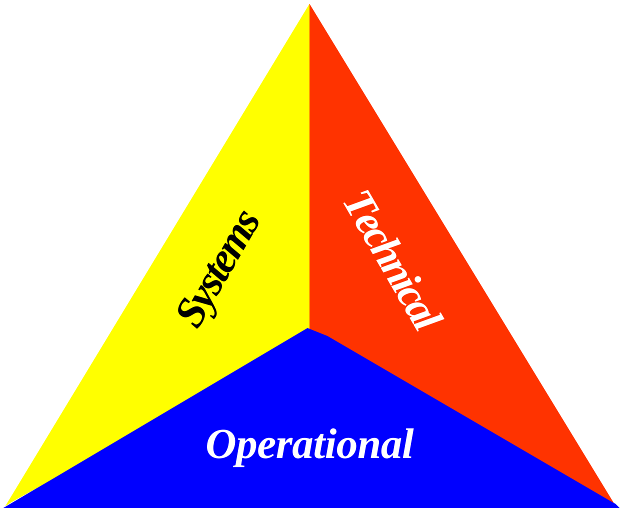
DoD Architecture Framework Working Group

DoD Architecture Framework Version 1.0



Deskbook

15 August 2003

TABLE OF CONTENTS

SECTION PAGE

1 INTRODUCTION 1-1

2 TECHNIQUES FOR DEVELOPING ARCHITECTURES 2-1

2.1 OVERVIEW 2-1

2.2 STRUCTURED METHODOLOGY FOR REQUIREMENTS-BASED IT ARCHITECTURE DEVELOPMENT 2-1
- 2.2.1 Introduction 2-1
- 2.2.2 Architecture Development Process 2-2
- 2.2.3 Developing the Architecture Views 2-5
- 2.2.4 Architecture Life Cycle 2-12
- 2.2.5 Supporting Processes 2-13
- 2.2.6 Conclusion 2-13
- 2.2.7 References 2-14

2.3 DEPARTMENT OF THE NAVY CHIEF INFORMATION OFFICER (DON CIO) PROCESS GUIDANCE 2-16
- 2.3.1 Department of the Navy's Architecture Development Process Model (ADPM) 2-16
- 2.3.2 ADPM Excerpt 2-17
- 2.3.3 Reference 2-27

2.4 EXAMPLE ARCHITECTURE USING STRUCTURED ANALYSIS AND UNIFIED MODELING LANGUAGE 2-28
- 2.4.1 Introduction 2-28
- 2.4.2 Common Products 2-29
- 2.4.3 Example Products Based on Structured Analysis Techniques 2-30
- 2.4.4 Example Products Based on Object-Oriented Techniques and the UML 2-45
- 2.4.5 References 2-56

2.5 USSPACECOM ARCHITECTURE DEVELOPED WITH OBJECT-ORIENTED METHODOLOGY 2-57
- 2.5.1 Objective of Case Study 2-57
- 2.5.2 Operational View 2-58
- 2.5.3 Systems View 2-65
- 2.5.4 Technical Standards View 2-68
- 2.5.5 Conclusion 2-72
- 2.5.6 References 2-73

TABLE OF CONTENTS (Cont)

SECTION PAGE

2.6 SECURITY/INFORMATION ASSURANCE ARCHITECTURE 2-74
2.6.1 Introduction 2-74
2.6.2 Risk Assessment Overview 2-74
2.6.3 Security/Information Assurance and the Framework 2-77
2.6.4 Summary and Conclusion 2-82
2.6.5 References 2-82

2.7 AN ARCHITECTURE PERSPECTIVE ON NCOW 2-84
2.7.1 Introduction 2-84
2.7.2 NCOW Tenets 2-84
2.7.3 Net-Centric Enterprise Services 2-86
2.7.4 NCOW Reference Model 2-87
2.7.5 NCOW Information Exchanges 2-90
2.7.6 Example Products Depicting NCOW 2-91
2.7.7 References 2-94

2.8 REPRESENTING THE ROLE OF HUMANS IN ARCHITECTURES 2-95
2.8.1 Overview 2-95
2.8.2 Roles of the Human Within Systems 2-95
2.8.3 Human-Centered Architecture Supplements 2-96
2.8.4 Human Systems Integration (HSI) Considerations for Architecture Products 2-98
2.8.5 Reference 2-105

2.9 CAPABILITY MATURITY PROFILE 2-106
2.9.1 Capability Maturity Roadmap 2-106
2.9.2 Capability Maturity Scale 2-107
2.9.3 Capability Maturity Profile Example 2-107
2.9.4 References 2-111

2.10 ARCHITECTURE LEVELS OF DETAIL 2-112
2.10.1 Introduction 2-112
2.10.2 Planner/Scope 2-113
2.10.3 Owner/Business Mode 2-115
2.10.4 Designer/System Model 2-118
2.10.5 Builder/Technology 2-120
2.10.6 Contractor 2-121

3 TECHNIQUES FOR USING ARCHITECTURES 3-1

3.1 INTRODUCTION 3-1

TABLE OF CONTENTS (Cont)

SECTION PAGE

3.2 AIR FORCE CAPABILITY-BASED ANALYSIS 3-1
3.2.1 Definition of Capability 3-1
3.2.2 Describing Capabilities..3-2
3.2.3 Composite Nature of Capability 3-3
3.2.4 Capability Reports 3-3
3.2.5 Prioritized Capability List (CR-1)...3-4
3.2.6 Capability to Requirements and/or Tasks Matrix (CR-2)...3-4
3.2.7 Capability to Systems/Programs Traceability Matrix (CR-5)...3-6
3.2.8 Capability Evolution Description (CR-6) 3-8
3.2.9 Integrated Capability Analysis Summary (CR-7) 3-10
3.2.10 References..3-10

3.3 NAVY'S MISSION CAPABILITY PACKAGE APPROACH 3-11
3.3.1 Introduction..3-11
3.3.2 Mission Capability Package 3-11
3.3.3 Architecture Analysis Approach 3-12
3.3.4 References..3-29

3.4 DESIGNATION AND USE OF KEY INTERFACE PROFILES...3-30
3.4.1 Introduction..3-30
3.4.2 Identifying a Key Interface 3-30
3.4.3 Analyzing an Architecture to Identify Key Interfaces......3-32
3.4.4 Developing a Key Interface Profile......................................3-33
3.4.5 Uses and Benefits of the KIP Profile 3-35
3.4.6 References..3-36

3.5 C4I SUPPORT PLANS 3-37
3.5.1 Applicable Policy Document 3-37
3.5.2 Recommended Framework Products..................................3-38
3.5.3 References..3-51

3.6 THE ROLE OF ARCHITECTURES IN CAPITAL PLANNING AND INVESTMENT CONTROL (CPIC) 3-52
3.6.1 Introduction..3-52
3.6.2 Overview of CPIC 3-52
3.6.3 Overview of Enterprise Architecture 3-53
3.6.4 Relating DoD Architecture Framework Products to the EA 3-55
3.6.5 Using the Enterprise Architecture in the CPIC Process..3-55

TABLE OF CONTENTS (Cont)

SECTION PAGE

3.6.6 Using the EA and Framework Products During the Select Phase....3-56
3.6.7 Using the EA and Framework Products During the Control Phase 3-57
3.6.8 Updating the EA and Framework Products During the Evaluate Phase....3-58
3.6.9 References....3-58

4 ADDITIONAL INFORMATION 4-1

4.1 INTRODUCTION....4-1

4.2 ARCHITECTURAL CONCEPTS AND CADM 4-1
4.2.1 Introduction....4-1
4.2.2 Basic Architectural Elements....4-2
4.2.3 Conceptual Descriptions of Basic Architecture Concepts 4-4
4.2.4 Using CADM 4-16
4.2.5 Relating Conceptual Basic Architectural Elements to Entities of the CADM 4-18
4.2.6 References....4-25

4.3 ARCHITECTURE MODELING AND REPOSITORY TOOLS ASSESSMENT CRITERIA AND APPROACH 4-26
4.3.1 Introduction....4-26
4.3.2 Scope 4-26
4.3.3 Uses of an Architecture Tool Set 4-26
4.3.4 Tool User Categories....4-27
4.3.5 Tools Assessment Criteria 4-27
4.3.6 Assessment Approach....4-32
4.3.7 Issues with Choosing a Tool 4-33
4.3.8 Issues With Organizational Use of Automated Tools 4-33
4.3.9 Recommended Solution....4-34
4.3.10 References....4-34

4.4 FEDERAL ENTERPRISE ARCHITECTURE REFERENCE MODELS – RELATIONSHIP TO DOD AND DOD ARCHITECTURE FRAMEWORK 4-35
4.4.1 Introduction....4-35
4.4.2 Comparison Between DoD and Office of Management and Budget (OMB) Approach to Architecture 4-35
4.4.3 Comparison Between DoD Enablers and Equivalent FEA RMs 4-38

TABLE OF CONTENTS (Cont)

SECTION PAGE

4.4.4 The DoD Architecture Framework and the FEA RMs..... 4-50
4.4.5 Conclusion 4-53
4.4.6 References.......... 4-54

4.5 UNIVERSAL REFERENCE RESOURCES 4-55
4.5.1 Introduction.......... 4-55
4.5.2 Missions and Functions: The Universal Joint Task List (UJTL).......... 4-56
4.5.3 Data Environment and Standards 4-57
4.5.4 Technical Implementation Criteria 4-60
4.5.5 Enterprise Capabilities and Services 4-63
4.5.6 Maturity Models and Transition Guidance.......... 4-68

ANNEX

A GLOSSARY.......... A-1
B DICTIONARY OF TERMS.......... B-1
C DICTIONARY OF UML TERMS C-1
D CADM KEY ENTITY DEFINITIONS.......... D-1

LIST OF FIGURES

FIGURE PAGE

2.2-1 Architecture Development Process.......... 2-2
2.2-2 Data-Centric Build Sequence.......... 2-5
2.2-3 Architecture Life Cycle.......... 2-12
2.2-4 Processes Supported by Architectures.......... 2-13

2.4-1 Activity Decomposition for *Conduct Joint Force Targeting*.......... 2-29
2.4-2 Activity Decomposition for *Conduct Combat Assessment*.......... 2-29
2.4-3 Notional Overview and Summary Information (AV-1).......... 2-31
2.4-4 Example High-Level Operational Concept Graphic (OV-1).......... 2-32
2.4-5 Example Operational Node Connectivity Description (OV-2).......... 2-33
2.4-6 Notional Operational Information Exchange Matrix (OV-3).......... 2-34
2.4-7 Example Organizational Relationships Chart (OV-4).......... 2-35
2.4-8 Example *Conduct Combat Assessment* Operational Activity Model (OV-5).......... 2-36
2.4-9 Notional Decision Tree for *Conduct Combat Assessment*.......... 2-37
2.4-10 Notional Operational State Transition Description (OV-6b).......... 2-38
2.4-11 Notional Operational Event-Trace Description (OV-6c).......... 2-39
2.4-12 Notional Logical Data Model (OV-7).......... 2-40
2.4-13 Example Systems Interface Description (SV-1).......... 2-41
2.4-14 Notional Systems Communications Description (SV-2).......... 2-42
2.4-15 Notional Systems Functionality Description (SV-4).......... 2-43
2.4-16 Notional Systems Evolution Description (SV-8) Template.......... 2-44
2.4-17 Structured English.......... 2-44
2.4-18 Notional Top-Level UML Collaboration Diagram Used for the Operational Node Connectivity Description (OV-2).......... 2-46
2.4-19 Second UML Collaboration Diagram Used for the Operational Node Connectivity Description (OV-2).......... 2-46
2.4-20 Third UML Collaboration Diagram Used for the Operational Node Connectivity Description (OV-2).......... 2-47
2.4-21 Notional UML Class Diagram Used for the Operational Node Connectivity Diagram (OV-2).......... 2-48
2.4-22 Notional UML Class Diagram Used for the Organizational Relationships Chart (OV-4).......... 2-49
2.4-23 Notional UML Use Case Diagram Used for Operational Activity Hierarchy Chart (OV-5).......... 2-50
2.4-24 Notional UML Use Case Diagram Used for Operational Activity Model (OV-5).......... 2-50
2.4-25 Notional UML Activity Diagram Used for the Operational Activity Model (OV-5).......... 2-51
2.4-26 Notional UML Class Diagram Used for the Logical Data Model (OV-7).......... 2-52
2.4-27 Notional Systems Interface Description (SV-1).......... 2-53

LIST OF FIGURES (Cont)

FIGURE PAGE

2.4-28 Notional Use Case Diagram Supporting Systems Functionality Description (SV-4) 2-54
2.4-29 Notional UML Class Diagram Used for SV-4 Data Flow Diagram........ 2-55
2.4-30 Notional UML Class Diagram Used for Physical Schema (SV-11) 2-56

2.5-1 Spirally Evolving to Integrated/Interoperable Command and Control 2-57
2.5-2 Reusable Structure of the CCIC2S Architecture Approach........ 2-58
2.5-3 Understanding and Communicating Requirements 2-59
2.5-4 Node Connectivity Description (UML Collaboration Diagram) 2-61
2.5-5 NCD Sequence (UML Sequence Diagram) 2-61
2.5-6 Creating C4ISR Architecture Framework Operational Views 2-62
2.5-7 Developing the Logical Data Model (OV-7) and Operational Event-Trace Description (OV-6c) 2-63
2.5-8 System Operational Sequence........ 2-64
2.5-9 Developing System Responsibilities........ 2-65
2.5-10 Mapping System Responsibilities to Product Lines........ 2-66
2.5-11 Refining Product Line Allocation to Components 2-67
2.5-12 Systems to Technical Standards Architecture Map 2-69
2.5-13 Integrated Space Command and Control Net-Centric Model........ 2-71
2.5-14 ISC2 Logical Data Model and the Enterprise Object Model........ 2-72

2.6-1 Node Connectivity Example Showing Security Attributes for a Needline 2-79
2.6-2 Systems Connectivity Example Showing Security Attributes for a Physical Link 2-81

2.7-1 NCOW Nodes on the Grid 2-85
2.7-2 Example As-Is and To-Be Information Exchanges........ 2-86
2.7-3 *Provide Net-Centric Information Environment* as a Common Activity in NCOW Environment Use Cases 2-89
2.7-4 Example OV-1 for NCOW........ 2-91
2.7-5 Example OV-2 for NCOW........ 2-92
2.7-6 Example SV-1 for NCOW........ 2-93
2.7-7 Example SV-2 for NCOW........ 2-94

2.8-1 Illustrative OV-4 2-101
2.8-2 Notional Example of Analysis of Human Activities for Time-Critical Targeting........ 2-103
2.8-3 Notional Example of Human-Centered Supplementary Information for the SV-4........ 2-104
2.8-4 Notional Example of Human Supplemental Information for the SV-5 2-105

2.9-1 IC Community Example Roadmap........ 2-107

LIST OF FIGURES (Cont)

FIGURE PAGE

2.9-2 Collaboration as the Focus Area 2-108
2.9-3 Basic Definition of Level 2 Collaboration.................. 2-108
2.9-4 Detailed Definition of Level 2 Collaboration.................. 2-109
2.9-5 Other Component Areas That Interact with Collaboration.................. 2-110
2.9-6 Capability Levels Needed in the Related Component Areas 2-110

2.10-1 DoDAF Support to the Planner 2-113
2.10-2 DoDAF to the Owner.................. 2-116
2.10-3 DoDAF Support to the Designer.................. 2-118
2.10-4 DoDAF Support to the Builder.................. 2-120
2.10-5 DoDAF Support to the Contractor.................. 2-122

3.2-1 Describing Capabilities with Architectures.................. 3-3
3.2-2 Basic Capability Relationships 3-4
3.2-3 Relating Capabilities and Activities 3-5
3.2-4 DOTMLPF: Factors Affecting Capability.................. 3-6
3.2-5 Operational Attributes Required to Achieve Capability 3-6
3.2-6 Relating Systems to Capabilities.................. 3-7
3.2-7 System Attributes Supporting Capabilities.................. 3-8
3.2-8 Illustrative Capability Evolution Description.................. 3-9

3.3-1 Using Architectures for Analysis (Navy’s RDA CHENG) 3-12
3.3-2 Example Activity Flow Diagram for Time-Critical Targeting 3-14
3.3-3 Mapping Activities to Systems to Systems Functions 3-15
3.3-4 Bar Chart of Number of Systems Supporting Each Function.................. 3-16
3.3-5 Example Activity Flow Diagram with Systems Mapping 3-16
3.3-6 1st Order Analysis.................. 3-18
3.3-7 2nd Order Analysis Part 1 3-21
3.3-8 2nd Order Analysis Part 2 3-22
3.3-9 2nd Order Analysis 3-23
3.3-10 3rd Level Analysis.................. 3-24
3.3-11 Activity Flow Diagram with Overlay of Events, Timing, and Systems.................. 3-24
3.3-12 Sample Capability Evolution Diagram.................. 3-27
3.3-13 Using Architecture Assessments and Systems Engineering to Develop the Acquisition Plan.................. 3-28

3.4-1 Illustrative Types of Key Interfaces.................. 3-31
3.4-2 Using Architecture Products to Identify a Key Interface 3-33
3.4-3 KIP Process.................. 3-35

3.5-1 Products Applicable to a C4I Support Plan 3-38
3.5-2 Notional OV-2.................. 3-42
3.5-3 Notional OV-3.................. 3-43

LIST OF FIGURES (Cont)

FIGURE PAGE

3.5-4 Notional SV-1 3-47
3.5-5 Notional SV-6 3-48
3.5-6 Notional TV-1 3-50
3.5-7 Relation of Architecture Products for C4ISP Analysis 3-51

3.6-1 CPIC Has Select, Control, and Evaluate Phases, Goal – Link Mission Needs, Information, and IT Effectively and Efficiently 3-53
3.6-2 OMB A-130 Enterprise Architecture Elements 3-54
3.6-3 OMB A-130 Enterprise Architecture Elements with Applicable Framework Products Overlay 3-55

4.2-1 Architecture Concepts Model.. 4-3
4.2-2 Operational Nodes Concept Model 4-4
4.2-3 Operational Activities Concept Model.. 4-5
4.2-4 Information and Data Concept Model.. 4-5
4.2-5 Interfaces Concept Model 4-6
4.2-6 Systems Nodes Concept Model 4-7
4.2-7 Systems Concept Model.. 4-8
4.2-8 Networks Concept Model.. 4-8
4.2-9 Performance Concepts Example.. 4-9
4.2-10 Standards Concepts Example 4-10
4.2-11 Architectures Concept 4-12
4.2-12 Architecture and Requirements Concept Examples 4-13
4.2-13 Schedules Concept Examples.. 4-14
4.2-14 Technologies Concept Example 4-15
4.2-15 Costs and Programmatics Concept Model.. 4-15

4.3-1 Scope of Architecture Tools With Respect to Other Tools 4-27
4.3-2 Architecture Modeling and Repository Tool Suite 4-28
4.3-3 Approaches to Utilizing Tools in Supporting Architecture Development 4-34
4.4-1 FEA Reference Models (used from FEA PMO Web site).. 4-37
4.4-2 Comparison Between DoD Budget Titles and FEA BRM Structure 4-39
4.4-3 FEA SRM Categories.. 4-41
4.4-4 Comparison Between DoD NCES and FEA SRM Structure 4-42
4.4-5 FEA TRM 4-43
4.4-6 FEA TRM Service Areas and Categories.. 4-44
4.4-7 Comparison Between DoD TRM and FEA TRM Structure 4-45
4.4-8 Six Measurement Areas of the PRM Capture the Relationship Among Inputs, Outputs, and Outcomes.. 4-48
4.4-9 Comparison Between DoD Performance-Based Budget and FEA PRM.... 4-50
4.4-10 Comparison Between DoDAF Architectures and Reference Models.......... 4-52

LIST OF FIGURES (Cont)

FIGURE PAGE

4.5-1 Shared Data Server Architecture 4-60
4.5-2 DoD Technical Reference Model 4-61
4.5-3 GIG Systems Reference Model 4-64
4.5-4 NCOW RM High-Level Concept Graphic (OV-1) 4-65
4.5-5 NCOW RM Emerging Commercial Standards 4-66
4.5-6 NCES are Core Services within the GIG 4-67
4.5-7 LISI Levels of Interoperability Sophistication 4-70
4.5-8 Components (Focus Areas) of the *IC IS Capability Roadmap* 4-73
4.5-9 The Standard Capability Maturity Scale 4-74
4.5-10 Details Provided for Each Component of the *IC IS Capability Roadmap* 4-75
4.5-11 Using the Roadmap in Architecture Analysis to Link Investments to Mission Effectiveness 4-76

LIST OF TABLES

TABLE PAGE

2.6-1 Security Goals 2-75
2.6-2 Asset Assessment 2-75
2.6-3 Damage Effects 2-76
2.6-4 Probability of Occurrence 2-76
2.6-5 Levels-of-Concern (Damage Effect * Probability of Occurrence) 2-76

2.7-1 A0 Provide Net-Centric Information Environment 2-88

2.8-1 Human-Centered Supplementary Architectural Information 2-97
2.8-2 Behavioral Processes and Definitions 2-98
2.8-3 Human-Centered Supplement to OV-4 2-101
2.8-4 Human Activities Notional Example 2-102

4.2-1 Relation of CADM Entities to Basic Architectural Elements 4-19

4.4-1 Example of Measurement Indicators Sorted by Measurement Category 4-49
4.4-2 PRM Alignment Table 4-50
4.4-3 Summary of Relationships Between FEA RMS and DoDAF Core Data Elements 4-53

4.5-1 Universal Reference Resources 4-55
4.5-2 Contributions of LISI to Selected C4ISR Architecture Framework Products 4-71

1 INTRODUCTION

The Department of Defense (DoD) Architecture Framework (DoDAF), Version 1.0, defines a common approach for DoD architecture description development, presentation, and integration. The Framework enables architecture descriptions to be compared and related across organizational boundaries, including Joint and multinational boundaries.

The Framework is partitioned into two volumes and a Deskbook. Volume I provides definitions, guidelines, and some background material. Volume II contains descriptions of each of the product types. This third volume is the DoD Architecture Framework Deskbook and provides supplementary guidance to Framework users.

The Deskbook presents several techniques for developing and using architectures. These various techniques were developed by different segments of the DoD community and do not represent coordinated community positions. Volumes I and II presented mandatory guidance to the DoD community. The techniques presented in this Deskbook are not mandatory but are provided for their insights and potential utility to the reader. However, readers should determine the applicability of a technique to their individual situation.

Because this Deskbook is being published as part of the DoDAF, the techniques herein were developed during the time that the C4ISR Architecture Framework was operative. Some, but not all, of the material has been updated to reflect the DoDAF. The reader may see some material that is C4ISR Architecture Framework specific. These small discrepancies should not interfere with the DoDAF-related value of the material.

The Deskbook also provides additional material for topics that were introduced in Volumes I and II. This material includes the All-DoD Core Architecture Data Model (CADM), architecture tools, Federal Enterprise Architecture Reference Models, and Universal Reference Resources.

2 TECHNIQUES FOR DEVELOPING ARCHITECTURES

2.1 OVERVIEW

The techniques presented in this section provide various approaches for developing architectures and architecture products. The material does not reflect community-agreed approaches but instead represents concepts developed in specific segments of the DoD community. This material was developed during the period that the C4ISR Architecture Framework V 2.0 was in effect. Some of the contents have been updated to reflect the DoD Architecture Framework (DoDAF); however, the material may differ slightly with the DoDAF.

The DoDAF does not support or endorse any specific process for developing architecture descriptions. Therefore, the various techniques presented in section 2 should not be considered officially supported or recommended. However, each has value and provides insights into architecture development.

2.2 STRUCTURED METHODOLOGY FOR REQUIREMENTS-BASED IT ARCHITECTURE DEVELOPMENT

"A Handbook for Building an IT Architecture (Short Version)"

2.2.1 Introduction

This handbook is intended to supplement the DoD Architecture Framework (DoDAF). While the DoDAF defines what should be developed as an architecture description, it does not define the process for building the architecture description. This handbook is a process guide that describes one method for developing information technology (IT) architectures to meet DoDAF requirements.

This handbook, based on the Structured Analysis and Design Technique, provides a step-by-step guide for building IT architectures. It should be used as a starting point for developing DoDAF-compliant Command, Service, and Agency (C/S/A) architectures that allow IT interoperability assessments, support IT decision-making, and meet Global Integrated Grid (GIG) and DoD Chief Information Officer (CIO) requirements. The process focuses on gathering information and building models required to conduct analyses supporting such objectives. While the handbook describes the building of DoDAF-compliant products, its real emphasis is on the interrelationship of these products and their use in describing an integrated architecture consisting of an Operational View (OV), Systems View (SV), and Technical Standards View (TV).

This process guide responds to concerns raised by C/S/A architecture organizations, many of which expressed the need for a common process that discusses how to build Framework products into complete and workable architectures. It was produced by Affiliated Computer Services (ACS) Defense as an outgrowth of its support to the Assistant Secretary of Defense for Networks and Information Integration (NII)/DoD CIO in developing the GIG Architecture. Handbook contents represent the combined experience of a number of architecture builders, as refined during development of GIG Architecture Version 1.

2.2.2 Architecture Development Process

This handbook describes a method for building DoDAF-compliant architectures. The method is data-centric rather than product-centric. The data-centric approach ensures concordance between the products and also ensures that all essential entity relationships are captured to support a wide variety of analysis tasks. The data underlying the architecture more directly supports analysis. The products created as a result of the architecture process become visual renderings of the underlying architecture data and are needed to convey information about the architecture to specific user communities. The methodology presented here shifts the focus to data and data relationships rather than products and moves the construction of the final products to the end of the process.

As depicted in **Figure 2.2-1**, there are six general steps that should be followed for developing the architecture and resultant products in accordance with the DoDAF. The six-step process provided in Figure 2.2-1 is a modification of the six-step process provided in Volume I of the DoDAF.

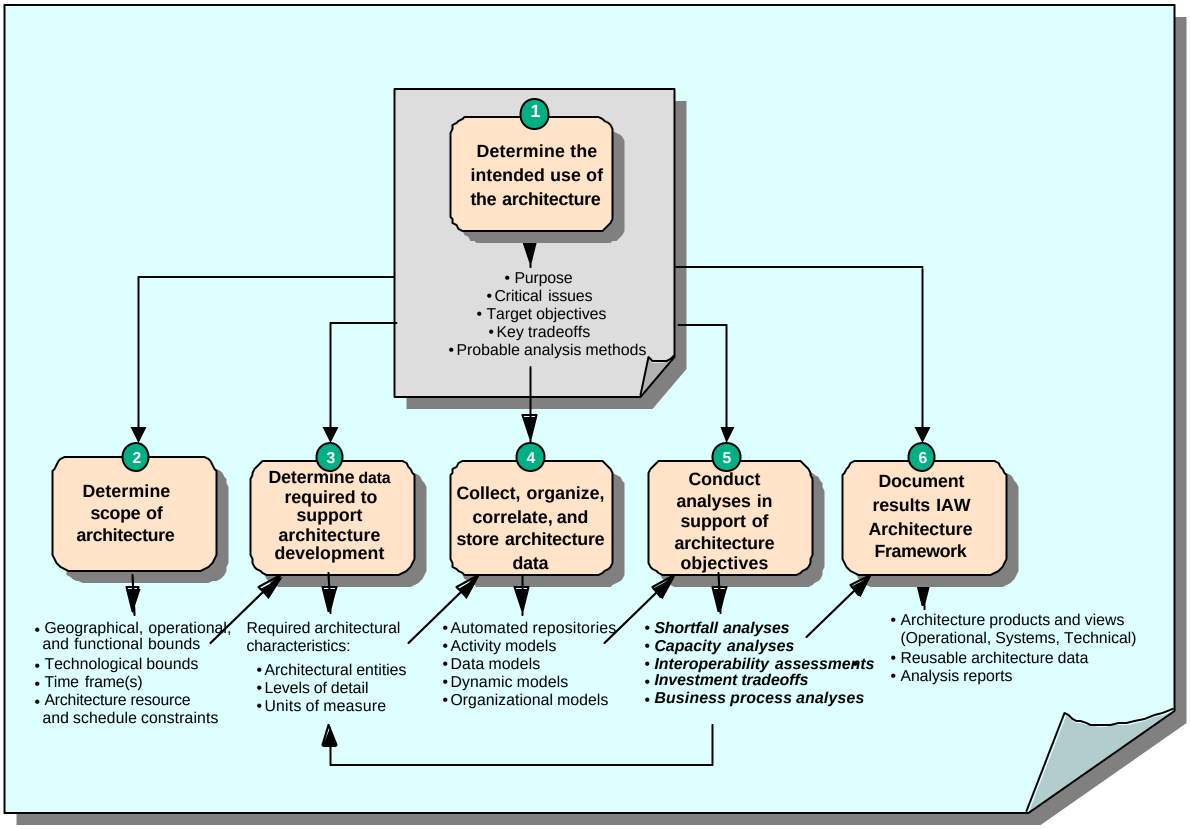

1
Determine the intended use of the architecture
• Purpose
• Critical issues
• Target objectives
• Key tradeoffs
• Probable analysis methods
2
Determine scope of architecture
• Geographical, operational, and functional bounds
• Technological bounds
• Time frame(s)
• Architecture resource and schedule constraints
3
Determine data required to support architecture development
Required architectural characteristics:
• Architectural entities
• Levels of detail
• Units of measure
4
Collect, organize, correlate, and store architecture data
• Automated repositories
• Activity models
• Data models
• Dynamic models
• Organizational models
5
Conduct analyses in support of architecture objectives
• Shortfall analyses
• Capacity analyses
• Interoperability assessments
• Investment tradeoffs
• Business process analyses
6
Document results IAW Architecture Framework
• Architecture products and views (Operational, Systems, Technical)
• Reusable architecture data
• Analysis reports

Figure 2.2-1. Architecture Development Process

This high-level process is the foundation for the architecture development methodology described later. The following paragraphs detail the six steps shown in the diagram.

Step 1: Determine Intended Use of Architecture

The purpose explains why the architecture is being developed. For example, it may be developed for business process reengineering (BPR) purposes (i.e., identifying nonmateriel solutions, such as improved procedures, realigning organizations, better training, or modifying functions), to establish and quantify acquisition requirements (e.g., systems, personnel, or facilities), or to assess the feasibility of attaining a particular vision under a specific set of circumstances. The purpose also explains what the architecture will accomplish and how it may affect organizations or system development. The importance of unambiguously stating the purpose is that it establishes clear and concise exit criteria to measure the architecture's satisfaction of the customer's overall requirement.

Step 2: Determine Architecture Scope

The scope defines the boundaries that establish the depth and breadth of the architecture. The scope bounds the architecture's problem set and helps define its context. Other elements of the context that bound the architecture are the environment and the organization's mission and vision. This step involves describing geographic, operational, functional, and technological limits of the architecture; determining applicable time frame(s); and recognizing available architecture development resources and schedule constraints.

The architecture's scope includes:

- **Subject Area** - describes the applicable capability, organizational area, or domain to which the architecture applies
- **Timeframe** - describes the point in time to which the architecture is applicable (Examples of words used to express time frame are current [As-Is or baseline], programmed [budgeted or planned], and objective [To-Be or future].)
- **Intended Users and Uses** - identifies the audience the architecture is intended to serve and how it is expected to use the architecture
- **Dimensionality** - helps identify the boundaries of the breadth and level of detail at which the architecture is to be developed; directly related to the purpose and perspective of the architecture

Step 3: Determine Data Required to Support Architecture Development

During this step, the data entities and attributes (such as activities, organizations, information elements, and other architecture components) are selected. Also selected is the level of detail to which these entities and attributes need to be identified to meet the objectives of the architecture.

This step determines the type of data that needs to be collected in Step 4. Recognized data types for consideration include:

- Rules that govern how activities should perform
- Guidance for mapping activities to organizational elements and nodes
- Information needed to accomplish activities
- Command relationships, task lists, required information about organizational elements and nodes
- Standard data dictionaries

- Rules on geo-distribution and environment
- Guidance for developing linkages among activities
- Results from specific activities
- Known likely external interfaces with other organizations (joint or coalition)
- Linkages to higher-level activities such as Universal Joint Task List (UJTL) tasks

Step 4: Collect, Organize, Correlate, and Store Architecture Data

Following data collection, cataloging, organizing, and entering the data into automated repositories permit subsequent analysis and reuse. As data is captured and stored, it should be defined and tagged with source information. Included in this step is the correlation of data in terms of activity, data, organizational, and dynamic models.

For reuse purposes, architecture data should be entered into a database. The contents of the database should be stored in terms of models. The database will include the scope, operational concept model, information process model, node connectivity model, behavioral model, and nodal-related data for the architecture. Information collected will be in sufficient detail to lead subject matter experts through the development of the activity model and related business rules.

Step 5: Conduct Analyses to Support Architecture Objectives

The types of analyses that are typically performed are:

- Determination of shortfalls between requirements and capabilities
- Assessments of processing and communications capacities
- Assessments of interoperability
- Analysis of alternatives to determine investment tradeoffs
- Analyses of business processes to determine possible non-materiel solutions

The analytical process provides insights into issues and concerns that were not readily apparent at the outset, and, as a result, Step 5 includes the identification of additional data collection requirements.

During analysis, the architect selects, compares, assesses, and transforms contextual and architectural inputs based upon the operational concept. The environment is then assessed and defined in terms of a set of assumptions and constraints (as specified in the operational concept) regarding operational, cultural, political, economic, and technological factors. These are examined against current and emerging doctrine, various threat conditions, and perceived needs. Typically, one or more scenarios are used to confirm expectations, discover shortfalls, or identify new opportunities. Operational impacts related to functions and capabilities enabled by leap-ahead technology are also considered.

Step 6: Document Results in Accordance with the Architecture Framework

The final step in the process involves building architecture products in accordance with templates established in the DoDAF. Architecture developers will build only those products necessary to meet the intended use of the architecture (Step 1). Architecture products will be captured in reusable and shareable form. A number of architecture tools are available to support this step. The tool should be selected based on the intended use of the architecture (Step 1).

2.2.3 Developing the Architecture Views

This section provides guidelines and a suggested build sequence for developing data-centric architecture products. The product ordering shown here takes advantage of the related nature of the products and the dependencies among products. This chronology does not imply a rigid course of events; however, there is an order of precedence that is required to ensure data integrity. An overview of the data-centric build sequence is provided in **Figure 2.2-2**.

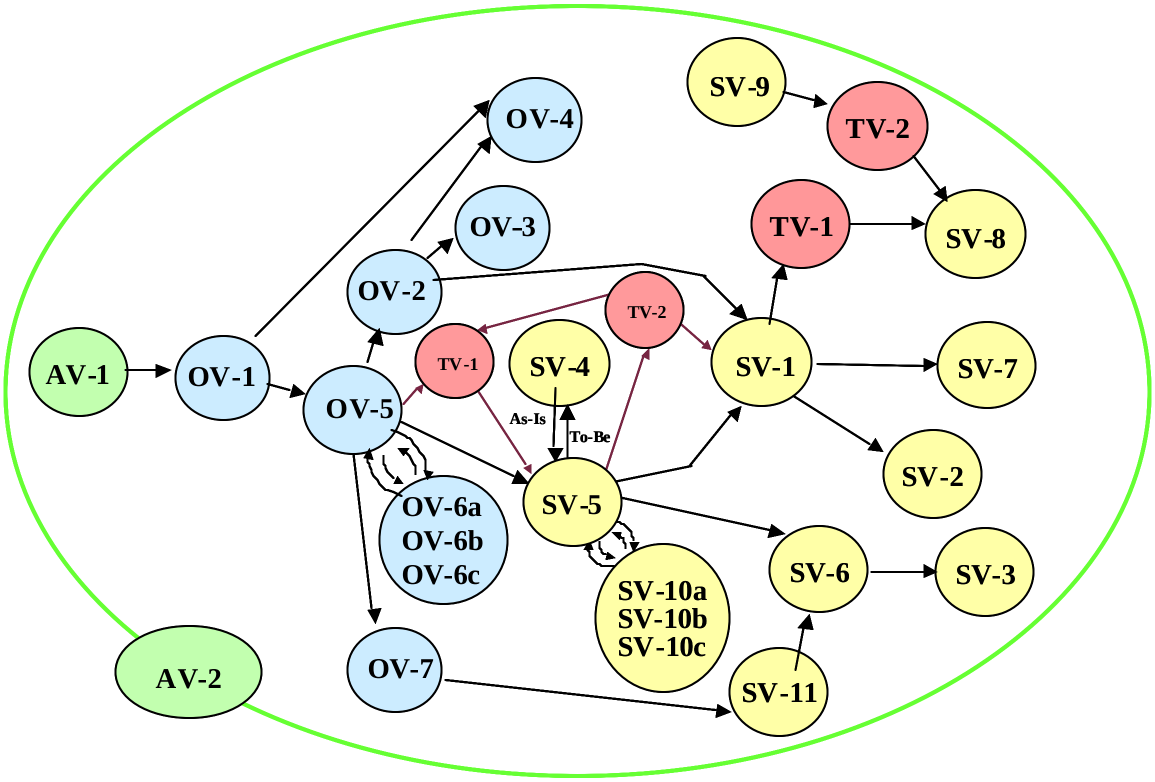


Figure 2.2-2. Data-Centric Build Sequence

The highly related nature of the products necessitate that they be developed in an iterative manner as greater understanding is achieved via the work process. It is not the intent of this section to suggest that a given product is developed and finalized before the next product is addressed.

2.2.3.1 Developing the All-Views Products

The All-Views products are started as the project begins and updated as the project progresses.

- Define purpose, scope, context, and tools: As noted in the six-step process, an understanding of the purpose, scope, and context is essential at the beginning of the project. Determining the automated tool set should also be accomplished prior to product builds. This information is documented in the early version of the **Overview and Summary Information (AV-1)**. The **AV-1** is updated as the project proceeds to include providing a description of the findings and recommendations that have been developed based on the architecture effort. Findings may include such things as identifications of shortfalls, recommended systems implementation, and opportunities for technology insertion.
- Define terms: The **Integrated Dictionary (AV-2)** should be started at the beginning of the architecture development process and updated continually throughout the development effort.

2.2.3.2 Developing the Operational View

The first view to be constructed is the Operational View. It has been noted that there is no relationship to the order of the products as presented in Version 1 of the DoDAF. However, from a data-centric perspective, there is an order in which the data is developed and organized to continually add layers of complexity to the description of the enterprise. This is due to the entity relationships that are inherent within the enterprise.

Understanding the difference between functional and organizational components is essential in collecting data in a manner that will support BPR. In order to support BPR, the business process must be captured independent of the organization or physical distribution of the business processes. The functional components describe the *why*, *what*, and *when* of the architecture establishing the requirements. The organizational components describe the *who* in terms of organizations, organizational relationships, and facilities.

The following is a suggested order for developing the OV products:

- Obtain or build an operational concept: This is the high-level concept and belongs to the business leader (thus providing his buy in to the process) and depicts the vision of how business is conducted. Relevant material to review and analyze includes pertinent joint, service, and command visions, doctrine, and tactics, techniques, and procedures (TTP). At this point, there is sufficient data to produce a **High-Level Operational Concept Graphic (OV-1)**.
- Document the business process: Next, the high-level concept is analyzed and a business process or activity model is constructed detailing the concept in the form of a set or sets of inter-related processes. To the extent possible, use activities from accepted standard tasks lists such as the UJTL, Joint Mission Essential Task Lists (JMETLs) developed by one or more of the commands, and/or Service Task Lists. Since the focus of these lists is primarily warfighting activities, they may not provide the needed coverage for support activities. However, using these lists to the extent possible and showing linkage between activities created for a given architecture and the activities in the standard lists provide a basis for architecture integration and facilitate an enterprise understanding. Determine the information flow associated with the

activity set. Identify inputs, controls, outputs, and mechanisms (ICOMs) associated with the activities. Use this information to produce the **Operational Activity Model (OV-5)**. To a very large extent, **OV-5** provides the foundation for the remaining OV products. Therefore, a reasonably stable version of **OV-5** should be developed before the other products are started.

After the **Operational Activity Model (OV-5)** is developed, the following products can be developed. The products are each dependent on **OV-5** but not on each other. Therefore, they may be developed in any sequence after **OV-5**.

- Document business rules associated with the business processes: Use scenario event threads (based on the **OV-5** activity model) to provide a context to capture business rules, state transitions, and event traces to produce the **Operational Activity Sequence and Timing Descriptions (OV-6)**. While there is an iterative aspect to the development of all architecture products, there is an especially strong iterative nature to the development of **OV-5** and **OV-6**. **OV-5** provides an initial set of operational activities for the initiation of the **OV-6** product set. Work on the **OV-6** product set may identify additional activities that then must be folded into **OV-5**. Similarly, as **OV-5** is matured, any new operational activities are incorporated into the **OV-6** product set.
- Aggregate activities into operational nodes: Organize activities into sets that will be logically collocated. Operational nodes are groupings of like activities that are performed together to carry out the operational concept. Nodes inherit the ICOMs associated with the activities performed at the nodes. **OV-5** provides the information flows among the activities performed at the various nodes. The information flows between two operational nodes are bundled into needlines. A needline represents an aggregation of information flows between two operational nodes, where the aggregated information exchanges are of a similar information type or share some characteristic. This results in the data required to produce the **Operational Node Connectivity Description (OV-2)**.
- Develop a Logical Data Model: Using the information exchanges identified in the activity model, develop the **Logical Data Model (OV-7)**.

After the **Operational Node Connectivity Description (OV-2)** is developed, the following products can be developed:

- Determine information exchange requirements: **OV-5** and **OV-2** provide the producing and consuming activities, the operational nodes at which they originate and to which they flow, and the information elements that they exchange. Relevant attributes of the information exchange are added to complete the matrix. Some automated tools will automatically generate the information exchange requirements (IER) matrix based on **OV-5** and **OV-2**. These IERs are documented in the **Operational Information Exchange Matrix (OV-3)**.

- Identify organization types that will perform the activities associated with the operational nodes: Referring back to **OV-1** and **OV-2**, organizations identified for the given operational concept and scenario are assembled into a force structure for conducting the designated operation. A key element of this structure is the relationship that must exist among the organizations that it comprises. This captures the data required to produce the **Organizational Relationships Chart (OV-4)**. This organizational laydown helps to capture the scenario-dependent long-haul communications requirements for the Systems View.
- Assign organizations and physical locations to operational nodes and activities: **OV-4** is then overlaid on **OV-2**. Principal and secondary organizations are assigned to each operational node, resulting in a new construct with both functional and physical characteristics called the operational facility (OpFac). The organizations assigned to an OpNode represent real (either type or specific) entities (e.g., units, offices, directorates, etc.) that will perform assigned activities at the node. Identify actual organizations to perform the activities and tasks delineated in earlier steps; update **OV-3** with the organizations associated with each information exchange. This captures the requirements of the individual organizations for systems and communications equipment to be satisfied by the Systems View.

2.2.3.3 Developing the Systems View

Once the Operational View has been completed, the Systems View can be created. The Systems View describes the *how* of the architecture and depicts systems elements, software, data, and connectivity required to support the enterprise business process. The basic high-level steps to develop the Systems View are:

- Identify physical node locations: This step is required to determine communications asset availability.
- Identify and characterize available systems in terms of owners, system functions, and performance: Document and characterize the available systems that support the business processes to be carried out at the operational nodes.

For As-Is Architectures:

- Identify the system functions provided in the current systems: Determine the logical relation between functions and associated subfunctions. Develop the **As-Is Systems Functionality Description (SV-4)**.
- Associate existing system functions with the operational activities they support: Using the **As-Is SV-4** and **OV-5**, map the existing system functions to the activities they support. Build the **As-Is Operational Activity to Systems Function Traceability Matrix (SV-5)**. **SV-5** provides the primary bridge between the Operational View and the Systems View. (The relation between the TV products and **SV-5** is discussed in the following section.)

For To-Be Architectures the order of developing **SV-4** and **SV-5** is reversed:

- Based on the operational activities, determine the required system functions: For the activities identified in **OV-5**, identify desired system functions to support each activity. Build the **To-Be Operational Activity to Systems Function Traceability Matrix (SV-5)**, which provides the primary bridge between the Operational View and the Systems View. (The relation between the TV products and **SV-5** is discussed in the following section.)
- Define the relationship among system functions: Given the system functions identified in the **To-Be SV-5**, develop a decomposition of those functions by identifying and organizing associated subfunctions. This provides the functional decomposition version of the **To-Be Systems Functionality Description (SV-4)**.

Develop the Physical Data Model: Using the **Logical Data Model (OV-7)**, determine how the logical data is physically implemented in the systems. This information becomes the **Physical Schema (SV-11)**. This product can be developed any time after **OV-7** is produced but must be available before the **Systems Data Exchange Matrix (SV-6)** is developed.

After the **Systems Functionality Description (SV-4)** (functional decomposition version) and the **Operational Activity to Systems Function Traceability Matrix (SV-5)** are developed, the **Systems Functionality Sequence and Timing Description (SV-10)** and the **Systems Interface Description (SV-1)** can be developed. The **SV-10** and the **SV-1** are each dependent on the **SV-4** and **SV-5** but not on each other. They may be developed in any sequence after the **SV-4** and **SV-5**.

- Determine systems' behavior: Following initial development of **SV-4** and **SV-5**, revisit the scenario threads previously developed for the Operational View and represented in the **OV-6** product set. Evaluate the threads for data exchanges and systems elements. Determine the timing and sequencing of events that capture a system performing the system functions described in **SV-4**. Both the **Systems State Transition Description (SV-10b)** and **Systems Event-Trace Description (SV-10c)** depict systems responses to sequences of events. Events may also be referred to as inputs, transactions, or triggers. When an event occurs, the action to be taken may be subject to a rule or set of rules as described in the **Systems Rule Model (SV-10a)**. Develop the **SV-10** product set (a, b, c). While there is an iterative aspect to the development of all architecture products, there is an especially strong iterative nature to the development of **SV-10** and **SV-4/SV-5**. **SV-4/SV-5** provides an initial set of system functions for the initiation of the **SV-10** product set. Work on the **SV-10** product set will likely identify system functions that then must be folded into **SV-4** and **SV-5**. Similarly, as **SV-4/ SV-5** is matured, any new system functions must be incorporated into the **SV-10** product set.
- Assign systems and their interfaces to the OPFACs: Referring to **OV-2** to which organizations and physical nodes have been attached, each organization assigned to an OPFAC brings with it a set of systems identified within the organization's authorization documents. Once the relevant systems at each

OPFAC are identified (i.e., those systems providing the functions associated with activities performed at the node), develop the **Systems Interface Description (SV-1)**.

- Map information exchange requirements into candidate systems: Referring to **OV-5** for information exchanges, **SV-5** for the relation between operational activities and system functions, and **SV-11** for the physical data model, develop the systems-related data exchange requirements that match the IERs presented in **OV-3**. Such an analysis supports a determination of how well information would flow during the operation. Produce the **Systems Data Exchange Matrix (SV-6)**.

After the **Systems Interface Description (SV-1)** is developed, the following products can be developed:

- Develop the **Systems Communications Description (SV-2)**:
  - Determine internodal networking requirements: Develop the networking requirements between systems located in different nodes. With this information, networks can be defined; and at this point, there is sufficient data to produce **SV-2** (for an internodal perspective).
  - Determine intranodal networking requirements: Determine the system-to-system communications requirements within nodes. This information is added to **SV-2** (for an intranodal perspective).
  - Identify available long-haul communications availability: Match internodal requirements to available long-haul communications. Add this information to **SV-2**.
  - Develop intranodal network and connectivity (long-haul communications and networks): At this point, there is sufficient data to finalize **SV-2**.
- Identify hardware and software performance parameters: Build the **Systems Performance Parameters Matrix (SV-7)**.

Following development of the **Systems Data Exchange Matrix (SV-6)**:

- Describe system-to-system relationships: From the system data exchange requirements assembled in **SV-6** and the system interfaces shown in **SV-1**, a matrix describing either existing and/or required system interfaces can be built. The **Systems-Systems Matrix (SV-3)** is derived from **SV-6** and **SV-1**. This requires **SV-6** and **SV-1** to be developed prior to **SV-3**. Some tools are able to generate **SV-3** based on the data associated with **SV-1** and **SV-6**.

Identify emerging technologies: Identify and quantify the emerging technologies, both hardware and software, that may be applicable to provide the best solution for the requirements described within the Operational View. This will provide detailed information to produce the **Systems Technology Forecast (SV-9)** as well as the **Technical Standards Forecast (TV-2)**.

The **SV-9** is not dependent on the other Systems View products; however, it is usually developed toward the latter part of the Systems View development process.

After the **Technical Standards Profile (TV-1)** and **Technical Standards Forecast (TV-2)** are developed, the following products can be developed:

- Document proposed systems migration strategy: This product may be developed to document existing evolution or migration strategies for the systems considered in this architecture. The existing standards defined in **TV-1** and the emerging standards from **TV-2** should be considered in determining the migration strategy. Construct the **Systems Evolution Description (SV-8)**. Alternatively, this product may be built after both the As-Is and To-Be architectures have been built, the migration strategy developed, and modifications against schedule determined. **SV-8** may be the last architecture product to be developed, since it is potentially dependent on **TV-1** and **TV-2**.

2.2.3.4 Developing the Technical Standards View

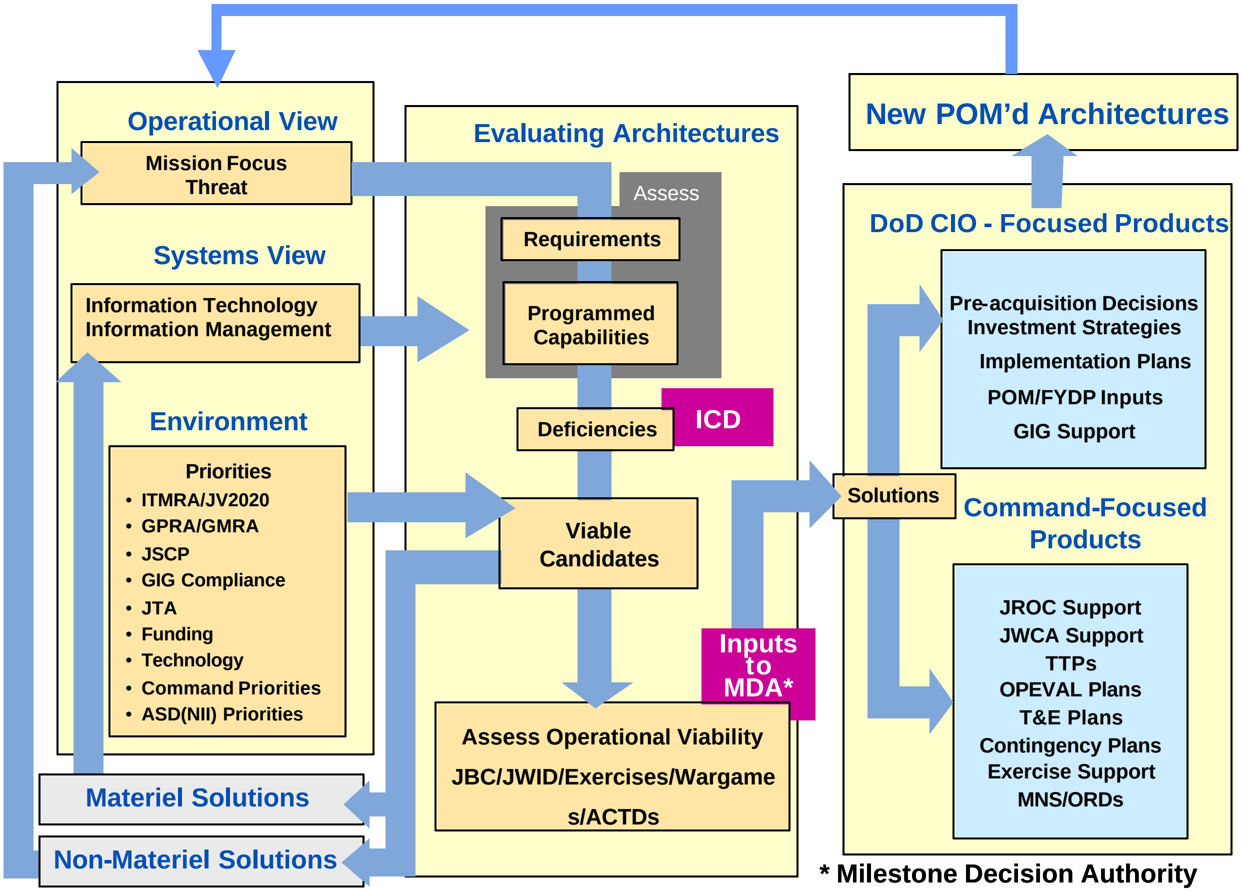
The Technical Standards View is used to identify standards for the enterprise and how they have been implemented.

- Determine applicable service areas: Begin developing an initial version of the **Technical Standards Profile (TV-1)** using the **Operational Activity Model (OV-5)** to determine the applicable service areas. Relate the service areas from **OV-5** to the services areas addressed in the JTA. Identify the standards used within the architecture for the service areas, and note whether they are consistent with the standards provided in the JTA. For service areas not included in the JTA, identify other applicable standards from existing sources of standards (International Standards Organization [ISO], Institute of Electrical and Electronics Engineers [IEEE], DII-COE, etc.). As the development of the SV products proceeds, use information on the system interfaces in **SV-1** for additional service areas and standards to be included in **TV-1**.
- Determine areas with no recognized standard: Compare the system functions identified in **SV-4/SV-5** to the service areas in **TV-1** to identify areas where standards do not yet exist. Document these areas to be addressed in the **Technical Standards Forecast (TV-2)**.
- Identify emerging standards: For service areas for which no accepted standard currently exists, identify emerging standards and expected time frames for adoption of the new standards. Incorporate this information into **TV-2**. In some cases, emerging standards may already be implemented at certain interfaces and, therefore, be reflected in **SV-1**. When this occurs, those emerging standards are also reflected back to **TV-1** as implemented standards. Use the **Systems Technology Forecast (SV-9)** developed earlier to identify projected system elements, associated emerging standards, and expected time frames for adoption of the new standards for incorporation into **TV-2**.

- As noted in the Systems View section above, **TV-1** and **TV-2** should be considered in developing the **Systems Evolution Description (SV-8)**.

2.2.4 Architecture Life Cycle

Figure 2.2-3 depicts the life of the architecture as it evolves and shows the process that the architecture description supports in the development, analysis, and evolution of the implemented architecture. In this illustration, the Operational View is used to drive the requirements that are evaluated against the Systems View. Operational deficiencies are derived from the analysis, and viable candidates are identified. These candidates can take the form of either materiel or non-materiel solutions and are modeled back into the Operational and Systems Views of the architecture. The architecture is re-analyzed, and the process continues until the operational deficiencies are minimized. The final sets of viable candidates are assessed for operational viability. Based on the results of the assessments, design changes are made and submitted for inclusion into the budgeting process. This process of developing, analyzing, and modifying continues throughout the architecture's life cycle.



* Milestone Decision Authority

Figure 2.2-3. Architecture Life Cycle

2.2.5 Supporting Processes

The enterprise IT architecture supports the six major institutional processes depicted in **Figure 2.2-4**. These processes are BPR; Planning, Programming, Budgeting, and Execution (PPBE) process; organization development; capability needs determination; research, development, and acquisition (RDA); and operations support. In addition, the architecture provides decision makers with information, common terms and concepts, procedures, models, and presentation products that can support operational, planning, and modernization requirements.

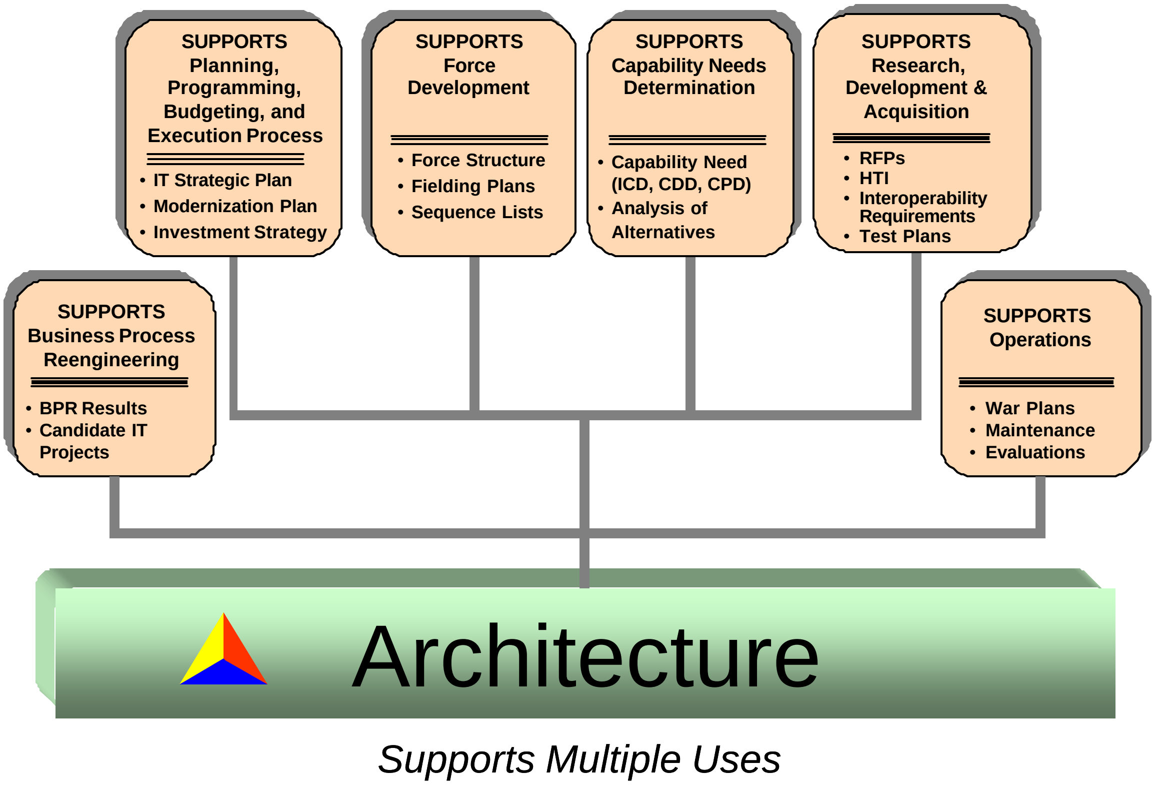

SUPPORTS Planning, Programming, Budgeting, and Execution Process
• IT Strategic Plan
• Modernization Plan
• Investment Strategy
SUPPORTS Force Development
• Force Structure
• Fielding Plans
• Sequence Lists
SUPPORTS Capability Needs Determination
• Capability Need (ICD, CDD, CPD)
• Analysis of Alternatives
SUPPORTS Research, Development & Acquisition
• RFPs
• HTI
• Interoperability Requirements
• Test Plans
SUPPORTS Business Process Reengineering
• BPR Results
• Candidate IT Projects
SUPPORTS Operations
• War Plans
• Maintenance
• Evaluations
Architecture
Supports Multiple Uses

Figure 2.2-4. Processes Supported by Architectures

2.2.6 Conclusion

Enterprise IT architectures provide decision makers with information, common terms and concepts, procedures, models, and presentation products that can support operational, planning, and modernization requirements. This document has provided a methodology used to develop enterprise IT architectures in compliance with the DoDAF Version 1.0. By using a six-step methodology, Operational, Systems, and Technical Standards Views are developed to provide enterprise-wide analysis of IT and support the following major institutional processes:

- BPR
- PPBE
- Organization development
- Capability needs determination

- RDA
- Operations support
- Interoperability analysis

This thorough and rigorous methodology adds value across the enterprise. It is an enabler for determining strategy-to-task traceability for all of the following within the Doctrine, Organization, Training, Materiel, Leadership, Personnel, and Facilities (DOTMLPF) construct:

- Doctrine: Architectures provide a basis for determining whether standard operating procedures are a fit for the required activities or if they require modification in moving from the As-Is to the To-Be.
- Organization: Through aggregation of the operational nodes identified in the Operational View, along with geographical, political, and real world constraints, the correct organizational fit can be determined.
- Training: The correct type of training required by the personnel to complete the activities identified in the Operational View can be identified through analysis.
- Materiel: The appropriate equipment required to complete the activities in the Operational View is identified in the Systems View; it is, therefore, traceable directly to strategy and business rules identified in the Operational View.
- Leadership: Command relationships, roles, and responsibilities with respect to the activities in the Operational View are identified.
- Personnel: The Operational View provides a basis for analysis of the correct type and number of personnel required to accomplish the identified activities.
- Facilities: The Operational View in the context of geographical, political, and real-world constraints determines the requirements for facilities.

While this methodology has been developed to support DoD and be compliant with the DoDAF, the process is applicable to any business. The analysis facilitated by this enterprise IT architecture methodology provides full strategy-to-task requirements traceability.

This methodology can be a key transformation enabler for realizing the vision of Decision Superiority outlined in ***Joint Vision 2020***:

> *"… to take advantage of superior information converted to superior knowledge to achieve "decision superiority" – better decisions arrived at and implemented faster than an opponent can react, or in a noncombat situation, at a tempo that allows the force to shape the situation or react to changes and accomplish its mission."*

2.2.7 References

Coffin, Jeffery, ACS Defense, *C4ISR Framework Compliant Enterprise Information Technology Architectures*, White Paper, 2001.

Coffin, Jeffery, ACS Defense, *A Structured Methodology for Requirements-Based Information Technology (IT) Architecture Development: A Handbook for Building a Useable IT Architecture*, Draft, February 4, 2002.

Coffin, Jeffery, and James Wise, ACS Defense, *Structured Methodology for Requirements-Based Information Technology (IT) Architecture Development: A Training Course for Implementing the Architecture Framework*, Briefing, January 15, 2003.

2.3 DEPARTMENT OF THE NAVY CHIEF INFORMATION OFFICER (DON CIO) PROCESS GUIDANCE

The Department of the Navy Chief Information Officer (DoN CIO) has developed process guidance for managers of architecture projects, showing them how to use the DoD Architecture Framework (DoDAF) to describe an architecture. Although the guidance was developed under Navy auspices, it is intended to be applicable to any organization. A description of the process guidance and a work breakdown structure (WBS) is provided in the following paragraphs.

2.3.1 Department of the Navy's Architecture Development Process Model (ADPM)

What is the ADPM?

The ADPM is a "roadmap" for the development of enterprise architecture descriptions as documented in the C4ISR Architecture Framework. The ADPM provides a step-by-step approach to developing Framework-compliant architecture descriptions. Using a set of hyperlinked documents, the ADPM provides a product-driven WBS for architecture description development, supplemented by:

- Task descriptions and dependencies (also available for download in a Microsoft Project process template)
- Best practices and lessons learned
- Product examples from various DoD functional areas
- Technical references to the Core Architecture Data Model (CADM) for each product

To accommodate the unique aspects of each architecture initiative (e.g., differences in scope, objectives, and functional application), the ADPM is intentionally generic and functionally independent. Modifications to the template to accommodate specific project needs can easily be made in the Microsoft Project process template.

Recommended approach for navigating through the ADPM

The ADPM is organized hierarchically. High-level tasks are outlined in the table of contents, and users may "drill down" to lower-level tasks from this page. On all lower-level pages, there is a "Return to Table of Contents" link to enable users to easily return to the high-level task listing. It is recommended that users either review the table of contents, or download and review (at a high level) the tasks in the Microsoft Project file to familiarize themselves with the task hierarchy.

Intended audience

The ADPM is intended to support architecture program and project managers from any functional area (personnel, logistics, C4ISR, etc.) who are responsible for overseeing the development of architectures. Its objective is to enable effective oversight and coordination of architectural description development efforts.

Future evolution

The DoN CIO is currently developing the next version of the ADPM. The upcoming version will include:

- Synchronization with the DoDAF, Version 1.0
- Hyperlinked references to a Primer Appendix containing sections for "hands on" developers of architectural products covering areas such as information exchange requirement (IER) development, Operational View to Systems View relation, standardization topics, and examples

How to get the ADPM

The ADPM is available on the DoN CIO Web site: www.doncio.navy.mil; follow the links to Architecture and Standards area. ADPM may be reached directly at: http://www.don-imit.navy.mil/adpm/ADPM%20Files/Architecture%20Development%20Process%20Model.htm

For comments, questions, or to request a CD, please e-mail: ADPM@hq.navy.mil.

