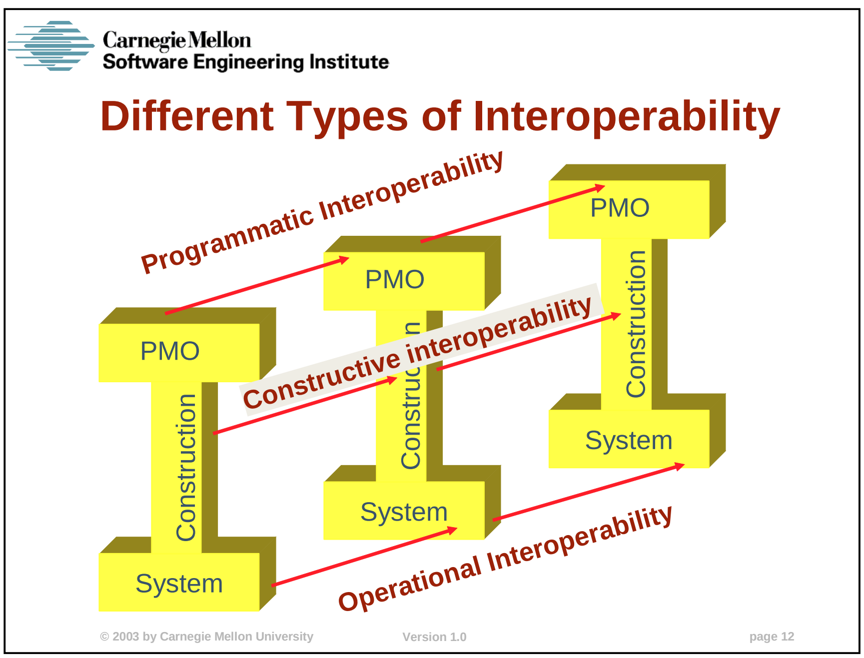
Model

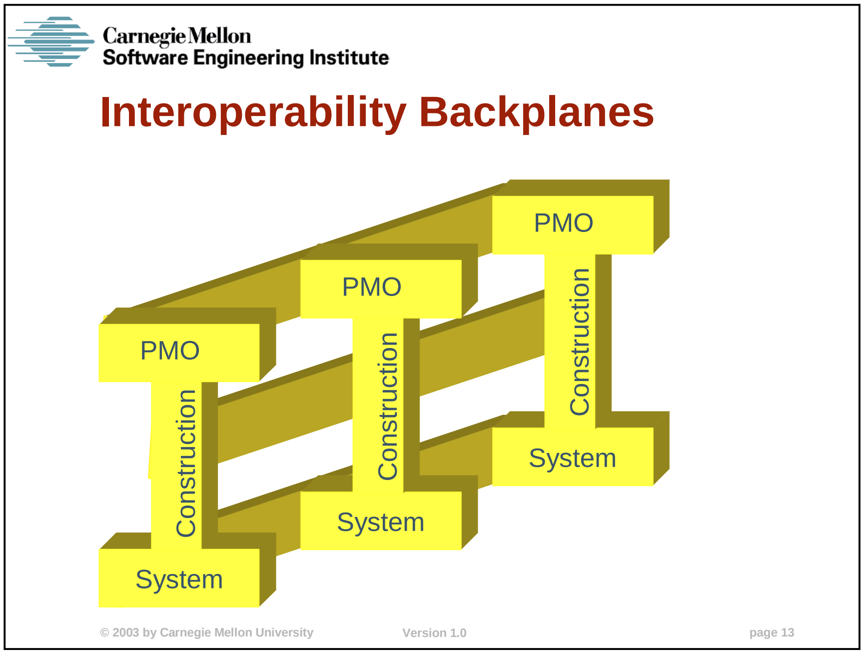
We have developed a model for describing interoperability at multiple organizational levels:

- Programmatic
- Constructive
- Operational

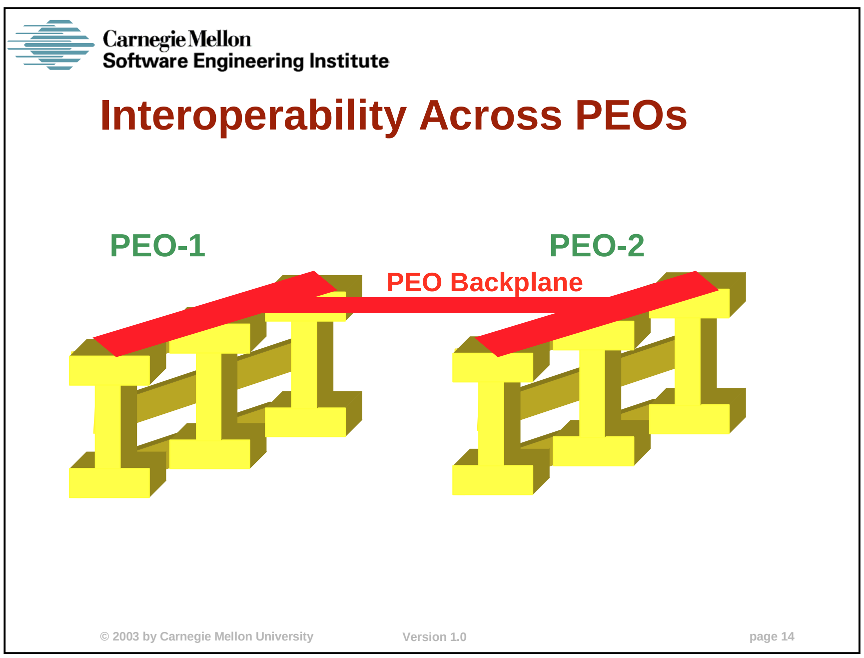
 Version 1.0 page 10


Carnegie Mellon
Software Engineering Institute
System Activities Model
Program Management
System Construction
Operational System
Activities performed to manage the acquisition of a system.
Activities performed to create a system. Focus is on architecture, standards, COTS.
Activities performed to operate a system. Focus is on interactions with other systems, and data distribution.
© 2003 by Carnegie Mellon University
Version 1.0
page 11


Carnegie Mellon
Software Engineering Institute
Different Types of Interoperability
Programmatic Interoperability
PMO
PMO
PMO
Construction
Construction
Construction
Constructive interoperability
System
System
System
Operational Interoperability
© 2003 by Carnegie Mellon University
Version 1.0
page 12



Carnegie Mellon
Software Engineering Institute
Interoperability Backplanes
PMO
PMO
PMO
Construction
Construction
Construction
System
System
System
© 2003 by Carnegie Mellon University
Version 1.0
page 13



Carnegie Mellon
Software Engineering Institute
Interoperability Across PEOs
PEO-1
PEO-2
PEO Backplane
© 2003 by Carnegie Mellon University
Version 1.0
page 14

Carnegie Mellon
Software Engineering Institute

Operational Interoperability

Scope of activities related to ***operation*** of a system in the context of other systems. Can include items such as

- data specification (and semantics)
- semantics of operations
- communication protocols

 Version 1.0 page 15

Carnegie Mellon
Software Engineering Institute

Operational Interoperability: Example

Multiple combat platforms are exchanging data over different communication links. Each type of link places different restrictions on the data. So different users get different views of the battle environment.

How can the users be wholly confident in the information that they are receiving?

 Version 1.0 page 16

Carnegie Mellon
Software Engineering Institute

Constructive Interoperability

Scope of activities related to ***construction*** (and maintenance) of one system in the context of another system. Can include collective use of

- architecture
- standards
- COTS Products

 Version 1.0 page 17

Carnegie Mellon
Software Engineering Institute

Constructive Interoperability: Example

Two systems are being built using industry standard object request brokers provided by different COTS vendors. The program offices assumed that conformance to an industry standard implied interoperability. Unfortunately, one vendor added unique features that extended the standard.

When the systems were tested, the systems did not interoperate as planned. Why?