2.3.2 ADPM Excerpt

Below is the core task listing from the ADPM. The listing includes task dependencies representing the iterative nature of architecture. Not included in this excerpt are the ADPM hyperlinks, task explanations and definitions, examples, and the CADM product views.

Task #	Outline #	Task Name	Predecessors
1	1	Initiate Architecture Initiative	
2	1.1	Plan the Architecture Initiative	
3	1.1.1	Define the Architecture	
4	1.1.2	Make Architecture Project Plan	
5	1.1.3	Obtain Architecture Project Approval	
6	1.2	Activate Architecture Project	
7	1.2.1	Publicize Architecture Project	
8	1.2.2	Acquire Facilities	
9	1.2.3	Train Architecture Project Team	
10	1.2.4	Architecture Project Control	
11	1.2.5	Motivate Architecture Participants	
12	1.2.6	Track Architecture Project Progress	
13	1.2.7	Revise Architecture Project Plan	
14	2	Build Product Set	
15	2.1	Build Essential Product Set	
16	2.1.1	Produce Overview and Summary Information (AV-1)	
17	2.1.1.1	Produce Planning Guide	
18	2.1.1.1.1	Document the Architecture's Identification Information	
19	2.1.1.1.1.1	Document Architecture Name	
20	2.1.1.1.1.2	Document Participating Organizations	19
21	2.1.1.1.1.3	Document Time Period	19
22	2.1.1.1.2	Document the Purpose of the Architecture	
23	2.1.1.1.2.1	Document Planned Analyses	34

Task #	Outline #	Task Name	Predecessors
24	2.1.1.1.2.2	Identify and Document Analytical Participants	23
25	2.1.1.1.2.3	Identify and Document Planned Architecture-Based Decisions	23
26	2.1.1.1.2.4	Allocate Planned Decisions to Participants	24
27	2.1.1.1.2.5	Identify and Document Expected Results for Planned Architecture	25
28	2.1.1.1.3	Determine and Document Architecture Scope	
29	2.1.1.1.3.1	Determine Product List	
30	2.1.1.1.3.1.1	List Essential Products	22
31	2.1.1.1.3.1.2	Determine Applicable Supporting Products	
32	2.1.1.1.3.1.2.1	Identify and List Supporting Products	22
33	2.1.1.1.3.2	Document Each Product's Time-Based View	29
34	2.1.1.1.4	Identify and Document Architectural Context	
35	2.1.1.1.4.1	Identify and Document the Architecture's Drivers	18
36	2.1.1.1.4.2	Identify and Document Tasking	18
37	2.1.1.1.5	Identify and Document Linkages to Other Architectures	
38	2.1.1.1.5.1	Identify and Document Anticipated Linkages to Other Architectures	34
39	2.1.1.1.5.2	Identify and Document Actual Linkages to Other Architectures	34
40	2.1.1.1.6	Identify and Document Assumptions	34
41	2.1.1.1.7	Identify and Document Constraints	34
42	2.1.1.1.8	Identify and Document Authoritative Sources	29, 34
43	2.1.1.1.9	Document Architecture Findings and Recommendations	431
44	2.1.1.1.10	Determine and Document Tools and File Formats for Architecture Development	28
45	2.1.1.2	Populate the Data Repository Structures in the CADM AV-1 View	
46	2.1.2	Produce Integrated Dictionary (AV-2)	
47	2.1.2.1	Develop Glossary of Terms	
48	2.1.2.1.1	Identify Applicable Terms From Framework Ver 2 (Appendix A) for the Architecture's Product List	19
49	2.1.2.1.2	Determine Additional Applicable Terms	19
50	2.1.2.2	Populate the Data Repository Structures in the CADM AV-2 View	47
51	2.1.3	Produce High-Level Operational Concept Graphic (OV-1)	
52	2.1.3.1	Identify and Document the Product's Time-Based View	33
53	2.1.3.2	Identify Contents of High Level-Operational Concept Graphic	
54	2.1.3.2.1	Identify the Environment in Which Architecture Resides	22
55	2.1.3.2.2	Identify the Architecture's High-Level Components	54
56	2.1.3.2.3	Identify the Actions of, and/or Relationships Between, High-Level Components	55
57	2.1.3.3	Illustrate Contents of High-Level Operational Concept Graphic	
58	2.1.3.3.1	Illustrate the Environment in Which the Architecture Resides	54
59	2.1.3.3.2	Illustrate the High-Level Components as Icons	55
60	2.1.3.3.3	Illustrate the High-Level Actions and/or Relationships as Line Types	56
61	2.1.3.4	Populate the Data Repository Structures in the CADM OV-1 View	53
62	2.1.4	Produce Operational Node Connectivity Description (OV-2)	
63	2.1.4.1	Identify and Document the Product's Time-Based View	33
64	2.1.4.2	Identify Operational Connectivity Diagram Components	
65	2.1.4.2.1	Identify and List Operational Nodes	51
66	2.1.4.2.2	Identify and List Activities of the Architecture	51
67	2.1.4.2.3	Identify and List the Activities Performed by the Operational Nodes	66

Task #	Outline #	Task Name	Predecessors
68	2.1.4.2.4	Identify and List the Needlines Between the Operational Nodes and Operational Elements	65, 67
69	2.1.4.2.5	Identify and List the Information Exchange Requirements (IERs) for Each Needline	68
70	2.1.4.2.6	Identify and List the Characteristics of Each IER	69
71	2.1.4.3	Illustrate Operational Connectivity Diagram	
72	2.1.4.3.1	Illustrate the Operational Nodes as Icons	65
73	2.1.4.3.2	Illustrate the Operational Elements as Icons	66
74	2.1.4.3.3	Illustrate the Needlines by Drawing Line Types Between the Icons	65, 70, 72
75	2.1.4.3.4	Annotate the IERs on the Needlines	69, 74
76	2.1.4.4	Populate the Data Repository Structures in the CADM OV-2 View	64
77	2.1.5	Produce Operational Information Exchange Matrix (OV-3)	
78	2.1.5.1	Identify and Document the Product's Time -Based View	33
79	2.1.5.2	Identify and List Information Exchange Requirements	
80	2.1.5.2.1	Reference and List the Operational Nodes From OV-2	
81	2.1.5.2.2	Identify and List Each Operational Node's IERs From a Consuming Perspective	
82	2.1.5.2.2.1	Identify and List the Operational Node's Consuming Activities	62
83	2.1.5.2.3	Identify and List the Corresponding Producing Activities Within the Operational Nodes	
84	2.1.5.2.4	Identify and List Each IER's Characteristics	62
85	2.1.5.2.5	Identify Additional Activities and/or Missions Applicable to the IER List	62
86	2.1.5.3	Compile Nodes and IERs in Matrix Format	
87	2.1.5.3.1	List IERs	
88	2.1.5.3.1.1	List Producing Operational Nodes	
89	2.1.5.3.1.1.1	List Producing Activities	
90	2.1.5.3.1.2	List Consuming Operational Nodes	
91	2.1.5.3.1.2.1	List Consuming Activities	82
92	2.1.5.3.1.3	List IER Characteristics	84
93	2.1.5.3.1.4	List Additional Activities and/or Missions	85
94	2.1.5.3.1.5	Populate the Data Repository Structures in the CADM OV-3 View	
95	2.1.6	Produce Systems Interface Description (SV-1)	
96	2.1.6.1	Identify and Document the Product's Time -Based View	33
97	2.1.6.2	Identify and Document the Product's Perspective (Internodal, Intranodal, or Intrasystem)	22
98	2.1.6.3	Identify Systems Interface Description Components	
99	2.1.6.3.1	Reference OV-2	
100	2.1.6.3.1.1	List Operational Nodes	65
101	2.1.6.3.1.2	List Needlines	68
102	2.1.6.3.2	Identify Systems Nodes	
103	2.1.6.3.2.1	Convert Operational Nodes to Systems Nodes	
104	2.1.6.3.2.1.1	Identify Resources That (Will) Implement Operational Node	100
105	2.1.6.3.3	Identify System Interfaces	
106	2.1.6.3.3.1	Convert Needlines to System Interfaces	
107	2.1.6.3.3.1.1	Identify System Interfaces That (Will) Implement Each Needline	101
108	2.1.6.3.4	Identify Capabilities of the Systems Nodes	104
109	2.1.6.4	Illustrate Systems Interface Description	

Task #	Outline #	Task Name	Predecessors
110	2.1.6.4.1	Illustrate the Systems Nodes as Icons	
111	2.1.6.4.1.1	Illustrate Resources Within Systems Nodes as Icons	104
112	2.1.6.4.2	Illustrate the System Interfaces by Drawing Types of Lines Between the Icons	
113	2.1.6.4.2.1	Annotate the Interface on Each System Interface Line	107
114	2.1.6.4.3	Illustrate System Capabilities	
115	2.1.6.4.3.1	Augment Capability Illustrations With Annotated Text	108
116	2.1.6.5	Populate the Data Repository Structures in the CADM SV-1 View	
117	2.1.7	Produce Technical Standards Profile (TV-1)	
118	2.1.7.1	Identify and Document the Product's Time -Based View	33
119	2.1.7.2	Identify the Architecture's Rules for its Implementation and Operation	
120	2.1.7.2.1	Identify and List Applicable Standards	28, 40, 41, 42
121	2.1.7.2.2	Identify and List Applicable Guidance	28, 40, 41, 42
122	2.1.7.2.3	Identify and List Applicable Policy	28, 40, 41, 42
123	2.1.7.3	Tailor Rules (Within Allowed Constraints)	28, 40, 41, 42
124	2.1.7.4	Populate the Data Repository Structures in the CADM TV-1 View	
125	2.2	Build Supporting Product Set	
126	2.2.1	Produce Organizational Relationships Chart (OV-4)	
127	2.2.1.1	Identify and Document the Product's Time -Based View	33
128	2.2.1.2	Identify Organizational Relationships Chart Components	
129	2.2.1.2.1	Identify Applicable Organizational Components	22, 29, 42
130	2.2.1.2.2	Identify Applicable Resources	22, 29, 42
131	2.2.1.2.3	Identify Applicable Relationships Between Organizations and/or Resources	
132	2.2.1.2.3.1	Characterize the Relationships Between Organizations and/or Resources	129, 130
133	2.2.1.3	Illustrate the Organizational Relationships Chart Components	
134	2.2.1.3.1	Illustrate Organizational Components as Icons	129
135	2.2.1.3.2	Illustrate Resources as Icons	130
136	2.2.1.3.3	Illustrate Relationships Between Organizations and/or Resources as Line Types	132
137	2.2.1.4	Populate the Data Repository Structures in the CADM OV-4 View	
138	2.2.2	Produce Operational Activity Model (OV-5)	
139	2.2.2.1	Identify and Document the Product's Time -Based View	33
140	2.2.2.2	Reference Existing Architecture Products	
141	2.2.2.2.1	Reference Operational Information Exchange Matrix (OV-3)	
142	2.2.2.2.1.1	Identify Applicable Information Consuming Activities	82
143	2.2.2.2.1.2	Identify Applicable Information Producing Activities	
144	2.2.2.2.2	Reference Operational Node Connectivity Description (OV-2)	
145	2.2.2.2.2.1	Identify Applicable IERs	69
146	2.2.2.2.3	Reference Overview and Summary Information (AV-1)	
147	2.2.2.2.3.1	Identify Applicable External Architecture Links	38,39
148	2.2.2.3	Identify Additional Activities	
149	2.2.2.3.1	Conduct Activity Modeling Session(s)	140
150	2.2.2.3.2	Solicit Input from Subject Matter Experts (SMEs)	140
151	2.2.2.4	Analyze Activities	148
152	2.2.2.5	Illustrate Activity Model (IDEF0 Format Recommended)	151
153	2.2.2.6	Validate Activity Model	

Task #	Outline #	Task Name	Predecessors
154	2.2.2.6.1	Confirm Activity Model Accuracy and Completeness with SMEs	152
155	2.2.2.7	Populate the Data Repository Structures in the CADM OV-5 View	
156	2.2.3	Produce Operational Rules Model (OV-6a)	
157	2.2.3.1	Identify and Document the Product's Time -Based View	33
158	2.2.3.2	Reference Existing Architecture Products	
159	2.2.3.2.1	Reference Logical Data Model (OV-7)	
160	2.2.3.2.1.1	Identify Applicable Relationship Types	201
161	2.2.3.2.1.2	Identify Applicable Attribute Types	201
162	2.2.3.2.1.3	Identify Applicable Domain Values	214
163	2.2.3.2.2	Reference Operational Activity Model (OV-5)	
164	2.2.3.2.2.1	Identify Applicable Activities	138
165	2.2.3.3	Select Formal Language	
166	2.2.3.4	Derive and Document Operational Rules	
167	2.2.3.4.1	Derive and Document Operational Structural Assertion Rules Set	159, 163, 165
168	2.2.3.4.2	Derive and Document Operational Action Assertion Rules Set	159, 163, 165
169	2.2.3.4.3	Derive and Document Operational Derivation Assertion Rules Set	159, 163, 165
170	2.2.3.5	Register Rules in Integrated Dictionary	166
171	2.2.3.6	Populate the Data Repository Structures in the CADM OV-6a View	
172	2.2.4	Produce Operational State Transition Description (OV-6b)	
173	2.2.4.1	Identify and Document the Product's Time -Based View	33
174	2.2.4.2	Reference Existing Architecture Products	
175	2.2.4.2.1	Reference Operational Activity Model (OV-5)	
176	2.2.4.2.1.1	Identify Applicable Events	138
177	2.2.4.2.1.2	Identify Applicable Business Processes	138
178	2.2.4.2.1.3	Identify State Transitions	
179	2.2.4.3	Illustrate State Transitions	
180	2.2.4.3.1	Depict States as Icons	178
181	2.2.4.3.2	Depict Transitions as Lines	178
182	2.2.4.3.3	Annotate Transition Lines	
183	2.2.4.3.3.1	Annotate Event Names	178
184	2.2.4.3.3.2	Annotate Action Names	178
185	2.2.4.3.3.3	Annotate Result Names	178
186	2.2.4.4	Populate the Data Repository Structures in the CADM OV-6b View	
187	2.2.5	Produce Operational Event-Trace Description (OV-6c)	
188	2.2.5.1	Identify and Document the Product's Time -Based View	33
189	2.2.5.2	Reference Existing Architecture Products	
190	2.2.5.2.1	Reference Operational State Transition Description (OV-6b)	
191	2.2.5.2.1.1	Identify Applicable Events	172
192	2.2.5.2.2	Reference Operational Node Connectivity Description (OV-2)	
193	2.2.5.2.2.1	Identify Applicable Operational Nodes	62
194	2.2.5.2.2.2	Identify Applicable IERs	62
195	2.2.5.3	Relate Operational Nodes to Events	
196	2.2.5.3.1	Sequence Events	191, 192
197	2.2.5.4	Illustrate Operational Event-Trace Description	
198	2.2.5.4.1	Depict Related Nodes and Events	196
199	2.2.5.4.2	Depict Sequential Relationship Between Nodes and Events	196
200	2.2.5.5	Populate the Data Repository Structures in the CADM OV-6c View	

Task #	Outline #	Task Name	Predecessors
201	2.2.6	Produce Logical Data Model (OV-7)	
202	2.2.6.1	Identify and Document the Product's Time -Based View	33
203	2.2.6.2	Reference Existing Architecture Products	
204	2.2.6.2.1	Refe rence Operational Information Exchange Matrix (OV-3)	
205	2.2.6.2.1.1	Identify Applicable IERs	69
206	2.2.6.2.2	Reference Operational Node Connectivity Description (OV-2)	
207	2.2.6.2.2.1	Identify Applicable Operational Nodes	65
208	2.2.6.2.3	Reference Overview and Summary Information (AV-1)	
209	2.2.6.2.3.1	Identify Applicable External Architecture Links	38,39
210	2.2.6.3	Identify Additional Data Requirements	
211	2.2.6.3.1	Conduct Data Modeling Session(s)	203
212	2.2.6.3.2	Solicit Input from SMEs	203
213	2.2.6.4	Analyze Data Requirements	203, 210
214	2.2.6.5	Normalize Data Model	213
215	2.2.6.6	Illustrate Data Model (IDEF1X Format Recommended)	214
216	2.2.6.7	Develop Subject Area Views	
217	2.2.6.7.1	Illustrate and Document Subject Area Views (IDEF1X Format Recommended)	215
218	2.2.6.8	Validate Data Model	
219	2.2.6.8.1	Confirm Data Model Accuracy and Completeness with SMEs	217
220	2.2.6.9	Update AV1, OV-2, OV-3, and SV-1 to Incorporate Additional Activities	
221	2.2.6.10	Populate the Data Repository Structures in the CADM OV-7 View	
222	2.2.7	Produce Systems Communications Description (SV-2)	
223	2.2.7.1	Identify and Document the Product's Time -Based View	33
224	2.2.7.2	Determine and Document View (Internodal or Intranodal)	16
225	2.2.7.3	Reference Existing Architecture Products	
226	2.2.7.3.1	Reference System Interface Description (SV-1)	
227	2.2.7.3.1.1	Identify Applicable System Nodes	103
228	2.2.7.3.1.2	Identify Applicable System Interfaces	106
229	2.2.7.3.1.3	Identify Physical Characteristics of Each Interface Between System Nodes	108
230	2.2.7.4	Illustrate Systems Communications Description	
231	2.2.7.4.1	Depict System Nodes as Icons	227
232	2.2.7.4.2	Depict System Interfaces as Line Types	228
233	2.2.7.4.3	Annotate Interface Physical Characteristics on Lines	229
234	2.2.7.4.4	Populate the Data Repository Structures in the CADM SV-2 View	
235	2.2.8	Produce Systems -Systems Matrix (SV-3)	
236	2.2.8.1	Identify and Document the Product's Time -Based View	33
237	2.2.8.2	Reference Existing Architecture Products	
238	2.2.8.2.1	Reference System Interface Description (SV-1)	
239	2.2.8.2.1.1	Identify Applicable Systems	104
240	2.2.8.2.1.2	Identify Applicable System Interfaces	107
241	2.2.8.3	Derive System-to-System Relationships From Interfaces	238
242	2.2.8.4	Characterize each System-to-System Relationship (Planned, Existing, Potential, etc.)	241
243	2.2.8.5	Illustrate Systems -Systems Matrix	
244	2.2.8.5.1	List Systems on Matrix Axes	239

Task #	Outline #	Task Name	Predecessors
245	2.2.8.5.2	Depict System to System Relationship Characteristics in Intersecting Matrix Cells	241, 242, 244
246	2.2.8.5.3	Populate the Data Repository Structures in the CADM SV-3 View	
247	2.2.9	Produce Systems Functionality Description (SV-4)	
248	2.2.9.1	Identify and Document the Product's Time -Based View	33
249	2.2.9.2	Reference Existing Architecture Products	
250	2.2.9.2.1	Reference Systems Interface Description (SV-1)	
251	2.2.9.2.1.1	Identify Applicable Systems	104
252	2.2.9.2.1.2	Identify Applicable Systems Nodes	104
253	2.2.9.2.1.3	Identify Applicable System Interfaces	107
254	2.2.9.2.2	Reference Operational Information Exchange Matrix (OV-3)	
255	2.2.9.2.2.1	Identify Applicable Operational Nodes	
256	2.2.9.2.3	Reference Operational Activity Model (OV-5)	
257	2.2.9.2.3.1	Identify Applicable Activities	142, 143
258	2.2.9.3	Allocate Activities to Operational Nodes	254, 256
259	2.2.9.4	Associate Operational Activities to Systems or System Functions	250, 258
260	2.2.9.5	Identify and Define the Data Flows for each System Interface	258, 259
261	2.2.9.6	Illustrate Systems Functionality Description (SV-4)	
262	2.2.9.6.1	Depict Systems as Icons	259
263	2.2.9.6.2	Depict System Functions as Icons	259
264	2.2.9.6.3	Depict Data Flows as Line Types	260
265	2.2.9.6.4	Annotate Data Flow Line Types with Data Flow Descriptions	264
266	2.2.9.6.5	Populate the Data Repository Structures in the CADM SV-4 View	
267	2.2.10	Produce Operational Activity to Systems Function Traceability Matrix (SV-5)	
268	2.2.10.1	Identify and Document the Product's Time -Based View	33
269	2.2.10.2	Reference Existing Architecture Products	
270	2.2.10.2.1	Reference Operational Activity Model (OV-5)	
271	2.2.10.2.1.1	Identify Applicable Activities	140, 148
272	2.2.10.2.2	Reference System Interface Description (SV-1)	
273	2.2.10.2.2.1	Identify Applicable Systems	104
274	2.2.10.2.2.2	Identify Applicable System Interfaces	107
275	2.2.10.2.3	Reference Systems Functionality Description (SV-4)	
276	2.2.10.2.3.1	Identify Associated Operational Activities and their Systems or Systems Functions	259
277	2.2.10.3	Illustrate Operational Activity to System Function Traceability Matrix	
278	2.2.10.3.1	List Systems Functions on One Axis of Matrix	272, 275
279	2.2.10.3.2	List Operational Activities on Other Axis of Matrix	270
280	2.2.10.3.3	Indicate Mappings in Cell Intersections	278, 279
281	2.2.10.3.4	Populate the Data Repository Structures in the CADM SV-5 View	
282	2.2.11	Produce Systems Data Exchange Matrix (SV-6)	
283	2.2.11.1	Identify and Document the Product's Time -Based View	33
284	2.2.11.2	Reference Existing Architecture Products	
285	2.2.11.2.1	Reference Systems Interface Description (SV-1)	
286	2.2.11.2.1.1	Identify Applicable Systems	104
287	2.2.11.2.1.2	Identify Applicable Systems Nodes	104
288	2.2.11.2.1.3	Identify Applicable System Interfaces	107

Task #	Outline #	Task Name	Predecessors
289	2.2.11.2.2	Reference Systems Functionality Description (SV-4)	
290	2.2.11.2.2.1	Identify Applicable Associated Operational Activities and Their Systems or Systems Functions	259
291	2.2.11.2.2.2	Identify Applicable Data Flows (Input/Output) for Each System Interface	260
292	2.2.11.3	Illustrate Systems Data Exchange Matrix	
293	2.2.11.3.1	List System Elements on One Axis	285
294	2.2.11.3.2	List System Functions With Their Associated Data Flows (Inputs/Outputs) on Other Axis	289
295	2.2.11.3.3	Characterize Mappings in Cell Intersections	293, 294
296	2.2.11.3.4	Populate the Data Repository Structures in the CADM SV-6 View	
297	2.2.12	Produce System Performance Parameters Matrix (SV-7)	
298	2.2.12.1	Reference Existing Architecture Products	
299	2.2.12.1.1	Reference Systems Interface Description (SV-1)	
300	2.2.12.1.1.1	Identify Applicable Systems	104
301	2.2.12.1.1.2	Identify Applicable Systems Nodes	104
302	2.2.12.1.1.3	Identify Applicable Capabilities	108
303	2.2.12.1.2	Identify and Document Current and Future Time Periods	299
304	2.2.12.1.3	Identify Future Performance Characteristics and Expectations	303
305	2.2.12.2	Illustrate System Performance Parameters Matrix	
306	2.2.12.2.1	List System and System Elements on One Axis	299
307	2.2.12.2.2	List Current and Future Time Periods Across Other Axis	303
308	2.2.12.2.3	Populate the Intersecting Cells With the Current and Future Performance Expectations	304
309	2.2.12.3	Populate the Data Repository Structures in the CADM SV-7 View	
310	2.2.13	Produce Systems Evolution Description (SV-8)	
311	2.2.13.1	Reference Existing Architecture Products	
312	2.2.13.1.1	Reference Operational Node Connectivity Description (OV-2)	
313	2.2.13.1.1.1	Identify Applicable Operational Nodes	
314	2.2.13.1.1.2	Identify Applicable Needlines	69
315	2.2.13.1.2	Reference Systems Interface Description (SV-1)	
316	2.2.13.1.2.1	Identify Applicable Systems	104
317	2.2.13.1.2.2	Identify Applicable Systems Nodes	104
318	2.2.13.1.2.3	Identify Applicable Capabilities (Performance Parameters)	108
319	2.2.13.1.3	Reference Systems Technology Forecast (SV-9)	
320	2.2.13.1.3.1	Identify Applicable Forecast Projections	
321	2.2.13.1.4	Reference Technical Standards Forecast (TV-2)	
322	2.2.13.1.4.1	Identify Applicable Forecast Projections	420
323	2.2.13.2	Determine Need Dates for Systems and Capabilities	311
324	2.2.13.3	Illustrate System Performance Parameters Matrix	
325	2.2.13.3.1	Annotate or Depict (as Icons) Systems/System Elements	315
326	2.2.13.3.2	Depict/Illustrate Timeline	323
327	2.2.13.3.3	Depict Associations Between System/System Elements and the Timeline	315, 323
328	2.2.13.4	Populate the Data Repository Structures in the CADM SV-8 View	
329	2.2.14	Produce Systems Technology Forecast (SV-9)	
330	2.2.14.1	Identify Applicable Technologies and Capabilities	16

Task #	Outline #	Task Name	Predecessors
331	2.2.14.2	Determine Appropriate Time Intervals (e.g., Short-, Mid- and Long-Term)	330
332	2.2.14.3	Identify Industry Trends Pertaining to each Capability for Each Time Frame	331
333	2.2.14.4	Identify Prediction Confidence Factors for Each Trend	332
334	2.2.14.5	Perform Impact Analysis	
335	2.2.14.5.1	Identify and Document Impacts of Technology Predictions to Architecture	333
336	2.2.14.6	Illustrate System Technology Forecast	
337	2.2.14.6.1	List Technologies and Capabilities on one Axis	330
338	2.2.14.6.2	List Timeframes Across Other Axis	331
339	2.2.14.6.3	Populate the Intersecting Cells with Impact Analysis Results	
340	2.2.14.7	Populate the Data Repository Structures in the CADM SV-9 View	
341	2.2.15	Produce Systems Rules Model (SV-10a)	
342	2.2.15.1	Identify and Document the Product's Time-Based View	33
343	2.2.15.2	Reference Existing Architecture Products	
344	2.2.15.2.1	Reference Operational Rules Model (OV-6a)	342
345	2.2.15.2.1.1	Identify Applicable Operational Structural Assertion Rules	167
346	2.2.15.2.1.2	Identify Applicable Operational Action Assertion Rules	168
347	2.2.15.2.1.3	Identify Applicable Operational Derivation Assertion Rules	169
348	2.2.15.2.2	Reference Systems Functionality Description (SV-4)	
349	2.2.15.2.2.1	Identify Applicable Procedures (e.g., Derivation Algorithms)	259
350	2.2.15.3	Select Formal Language	
351	2.2.15.4	Derive and Document System Rules	
352	2.2.15.4.1	Derive and Document System Structural Assertion Rules Set	345
353	2.2.15.4.2	Derive and Document System Action Assertion Rules Set	346
354	2.2.15.4.3	Derive and Document System Derivation Assertion Rules Set	347
355	2.2.15.5	Register Rules in Integrated Dictionary	351
356	2.2.16	Produce Systems State Transition Description (SV-10b)	
357	2.2.16.1	Identify and Document the Product's Time-Based View	33
358	2.2.16.2	Reference Existing Architecture Products	
359	2.2.16.2.1	Reference Systems Functionality Description (SV-4)	
360	2.2.16.2.1.1	Identify Applicable Results of Functions Performed at System Nodes	247
361	2.2.16.2.1.2	Identify State Transitions	247
362	2.2.16.3	Illustrate State Transitions	
363	2.2.16.3.1	Depict Results of Functions Performed at System Nodes as Icons	359
364	2.2.16.3.2	Depict Transitions as Lines	359
365	2.2.16.3.3	Annotate Transition Lines	
366	2.2.16.3.3.1	Annotate Event Names	359
367	2.2.16.3.3.2	Annotate Action Names	359
368	2.2.16.3.3.3	Annotate Result Names	359
369	2.2.16.4	Populate the Data Repository Structures in the CADM SV-10b View	
370	2.2.17	Produce Systems Event-Trace Description (SV-10c)	
371	2.2.17.1	Identify and Document the Product's Time-Based View	33
372	2.2.17.2	Reference Existing Architecture Products	
373	2.2.17.2.1	Reference Systems State Transition Description (SV-10b)	
374	2.2.17.2.1.1	Identify Applicable Results of Functions Performed at System Nodes	360
375	2.2.17.2.2	Reference Systems Interface Description (SV-1)	

Task #	Outline #	Task Name	Predecessors
376	2.2.17.2.2.1	Identify Applicable Systems Nodes	104
377	2.2.17.2.2.2	Identify Applicable System Interface Lines	107
378	2.2.17.3	Relate System Nodes to Events	363
379	2.2.17.4	Sequence Events	378
380	2.2.17.5	Illustrate System Event-Trace Description	
381	2.2.17.5.1	Depict Related System Nodes and Events	378
382	2.2.17.5.2	Depict Sequential Relationship Between System Nodes and Events	379
383	2.2.17.5.3	Populate the Data Repository Structures in the CADM SV-10c View	
384	2.2.18	Produce Physical Schema (SV-11)	
385	2.2.18.1	Identify and Document the Product's Time -Based View	33
386	2.2.18.2	Produce Physical Data Model	
387	2.2.18.2.1	Reference Existing Architecture Products	
388	2.2.18.2.1.1	Reference Logical Data Model (OV-7)	
389	2.2.18.2.1.1.1	Identify Applicable Entity Types	214
390	2.2.18.2.1.1.2	Identify Applicable Attribute Types	214
391	2.2.18.2.1.1.3	Identify Applicable Relationship Type	214
392	2.2.18.2.1.1.4	Identify Applicable Domain Values	214
393	2.2.18.2.1.2	Reference Operational Rules Model (OV-6a)	
394	2.2.18.2.1.2.1	Identify Applicable Operational Rules	166
395	2.2.18.2.1.3	Reference Systems Rules Model (SV-10a)	
396	2.2.18.2.1.3.1	Identify Applicable System Rules	351
397	2.2.18.2.1.4	Reference System Performance Parameters Matrix (SV-7)	
398	2.2.18.2.1.4.1	Identify Applicable Performance Parameters	308
399	2.2.18.2.1.5	Reference Operational Node Connectivity Description (OV-2)	
400	2.2.18.2.1.5.1	Identify Applicable IER Characteristics	70
401	2.2.18.2.1.6	Reference Systems Interface Description (SV-1)	
402	2.2.18.2.1.6.1	Identify Applicable Systems	104
403	2.2.18.2.1.6.2	Identify Applicable System Elements	104
404	2.2.18.2.1.6.3	Identify Applicable Systems Interfaces	108
405	2.2.18.2.1.6.4	Identify Applicable System Nodes	104
406	2.2.18.2.1.7	Reference Overview and Summary Information (AV-1)	
407	2.2.18.2.1.7.1	Identify Initial Tool Selection	44
408	2.2.18.2.2	Reassess Tool Selection Decision from AV-1	
409	2.2.18.2.2.1	Confirm or Modify Physical Data Model Tool Choice	387, 407
410	2.2.18.2.3	Develop Physical Database Design	
411	2.2.18.2.3.1	Develop Business System Design	
412	2.2.18.2.3.1.1	Design System Structure	387, 407
413	2.2.18.2.3.1.2	Design Preliminary Data Structures	412
414	2.2.18.2.3.1.3	Design Procedures	412, 419, 424
415	2.2.18.2.4	Populate the Data Repository Structures in the CADM SV-11 View	
416	2.2.19	Produce Technical Standards Forecast (TV-2)	
417	2.2.19.1	Identify and Characterize Standards	
418	2.2.19.1.1	Determine Appropriate Time Intervals (e.g., Short-, Mid- and Long-Term)	17
419	2.2.19.1.2	Identify Applicable Service Areas	17
420	2.2.19.1.3	Identify Current, Emerging, and Predicted Standards for Specific Time Intervals (Forecast Projections)	419

Task #	Outline #	Task Name	Predecessors
421	2.2.19.1.4	Identify Known or Predicted Dates of Obsolescence of Standards	420
422	2.2.19.1.5	Identify Prediction Confidence Factors for Each Prediction	421
423	2.2.19.2	Perform Impact Analysis	
424	2.2.19.2.1	Identify and Document Impacts of Standards Predictions to Architecture	417
425	2.2.19.3	Illustrate Standards Technology Forecast	
426	2.2.19.3.1	List Service Areas on One Axis	419
427	2.2.19.3.2	List Time Frames across Other Axis	418
428	2.2.19.3.3	Populate the Intersecting Cells with Name of Standard(s)	420
429	2.2.19.3.4	Characterize the Impact Analysis Results in the Comment Column	424
430	2.2.19.4	Populate the Data Repository Structures in the CADM TV-2 View	
431	2.3	Milestone - Architecture Version Complete	
432	3	End Architecture Initiative	

2.3.3 Reference

Department of the Navy, *Architecture Development Process Model,* Available: http://www.don-imit.navy.mil/adpm/ADPM%20Files/Architecture%20Development%20Process%20Model.htm

2.4 EXAMPLE ARCHITECTURE USING STRUCTURED ANALYSIS AND UNIFIED MODELING LANGUAGE

2.4.1 Introduction

This section presents illustrative architecture products based on a USCENTCOM architecture developed with structured analysis and then depicts the same information presented in products developed using Unified Modeling Language (UML). Some products are independent of the descriptive technique; these products are discussed only in the structured analysis example products subsection and are not repeated in the UML example products subsection. These common products are listed in section 2.4.2.

The architecture products presented here are notional and are intended to illustrate both products developed in a structured analysis approach and products developed using UML. The products were developed under the C4ISR Architecture Framework Version 2.0 and have been modified slightly to be consistent with DoD Architecture Framework (DoDAF) Version 1.0 for demonstration purposes. The products are a composite of data extracted and paraphrased from the "USCENTCOM Objective Architecture Concerning Targeting" and the "C4ISR Mission Assessment" supplemented with notional examples for those products not developed within the two references.[1] Most of the figures are extracts, not complete architecture sets, and represent portions of the USCENTCOM targeting process, circa 1998. Clearly, USCENTCOM's targeting procedures, systems, and processes have undergone significant changes since 1998. The figures are representative examples of architecture products and should not be used for content.

The USCENTCOM architecture addressed *Conduct Joint Force Targeting*. A decomposition of the activity for *Conduct Joint Force Targeting* is shown in **Figure 2.4-1**. The context for the architecture is the air tasking cycle that produces a daily Air Tasking Order (ATO). An ATO directs the air operations for a 24-hour period and is based on mission objectives, targets, and resources available. The effectiveness of the air operations against enemy targets is assessed, and feedback is provided into the ATO development process. Targets are then included for re-strike either immediately or during a subsequent cycle.

Some of the products focus on *Conduct Combat Assessment*. *Conduct Combat Assessment* is defined as "to determine the overall effectiveness of Service, joint, and multinational attacks employed in the theater, as it relates to the joint force commander's (JFC) campaign objectives.[2] The objective of combat assessment is to identify recommendations for the course of military operations."[3] As shown in **Figure 2.4-2**, *Conduct Combat Assessment* is composed of three sub-activities: *Conduct Battle Damage Assessment (BDA), Conduct Munitions Effects Assessment (MEA),* and *Recommend Restrike.* (Definitions are provided in section 2.4.3.7.) These activities are often conducted simultaneously.

[1] Figures designated as notional architecture products were developed specifically for this Deskbook.

[2] CJCS Manual 3500.04B, *Universal Joint Task List*, July 1,2002, p.B-C-B-48

[3] Joint Publication 1-02, *DoD Dictionary of Military Terms*, April 12, 2001, p. 76.

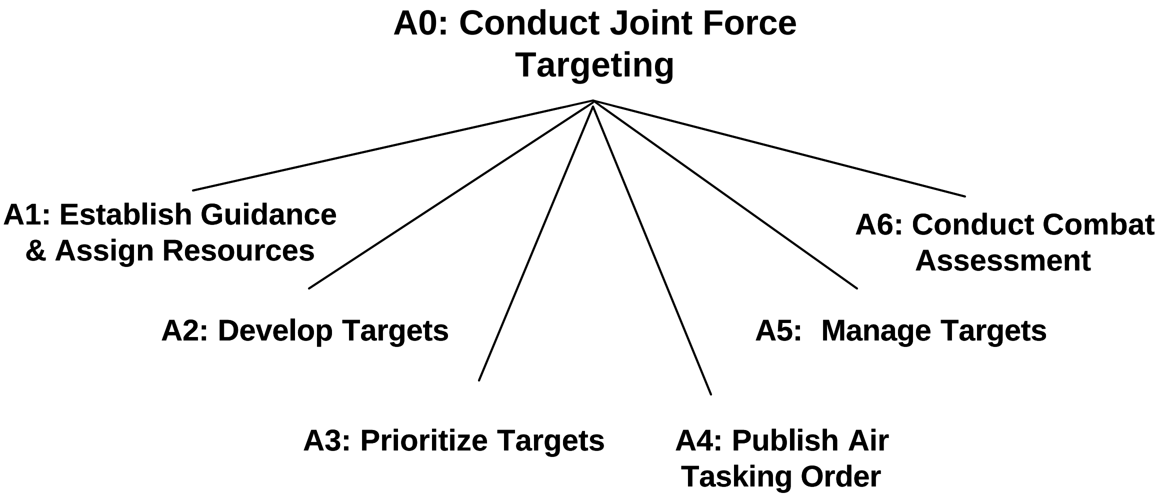

A0: Conduct Joint Force Targeting
A1: Establish Guidance & Assign Resources
A2: Develop Targets
A3: Prioritize Targets
A4: Publish Air Tasking Order
A5: Manage Targets
A6: Conduct Combat Assessment

Figure 2.4-1. Activity Decomposition for *Conduct Joint Force Targeting*

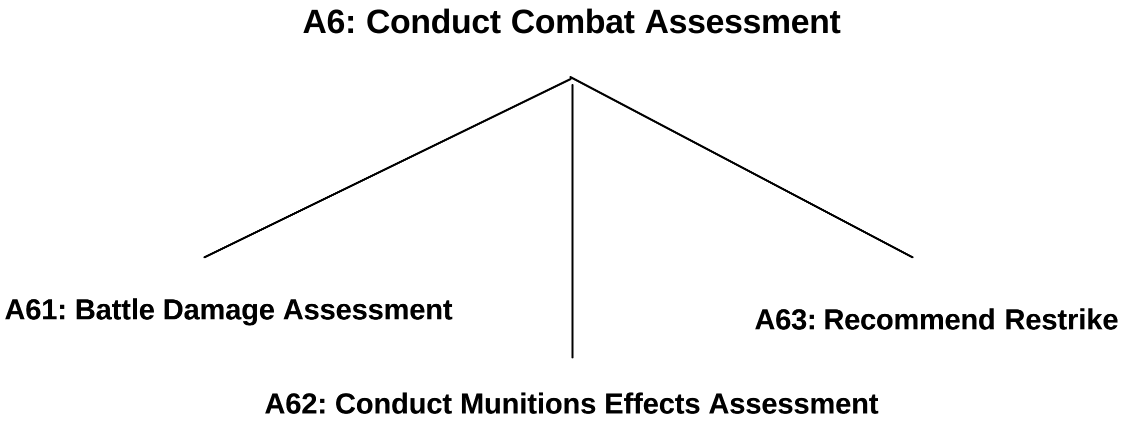

A6: Conduct Combat Assessment
A61: Battle Damage Assessment
A62: Conduct Munitions Effects Assessment
A63: Recommend Restrike

Figure 2.4-2. Activity Decomposition for *Conduct Combat Assessment*

2.4.2 Common Products

This section lists architecture products that are independent of the descriptive technique used. Common products include:

- AV-1: Overview and Summary Information
- AV-2: Integrated Dictionary
- OV-1: High-Level Operational Concept Graphic
- OV-3: Operational Information Exchange Matrix
- OV-6a: Operational Rules Model
- OV-6b: Operational State Transition Description
- OV-6c: Operational Event-Trace Description
- SV-2: Systems Communications Description

- SV-3: Systems-Systems Matrix
- SV-5: Operational Activity to Systems Function Traceability Matrix
- SV-6: Systems Data Exchange Matrix
- SV-7: Systems Performance Parameters Matrix
- SV-8: Systems Evolution Description
- SV-9: Systems Technology Forecast
- SV-10a: Systems Rules Model
- SV-10b: Systems State Transition Description
- SV-10c: Systems Event-Trace Description
- TV-1: Technical Standards Profile
- TV-2: Technical Standards Forecast

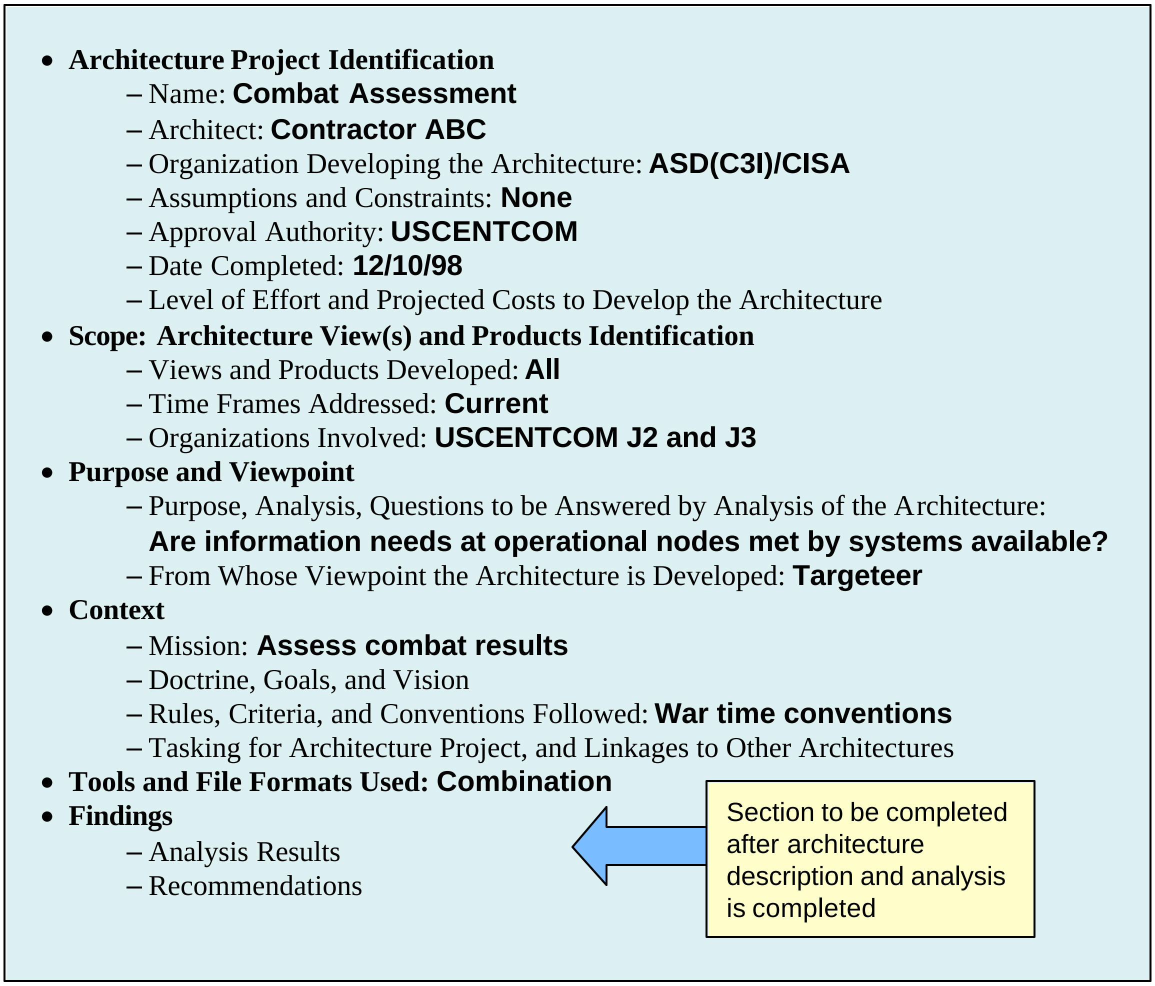
These common products are discussed in the section 2.4.3 (structured analysis) and are not repeated in section 2.4.4 (UML).

2.4.3 Example Products Based on Structured Analysis Techniques

This section presents architecture products based on the structured analysis tools and diagramming techniques. This section also discusses common products that are independent of the descriptive technique.

2.4.3.1 Overview and Summary Information (AV-1)

This product consists of textual information. The information presented in the Notional Overview and Summary Information (AV-1) in **Figure 2.4-3** focuses on *Conduct Combat Assessment*. This notional AV-1 product includes information about the identification, purpose and viewpoint, scope, and context of the architecture. These areas need to be well defined before any other architecture products are developed. The findings portion is completed after the other architecture products have been completed and after the needed analysis has been conducted. AV-1 is independent of the descriptive technique.



- **Architecture Project Identification**
  - – Name: **Combat Assessment**
  - – Architect: **Contractor ABC**
  - – Organization Developing the Architecture: **ASD(C3I)/CISA**
  - – Assumptions and Constraints: **None**
  - – Approval Authority: **USCENTCOM**
  - – Date Completed: **12/10/98**
  - – Level of Effort and Projected Costs to Develop the Architecture
- **Scope: Architecture View(s) and Products Identification**
  - – Views and Products Developed: **All**
  - – Time Frames Addressed: **Current**
  - – Organizations Involved: **USCENTCOM J2 and J3**
- **Purpose and Viewpoint**
  - – Purpose, Analysis, Questions to be Answered by Analysis of the Architecture: **Are information needs at operational nodes met by systems available?**
  - – From Whose Viewpoint the Architecture is Developed: **Targeteer**
- **Context**
  - – Mission: **Assess combat results**
  - – Doctrine, Goals, and Vision
  - – Rules, Criteria, and Conventions Followed: **War time conventions**
  - – Tasking for Architecture Project, and Linkages to Other Architectures
- **Tools and File Formats Used: Combination**
- **Findings**
  - – Analysis Results
  - – Recommendations

Section to be completed after architecture description and analysis is completed

Figure 2.4-3. Notional Overview and Summary Information (AV-1)

2.4.3.2 Integrated Dictionary (AV-2)

The Integrated Dictionary (AV-2) contains definitions of terms used in the given architecture. It consists of textual definitions in the form of a glossary, a repository of architecture data, their taxonomies, and their metadata. No example AV-2 is provided. AV-2 is independent of the descriptive technique.

2.4.3.3 High-Level Operational Concept Graphic (OV-1)

Figure 2.4-4 is the High-Level Operational Concept Graphic (OV-1) for A0: *Conduct Joint Force Targeting*. This graphic depicts deep operations conducted in the joint operations area forward of the Fire Support Coordination Line (FSCL). The command organizations are those typically found with engaged forces. The Marine force component establishes the Battlefield Coordination Line (BCL) between the forward line of its own forces and the FSCL. The surveillance and reconnaissance platforms are illustrative of those perceived to be most helpful in supporting the targeting activity. Weapon systems are illustrative of theater assets that would be employed beyond the FSCL. A sanctuary is any location outside the joint operations area that supports operations in the joint operations area.[4] OV-1 is independent of the descriptive technique.

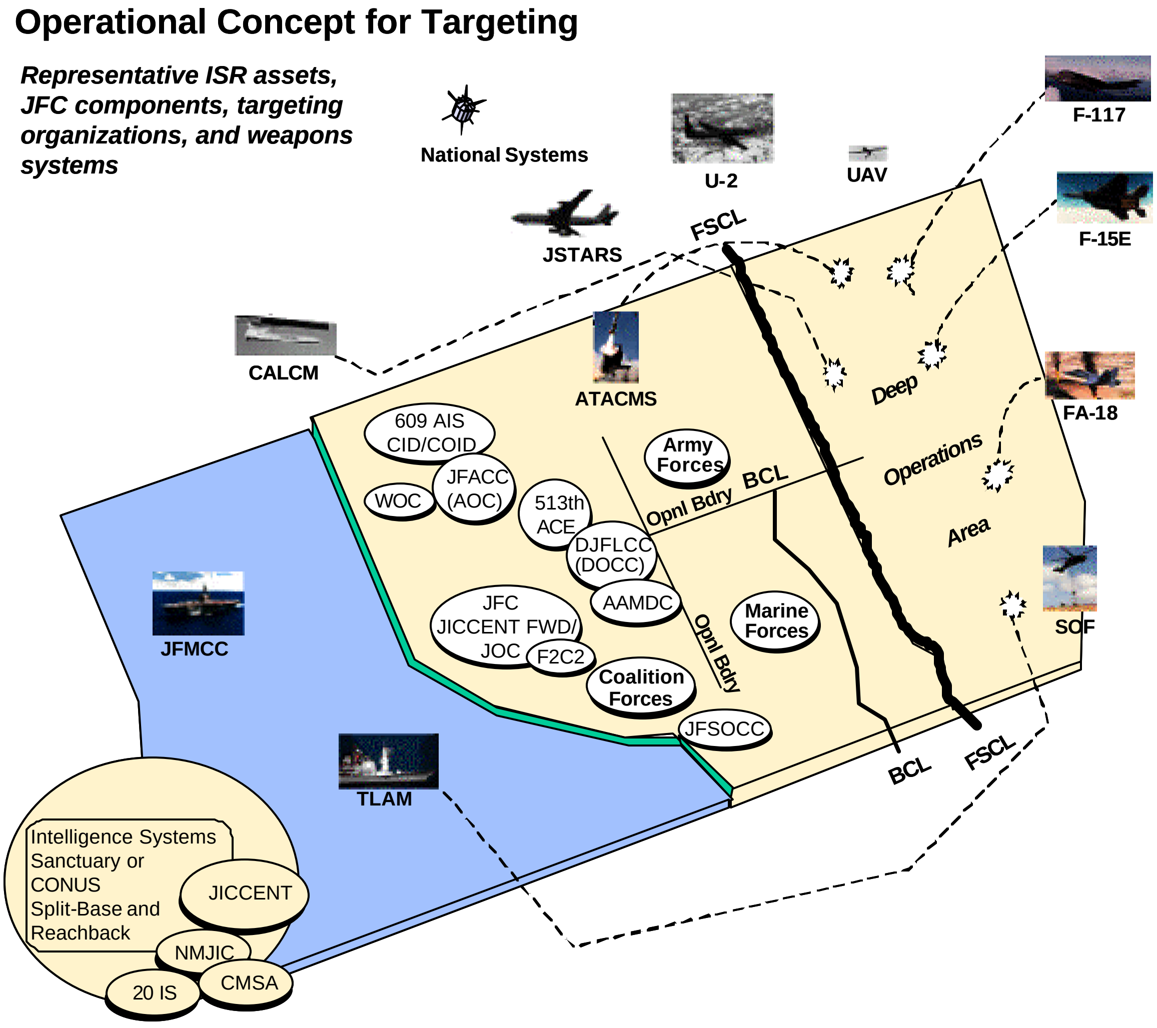


Figure 2.4-4. Example High-Level Operational Concept Graphic (OV-1)

[4] *U.S. Central Command's Objective Architecture Concerning Targeting*, Volume I, March 1998, p. 2-2.

2.4.3.4 Operational Node Connectivity Description (OV-2)

An Operational Node Connectivity Description (OV-2) (see **Figure 2.4-5** depicts the nodes, activities, and information exchanges involved in *Conduct Joint Force Targeting*. The filled rounded boxes represent nodes. The bullets in rectangles between two nodes are information exchanges. The bullets next to a node are activities preformed by that node. Needlines are numbered 1 through 11. The nodes, information exchanges, and activities shown are illustrative and do not represent a complete set.

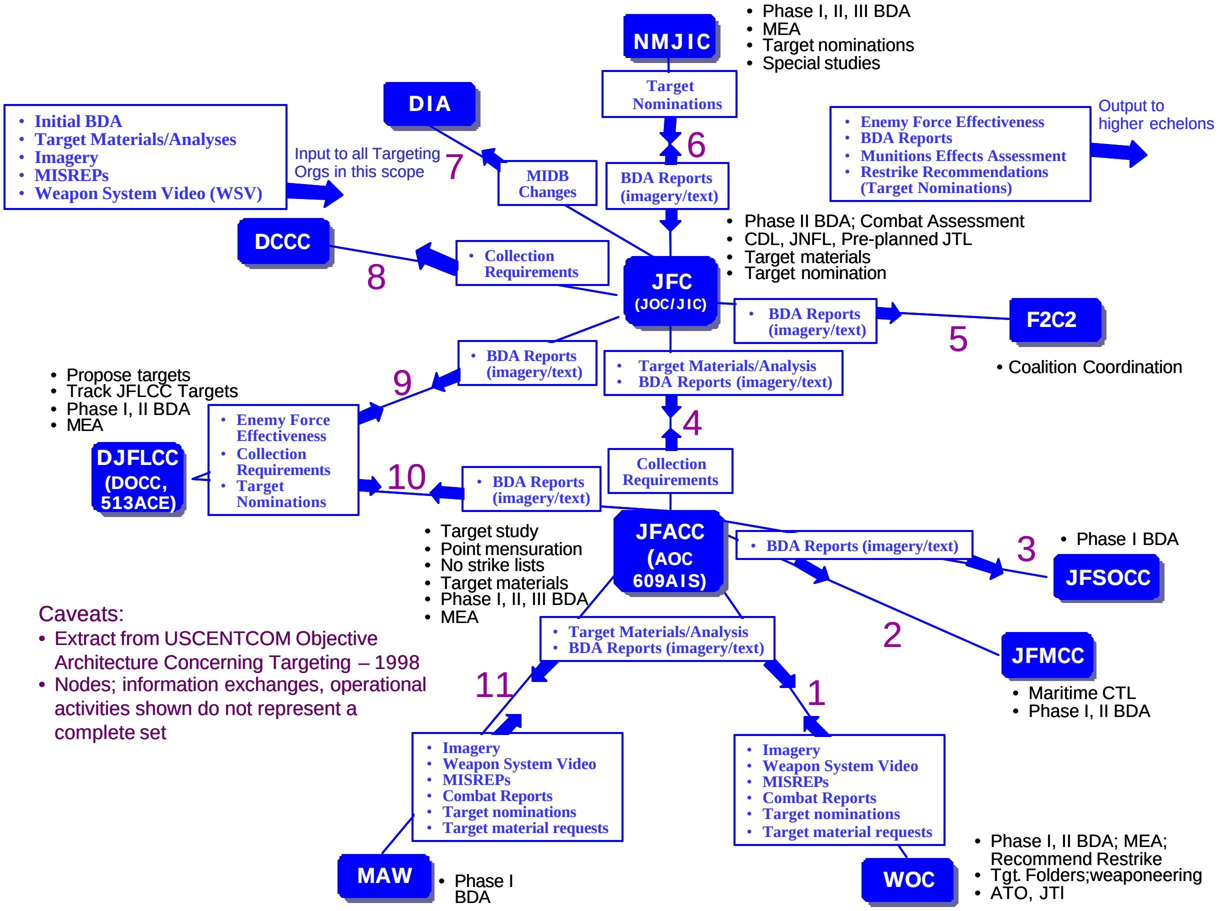

NMJIC
• Phase I, II, III BDA
• MEA
• Target nominations
• Special studies
Target Nominations
6
BDA Reports (imagery/text)
DIA
MIDB Changes
7
Input to all Targeting Orgs in this scope
• Initial BDA
• Target Materials/Analyses
• Imagery
• MISREPs
• Weapon System Video (WSV)
• Enemy Force Effectiveness
• BDA Reports
• Munitions Effects Assessment
• Restrike Recommendations (Target Nominations)
Output to higher echelons
DCCC
8
• Collection Requirements
JFC (JOC/JIC)
• Phase II BDA; Combat Assessment
• CDL, JNFL, Pre-planned JTL
• Target materials
• Target nomination
• BDA Reports (imagery/text)
5
F2C2
• Coalition Coordination
• Propose targets
• Track JFLCC Targets
• Phase I, II BDA
• MEA
9
• BDA Reports (imagery/text)
• Target Materials/Analysis
• BDA Reports (imagery/text)
4
Collection Requirements
DJFLCC (DOCC, 513ACE)
• Enemy Force Effectiveness
• Collection Requirements
• Target Nominations
10
• BDA Reports (imagery/text)
JFACC (AOC 609AIS)
• Target study
• Point mensuration
• No strike lists
• Target materials
• Phase I, II, III BDA
• MEA
• BDA Reports (imagery/text)
3
• Phase I BDA
JFSOCC
2
JFMCC
• Maritime CTL
• Phase I, II BDA
Caveats:
• Extract from USCENTCOM Objective Architecture Concerning Targeting – 1998
• Nodes; information exchanges, operational activities shown do not represent a complete set
• Target Materials/Analysis
• BDA Reports (imagery/text)
11
1
• Imagery
• Weapon System Video
• MISREPs
• Combat Reports
• Target nominations
• Target material requests
• Imagery
• Weapon System Video
• MISREPs
• Combat Reports
• Target nominations
• Target material requests
MAW
• Phase I BDA
WOC
• Phase I, II BDA; MEA; Recommend Restrike
• Tgt. Folders;weaponeering
• ATO, JTI

Figure 2.4-5. Example Operational Node Connectivity Description (OV-2)

2.4.3.5 Operational Information Exchange Matrix (OV-3)

An Operational Information Exchange Matrix (OV-3) describes the information exchanges that support the operational needs. Three notional information exchanges associated with *Conduct Combat Assessment* are presented in **Figure 2.4-6**. OV-3 is independent of the descriptive technique.

Needline Identifier	Information Exchange Identifier	Information Element Description					Producer		Consumer		Nature of Transaction				
		Information Element Name and Identifier	Content	Scope	Accuracy	Language	Sending Op Node Name and Identifier	Sending Op Activity Name and Identifier	Receiving Op Node Name and Identifier	Receiving Op Activity Name and Identifier	Mission / Scenario UJTL or METL	Transaction Type	Triggering Event	Interoperability Level Required	Criticality
1	WOC-JFAC C1	BDA Report	Report on Battle Damage	Theater	1 Day	English	WOC	Conduct Battle Damage Assessment	JFACC	Conduct Munitions Effects Assessment	Combat Assessment	Collaborate	Air Strike 072200, 0615am	2A	High
1	WOC-JFAC C2	Target Nominations	Report on Possible Targets	Theater	2 Hours	English	WOC	Recommend Restrike	JFACC	Request Target Materials	Combat Assessment	Direct	AitTO XX, 072300	1B	High
2															
…															
11	MAW-JFAC C1	BDA Report	Report on Battle Damage	Theater	1 Day	English	MAW	Conduct Battle Damage Assessment	JFACC	Conduct Munitions Effects Assessment	Combat Assessment	Collaborate	Air Strike 072200, 0615am	2A	High

Figure 2.4-6. Notional Operational Information Exchange Matrix (OV-3)

2.4.3.6 Organizational Relationships Chart (OV-4)

Figure 2.4-7 depicts the command relationships for targeting as they are portrayed in the 1998 USCENTCOM Targeting Architecture. All components and coalition forces support the targeting process. The graphic denotes those organizations with a primary role and those with a supporting role.

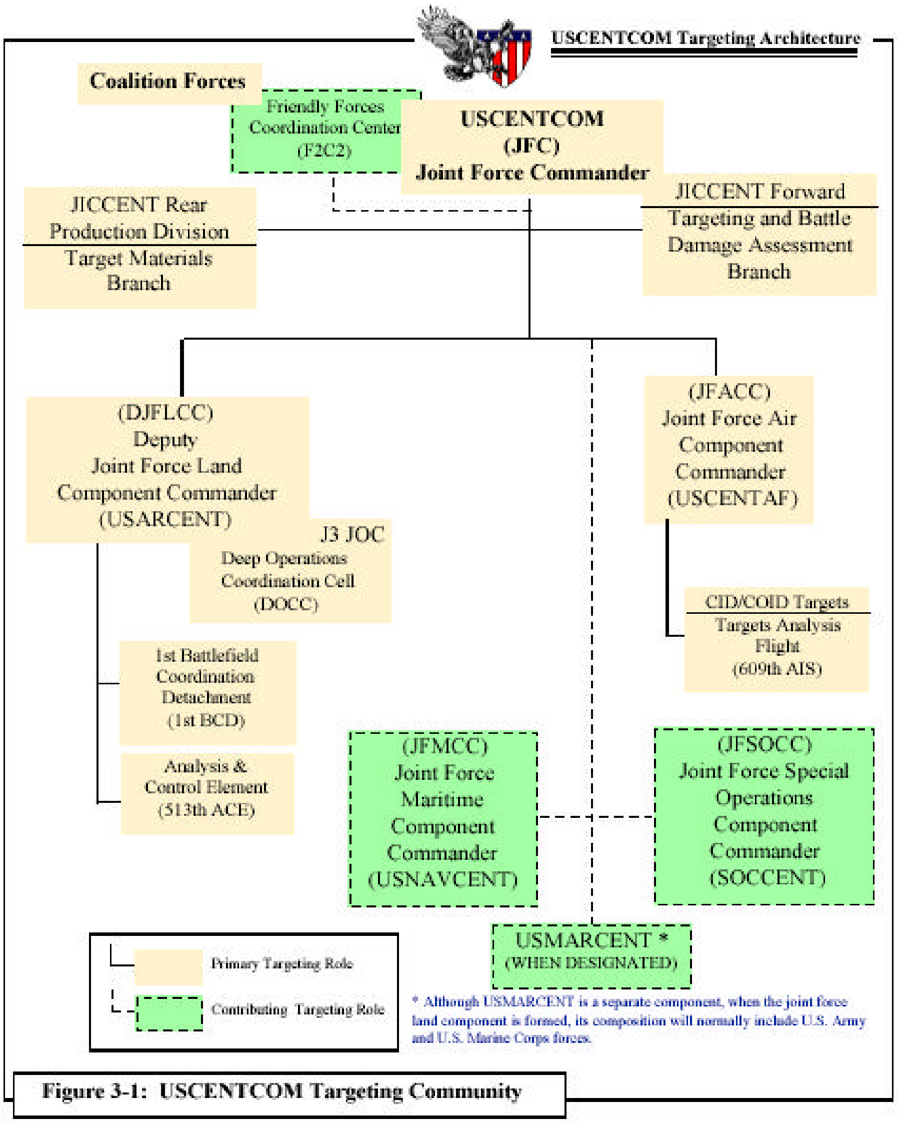

USCENTCOM Targeting Architecture
Coalition Forces
Friendly Forces Coordination Center (F2C2)
USCENTCOM (JFC) Joint Force Commander
JICCENT Rear Production Division
Target Materials Branch
JICCENT Forward
Targeting and Battle Damage Assessment Branch
(DJFLCC) Deputy Joint Force Land Component Commander (USARCENT)
J3 JOC Deep Operations Coordination Cell (DOCC)
1st Battlefield Coordination Detachment (1st BCD)
Analysis & Control Element (513th ACE)
(JFACC) Joint Force Air Component Commander (USCENTAF)
CID/COID Targets
Targets Analysis Flight (609th AIS)
(JFMCC) Joint Force Maritime Component Commander (USNAVCENT)
(JFSOCC) Joint Force Special Operations Component Commander (SOCCENT)
USMARCENT * (WHEN DESIGNATED)
Primary Targeting Role
Contributing Targeting Role
* Although USMARCENT is a separate component, when the joint force land component is formed, its composition will normally include U.S. Army and U.S. Marine Corps forces.
Figure 3-1: USCENTCOM Targeting Community

Figure 2.4-7. Example Organizational Relationships Chart (OV-4)

2.4.3.7 Operational Activity Model (OV-5)

An Operational Activity Model (OV-5) may include both Operational Activity Hierarchy Chart(s) and Operational Activity Diagram(s). Figure 2.4-1 and Figure 2.4-2 discussed in section 2.4.1 are extracts of the Operational Activity Hierarchy Chart of this architecture. The OV-5, shown in **Figure 2.4-8,** depicts the three sub-activities of *Conduct Combat Assessment*. The sub-activities are:

- Conduct Battle Damage Assessment (BDA) – To conduct timely and accurate estimate of damage resulting from the application of military force, either lethal or non-lethal, against predetermined operational objectives. BDA can be applied to all types of systems throughout the range of military operations.[1]
- Conduct Munitions Effects Assessment (MEA) – To evaluate damage from munitions employed to determine more effective munitions for continuing attack of those targets in subordinate campaigns and major operations.[2]
- Recommend Re-strike – To evaluate the overall impact and effectiveness of operations against the enemy and what, if any, changes or additional efforts need to take place to meet the operational commander's objectives in the current major operation or phase of the subordinate campaign.[3]

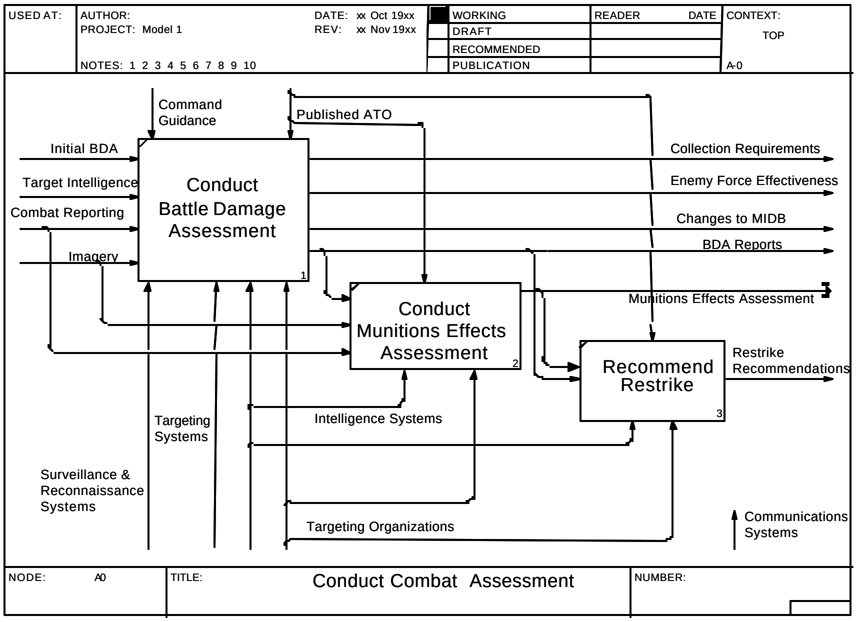


Figure 2.4-8. Example *Conduct Combat Assessment* Operational Activity Model (OV-5)

[1] CJCS Manual 3500.04B, *Universal Joint Task List*, July 1, 2002, p. B-C-C-55.

[2] Ibid, p. B-C-C-56.

[3] Ibid.

2.4.3.8 Operational Rules Model (OV-6a)

Operational rules can be expressed in a number of different formats. Notional example rules expressed in structured English and a decision tree follow. OV-6a is independent of the descriptive technique.

Structured English – Many business rules are simply statements that can be considered as basic requirements or design constraints. Some examples for the current sample architecture might be:

- ATOs are developed on a 24- to 96-hour planning cycle.
- BDAs are developed for every target that has had an air sortie directed against it.
- MEAs are developed to identify deficiencies in weapon system/munitions performance, tactics, or aim point selection.
- Targets for which the BDA/MEA falls below a specified threshold (as developed by the Joint Force Commander [JFC]/Joint Force Air Component Commander [JFACC] in accordance with the Commander's intent) are recommended for re-strike.

Decision Tree – Many business rules can be expressed with decision trees. A decision tree describes several possible courses of decision paths and outcomes or activities to be conducted at the end of decision paths.

Figure 2.4-9 is presented as an example only. The context for the rule is the damage assessment operational activities. The basic concept behind this rule set is to determine whether a target should be scheduled for a re-strike, and if so, whether to schedule an immediate re-strike or to add the target back to the target list for the succeeding ATO cycle. This represents essentially four sequential decisions. The first decision is to determine if a target has been struck (Decision 1 in Figure 2.4-9). The second decision is based on the BDA-MEA results (Decision 2 in Figure 2.4-9). The third and fourth decisions are based on the time sensitivity of the target and the availability of resources (Decisions 3 and 4 in Figure 2.4-9, respectively).

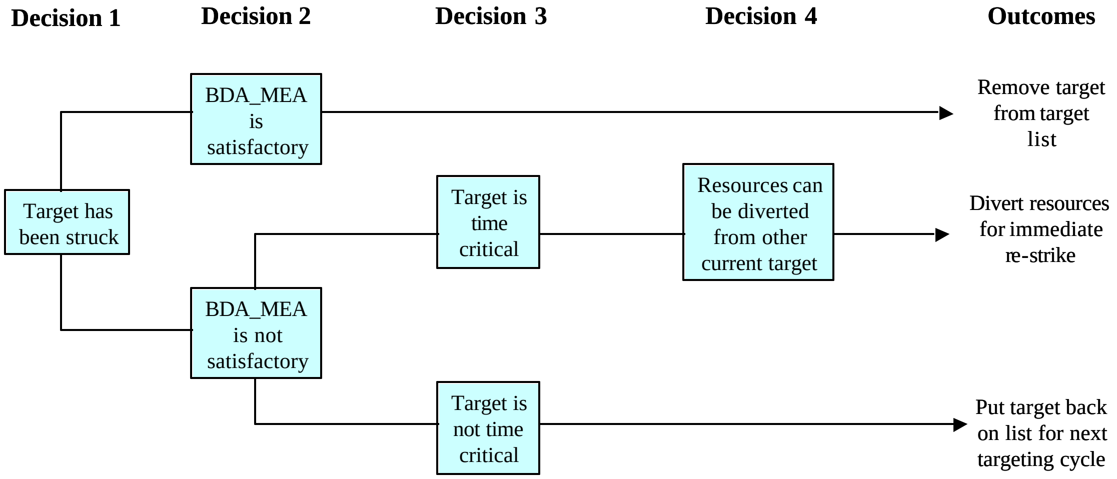

Decision 1
Decision 2
Decision 3
Decision 4
Outcomes
BDA_MEA is satisfactory
Remove target from target list
Target has been struck
Target is time critical
Resources can be diverted from other current target
Divert resources for immediate re-strike
BDA_MEA is not satisfactory
Target is not time critical
Put target back on list for next targeting cycle

Figure 2.4-9. Notional Decision Tree for *Conduct Combat Assessment*

2.4.3.9 Operational State Transition Description (OV-6b)

Figure 2.4-10 is a notional Operational State Transition Description (OV-6b) depicting high-level activities within *Conduct Joint Force Targeting*. This notional product was developed for the Deskbook based on information in the USCENTCOM Targeting Architecture. An architecture description would likely have a number of such diagrams. Each diagram should describe independent behavior of the operational thread or sequence of operational activities performed by the operational nodes. OV-6b is independent of the descriptive technique.

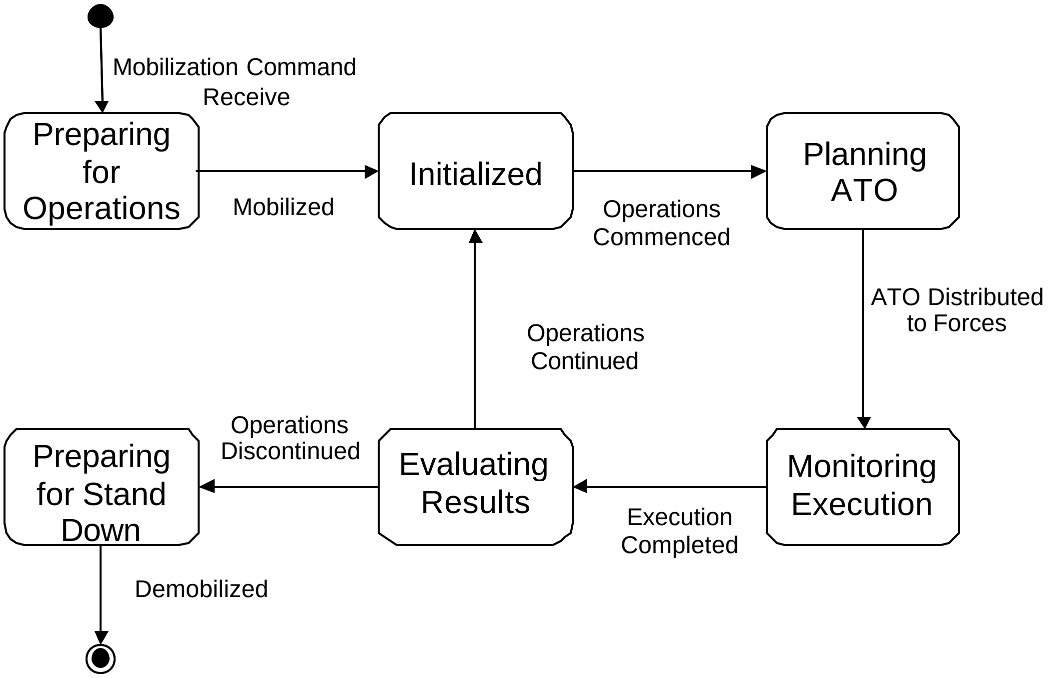


Figure 2.4-10. Notional Operational State Transition Description (OV-6b)

2.4.3.10 Operational Event-Trace Description (OV-6c)

Figure 2.4-11 is a notional Operational Event-Trace Description (OV-6c) depicting a sequence of events within *Conduct Combat Assessment*. This notional product was developed for the Deskbook based on information in the USCENTCOM Targeting Architecture. Each diagram corresponds to a particular sequence of events within a specified scenario or due to a specific set of pre-conditions. An architecture would likely include a large number of these diagrams. OV-6c is independent of the descriptive technique.

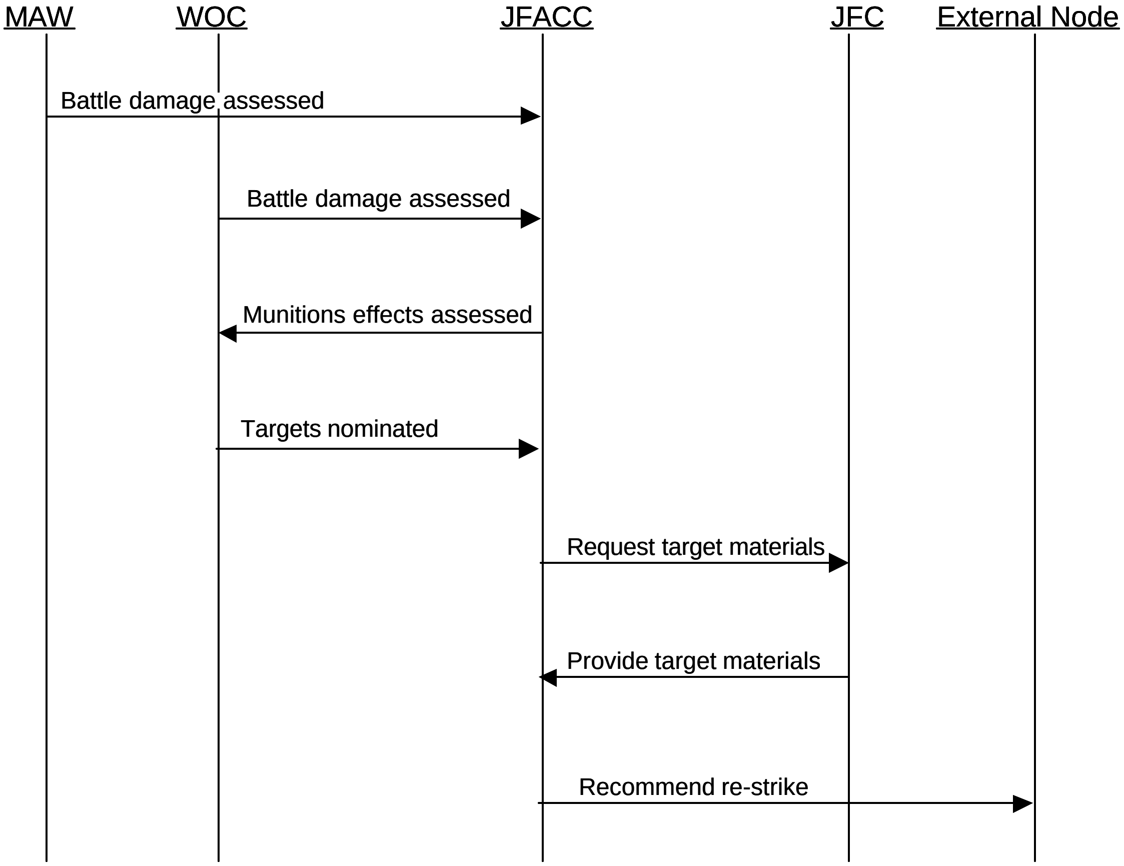


Figure 2.4-11. Notional Operational Event-Trace Description (OV-6c)

2.4.3.11 Logical Data Model (OV-7)

Figure 2.4-12 is a notional Logical Data Model (OV-7) depicting data associated with *Conduct Joint Force Targeting*. This notional product was developed for the Deskbook based on information in the USCENTCOM Targeting Architecture. This figure represents only a small portion of what a complete data model would look like. Data models usually extend over several pages, each page showing the data entities that are involved in a particular operational activity or mission. Depending on the architecture purpose, a finished OV-7 may or may not have attributes defined for entity types.

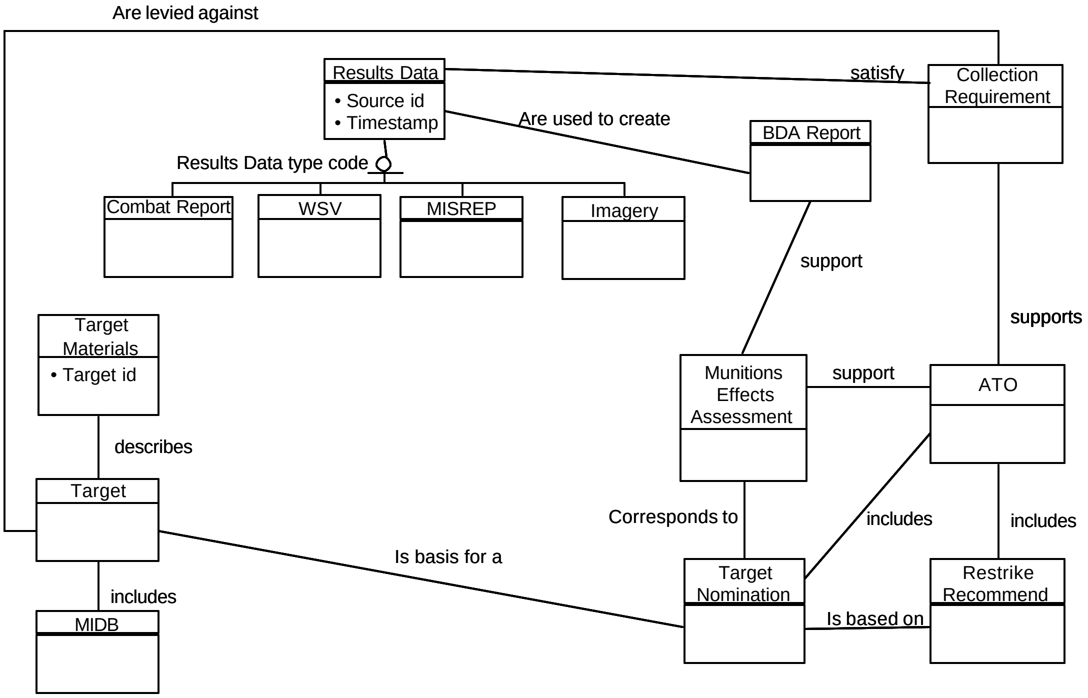
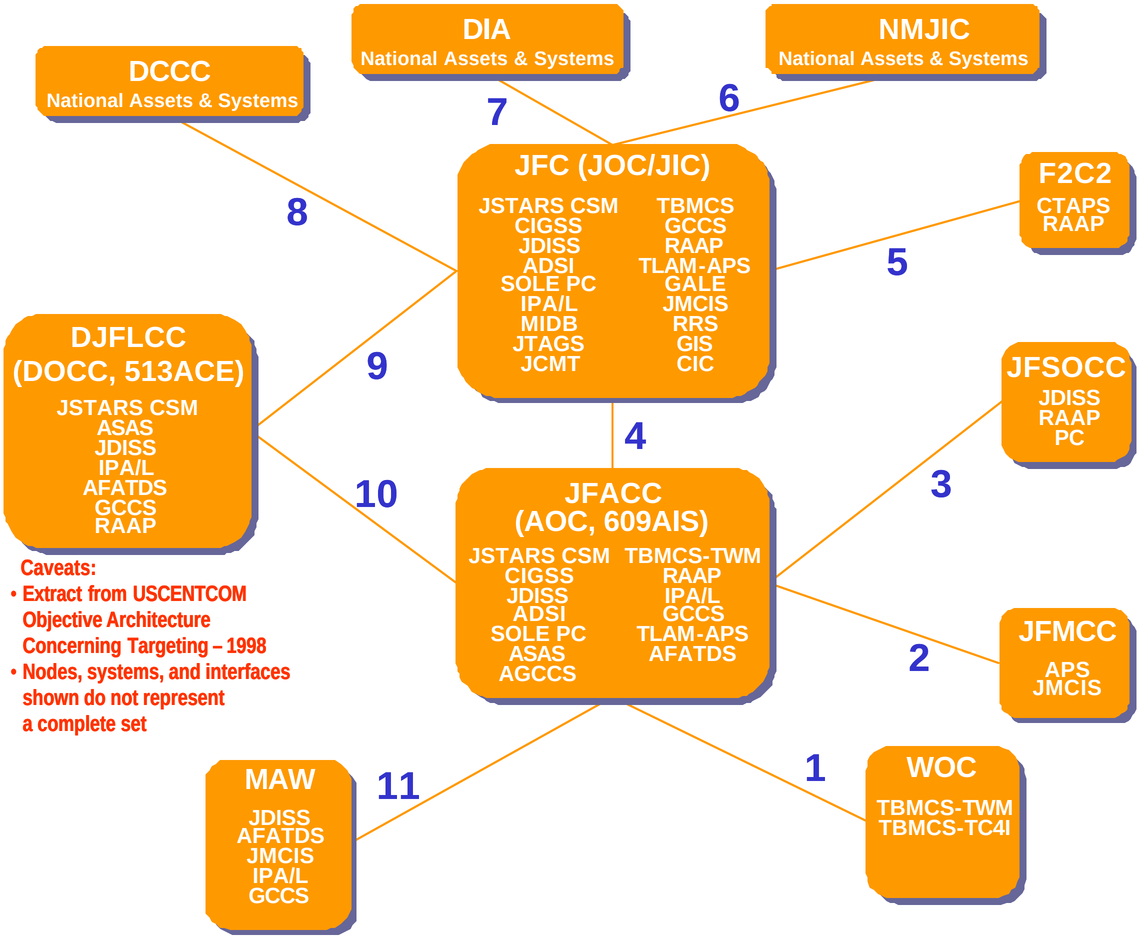


Figure 2.4-12. Notional Logical Data Model (OV-7)

2.4.3.12 Systems Interface Description (SV-1)

The Systems Interface Description (SV-1) presented in **Figure 2.4-13** depicts systems at systems nodes and system interconnections involved in *Conduct Joint Force Targeting*. This figure contains an illustrative set of systems and systems nodes from the 1998 USCENTCOM Targeting Architecture.



Caveats:

- Extract from USCENTCOM Objective Architecture Concerning Targeting – 1998
- Nodes, systems, and interfaces shown do not represent a complete set

Figure 2.4-13. Example Systems Interface Description (SV-1)

2.4.3.13 Systems Communications Description (SV-2)

A notional Systems Communications Description (SV-2) is shown in **Figure 2.4-14**. In a full architecture description, additional details about each of the communications links would be provided, as per the descriptions in Volume II. SV-2 is independent of the descriptive technique.

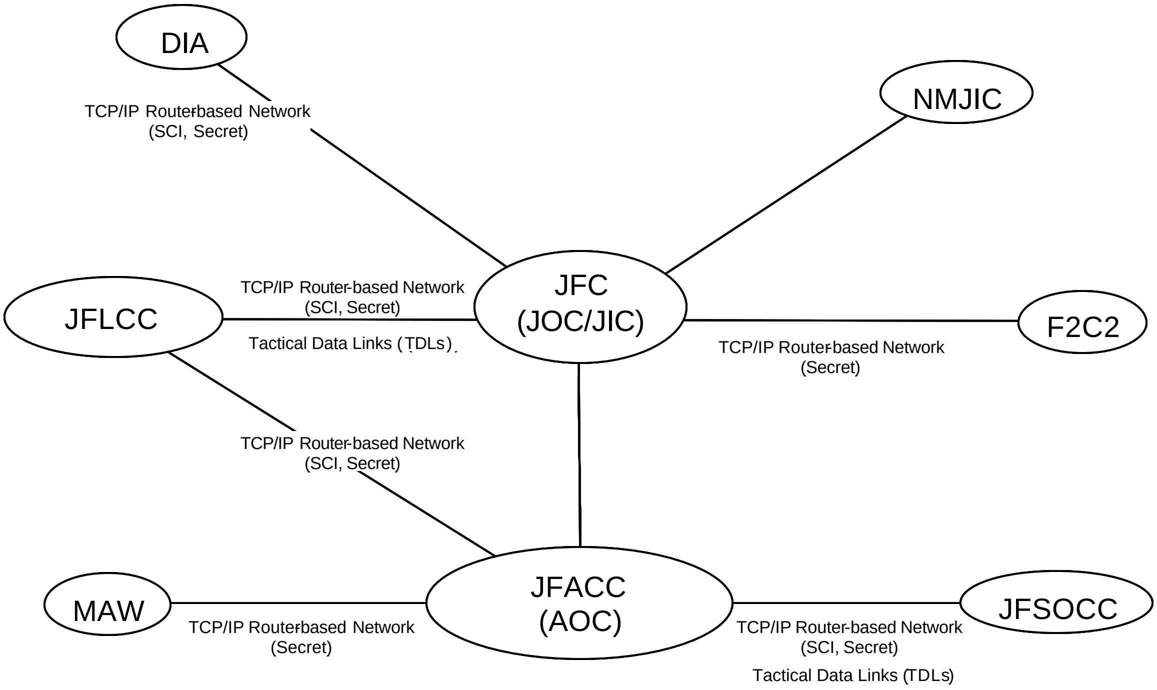

DIA
TCP/IP Routerbased Network
(SCI, Secret)
NMJIC
JFLCC
TCP/IP Router-based Network
(SCI, Secret)
Tactical Data Links (TDLs)
JFC
(JOC/JIC)
F2C2
TCP/IP Routerbased Network
(Secret)
TCP/IP Router-based Network
(SCI, Secret)
MAW
TCP/IP Routerbased Network
(Secret)
JFACC
(AOC)
JFSOCC
TCP/IP Router-based Network
(SCI, Secret)
Tactical Data Links (TDLs)

Figure 2.4-14. Notional Systems Communications Description (SV-2)

2.4.3.14 Systems-Systems Matrix (SV-3)

No example product is provided. SV-3 is independent of the descriptive technique. The format presented in Volume II should be used for both structured analysis and UML.

2.4.3.15 Systems Functionality Description (SV-4)

Figure 2.4-15 is a notional Systems Functionality Description (SV-4) depicting system functions, data flows, and an external sink within *Conduct Combat Assessment*. This notional product was developed for the Deskbook based on information in the USCENTCOM Targeting Architecture.

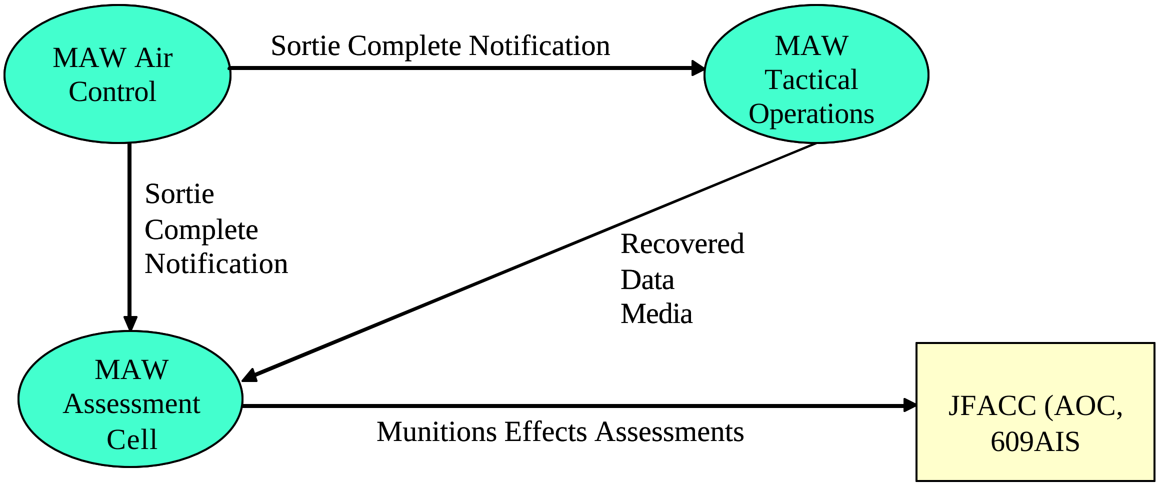


Figure 2.4-15. Notional Systems Functionality Description (SV-4)

2.4.3.16 Operational Activity to Systems Function Traceability Matrix (SV-5)

No example product is provided. SV-5 is independent of the descriptive technique. The format presented in Volume II should be used for both structured analysis and UML.

2.4.3.17 Systems Data Exchange Matrix (SV-6)

No example product is provided. SV-6 is independent of the descriptive technique. The format presented in Volume II should be used for both structured analysis and UML.

2.4.3.18 Systems Performance Parameters Matrix (SV-7)

No example product is provided. SV-7 is independent of the descriptive technique. The format presented in Volume II should be used for both structured analysis and UML.

2.4.3.19 Systems Evolution Description (SV-8)

The System Evolution Description (SV-8) defines the expected evolution of the systems within the architecture as a function of time. SV-8 is independent of the descriptive technique. The USCENTCOM architecture did not include this product. **Figure 2.4-16** is a template of such a product, provided here as a notional example.

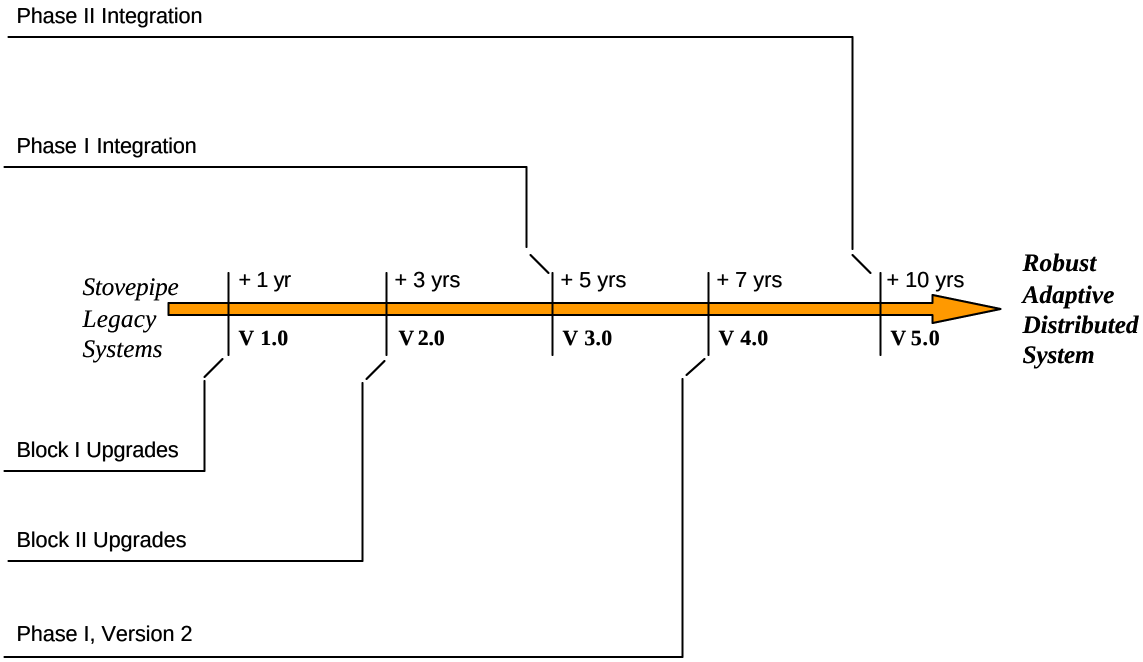


Figure 2.4-16. Notional Systems Evolution Description (SV-8) Template

2.4.3.20 Systems Technology Forecast (SV-9)

No example product is provided. SV-9 is independent of the descriptive technique. The format presented in Volume II should be used for both structured analysis and UML.

2.4.3.21 Systems Rules Model (SV-10a)

The Systems Rules Model (SV-10a) has the same format as the Operational Rules Model (OV-6a); however, the scope and applicability of the rules here are for individual systems, where OV-6a applies the architecture as a whole. SV-10a is independent of the descriptive technique. An example of a system's rule is presented in **Figure 2.4-17**.

All systems using the Link-33 communications terminals that receive Message A4, Request for Active Missile Tracks, must respond within 1second with a Message A6, Active Missile Tracks Update.

Figure 2.4-17. Structured English

2.4.3.22 Systems State Transition Description (SV-10b)

The Systems State Transition Description (SV-10b) has the same format as the Operational State Transition Description (OV-6b), except the scope of the state transition description is limited to individual systems rather than the architecture as a whole. No example product is provided. SV-10b is independent of the descriptive technique.

2.4.3.23 Systems Event-Trace Description (SV-10c)

The Systems Event-Trace Description (SV-10c) has the same format as the Operational Event-Trace Description (OV-6c), except the scope of the event-trace description is limited to individual systems rather than the architecture as a whole. No example product is provided. SV-10c is independent of the descriptive technique.

2.4.3.24 Physical Schema (SV-11)

The Physical Schema (SV-11) may be in the form of an entity relationship (ER) diagram, a data definition language (DDL) model, message formats, or a file structure. No example product is provided.

2.4.3.25 Technical Standards Profile (TV-1)

No example product is provided. TV-1 is independent of the descriptive technique. The format presented in Volume II should be used for both structured analysis and UML.

2.4.3.26 Technical Standards Forecast (TV-2)

No example product is provided. TV-2 is independent of the descriptive technique. The format presented in Volume II should be used for both structured analysis and UML.

2.4.4 Example Products Based on Object-Oriented Techniques and the UML

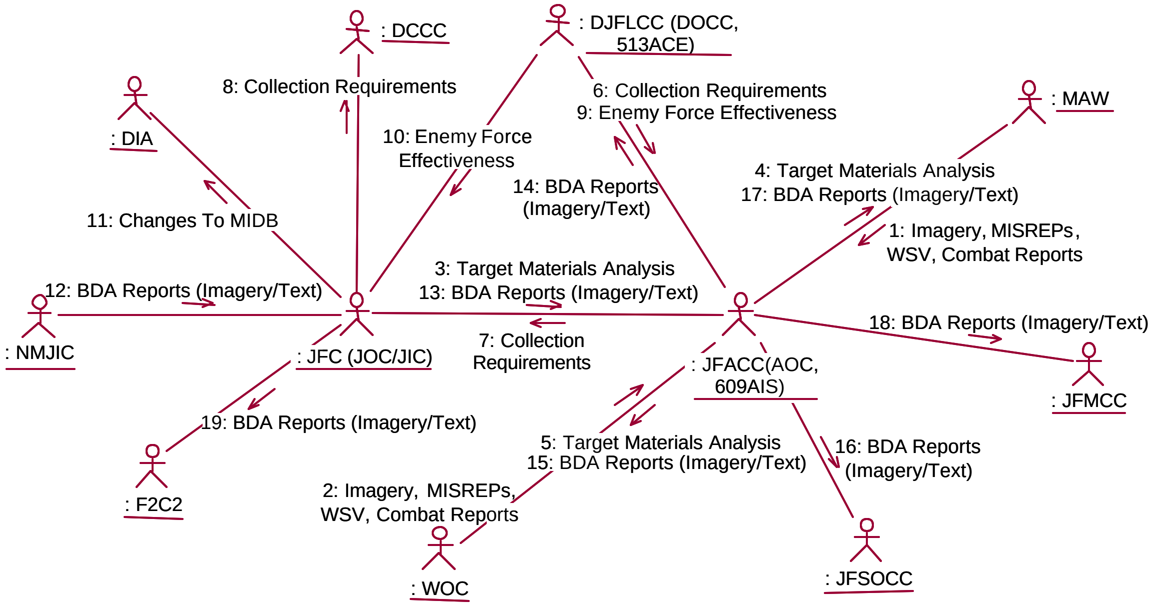
This section presents the applicable products with UML representation for the illustrative architecture. This comparison is done in accordance with the Framework template/UML representation presented in Volume II. It is provided as initial guidance to architects who choose UML for describing architectures in accordance with the DoDAF.

In this section, UML diagrams consisting of use case, collaboration, class, deployment, and component diagrams have been developed to represent the Framework products for which a UML representation has been defined (sections 3, 4, and 5 in Volume II). The representation of OV-6b (state diagram) and OV-6c (sequence diagram) is already in UML; therefore, they are considered common products and are not repeated in this section. The information content of the other common products not included in this example can be extracted from the UML diagrams, but they are not represented using UML notation.

2.4.4.1 Operational Node Connectivity Description (OV-2)

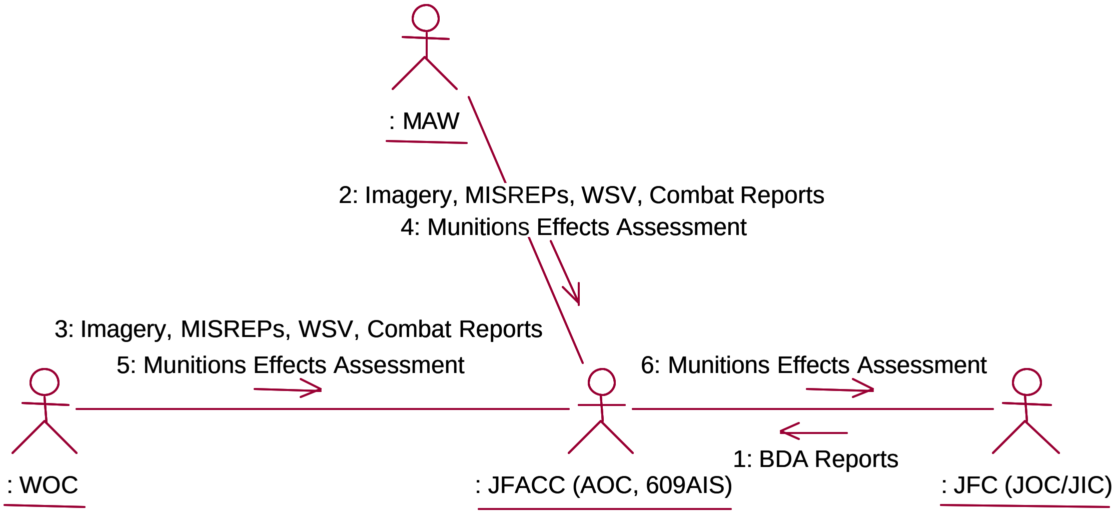
The example Operational Node Connectivity Description (OV-2) is shown in **Figure 2.4-18**. The UML collaboration diagram format has been used for this figure.

UML collaboration diagrams, like UML sequence diagrams, are dependent upon the specific scenario being executed. Different scenarios or sets of pre-conditions may lead to different sequences of information exchanges, and thus a different diagram. Several collaboration diagrams may be needed to describe all of the actual node connectivities (see **Figure 2.4-19** and **Figure 2.4-20**). Some specific node-to-node connections may not be involved in all scenarios that the overall architecture may execute.



: DCCC
: DJFLCC (DOCC, 513ACE)
: DIA
8: Collection Requirements
6: Collection Requirements
9: Enemy Force Effectiveness
: MAW
10: Enemy Force Effectiveness
11: Changes To MIDB
14: BDA Reports (Imagery/Text)
4: Target Materials Analysis
17: BDA Reports (Imagery/Text)
1: Imagery, MISREPs, WSV, Combat Reports
3: Target Materials Analysis
13: BDA Reports (Imagery/Text)
12: BDA Reports (Imagery/Text)
: NMJIC
: JFC (JOC/JIC)
7: Collection Requirements
18: BDA Reports (Imagery/Text)
: JFACC(AOC, 609AIS)
: JFMCC
19: BDA Reports (Imagery/Text)
5: Target Materials Analysis
15: BDA Reports (Imagery/Text)
16: BDA Reports (Imagery/Text)
: F2C2
2: Imagery, MISREPs, WSV, Combat Reports
: WOC
: JFSOCC

Figure 2.4-18. Notional Top-Level UML Collaboration Diagram Used for the Operational Node Connectivity Description (OV-2)



: MAW
2: Imagery, MISREPs, WSV, Combat Reports
4: Munitions Effects Assessment
3: Imagery, MISREPs, WSV, Combat Reports
5: Munitions Effects Assessment
6: Munitions Effects Assessment
1: BDA Reports
: WOC
: JFACC (AOC, 609AIS)
: JFC (JOC/JIC)

Figure 2.4-19. Second UML Collaboration Diagram Used for the Operational Node Connectivity Description (OV-2)

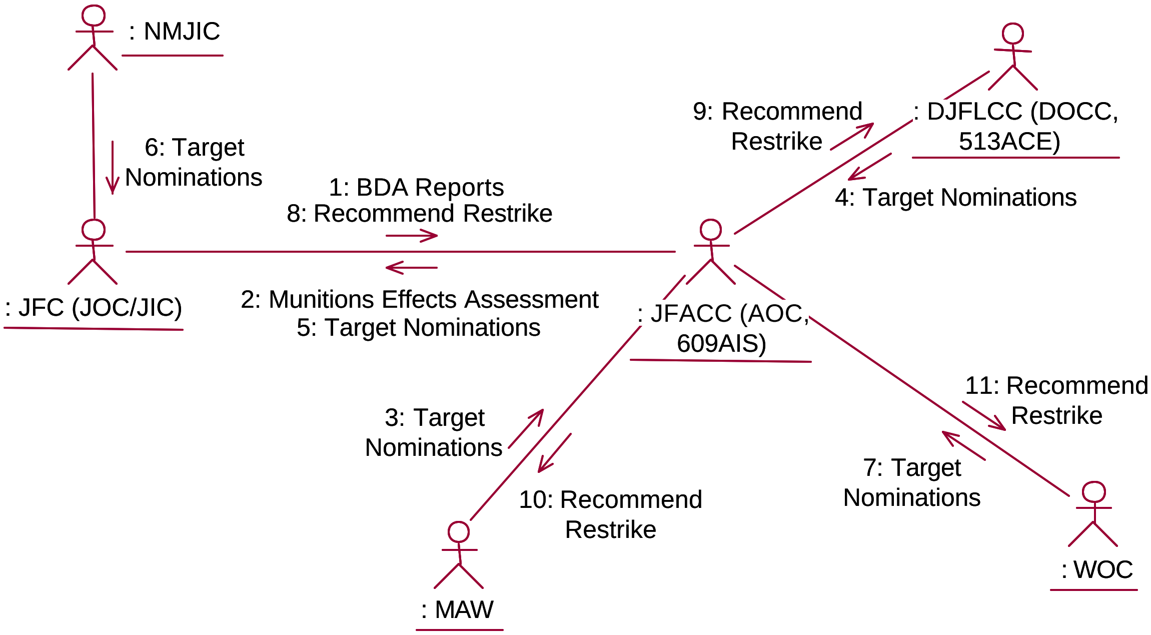


Figure 2.4-20. Third UML Collaboration Diagram Used for the Operational Node Connectivity Description (OV-2)

In addition, a UML class diagram can be used to document the classes that represent the operational nodes, their methods and the information provided or required by these classes (operational nodes), and their static structure of relationships representing the needlines between them. **Figure 2.4-21** shows the class diagram that details the operational nodes (classes), the activities performed by each operational node (methods), and needlines (relationships) between the operational nodes. The information exchanges are not shown in the example but may be noted as the names of the relationships, if desired.

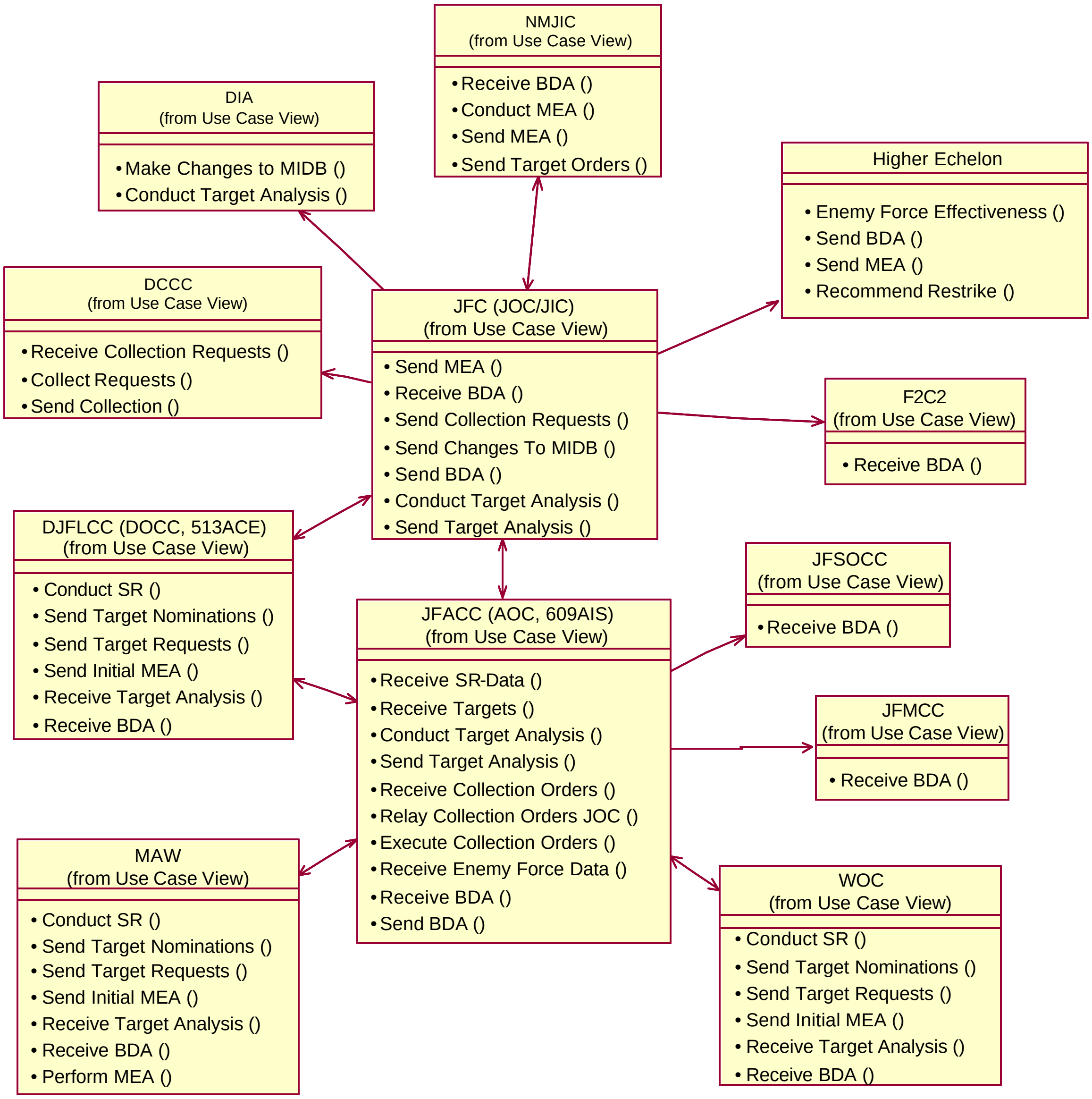


Figure 2.4-21. Notional UML Class Diagram Used for the Operational Node Connectivity Diagram (OV-2)

2.4.4.2 Organizational Relationships Chart (OV-4)

An example Organizational Relationship Chart (OV-4) in UML representation is shown in **Figure 2.4-22**. In a full architecture description there may be many such diagrams, or the total organizational structure may be divided over multiple pages.

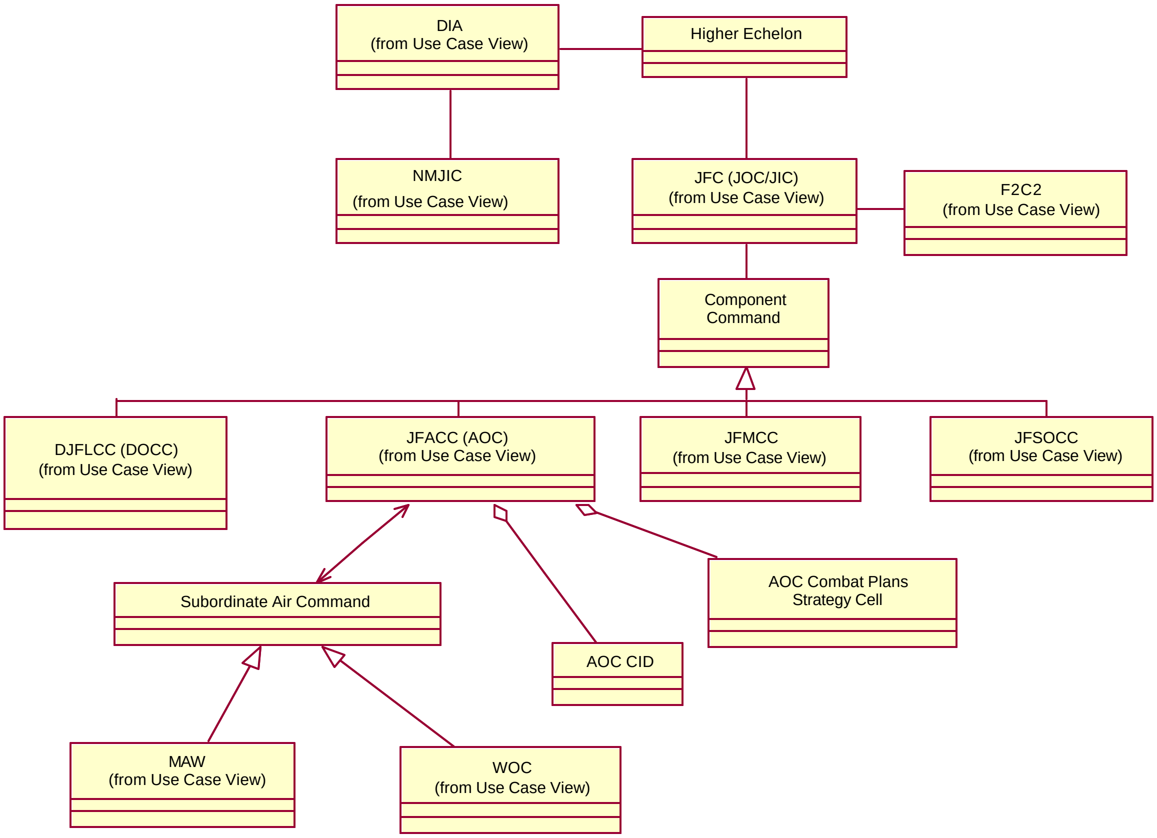


Figure 2.4-22. Notional UML Class Diagram Used for the Organizational Relationships Chart (OV-4)

Note that operations have been suppressed from the class icons for clarity in this figure. Also note that the class icons in the class diagram can be replaced with actor icons to show these classes represent human (organizations).

2.4.4.3 Operational Activity Model (OV-5)

The Operational Activity Model (OV-5) can be represented with UML use case diagrams as shown in **Figure 2.4-23** and **Figure 2.4-24**. The first use case diagram is intended to parallel the Operational Activity Hierarchy Chart extracts shown in Figure 2.4-1 and Figure 2.4-2. The second use case diagram is intended to parallel the Operational Activity Diagram shown in Figure 2.4-8.

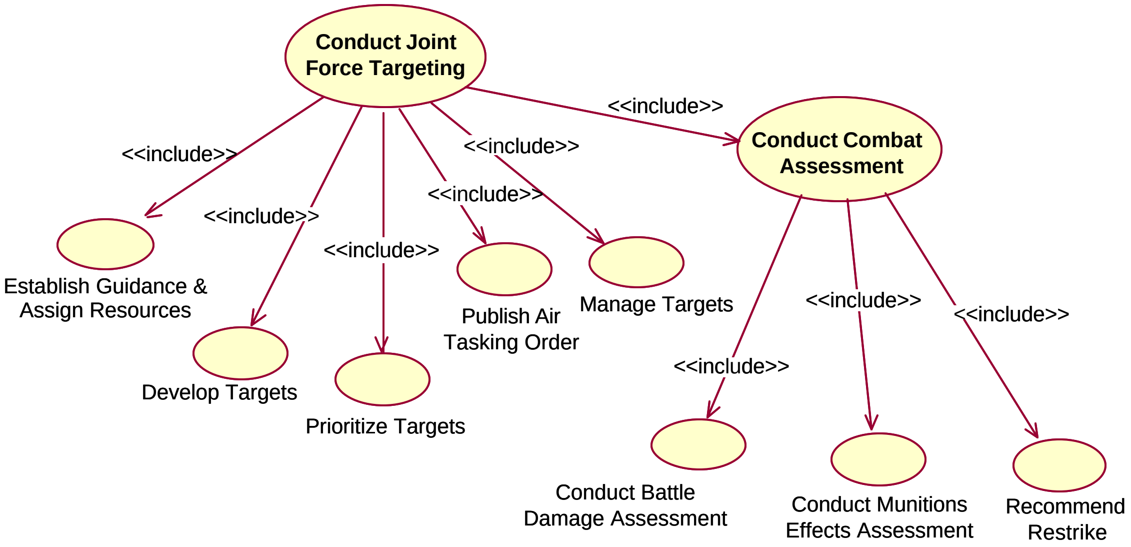

Conduct Joint Force Targeting
<<include>>
Conduct Combat Assessment
<<include>>
<<include>>
<<include>>
<<include>>
<<include>>
<<include>>
Establish Guidance & Assign Resources
Develop Targets
Prioritize Targets
Publish Air Tasking Order
Manage Targets
<<include>>
<<include>>
<<include>>
Conduct Battle Damage Assessment
Conduct Munitions Effects Assessment
Recommend Restrike

Figure 2.4-23. Notional UML Use Case Diagram Used for Operational Activity Hierarchy Chart (OV-5)

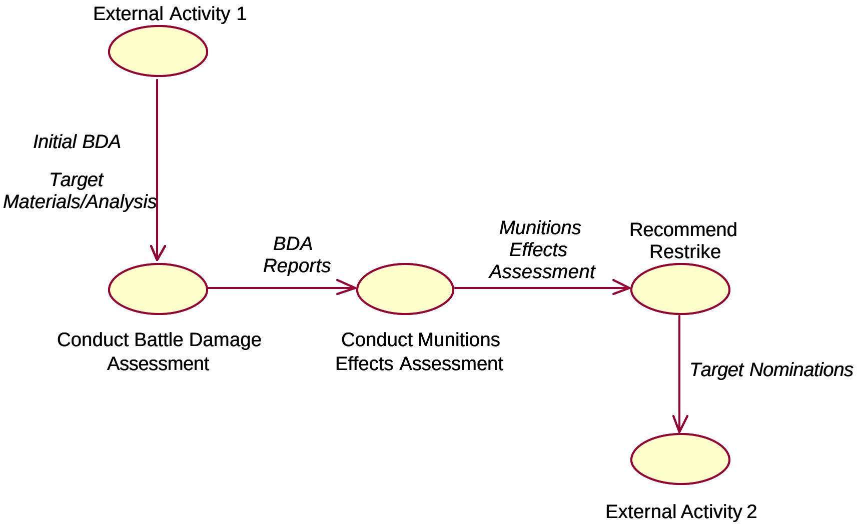

External Activity 1
Initial BDA
Target Materials/Analysis
BDA Reports
Munitions Effects Assessment
Recommend Restrike
Conduct Battle Damage Assessment
Conduct Munitions Effects Assessment
Target Nominations
External Activity 2

Figure 2.4-24. Notional UML Use Case Diagram Used for Operational Activity Model (OV-5)

To show a mission thread, a corresponding activity diagram may be developed to model the sequence of activities that support the use cases. **Figure 2.4-25** illustrates the complete mission thread for one cycle of combat assessment, starting with BDA, followed by MEA, and concluding with Recommend Re-strike.

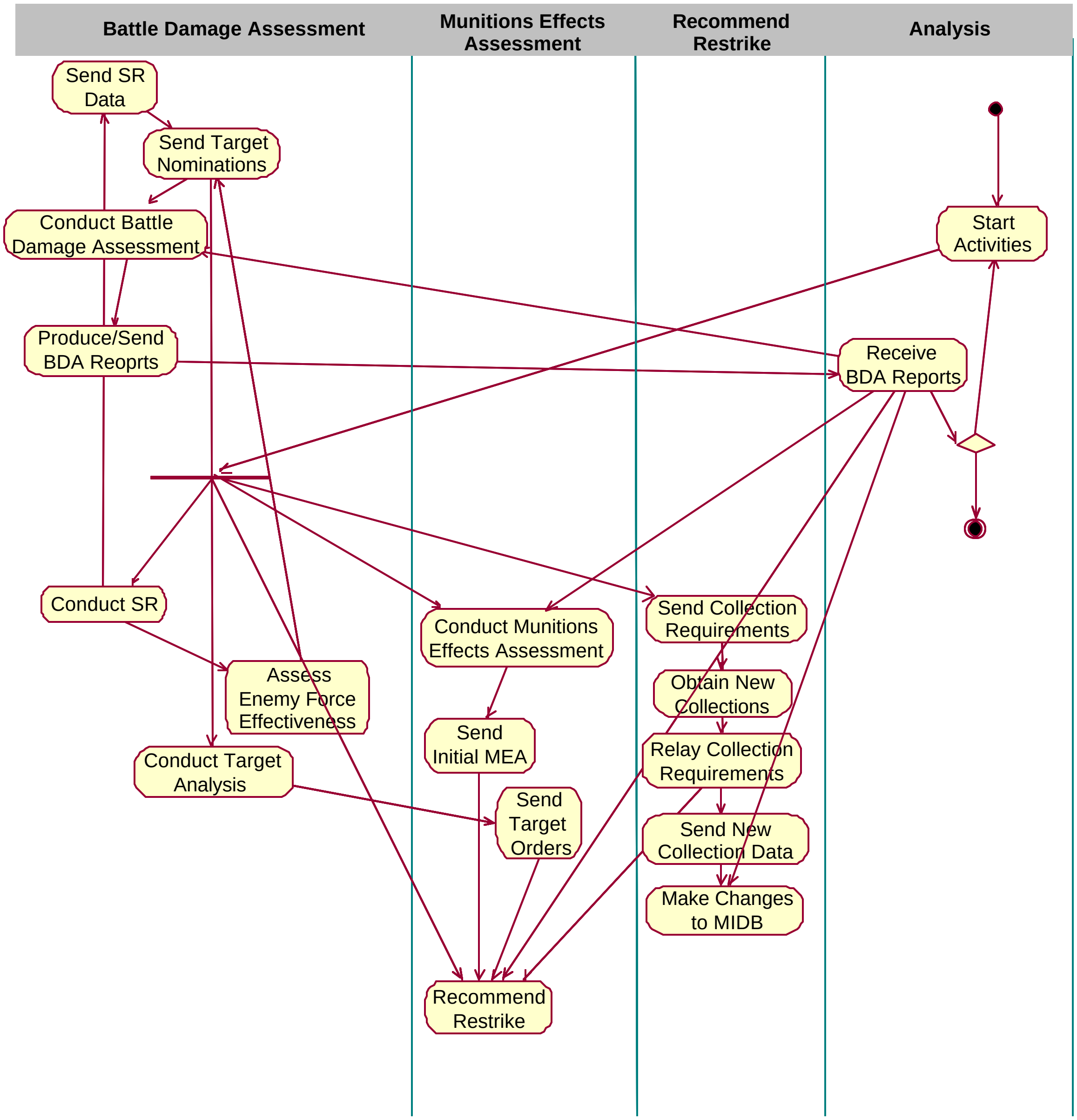


Figure 2.4-25. Notional UML Activity Diagram Used for the Operational Activity Model (OV-5)

2.4.4.4 Logical Data Model (OV-7)

The Logical Data Model (OV-7) can be represented with the UML class diagram, as shown in **Figure 2.4-26**. Only a portion of the overall class diagram is shown here for simplicity. The portion presented here is intended to parallel the portion shown in the structured analysis section of this document (see Figure 2.4-12).

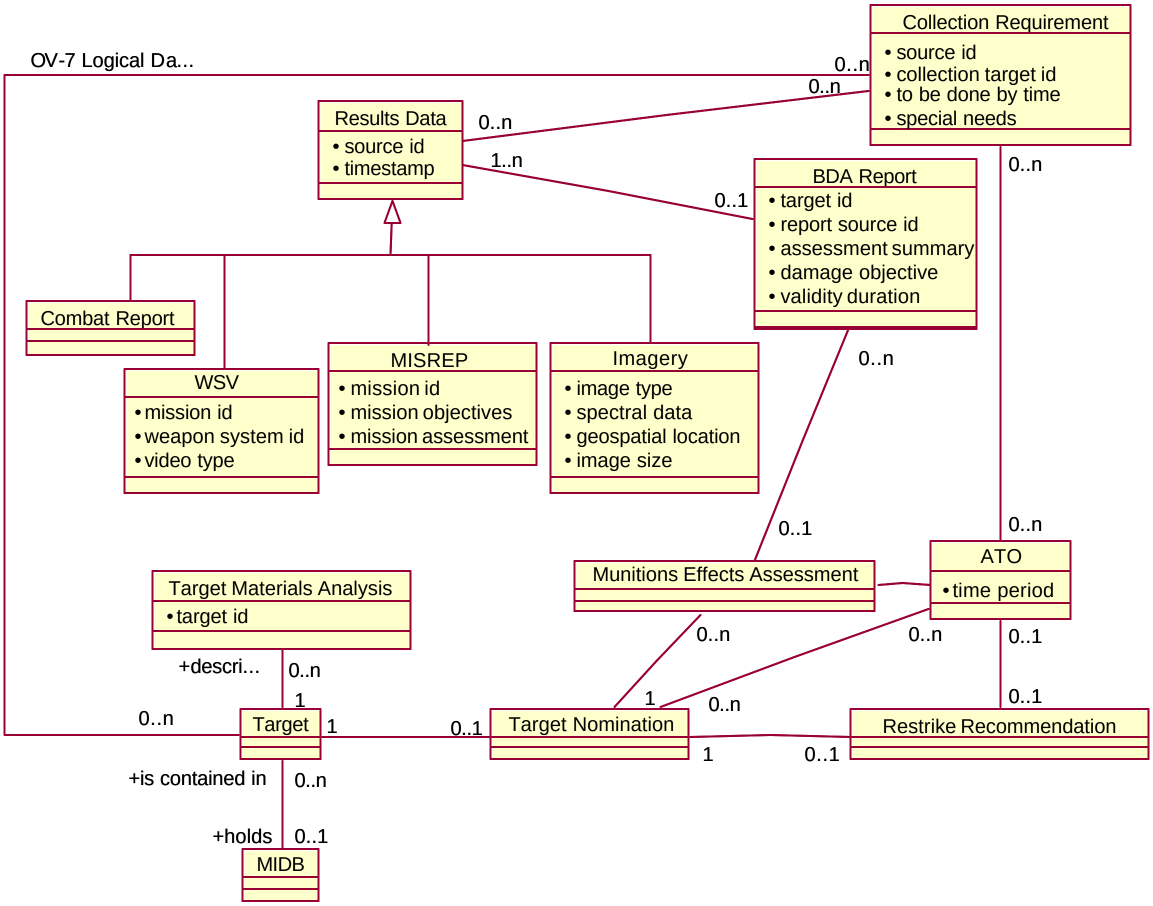


Figure 2.4-26. Notional UML Class Diagram Used for the Logical Data Model (OV-7)

2.4.4.5 Systems Interface Description (SV-1)

The Systems Interface Description (SV-1) (see **Figure 2.4-27**) depicts the systems nodes, and system interfaces needed to implement the automated portions of the operational activities (of OV-5) and the information exchanges (OV-2, OV-3, OV-5) involved in the *Conduct Joint Force Targeting* process. A Deployment Diagram (representing systems nodes) with the applicable components (representing systems) mapped to these nodes was used to develop this SV-1 product using UML notation. All elements are notional and do not represent actual information.

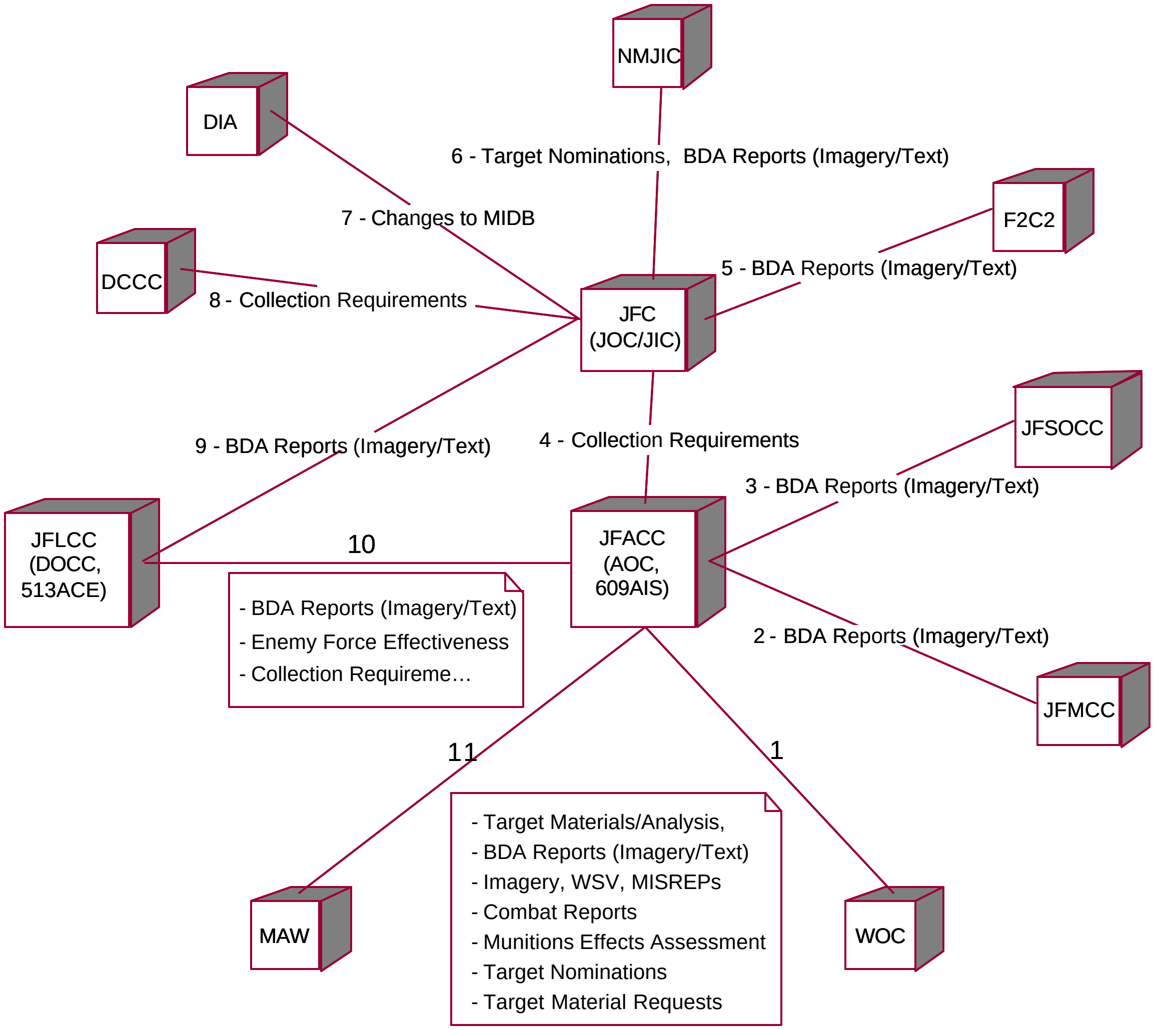

NMJIC
DIA
6 - Target Nominations, BDA Reports (Imagery/Text)
F2C2
7 - Changes to MIDB
DCCC
5 - BDA Reports (Imagery/Text)
8 - Collection Requirements
JFC (JOC/JIC)
JFSOCC
9 - BDA Reports (Imagery/Text)
4 - Collection Requirements
3 - BDA Reports (Imagery/Text)
JFLCC (DOCC, 513ACE)
10
JFACC (AOC, 609AIS)
- BDA Reports (Imagery/Text)
- Enemy Force Effectiveness
- Collection Requireme…
2 - BDA Reports (Imagery/Text)
JFMCC
11
1
- Target Materials/Analysis,
- BDA Reports (Imagery/Text)
- Imagery, WSV, MISREPs
- Combat Reports
- Munitions Effects Assessment
- Target Nominations
- Target Material Requests
MAW
WOC

Figure 2.4-27. Notional Systems Interface Description (SV-1)

2.4.4.6 Systems Functionality Description (SV-4)

The Systems Functionality Description (SV-4) can be represented with the UML use case diagram and class diagram, as shown in **Figure 2.4-28** and **Figure 2.4-29**. The class diagram presented here is intended to parallel the notional data flow diagram presented in subsection 2.4.3.15.