 Version 1.0 page 18

Carnegie Mellon
Software Engineering Institute

Programmatic Interoperability

Scope of activities related to ***management*** of a program in the context of another program. Can include items such as

- Synchronization of schedules (budgets too?)
- Joint risk management
- Coupled award fee boards
- Linked promotions

 Version 1.0 page 19

Carnegie Mellon
Software Engineering Institute

Programmatic Interoperability: Example

One program is building a large distributed combat system. They are integrating many subsystems, most provided by other programs. As part of their system test, they require a simulator, which is developed by another program.

Funny, but the simulator was late. When it arrived the simulator did not implement the interface as expected. Should we be surprised when things started going wrong?

 Version 1.0 page 20

Carnegie Mellon
Software Engineering Institute

Assessment

As part of the IR&D we want to assess the state of the practice.

We plan to use structured discussions with *key* individuals.

We are interested in
- Who are the right people to talk to?
- What is the subject matter to address?
- What are the most important questions for us to ask?

Carnegie Mellon
Software Engineering Institute

Future Steps

What is necessary in order to achieve interoperability and how do you know it?

Where are the key holes that need to be filled?

Carnegie Mellon
Software Engineering Institute

Context for the Future

	Categories (e.g., goals, policies, processes)
PMO Programmatic	
PMO Constructive	
PMO Operational	
PEO	
Joint Programs	

 Version 1.0 page 23

Carnegie Mellon
Software Engineering Institute

Summary

We thank you for your attendance and cooperation to make this effort a success.

We look forward to more cooperation in the future.

 Version 1.0 page 24

Carnegie Mellon
Software Engineering Institute

Backups

 Version 1.0 page 25

Carnegie Mellon
Software Engineering Institute

Defining Interoperability

Lots of definitions are possible, and involve different views, e.g.,

“ The ability of systems to work together.”

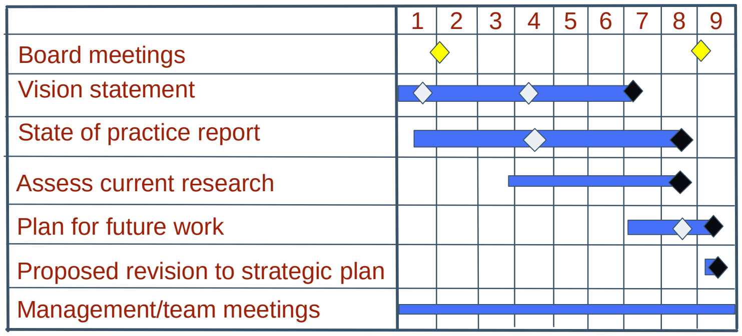
“The ability of systems to exchange and use services.”

Ours:
“The degree to which a set of communicating systems are (i) able to exchange specified state data, and (ii) operate on that state data according to specified, agreed to, operational semantics.”

 Version 1.0 page 26

Carnegie Mellon
Software Engineering Institute

Schedule



 Version 1.0 page 27

References

[Joint 00] Joint Chiefs of Staff. *Joint Vision 2020*. Washington, D.C.: U.S. Government Printing Office, June 2000. <http://www.dtic.mil/jointvision/jvpub2.htm>. (URL valid as of June 2003)

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE June 2003	3. REPORT TYPE AND DATES COVERED Final
4. TITLE AND SUBTITLE Proceedings of the System of Systems Interoperability Workshop (February 2003)		5. FUNDING NUMBERS F19628-00-C-0003
6. AUTHOR(S) Linda Levine, B. Craig Meyers, Ed Morris, Patrick R. H. Place, Daniel Plakosh		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Software Engineering Institute Carnegie Mellon University Pittsburgh, PA 15213		8. PERFORMING ORGANIZATION REPORT NUMBER CMU/SEI-2003-TN-016
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 Software Engineering Institute has initiated an internal research and development effort to investigate interoperability between systems. As part of the research, a workshop was held in February 2003 with an advisory board of Department of Defense experts. A preliminary model of interoperability was presented and feedback on the model was requested. This technical note documents the model of interoperability presented and the findings from the workshop.		
14. SUBJECT TERMS internal research and development, interoperability, model, system of systems		15. NUMBER OF PAGES 34
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