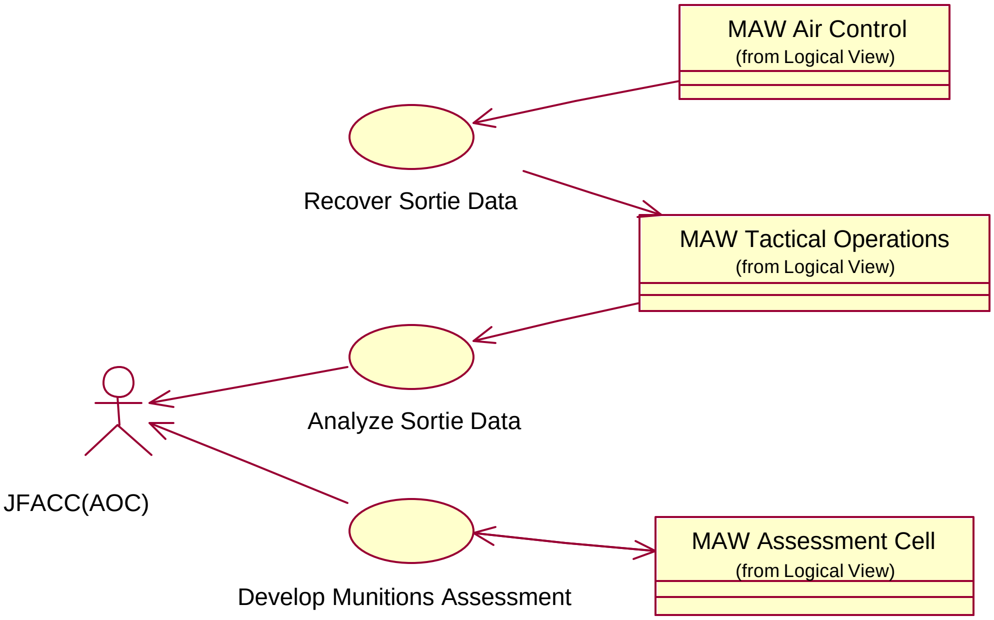


Figure 2.4-28. Notional Use Case Diagram Supporting Systems Functionality Description (SV-4)

Actors on use cases can also appear as classes on class diagrams. Whether an actor is "external" to the system depends completely on the scope of the current view. An entity that is included in one scope can be external in a more detailed view. For example, the entire aircraft could be viewed as a single system, with a number of components. However, the designers of a single subsystem for that aircraft, such as the navigation system, might well view the other neighboring subsystems on the aircraft as externals. Data can be exchanged between entities, whether they are described as "actors" or not. UML allows and supports the definition of class type characteristics for actor classes, such as class attributes and operations. The full spectrum of associations and relationships are also allowed for actors.

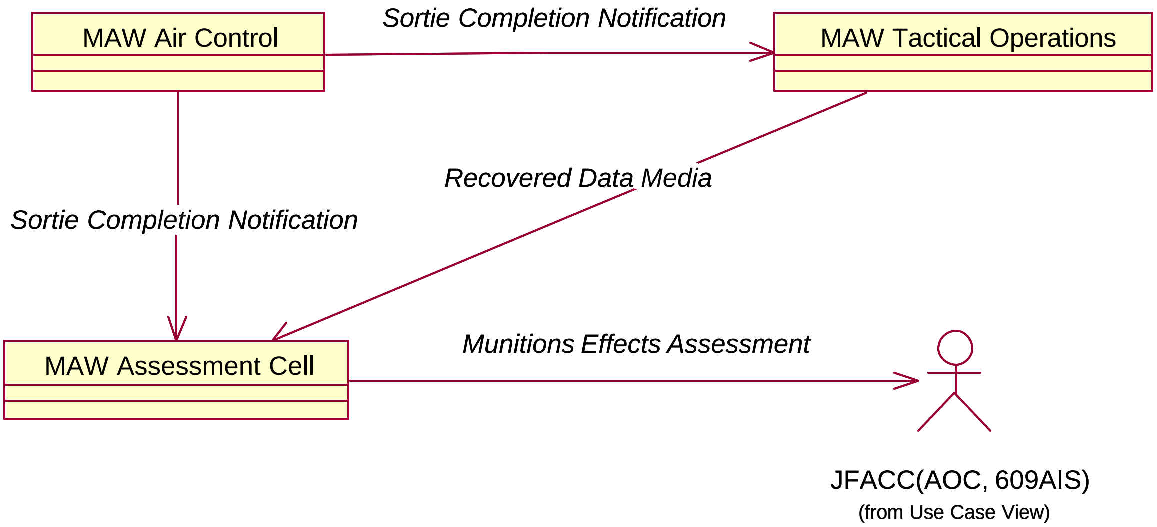


Figure 2.4-29. Notional UML Class Diagram Used for SV-4 Data Flow Diagram

Rarely will a single use case diagram or class diagram be sufficient for the architectures to which the DoDAF will be applied. Although the complexity of the architecture in question and the style of program management will affect the number of use cases and classes, tens of diagrams would not be unreasonable.

2.4.4.7 Physical Schema (SV-11)

The Physical Schema (SV-11) can be represented in the UML class diagram, as shown in **Figure 2.4-30**. A UML class diagram can capture all information an entity-relationship diagram captures. This example is related to the example shown for the Logical Data Model (OV-7) in Figure 2.4-12 and Figure 2.4-26. The data entities should be described in more detail here, providing sufficient information to support the subsequent phases of design and implementation.

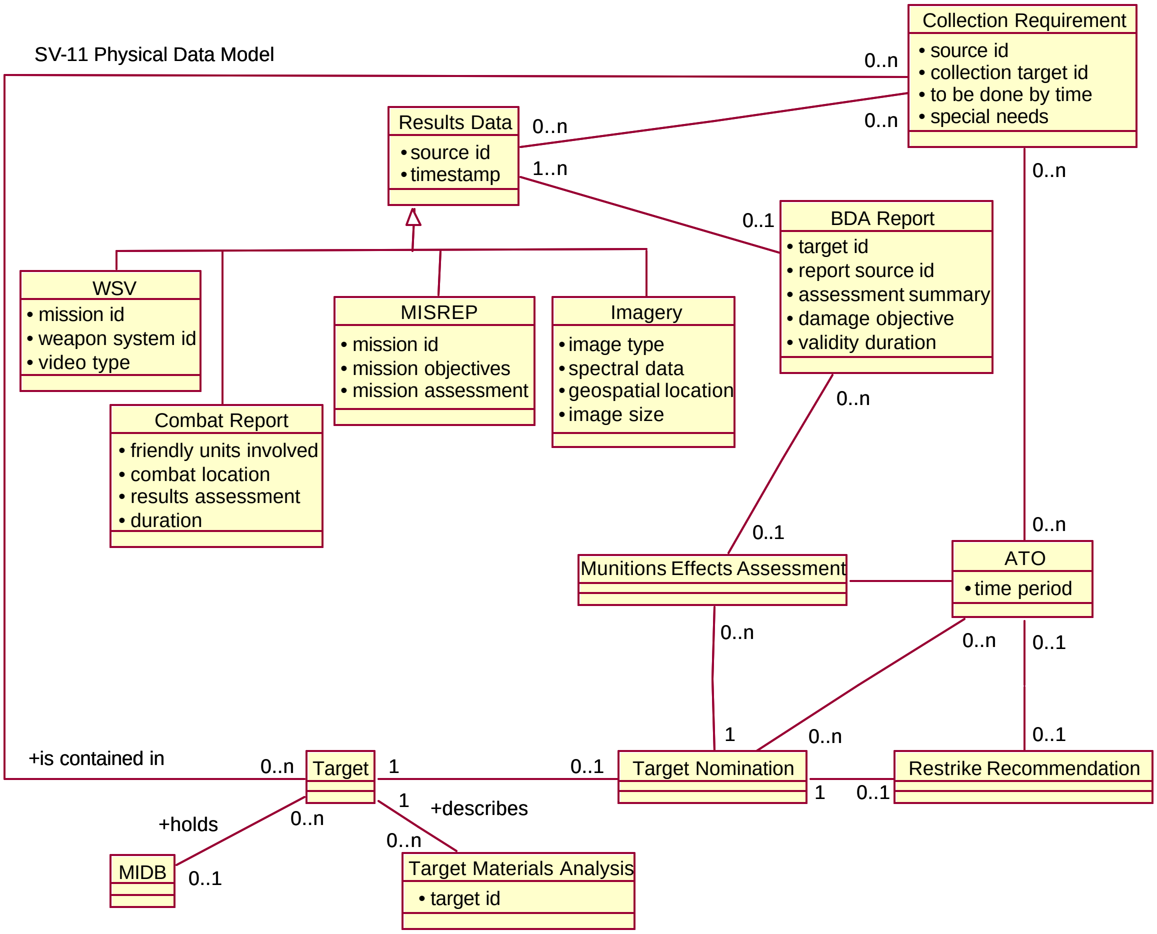


Figure 2.4-30. Notional UML Class Diagram Used for Physical Schema (SV-11)

2.4.5 References

CJCS Manual 3500.04B, *Universal Joint Task List*, July 1, 2002.

Joint Publication 1-02, *DoD Dictionary of Military Terms*, April 12, 2001.

OASD(C3I)/C4I Integration Support Activity (CISA), *US Central Command's Objective Architecture Concerning Targeting*, Volume I, March 1998.

OASD(C3I)/CISA, *C4I Mission Assessment*, Coordination Draft, March 1998.

2.5 USSPACECOM ARCHITECTURE DEVELOPED WITH OBJECT-ORIENTED METHODOLOGY

"Migrating Stovepipe Systems to Integrated/Interoperable Platforms Using the Technical Reference Model and Object-Oriented Operational Architectures"[1]

2.5.1 Objective of Case Study

This case study demonstrates how both the Technical Reference Model (TRM) and an object-oriented (OO) Operational View (OV) can be used to migrate disparate, stovepiped systems into an integrated and interoperable system using the ideas and concepts of spiral development, evolutionary acquisition, and the DoD Architecture Framework (DoDAF) (**Figure 2.5-1**). The traditional approach to system development, testing, and deployment is to build standalone systems capable of supporting specific functionality within a given mission (i.e., space, air, missile, etc.). This design strategy laid the foundation for our current redundant system architecture, which supports many operational systems and uses different programming languages. Maintaining hardware and software upgrades on standalone architecture designs of this nature imposes major limitations. These inflexible, stovepiped systems cannot meet the growing information exchange requirements of today's operational environment or provide a means to keep pace with evolving technology. The Combatant Commanders Integrated Command and Control System (CCIC2S) (previously known as the North American Aerospace Defense Command [NORAD]/United States Space Command [USSPACECOM] Warfighting Support System [N/UWSS]) project was initiated to resolve the problem of multiple, complex systems that retain an abundance of overlapping features and functions.

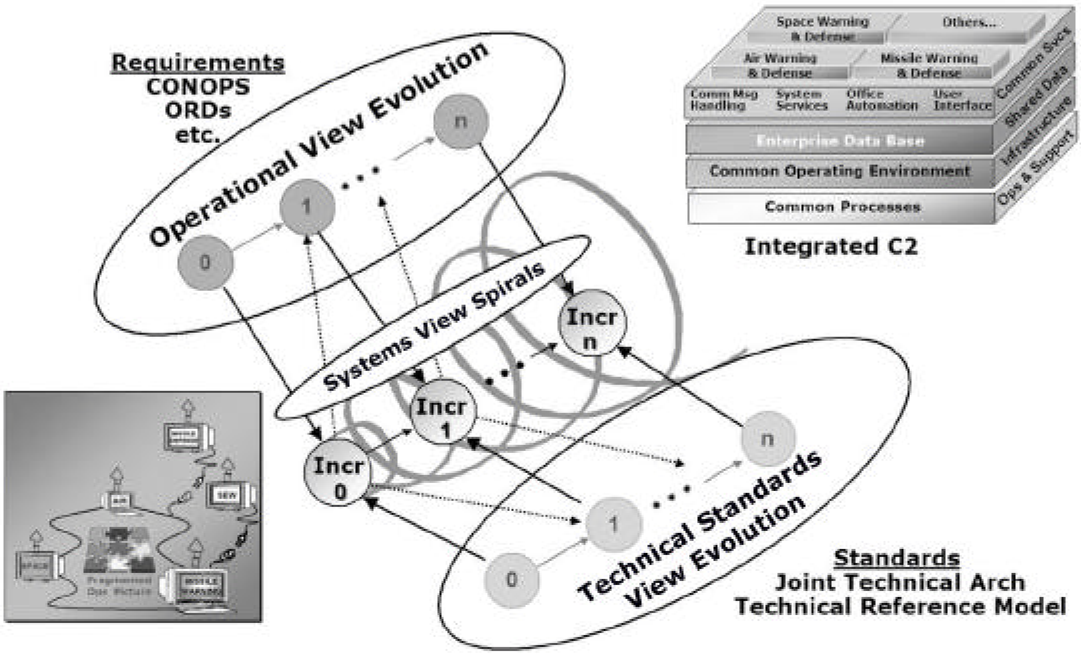


Figure 2.5-1. Spirally Evolving to Integrated/Interoperable Command and Control

[1] This section is a copy of a paper developed by NORAD/USSPACECOM.

The initial stages of the CCIC2S program were focused on identifying the baseline operational environment, determining redundancies, and describing a vision for migrating current systems. The dilemma was in determining a method to examine the operational activities across mission areas while realizing or determining functional redundancies within each stovepiped process or activity. Utilizing subject matter expertise from within the DoD community and an OO methodology, the CCIC2S Core Team identified and captured system redundancies and common functions within the existing Cheyenne Mountain Operations Center (CMOC) architecture, as well as the unique functions required to perform each mission.

The migration concept describes the vision and philosophy for migrating from the current complex of systems to a single multi-layered interoperable system that enables warfighters to accomplish their mission. The vision is a virtual environment that combines access to all air, space, missile, and intelligence mission information with automatic sharing of information with any authorized user who needs it worldwide.

Using the DoD TRM and Unified Modeling Language (UML) based Operational and Systems Views (SVs), the CCIC2S Team has created an overall DoDAF model that provides the migration of legacy systems to interoperability and provides the maximum level of reusability (**Figure 2.5-2**).

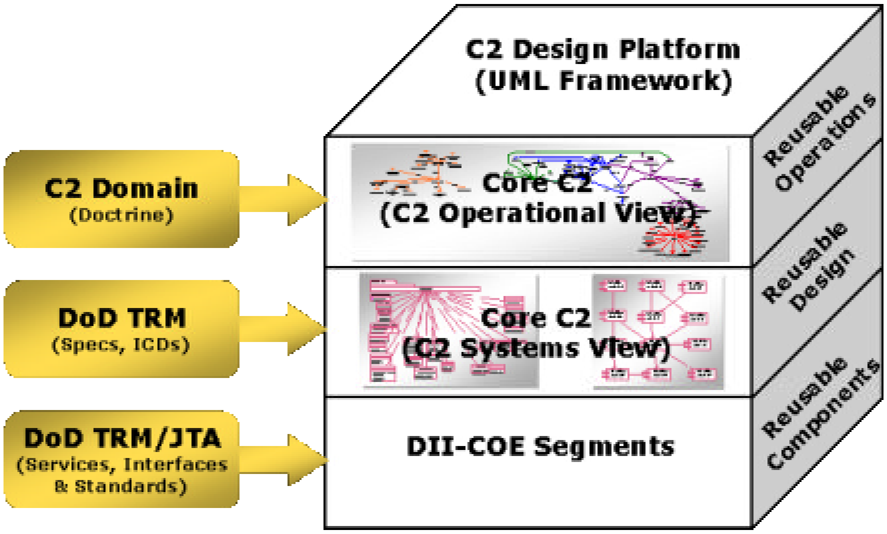


Figure 2.5-2. Reusable Structure of the CCIC2S Architecture Approach

2.5.2 Operational View

The CCIC2S Operational Architecture (OA) Team used OO UML to create an Operational View because of its robustness in symbol sets and OO characteristics such as generalization, specialization, and inheritance. UML use cases[2] may modify (inherit) behavior of a second use case; capture data interaction among operators, nodes, and systems; and allocate behavior responsibility to systems (UML objects) (**Figure 2.5-3**).

[2] Use case - A description of system behavior, in terms of sequences of actions. A use case should yield an observable result of value to an actor. A use case contains all alternate flows of events related to producing the "observable result of value." More formally, a use case defines a set of use-case instances or scenarios. The specification of a sequence of actions, including variants, that a system (or other entity) can perform, interacting with actors of the system.

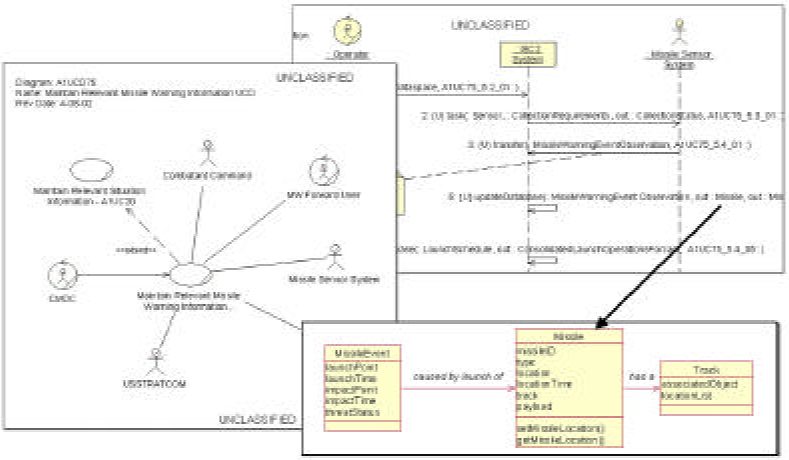


Figure 2.5-3. Understanding and Communicating Requirements

The UML operational architecture approach provides a comprehensive understanding of operational requirements, identifies testing and training requirements earlier in system evolution, determines visually recognizable reuse, works with smaller components, provides an open design space, and focuses on system interfaces. The UML approach has a very high focus on concepts of operations (CONOPS) early in architecture evolution and the visual aspect of use cases provide a standardized method to evolve the system requirements. In addition, operators, subject matter experts, and stakeholders quickly grasp the UML use case concept. A s a result, UML provides a higher level of operator understanding of operational needs by identifying use case observable results of value; scope; operator, element, center, organization, and system roles; and actor specification of a sequence of actions in developing the overall enterprise. Because use cases are highly focused on CONOPS as they evolve to requirements, testing and training planning can occur earlier in the process. Traditionally, testing and training plans are not assembled until the system reaches a maturity level near completion. Because of the nature of use cases, the operational analyst can understand the relationships between use case results of value and more easily identify design reuse in the operational process. The process also lays the foundation for developed components that are smaller and more reusable reducing the cost of potential rework. Using the OO approach to Operational View development, the process lays the foundation for system design without imposing technological restrictions on the developer's solution. Finally, the UML process is highly focused on system interfaces. By focusing on system interfaces, developers can produce a UML Systems Views that shows product line interaction and traditional use case views to design and build the system. The following sections provide a general overview of how this business-reengineering[3] concept presents Operational Views useful in the development of Systems Views.

2.5.2.1 The Operational View Process

The Operational View process begins by identifying relevant use cases with observable results of value (ROV) distinctiveness (e.g., Figure 2.5-3 depicts Missile Warning Information ROV). Identified in the scope of the use case, the results of value are usually data objects (the

[3] Business Reengineering - To perform business engineering where the work of change includes taking a comprehensive view of the entire existing business and thinking through why you do what you do. You question all existing business processes and try to find completely new ways of reconstructing them to achieve radical improvements. Other names for this are business process reengineering (BPR) and process innovation.

beginning of Logical Data Model [OV-7] development) created or maintained by the use case activity. Relationships between use cases, if required (e.g., <<extend>>[4] or <<include>>[5]), are determined by understanding whether the extended use case is a modification (adaptation) of behavior[6] of the parent case (base use case) (generally referred to as generalization/ specialization[7]) or a reusable use case by the base use case. The general way to understand <<extend>> or <<include>> is that we use <<extend>> to “do on a condition of a parent case” and <<include>> “to always use a particular use case.” This is the foundation of use case relations in an operational level UML model and provides much payback in identifying operational patterns and reusability—essential to increasing the efficiency of the development activity. Ultimately, through an iterative process, the use case results in a System Operational Sequence (SOS) assigning behavior responsibility to the system to be built (Figure 2.5-8) using the ideas and concepts of the Rational Unified Process for Systems Engineering (RUP SE).

Once the use case and its associated scope (to include ROV) are well understood, the architect, working with subject matter experts and operators, determines relevant actors and roles involved in the use case activity. UML actors can be organizations, centers, nodes, or systems inside [denoted by] or outside [denoted by] the enterprise. These actors form the relevant nodes for developing the Node Connectivity Description (NCD) OV-2 (**Figure 2.5-4**).

Based on the evolving NCD, again iteratively, the architecture team develops the NCD sequence diagram (**Figure 2.5-5**), a view of OV-5 and OV-2, that describes the desired activities, system transactions, behaviors between nodes and the underlying capabilities that depict the overall desired UML activity. Information exchanges are indicated by the UML message lines (lines with arrows) on the collaboration and sequence views (Figure 2.5-4 and Figure 2.5-5). The collaboration and sequence views relate to each other at the node[8] and information exchange level.

[4] Extend - A relationship from an extension use case to a base use case, specifying how the behavior defined for the extension use case can be inserted into the behavior defined for the base use case.

[5] Include - A relationship from a base use case to an inclusion use case, specifying how the behavior defined for the inclusion use case can be inserted into the behavior defined for the base use case.

[6] Behavior - The observable effects of an operation or event, including its results.

[7] Generalization/specialization - A taxonomic relationship between a more general element and a more specific element. The more specific element is fully consistent with the more general element and contains additional information. An instance of the more specific element may be used where the more general element is allowed.

[8] Node - A representation of an element of architecture that produces, consumes, or processes data.

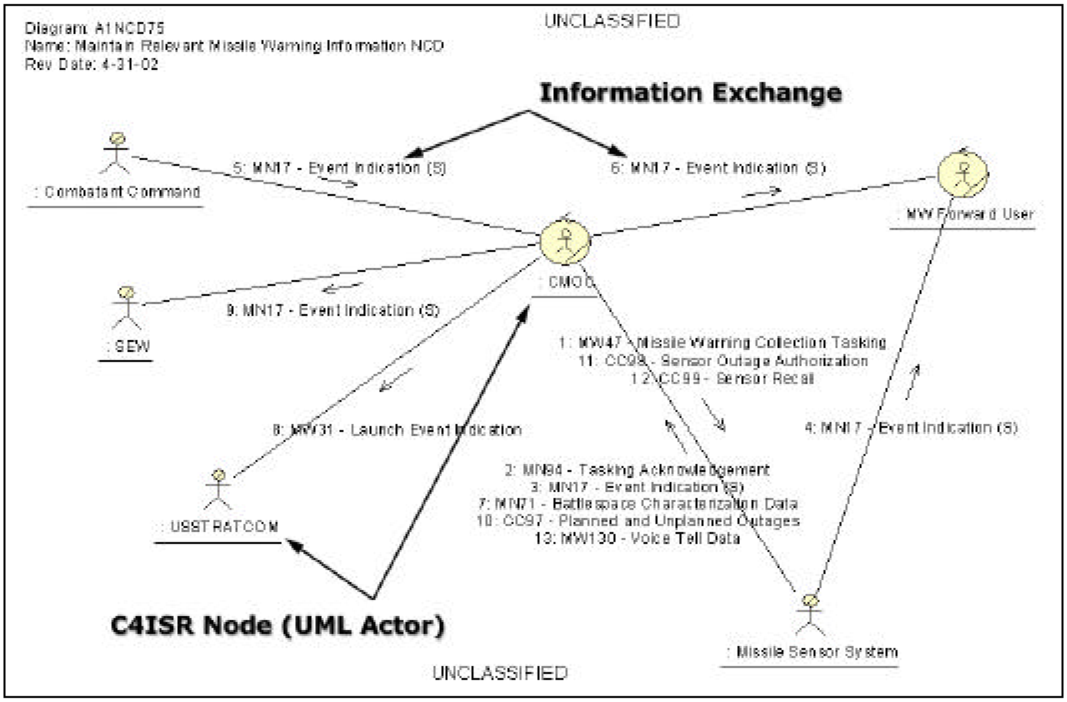

UNCLASSIFIED
Diagram: A1NCD75
Name: Maintain Relevant Missile Warning Information NCD
Rev Date: 4-31-02
Information Exchange
: Combatant Command
5: MN17 - Event Indication (S)
6: MN17 - Event Indication (S)
: MW Forward User
: CMOC
: SEW
9: MN17 - Event Indication (S)
1: MW47 - Missile Warning Collection Tasking
11: CC98 - Sensor Outage Authorization
12: CC99 - Sensor Recall
8: MW31 - Launch Event Indication
4: MN17 - Event Indication (S)
2: MN94 - Tasking Acknowledgement
3: MN17 - Event Indication (S)
7: MN71 - Battlespace Characterization Data
10: CC97 - Planned and Unplanned Outages
13: MW130 - Voice Tell Data
: USSTRATCOM
C4ISR Node (UML Actor)
: Missile Sensor System
UNCLASSIFIED

Figure 2.5-4. Node Connectivity Description (UML Collaboration Diagram)

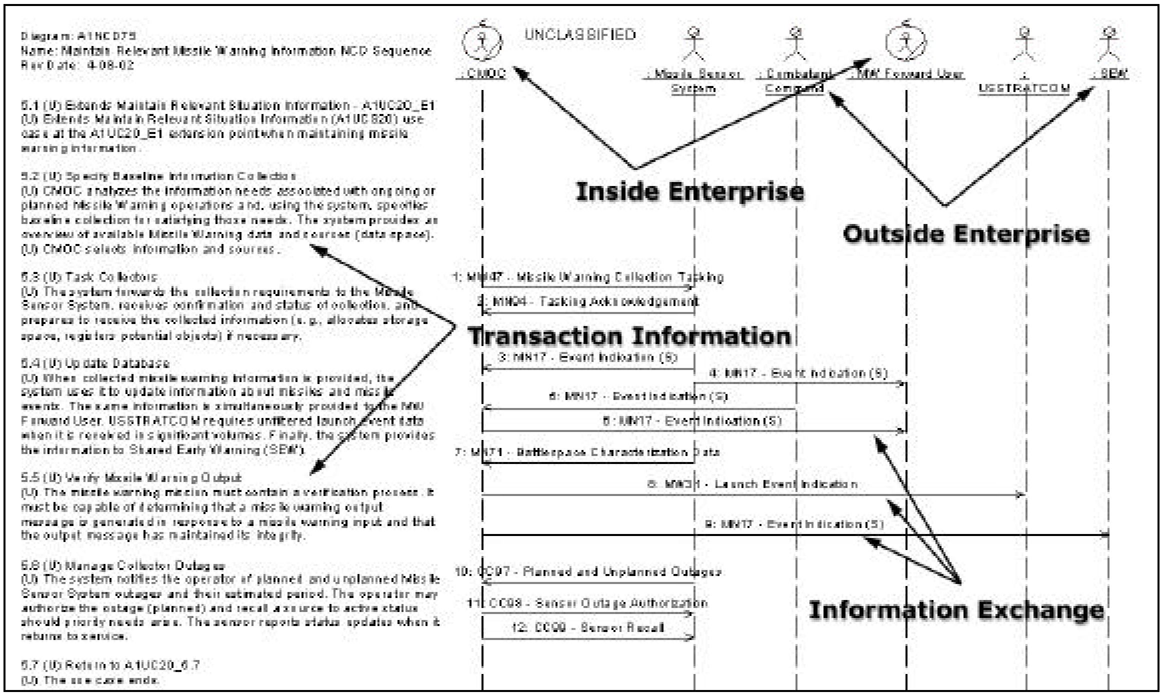


Diagram: A1NCD75
Name: Maintain Relevant Missile Warning Information NCD Sequence
Rev Date: 4-08-02

5.1 (U) Extends Maintain Relevant Situation Information - A1UC20_E1
(U) Extends Maintain Relevant Situation Information (A1UCS20) use case at the A1UC20_E1 extension point when maintaining missile warning information.

5.2 (U) Specify Baseline Information Collection
(U) CMOC analyzes the information needs associated with ongoing or planned Missile Warning operations and, using the system, specifies baseline collection for satisfying those needs. The system provides an overview of available Missile Warning data and sources (data space).
(U) CMOC selects information and sources.

5.3 (U) Task Collectors
(U) The system forwards the collection requirements to the Missile Sensor System, receives confirmation and status of collection, and prepares to receive the collected information (e.g., allocates storage space, registers potential objects) if necessary.

5.4 (U) Update Database
(U) When collected missile warning information is provided, the system uses it to update information about missiles and missile events. The same information is simultaneously provided to the MW Forward User. USSTRATCOM requires unfiltered launch event data when it is received in significant volumes. Finally, the system provides the information to Shared Early Warning (SEW).

5.5 (U) Verify Missile Warning Output
(U) The missile warning mission must contain a verification process. It must be capable of determining that a missile warning output message is generated in response to a missile warning input and that the output message has maintained its integrity.

5.6 (U) Manage Collector Outages
(U) The system notifies the operator of planned and unplanned Missile Sensor System outages and their estimated period. The operator may authorize the outage (planned) and recall a source to active status should priority needs arise. The sensor reports status updates when it returns to service.

5.7 (U) Return to A1UC20_5.7
(U) The use case ends.

UNCLASSIFIED
: CMOC
: Missile Sensor System
: Combatant Command
: MW Forward User
: USSTRATCOM
: SEW
Inside Enterprise
Outside Enterprise
1: MW47 - Missile Warning Collection Tasking
2: MN94 - Tasking Acknowledgement
Transaction Information
3: MN17 - Event Indication (S)
4: MN17 - Event Indication (S)
5: MN17 - Event Indication (S)
6: MN17 - Event Indication (S)
7: MN71 - Battlespace Characterization Data
8: MW31 - Launch Event Indication
9: MN17 - Event Indication (S)
10: CC97 - Planned and Unplanned Outages
11: CC98 - Sensor Outage Authorization
12: CC99 - Sensor Recall
Information Exchange

Figure 2.5-5. NCD Sequence (UML Sequence Diagram)

The process continues until the use case relationship diagram represents the overall desired activities and their relationships (item 2 in **Figure 2.5-6**). There are other significant views such as actor relationships (refined command relationships ~ OV-4), Operational Information Exchange Matrix (OV-3) database, and the use case specification, which are outside the scope of this case study discussion. Together, their views and relationships are the foundation for understanding the desired operational behavior to form the operational requirements. The overall relationships between the products views are depicted in **Figures 2.5-6** through **2.5-11** showing product relationships 1 through 12 (shown by numbers in yellow circles).

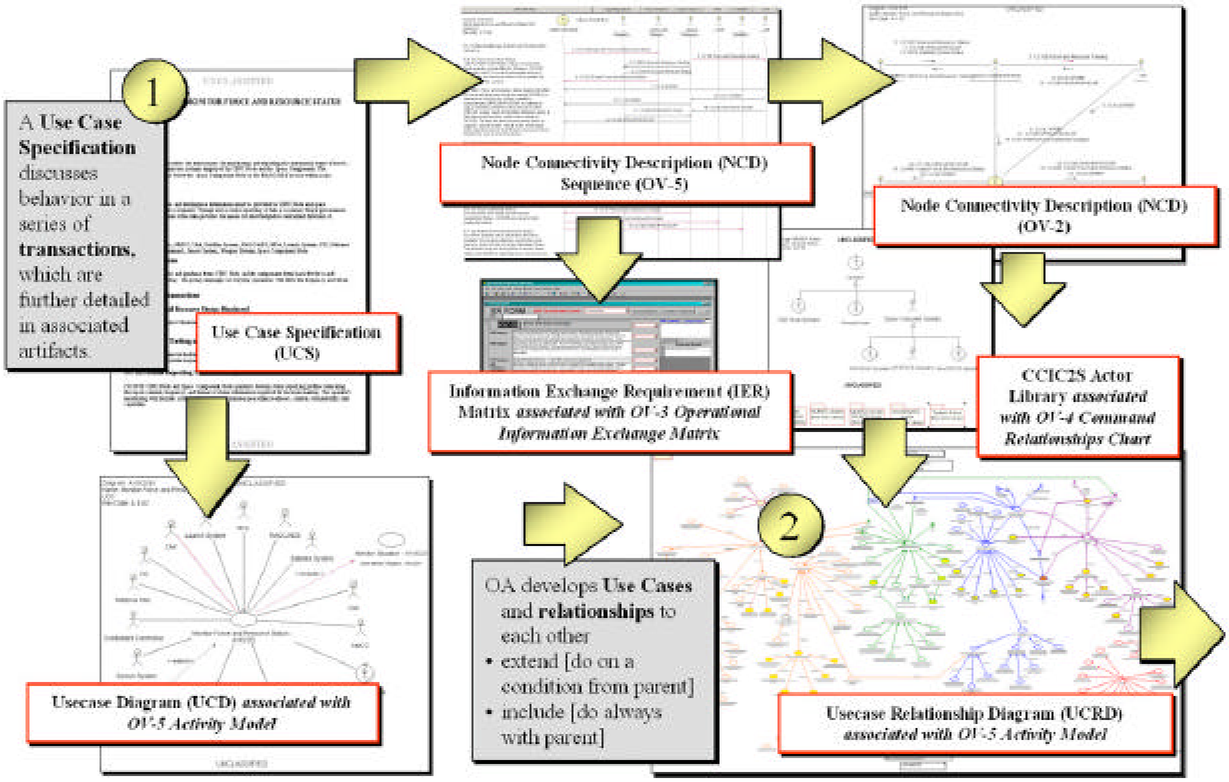

1
A Use Case Specification discusses behavior in a series of transactions, which are further detailed in associated artifacts.
Use Case Specification (UCS)
Node Connectivity Description (NCD) Sequence (OV-5)
Node Connectivity Description (NCD) (OV-2)
Information Exchange Requirement (IER) Matrix associated with OV-3 Operational Information Exchange Matrix
CCIC2S Actor Library associated with OV-4 Command Relationships Chart
Usecase Diagram (UCD) associated with OV-5 Activity Model
OA develops Use Cases and relationships to each other
• extend [do on a condition from parent]
• include [do always with parent]
2
Usecase Relationship Diagram (UCRD) associated with OV-5 Activity Model

Figure 2.5-6. Creating C4ISR Architecture Framework Operational Views

Using action verb titles derived from the primary transactions developed in the NCD sequence (Figure 2.5-5), the architect develops the use case activity diagram (Operational Activity Model [OV-5]) (note: one per use case), which shows the key decision points in the operational flow and provides the view to identify consuming and producing data objects for each activity. Again, the details of this process are beyond the scope of this case study. However, because of its simplicity and visual depiction of operations, the use case activity diagram is a popular view with operators and stakeholders. Activity diagrams make it easier to understand the process, and they show the operational flow of information necessary to support the operational activity and the underlying Logical Data Model (OV-7). See **Figure 2.5-7**.

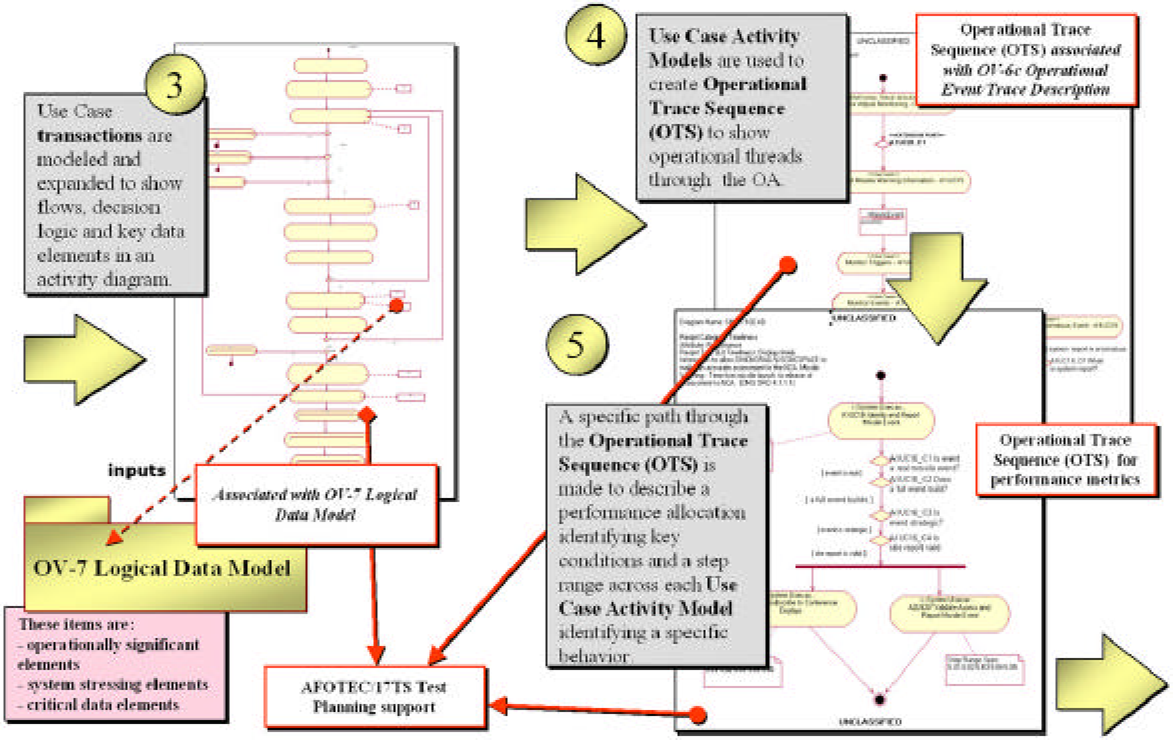

3
Use Case transactions are modeled and expanded to show flows, decision logic and key data elements in an activity diagram.
inputs
Associated with OV-7 Logical Data Model
OV-7 Logical Data Model
These items are:
- operationally significant elements
- system stressing elements
- critical data elements
AFOTEC/17TS Test Planning support
4
Use Case Activity Models are used to create Operational Trace Sequence (OTS) to show operational threads through the OA.
Operational Trace Sequence (OTS) associated with OV-6c Operational Event Trace Description
5
A specific path through the Operational Trace Sequence (OTS) is made to describe a performance allocation identifying key conditions and a step range across each Use Case Activity Model identifying a specific behavior.
Operational Trace Sequence (OTS) for performance metrics
UNCLASSIFIED

Figure 2.5-7. Developing the Logical Data Model (OV-7) and Operational Event-Trace Description (OV-6c)

Once use cases mature to a conceptual level to include ROV and scope, again using UML activity models, they are abstracted to a UML activity, stereotyped as <<use case>>, and form the basis for building operational threads through the model. Called Operational Trace Sequences (OTS) (~ OV-6c), these high-level views are useful for representing key performance parameters, thresholds, and objectives, and for building conceptual operational threads throughout the model. Operationally significant data objects and the information they contain form the information building block to describe significant operational flow through the use cases. The OTS provides a way to represent CONOPS and thread the pieces of the model productively. Rational Software views the OTS as a high-level scenario[9] oriented use case. After demonstrating how to represent all the necessary DoDAF views in UML, there was, initially, no way to transition from Operational Views to Systems Views. As a result, a new view not discussed by the DoDAF called SOS was created.

2.5.2.2 Transition from Operational View to Systems View

To solve the transition problem, an additional product was added to those identified by the DoDAF to join the Operational View and Systems View when using OO techniques. Specifically, a UML sequence diagram (**Figure 2.5-8**) (Note: one per use case) was developed to explicitly allocate system responsibilities (transactions conveying data objects) to systems that satisfy the behavioral requirements identified in the Operational View. The principles are the same as those discussed in the RUP SE.

[9] Scenario - A described use case instance; a subset of a use case. A specific sequence of actions that illustrates behaviors. A scenario may be used to illustrate an interaction or the execution of a use case instance.

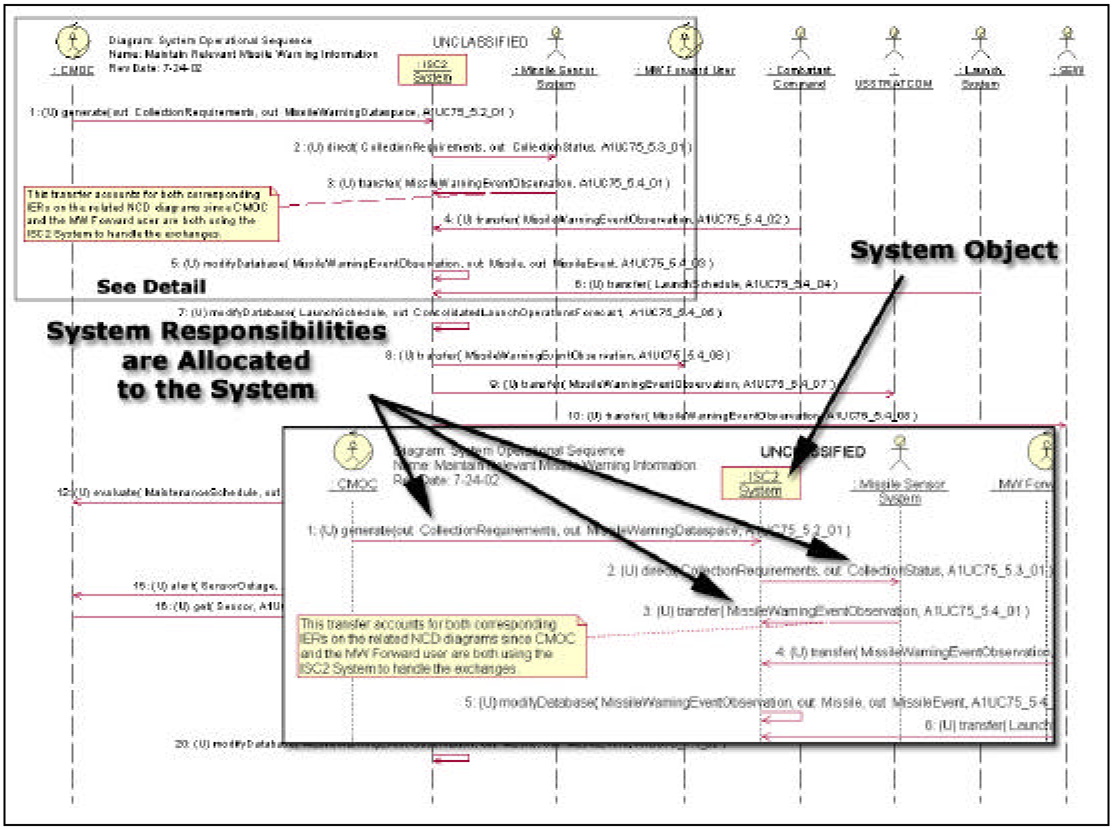


Diagram: System Operational Sequence
Name: Maintain Relevant Missile Warning Information
Rev Date: 7-24-02
UNCLASSIFIED
: CMOC
: ISC2 System
: Missile Sensor System
: MW Forward User
: Combatant Command
: USSTRATCOM
: Launch System
: SBIRS
1 : (U) generate(out CollectionRequirements, out MissileWarningDataspace, A1UC75_5.2_01)
2 : (U) direct(CollectionRequirements, out CollectionStatus, A1UC75_5.3_01)
3: (U) transfer(MissileWarningEventObservation, A1UC75_5.4_01)
4: (U) transfer(MissileWarningEventObservation, A1UC75_5.4_02)
This transfer accounts for both corresponding IERs on the related NCD diagrams since CMOC and the MW Forward user are both using the ISC2 System to handle the exchanges.
5: (U) modifyDatabase(MissileWarningEventObservation, out Missile, out MissileEvent, A1UC75_5.4_03)
6: (U) transfer(LaunchSchedule, A1UC75_54_04)
7: (U) modifyDatabase(LaunchSchedule, out ConsolidatedLaunchOperationsForecast, A1UC75_5.4_05)
8: (U) transfer(MissileWarningEventObservation, A1UC75_5.4_06)
9: (U) transfer(MissileWarningEventObservation, A1UC75_5.4_07)
10: (U) transfer(MissileWarningEventObservation, A1UC75_5.4_08)
12: (U) evaluate(MaintenanceSchedule, out
15: (U) alert(SensorOutage,
16: (U) get(Sensor, A1U
20: (U) modifyDatab
See Detail
System Object
System Responsibilities are Allocated to the System

Figure 2.5-8. System Operational Sequence

This product yields several key advantages that includes the following:

- Provides a detailed basis for tracing operational activities to system functions, contributing the operational activities identified in the Operational Activity to System Function Traceability Matrix (SV-5) of the Systems View
- Allows a single Operational View to support multiple systems as well as the force providers and the programs that build them
- Facilitates the expression of lower-level CONOPs
- Clearly identifies system boundary behavior

Thus, the scope of an Operational View can be expanded beyond a single system to better define cohesive operations across a whole domain or enterprise and still explicitly allocate behavioral requirements to one or more system.

Use cases provide insight into what operational activities the system must support and to whom the supporting system capabilities must be delivered. This provides important system to node allocation information in developing the Systems Interface Description (SV-1) and Systems Communications Description (SV-2).

To better facilitate requirements management and provide a way to be tool independent, a Rational Rose Script called the System Responsibility Report was developed (**Figure 2.5-9**) that pulls all the information out of the SOS and builds a comma separated view (CSV) file importable into nearly any application (e.g., Excel, Access, Oracle, etc.). The added view also allows additional traceability to logical data model elements and provides a means to perform horizontal analysis on the requirements. Finally, this additional view provides additional linkage to the Systems View.

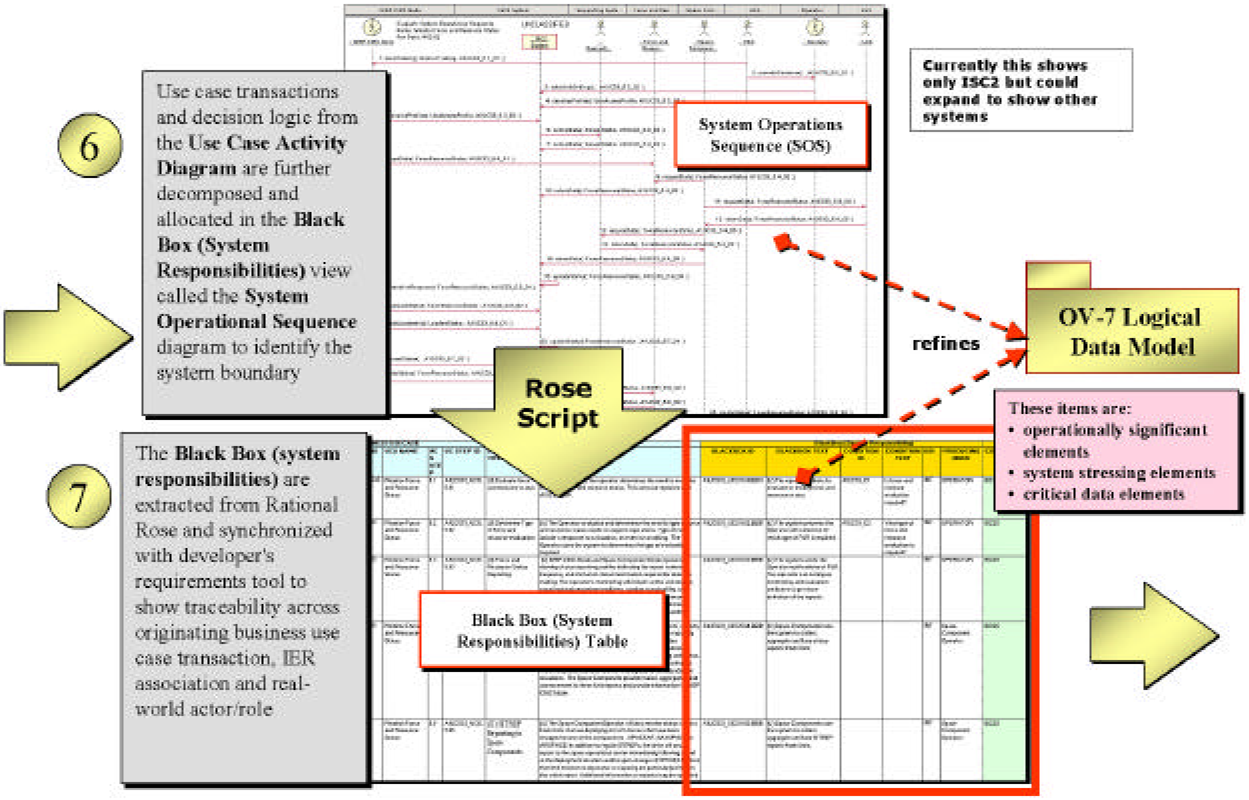

6
Use case transactions and decision logic from the **Use Case Activity Diagram** are further decomposed and allocated in the **Black Box (System Responsibilities)** view called the **System Operational Sequence** diagram to identify the system boundary
System Operations Sequence (SOS)
Currently this shows only ISC2 but could expand to show other systems
refines
OV-7 Logical Data Model
Rose Script
These items are:
• operationally significant elements
• system stressing elements
• critical data elements
7
The **Black Box (system responsibilities)** are extracted from Rational Rose and synchronized with developer's requirements tool to show traceability across originating business use case transaction, IER association and real-world actor/role
Black Box (System Responsibilities) Table

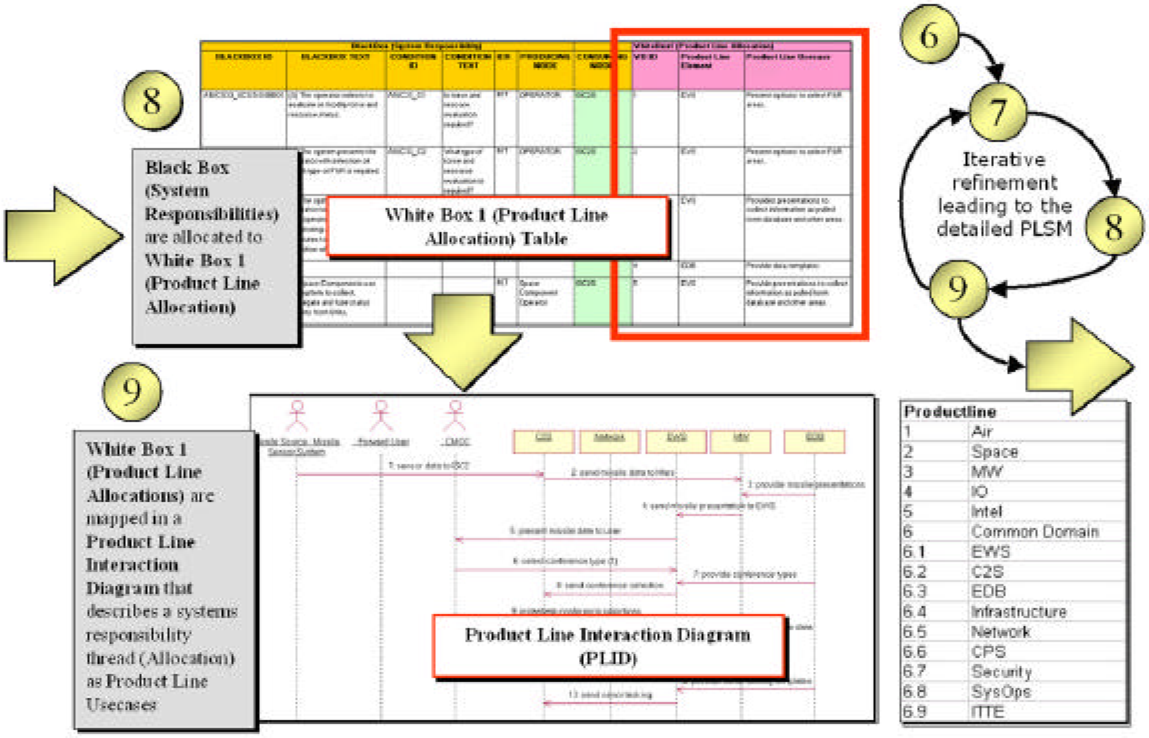
Figure 2.5-9. Developing System Responsibilities

Linkage between the Operational and Systems Views is also established through the Logical Data Model (OV-7). This model identifies operationally significant objects and their relationships. The Systems View products show inheritance or traceability to these objects via generalization or dependency mechanisms. Therefore, they directly influence the objects in the Physical Schema (SV-11) in a loosely coupled manner. In addition, these objects are used on the OTS to describe the object dependency in operational threads.

OV-6c is used to highlight dynamics associated with key operational threads through the architecture (e.g., demonstrate how missions are supported and indicate how performance metrics apply to operations). These are associated with the Systems Event-Trace Descriptions (SV-10c), to support testing as capabilities are fielded. The OTS describes the required behavior of the system, and SV-10c identifies what portions of the system provide the behavior. Using these in combination, the testers can determine for which behavior to test and what system configuration to test.

2.5.3 Systems View

The Systems View combines the elements of the Technical Standards View (TV) to provide the behavior described in the Operational View. Meta-models of the various products were constructed to ensure semantic linking of the Operational and Systems Views. Although this approach can accommodate multiple systems and developers, the primary focus, at this time is on the Integrated Space Command and Control (ISC2) system contractor. A layered architecture approach was adopted in concert with TRM recommendations. The Systems View draws on product lines and products identified in these layers to structure the components that satisfy system responsibilities allocated to the system in the Operational View (**Figure 2.5-10**). The traceability (from the OV-generated system responsibilities to the system components that provide functionality) is accomplished through a recursive set of sequence diagrams, allocating the responsibilities to progressively finer-grain system elements from product lines through their constituent components (see “Rational Unified Process for System Engineering 1.0, a Rational Software White Paper” for discussion of such an approach).


8
Black Box (System Responsibilities) are allocated to White Box 1 (Product Line Allocation)
White Box 1 (Product Line Allocation) Table
6
7
Iterative refinement leading to the detailed PLSM
8
9
White Box 1 (Product Line Allocations) are mapped in a Product Line Interaction Diagram that describes a systems responsibility thread (Allocation) as Product Line Usecases
Product Line Interaction Diagram (PLID)

Productline	
1	Air
2	Space
3	MW
4	IO
5	Intel
6	Common Domain
6.1	EWS
6.2	C2S
6.3	EDB
6.4	Infrastructure
6.5	Network
6.6	CPS
6.7	Security
6.8	SysOps
6.9	ITTE

Figure 2.5-10. Mapping System Responsibilities to Product Lines

This process establishes the architectural and design structure that ensures that the components work together to produce the required system behavior needed by operators to conduct operations (**Figure 2.5-11**). Design activities such as modeling, coding, other generation representations (e.g., XML), and roundtrip engineering then produce the code that completes the system. Using the TRM, Specifications, and Interface Control Documents are applied at this level of design. Deployment views show how the software components are fielded on hardware components as well as how the latter are interconnected. This becomes a further basis for SV-1 and SV-2.

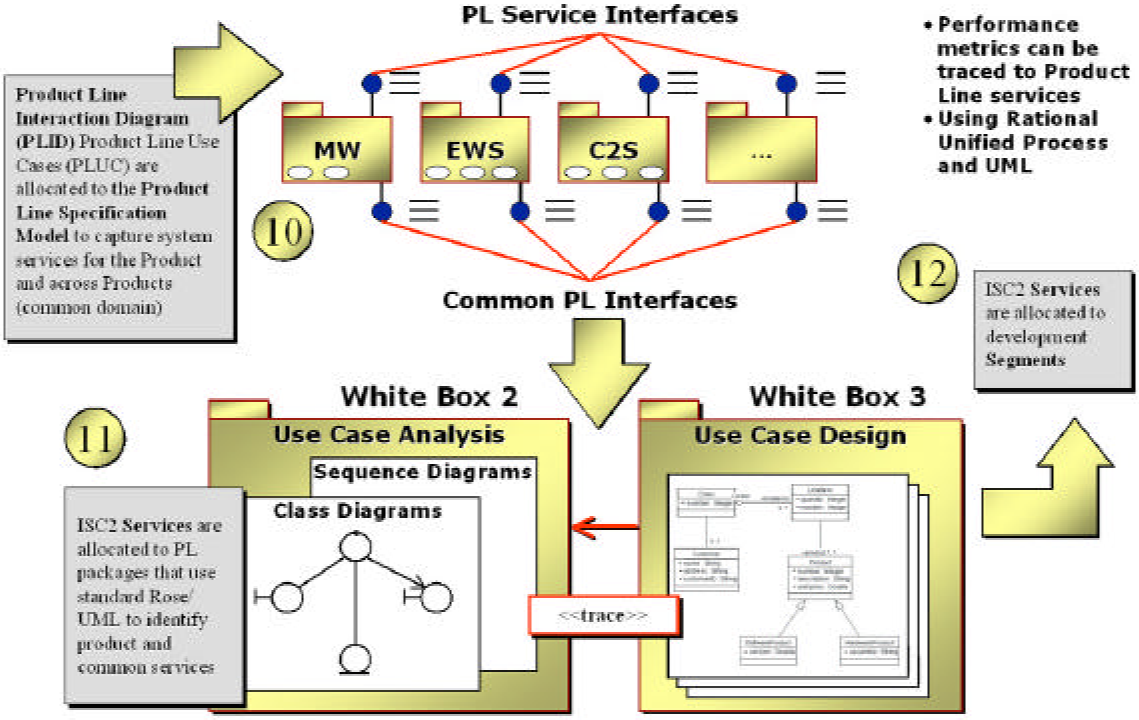

PL Service Interfaces
• Performance metrics can be traced to Product Line services
• Using Rational Unified Process and UML
Product Line Interaction Diagram (PLID) Product Line Use Cases (PLUC) are allocated to the Product Line Specification Model to capture system services for the Product and across Products (common domain)
MW
EWS
C2S
...
10
12
ISC2 Services are allocated to development Segments
Common PL Interfaces
11
White Box 2
Use Case Analysis
Sequence Diagrams
Class Diagrams
ISC2 Services are allocated to PL packages that use standard Rose/ UML to identify product and common services
<<trace>>
White Box 3
Use Case Design

Figure 2.5-11. Refining Product Line Allocation to Components

While the Systems View describes the "To Be" visionary architecture, there is a transition from legacy systems to future systems that embody the architecture. One must convey how the intervening mixture of legacy and future systems are deployed and cooperate to maintain continuity of operations over the development period. This is accomplished through a series of data-driven deployment diagrams based on operational delivery plans.

2.5.3.1 Transition from Systems View to Technical Standards View

The nominal interpretation of the Technical Standards View is a minimal but sufficient time-phased list of standard technology specifications applicable to the realization of the system's requirements. Time phasing consists of delineating the current specifications, forecasting the major technology standardization trends and updating the associations between standards and system architecture elements as the system matures through its evolution. Since the standards are selected after related system capabilities are identified, it is natural to think in terms of a sequential process in which the specifications are selected after the system architecture has been defined. However, the real process, whether formal or not, involves intense analytical interaction between the system architecture and technical architecture domains. From the system architecture perspective, the design of both the logical and physical aspects is informed and constrained by what the architects consider practical in terms of available technology, standards, and architectural patterns. From the perspective of the technical architecture, the selection of specifications from the vast domain of technology specifications must, in turn, be filtered by the context of the system architecture.

The success of this highly iterative interaction between views is currently quite dependent on the artfulness and experience of the architects. This is evident in light of orthogonal integration of emerging command and control (C2) standards, such as Common

Operating Environment (COE), with other major system architecture concepts. The future hope is that the framework will mature to provide readily accessible architecture patterns and technology specifications pre-organized by system domain.

The range of specifications and patterns that must be considered in this process within the domain of strategic C2 systems are currently dominated by component container middle-ware technology, tiered client-server patterns, object-relational mapped persistence, public key infrastructure, and communication technologies that cross the full spectrum of geographic distribution. In addition, technologies developed by the DISA Network-Centric Enterprise Services (NCES, formerly the DII COE) effort have recently become available for this domain. Significant examples of COE standards include workstation and user interface facilities, various types of message processing, and software build configuration management mechanisms. Further, each of these standard areas is experiencing strong change trends that must be forecast against the expected evolutionary life of the system.

2.5.3.2 Traceability from Systems View to Technical Standards View

The problem of associating the domain's standards, patterns and their trends with logical and physical elements in the system architecture is a relational challenge for both system and technical architectures. The ISC2 developer addressed the problem with a few principles.

The first principle is called "greatest scope of technical constraint" in which a technical constraint should be mapped to the applicable system architecture element with the largest technical scope. The primary benefit of using the greatest scope principle is that the scope hierarchy inherent in the system architecture is leveraged to eliminate tedious, redundant, error-prone and probably unsustainable mappings between details of a standard and the recursive decomposition of the relevant system architecture element. (For example, user interface standards should be mapped to the enterprise workstation rather than each individual user interface display, or a security guard pattern should be mapped to the entire communication processing system rather than each individual guard element).

The second principle is "no orphan standards." This principle is easy to understand but may be difficult to implement given the sheer size of strategic C2 domain. It can be a substantial effort to review the mapping to ensure that every standard listed in the technical architecture is indeed associated with and appropriately constrains some system architecture element. There are several obvious benefits: reduced architect and implementer learning load and reduced workload for quality assurance. A more subtle benefit is that building and cross-checking the mapping for orphans provides an important cognitive review of the architecture. These principles lead to a simple relational expression for the mapping in **Figure 2.5-12**.

This pattern is easily implemented using any relational tool such as a relational database (i.e., Access, Oracle, etc.).

2.5.4 Technical Standards View

To summarize in terms of traceability, the Operational View provides the source of functional requirements for the CCIC2S enterprise system. The functionality identified is based on Tier 1 C2 Battle Management and support mission functions based on traditional (existing) and non-traditional (new, emerging) threats. The CCIC2S-ISC2 requirements flow-down process allows the capability to be defined, refined, aligned, and allocated in terms of functional and performance requirements to selected logical systems of interest (associated with current existing or future systems). The functionality is subject to factoring, aggregation, consolidation, and realignment to core system capabilities. The allocations also include the complex cross-mapping with legacy (existing) systems, including both those that are and are not migrating and the new/emerging systems of interest. This is to focus uniquely on mission applications while enabling the reduction/consolidation of infrastructure and common/shared functionality. The next flow-down association is to the Technical Standards View.

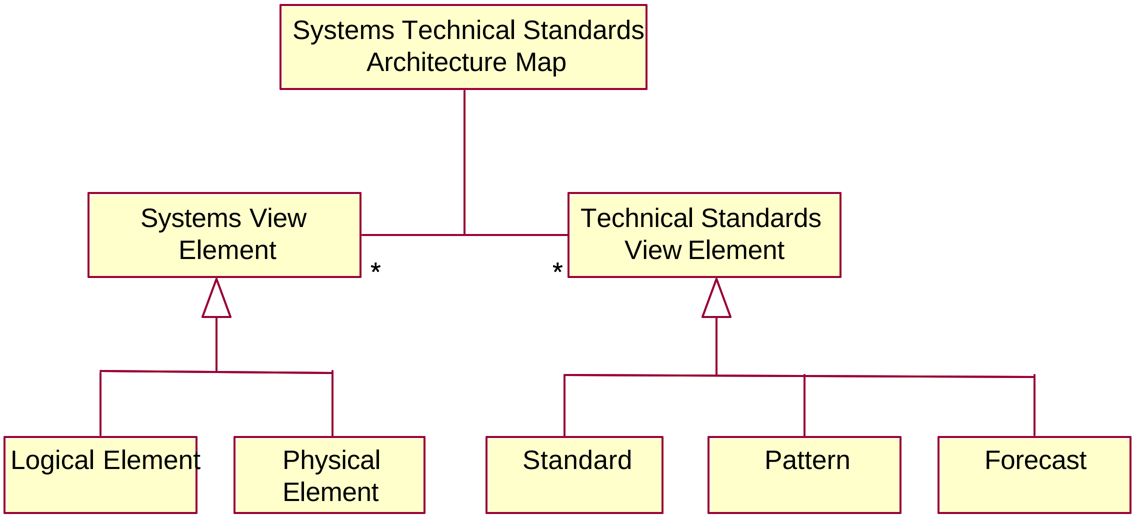

Systems Technical Standards Architecture Map
Systems View Element
Technical Standards View Element
*
*
Logical Element
Physical Element
Standard
Pattern
Forecast

Figure 2.5-12. Systems to Technical Standards Architecture Map

The Joint Technical Architecture (JTA) provides DoD with the fundamental building codes for the warfighter to develop the capability for interoperability, seamless information flow, and plug-and-play. The CCIC2S program assimilated the JTA, the current As-Is CCIC2S environment and developed a minimum set of standards, the CCIC2S TA (CSTA), which is applied to the ISC2 program. This set of mandated standards and guidelines provided the starting point for the evolution of the CCIC2S enterprise systems architecture. The ISC2 developer took this set of standards and applied in-depth industry evaluations, trade studies, and comparative analyses with other standards needed to achieve the defined system functionality that was allocated to ISC2 system from the CCIC2S Operational View. This evaluation continues, to keep the comparative analysis current. It provides a constant forward-looking perspective to exploit new standards and technology, a constant examination of others standards within the CCIC2S enterprise to consider in the ISC2 Systems View, and a constant scrutiny of the need for a corresponding service within the ISC2 enterprise system. The technology forecast tracks near- and long-term technology trends in order to identify promising new technologies that can be effectively applied to reduce ISC2 evolution risks and costs and increase capability.[10] New technologies are constantly evaluated from a cost/benefit standpoint to determine applicability to future ISC2 releases to deliver the most capability for the minimum cost.

The ISC2 program falls within the NCES COE environment and also the Tier 1 and lower echelon C2 business area. Currently, the Global Command and Control System (GCCS) is emerging as the core warfighter distributed, federated C4I system. By default, the set of capabilities with the DISA-provided COE become the starting point for capabilities mapping

[10] In the DoDAF, technology forecasts are contained in the Systems Technology Forecast (SV-9). However, in practice, some combine the SV-9 with the Technical Standards Forecast (TV-2).

between the associated system-define capabilities and the DISA-provided COE capabilities. The ISC2 developer is tasked to reuse, expand, enhance, tailor, or build new COE capabilities following the standard DISA processes. The ISC2 developer is exploring enhancing these processes to support faster cycles for spiral evolution, for research and development, and to meet user needs for short/no-notice response mission demands. The ISC2 program open standards approach emphasizes architecture constraints and driving requirements in the selected standards and technologies, which are defined in the CSTA.

The demands of legacy system migration provide a challenge to TV-1 and TV-2. The ISC2 developer must manage a diverse set of standards that at times conflict or cannot be applied until a certain point in the phased migration of legacy systems. And this must be done with no impact to ongoing mission operations. This complex cross-phasing has been described as changing an engine while in flight. The ISC2 developer has established a full cross-matrixed ISC2 product line with capability deliveries and synchronization points that align selected systems migrations/deployments. This is managed in the ISC2 Master Integrated Evolution Plan (IMEP). This evolution involves applying the CSTA to enable a core systems infrastructure and a core database infrastructure. These areas are further permutated by other CCIC2S enterprise systems (outside the current scope of the ISC2 program) that are themselves migrating/evolving and which are within the JTA but not completely compliant with the ISC2 Technical Standards View. The ISC2 developer must maintain situational awareness of all external ISC2 interfaces (functionally derived from the Operational View) to apply a standard industry or custom technological solution to bridge these systems. The ISC2 developer uses the TV-1 and TV-2 to inform, collaborate, or guide other programs that need to interface or integrate with ISC2 core systems. This common foundation also supports system-of-systems testing, joint testing, scalable product line, flexibility in evolution, and higher fidelity in a capability component-based architecture.

Additional challenges are emerging with the DISA-provided COE and the GCCS environment in terms of standards that define specific capabilities of legacy systems that are either not needed or used by the GCCS community as a whole (due to mission uniqueness) or require state-of-the-art abilities to support real-time information exchanges or capabilities (such as for battle management execution and runtime). The ISC2 developer is also working to evolve multiple environments across various missions to a common framework while being constrained by current legacy warfighter processes and warfighting paradigms. Another challenge to a common TA application is the distributed nature of the CCIC2S environment into warfighter environments (Combatant Command/Theaters). The ISC2 developer is focusing on instantiating mission portals, either as client/server or very normalized COTS/GOTS structures to be able to respond to warfighter mission needs, in some instances regardless of whether the warfighter is COE-compliant or noncompliant (such as Web and client/server technologies). The ISC2 TV-1 and TV-2 documents have to be dynamic "living" documents and identify elements that are sustainable and affordable.

The ISC2 developer is evolving a DISA-provided COE-compliant system with standard segment taxonomy structure aligned with the CSTA. The ISC2 product line was also designed from its inception to have a similar taxonomy of functionality partitioning aligned with the DISA-provided COE architecture including the concepts of API layers, kernel capabilities, SHADE, COTS and GOTS, style guides, and segmentation design. The ISC2 developer is evolving the ISC2 system to meet COE compliance in both structures, constructs, and processes. The development of COE applications and components each offers various options on sustainment and processes to follow. The ISC2 Developer is evolving these options successfully to facilitate spiral evolution and enhancement of migration.

2.5.4.1 Traceability from Technical Standards View to Technical Reference Model

The purpose of the TRM is to provide a common conceptual framework in a defined common vocabulary of the various components of the target system. The TRM provides the taxonomy for identifying a discrete set of conceptual layers, entities, interfaces and diagrams that provides for the specification of standards. The IMEP, Section 2 – Target System Architecture, is designed to be aligned with TRM constructs. The ISC2 architecture model describes the application layers, data services, distributed operations management services, middleware services, network, platform, security, and Web services. The ISC2 TSA was designed to support the DoD TRM and to provide a common vocabulary to define the ISC2 open systems services and capabilities to enable interoperability, scalability, and software reuse and to facilitate product line manageability. The ISC2 product line aligns with the CCIC2S Operational View, COE, and TRM constructs. The ISC2 architecture solution is a standards-based implementation of an e-business system design, leveraging mature commercial capabilities to bring robust mission capabilities to any authorized warfighter, anywhere, at any time. It also provides a high degree of flexibility and scalability to accommodate changes in CONOPS, threats, and the resultant impacts on sensors, internal and external interfaces, mission capabilities, and users. The ISC2 net-centric model shown in **Figure 2.5-13** demonstrates how the ISC2 product line is aligned with the COE segmentation approach.

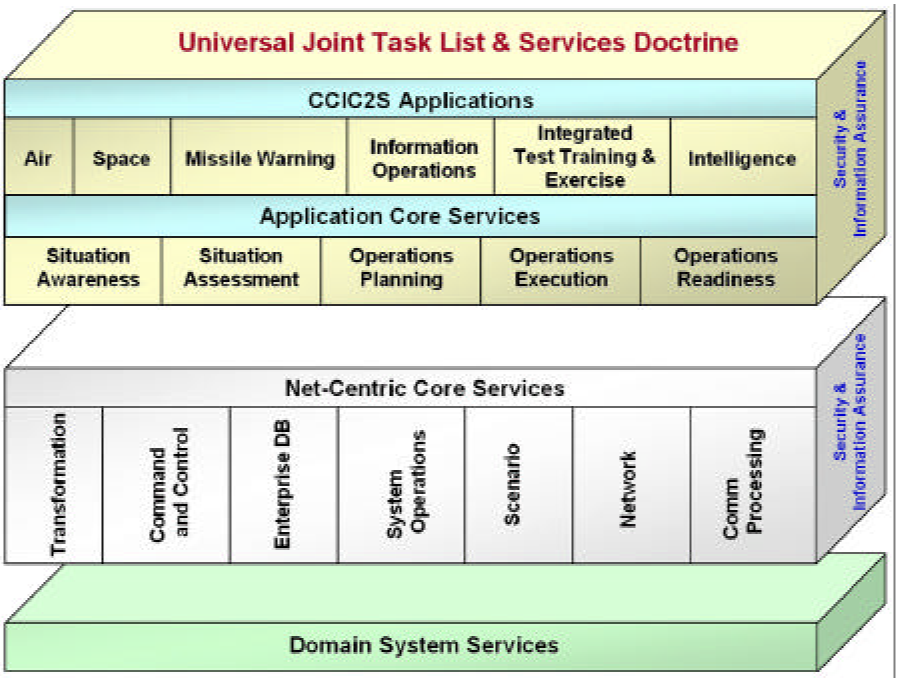
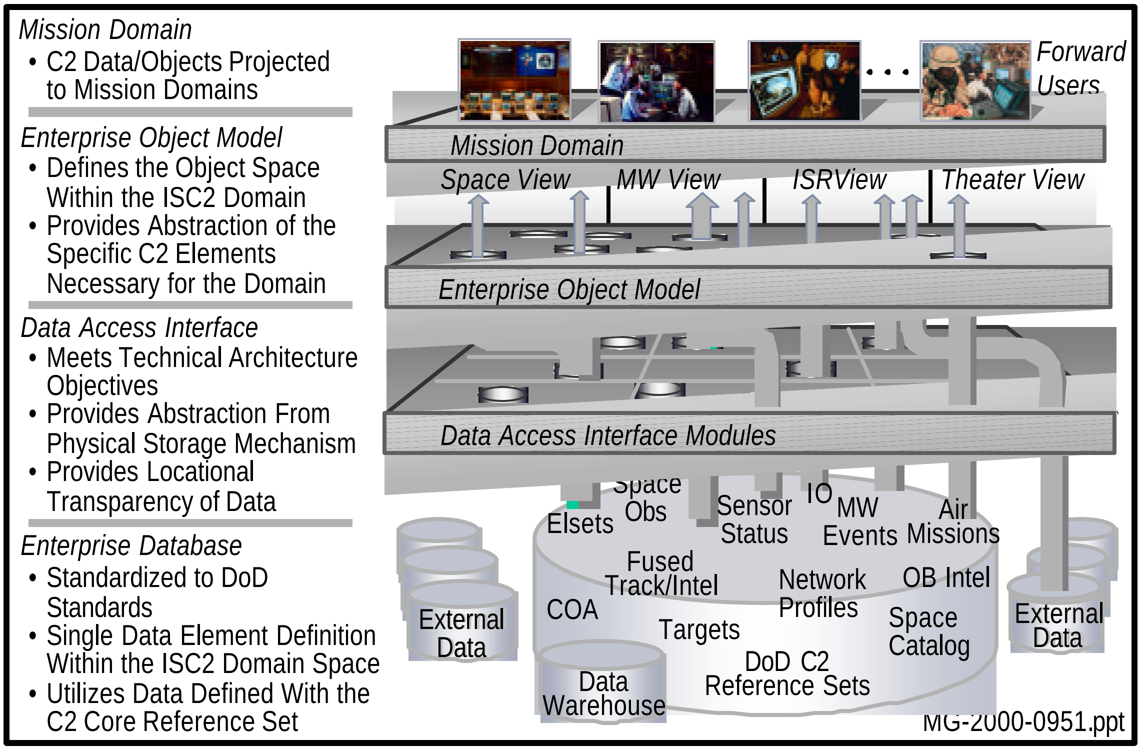


Figure 2.5-13. Integrated Space Command and Control Net-Centric Model

The Enterprise Database is based on a single OV-7 traced from the CCIC2S Operational View. These data elements are standardized according to the DoD Data Element Standardization requirements (DoD Std 8320-1), where appropriate. Existing C2 Core Reference Sets/Models are used where those definitions and ISC2 requirements coincide. The Enterprise Object Model defines the hierarchy of the data objects and available methods to implement the C2 business rules within the ISC2 system. The example below (**Figure 2.5-14**) from the IMEP, TSA section, demonstrates the Data Access Interface layer relationship between the ISC2 OV-7 and the Enterprise Object Model.



Mission Domain

- C2 Data/Objects Projected to Mission Domains

Enterprise Object Model

- Defines the Object Space Within the ISC2 Domain
- Provides Abstraction of the Specific C2 Elements Necessary for the Domain

Data Access Interface

- Meets Technical Architecture Objectives
- Provides Abstraction From Physical Storage Mechanism
- Provides Locational Transparency of Data

Enterprise Database

- Standardized to DoD Standards
- Single Data Element Definition Within the ISC2 Domain Space
- Utilizes Data Defined With the C2 Core Reference Set

Figure 2.5-14. ISC2 Logical Data Model and the Enterprise Object Model

The IMEP, Section 4 – Evolution, visually provides target representations of the multiple system evolution by fiscal year. The system evolution is based on the physical environment including localities, devices, systems, communications infrastructures, and interfaces both internal and external. The IMEP, TSA section also contains visual representations of TRM relationships across other common services, infrastructure elements, and mission applications.

2.5.5 Conclusion

The Integrated Space Command and Control net-centric model (details depicted in the ISC2 documentation) represents the initial derivation of a domain C2 reference model using the DoD TRM. This model is in the process of being fully transitioned into the DoD Net-Centric Operational and Warfare (NCOW) Reference Model (RM) under development by the DoD TRM Working Group. Current service definitions of the ISC2 model are consistent with those offered by the NCOW RM and are easily accommodated within that model. As the DoD NCOW RM is evolved and baselined, it is expected to enhance the ISC2 model. This parallels previous efforts in the development of the DoD TRM that subsequently resulted in the establishment of a singular referential platform-centric TRM that is tailorable for all DoD domains. The expectation in this early stage of model maturation is that identification of and convergence to a DoD NCOW RM will facilitate the development of NCES segments and other reusable software.

The team developed a method to achieve full DoDAF traceability while migrating to interoperable systems using the ideas and concepts of the DoD TRM and OO UML Operational and Systems Views. Many hurdles were overcome to include end-to-end traceability and the difficult migration problems to spirally evolve stovepiped systems to an interoperable common operating picture. Using industry best practice and expertise from many leading edge companies, the team solved a complex and difficult problem that continues to agitate developers throughout the DoD (i.e., ability to trace and link requirements across the Operational, Systems, and Technical Standards Views; and integrating the DoD TRM and its methodology to support interoperability and technology insertion/transition issues). The result is a seamless and systematic approach to the complex problems the DoD must face to enter the net-centric environment in the future.

2.5.6 References

C4ISR Architecture Working Group, *C4ISR Architecture Framework*, Version 2.0, Available: http://www.defenselink.mil/nii/org/cio/i3/AWG_Digital_Library/pdfdocs/fw.pdf, December 18, 1997.

Defense Information Systems Agency, *DoD Technical Reference Model*, Version 2.0, Available: http://www-trm.itsi.disa.mil/document.htm, April 9, 2001.

Division E of Public Law 104-106, *Information Technology Management Reform Act of 1996*, August 8, 1996.

IBM, *Rational Unified Process*, Rational Software, Available: http://www.rational.com/products/rup/index.jsp

IBM, *Rational Unified Process for Systems Engineering 1.0*, Rational Software White Paper, Available: http://www.rational.com/products/whitepapers/wprupsedeployment.jsp, 2001.

NORAD/USSPACECOM, *Migrating Stovepipe Systems to Integrated/Interoperable Platforms Using the Technical Reference Model and Object-Oriented Operational Architectures*, January 2003.

2.6 SECURITY/INFORMATION ASSURANCE ARCHITECTURE

2.6.1 Introduction

A security and information assurance architecture consists of those attributes of the architecture that deal with the protection or safeguarding of operational assets, including information assets. Since security is an emergent property, the security architecture cannot be addressed independently of the rest of the architecture but must be fully integrated with it. The security architecture is used to support security analysis (i.e., the evaluation of the overall security of the enterprise and its constituent systems and the degree to which the implemented security procedures meet the operational needs for secure operational assets and secure systems).

Policy and law are forces that guide the development of systems and the level of security expected of these systems. Security engineers are responsible for ensuring that a particular area of concern includes measures that ensure compliance with the security policy guidance. An area of concern is simply the area that needs to be protected. This area could be a country or a software package. Certification and Accreditation (C&A) of systems is used in government to ensure that security policy is properly implemented, so that systems in a certain area of concern can be deemed secure. The C&A of systems is also a primary concern for security engineers.

Systems engineers are expected to comply with the security policy when creating security safeguards during the development of a system. Security policy is usually incorporated in the design based on a subjective interpretation of the policy by the system engineers, subjective interpretation of the implementation by security engineers, and subjective accreditation criteria. The C&A process usually involves a negotiation phase, where systems engineers and security engineers debate and discuss the ramifications of design decisions and the costs to implement security that complies with policy. This results in certain parts of the policy being ignored or omitted, or cost overruns of projects.

Security analysis needs to be performed throughout all phases of the systems engineering process. That would lead to security requirements that compliment the systems requirements and security that compliments the Systems View and the systems architecture implementation.

This paper provides an overview of how security goals can be identified and how a risk assessment may be conducted for an area of concern. Risk assessments can be used in conjunction with an architecture effort to provide a clear understanding of security goals. This paper also discusses which products in the DoD Architecture Framework (DoDAF) can be used to document security goals and how those goals will be achieved. Security goals are correlated to the Framework product guidance and the products' data elements. Security goals must be correlated, and corresponding security attributes must be captured in architecture products in order to specify or document security aspects of an architecture.

2.6.2 Risk Assessment Overview

The importance of a risk assessment should not be underestimated. The results of a good assessment will ensure that the security measures that are in place are actually performing the protection function for which they were designed. Its efforts are channeled into solving the right security problem.

Security and information assurance are necessary because of an asset's relative importance to the area of concern. A necessary first step is a definition of the key security goals required to protect the important assets. These fundamental security goals provide the foundation to which all security services can be traced. Setting well-defined security goals is crucial to understanding procedures needed to address security issues at all levels.

Table 2.6-1 lists the goals for security and information assurance.

Table 2.6-1. Security Goals

Goal	Definition
Confidentiality	Ensures the inadvertent/unauthorized disclosure of information; privacy is a related concept that concerns the confidentiality of personnel information.
Integrity	Ensures the inadvertent/unauthorized mo dification of an asset.
Availability	Ensures that a system is operational, functional, and accessible at a given moment. Loss of availability is sometimes referred to as denial of service.
Accountability	Ensures that responsibility for actions/events can be attributed to an actor willingly or by obligation.

In order to set goals that achieve security for assets, the Levels-of-Concern[1] for these operational assets and information systems, and the information they handle, needs to be determined. The Level-of-Concern is determined by assessing the damage that would be caused to the enterprise and the probability that a scenario leading to that damage would occur. **Table 2.6-2** contains definitions of the terms used in this section.

Table 2.6-2. Asset Assessment

Asset Assessment	Definition
Scenario	Describes a series of steps or events that occur to produce a damage effect.
Levels-of-Concern	States the amount of resources that the decision maker is willing to have allocated to prevent a scenario from happening.
Damage Effect	States the expected amount of damage to the ***area of concern*** resulting from a scenario.
Probability of Occurrence	States the chance of a scenario occurring.

In order to determine the Levels-of-Concern, the effect of damage to the enterprise by an asset compromise needs to be assessed. **Table 2.6-3** lists the categories of damage effects.

[1] Levels-of-Concern is a term used in the Director of Intelligence Directive 6/3 to rate an information system "based on the sensitivity of the information that the IS maintains, processes, and transmits."

Table 2.6-3. Damage Effects[2]

Damage Effect	Explanation
Catastrophic	Death, financial ruin, loss of critical information, operations/system destruction, widespread environmental destruction, failure to determine responsible party for catastrophic effects or modification of an asset that results in catastrophic effects, disclosure of information leading to a catastrophic effect.
Major	Moderate to severe injury/illness, moderate to great financial loss, loss of important to proprietary information, operations/system disruption for an hour or more, moderate to great environmental destruction, failure to determine responsible party for major effects or modification of an asset that results in major effects, disclosure of information leading to a major effect.
Minor	Moderate injury/illness, moderate financial loss, loss of any non-major information, operations/system disruption that lasts for under an hour, failure to determine responsible party for minor effects, or modification of an asset that results in minor effects, disclosure of information leading to a minor effect.
Nominal	Light illness, light financial loss, insignificant operations/system disruption.

The probability that an act that causes damage will occur is the next item to be assessed. **Table 2.6-4** gives categories of probabilities.

Table 2.6-4. Probability of Occurrence[3]

Probability	Explanation
Frequent	Possibility of repeated incidents within the short term[4]
Likely	Possibility of isolated incidents within the short term
Occasional	Possibility of repeated incidents within the long term[5]
Remote	Possibility of isolated incidents within the long term
Improbable	Practically impossible

The Level-of-Concern is a product of damage effect and probability of occurrence. **Table 2.6-5** lists the Levels-of-Concern.

Table 2.6-5. Levels-of-Concern (Damage Effect * Probability of Occurrence)

Levels-of-Concern	Explanation
High	Concern enough to allocate a substantial amount of resources to avert (e.g., X > 70% of allocated resources)
Medium	Concern enough to allocate a moderate amount of resources to avert (e.g., 20% < X < 70% of allocated resources)
Basic	Concern enough to allocate a minimal amount of resources to avert (e.g., X <= 20% of allocated resources)

[2] The GAO report Information Security Risk Assessment Practices of Leading Organizations organized these damage effects into categories and refer to them as Severity Levels. The categories were given names in this paper for clarity.

[3] Probability of occurrence can be found in the GAO Report Information Security Risk Assessment Practices of Leading Organizations. The list of probabilities was expanded upon and some of the names changed for clarity.

[4] Economists define short term as the period of time within a year. This, however, may be too long when dealing with information systems.

[5] Economists define long term as the period of time over a year.

The Levels-of-Concern should give a decision maker insight into the amount of resources that they are willing to allocate to minimize the damage effect.

The terms defined in the tables of this section all contribute to defining an organization's risk. The goal of a risk assessment is to identify the important assets to the enterprise and then to ensure that the functionality designed to protect the asset achieves the relative confidentiality, integrity, availability, and accountability of the system. The enterprise decision makers *must* define what the relative probabilities, damage effects, and Levels-of-Concern are. This will allow the systems engineers responsible for creating a security strategy to identify sound functionality that contributes to the protection of the enterprise.

After a risk assessment is completed, a security architecture can be developed (as part of an architecture effort) to document required security attributes (that meet the level of risk deemed acceptable) or to determine whether existing systems (and their architectures) meet the acceptable risk level.

The next section of this paper outlines the Framework products that can be used to document risk levels and specify and achieve security goals. The security attributes are mapped to the Framework products and to the architecture data elements.

2.6.3 Security/Information Assurance and the Framework

Security and information assurance concerns apply to most if not all of the Framework products. The challenge is determining the functional strategy that fulfills the security goal. The creation of a "security view" should happen when the Operational View (OV) products are being put together. From a security perspective, the All-Views (AV) products should specify the operational goals, strategies, and critical success factors that involve or are related to security. The OV products should specify the security goals that are most important to the enterprise, the types of assets that need protecting, and a rating of the assets' importance to the enterprise. The Systems View (SV) products should specify the security systems and the functionality that helps accomplish the operational security goal(s). The Technical Standards View (TV) products should outline the standards that are necessary to make systems acceptable with respect to operational security goals.

2.6.3.1 Security Attributes for All-Views Products

The security and information assurance policy and goals should contain a summary of the highest risk issues to the enterprise. For example, the types of data to be protected, such as Classified, or Sensitive But Unclassified (SBU), and expected information about the threat environment, other threats and environmental conditions, and geographical areas addressed by the architecture. This information is identified in the Overview and Summary Information (AV-1).

The operational goals, strategies, and critical success factors that involve or are related to security (i.e., the operational drivers for security) should be identified in AV-1. In addition, Business Unit Risk Assessments[6] and business plans, including budget constraints, are also key motivations for selecting the priorities for providing resources for security. These drivers for

[6] U.S. General Accounting Office, Accounting and Information Management Division, *Information Security Management: Learning from Leading Organizations*, May 1998.

security should include the relevant laws and regulations and the enterprise security policy of the encompassing (or parent) architecture. Drivers also include any Memoranda of Understanding with external (to the architecture) organizations regarding shared assets.

The Integrated Dictionary (AV-2) should contain all security and information assurance relevant architecture data elements and data definitions.

2.6.3.2 Security Attributes for Operational View Products

Operational Environment. Any product that depicts attributes of the operational environment should give a representation of the importance of asset protection. The security and information assurance attributes for the product must identify the elements that are security relevant. Relevance is always determined by the Levels-of-Concern for the type of asset under consideration and the security goals set for the protection of that asset. The key step is to identify the security implications of the operational environment. Usually, policy and organizational structures dictate who is in charge. What is not stated is what makes them accountable for their actions. In order to achieve the security goal of accountability, what is needed to make the key players in operational nodes responsible for their actions should be documented. This documentation does not always have to have negative consequences for unwanted behavior. Positive statements with regards to desired behavior are preferable. The documentation must specify what consequences would occur at the nodes if security procedures were not implemented. In addition, every information exchange that is detailed in the Operational View must have accompanying details on the security activities that are to be instituted to protect the information exchange. All information exchanges that are indicated in the Operational View must have information with regards to the security service that is to offer protection. Also, each operational activity has a security consequence that must be assessed.

As an example, **Figure 2.6-1** depicts operational nodes and activities performed at the nodes. The needlines represent information exchanges that must occur between the nodes in order for the activities to be realized. From a security perspective, each information exchange must be specified with attributes indicating the security goals. In the example, an information exchange (represented here with a needline) between Nodes B and C has attributes of confidentiality and integrity. Confidentiality and integrity are security goals that make an information exchange security relevant. Each information exchange that moves across that needline needs to be expanded in the security documentation. The security attributes of activities (performed at the node, and documented in the Operational Activity Model) necessary to achieve the security goal should be in the documentation as well.

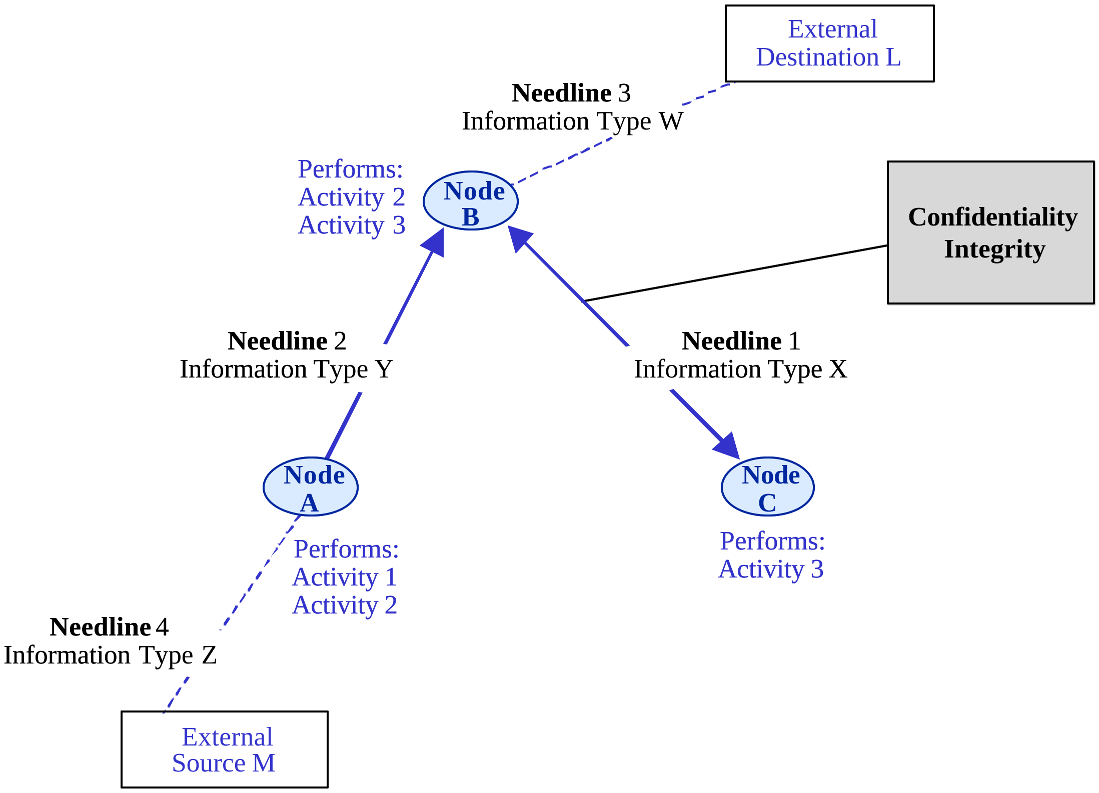


Figure 2.6-1. Node Connectivity Example Showing Security Attributes for a Needline

The rest of this subsection contains a more detailed discussion of the security attributes of some OV data elements.

Information Assets. The operational assets that need protection should be identified, as well as the types of security goals (i.e., protections) that they need. For information assets, these security goals include protection from inadvertent or unauthorized disclosure (confidentiality), inadvertent or unauthorized modification (integrity), and denial of authorized access (availability). These attributes of the security architecture should be documented in the Operational Information Exchange Matrix (OV-3) as security attributes of an information exchange, and in the Logical Data Model (OV-7) as security attributes of any of the allowable entity types.

Operational assets with security goals should be associated with indicators that provide a measure of how critical these operational assets are and what priority they have in terms of allocating resources to ensure their security. For example, Levels-of-Concern could be indicated through a rating system of high, medium, or low. In addition, a generic security approach should be indicated for each security goal of an operational asset. (For more information on security approaches, see “A Practical Approach to Integrating Information Security into Federal Enterprise Architectures” by John DiDuro.) For information assets, integrity corruption can be detected and corrected; so generic security approaches such as prevention, detection, and response are usually selected, with the relative emphasis on these approaches dependent on the specific enterprise. For some types of operational assets with the integrity property, corruption of the asset may mean that it is worthless and must be discarded. For this type of asset, the security approach indicated should focus on prevention. These attributes of the security architecture should be documented in OV-7.

Operational Activities. The operational activities that access protected operational assets should be identified. Access (i.e., security relevant operational activities) can be in the form of read, write/modify, create, or delete activities. The operational assets that may be accessed consist of Operational Activity Model (OV-5) inputs/outputs, OV-3 information elements, and/or OV-7 entity types. Operational activities that require such access to protected operational assets should be documented in OV-5.

In addition, the flow across organizational areas of authority (or protection responsibility) of protected operational assets should be identified. This attribute of the security architecture should be documented in OV-5 (i.e., the organizations involved in each operational activity (possibly multiple organizations per activity) of OV-5 should be identified).

Operational Locations. The operational locations where protected operational assets are stored, or where operational activities that access protected operational assets are performed, should be identified. This attribute of the security architecture should be documented in the Operational Node Connectivity Description (OV-2).

Organizations. The organizations responsible for protected operational assets should be identified as well as organizations that have security management responsibilities (e.g., security planning, security operations, and system certification and accreditation). This attribute of the security architecture should be documented in the Organizational Relationships Chart (OV-4).

Policy, Security Goals, and Security Rules. Operational security rules derived from security policies should be documented in the Operational Rules Model (OV-6a).

Operational Events. The major operational events (e.g., triggers for information exchanges and events that affect threads of operational activities) that involve protected operational assets should be identified. This attribute of the security architecture should be documented in OV-3, the Operational State Transition Description (OV-6b), and/or the Operational Event-Trace Description (OV-6c). Operational events that involve protected operational assets (e.g., trigger a security related operational activity to occur or result in an information exchange that needs to be protected) should be flagged and documented, and their relevant security goals should be documented. For example, a new enlistment results in the creation of protected information (i.e., Sensitive But Unclassified [SBU]). The new enlistment is an operational event that is security relevant.

2.6.3.3 Security Attributes of Systems View Products

The security elements for Systems View products include:

- The allocation of security requirements (from the Operational View) to physical, procedural and automated systems in a Systems Interface Description (SV-1) and a Systems Communications Description (SV-2)
- The allocation of certification and accreditation requirements for each system
- The allocation of authority for each system

The security documentation created in the Systems View should state the system functions, systems, or subsystems that will be protecting a system interface (or the corresponding system data exchanges). The documentation should also indicate the security goals that were identified at the operational level for the type of information/assets traversing that interface. The system performing the security function must also have criteria for satisfying the identified security goals.

For example, the blue line (which represents system data exchanges) in **Figure 2.6-2** is labeled virtual private network (VPN) B. Each VPN implements a given set of security functionality for a system data exchange. From a security perspective, a VPN is a proven way to protect system data exchanges across the Internet. However, to complete the security analysis, the architect or the security engineer must be able to document that the use of VPN satisfies the security goals of the enterprise.

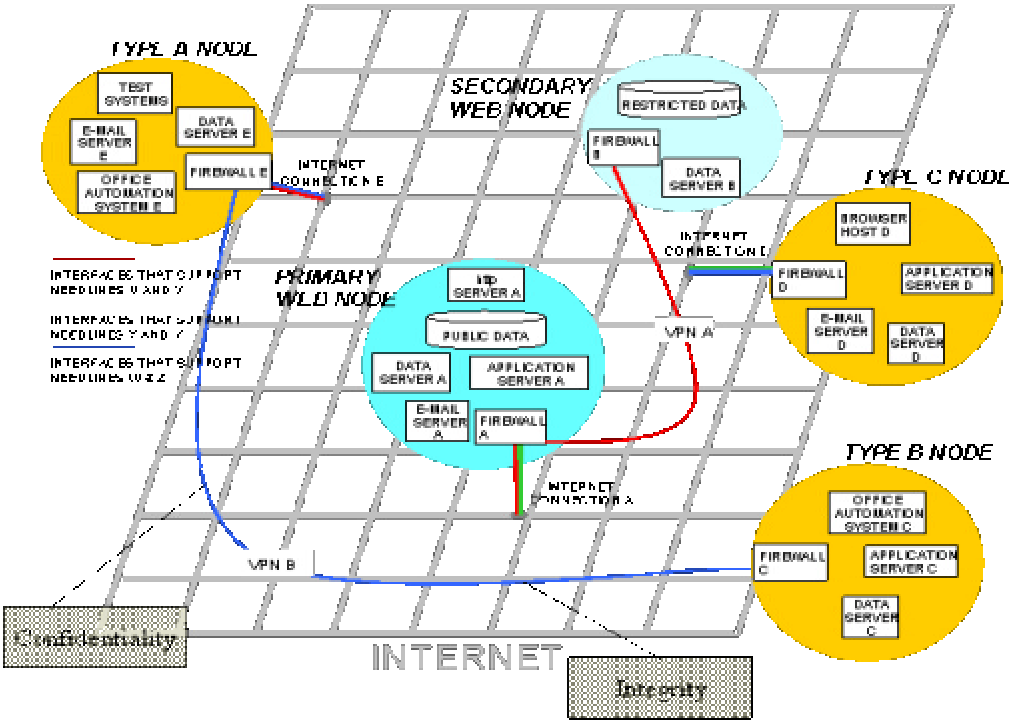


Figure 2.6-2. Systems Connectivity Example Showing Security Attributes for a Physical Link

Figure 2.6-2 indicates that VPN B is used for system data exchanges between Firewall C and Firewall B. There are two boxes that label the security goals that the enterprise wants achieved.

Confidentiality: The VPN gives a level of assurance that the data traversing VPN B will not be easily interpreted by a hostile entity.

Integrity: The firewalls on either end of the interface ensure that data traversing the interface will not be allowed into type A or type B node unless it follows a particular protocol. That, in itself, gives a level of assurance about the integrity of the data. However, if the enterprise gave an integrity goal because of the types of decisions that are going to be made based on the data; this interface does not offer any protection as far as ensuring that the data traversing that interface has not been modified. There is no guarantee that the information sent from Node A is the information received by Node B. The firewall or some other system must provide the functionality that realizes that security goal.

Security systems should be treated as systems in the Systems View. Any system documentation should include security systems. Traceability matrixes (such as SV-3 and SV-5) that have security goals listed as attributes should list the system/subsystem or automated process responsible for achieving that security goal.

Relationships between information entities/assets, security services, and level of protection can be captured in a Systems Data Exchange Matrix (SV-6). Security services (either as system functions or characteristics of communications links), structural organization of systems and system functions, allocation of information assets to systems, and references to certification and accreditation requirements and authorities can be captured in SV-1, SV-2, and a Systems Functionality Description (SV-4).

2.6.3.4 Security Attributes of Technical Standards View Products

The Technical Standards Profile (TV-1) should identify the security standards as it would for all other systems standards. Security relevant standards (e.g., encryption algorithms, secure protocols, and cyclic redundancy check algorithms) should be documented.

Security elements include the types of mechanisms (e.g., token based I&A or PKI) to be used for the security services and the security standards that should be applied. The choice of security standards in TV-1 may be constrained by the Joint Technical Architecture. The specific security standards used in the architecture can be captured in the corresponding architecture data elements of the SV products (e.g., encryption standards may be listed in SV-6). In addition to specifying a security standard(s), a measure (e.g., high, medium, or low) of the level of protection to be provided by the security services may be specified. The overall organization of these security services and the required security management services within the overall enterprise system model (both structural and dynamic attributes) may also be indicated in TV-1.

2.6.4 Summary and Conclusion

The security of operational assets is crucial to ensure mission success. A security architecture, consisting of security attributes that are mapped to the DoDAF products and the products' data elements, is an integrated method of ensuring that security policies are set, and that security procedures and standards are implemented throughout an architecture.

This paper provided an overview of how security goals can be identified and how a risk assessment may be conducted for an area of concern. The paper also outlined how security attributes may be incorporated into some of the products of the DoDAF.

2.6.5 References

DiDuro, John, et al., *A Practical Approach to Integrating Information Security into Federal Enterprise Architectures*, McLean: Logistics Management Institute, October 2002.

Director of Central Intelligence Directive, *Protecting Sensitive Compartmented Information Within Information Systems (DCID 6/3)-Manual*, Available: http://www.fas.org/irp/offdocs/DCID_6-3_20Manual.html, June 5, 1999.

Internet Security Alliance, *Common Sense Guide for Senior Managers: Top Ten Recommended Information Security Practices*, Available: http://www.isalliance.org/news/BestPractices.pdf, July 2002.

Joint Staff Directorate for C4 Systems (J6), *Information Assurance: Legal, Regulatory, Policy and Organizational Considerations*, 4th Edition, Available: http://www.dtic.mil/jcs/j6/j6k/ia.pdf, August 1999.

National Association of State Information Resource Executives, *Public-Sector Information Security: A Call to Action for Public-Sector CIOs*, Available: http://www.nascio.org/2001/11/securityforum011113-14.cfm, July 2002.

U.S. General Accounting Office, Accounting and Information Management Division, *Information Security Risk Assessment Practices of Leading Organizations*, Available: http://www.gao.gov/special.pubs/ai00033.pdf, November 1999.

U.S. General Accounting Office, Accounting and Information Management Division, *Information Security Management: Learning from Leading Organizations*, Available: http://www.gao.gov/special.pubs/ai9868.pdf, May 1998.

U.S. Office of Management and Budget, *Guidance for Preparing and Submitting Security Plans of Action and Milestones*, Memorandum for the Heads of Executive Departments and Agencies, Available: http://www.whitehouse.gov/omb/memoranda/m02-01.html, October 17, 2001.

Workstation Support Services, *Computer Security Framework and Principles*, Available: http://wssg.berkeley.edu/SecurityInfrastructure/reports/framework.html, March 25, 1998.

2.7 AN ARCHITECTURE PERSPECTIVE ON NCOW

> **Net-Centric Warfare** "Does not focus on network-centric computing and communications, but rather on information flows, the nature and characteristics of battlespace entities, and how they need to interact."
> Network Centric Warfare, 2nd Edition, Alberts, et al

2.7.1 Introduction

The Office of the Assistant Secretary of Defense for Networks and Information Integration (OASD[NII])/Deputy Chief Information Officer (DCIO), Directorate of Architectures and Integration is developing the Global Information Grid (GIG)[1] Architecture Version 2 as an objective architecture based in the Net-Centric Operations and Warfare (NCOW) environment. A primary objective of this effort is to gain an understanding of the information technology (IT) requirements for supporting the NCOW warfighting concepts.

GIG Architecture v2 is set in 20XX, a non-specified future year, far enough in the future to provide for the implementation of known technologies assumed to be contributing to net-centricity. The architecture is Framework-compliant with regard to the architecture data elements underpinning the products. The visualization of the products is intended to emphasize NCOW.

The DoD community is continuing to evolve NCOW tenets, services, and architecture visualization approaches. There is not yet a community position on many aspects regarding NCOW. Emerging concepts are being staffed and coordinated across DoD. The material in this section presents some of those concepts. The material should not be considered to reflect a coordinated community position.

This section discusses basic tenets of NCOW, describes the Provide Net-Centric Environment activity, discusses NCOW information exchanges, and provides examples of a High-Level Operational Concept Graphic (OV-1), an Operational Node Connectivity Description (OV-2), a Systems Interface Description (SV-1), and a Systems Communications Description (SV-2) depicting the NCOW environment. The architecture concepts and example products are drawn from GIG Architecture v2.

2.7.2 NCOW Tenets

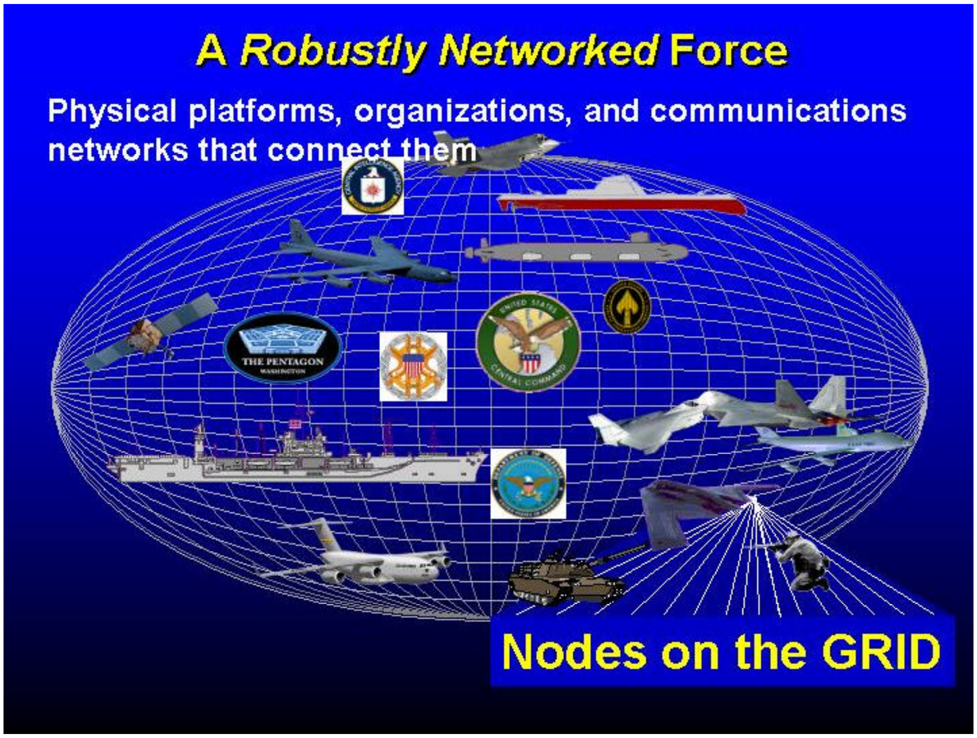
Basic tenets of NCOW are a robustly networked force that improves information sharing and uses information to gain shared battlespace awareness. Shared battlespace awareness is accomplished through virtual integration and collaboration. The sum of these enable shared understanding that generates increased combat power, increased speed of command, higher operating tempos, and increased survivability.[2]

[1] The GIG is "the globally interconnected end-to-end set of information capabilities, associated processes, and personnel for collecting, processing, storing, disseminating, and managing information on demand to warfighters, policy makers, and support personnel." As such, the GIG contains all information technology and communications assets of the DoD, including both warfighting and business assets.

[2] *Report on Network Centric Warfare Sense of the Report*, submitted to the Congress in partial fulfillment of Section 934 of the Defense Authorization Act for FY01 (Public Law 106-398), March 2001, Arthur L. Money, Assistant Secretary of Defense (C3I).

NCOW introduces new concepts of information exchange and node connectivity. Information is published to the grid, and users obtain information from the grid. ASD(NII) has envisioned the principals of information exchange within NCOW as Task, Post, Process, and Use (TPPU). Raw data and information at various levels of processing and analysis are posted to the grid. Users have immediate access to information as posted data becomes available. Users subscribe to information either via specific requests, standing queries, or as a result of combined effects of the user's search profile and intelligent agents. Users also conduct ad hoc searches to receive information. Users virtually integrate and collaborate using communities of interest (COI), where multiple people interact synchronously and asynchronously. Information is technically accessible to all users with availability limited only by policy.

As depicted in **Figure 2.7-1**, users, organizations, OpNodes, platforms, and facilities are all nodes on the network grid. The primary connectivity within NCOW is between the nodes and the grid. Sets of services, referred to as Net-Centric Enterprise Services (NCES), are provided via the grid and are available to all users. NCES provides information access, manipulation, and transmittal capabilities to all users. These services can be considered to be provided via a virtual node. For visualization purposes, this virtual node is referred to as the Net-Centric Information Domain (NCID) in the example OV products presented in this document.


A Robustly Networked Force

Physical platforms, organizations, and communications networks that connect them

Figure 2.7-1. NCOW Nodes on the Grid

Figure 2.7-2 provides a comparison of the information exchanges of a Situation Report in the As-Is environment and in the objective NCOW environment.

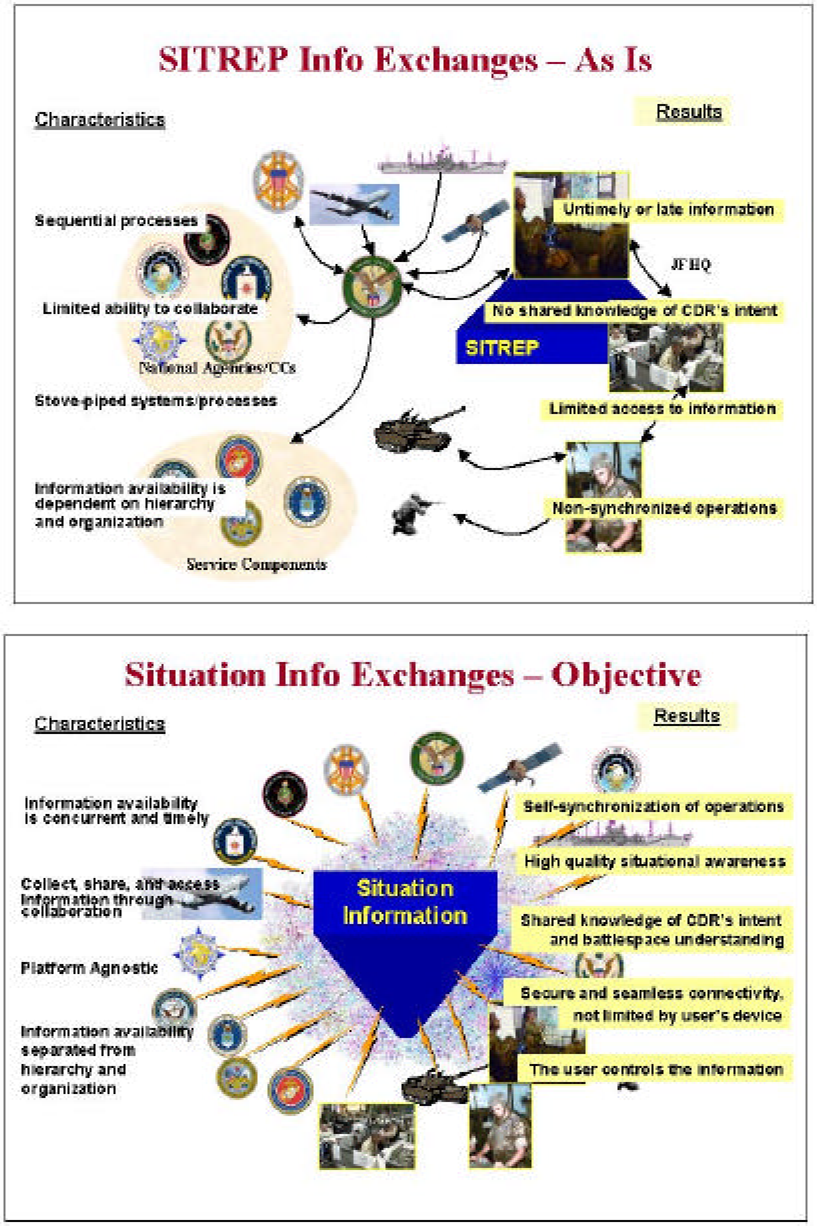

SITREP Info Exchanges – As Is
Characteristics
Results
Sequential processes
Limited ability to collaborate
National Agencies/CCs
Stove-piped systems/processes
Information availability is dependent on hierarchy and organization
Service Components
Untimely or late information
JFHQ
No shared knowledge of CDR's intent
SITREP
Limited access to information
Non-synchronized operations
Situation Info Exchanges – Objective
Characteristics
Results
Information availability is concurrent and timely
Collect, share, and access information through collaboration
Platform Agnostic
Information availability separated from hierarchy and organization
Situation Information
Self-synchronization of operations
High quality situational awareness
Shared knowledge of CDR's intent and battlespace understanding
Secure and seamless connectivity, not limited by user's device
The user controls the information

Figure 2.7-2. Example As-Is and To-Be Information Exchanges

2.7.3 Net-Centric Enterprise Services

NCOW is a service-enabled architecture. DISA is lead integrator for the services and will be responsible for providing some but all not of the services. Net-Centric Enterprise Services (NCES) replaces the concept of the Defense Information Infrastructure Common Operating Environment (DII COE) in the future NCOW context. NCES includes nine core enterprise services:

- **Enterprise Service Management** – Provides end-to-end GIG performance monitoring, configuration management, and problem detection/resolution as well as enterprise IT resource accounting and addressing (e.g., for users, systems, devices)
- **Messaging Services** – Ability to exchange information among users or applications on the enterprise infrastructure (e.g., E-mail, Defense Message Service, Variable Message Format, U.S. Message Text Format, Tactical Digital Information Link, Message Oriented Middleware, America-On-Line instant messenger, Wireless Services, and Alert Services)
- **Discovery Services –** Processes for discovery of information content or services that exploit metadata descriptions of network resources stored in Directories, Registries, and Catalogs (includes search engines)
- **Mediation Services –** Services that help broker, translate, aggregate, fuse, or integrate data
- **Collaboration Services –** Allows users to work together and jointly use selected capabilities on the network (i.e., chat, online meetings, work group software, etc.)
- **User Assistant Services –** Automated "helper" capabilities that reduce effort required to perform manpower-intensive tasks
- **Application Services –** Infrastructure to host and organize distributed online processing capabilities
- **Security Services** – Capabilities that address vulnerabilities in networks, services, capabilities, or systems
- **Storage Services** – Physical and virtual places to host data on the network with varying degrees of persistence (e.g., archiving, Continuity of Operations, and content staging)

2.7.4 NCOW Reference Model

As this Deskbook is finalized, DoD is developing a NCOW Reference Model (RM). The goal of the NCOW RM is to provide a common lexicon for NCOW concepts and terminology, supported by recognizable architectural descriptions. While the RM is not yet completely defined, much consideration has been given to the activity model depicting net-centric information enterprise activities. These activities are not specific to a warfighting or business mission area, but instead apply across all mission and business areas. The intent is that this activity model defines those activities associated with operating and using the information grid within the NCOW concept.

Table 2.7-1 presents a high-level version of the activity model as defined in NCOW RM Draft Version 0.9, July 2, 2003. The overall activity is *Provide Net-Centric Information Environment*. The activity model will continue to evolve and mature as greater understanding is reached. The model is being coordinated within the DoD architecture community with special emphasis on DISA, the DoD program manager for NCES, and with the OASD(NII) manager for Horizontal Fusion.

Table 2.7-1. A0 Provide Net-Centric Information Environment
(Draft Version 0.9 July 2003)

A1 Interact with Net-Centric Enterprise Services

User-performed activities to access, post, modify, and use information in the conduct of their missions and tasks.

A11 Request Access to the Information Environment

A111 Login to Information Environment
A112 Request New User Account

A12 Request Services/Functional Capabilities

A13 Create/Maintain User/Entity Profile

A14 Provide Information/Objects to the Information Environment

A141 Develop Information/Objects
A142 Post Information/Objects
A143 Import Information/Objects

A15 Get Information/Objects

A151 Subscribe to Information
A152 Discover Information

A16 Request Collaboration Services

A161 Identify Collaboration Requirements
A162 Identify Participants
A163 Request Collaboration
A165 Conduct Collaborations

A2 Perform Net-Centric User/Entity Services

Provides a semi-autonomous agent and advocate capability between the user/entity and the services provided within the information environment.

A21 Evaluate/Ingest Inputs

A22 Assist User/Entity

A221 Personalize User Environment
A222 Learn from User Interactions
A223 Add Value to User Interactions

A23 Invoke Net-Centric Capabilities/Services

A3 Provide Net-Centric Enterprise Services

Provides the Net-Centric Enterprise Services that enable users to dynamically interact, share, and use information in a net-centric environment.

A31 Provide Core Services

A311 Perform Discovery Services
A312 Provide Collaborations Services
A313 Provide Messaging Services
A314 Perform Information Mediation Services
A315 Perform Information Storage Services
A316 Provide Core Applications/Functions

A32 Provide COI Services

A33 Perform Environment Control Services

A4 Resource Service Requests

Takes an infrastructure resource service request and provides the resources (e.g., processors, memory, storage media, bandwidth, etc.) necessary to satisfy that request.

A41 Provide Computing Resources

A42 Provide Communications Resources

A43 Provide Media Resources

A5 Manage Net-Centric Information Environment

Planning, organizing, coordinating, and controlling the establishment, maintenance, and dissolution of all the capabilities of and services provided by the information environment.

A51 Develop Information Environment Capabilities

A52 Manage System & Network Configurations

A53 Manage Core Enterprise Services

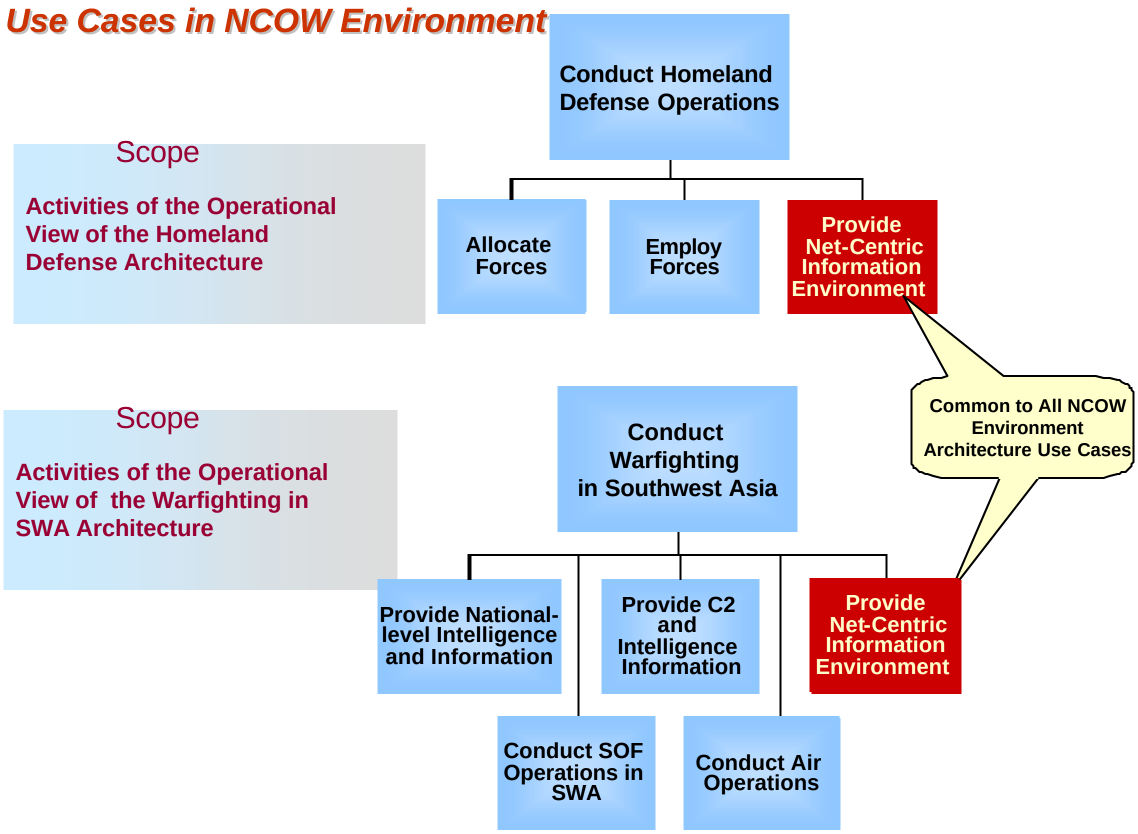
A54 Manage Accounts

A55 Manage Cryptographic Services Infrastructure

A56 Manage Monitoring Activity

A57 Manage Response Activity

Because *Provide Net-Centric Enterprise Environment* applies across all mission-oriented activities, it is intended to be included as an activity set within any objective architecture based in NCOW. **Figure 2.7-3** illustrates this concept.



Use Cases in NCOW Environment

Scope

Activities of the Operational View of the Homeland Defense Architecture

Conduct Homeland Defense Operations

Allocate Forces

Employ Forces

Provide Net-Centric Information Environment

Common to All NCOW Environment Architecture Use Cases

Scope

Activities of the Operational View of the Warfighting in SWA Architecture

Conduct Warfighting in Southwest Asia

Provide National-level Intelligence and Information

Provide C2 and Intelligence Information

Provide Net-Centric Information Environment

Conduct SOF Operations in SWA

Conduct Air Operations

Figure 2.7-3. *Provide Net-Centric Information Environment* as a Common Activity in NCOW Environment Use Cases

2.7.5 NCOW Information Exchanges

In the NCOW context, all information exchanges using IT are on the grid and utilize assets of the NCID virtual node. Information is provided to the grid. Information provided to the grid is fused on the grid to create new composition information products. Information is received from the grid, either by searching the grid or subscribing to information. Subscriptions include general specifications of information via a personal profile with the aid of intelligent agents as well as specific designations or required information. Collaboration among multiple parties becomes a common approach for performing necessary activities.

The direct mapping between the information producer and information consumer does not necessarily apply in NCOW. Multiple producers provide information to the grid, where it may be fused into a variety of information products. Multiple consumers receive the information in various fused versions. The producer may have no knowledge of the consumers of his information. Similarly, the consumer may have no knowledge of the producers of the information consumed.

In Volume II, the DoD Architecture Framework (DoDAF) has included Transaction Type as an attribute in the Operational Information Exchange Matrix (OV-3). Transaction Type can be used to more accurately portray information exchanges in an Internet context. The following values apply:

- **Direct** denotes point-to-point information exchanges from a sending/producing node to a receiving/consuming node. Examples of this type of exchange include transmittal of specific information products from one OpNode to a designated second OpNode (such as sending information via e-mail or via a telephone conversation) or a consuming OpNode directly accessing information on the grid via a specified URL. Even though the information exchange is “direct,” it is made via grid assets.
- **Post** makes information available on a Web site or network.
- **Subscribe** receives delivery of predefined or newly defined information products and updates as they become available or according to a delivery schedule. Information products and updates are selected based on specific requests by the user (administrative subscription), a standing user-defined query, or as a result of the combined efforts of the user’s search profile and intelligent agents.
- **Search** receives information based on the initiation of an ad hoc query for information.
- **Collaborate** allows interaction with other people reviewing and sharing multimedia data, information, applications, and common situational perspectives, including conducting asynchronous and session-based dialogues/meetings with each other. This includes use of collaboration capabilities such as whiteboards, teleconferencing, chat, and shared/distributed applications.

When Post is used, the receiving node is a virtual node on the grid. When Subscribe or Search is used, the sending node is the virtual node. When multiple people collaborate on-line, their associated nodes become both sending and receiving nodes, but each connects to the virtual node.

Since information access is determined by policy, it may be desirable to add a new attribute to denote the access policy associated with information published to the grid.

2.7.6 Example Products Depicting NCOW

2.7.6.1 High-Level Operational Diagram

Figure 2.7-4 is for a tactical use case in the NCOW environment. The graphic emphasizes the connectivity of all participating OpNodes via the grid. As noted earlier in this section, the NCID refers to virtual node of the grid.

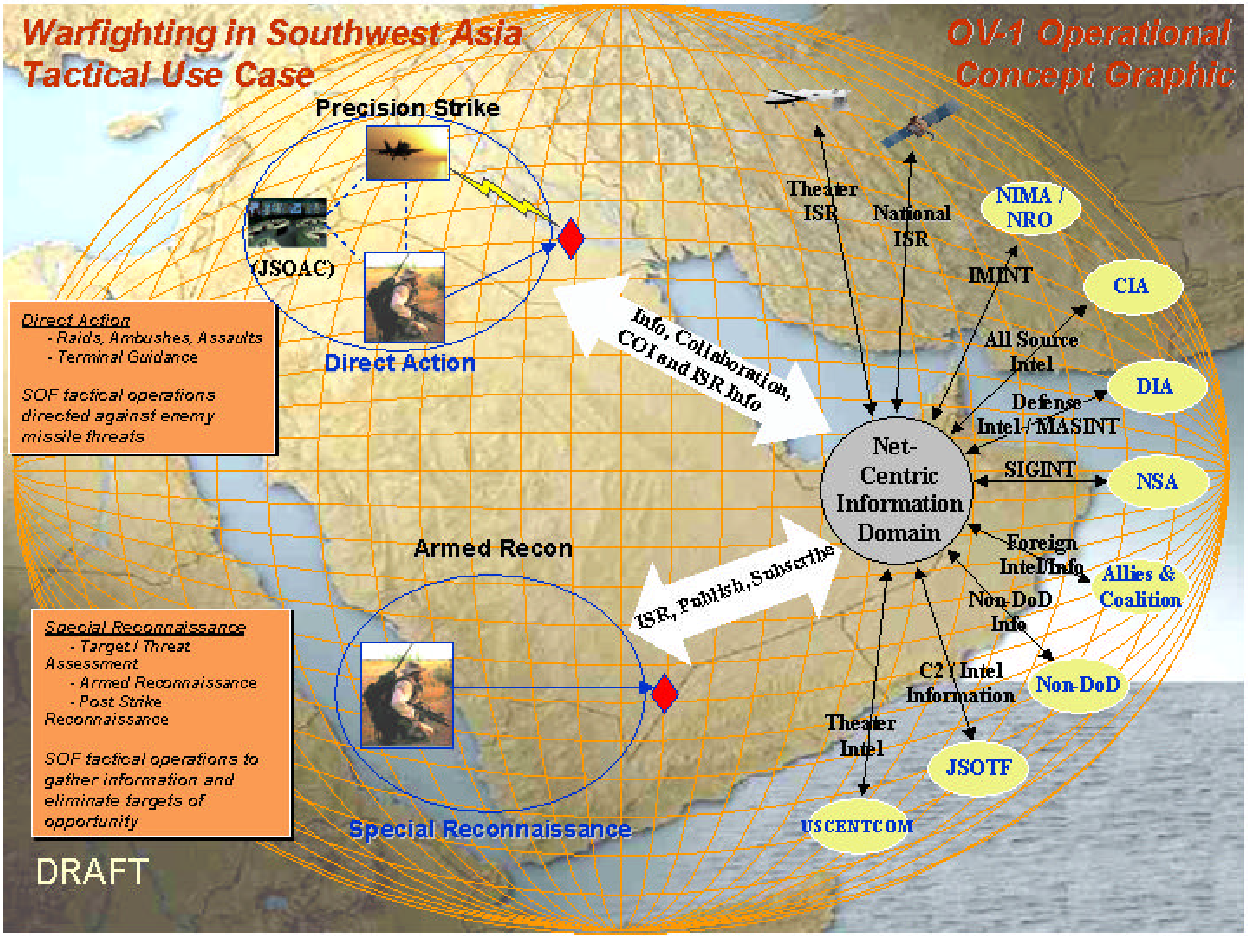


Figure 2.7-4. Example OV-1 for NCOW

2.7.6.2 Node Connectivity Description

Figure 2.7-5 portrays the multiple OpNodes exchanging information to support Special Operations Forces (SOFs) in a tactical use case. Needlines are between each OpNode and the virtual node of the Net-Centric Information Grid.

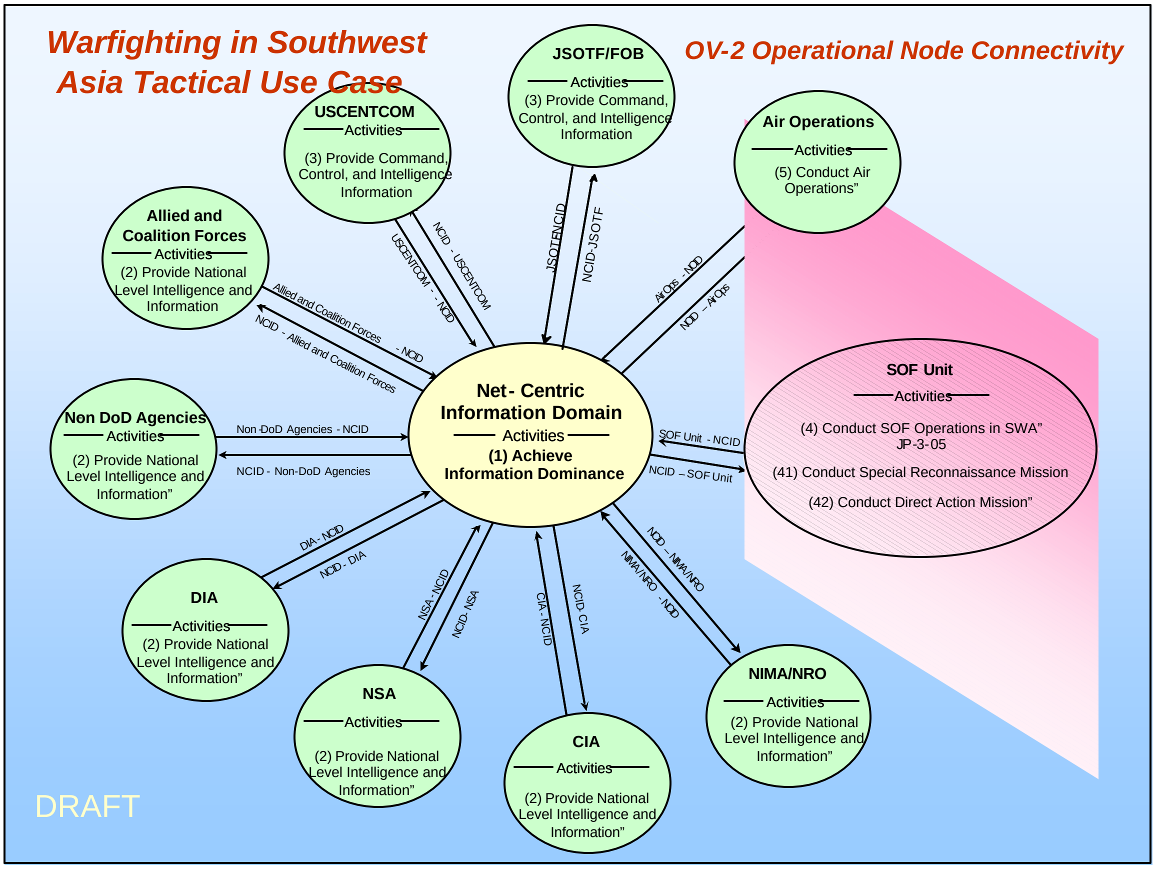

Warfighting in Southwest Asia Tactical Use Case
OV-2 Operational Node Connectivity
JSOTF/FOB
Activities
(3) Provide Command, Control, and Intelligence Information
USCENTCOM
Activities
(3) Provide Command, Control, and Intelligence Information
Air Operations
Activities
(5) Conduct Air Operations"
Allied and Coalition Forces
Activities
(2) Provide National Level Intelligence and Information
Net- Centric Information Domain
Activities
(1) Achieve Information Dominance
SOF Unit
Activities
(4) Conduct SOF Operations in SWA"
JP-3-05
(41) Conduct Special Reconnaissance Mission
(42) Conduct Direct Action Mission"
Non DoD Agencies
Activities
(2) Provide National Level Intelligence and Information"
DIA
Activities
(2) Provide National Level Intelligence and Information"
NSA
Activities
(2) Provide National Level Intelligence and Information"
CIA
Activities
(2) Provide National Level Intelligence and Information"
NIMA/NRO
Activities
(2) Provide National Level Intelligence and Information"
USCENTCOM - NCID
NCID - USCENTCOM
JSOTF-NCID
NCID-JSOTF
AirOps - NCID
NCID – AirOps
Allied and Coalition Forces - NCID
NCID - Allied and Coalition Forces
Non-DoD Agencies - NCID
NCID - Non-DoD Agencies
SOF Unit - NCID
NCID – SOF Unit
DIA - NCID
NCID - DIA
NSA - NCID
NCID- NSA
CIA - NCID
NCID- CIA
NCID – NIMA/NRO
NIMA/NRO - NCID
DRAFT

Figure 2.7-5. Example OV-2 for NCOW

2.7.6.3 Systems Interface Description

Figure 2.7-6 depicts organization nodes operating client services with connectivity to the services provided by the virtual node of the grid. NCES are operating on the grid and are composed of:

- Applications Services
- Enterprise Service Management
- Security Services
- Collaboration Services
- Mediation Services
- Storage Services
- Discovery Services
- Messaging Services
- User Assistant Services

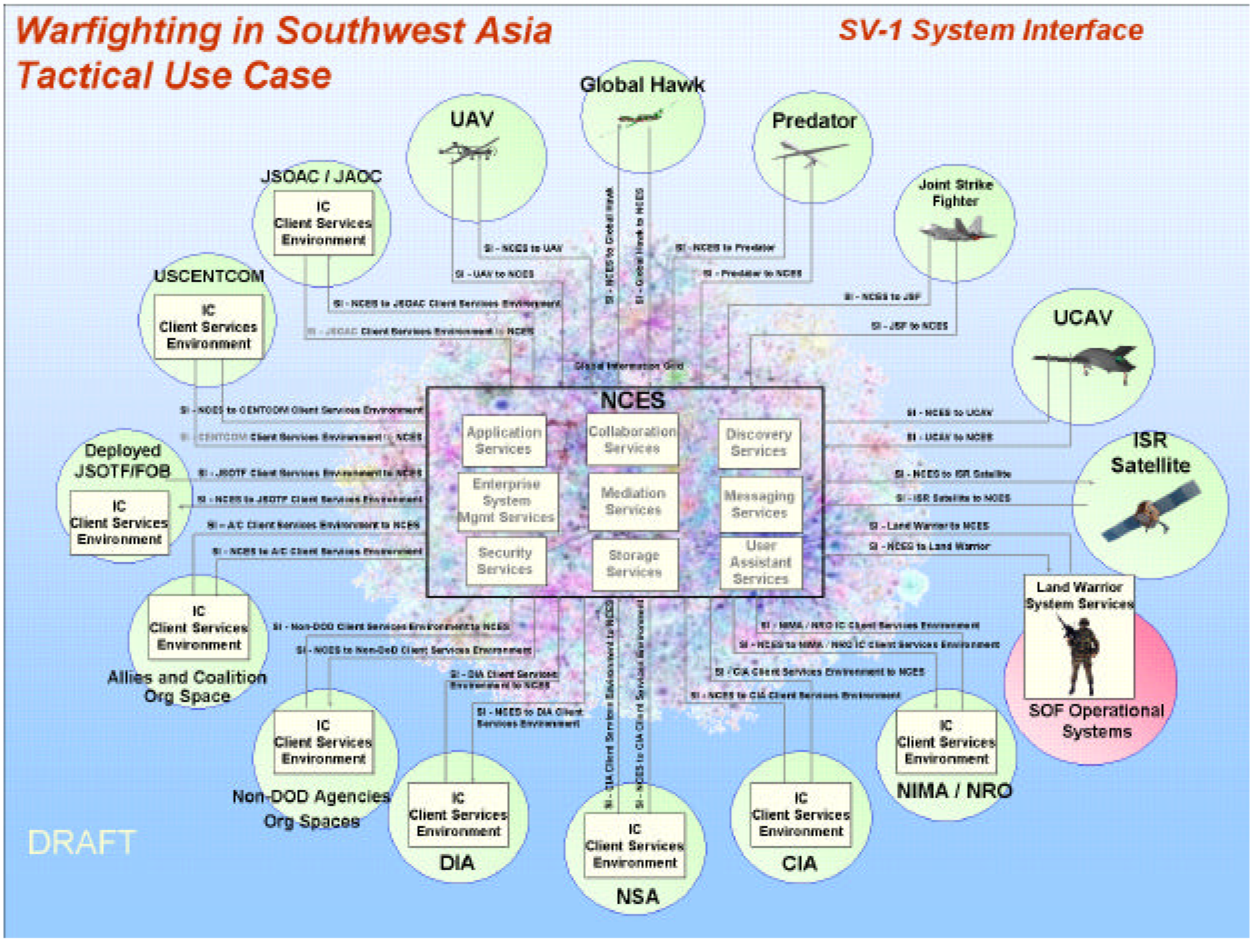

Warfighting in Southwest Asia
Tactical Use Case
SV-1 System Interface
Global Hawk
UAV
Predator
JSOAC / JAOC
Joint Strike Fighter
USCENTCOM
UCAV
IC Client Services Environment
SI - NCES to UAV
SI - UAV to NCES
SI - NCES to Global Hawk
SI - Global Hawk to NCES
SI - NCES to Predator
SI - Predator to NCES
SI - NCES to JSF
SI - JSF to NCES
SI - NCES to JSOAC Client Services Environment
SI - JSOAC Client Services Environment to NCES
Global Information Grid
NCES
Application Services
Collaboration Services
Discovery Services
Enterprise System Mgmt Services
Mediation Services
Messaging Services
Security Services
Storage Services
User Assistant Services
SI - NCES to CENTCOM Client Services Environment
SI - CENTCOM Client Services Environment to NCES
SI - NCES to UCAV
SI - UCAV to NCES
Deployed JSOTF/FOB
ISR Satellite
SI - JSOTF Client Services Environment to NCES
SI - NCES to JSOTF Client Services Environment
SI - NCES to ISR Satellite
SI - ISR Satellite to NCES
SI - A/C Client Services Environment to NCES
SI - NCES to A/C Client Services Environment
SI - Land Warrior to NCES
SI - NCES to Land Warrior
Land Warrior System Services
SOF Operational Systems
Allies and Coalition Org Space
SI - Non-DOD Client Services Environment to NCES
SI - NCES to Non-DoD Client Services Environment
SI - NIMA / NRO IC Client Services Environment
SI - NCES to NIMA / NRO IC Client Services Environment
SI - DIA Client Services Environment to NCES
SI - NCES to DIA Client Services Environment
SI - CIA Client Services Environment to NCES
SI - NCES to CIA Client Services Environment
SI - NSA Client Services Environment to NCES
SI - NCES to NSA Client Services Environment
Non-DOD Agencies Org Spaces
DIA
NSA
CIA
NIMA / NRO
DRAFT

Figure 2.7-6. Example SV-1 for NCOW

Although SV-1 is generally a representation of physical nodes, the grid has been represented in a conceptual manner for GIG v2. This approach was used because the primary intent is to gain an understanding of the conceptual aspects of NCOW before moving to specific physical implementations.

2.7.6.4 Systems Communications Description

As with SV-1, **Figure 2.7-7** depicts the communications between specific physical nodes and the virtual node of the grid. The grid connects to nodes via the Internet Protocol Network in either the terrestrial or wireless modes.

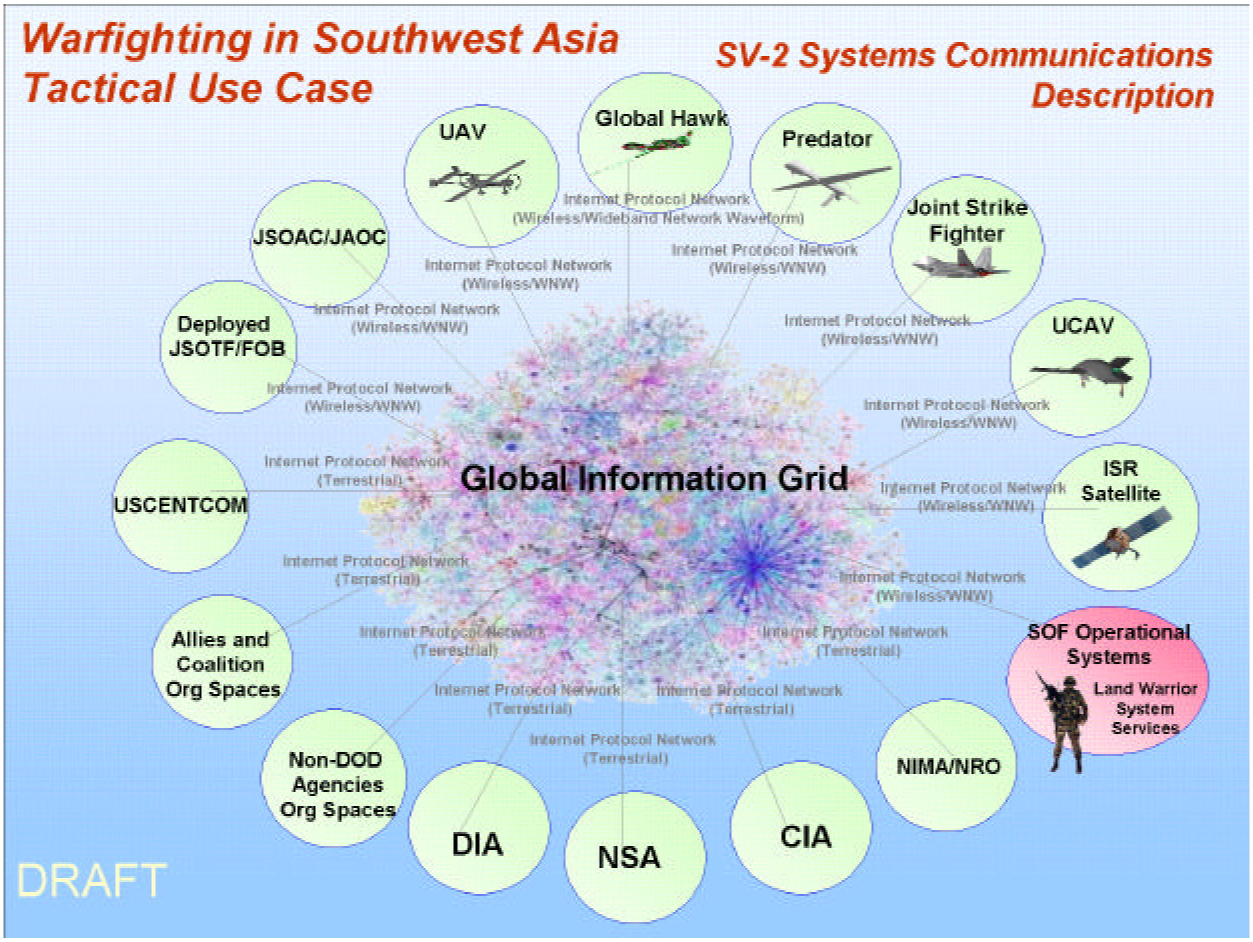

Warfighting in Southwest Asia
Tactical Use Case
SV-2 Systems Communications Description
Global Information Grid
UAV
Global Hawk
Predator
Joint Strike Fighter
UCAV
ISR Satellite
SOF Operational Systems
Land Warrior System Services
NIMA/NRO
CIA
NSA
DIA
Non-DOD Agencies Org Spaces
Allies and Coalition Org Spaces
USCENTCOM
Deployed JSOTF/FOB
JSOAC/JAOC
Internet Protocol Network (Wireless/Wideband Network Waveform)
Internet Protocol Network (Wireless/WNW)
Internet Protocol Network (Terrestrial)
DRAFT

Figure 2.7-7. Example SV-2 for NCOW

2.7.7 References

DoD CIO, Directorate of Architectures and Integration, *Global Information Grid Architecture*, Version 2.0, Revised Draft, January 17, 2003.

DoD CIO, Directorate of Architectures and Integration, *Net-Centric Operations and Warfare (NCOW) Reference Model*, Version 0.9, July 2, 2003.

DoD CIO, Directorate of Information Management, *Net-Centric Enterprise Services*, Briefing, June 12, 2003.

2.8 REPRESENTING THE ROLE OF HUMANS IN ARCHITECTURES

2.8.1 Overview

For most systems, humans have a significant role in how systems perform and are operated. A description of the human tasks, activities, and the flow of information needed by humans to accomplish or support military operations can be represented in operational and systems architecture views.

Human-centered engineering plays a role in how systems are designed and how information is displayed. For example, the Joint Technical Architecture (JTA) defines specific guidelines for the design of human-computer interfaces (HCI). These JTA standard guidelines are intended to ensure that information is presented to the operator consistently across systems and in ways advantageous to human performance. However, before the detailed "how to" guidelines of HCI can be implemented, the operational aspects of human roles should be examined, described through the systems architecture view, and then used in analyses to help designers determine the scope of what information should be displayed or available. Then, and only then, the HCI guidelines mentioned in the JTA can be used to "engineer" an appropriate human interface.

Human supplements in military architectures extend far beyond computer interface design and can be used to address the full spectrum of human system integration (HSI) domains, including human factors engineering, manpower, personnel, training, survivability, safety, and health. The term *human supplement*, as used here, refers to information on human behavior added to an architecture description.

Humans also determine the operational use of systems. Systems must be supported by sufficient manpower, adequately trained to operate the system in the context of an operational mission. If human systems issues are not represented in the architectures during system design, factors affecting design, manpower, training and other human-related issues may be overlooked to the detriment of overall systems performance.

Modest investment in HSI during architecture development can potentially reduce total ownership costs. Over the life cycle of the system or system-of-systems, humans within the system are consistently the most costly resource. Efficient and effective use of humans within the system can ultimately reduce costs. Considering human factors during architecture development can also enhance overall systems performance by improving human performance through systems design, by helping to design effective training programs, and by validating manning requirements.

The human supplements recommended here are intended to help collectively define and describe the role of the human in the overall system. They are detailed and designed to link the Operational View (OV), Systems View (SV), and Technical Standards View (TV) to a human-centered style of systems engineering. Human supplements characterize the logical relationship between the human and the "machine" operating as a total system.

2.8.2 Roles of the Human Within Systems

Humans may have many roles associated with the operation of a system. A single human operator may occupy several different roles in the same overall system depending upon the situation and the interplay among the human, hardware, and software. A range of human roles is described below.

- *Passive Monitor*: In systems that are largely automated, the human may simply have to monitor system status and keep the system running. Passive monitoring of a system requires the operator to receive appropriate feedback from the system so that proper situational awareness can be maintained. That feedback must provide the right kind of information at the right time so that the human operator is able to make appropriate assessments.
- *Active Initiator*: Humans can also be active initiators of system processes to accomplish mission objectives. This is probably the role most familiar to people: "pushing buttons and making things go." Systems that require active participation and initiation from the operator are some of the most demanding to architect and to engineer. The architecture should reflect the human role and aspects of feedback, situational awareness, process initiation, and the principles of human factors engineering required to achieve human integration with the system. The architecture should also reflect how the system is actively carrying out operator initiated system functions to accomplish the mission. This means describing the mechanical underpinnings of system functions or how all the hardware and software work together.
- *Reporter*: Humans may have reporting roles whereby they may observe, monitor, and report or pass on information to higher decision-making authorities.
- *Planner & Decision Maker*: Many modern military systems are tools to assist the human's role as a planner or as a decision maker. These systems are usually information rich and provide the human with an intense capability to filter and analyze the information before postulating a plan or making a decision. Some decision-aiding systems can even provide recommendations based upon rules that simulate the human reasoning process. The advantage is a more complete processing of large amounts of data that would typically overwhelm human capacities in a short amount of time.

2.8.3 Human-Centered Architecture Supplements

Including the role of the human and human activities associated with the systems provides a basis for addressing human issues in engineering analyses. Supplementary human-centered information provides detail necessary to address human systems integration issues in engineering analyses.

Universal task lists are inadequate when system performance, requirements, technical, or cost-benefit analyses need to be conducted because they do not describe tasks to the level of detail necessary. A typical task list such as the Unified Joint Task List serves the purpose of outlining broad-area human tasking associated with the system so there is a general understanding of tasks that are performed. These task lists are useful when developing training. Human-centered architectural supplements provide additional levels of decomposition of universal task lists. This aids in understanding human performance issues associated with the system and enhances engineering analyses.

Issues such as manpower, personnel, training, and human factors engineering can be addressed in architectures that clearly reflect human activities associated with the system. Human-centered architectural supplements support human performance analyses as well as other systems engineering analyses such as:

- *Requirements Analysis* can contain an analysis of the human/operator requirements necessary to accomplish the mission (i.e., human information requirements, reporting requirements, and decision-making requirements).
- *Technology Analysis* can contain human factors engineering design requirements and criteria that can enhance human performance. The design of the system to meet human capabilities and address human limitations can be critical to system performance.
- *Performance Analysis* can contain results of human performance analyses of data from field experiments and exercises and from the modeling of architectures depicting human supplements. Overall system performance can be greatly impacted by the performance of humans who operate the system.
- *Cost-Benefit Analysis* can include the impact of the human element on total system cost. Human presence in a system can greatly impact cost-benefit trades when humans require training, life support, quality of life, protection, pay, and so forth. Production of human-centered architectural supplements that help describe numbers and quality of personnel needed to operate the system could provide the basic data necessary to address these issues in the Cost-Benefit Analysis.

Table 2.8-1 shows the human-centered architectural supplements as they relate to existing architectural products. Human-centered architecture supplements are an extension of the Organizational Relationships Chart (OV-4), Operational Activity Model (OV-5), Systems Functionality Description (SV-4), and Operational Activity to Systems Function Traceability Matrix (SV-5). The supplements add necessary information regarding human roles and activities.

Table 2.8-1. Human-Centered Supplementary Architectural Information

Applicable Architecture View	Product Reference	Architecture Product	General Nature of Product	Human Architecture Supplement	General Nature of Supplement
Operational	OV-4	Organizational Relationships Chart	Command, control, coordination relationships among organizations	Human Roles and Responsibilities	Define human roles and responsibilities related to the organizational structure
Operational	OV-5	Operational Activity Model	Activities, relationships, constraints, mechanisms to perform activities	Human Activity Model	Description of human functions, broad-level tasks and activities related to the operation of the system

Applicable Architecture View	Product Reference	Architecture Product	General Nature of Product	Human Architecture Supplement	General Nature of Supplement
Systems	SV-4	Systems Functionality Description	Functions performed by systems and info flow among functions	System Function Allocation	Description of functions to be performed by humans and those that are to be performed by machine/system
Systems	SV-5	Operational Activity to Systems Function Traceability Matrix	System functions to operational activities	Human Activities to Operational Activities Traceability Matrix	Human and system activities to operational activities

2.8.4 Human Systems Integration (HSI) Considerations for Architecture Products

2.8.4.1 Integrated Dictionary (AV-2)

Behavioral and human engineering terms can be used in human supplementary information. When used in any product, the terms should become part of AV-2. **Table 2.8-2**[1] provides useful terms for describing human activities associated with a system. Definitions in bold have been added to the original referenced source list.

Table 2.8-2. Behavioral Processes and Definitions

Processes	Activities	Specific Behaviors	Definitions
1. Perceptual	1.1 Searching For and Receiving Information	1.1.1 Inspects	To examine carefully or to view closely with critical appraisal.
		1.1.2 Observes	To attend visually to the presence or current status of an object, indication, or event.
		1.1.3 Reads	To examine visually information that is presented symbolically.
		1.1.4 Monitors	To keep track of over time.
		1.1.5 Scans	To quickly examine displays or other information sources to obtain a general impression.
		1.1.6 Detects	To become aware of the presence or absence of a physical stimulus.

[1] "Handbook of Human Factors," 1987. Edited by Gavriel Salvendy, pg. 398.

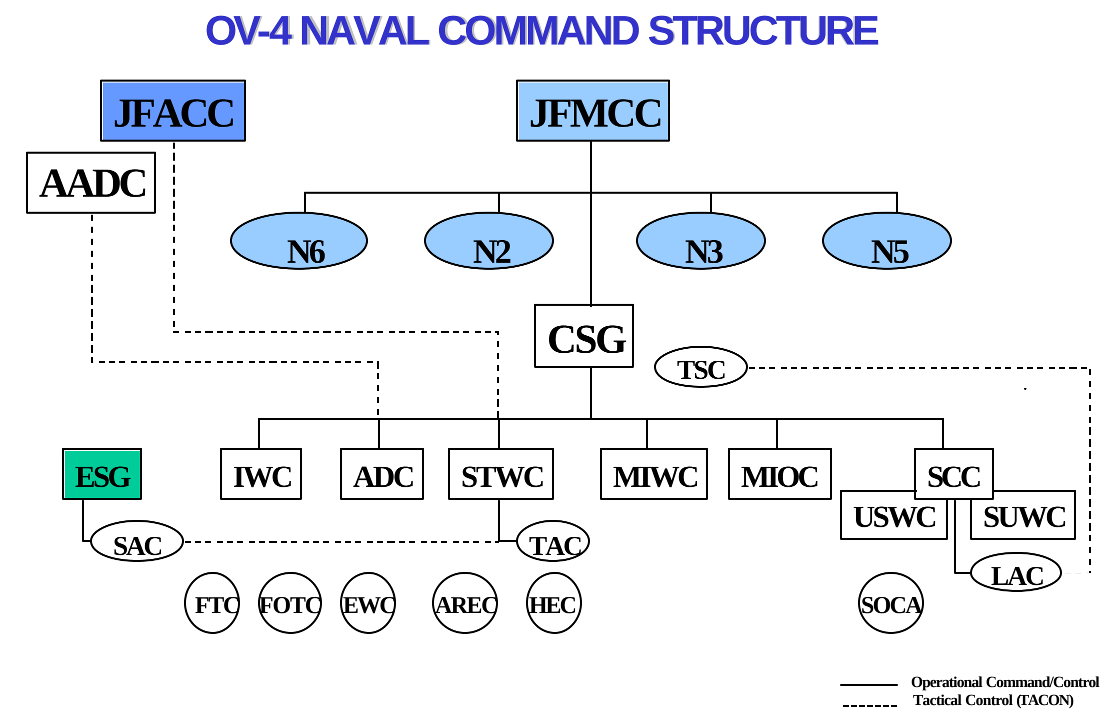
Processes	Activities	Specific Behaviors	Definitions
	1.2 Identifying Objects, Actions, Events	1.2.1 Identifies	To recognize the nature of an object or indication according to implicit or predetermined characteristics.
		1.2.2 Locates	To seek out and determine the site or place of an object.
2. Cognitive	2.1 Information Processing	2.1.1 Interpolates	To determine or estimate intermediate values from two given values.
		2.1.2 Verifies	To confirm.
		2.1.3 Remembers	To retain information (short-term memory) or to recall information (long-term memory) for consideration.
		2.1.4 Reviews	***To perceive and comprehend information.***
	2.2 Problem Solving and Decision Making	2.1.5 Calculates	To determine by mathematical processes.
		2.1.6 Chooses	To select after consideration of alternatives.
		2.1.7 Compares	To examine the characteristics or qualities of two or more objects or concepts for the purpose of discovering similarities or differences.
		2.1.8 Plans	To devise or formulate a program of future or contingency activity.
		2.1.9 Decides	To come to a conclusion based on available information.
		2.1.10 Diagnoses	To recognize or determine the nature or cause of a condition by consideration of signs or symptoms or by the execution of appropriate tests.
		2.1.11 Analyzes	***To review and interpret information.***
		2.1.12 Aggregates	***To combine information from multiple sources into a composite perspective.***
		2.2.9 Predicts	***To project future outcomes based on current events/information.***

Processes	Activities	Specific Behaviors	Definitions
3. Motor	3.1 Simple/Discrete	3.1.1 Moves	To change the location of an object.
		3.1.2 Holds	To apply continuous pressure to a control.
		3.1.3 Pushes/Pulls	To exert force away from/toward the actor's body.
	3.2 Complex/Continuous	3.2.1 Positions	To operate a control which has discrete states.
		3.2.2 Adjusts	To operate a continuous control.
		3.2.3 Types	To operate a keyboard.
4. Communication		4.0.1 Answers	To respond to a request for information.
		4.0.2 Informs	To impart information.
		4.0.3 Requests	To ask for information or an action.
		4.0.4 Records	To document something, as in writing.
		4.0.5 Directs	To order an action.
		4.0.6 Receives	To be given written or verbal information.
		4.0.7 Coordinates	***To manage people, resources, and activities for a specific objective or goal. This will include elements of the other communication functions (e.g., directs, informs, requests, etc.), depending on the situation.***

2.8.4.2 Command Relationships Chart (OV-4)

OV-4 describes the command, control, and coordination relationships among organizations (see **Figure 2.8-1**). A human-centered supplement to OV-4 describes roles and responsibilities of human operators and decision makers who populate the organizations. Gaps, overlaps, and unique roles for organizations and key personnel then become apparent.

Operational activities defined in OV-5 can be correlated with the position responsibilities to associate organizations and positions with operational activities. The description of roles and responsibilities can also help to address training as well as operational issues.



JFMCC	Joint Force Maritime Commander	IWC	Information Warfare Commander	MIWC	Mine Warfare Commander
JFACC	Joint Force Air Component Commander	ADC	Air Defense Commander	MIOC	Maritime Intercept Operation Commander
N6	Navy Staff - Communications	STWC	Strike Warfare Commander	SCC	Sea Combat Commander
N2	Navy Staff - Intelligence	TAC	Tactical Air Coordinator	USWC	Undersea Warfare Commander
N3	Navy Staff - Operations	AREC	Air Resource Element Coordinator	SUWC	Surface Warfare Commander
N5	Navy Staff - Plans	HEC	Helicopter Element Coordinator	TSC	Tomahawk Strike Coordinator
CSG	Carrier Strike Group	EWC	Electronic Warfare Coordinator	LAC	Launch Area Coordinator
ESG	Expeditionary Strike Group	FTC	Force Track Coordinator	SOCA	Submarine Operational Controlling Authority
SAC	Supporting Arms Coordinator	FOTC	Force Over-the-Horizon Track Coordinator		

Figure 2.8-1. Illustrative OV-4

Table 2.8-3 is a notional example of an OV-4 human supplement depicting human roles and responsibilities for positions outlined in Figure 2.8-1.

Table 2.8-3. Human-Centered Supplement to OV-4

Roles/Positions	Responsibility
JFMCC	Approval of Maritime action plans Redirection of all assets to support action on emergent threats
JFACC	Development and approval of air tasking Redirection of air assets to support action on emergent threats
N6	…
N2	…

2.8.4.3 Operational Activity Model (OV-5)

A supplement to OV-5 specifically addressing human activities can provide further understanding of the human role in the system so that human systems integration issues may be addressed. The operational functions described in a typical OV-5 are decomposed in this supplement into descriptions of activities that are accomplished by the system's human operator.

For each human activity listed, the information requirements are defined. Unlike information exchange requirements associated with systems, human information requirements are not restricted to streams of electronic information. Human information can originate from a variety of sources such as reports, databases, sensors, displays, or verbal communication. It is important to capture the source and modality of that information in the human supplement to OV-5.

For each human activity listed, there is a corresponding human behavioral process (e.g., a cognitive process) as well as the specific human behavior associated with that process (e.g., decision making). **Table 2.8-4** provides a sampling of the human activities and associated information that can be provided with OV-5. As shown in Table 2.8-4, cognitive processes and specific behavior can be included in the supplement.

Table 2.8-4. Human Activities Notional Example

(Partial list of human activities for Time-Critical Targeting [TCT])

TCT Activity	Human Functional Activity	Information Requirements	Process	Specific Behavior
1.1 Assess ISR Cue				
1.1.1	Determine if cue qualifies as TCT	TCT list, Commander's guidance, ROE	Cognitive	Decides
1.1.2	Determine time latency of cueing by subtracting current time from time stamp of cue (Provides indication of likelihood that target is still in same location)	Current time, time stamp of cue	Cognitive	Calculates
1.1.3	Compare IMINT cue characteristics against known IMINT data	IMINT characteristics, access to relevant data bases, TCT list	Cognitive	Compares
1.1.4	Compare SIGINT cue characteristics against known SIGINT data	SIGINT data, access to relevant data bases, TCT list	Cognitive	Compares
1.1.5	Compare voice cues against other target data	Voice reports, access to relevant data bases, TCT list	Cognitive	Compares
1.1.6	Compare text messages against other target data	Contents of messages, access to relevant data bases, TCT list	Cognitive	Compares
1.1.7	Review and understand Commander's guidance and ROE	Commander's Guidance, ROE	Cognitive	Reviews
1.1.8	Decide if cueing criteria is satisfied	Cue content, Cueing criteria	Cognitive	Decides

As human activities are described, the activities can be concurrently categorized by behavioral process and specific behavior into one chart. This allows some preliminary analysis to be done concurrently with the identification of the human activities. A frequency chart for the processes and specific behaviors can be developed from the table of human activities in the example above. **Figure 2.8-2** provides an example tabulation of the human activities involved in a typical time critical targeting scenario. As seen in Figure 2.8-2, human cognitive and communication activities dominate the TCT task; therefore, system designers should address these operator needs so that overall system performance is enhanced.

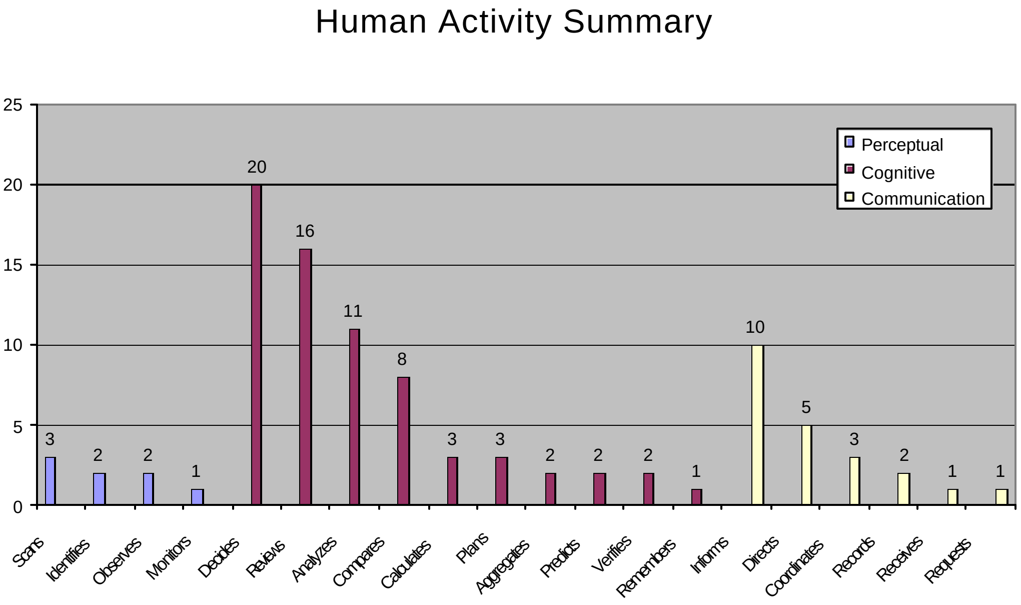


Figure 2.8-2. Notional Example of Analysis of Human Activities for Time-Critical Targeting

The human activities model maintains the same hierarchical nature as the OV-5. However, human activities are documented that add more detail to the operational activities in OV-5.

The human activities model also supplements SV-5. Human activities are mapped to operational activities in SV-5.

2.8.4.4 Systems Functionality Description (SV-4)

Human supplementary information for SV-4 describes which operational activities are supported by machine functions and which humans perform. SV-4 documents system functional hierarchies, system functions, and the data flows between them. This supplement focuses on human activities and the data flow between human activities. Data flow between human activities is essential to address decision-making issues. System functions that support human activities are also depicted in this supplement (see **Figure 2.8-3**).

In keeping with the human-centered design approach, it is important for system architects and system designers to outline the responsibilities of human operators with regard to the system. Details of the operational activities performed by humans and operational activities performed by machines should be depicted in this supplement.

Strike		System Functions																Human Activities							
SF#	System Functions - 3.0 (Act)	2.2.1	2.2.3	2.2.3.1	2.2.3.4	2.3	2.4	2.4.1	2.4.2	2.4.3	2.4.4.4	2.4.5.5	2.4.6	3.1.1	3.1.5	5.1.1.1	5.3	3.1	3.1.1	3.1.2	3.1.3	3.1.4	3.1.5	3.1.6	3.1.7
3.1	**Engagement Execution**																								
3.1.1	Direct Attack/Evasive Maneuvers													X		X			X					X	X
3.1.2	Determine Engageability											X					X			X		X			
3.1.2.1	Develop Intercept Prediction											X					X								
3.2	**Target Development**																								
3.2.1	Employ Targeting Assets	X										X													
3.2.1.1	Task/Re-task Targeting Assets	X										X							X				X		
3.2.1.1.1	Transmit Tasking and Target Data to Targeting Assets	X	X	X								X									X			X	
3.2.2	Designate Target												X	X											X
3.2.4.1	Determine Target Location			X		X						X	X	X			X								X
HA#	Human Activities – 3.0 (Act)																								
3.1	**Assess ISR Cue**																								
3.1.1	Determine if cue qualifies as TCT	X																							
3.1.2	Determine time latency of cueing	X			X																				
3.1.3	Compare IMINT cue characteristics against known IMINT data		X																						
3.1.4	Compare SIGINT cue characteristics against known SIGINT data		X																						

Figure 2.8-3. Notional Example of Human-Centered Supplementary Information for the SV-4

The SV-4 human supplement can help address HSI issues of human workload associated with the system by helping to determine how best to allocate tasks for machine assistance. In turn, an understanding of the workload for human operators can help address issues of human performance, potential operator fatigue and design issues focused on offsetting human workload. This supplement can be a substantial aid in understanding the human role in legacy systems. New systems should be designed to concurrently address human workload issues by the application of human-centered design approaches.

2.8.4.5 Operational Activity to Systems Function Traceability Matrix (SV-5)

As a part of SV-5, human activities associated with each operational activity are tabulated along with the system functions. This provides supplemental insight into the level of responsibility of human operator in the achievement of the operational activity.

If the human operator is eliminated, other hardware and/or software systems elements will be required to perform the activities once performed by humans within the system. The matrix provides a means of visually correlating system functions and human activities to operational activities in a concise manner. Understanding the human activities within the context of the system allows for better precision within the system engineering analyses. On a systems and a system-of-systems level, engineering analyses of human activities tend to remain a "black box" factor.

The SV-5 human supplement can provide supplemental engineering information (see **Figure 2.8-4**) to aid in analyses and resource allocation decisions by the system architects and engineers. The information will also aid in understanding the human role within legacy systems.

		Operational Activities Assess															
Strike		Collect				Exploit								Remove from Target List	Decide TCT Negation – Pair Weapon/Pltfm/Snsr to Target		
SF#	System Functions - 3.0 (Act)	2.2.1	2.2.3	2.2.3.1	2.2.3.4	2.3	2.4	2.4.1	2.4.2	2.4.3	2.4.4.4	2.4.5.5	2.4.6	3.1.1	3.1.5	5.1.1.1	5.3
3.1	**Engagement Execution**																
3.1.1	Direct Attack/Evasive Maneuvers													X		X	
3.1.2	Determine Engageability											X					X
3.1.2.1	Develop Intercept Prediction											X					X
3.2	**Target Development**																
3.2.1	Employ Targeting Assets	X										X					
3.2.1.1	Task/Re-task Targeting Assets	X										X					
3.2.1.1.1	Transmit Tasking and Target Data to Targeting Assets	X	X	X								X					
3.2.2	Designate Target												X	X			
3.2.4.1	Determine Target Location			X		X						X	X	X			X
HA#	Human Activities – 3.0 (Act)																
3.1	**Assess ISR Cue**																
3.1.1	Determine if cue qualifies as TCT	X															
3.1.2	Determine time latency of cueing	X			X												
3.1.3	Compare IMINT cue characteristics against known IMINT data		X														
3.1.4	Compare SIGINT cue characteristics against known SIGINT data		X														

Figure 2.8-4. Notional Example of Human Supplemental Information for the SV-5

2.8.5 Reference

Salvendy, Gavriel, editor, *Handbook of Human Factors*, 1987.

2.9 CAPABILITY MATURITY PROFILE

The Capability Maturity Profile was proposed as a third All-Views product in the C4ISR Architecture Framework Draft, October 1, 2001. It was not retained as a product in the DoD Architecture Framework (DoDAF).

Definition. This profile addresses an architecture's current capabilities and future requirements, which are derived from and guided by the encompassing enterprise's requirements. The Capability Maturity Profile for an architecture also includes an analysis of the current capability maturity level achieved and the future capability maturity level required.

Purpose. The Capability Maturity Profile can aid in the transition from an As-Is to a To-Be architecture.

Detailed Description. To build a Capability Maturity Profile, an architect may start by using a Capability Maturity Roadmap. Such a roadmap is usually defined by an enterprise. Various architecture projects within the same enterprise then follow the same roadmap. In this manner, an enterprise may transition to the level of maturity goals it has set. Plans for achieving the maturity goals can be laid out in the Systems Evolution Description (SV-8).

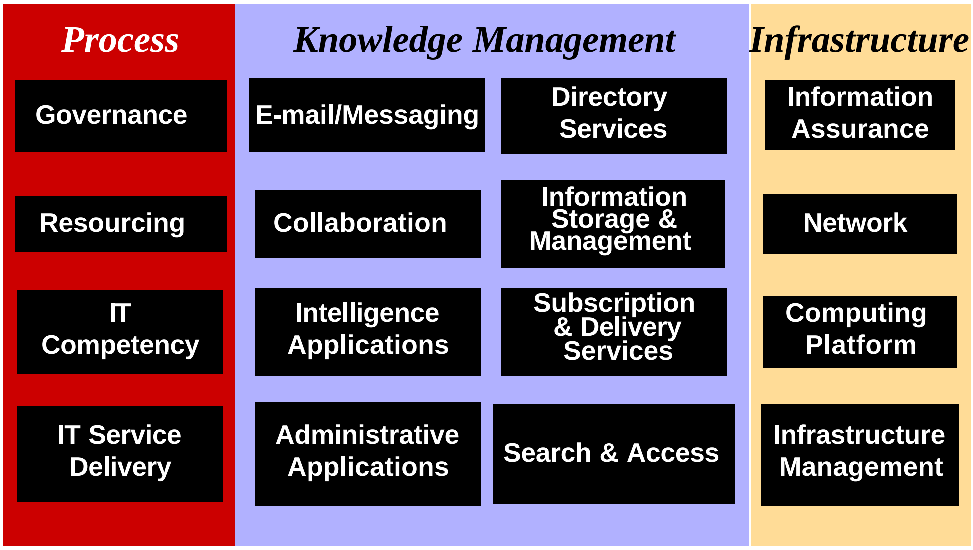
2.9.1 Capability Maturity Roadmap

Building a Capability Maturity Profile involves the use of an enterprise-wide Capability Maturity Roadmap. Creating such a roadmap for an enterprise consists of identifying and prioritizing requirements and developing investment strategies for maturing enterprise capabilities, guiding resource and policy decisions, coordinating executive agent activities, and tracking progress.

A general approach to structuring a roadmap may consist of breaking requirements (i.e., constructing a future or a To-Be roadmap structure) into generic categories. For the purposes of description, an example roadmap from the Intelligence Community (IC) is used (see "Intelligence Community Information Systems Capabilities Roadmap" section in the Universal Reference Resources described in this Deskbook). This example roadmap was developed by the IC to track their information technology (IT) maturity levels. Three example IT categories, as defined by the IC, are used. These are Process, Knowledge Management, and Infrastructure.

The Process category includes the roadmap components that are concerned with enterprise IT governance, resource provisioning, training, and fielding. The Knowledge Management category includes the roadmap components that are responsible for building and evolving the shared information space of the enterprise and for equipping IC users and mission customers with the means of expediently acquiring the information they need. The Infrastructure category includes the backbone components of the enterprise that support the Knowledge Management capabilities and services. An example roadmap developed for use by the IC appears in **Figure 2.9-1**.

The example roadmap shown in Figure 2.9-1 is further divided into 16 component areas. These 16 component areas can be used as the fundamental IT building blocks—or modules—to consider in assessing an As-Is architecture and in examining options for a corresponding To-Be architecture. It should be noted that there are many ways that an enterprise's IT capability can be categorized and segmented within categories. However, the above IT roadmap example is the component breakout that resulted from many iterations with IC Chief Information Officers for use in building an IT Capability Maturity Profile.



Process	*Knowledge Management*		*Infrastructure*
Governance	E-mail/Messaging	Directory Services	Information Assurance
Resourcing	Collaboration	Information Storage & Management	Network
IT Competency	Intelligence Applications	Subscription & Delivery Services	Computing Platform
IT Service Delivery	Administrative Applications	Search & Access	Infrastructure Management

Figure 2.9-1. IC Community Example Roadmap

2.9.2 Capability Maturity Scale

A standard capability maturity scale may be employed to describe the maturity model for the components of a roadmap such as the one defined by the Software Engineering Institute's (SEI) Capability Maturity Mode (CMM). SEI's standard scale addresses five generic levels of increasing capability. The levels progress from an ad hoc state (wherein each organization of the enterprise acts autonomously), to an optimizing state (wherein all member organizations of the enterprise—and the global partners of the enterprise—experience the benefits of interoperability and resource sharing). It is important to note that there are two main dimensions of enterprise capability that change as a roadmap component progresses from Level 1 through Level 5: breadth of outreach or global participation and sophistication of capability. Increase in outreach is fundamentally enabled by cultural and policy changes. Sophistication of capability is heavily influenced by technology evolution and cultural assimilation.

2.9.3 Capability Maturity Profile Example

An example from the IC is used as a basis for explaining the steps to build a Capability Maturity Profile. The example provides a graphical sequence that illustrates how a particular Capability Maturity Profile can be built for a specific architecture. This walk-through is an example only, intended to illustrate the analytical process used in building this profile. In this example, it is assumed that analysis of the Operational View (OV) and the As-Is Systems View (SV) of the architecture indicates that a higher degree of collaboration capability needs to be achieved within the enterprise. The focus on the collaboration component is illustrated in **Figure 2.9-2**.

Example: Achieving Improved *Collaboration* was Determined to be the To-Be Focus of the Systems View

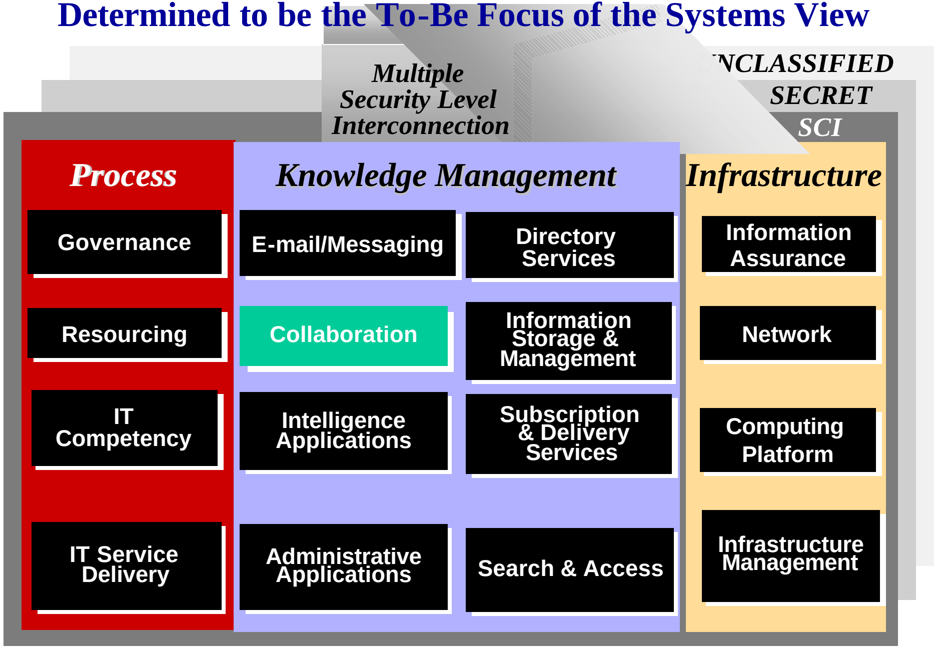
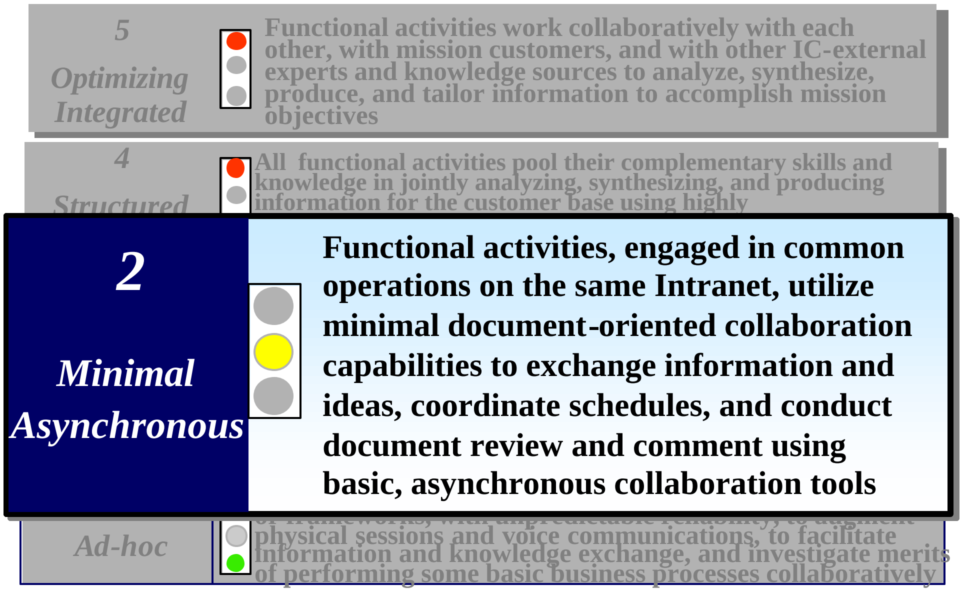


Figure 2.9-2. Collaboration as the Focus Area

For this example, by examining the definitions of each of the five maturity levels of collaboration, the architect has determined that the enterprise in question currently has a Level 1 collaboration capability and that the goal of his enterprise is to achieve Level 2 collaboration. **Figure 2.9-3** displays an example definition of Level 2 collaboration. The yellow "traffic light" circle in the gray box indicates that the enterprise in question has already achieved some, but not all, of the requirements for Level 2 collaboration. Each architecture project should develop its own definitions for the various component capability levels and use those definitions in building its capability profile.



Functional activities work collaboratively with each other, with mission customers, and with other IC-external experts and knowledge sources to analyze, synthesize, produce, and tailor information to accomplish mission objectives

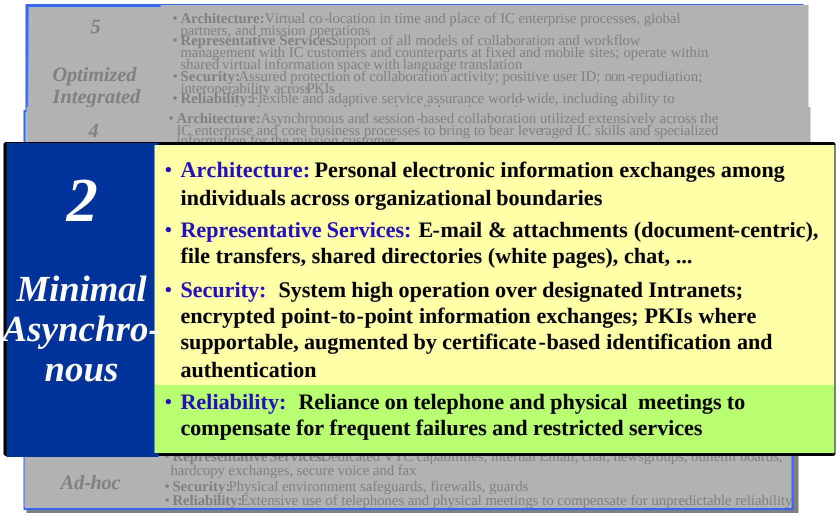
All functional activities pool their complementary skills and knowledge in jointly analyzing, synthesizing, and producing information for the customer base using highly

Functional activities, engaged in common operations on the same Intranet, utilize minimal document-oriented collaboration capabilities to exchange information and ideas, coordinate schedules, and conduct document review and comment using basic, asynchronous collaboration tools

physical sessions and voice communications, to facilitate information and knowledge exchange, and investigate merits of performing some basic business processes collaboratively

Figure 2.9-3. Basic Definition of Level 2 Collaboration

The basic example definition of Level 2 collaboration does not by itself provide enough information to make the kinds of decisions that will be needed to build a To-Be Systems View. Therefore, the architect also needs to develop a more detailed definition. A detailed example definition of Level 2 collaboration is highlighted in **Figure 2.9-4**.



5

Optimized Integrated

- **Architecture:** Virtual co-location in time and place of IC enterprise processes, global partners, and mission operations
- **Representative Services:** Support of all models of collaboration and workflow management with IC customers and counterparts at fixed and mobile sites; operate within shared virtual information space with language translation
- **Security:** Assured protection of collaboration activity; positive user ID; non-repudiation; interoperability across PKIs
- **Reliability:** Flexible and adaptive service assurance world-wide, including ability to

4

- **Architecture:** Asynchronous and session-based collaboration utilized extensively across the IC enterprise and core business processes to bring to bear leveraged IC skills and specialized information for the mission customer

2

Minimal Asynchro-nous

- **Architecture:** Personal electronic information exchanges among individuals across organizational boundaries
- **Representative Services:** E-mail & attachments (document-centric), file transfers, shared directories (white pages), chat, ...
- **Security:** System high operation over designated Intranets; encrypted point-to-point information exchanges; PKIs where supportable, augmented by certificate-based identification and authentication
- **Reliability:** Reliance on telephone and physical meetings to compensate for frequent failures and restricted services

Ad-hoc

- **Representative Services:** Dedicated VTC capabilities, internal Email, chat, newsgroups, bulletin boards, hardcopy exchanges, secure voice and fax
- **Security:** Physical environment safeguards, firewalls, guards
- **Reliability:** Extensive use of telephones and physical meetings to compensate for unpredictable reliability

Figure 2.9-4. Detailed Definition of Level 2 Collaboration

At this point in the example, the architect has determined the primary focus of the enterprise improvement effort (collaboration component), the level of collaboration needed (Level 2), and the details of what constitutes that level. However, collaboration should not be addressed independently but must be considered in terms of its relation to other roadmap component areas. Attempting to improve collaboration without regard for these other areas may negatively impact other areas to an unacceptable extent. For example, the architect might introduce a video-teleconferencing capability without increasing the bandwidth on the supporting network. Improving the accessibility and sharing of information might raise the collaboration level, but at the expense of adding security risks if appropriate information assurance measures are not implemented. Having adequate backup to the desired collaboration level in the form of basic e-mail with attachments might be overlooked. Also the critical importance of Directory Services should be considered, since Directory Services enables collaborators to identify with whom they want to share information and how to reach them. Finally, network connectivity might need to be extended to include collaboration partners separated by firewalls or different protocols today. Thus, the architect needs to examine the other component areas to determine which of those other areas potentially affect, or are affected by, Level 2 collaboration.

By examining the definitions of other roadmap component areas, the architect determines the roadmap component areas that are most directly linked with achieving a desired capability level. In this example, the component areas related to (or that influence) the desired collaboration Level 2 are Network, E-mail/Messaging, Directory Services, and Information Assurance. **Figure 2.9-5** highlights the interdependent component areas in this example.

The next step is to determine what capability level is needed in each of these other component areas as a prerequisite for achieving Level 2 collaboration. In this example, **Figure 2.9-6** illustrates that the architect has determined that a Level 2 capability is needed in the Network component area, and a Level 3 capability is needed in the Information Assurance, E-mail/Messaging, and Directory Services component areas.

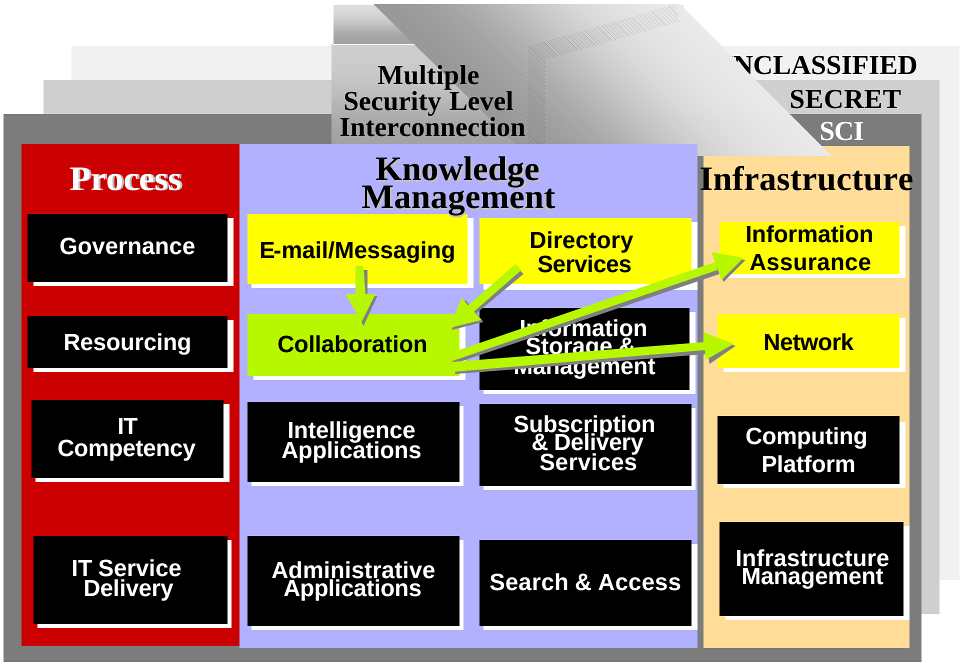
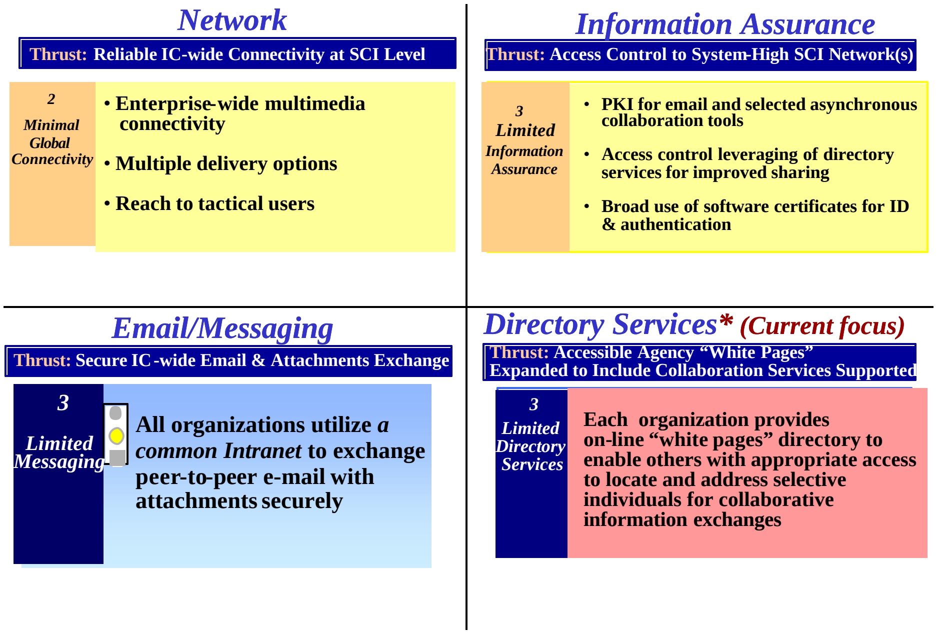


Figure 2.9-5. Other Component Areas That Interact with Collaboration



Network

Thrust: Reliable IC-wide Connectivity at SCI Level

2 Minimal Global Connectivity

- Enterprise-wide multimedia connectivity
- Multiple delivery options
- Reach to tactical users

Information Assurance

Thrust: Access Control to System-High SCI Network(s)

3 Limited Information Assurance

- PKI for email and selected asynchronous collaboration tools
- Access control leveraging of directory services for improved sharing
- Broad use of software certificates for ID & authentication

Email/Messaging

Thrust: Secure IC-wide Email & Attachments Exchange

3 Limited Messaging

All organizations utilize *a common Intranet* to exchange peer-to-peer e-mail with attachments securely

Directory Services (Current focus)*

Thrust: Accessible Agency "White Pages" Expanded to Include Collaboration Services Supported

3 Limited Directory Services

Each organization provides on-line "white pages" directory to enable others with appropriate access to locate and address selective individuals for collaborative information exchanges

Figure 2.9-6. Capability Levels Needed in the Related Component Areas

The above example has demonstrated that the process of building the Capability Maturity Profile is a process of *selecting and extracting* the appropriate materials from a defined roadmap, not one of *creating* new material. This process is similar in that way to the process of creating the Technical Standards Profile (TV-1) by extracting the appropriate standards from existing standards documents.

The composite capability profile for a specific architecture provides a basis for capability-based investment strategies and budgeting practices. In the example provided, the capability is the achievement of the ability to conduct asynchronous collaboration among the participants engaged in a particular mission operation. Once specific process and technology solutions and improvements are examined and selected for implementing the elements in the composite Capability Maturity Profile, the costs associated with and summed across all of the profile elements represent the total cost for achieving a desired capability. The ability to relate component costs to the enabling of an enterprise capability can serve to better defend investment proposals and to support cost-effectiveness arguments.

2.9.4 References

Intelligence Community Chief Information Officer, *Intelligence Community Information Systems Capabilities Roadmap*, 2000.

Intelligence Community Chief Information Officer, *Strategic Direction for Intelligence Community Information Systems*, April 2000.

2.10 ARCHITECTURE LEVELS OF DETAIL

2.10.1 Introduction

Most graphical products (e.g., Operational Node Connectivity Description [OV-2], Operational Activity Model [OV-5], and Systems Interface Description [SV-1]) permit the modeling of their respective data elements using decomposition (i.e., several diagrams of the same product may be developed, where each diagram shows an increasing level of detail). This section discusses levels of detail from the perspective of five representative types of users: *Planner, Owner, Designer, Builder,* or *Contractor*.[1] In general, the level of usable detail increases as the perspective changes from the Planner, to the Contractor, when systems are actually implemented. These perspectives influence the amount of detail described by the architecture and evolve the architecture from the scope (mission area or domain) model, to the business (or operational) model, to the system, to the technology, and to the detailed representation models.

There are six data *primitives* that constitute the building blocks for describing an architecture.[2] In the DoD Architecture Framework (DoDAF), these are called *architecture data elements*. In the Institute of Electrical and Electronics Engineers (IEEE) STD 1472, they are referred to as *components*. These primitives are data, functions, networks, people, time, and motivation. They constitute the *what, how, where, who, when, and why* of architecture.

The DoDAF products represent composites of data primitives. For example, the Systems Functionality Description (SV-4) describes system functions (i.e., *how*) as well as the data (i.e., *what*) produced and consumed (i.e., *relationships*) by the functions. SV-4 may be developed for several perspectives depending on the intended use (e.g., Designer, Builder, or Contractor) and audience for the architecture. The DoDAF products allow the architect to develop an architecture description in accordance with the IEEE STD 1472 definition: "An architecture is the fundamental organization of a system embodied in its *components*, their *relationships to each other,* and to the *environment and the principles guiding* its design and evolution."

Consequently, a set of applicable DoDAF products may be developed for each of the user perspectives. Each set represents a different architecture model. Usually, an architecture model for a specific perspective is developed by refining products from the model developed for the preceding perspective, by adding detail to these products, by adding new products as needed, and by possibly iterating through the perspectives as new information is gathered and insight into architecture primitives and their relationships is gained throughout the description process.

This section provides suggested guidelines on which products are applicable for each user perspective or level of detail. As the architect moves from the Planner's perspective to the Contractor's perspective, the descriptions expand in terms of the level of detail provided in each product. Additional products are also added that describe architecture characteristics relevant to the perspective at hand. In the following sections, a brief listing of applicable products is provided for each perspective or level of detail (starting with the top row in the Zachman Framework). Products that were developed for one perspective will not always be repeated (or always show on the figure) for the following perspective, unless added emphasis on the refinement of the specific product is needed. However, the guidance provided in this section

[1] The designation of Planner, Owner, Designer, and Builder was initially defined by Zachman (Zachman, 1987).

[2] The six architecture primitives were defined in the Zachman Framework (Zachman, 1987).

assumes a cumulative development approach and a continuing refinement of products as the level of detail proceeds from the Planner level to the Contractor level.

2.10.2 Planner/Scope

At the Planner level, certain products provide a brief overview and a summary of the data, functions, networks, people, time, and motivation (what, how, where, who, when, and why) for a certain areas of concern (see **Figure 2.10-1**). A brief explanation of how the products appearing in this figure are useful to the Planner follows.

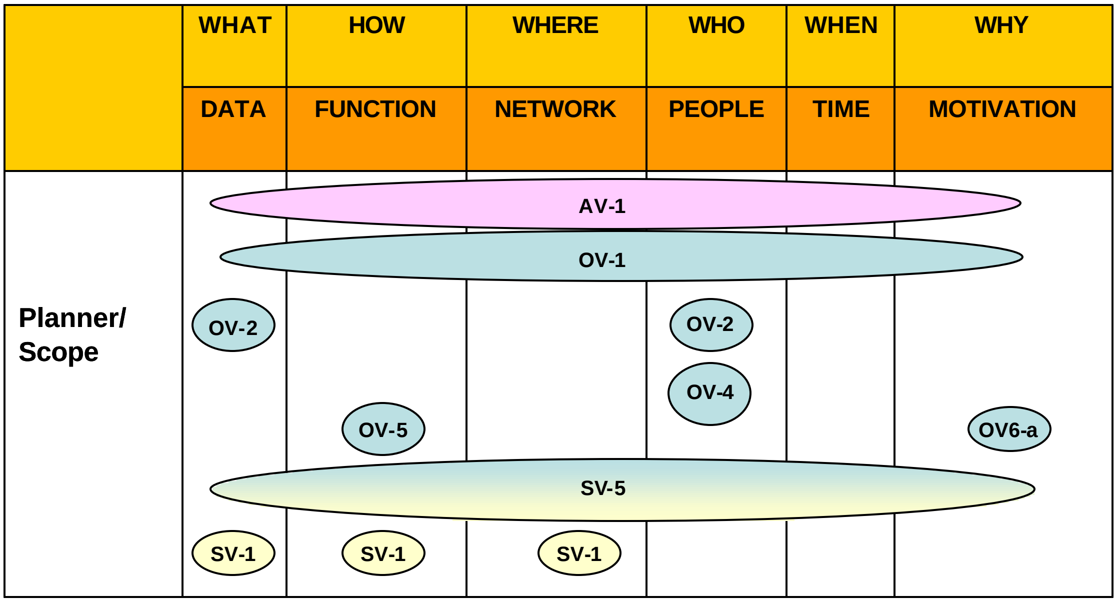


Figure 2.10-1. DoDAF Support to the Planner

2.10.2.1 Overview and Summary Information (AV-1)

The Overview and Summary Information (AV-1) provides architecture information on what, how, where, who, when, and why, at the Planner's level of detail. During the course of developing an architecture, several versions of this product may be produced:

- An initial version of this product may be produced to focus the effort and to document its scope, the organizations involved, and so forth.
- After other products within the architecture's scope have been developed and verified, another version of this product may be produced to reflect adjustments to the scope and to other architecture aspects that may have been identified as a result of the architecture development, so that an accurate record of these aspects of the architecture may be documented.
- After the architecture has been used for its intended purpose, and the appropriate analysis has been completed, yet another version may be produced to summarize these findings, in order to present them to high-level decision makers. In this final version, the AV-1 product (along with a corresponding graphic in the form of an OV-1 product) serves as the executive summary for the architecture.

2.10.2.2 High-Level Operational Concept Graphic (OV-1)

The High-Level Operational Concept Graphic (OV-1) provides a visual summary of the architecture information on what, how, where, who, when, and why, at the Planner level of detail. During the course of developing an architecture, several versions of this products may be produced:

- An initial version of this product may be produced to focus the effort and illustrate its scope.
- After other products within the architecture's scope have been developed and verified, another version of this product may be produced to reflect adjustments to the scope and other architecture details that may have been identified as a result of the architecture development.
- After the architecture has been used for its intended purpose, and the appropriate analysis has been completed, yet another version may be produced to summarize these findings, in order to present them to high-level decision makers.

2.10.2.3 Operational Node Connectivity Description (OV-2)

The Operational Node Connectivity Description (OV-2) graphically illustrates the "people" (Operational Nodes) and the "data" (needlines depicting an aggregation of required information) primitives and the relationships among them (i.e., which people exchange what data with whom) at the Planner and Owner levels of detail.

2.10.2.4 Organizational Relationships Chart (OV-4)

The Organizational Relationships Chart (OV-4) illustrates "people" primitives and the relationships between them (e.g., command relationships) at the Planner and Owner levels of detail. They correspond to the operational nodes of OV-2. At the Planner level, the organizations at the top of the organizational chart hierarchy are presented as Planner level operational nodes on OV-2. At the Owner level, the sub-organizations or human roles at the leaf level of the organizational chart hierarchy are presented as Owner level operational nodes on OV-2.

2.10.2.5 Operational Activity Model (OV-5)

The Operational Activity Model (OV-5) graphically illustrates "functions" (operational activities) and "data" (Information Elements) primitives and the relationships between them (which functions produce what data and which functions consume what data) of a given architecture at the Planner and Owner levels of detail. In addition, OV-5 may be annotated to describe the "people" (relationship to OV-2 Operational Nodes) performing these functions, at the Planner level of detail. At the Planner level, OV-5 may also be used to associate capabilities with sequences of operational activities.

2.10.2.6 Operational Rules Model (OV-6a)

The Operational Rules Model (OV-6a) may be used to provide "motivation" primitives for the architecture. At the Planner level, it provides the purpose and scoping rules for the architecture. At the Owner level of detail, it provides rules for business operations or doctrine influencing operational activities.

2.10.2.7 Systems Interface Description (SV-1)

The Systems Interface Description (SV-1) product graphically illustrates "network" (systems nodes) primitives and the relationships among them at the Planner, Owner, and Designer levels of detail. In addition, it provides "data" characteristics via the system interfaces, which denote that data is exchanged among the systems at the Planner level of detail. The relationship denoting which people are deployed at which systems nodes is also described in this product by tying the systems nodes (housing the systems) to the operational nodes (using the systems).

2.10.2.8 Operational Activity to Systems Function Traceability Matrix (SV-5)

For the Planner, the Operational Activity to Systems Function Traceability Matrix (SV-5) depicts the mapping between the capabilities and systems, and thus identifies the transition of a capability into a planned or fielded system. Such a matrix allows decision makers to quickly identify stovepiped systems, redundant/duplicative systems, gaps in capability, and possible future investment strategies all in accordance with the time stamp given to the architecture.

SV-5 can also be used to identify system functions that would *not* be satisfied if a specific system is *not* fielded to a specific unit in the architecture.

2.10.3 Owner/Business Mode

At the Owner level, the Planner's perspective is refined by adding architecture detail to the products previously listed. Care should be taken not to make design decisions that are too detailed as to limit the ability of the systems Designers (next perspective) to explore several design options. Additional products are also developed (see **Figure 2.10-2**).

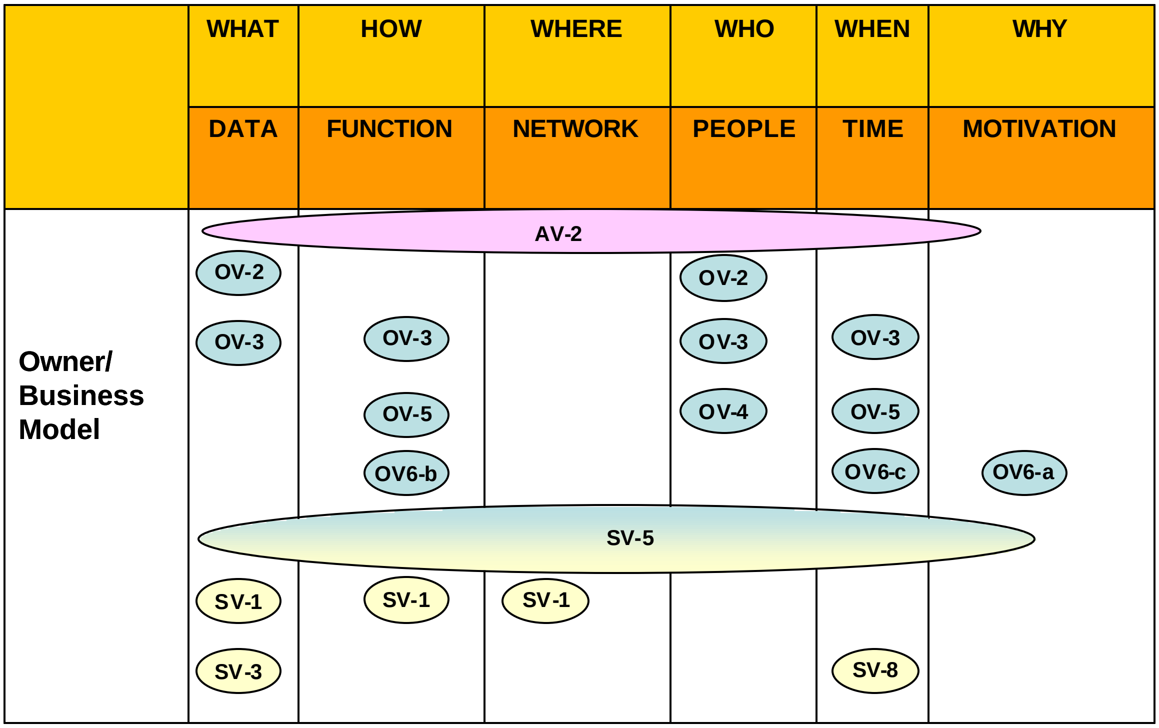


Figure 2.10-2. DoDAF to the Owner

2.10.3.1 Integrated Dictionary (AV-2)

The Integrated Dictionary (AV-2) is intended for use as a reference at the Owner, Designer, and/or Builder levels of detail. It provides a centralized location where the data, functions, networks, people, time, and motivation (i.e., what, how, where, who, when, and why) for a certain area of concern can be reviewed.

2.10.3.2 Operational Information Exchange Matrix (OV-3)

The Operational Information Exchange Matrix (OV-3) allows the architect to provide details on the architecture's "data" primitives, here called *information elements*. The matrix also relates these information elements to "people" (or Operational Nodes), "functions" (or business processes/operational activities of OV-5), "time" (periodicity), and other attributes associated with the exchange of the information. At this level, the matrix is an expansion of the needlines of the Planner level into their constitute information elements and the characteristics of the information exchange.

2.10.3.3 Operational Activity Model (OV-5)

At the Owner level of detail, the Operational Activity Model (OV-5) may be used to further decompose "functions" (operational activities) and "data" (Information Elements) primitives and the relationships between them (i.e., which functions produce what data and which functions consume what data) of a given architecture.

In addition to describing which functions produce what data and which functions consume what data, OV-5 may be annotated to describe "time" primitives (or describing a sequence of the activities) at the Owner level of detail by associating this product with an OV-6c product (see OV-6c description).

2.10.3.4 The Operational State Transition Description (OV-6b)

The Operational State Transition Description (OV-6b) provides details on the "function" primitives at the Owner, Designer, and Builder levels of detail. It specifies what happens when a certain function is executed or when a certain input is received. It relates the how to the why primitives.

2.10.3.5 Operational Event-Trace Description (OV-6c)

The Operational Event-Trace Description (OV-6c) product describes "time" primitives at the Owner level of detail. At the Planner level, OV-5 is used to associate capabilities with sequences of operational activities. At the Owner level, the capability is further characterized by the timing of these sequences of activities, and the details of the information exchanges between them. This can be accomplished by OV-6c, which relates the functions from OV-5 to the information in a time-sequenced manner.

2.10.3.6 Systems Interface Description (SV-1)

The Systems Interface Description (SV-1) product graphically illustrates "network" (systems nodes) primitives and "data" characteristics and the relationships among them at the Planner, Owner, and Designer levels of detail.

At the Owner level, systems and/or subsystems resident at the systems nodes, or the system functions required to automate some of the operational activities (from OV-5), or to implement certain capabilities, may be specified. That is, SV-1 relates the *network, data,* and *function* primitives at the Owner level of detail. It is more relevant to specify system functions at this level than it is to assign functionality to existing or future systems. What is required at this level is to show what automated functionality is needed at various nodes to implement or support certain capabilities, and not to include detail (such as assigning new functionality to systems) that might restrict or prematurely influence the design decision that are the domain of the next level.

For architectures that involve legacy systems, SV-1 may be used to specify the functions that are already supported by these legacy systems and that play a part (i.e., restrict, constrain, or influence design decisions during the next levels of detail) in the architecture development at hand. Analysis as to which functions exist in which legacy systems, and the identification of redundant functionality supported by multiple systems may be one use of the systems view products at the owner's level of detail.

2.10.3.7 Operational Activity to Systems Function Traceability Matrix (SV-5)

An Operational Activity to Systems Function Traceability Matrix (SV-5) correlates capability requirements that would not be satisfied if a specific system is not fielded to a specific unit in the architecture.

At the Owner level, the matrix depicts the mapping of operational activities (from OV-5 at the Owner level) to system functions (from SV-1 at the Owner level, showing the assignment of system functions to systems nodes) and essentially identifies the transformation of an operational need into a purposeful action performed by a system.

2.10.3.8 Systems-Systems Matrix (SV-3)

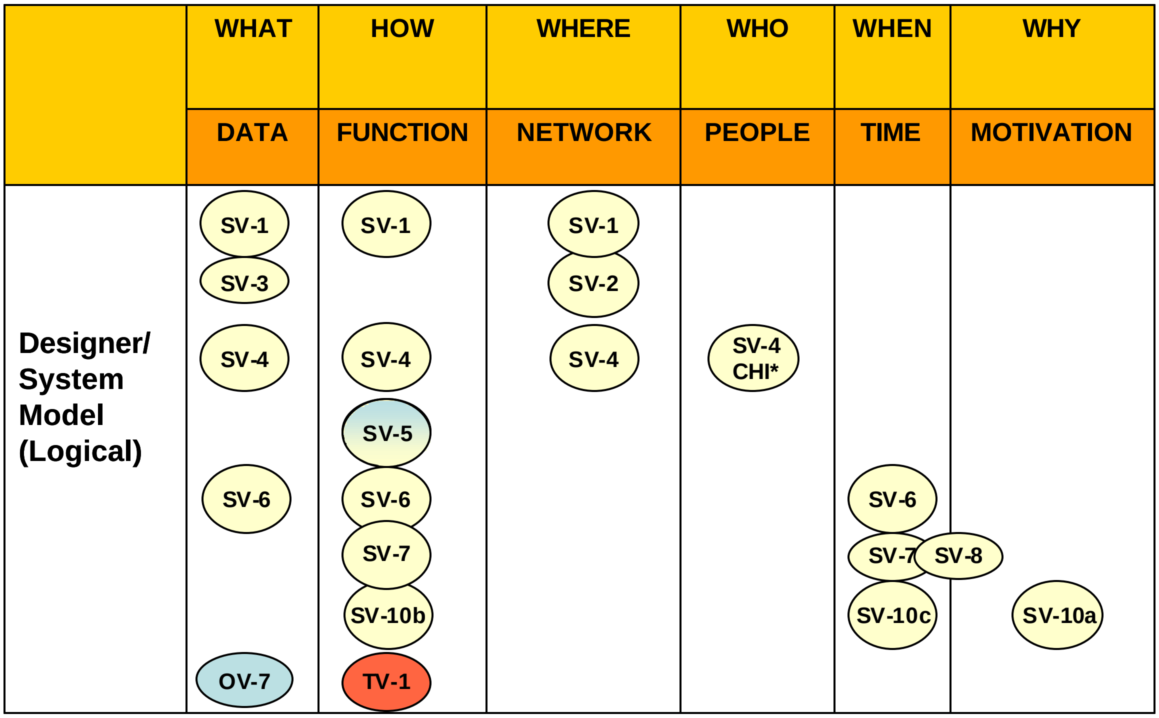
The Systems-Systems Matrix (SV-3) illustrates characteristics of “data” primitives (system interfaces or aggregates of the system data exchanges) at the Owner level of detail. Designations of key interfaces may be detailed at this level.

2.10.3.9 Systems Evolution Description (SV-8)

The Systems Evolution Description (SV-8) provides migration or evolution characteristics of the “functions” (systems or system functions) as they relate to “time” primitives at the Owner level of detail.

2.10.4 Designer/System Model

From the Designer’s perspective, details such as the system functions, the data produced and consumed by these functions, the systems that implement these functions, and the technical standards that constrain the design are specified at this level. The SV products developed in the Owner’s perspective are further refined here. **Figure 2.10-3** lists the SV products that add design detail to the Owner’s perspective.



	WHAT	HOW	WHERE	WHO	WHEN	WHY
	DATA	FUNCTION	NETWORK	PEOPLE	TIME	MOTIVATION
Designer/ System Model (Logical)	SV-1 SV-3 SV-4 SV-6 OV-7	SV-1 SV-4 SV-5 SV-6 SV-7 SV-10b TV-1	SV-1 SV-2 SV-4	SV-4 CHI*	SV-6 SV-7 SV-8 SV-10c	SV-10a

*CHI - Computer Human Interface

Figure 2.10-3. DoDAF Support to the Designer

2.10.4.1 Logical Data Model (OV-7)

The Logical Data Model (OV-7) product provides architecture information on “data” primitives at the Builder level of detail. The data primitives defined in SV-4 and SV-6 are refined here as a logical model of the data entities with associated attributes.

2.10.4.2 Systems Communications Description (SV-2)

Whereas SV-1 graphically illustrates “network” (systems nodes) primitives and the relationships among them at the Planner, Owner, and Designer levels of detail, the Systems Communications Description (SV-2) graphically illustrates “network” (i.e., communications systems, and communications networks, and links) primitives at the Designer and Builder levels of detail, if the architecture scope includes the communications infrastructure.

2.10.4.3 Systems Functionality Description (SV-4)

The Systems Functionality Description (SV-4) is an architecture composite that graphically illustrates “functions” (system functions) and “data” (details of Systems Data Exchanges) primitives of a given architecture at the Designer and Builder levels of detail.

2.10.4.4 Systems Data Exchange Matrix (SV-6)

The Systems Data Exchange Matrix (SV-6) is an architecture composite that contains details on the architecture’s “data” primitives, here called Systems Data Elements and their attributes grouped under the title Data Exchanges. This matrix provides the Designer’s systems details that implement the operational information exchange requirements specified in OV-3 at the Owner level of detail. The matrix also relates these primitives to “people” (or Systems Nodes), “functions” (system functions of SV-4), and “time” at the Designer and Builder levels of detail.

2.10.4.5 Performance Parameters Matrix (SV-7)

The Performance Parameters Matrix (SV-7) provides characteristics for the “functions” and “time” primitives of the architecture at the Designer and Builder levels of detail. Non-functional systems requirements are specified here.

2.10.4.6 Systems Evolution Description (SV-8)

The Systems Evolution Description (SV-8) provides migration or evolution characteristics of the “functions” (systems or system functions), as they relate to “time” primitives at the Owner level of detail.

2.10.4.7 Systems Rules Model (SV-10a)

The Systems Rules Model (SV-10a) provides “motivation” primitives at the Designer or Builder levels of detail. It provides the constraints for the systems architecture.

2.10.4.8 Systems State Transition Description (SV-10b)

The Systems State Transition Description (SV-10b) describes “functions” primitives at the Designer and Builder levels of detail. It specifies what happens when a certain function is executed or when a certain input is received. It relates the how to the why primitives.

2.10.4.9 Systems Event-Trace Description (SV-10c)

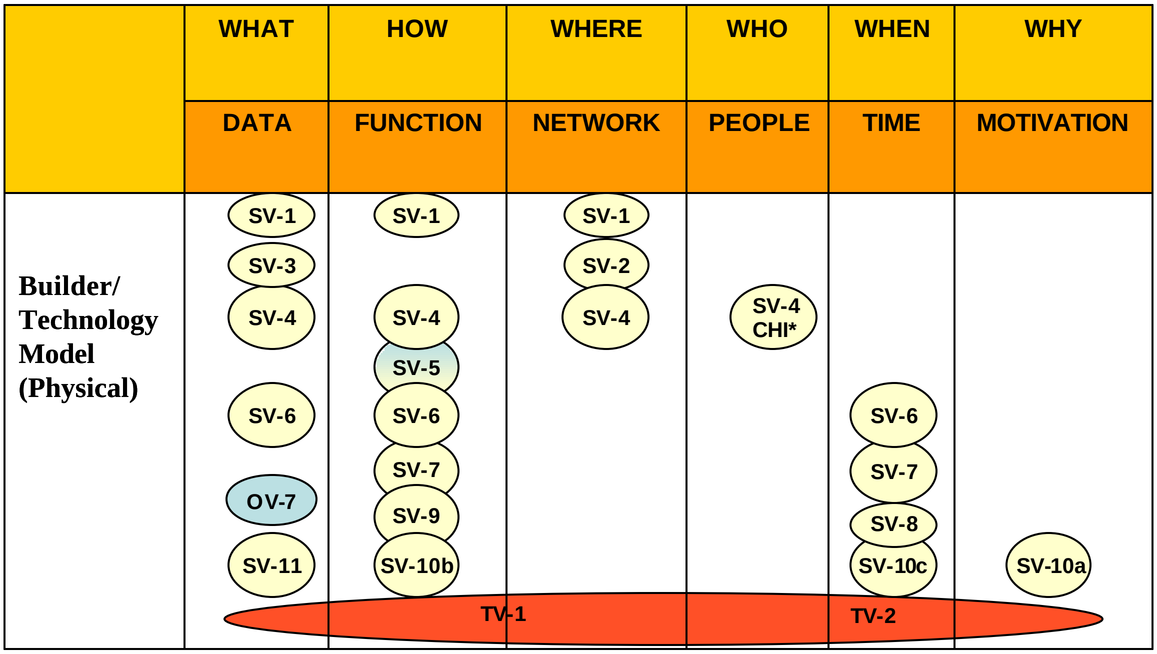
The Systems Event-Trace Description (SV-10c) product describes “time” primitives at the Designer and Builder levels of detail. SV-10c relates the functions from SV-4 to the data in a time-sequenced manner.

2.10.4.10 Technical Standards Profile (TV-1)

The Technical Standards Profile (TV-1) provides technical standards characteristics for “functions” (data formats or system data exchanges) at the Designer, Builder, and Contractor levels of detail.

2.10.5 Builder/Technology

From the Builder’s perspective, details such as the high-level design of the systems and system functions (as presented in the Designer’s perspective) supporting the operational needs and requirements (as presented in the Owner’s perspective) are specified at this level. The SV products developed in the previous perspective are further refined here. **Figure 2.10-4** lists the SV products that add system functionality assignments to the systems present in the Designer’s perspective.



	WHAT	HOW	WHERE	WHO	WHEN	WHY
	DATA	FUNCTION	NETWORK	PEOPLE	TIME	MOTIVATION
Builder/ Technology Model (Physical)	SV-1 SV-3 SV-4 SV-6 OV-7 SV-11	SV-1 SV-4 SV-5 SV-6 SV-7 SV-9 SV-10b	SV-1 SV-2 SV-4	SV-4 CHI*	SV-6 SV-7 SV-8 SV-10c	SV-10a
		TV-1			TV-2	

***CHI - Computer Human Interface**

Figure 2.10-4. DoDAF Support to the Builder

2.10.5.1 Systems Technology Forecast (SV-9)

The Systems Technology Forecast (SV-9) contains predictions about the availability of emerging technological capabilities and about industry trends in specific time periods. It relates “function” primitives to the “time” primitives of the architecture at the Builder level of detail. The focus is on the supporting technologies that may most affect the architecture or its systems. At this level, the emphasis is on specifying which new technological capabilities and which existing systems upgrades (projected for the Subcontractor’s level) may depend on or be driven by the availability of new technology.

2.10.5.2 Physical Schema (SV-11)

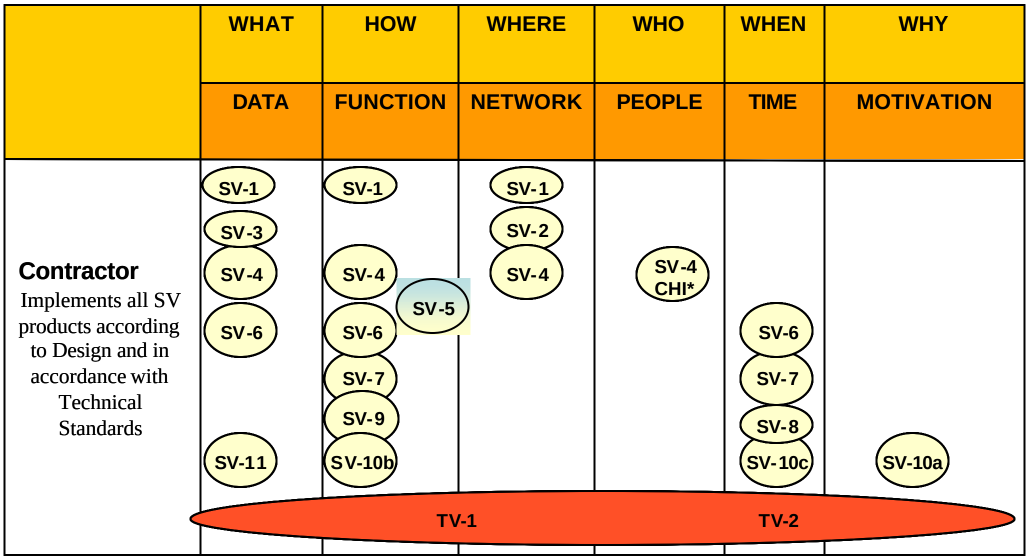
The Physical Schema (SV-11) product provides architecture information on “data” primitives at the Builder and Contractor levels of detail. It provides the implementation detail for the OV-7 specified at the Designer level of detail.

2.10.5.3 Technical Standards Forecast (TV-2)

The Technical Standards Forecast (TV-2) provides technical standards characteristics for “function” (system functions), “data” (data formats or system data exchanges), and “network” (systems—including communications systems, system components—hardware and software, and physical links—including LAN/WAN and network protocols) primitives of the architecture at the Builder and Contractor levels of detail.

2.10.6 Contractor

From the Contractor’s perspective, the systems and their system functions as designed in the system model (Builder’s perspective) are actually implemented at this stage. Further refinements/iterations of the SV products developed for the Designer and Builder perspectives may be needed, as technological and implementation issues arise. Implementation and technology issues/improvements may influence and retroactively cause the design to be modified. These implementation and technological details are reflected back in the SV products developed at the Builder level as appropriate. **Figure 2.10-5** lists the SV and TV products from the Builder’s perspective that are used to implement the systems and functionality at the Contractor level of detail.



	WHAT	HOW	WHERE	WHO	WHEN	WHY
	DATA	FUNCTION	NETWORK	PEOPLE	TIME	MOTIVATION
Contractor Implements all SV products according to Design and in accordance with Technical Standards	SV-1 SV-3 SV-4 SV-6 SV-11	SV-1 SV-4 SV-5 SV-6 SV-7 SV-9 SV-10b	SV-1 SV-2 SV-4	SV-4 CHI*	SV-6 SV-7 SV-8 SV-10c	SV-10a
		TV-1			TV-2	

***CHI - Computer Human Interface**

Figure 2.10-5. DoDAF Support to the Contractor

3 TECHNIQUES FOR USING ARCHITECTURES

3.1 INTRODUCTION

This section presents several sets of analytic techniques using architectures to improve interoperability and to support the Department of Defense (DoD) Planning, Programming, Budgeting and Execution (PPBE) process, Joint Capabilities Integration and Development System (JCIDS), and the Defense Acquisition System. These analytic techniques have been developed within different segments of the DoD community and do not reflect coordinated community positions. The techniques provide valuable insights into using architecture information to impact DoD decision making.

3.2 AIR FORCE CAPABILITY-BASED ANALYSIS

CJCS 3170.01C establishes the JCIDS as a capability-based approach for identifying improvements to existing capabilities and to develop new warfighting capabilities. JCIDS replaces the former Requirements Generation Process. JCIDS utilizes joint concepts and integrated architectures to identify prioritized capability gaps and integrated Doctrine, Organization, Training, Materiel, Leadership & education, Personnel, and Facilities (DOTMLPF) solutions (materiel and nonmateriel).

There is strong interest within many DoD organizations to define and explore capability-based analytical processes. The DoD community has not yet attained a consensus on these concepts. This section presents some of the concepts being considered and introduces a set of architecture-based Capability Reports. The section focuses on the relationships among capabilities, activities, systems, and requirements and defines several Capability Reports[1] that could be associated with an architecture.

This section outlines how integrated architectures containing DOTMLPF information can provide a structured and organized approach for defining capabilities and understanding the underlying requirements for achieving those capabilities. As such, architectures can be used to conduct capability assessments, develop integrated roadmaps for achieving capabilities, and guide the development of systems and associated investment plans.

This section draws heavily from concepts developed within the Air Force as part of their addressal of Air Force Task Forces. It also incorporates material from the Navy's Mission Capability Package (MCP) concept. MCP is more fully addressed in a section 3.3.

3.2.1 Definition of Capability

The term *capability* is the ability to accomplish a particular task, function, or service. Webster's Dictionary defines capability as "the faculty or potential for an indicated use or deployment."

[1] A combination of architecture data elements from one or more products combined with additional information. Reports provide a different way of looking at architecture data.

The Chairman of the Joint Chiefs of Staff Instruction (CJCSI) for JCIDS, CJCSI 3170.01C, defines capability as “the ability to execute a specified course of action.” It is defined by an operational user and expressed in broad operational terms in the format of an Initial Capabilities Document (ICD) or a DOTMLPF change recommendation. In materiel proposals, the definition progressively evolves to DOTMLPF performance attributes identified in the Capability Development Document (CDD) and Capability Production Document (CPD).

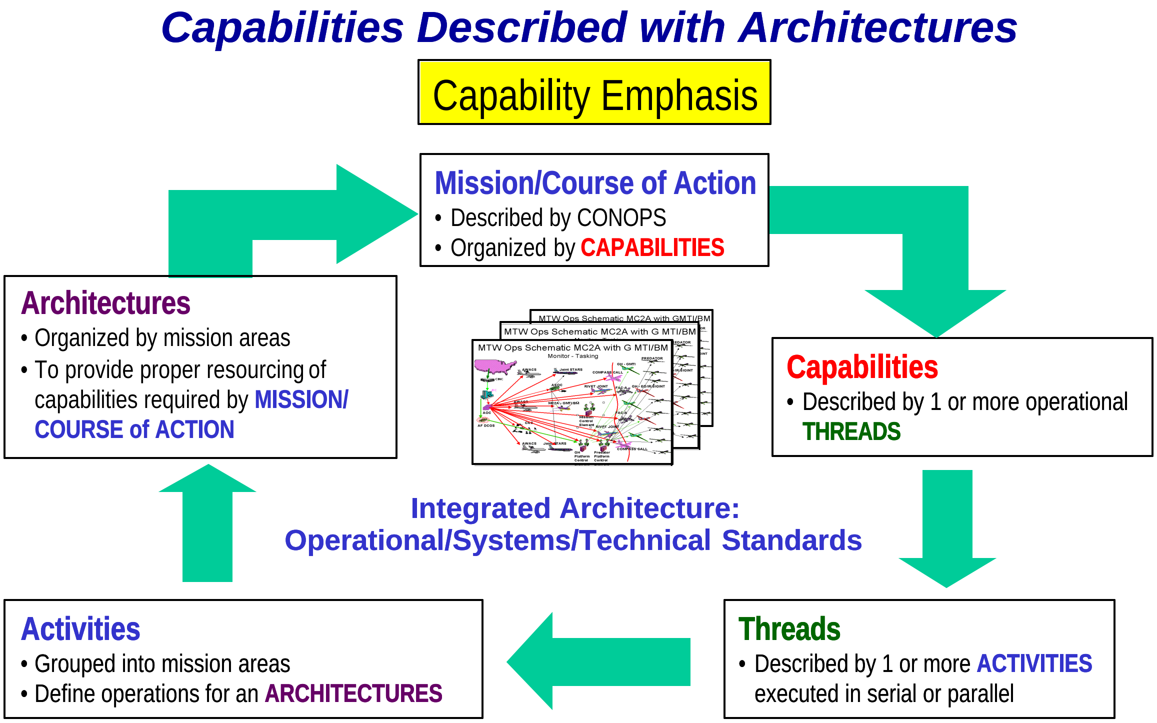
3.2.2 Describing Capabilities

The JCIDS Manual, CJCSM 3170.01, states that definitions of identified capabilities must satisfy two rules:

- Capability definitions must contain the required attributes with appropriate measures of effectiveness (e.g., time, distance, effect [including scale] and obstacles to overcome).
- Capability definitions should be general and not influence a decision in favor of a particular means of implementation. The definition should be specific enough to evaluate alternative approaches to implement the capability.

Capabilities are organized around concepts of operations (CONOPS), because the CONOPS describe how a specified course of action is to be executed. The ability to execute the specified course of action depends on many factors and the relationship between those factors.

Capabilities can be described as one or more sequences of activities, referred to as operational threads. The threads are composed of a set of activities that can be grouped to form the basis for a mission area architecture. The architecture then provides the structure for defining and understanding the many factors that impact the capability. **Figure 3.2-1** illustrates this sequence of relationships.



Capabilities Described with Architectures

Capability Emphasis

Mission/Course of Action
- Described by CONOPS
- Organized by **CAPABILITIES**

Capabilities
- Described by 1 or more operational **THREADS**

Threads
- Described by 1 or more **ACTIVITIES** executed in serial or parallel

Activities
- Grouped into mission areas
- Define operations for an **ARCHITECTURES**

Architectures
- Organized by mission areas
- To provide proper resourcing of capabilities required by **MISSION/ COURSE of ACTION**

Integrated Architecture: Operational/Systems/Technical Standards

MTW Ops Schematic MC2A with GMTI/BM
MTW Ops Schematic MC2A with G MTI/BM
MTW Ops Schematic MC2A with G MTI/BM

Figure 3.2-1. Describing Capabilities with Architectures

The Navy has also endorsed using architectures to understand and analyze capabilities and their associated requirements. The Navy performs this architecture analysis based on the concept of MCPs. The intent is to consider all of the factors that contribute to the desired mission capability as an integrated system. An MCP is defined as “a task-oriented bundle of CONOPS, processes, and organization structures supported by networks, sensors, weapons, and systems, as well as personnel training and support services to sustain a core naval capability.” The MCP and associated analysis then provide the basis for acquisition decisions.

3.2.3 Composite Nature of Capability

Capabilities can be decomposed into sub-capabilities. Increased resolution achieved through decomposition creates inter-relationships. Integrated architectures that include DOTMLPF can provide the context for understanding this decomposition and its associated interrelationships. Operationa l threads expressed as sequenced sets of activities are like action sequences and provide a basis for defining and understanding the various sub-capabilities. Understanding the relationship between the various capabilities allows for system assignments to be reused in multiple mission areas. This construct facilitates identification of common requirements/capabilities that can be optimized in order to avoid stovepiped system implementation with a performance impact that is less than optimal.

3.2.4 Capability Reports

The Air Force has proposed a set of reports, listed in **Table 3.2-1** that uses architecture information to achieve a more in-depth understanding of capabilities and associated requirements. The reports provide an understanding of those aspects of integrated architectures that most impact capabilities and the manner and extent of that impact. The foundation of the report suite is the relationships between capabilities, requirements, activities and systems (see **Figure 3.2-2**). The report suite is intended to assist senior decision makers in planning, programming, and acquisition. Reports with an asterisk are discussed further in this section.

Table 3.2-1. Architecture Reports for Capability Analysis

Architecture Reports for Capability Analysis	
CR-1*	**Prioritized Capability List** – References: Strategic Plan, CONOPS, CDD, CPD, Task List, Capability Decision Packages
CR-2*	**Capability to Requirements and/or Tasks Matrix** – Maps capabilities to applicable requirements and/or tasks and activities
CR-3	**Operational Profile** – Content: mission objectives, threat situation, physical environment, U.S. & Allied systems, design reference mission, similar to CONOPS or scenario based, OPLAN
CR-4	**Capability Metrics Description** – Used to describe metrics for evaluating capabilities
CR-5*	**Capability to Systems/Programs Traceability Matrix** – Key product that leverages applicable Operational and System Views
CR-6*	**Capability Evolution Description** – Identifies when capabilities will be achieved; supports funding decisions
CR-7*	**Integrated Capability Analysis Summary** – Key end product that presents decision makers with results of analysis

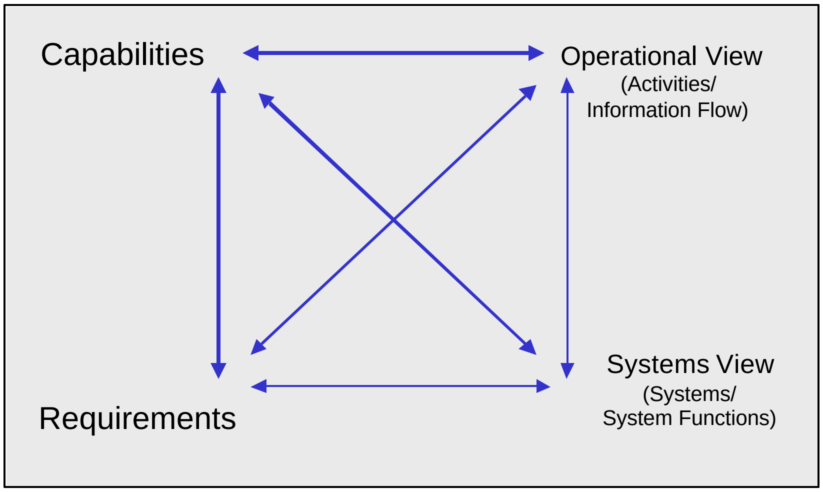


Figure 3.2-2. Basic Capability Relationships

3.2.5 Prioritized Capability List (CR-1)

The Prioritized Capability List (CR-1) assigns priorities to capabilities. It draws on related strategic plans, CONOPS, CDD, and capability decision packages. The capability list should include subcategories of capabilities and define their relation to the higher capability.

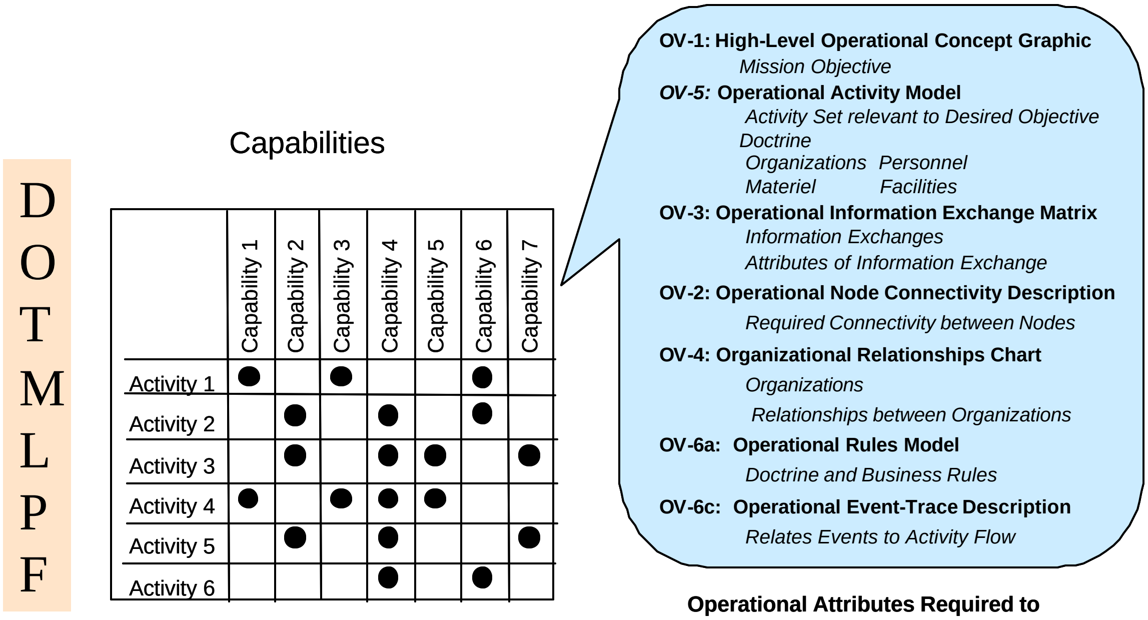
3.2.6 Capability to Requirements and/or Tasks Matrix (CR-2)

The Capabilities to Requirements and/or Tasks Matrix (CR-2) can be used to map capabilities to requirements and/or capabilities to activities. An activity set provides the basis for understanding the relationship between the DOTMLPF and other factors that influence achieving the capability.

As stated in the USPACOM Information Capabilities Framework, “Requirements and capabilities are two sides of the same coin.” A requirement is a capability that is needed but not yet fully delivered or sustained.

A set of activities and the associated information flow provide a foundation for understanding and describing the required operational capability and the various attributes that impact on that capability.

Figure 3.2-3 depicts the mapping of capabilities to activities and illustrates how Operational View (OV) reports and DOTMLPF describe and relate the many attributes that contribute to the definition of the capability. **Figure 3.2-4** illustrates attributes within DOTMLPF that can impact capability.



Capabilities

	Capability 1	Capability 2	Capability 3	Capability 4	Capability 5	Capability 6	Capability 7
Activity 1	●		●			●	
Activity 2		●		●		●	
Activity 3		●		●	●		●
Activity 4	●		●	●	●		
Activity 5		●		●			●
Activity 6				●		●	

OV-1: High-Level Operational Concept Graphic
Mission Objective
OV-5: **Operational Activity Model**
Activity Set relevant to Desired Objective
Doctrine
Organizations Personnel
Materiel Facilities
OV-3: Operational Information Exchange Matrix
Information Exchanges
Attributes of Information Exchange
OV-2: Operational Node Connectivity Description
Required Connectivity between Nodes
OV-4: Organizational Relationships Chart
Organizations
Relationships between Organizations
OV-6a: Operational Rules Model
Doctrine and Business Rules
OV-6c: Operational Event-Trace Description
Relates Events to Activity Flow

Operational Attributes Required to Achieve Capability

Figure 3.2-3. Relating Capabilities and Activities

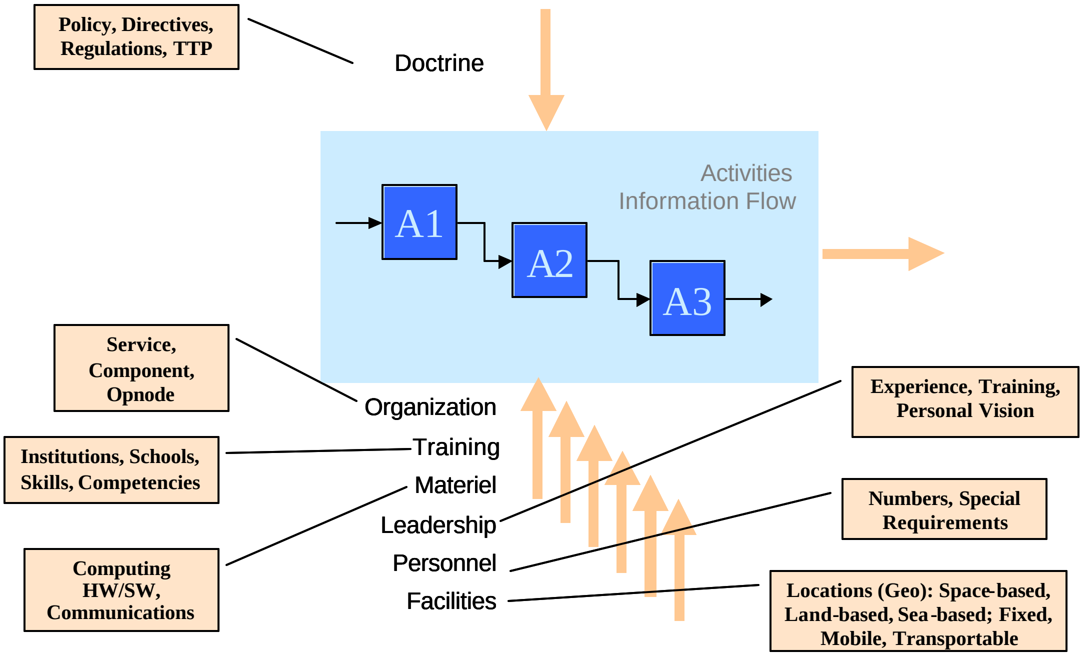

Policy, Directives, Regulations, TTP
Doctrine
Activities
Information Flow
A1
A2
A3
Service, Component, Opnode
Organization
Institutions, Schools, Skills, Competencies
Training
Materiel
Leadership
Computing HW/SW, Communications
Personnel
Facilities
Experience, Training, Personal Vision
Numbers, Special Requirements
Locations (Geo): Space-based, Land-based, Sea-based; Fixed, Mobile, Transportable

Figure 3.2-4. DOTMLPF: Factors Affecting Capability

DOTMLPF, while necessary, is by itself insufficient for defining the required attributes for achieving a capability and understanding the relationship between those required attributes. An integrated architecture, combined with DOTMLPF, can provide that definition and understanding. **Figure 3.2-5** illustrates capability attributes contained in various OV products.

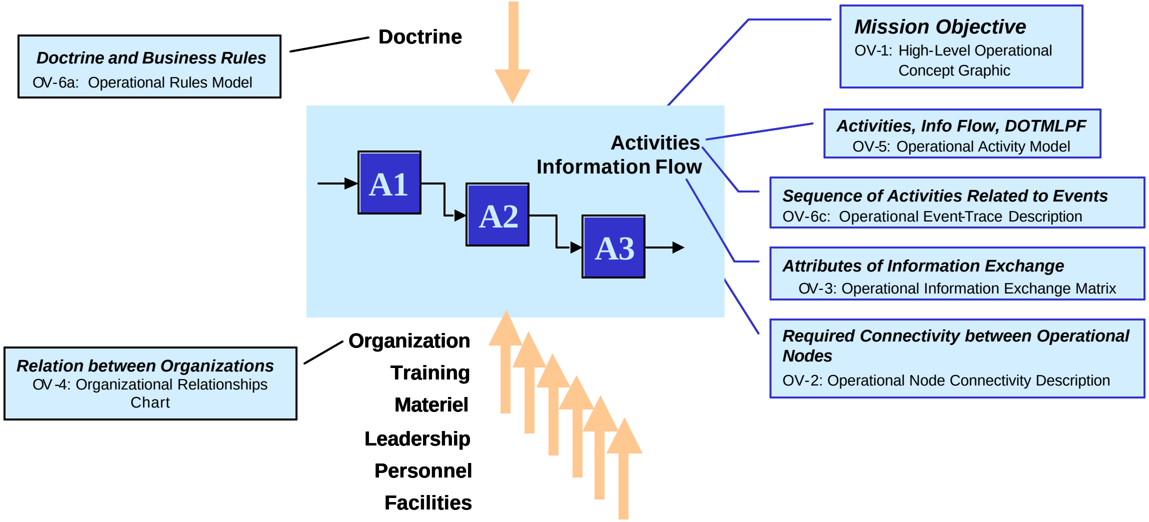

Doctrine and Business Rules
OV-6a: Operational Rules Model
Doctrine
Mission Objective
OV-1: High-Level Operational Concept Graphic
Activities, Info Flow, DOTMLPF
OV-5: Operational Activity Model
Activities
Information Flow
A1
A2
A3
Sequence of Activities Related to Events
OV-6c: Operational Event-Trace Description
Attributes of Information Exchange
OV-3: Operational Information Exchange Matrix
Required Connectivity between Operational Nodes
OV-2: Operational Node Connectivity Description
Relation between Organizations
OV-4: Organizational Relationships Chart
Organization
Training
Materiel
Leadership
Personnel
Facilities

Figure 3.2-5. Operational Attributes Required to Achieve Capability

3.2.7 Capability to Systems/Programs Traceability Matrix (CR-5)

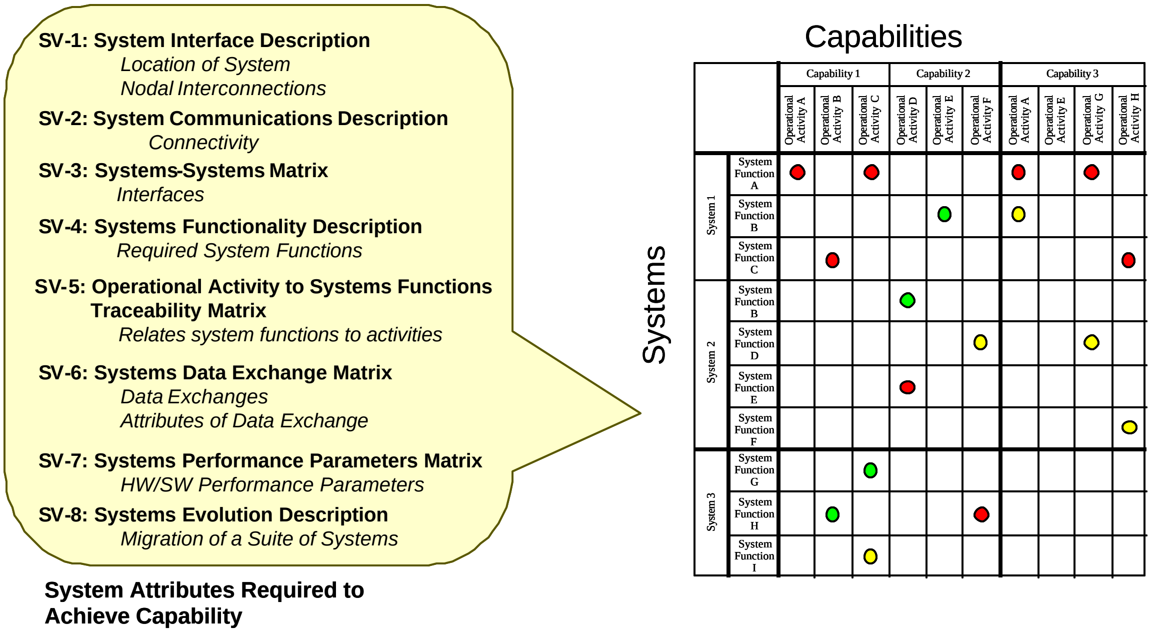
The relationship between capabilities and systems is generally of high interest, because it drives acquisition and associated budgets (see **Figure 3.2-6**). The Capability to Systems/Programs Traceability Matrix (CR-5) is a key report that leverages information in the integrated architecture in order to understand the relation between capabilities and systems.

The relationship between capabilities and systems is defined based on the association between capabilities and activities, the association of activities to system functions (defined in the Operational Activity/Systems Function Traceability Matrix [SV-5]), and the association of system functions to systems. System functions may be associated with systems as part of the Systems Interface Description (SV-1). Systems may also be related to activities that include systems as a mechanism in the Operational Activity Model (OV-5).

		Capability 1			Capability 2			Capability 3			
		Operational Activity A	Operational Activity B	Operational Activity C	Operational Activity D	Operational Activity E	Operational Activity F	Operational Activity A	Operational Activity E	Operational Activity G	Operational Activity H
System 1	System Function A	🔴		🔴				🔴		🔴	
	System Function B					🟢		🟡			
	System Function C		🔴								🔴
System 2	System Function B				🟢						
	System Function D						🟡			🟡	
	System Function E				🔴						
	System Function F										🟡
System 3	System Function G			🟢							
	System Function H		🟢				🔴				
	System Function I			🟡							

Figure 3.2-6. Relating Systems to Capabilities

In order for a system to contribute to a given capability, the system must possess certain specific attributes. **Figure 3.2-7** illustrates how Systems View (SV) products describe and relate the attributes of the systems that contribute to the various capabilities.



SV-1: System Interface Description
Location of System
Nodal Interconnections

SV-2: System Communications Description
Connectivity

SV-3: Systems-Systems Matrix
Interfaces

SV-4: Systems Functionality Description
Required System Functions

SV-5: Operational Activity to Systems Functions Traceability Matrix
Relates system functions to activities

SV-6: Systems Data Exchange Matrix
Data Exchanges
Attributes of Data Exchange

SV-7: Systems Performance Parameters Matrix
HW/SW Performance Parameters

SV-8: Systems Evolution Description
Migration of a Suite of Systems

System Attributes Required to Achieve Capability

Capabilities

Systems		Capability 1			Capability 2			Capability 3			
		Operational Activity A	Operational Activity B	Operational Activity C	Operational Activity D	Operational Activity E	Operational Activity F	Operational Activity A	Operational Activity E	Operational Activity G	Operational Activity H
System 1	System Function A	🔴		🔴				🔴		🔴	
	System Function B					🟢		🟡			
	System Function C		🔴								🔴
System 2	System Function B				🟢						
	System Function D						🟡			🟡	
	System Function E				🔴						
	System Function F										🟡
System 3	System Function G			🟢							
	System Function H		🟢				🔴				
	System Function I			🟡							

Figure 3.2-7. System Attributes Supporting Capabilities

Each mapping between an operational activity to a system function is described by a stoplight colored circle to indicate the status of the system support. Red indicates functionality planned but not developed. Yellow indicates either partial or full functionality provided, but the system has not been fielded. Green indicates full functionality provided and system fielded. A blank cell indicates that there is no system support planned for an operational activity, or that a relationship does not exist between the operational activity and the system function. In this manner, the association between a certain capability and a specific system can be illustrated via a many-to-many relationship: *many* operational activities contribute to a capability, and *many* system functions are executed by a system.

Once the relationship between systems and capabilities is understood, the Requirements Traceability Matrix can be developed based on those system attributes necessary to achieve the desired capability but currently unavailable.

3.2.8 Capability Evolution Description (CR-6)

The Capability Evolution Description (CED) (CR-6) is a description of programs, platforms, and systems aligned to capability objectives and increments over time. The CED supports achieving capability objectives by facilitating program alignment. The report is based on an analysis of the:

- Dependencies between capabilities and systems
- Relation between those systems and requirements
- Relation between the requirements and acquisition programs

Figure 3.2-8 is an illustrative CED provided by the Navy. The following Navy definitions apply:

- **Mission Capability** is the possession of the means to use military force to achieve an intended effect with the battlespace that can be measured.
- **Capability Objective** is a capability or related set of capabilities with decisive and attainable goals toward achieving a Mission Capability.
- **Capability Component** is a major element of a capability objective that the owning service wants to measure and assess.
- **Capability Increment** is a bundling of networks, sensors, weapons, and platforms aligned in acquisition over a determined time that enables a capability objective.

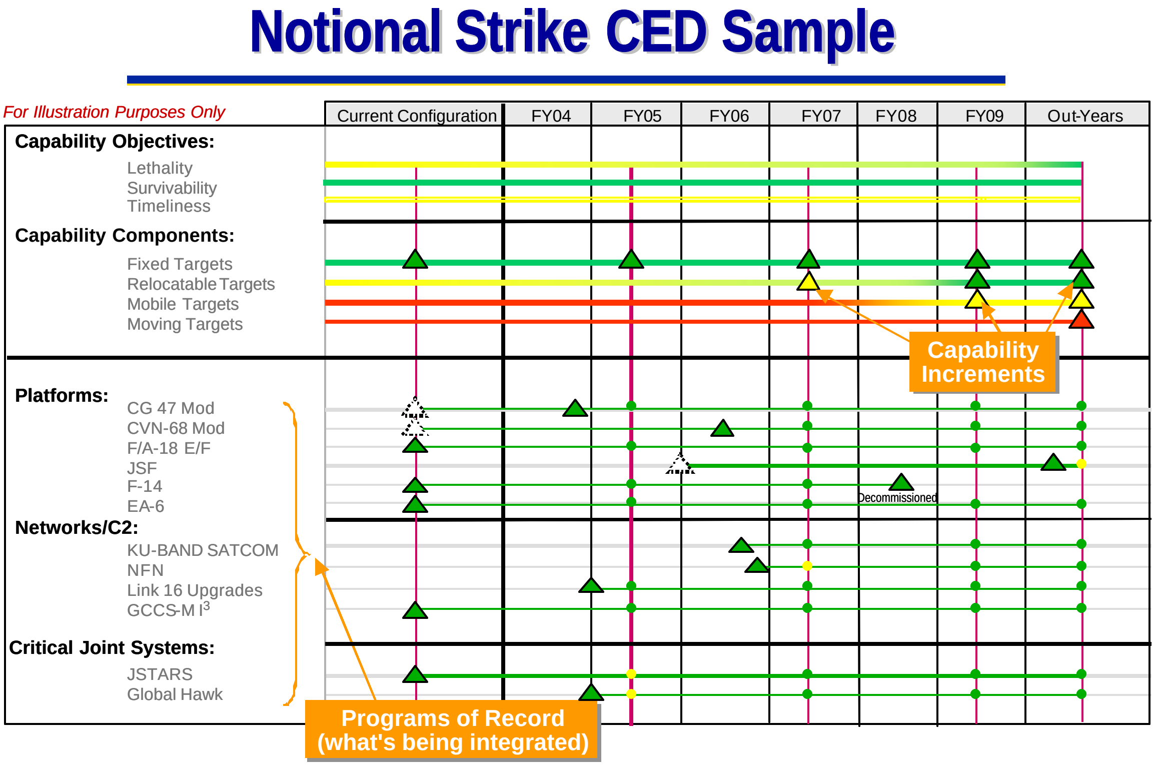


Figure 3.2-8. Illustrative Capability Evolution Description

Figure 3.2-8 is an illustrative example from the Navy's Research, Development, and Acquisition Chief Engineer. It addresses capabilities of lethality, survivability, and timeliness against fixed, relocatable, mobile, and moving targets. For the platforms and systems in the lower half of the diagram, the green triangle denotes the point at which the platforms or systems have the required attributes for its contribution toward achieving the capabilities.

In odd-numbered years, the capability increment is noted against the capability components that it impacts and the overall degree of achievement of the capability objective. For example, the attributes of the platforms and systems in their current configuration satisfy the needs for fixed targets. The required capability for relocatable targets starts to move from yellow to green with the FY07 capability increment and fully becomes green with the FY09 capability increment. Mobile targets are red but become yellow with the FY09 capability increment, while moving targets continue to be red throughout the time period. This indicates that the levels of the required capability objectives of lethality, survivability, and timeliness continue to be insufficient for moving targets at the end of the time period.

The color of the capability objective line indicates the degree to which the attribute is achieved against the capability components. Survivability is at an acceptable rate throughout the time period, while lethality is at a less than desired level but achieves the desired level (turning green) at the end of the time period. Timeliness continues to be less than desired during the time period.

3.2.9 Integrated Capability Analysis Summary (CR-7)

The Integrated Capability Analysis Summary (CR-7) is the end report of the capability-based analysis. It pulls from and builds on the CR-1 through CR-6 to provide decision makers with insights into the key issues relating to achieving the stated capabilities. These issues can include critical new systems or critical system enhancements, activities in critical status because of lack of either operational attributes or system attributes related to the target capability, and requirements that are being satisfied in a timely manner to achieve target capabilities.

3.2.10 References

Thilenius, Jim (MITRE), Air Force Chief Architect Office, *Proposal for New Capability View of DoD Architectures*, Briefing, June 6, 2002.

Thilenius, Jim (MITRE), Air Force Chief Architect Office, *Using Architectures to Support Air Force Core Processes,* Briefing, November 15, 2002.

CJCSI 3170.01C, *Joint Capabilities Integration and Development System*, June 24, 2003.

CJCSM 3170.01, *Joint Capabilities Integration and Development System Manual*, June 24, 2003.

USPACOM, *Information Capabilities Framework*, February 25, 2002.

3.3 NAVY'S MISSION CAPABILITY PACKAGE APPROACH

3.3.1 Introduction

The Navy has developed the concept of Mission Capability Packages (MCPs) as a mechanism for understanding and assessing capability needs, defining capability requirements, and planning fo r future warfighting. The number of systems and the complex relationships between users, systems, and developers lead to various and frequently non-interoperable approaches for satisfying operational requirements. Architectures offer a means of standardizing and organizing the many forms of information created or used to develop these systems. As part of their work with MCPs, the Navy has developed analytical techniques based on architectures to:

- Assess system functionality (1st Order Analysis)
- Examine system connectivity (2nd Order Analysis)
- Assess architecture performance and behavior (3rd Order Analysis)
- Align the evolution of systems technologies and standards into an acquisition strategy

The techniques presented in this section originated as part of the Navy's development of the Naval Time Critical Targeting (TCT) architecture. The TCT architecture was a collaborative effort of the Chief of Naval Operations Warfare Requirements Division (CNO N70), the Commander, Navy Warfare Development Command (NWDC), the Naval Targeting Afloat Integrated Report Team (NTA IRT), and the Assistant Secretary of the Navy for Research, Development and Acquisition (ASN [RDA]) Chief Engineer (RDA CHENG). The objective of the Integrated Naval Targeting Architecture (INTA) Team was to produce a TCT mission area architecture (Operational, Systems, and Technical Standards Views) supporting the N70 TCT Mission Capability Package for the Program Objective Memorandum (POM) 2004 build. As part of the assessment of the TCT architecture, the INTA team was tasked to identify key performance and systems integration solutions for TCT that would impact mission capability and evaluate the possible acquisition of those solutions for the fleet. The techniques developed by the INTA for conducting architecture-based assessments to support MCPs have general applicability and lend themselves to use in other mission areas.

This section provides a description of MCPs followed by descriptions of the associated analytical techniques. While these techniques were developed to support the MCP concept, they are applicable to a wide range of architectures where there is interest in identifying duplications and gaps in system functionality and interoperability issues.

This section is based on briefings and documents developed by the RDA CHENG in 2002. In August 2003, RDA CHENG published a more extensive addressal of this topic in *Using Architectures for Research, Development, and Acquisition*, undated.

3.3.2 Mission Capability Package

The Navy defines MCPs as a task-organized bundle of:

- Processes with associated CONOPS and organizational structures
- Systems to include networks, sensors, weapons, and information technology (IT) systems
- People, training, and support services to sustain the processes and systems

The above are treated not as a collection of things and processes but as an integrated system.

The MCP addresses a multi-system/platform capability from the operational/tactical Commander's perspective (not as a mission area or business unit). The MCP represents "a task organized approach to program planning." It is founded on the users' current and proposed concept of operations (CONOPS), tasks, and information exchange requirements and the resulting Operational View (OV).

3.3.3 Architecture Analysis Approach

3.3.3.1 Overview

Three levels of analysis are used to assess the architecture, and an acquisition strategy is developed. **Figure 3.3-1** depicts the use of architecture to identify duplications and gaps in system capabilities. While the Navy focused their use of the analytical process in support of the POM, the results can also provide recommendations into the requirements and acquisition processes.

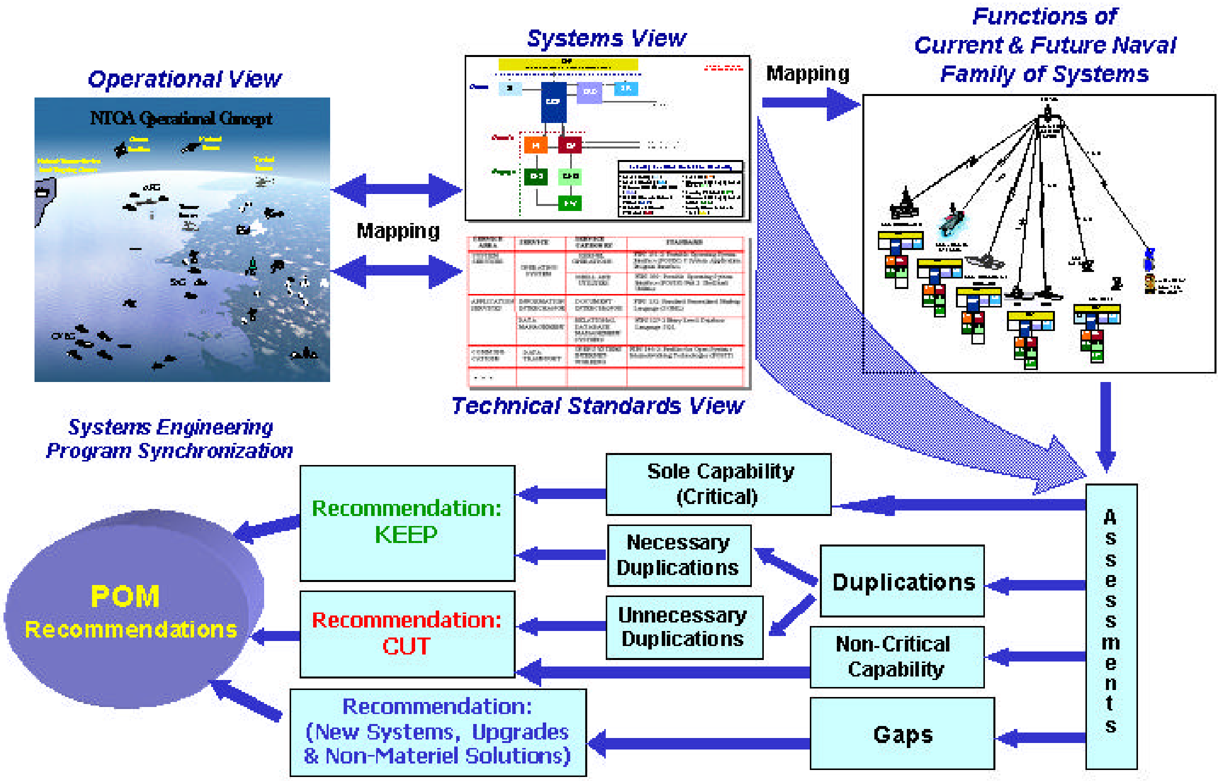


Figure 3.3-1. Using Architectures for Analysis (Navy's RDA CHENG)

3.3.3.2 1st Order Analysis: System Functionality

The objectives of 1st Order Analysis are to determine:

- Does the family of system's (FoS's) system architecture provide the functionality to support the desired mission capabilities?
- Are the systems correct?
- Are there gaps or duplications in system functionality?

To achieve these objectives, the relation of activities to systems to system functions is defined and analyzed.

Phase 1: Concept Development

The operational concept is a high-level abstraction of the mission to be accomplished and the proposed approach for accomplishing the mission. A mission may be a military mission or a business process.

Þ **Start with**: Three Framework products provide a basis for understanding the operational concept. These products lay the foundation for systems development and facilitate communication by providing context, orientation, and focus. They also serve as the entry point for requirements flowdown into the architecture. The three products are:

- **High-Level Operational Concept Graphic (OV-1)** provides a high-level description of what the military force is and its intended effects on the defined threat. It should also establish the boundaries of the battlespace and the uses of the military force to achieve effects. For business processes, OV-1 provides a graphical high-level overview of the business process, the entities that participate in that process, and how they participate. OV-1 may also be used to depict an evolution of capability increments that lead to full capability.
- **Operational Activity Model (OV-5)** describes the activities through which each military mission or business process is accomplished. These activities, along with the input and output of information between them, form the activity model (e.g., an IDEF0). However, the activity model does not establish order of execution or timing relations among the activities.
- **Organizational Relationships Chart (OV-4)** documents the command relations over the operational activities, establishing by what organizational authority activities are directed to execute.

Þ **Supplement the above with**: Other documents that describe operational concepts help to gain additional understanding of the flow of the activities. The documents include:

- **Concept of Operations (CONOPS)** consist of high-level approved scenarios with supporting Operations Plans that detail how forces/organizations will conduct operations for purposes of analyzing capabilities.

- **Operational Situation (OpSit) Descriptions**, for analysis purposes, provide a description of the situation and conditions being applied under a particular scenario to define the military objectives within a mission area.
- **Tactics, Techniques, and Procedures (TTP)** are the actions and methods that implement doctrine and describe how forces will be employed in operations.

Þ **Develop**: an Activity Flow Diagram (**Figure 3.3-2**) using the supplementary documents above to assist in tailoring the Operational Activity Model (OV-5). The Activity Flow Diagram can be used to examine activity sequencing and execution, identify parallel activities, and as a basis to evaluate timeline issues. This Activity Flow Diagram is a tailored **OV-6c**: **Operational Event-Trace Description**, where the activities are used as states.

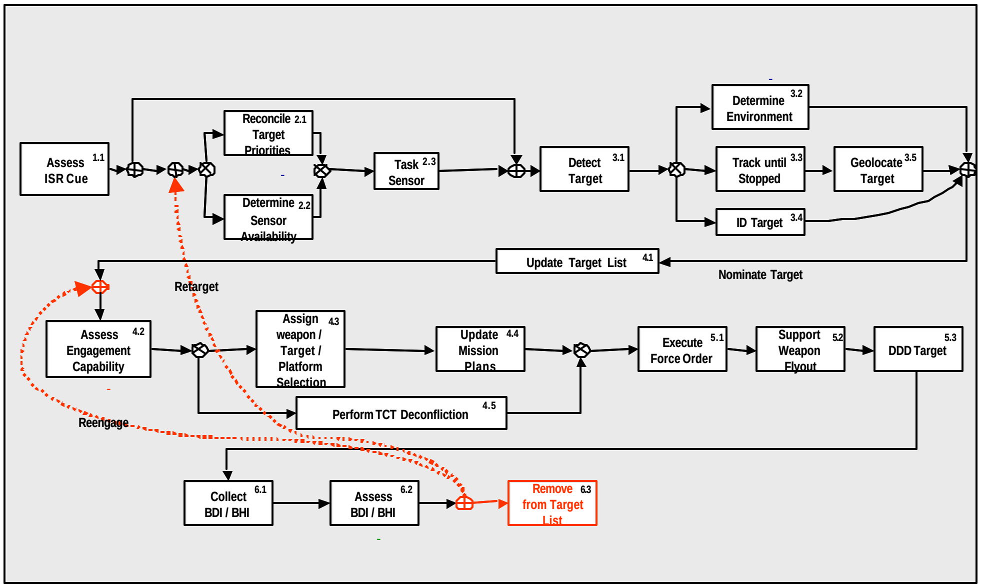


Figure 3.3-2. Example Activity Flow Diagram for Time-Critical Targeting
(Original from Navy's RDA CHENG)

Phase 2: System Functional Mapping

Due to the complexity of FoS, bookkeeping the data describing the systems, their relationships, and evolution can be an overwhelming task. The system functional mapping provides a stable model for depicting the FoS and is easier to manage and assess. The functional mapping is the first level of the analysis that supports systems assessments. Assessments using this functional group of products provide the basis for a 1st Order Analysis of combinations of systems proposed to comprise the FoS. In the systems engineering process, attention will be focused on a FoS that is intended to solve the problems laid out in the OV-1. For example, an analysis of gaps and duplications reduces the size of the system trade space. The result of the first order architecture analysis is the starting point for systems engineering tradeoff analysis.

Þ **Use**: Three Framework products provide the basis for the system functional mapping. Building on information from Phase 1, these products provide the linkage and traceability of capabilities and requirements flowdown between the Operational View and the

Systems View (SV). **Figure 3.3-3** depicts activity to system function to system relationships and source products providing those mappings.

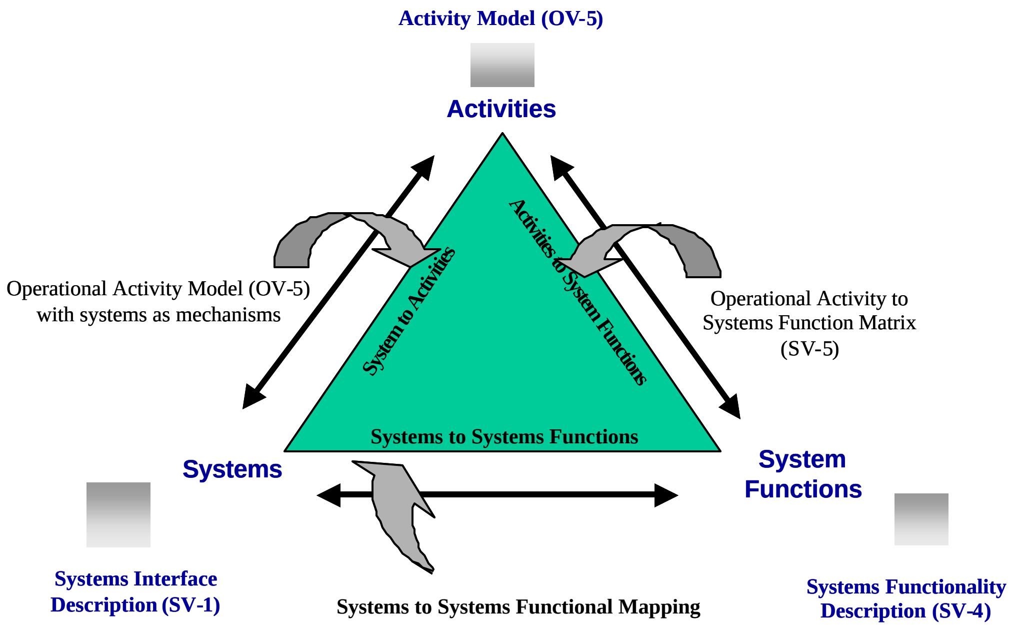

Activity Model (OV-5)
Activities
Activities to System Functions
System to Activities
Operational Activity Model (OV-5) with systems as mechanisms
Operational Activity to Systems Function Matrix (SV-5)
Systems to Systems Functions
Systems
System Functions
Systems Interface Description (SV-1)
Systems to Systems Functional Mapping
Systems Functionality Description (SV-4)

Figure 3.3-3. Mapping Activities to Systems to Systems Functions

- **Systems Functionality Description (SV-4)** is the list of the system functions that are used to enable or execute the operational activities. The SV-4 used at this stage is the variation that depicts the system (application) functions using hierarchical decomposition. The decomposition should allow description of the application's functions at whatever level of detail is required. The level of detail will emerge in the course of the analysis, so that initial characterizations can be fairly high level. As the analysis proceeds, details can be added in specific areas of interest.
- **Systems to System Functions Mapping** is developed using data elements associated with the Systems Interface Description (SV-1) to map systems to system functions.
- **Operational Activity to Systems Function Traceability Matrix (SV-5)** summarizes individual system functions that are used to enable or execute individual operational activities. Each cell in the matrix points to a use case of the system functions. Using the system functions, SV-5 provides the traceability of operational capabilities into the FoS.

Phase 3: **Analyses**

Þ **Develop Bar Chart**: Using the Systems to System Functions Mapping, determine the number of systems supporting each function and depict on a bar chart (**Figure 3.3-4**). Identify the functions supported by the most and least number of systems.

Þ **Overlay the Activity Flow Diagram with the supporting systems**: **Figure 3.3-5** depicts supporting systems related to activity flow.

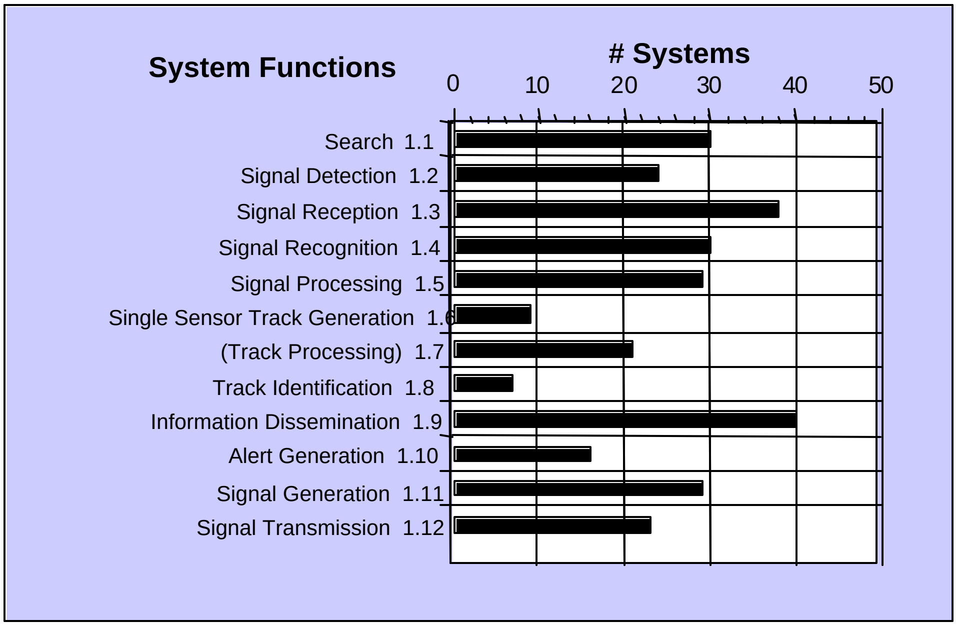


Figure 3.3-4. Bar Chart of Number of Systems Supporting Each Function
(Extracted from Navy's RDA CHENG Graphic)

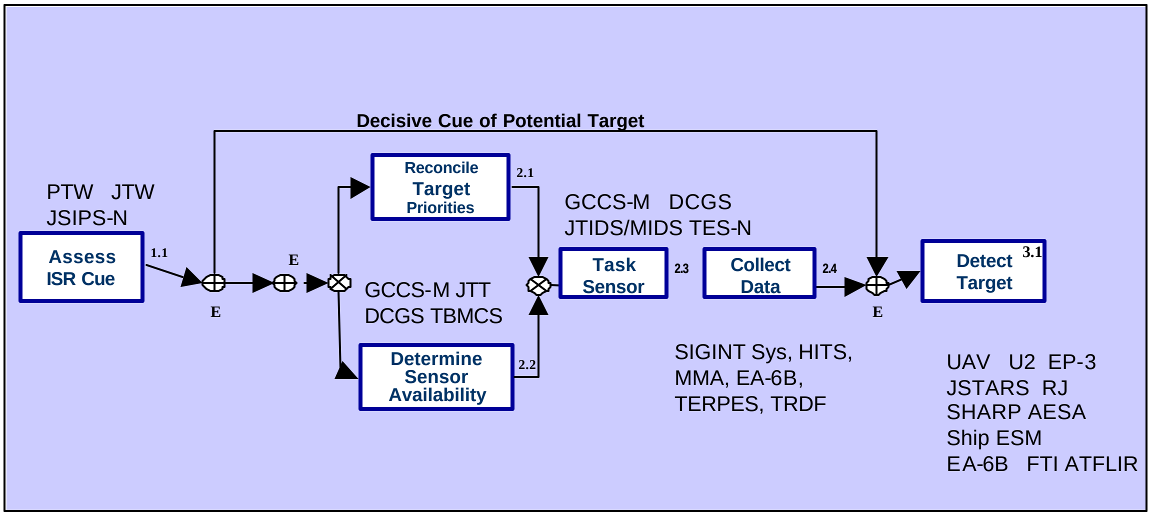


Figure 3.3-5. Example Activity Flow Diagram with Systems Mapping
(Extracted from Navy's RDA CHENG Graphic)

Þ **Identify duplications in system functions**: For each activity that is supported by more than one system, compare the functions of the multiple systems that support a given activity. Identify duplications in system functions supporting specific activities. Complete the above for all activities to identify systems that appear to have significant duplications in system functions.

Þ **Identify gaps in functionality**: Examine activities that require a function not provided by supporting systems.

Þ **Assess individual systems**: Building on the identification of system functionality duplication and gaps, assess individual systems in terms of functional utility, usability, supportability, interoperability, integration and performance attributes, and cost.

Þ **Identify duplication in systems**: Based on the above, identify potential duplications in systems.

Results: The analyses above can be used as the basis for recommending functional consolidation and program elimination. Identifying additional required functionality for a specific system can also be an output. This knowledge can assist in decision making to support the programmatic recommendations as the Navy is doing. The knowledge may also be used to support requirements definition and acquisition recommendations.

Figure 3.3-6 depicts the logical process for 1st Order Analysis.

3.3.3.3 2nd Order Analysis: System Connectivity

The objectives of 2nd Order Analysis are to build on the system functionality mapping of the 1st Order Analysis to determine:

- Are the system connectivity and data content at the interfaces correct?
- Are the logical interfaces correctly connected?
- Are the systems correctly connected?
- Have the appropriate standards been applied?
- Are the levels of interoperability properly aligned so that individual systems in the FoS can be expected to interoperate with each other successfully to enable the required functionality?

In 2nd Order Analysis, the focus is on System Interface Mapping—considering both physical and logical interfaces. The analysis builds on information from all three core architecture views—Operational, System, and Technical Standards. The physical domain is related to the information flow and operational activity sequence, and standards at interfaces are examined.

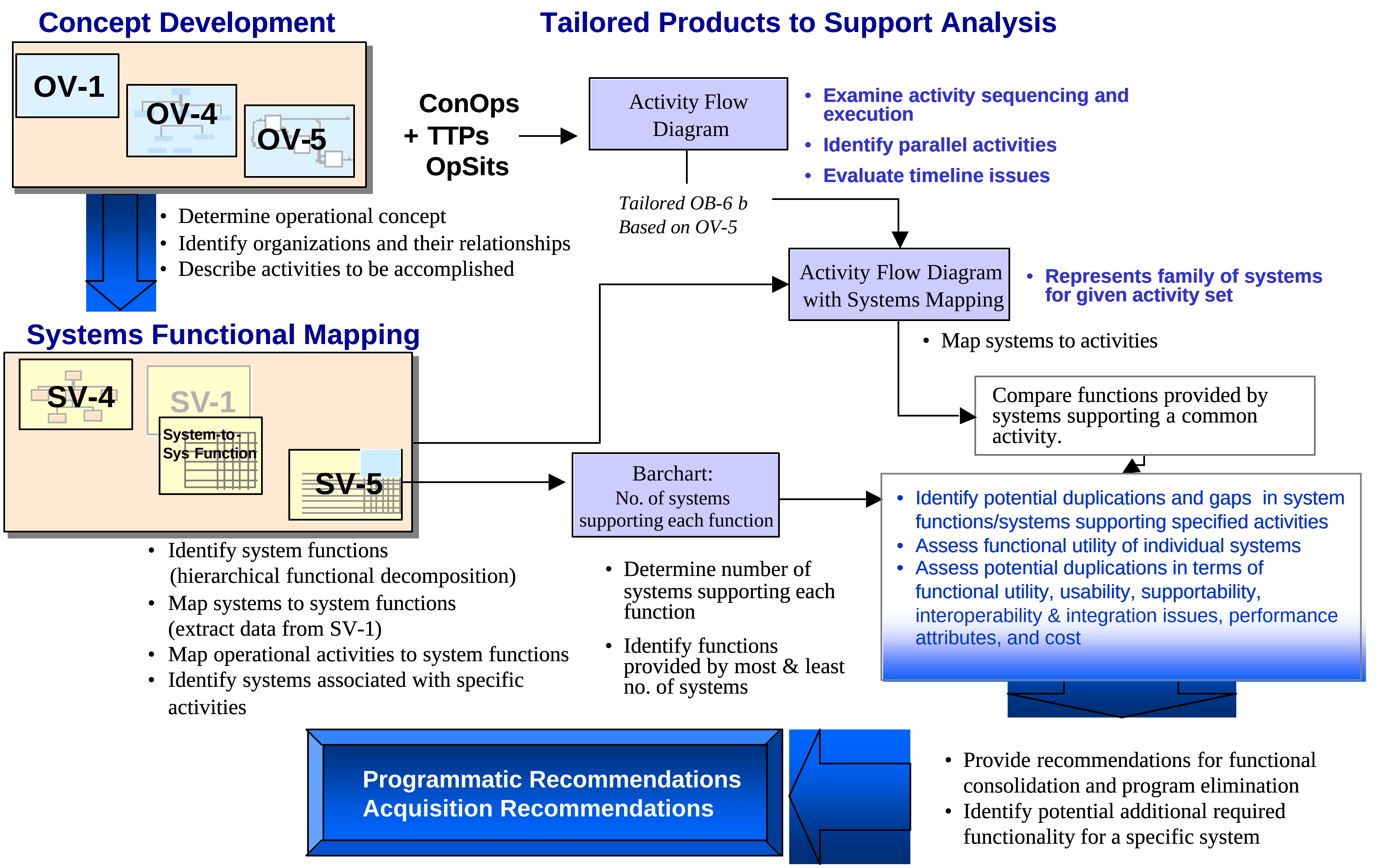
Phase 1: Data Organization

Þ **Identify** entities, information flows, and standards. The 1st Order Analysis has identified activities, systems, and system functions. When combined with the content of the products below, all the core architecture information for relating activities to systems to standards is present. The following products contribute data to this phase:

- **Operational Node Connectivity Description (OV-2)** identifies the operational nodes and information flow needlines between nodes. The nodes can be thought of as task-oriented cells where work is accomplished. Because the activities of OV-5 carry input and output relations, the nodes of OV-2 inherit these relations, which are referred to as needlines. Needlines are not the communications paths.

1st Order Analysis

- Focus on System Functionality, Duplication, and Gaps
- Quick-Look Analysis of Systems and Operational Drivers (Systems, Functions, Nodes)



Concept Development

- Determine operational concept
- Identify organizations and their relationships
- Describe activities to be accomplished

Tailored Products to Support Analysis

ConOps + TTPs OpSits

Activity Flow Diagram

- Examine activity sequencing and execution
- Identify parallel activities
- Evaluate timeline issues

Tailored OB-6 b Based on OV-5

Activity Flow Diagram with Systems Mapping

- Represents family of systems for given activity set

- Map systems to activities

Compare functions provided by systems supporting a common activity.

Systems Functional Mapping

- Identify system functions (hierarchical functional decomposition)
- Map systems to system functions (extract data from SV-1)
- Map operational activities to system functions
- Identify systems associated with specific activities

Barchart: No. of systems supporting each function

- Determine number of systems supporting each function
- Identify functions provided by most & least no. of systems

- Identify potential duplications and gaps in system functions/systems supporting specified activities
- Assess functional utility of individual systems
- Assess potential duplications in terms of functional utility, usability, supportability, interoperability & integration issues, performance attributes, and cost

Programmatic Recommendations
Acquisition Recommendations

- Provide recommendations for functional consolidation and program elimination
- Identify potential additional required functionality for a specific system

Figure 3.3-6. 1st Order Analysis

- **Operational Information Exchange Matrix (OV-3)** identifies the information flow between activities occurring at specific nodes. In doing this, OV-3 relates three entities (activities, operational nodes, and information flow) of the Operational View of the architecture with a focus on the specific aspects of information flow. This flow identifies who exchanges what information with whom, why the information is necessary, and in what manner.
- **Systems Interface Description (SV-1)** identifies systems nodes and interfaces. SV-1 provides a connection between the Operational View and Systems View by mapping systems and their interfaces to the nodes and needlines described in OV-2. SV-1 depicts the systems nodes, the systems at those nodes, and the links among them. While OV-2 depicts operational nodes, the systems nodes of SV-1 are facilities where the system hardware and software reside.
- **Systems Communications Description (SV-2)** identifies the communications physical nodes and interconnections. This product represents the specific communications systems pathways or networks and the details of their configurations through which the physical nodes and systems interface. This product focuses on the physical aspect of the information needlines represented in OV-2.
- **Technical Standards Profile (TV-1)** identifies the technical standards that apply to the architecture. It may be appropriate to decompose TV-1 into interface standards that align to an overarching accepted standard like the Open System Interface (OSI) standard and into other standards related to services and physical systems.

Phase 2: **Systems Interface Mapping**

To provide the basis for connectivity/interoperability analysis, this phase focuses on the interfaces between systems and between system functions and the standards associated with those interfaces. One architecture product not used yet in the analysis is used. Two products used earlier are applied here, but different attributes within the products are relevant.

Þ **Identify system interfaces**. Determine the presence of planned and existing system-to-system interfaces from the Systems-Systems Matrix (SV-3).

Þ **Systems-Systems Matrix (SV-3)** defines system-to-system relationships by depicting the status (e.g., existing, planned, potential) of the interface between the systems.

Þ **Determine the information flow among system functions** using the Systems Functionality Description (SV-4).

Þ **Associate information exchanges with systems** using the Systems Data Exchange Matrix (SV-6).

Þ **Identify protocols and data/media formats** for system information exchanges from the Systems Data Exchange Matrix (SV-6).

Phase 3: **Connectivity/Interoperability Analysis**

Þ **Use 1st Order Analysis results**. The 1st Order Analysis provides systems mapped to activity flow, systems mapped to system functions, and an identification of system functional duplications and gaps.

Þ **Determine whether the logical interfaces are correctly connected**. **Figure 3.3-7** depicts the step-by-step sequential mapping to accomplish this. Relate activities to system function flow; associate the information exchanges from the OV-3 to system data exchanges in the SV-6; relate the data flow to the system functions; and relate standards to the data flow. Compare the data/media standards in the SV-6 to the appropriate standards in the TV-1. Identify mismatches. Identify areas where standards are not present in either the SV-6 or TV-1.

Þ **Determine whether the systems are correctly connected** (identify static interoperability issues). **Figure 3.3-8** depicts the sequential mapping for this analysis. Relate data flow to systems and relate standards to the data flows. Identify points at which multiple systems appear to support the same data flow, and determine whether each of those systems is essential.

Results: 2nd Order Analysis can identify interoperability issues and the need for specific systems to adhere to enterprise standards for certain data/media formats or the need for a standard to be established for certain data/media formats. In addition to providing a static interoperability assessment, the 2nd Order Analysis identifies disconnects in logical interfaces and systems connectivity. These analytical results support decision making related to interoperability issues and POM decisions.

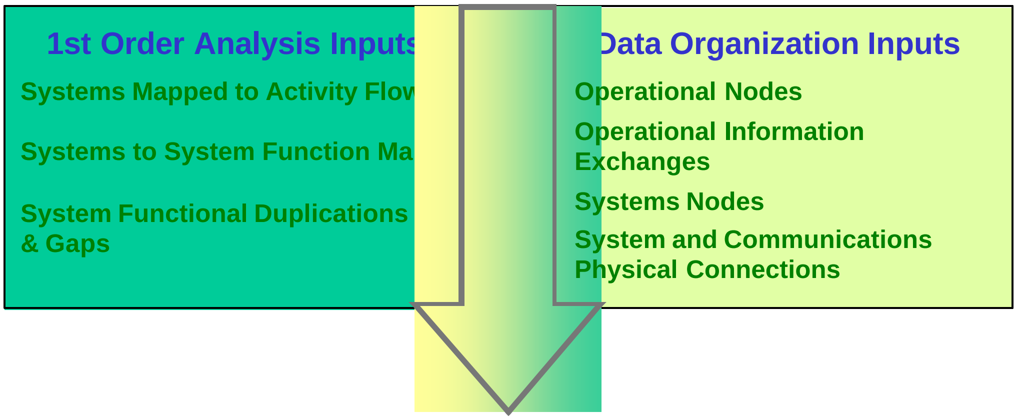
Figure 3.3-9 provides an overview of the logical process for 2nd Order Analysis.

3.3.3.4 3rd Order Analysis: Architecture Performance and Behavior

The 1st and 2nd Order Analyses examine the functionality and connectivity of the architecture with traceability to operational capability. As such, these *uses* of Framework products provided an early validation of the architecture and serve to answer the question: *What can the architecture enable the FoS to actually do*? However, the architecture is not (abstractly) validated until it can be executed as a flow of events; this is accomplished through the products of performance and behavior. An executable model is used with the Operational Event-Trace Description (OV-6c) and Systems Performance Parameters Matrix (SV-7) to depict the architecture dynamically and to provide a dynamic interoperability assessment (see **Figure 3.3-10**).

The objectives of 3rd Order Analysis are to determine:

- How well does the architecture perform (to deliver mission capabilities)?
- Does the architecture behave in ways acceptable to the users?
- Is data accuracy and timing among systems correct?



1st Order Analysis Inputs

Systems Mapped to Activity Flow

Systems to System Function Ma

System Functional Duplications & Gaps

Data Organization Inputs

Operational Nodes

Operational Information Exchanges

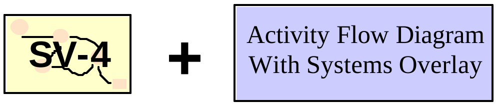
Systems Nodes

System and Communications Physical Connections

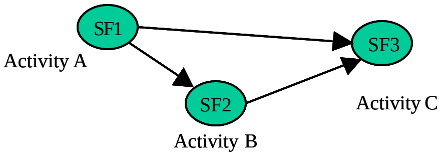
Are the Logical Interfaces Correctly Connected?

1. Determine System Function Flow

Use

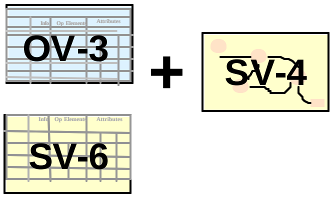


Map Operational Activities to System Functional Flow

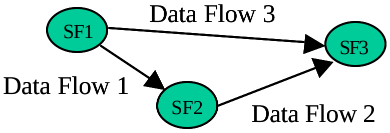


2. Identify System Data Elements Associated with System Function Flow

Use

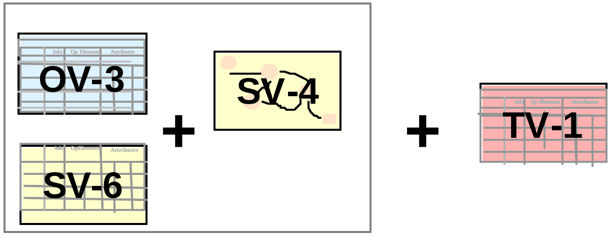


Associate Data Flow to System Function Flow

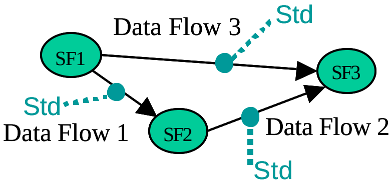


3. Identify Logical Interface Standards

Use

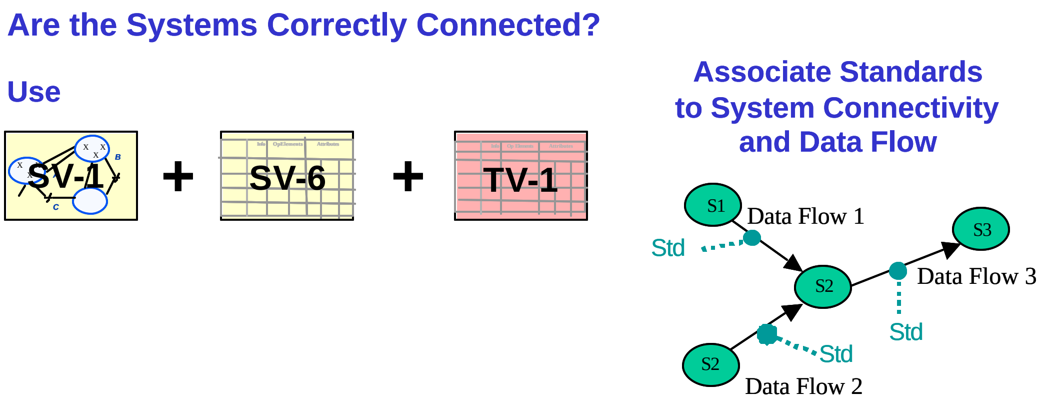


Associate Standards to System Function Flow



Compare standards in TV-1 to standards listed in the SV-6

Figure 3.3-7. 2nd Order Analysis Part 1



Are the Systems Correctly Connected?

Use

Associate Standards to System Connectivity and Data Flow

Figure 3.3-8. 2nd Order Analysis Part 2

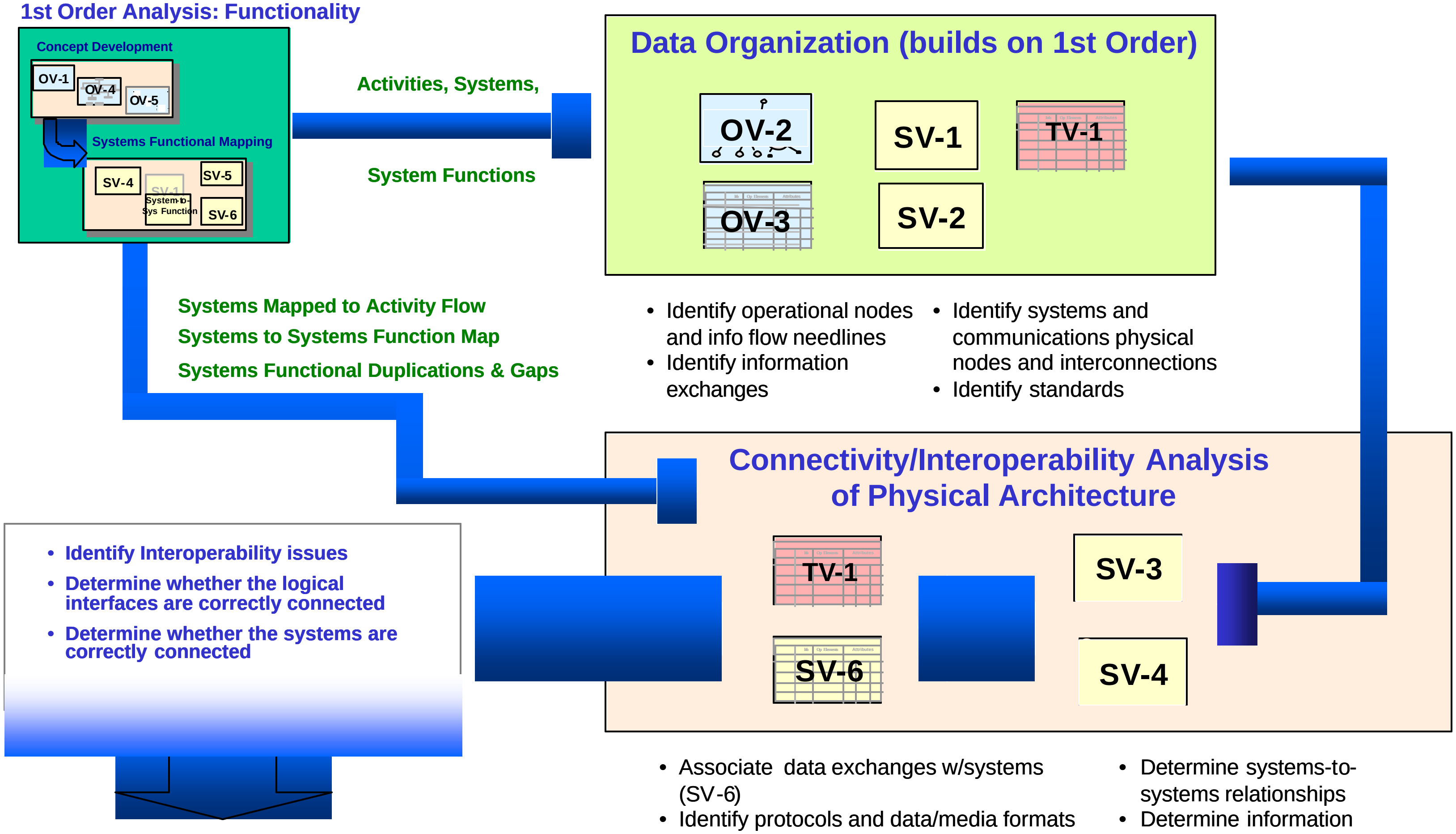
Þ **Define dynamic behavior**. Two Framework products address operational and system sequencing.

- **Operational Event-Trace Description (OV-6c)**, sometimes called a sequence diagram, is a basic product for addressing the executability (or dynamic validity) of the Operational View of the architecture. It enables the traceability of actions in a scenario or critical sequence of events. OV-6c organizes OV-5 activities around OV-2, using OV-4 for command and control of architecture responses to scenario events. It introduces timing and sequencing into the activity model (OV-5). Insight into dynamic validity, throughput, and node loading is gained. However, this view does not address architecture performance. The performance of the architecture is determined by the performance of the systems and personnel that enable or execute the operational activities.
- **System Event-Trace Description (SV-10c)** is inherited from OV-6c using the mapping of SV-5 and other SV-3 and SV-4 products.

Þ **Overlay the activity flow with the associated systems**. Using OV-6c as the foundation, map the systems associated with each activity. This is similar to the activity flow diagram used in the 1st Stage Analysis and shown in Figure 3.3-5. However, in 3rd Order Analysis, as shown in, **Figure 3.3-11** events, timing, and systems are overlayed on the activity sequence.

2nd Order Analysis

- Focus on Physical and Logical Interfaces and Static Interoperability
- Relate Operational Activity Sequence to Physical Domain


1st Order Analysis: Functionality
Concept Development
OV-1
OV-4
OV-5
Systems Functional Mapping
SV-4
SV-1
SV-5
System-to-Sys Function
SV-6
Activities, Systems,
System Functions
Data Organization (builds on 1st Order)
OV-2
OV-3
SV-1
SV-2
TV-1
Systems Mapped to Activity Flow
Systems to Systems Function Map
Systems Functional Duplications & Gaps
Connectivity/Interoperability Analysis of Physical Architecture
TV-1
SV-6
SV-3
SV-4

- Identify operational nodes and info flow needlines
- Identify information exchanges

- Identify systems and communications physical nodes and interconnections
- Identify standards

- Identify Interoperability issues
- Determine whether the logical interfaces are correctly connected
- Determine whether the systems are correctly connected

- Associate data exchanges w/systems (SV-6)
- Identify protocols and data/media formats for system info exchanges (SV-6)
- Identify technical standards, protocols, and formats as architecture standards (TV-1)

- Determine systems-to-systems relationships
- Determine information flow among system functions

POM and Interoperability

Figure 3.3-9. 2nd Order Analysis

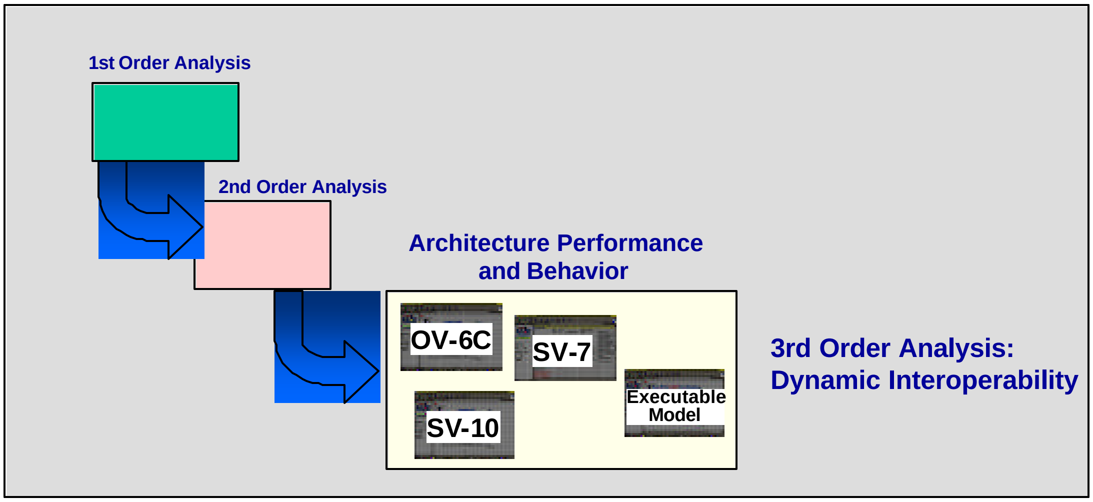

1st Order Analysis
2nd Order Analysis
Architecture Performance and Behavior
OV-6C
SV-7
SV-10
Executable Model
3rd Order Analysis: Dynamic Interoperability

Figure 3.3-10. 3rd Level Analysis

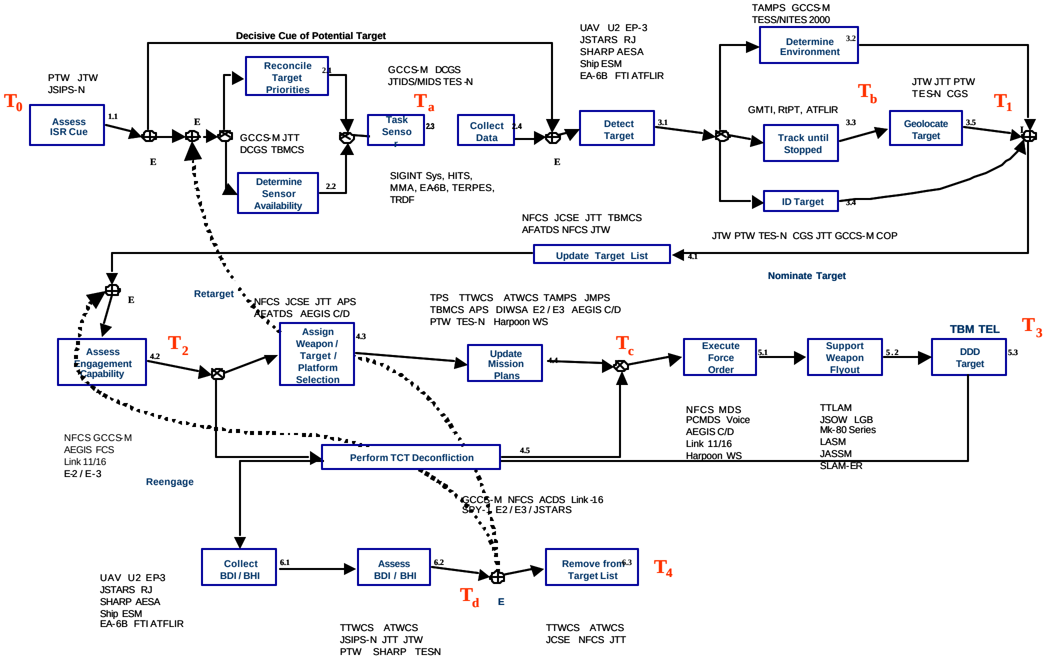


Figure 3.3-11. Activity Flow Diagram with Overlay of Events, Timing, and Systems
(Based on Navy RDA CHENG Graphic)

Þ **Identify performance characteristics**. Use the Framework product below.

- **Systems Performance Parameters Matrix (SV-7)** builds on the Systems Interface Description (SV-1) to depict the current performance characteristics of each system and the expected or required performance characteristics at specified times in the future. The expected characteristics relate to the Systems Evolution Description (SV-8), whereas the performance requirements for physical systems are traceable only when an allocated baseline has been established (i.e., functions and requirements have been allocated to physical systems).

⇒ **Evaluate execution of the architecture**. Evaluation of the architecture dynamics in an Executable Model of the architecture is required for both validation and analysis. A number of popular tools are available. RDA CHENG has been using a popular tool developed for structured analysis. However, future work will move towards object orientation using the Unified Modeling Language (UML). This will allow for better re-use of the architecture products and provide better control of attributes through the inheritance property of UML.

3.3.3.5 Acquisition Strategy

A capability-based acquisition strategy aligns the evolution of systems, technologies, and standards into an acquisition strategy to support the evolving capabilities needed for the FoS.

Þ **Identify the evolution of technologies and standards**. Two Framework products provide this information.

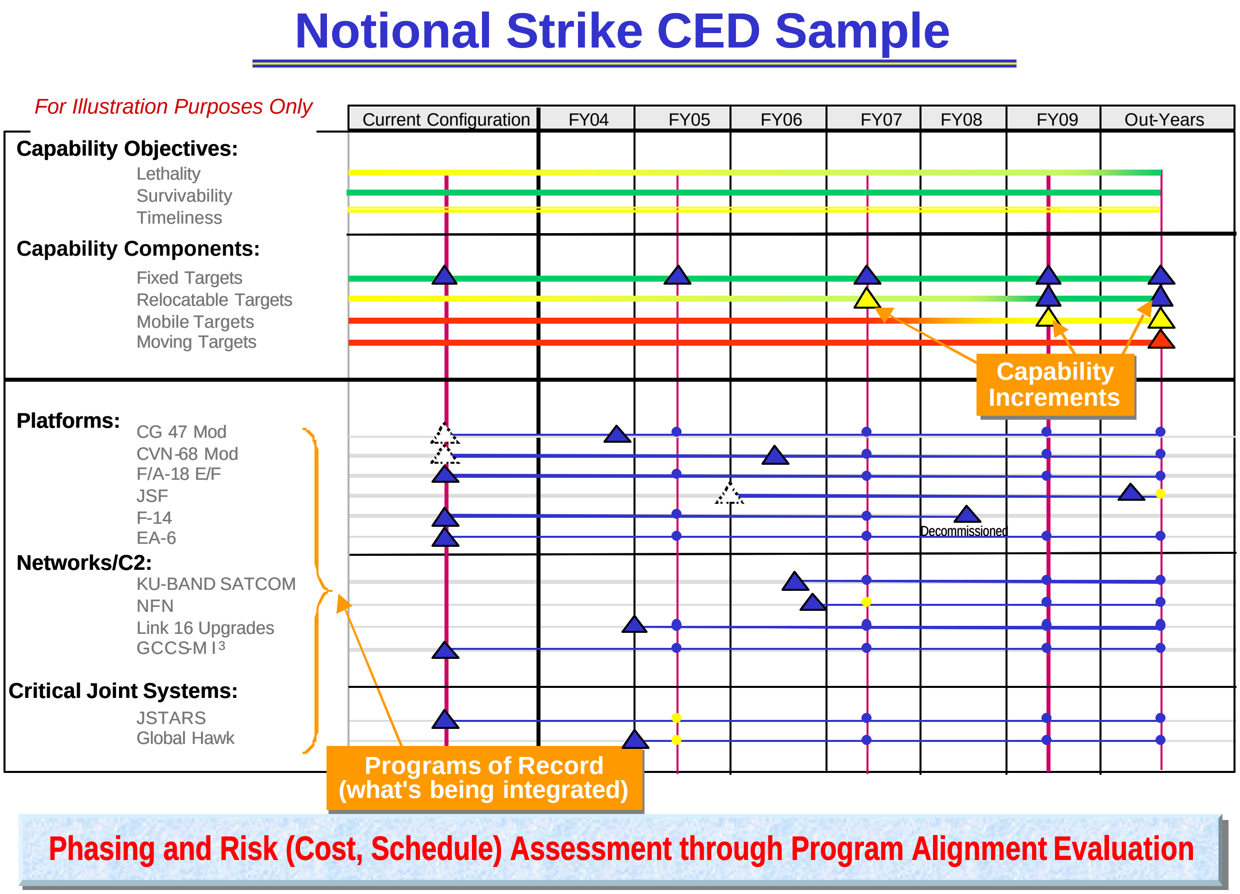
- **Systems Technology Forecast (SV-9)** is a detailed description of emerging technologies and specific hardware and software products. It contains predictions about the availability of emerging capabilities and industry trends in specific time frames (e.g., 6-month, 12-month, 18-month intervals) and confidence factors for the predictions. The forecast includes potential technology impacts on current architectures and thus influences the development of transition and objective architectures. The forecast should be tailored to focus on technology areas that are related to the purpose for which a given architecture is being built, and should identify issues that will affect the architecture.
- **Technical Standards Forecast (TV-2**) is a detailed description of emerging technical standards relevant to the systems and business processes covered by the architecture. It contains predictions about the availability of emerging standards and the likely obsolescence of existing standards in specific time frames (e.g., 6-month, 12-month, 18-month intervals) and confidence factors for the predictions. It also contains matching predictions for market acceptance of each standard and an overall risk assessment associated with using the standard. The forecast includes potential standards impacts on current architectures and thus influences the development of transition and objective architectures. The forecast should be tailored to focus on technical standards areas that are related to the purpose for which a given architecture description is being built and should identify issues that will affect the architecture.

Þ **Determine the evolution of systems**. Correlate emerging technologies with the systems evolution at points in time. Correlate the emerging technical standards with the emerging technologies and thus with the evolving systems.

- **System Evolutions Description (SV-8)** describes plans for "modernizing" a system or suite of systems over time. Such efforts typically involve the characteristics of evolution (spreading in scope while increasing functionality and flexibility) or migration (incrementally creating a more streamlined, efficient, smaller, and cheaper suite), and will often combine the two thrusts. SV-8 should draw heavily from SV-9 and TV-2.

Þ **Develop the Capabilities Evolution Description (CED)**. CED depicts required program plans aligned to capability to objectives and increments over time. As such, it exhibits the integration strategy for networks, sensors, weapons, and platforms. Navy definitions of capability are provided at the inset below.

In the sample CED in **Figure 3.3-12**, Capability Objectives (lethality, survivability, and timeliness) are noted in the top left. Capability Components (fixed, relocatable, mobile, and moving targets) are listed just below the Capability Objectives and are the various objects against which the capability is required. The types of mechanisms for achieving the required mission capability are listed along the left axis. Specific systems belonging to each mechanism type (platforms, networks/C2, and critical joint systems) and contributing to the required mission capability are listed under the mechanism type. The triangles (yellow and green) denote the points in time at which a specific system achieves an improvement in capabilities. For example, the CG-47 Mod achieves the required attributes in FY04; the CVN-68 Mod achieves required attributes in FY06; and the F/A-18 E/F has the necessary capability in its current configuration.



Notional Strike CED Sample

For Illustration Purposes Only

Phasing and Risk (Cost, Schedule) Assessment through Program Alignment Evaluation

Figure 3.3-12. Sample Capability Evolution Diagram

The horizontal lines aligned with the capabilities provide an overall measurement of the capability achieved at that time by the combination (bundling) of the depicted mechanisms (Capability Increment). The yellow and green dots and triangles on the red vertical lines (at FY05, FY07, FY09, and out-years) are the measure of the level of capability that each system has achieved at that time. The triangles on these horizontal lines depict the measure of the Capability Increment achieved by that time against the Capability Components. For example, required capabilities (green) are achieved for relocatable targets in FY09. The color of the horizontal lines aligned with capabilities denotes a measure of that overall capability. The graphic depicts lethality as yellow beginning to turn green at the end of FY09.

Capability – The ability to execute a specified Course of Action (COA). Used in defining requirements, having a quality or an ability to perform a group of tasks in performance of mission.

Mission Capability – The possession of the means to use military force to achieve an intended effect within the battlespace that can be measured.

Capability Objective – A capability or related set of capabilities with decisive and attainable goals toward achieving mission capabilities.

Capability Increment – A bundling of networks, sensors, weapons, and platforms aligned in acquisition over time that enables a capability objective.

The CED depicts the evolution of capabilities achieved through connected, interoperable sets of systems that together provide desired combinations of capabilities required for mission accomplishment. The FoS derives its capabilities through the interoperation of systems, not just through the operation of individual systems. Thus, the evolution of system connectivity can be given equal attention with individual system evolution. The delivery of systems and the associated integration and interoperability strategy are aligned and displayed in the CED, so that connectivity, alignment, and traceability to capabilities are all displayed in one graphic.

The CED is intended to assist managers and executives in making acquisition and investment decisions and is a bridge between the Planning, Programming, Budgeting, and Execution (PPBE) process and the Defense Acquisition System.

Þ **Develop Acquisition Strategy**. Using CEDs, portfolios of programs can be bundled by the capability increments referred to in the High-Level Operational Concept Graphic (OV-1). Increments of capability introduced over time would then establish the evolution of the FoS in acquisition. The CED, used with SV-8, SV-9, and TV-2, provides a description of the evolution and acquisition of the system improvements to the FoS that is traceable to mission capabilities.

Figure 3.3-13 provides an overview of how the three orders of analysis contribute toward developing an acquisition plan.

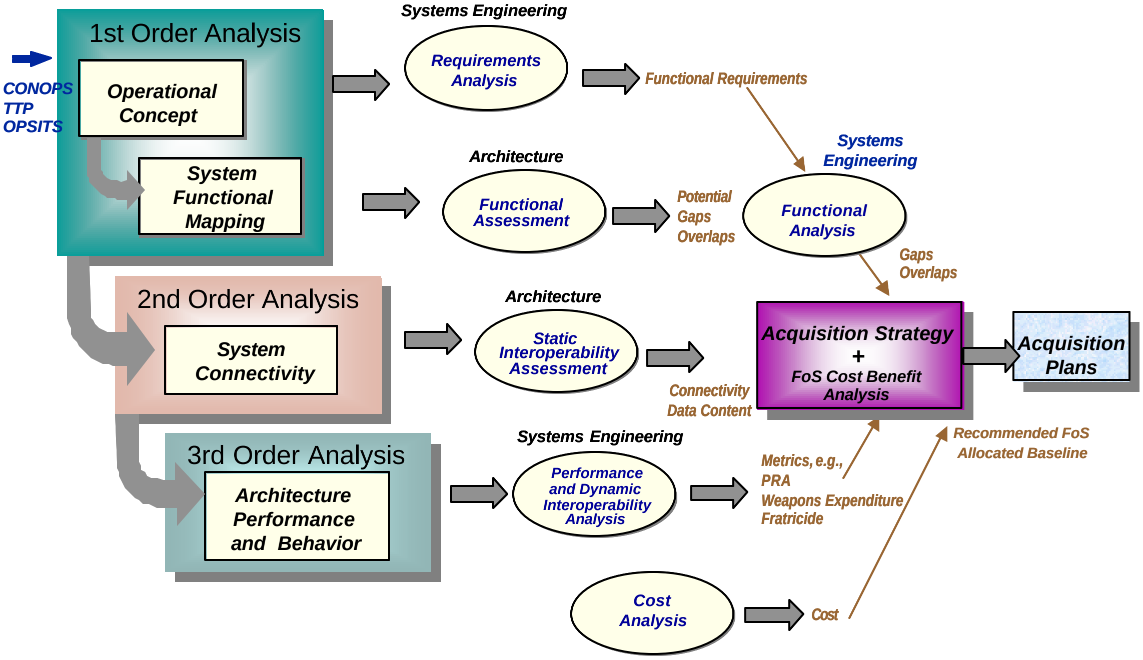


Figure 3.3-13. Using Architecture Assessments and Systems Engineering to Develop the Acquisition Plan
(Adapted from a Navy RDA CHENG slide)

3.3.4 References

Assistant Secretary of the Navy for Research, Development, and Acquisition (ASN (RDA)) Chief Engineer (CHENG), *Architecture Reports in Support of the Development of Mission Capability Package (MCP) for Time Critical Targeting (TCT)*, March 21, 2002.

Charles, Phil, ASN RDA CHENG, *Using Architectures for Interoperability Assessments*, Briefing, January 24, 2002.

Dickerson, Charles (RDA CHENG, Director of Architectures) and CAPT (USN Ret) Steve Soules (Booz Allen and Hamilton), *Using Architecture Analysis for Mission Capability Acquisition*, May, 2002.

Dickerson, Charles (RDA CHENG, Director of Architectures), Soules, S. M., Sabins, M. R., and Charles, P. H., *Using Architectures for Research, Development, and Acquisition*, undated.

RDA CHENG, *Integration and Interoperability Strategy for Capabilities Based Acquisition*, Briefing, March 6, 2002.

Thilenius, Jim (MITRE), Air Force Chief Architects Office, *Proposal for New Capability View of DoD Architectures*, Briefing, June 6, 2002.

3.4 DESIGNATION AND USE OF KEY INTERFACE PROFILES

3.4.1 Introduction

Architectures typically include multiple networks hosting multiple applications and data sets. Interoperability across the interfaces between the various networks, applications, and data sets is especially challenging. One approach for improving interoperability is to manage interoperability across interfaces via Key Interface Profiles (KIP).

KIPs provide a net-centric oriented approach for managing interoperability across the Global Integrated Grid (GIG) based on the configuration control of key interfaces. The KIP is a set of documentation produced as a result of interface analysis which:

- Designates an interface as key
- Analyzes it to understand its architectural interoperability, test, and configuration management characteristics
- Documents those characteristics in conjunction with solution sets for issues identified during the analysis

An interface-oriented approach for managing interoperability offers several benefits. A single interface specification is easier to develop, implement, maintain, and enforce than maintaining synchronization of the internals of numerous systems. The approach is more legacy tolerant, since it does not always assume or require changes to the internals of related systems. The approach is also system evolution tolerant; system internals could be changed, capabilities enhanced, and new technology incorporated, as long as the interface remains stable or evolves in a measured way consistent with a defined configuration management process.

The concept of key interfaces is based on an interface at a boundary. The interface may be point-to-point or between multiple points. The KIP process provides a methodology for working interoperability issues across these interfaces. DoD is moving toward Net-Centric Operations and Warfare (NCOW), where all entities operate on “the grid.” The KIP process can assist in realizing the interoperability stepping-stones that will move DoD from its current system-of-systems environment to a NCOW grid concept. As the connectivity grid is realized, the KIP process offers value for addressing issues of interoperability with the grid. In the NCOW context, the interface is between some type of information technology mechanism (such as a router, a gateway, or a firewall) and the grid.

3.4.2 Identifying a Key Interface

The DoD Dictionary of Military and Associated Terms defines the term *interface* as “a boundary or point common to two or more similar systems, subsystems, or other entities against which necessary information flow takes place.” Per Military Handbook 61A, interfaces are defined in functional and physical characteristics that exist at a common boundary with co-functioning items and allow systems, equipment, software, and data to be compatible. Illustrative types of key interfaces are depicted in **Figure 3.4-1**. An interface may be designated as a Key Interface when one or more of the following criteria are met:

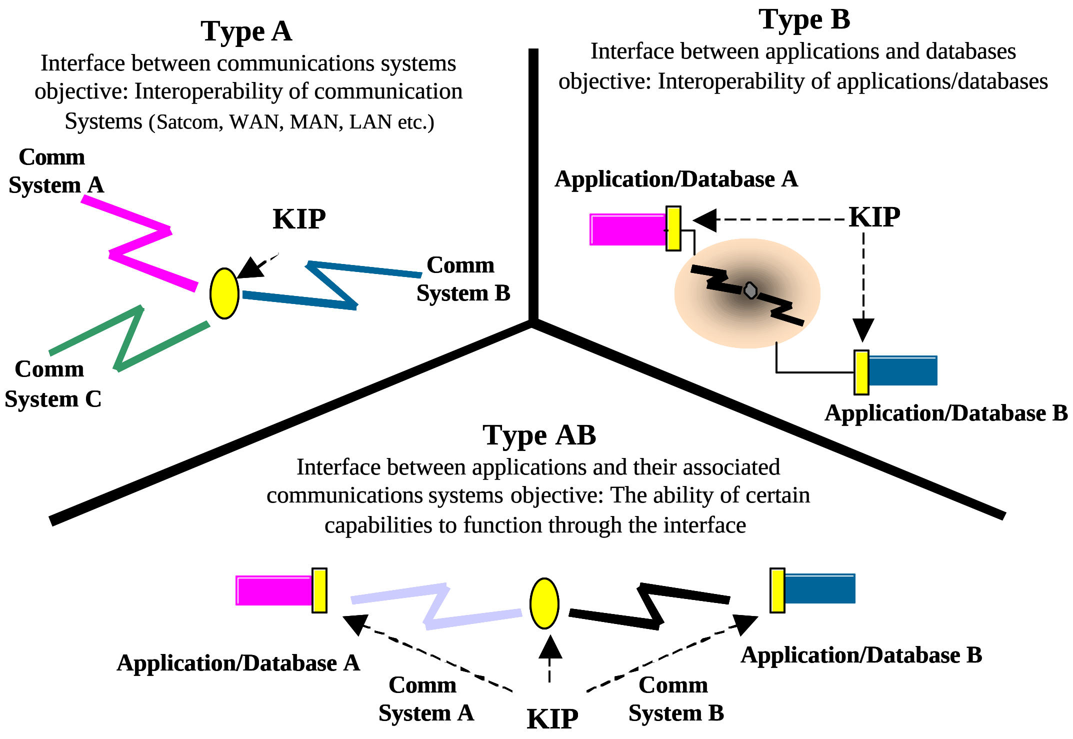

Type A
Interface between communications systems objective: Interoperability of communication Systems (Satcom, WAN, MAN, LAN etc.)
Comm System A
KIP
Comm System B
Comm System C
Type B
Interface between applications and databases objective: Interoperability of applications/databases
Application/Database A
KIP
Application/Database B
Type AB
Interface between applications and their associated communications systems objective: The ability of certain capabilities to function through the interface
Application/Database A
Comm System A
KIP
Comm System B
Application/Database B

Figure 3.4-1. Illustrative Types of Key Interfaces

- The interface spans organizational boundaries. Different entities (service, agency, organization) have ownership and authority over the hardware and software capabilities on either side of the boundary.
- The interface is mission critical. Data from joint organizations, multiple services, and/or multiple agencies/organizations must move across the interface to satisfy joint information flow requirements. If systems are not interoperable at that interface, the ability to accomplish the mission is endangered.
- The interface is difficult or complex to manage.
- There are capability, interoperability, or efficiency issues associated with the interface. Types of issues/problems include the following:
  - Does not support required information flow
  - Lack of functionality
  - Inefficiencies (such as specialized interface connections)
  - Lack of connectivity
  - Lack of required interoperability
  - Lack of an appropriate electronic connection
- The interface impacts multiple acquisition programs, usually more than two (e.g., network points of presence, many-to-many or one-to-many connections).
- The interface is vulnerable or important from a security perspective.

A Key Interface can exist at boundaries involving two or more responsible entities. For example, Key Interfaces may exist at boundaries involving:

- Different services or a service and a joint or DoD organization
- Different security domains
- DoD functional proponents and other DoD entities
- Different Joint Mission Areas
- DoD and non-DoD U.S. organizations
- DoD and non-Federal Government organizations
- U.S. and non-U.S. forces

3.4.3 Analyzing an Architecture to Identify Key Interfaces

Figure 3.4-2 depicts how architecture products and reports can be used to identify Key Interfaces. The Prioritized Capability List (CR-1), Operational Profile (CR-3),[1] Overview and Summary Information (AV-1), and High-Level Operational Concept Graphic (OV-1) provide information on critical mission capabilities and associated shortfalls. The Operational Node Connectivity Description (OV-2) and the Operational Information Exchange Matrix (OV-3) provide a basis for determining where information has to flow between nodes that are under the authority of different organizations. To facilitate determining when nodes are under different authorities, organization name as well as node name can be included in OV-3.

The Systems Interface Description (SV-1) and Systems Communications Description (SV-2) provide the systems overlay for the information flow. The owning authority for the IT and communication systems in these products facilitates determining when information has to move over and between systems that are under different authorities. The Systems-Systems Matrix (SV-3) notes when an interface is existing, planned, or missing. The Systems Data Exchange Matrix (SV-6) provides data exchanges with sending and receiving systems. Communications systems that move the data along with the IT systems that operate on the data facilitates identifying potential Key Interfaces. Identifying the Executive Agents and relevant Acquisition Programs brings in another dimension to consider when identifying a Key Interface, and could be included as part of the interface information in SV-3.

[1] Capability Reports are discussed in Section 3.2: Air Force Capability-Based Analysis.

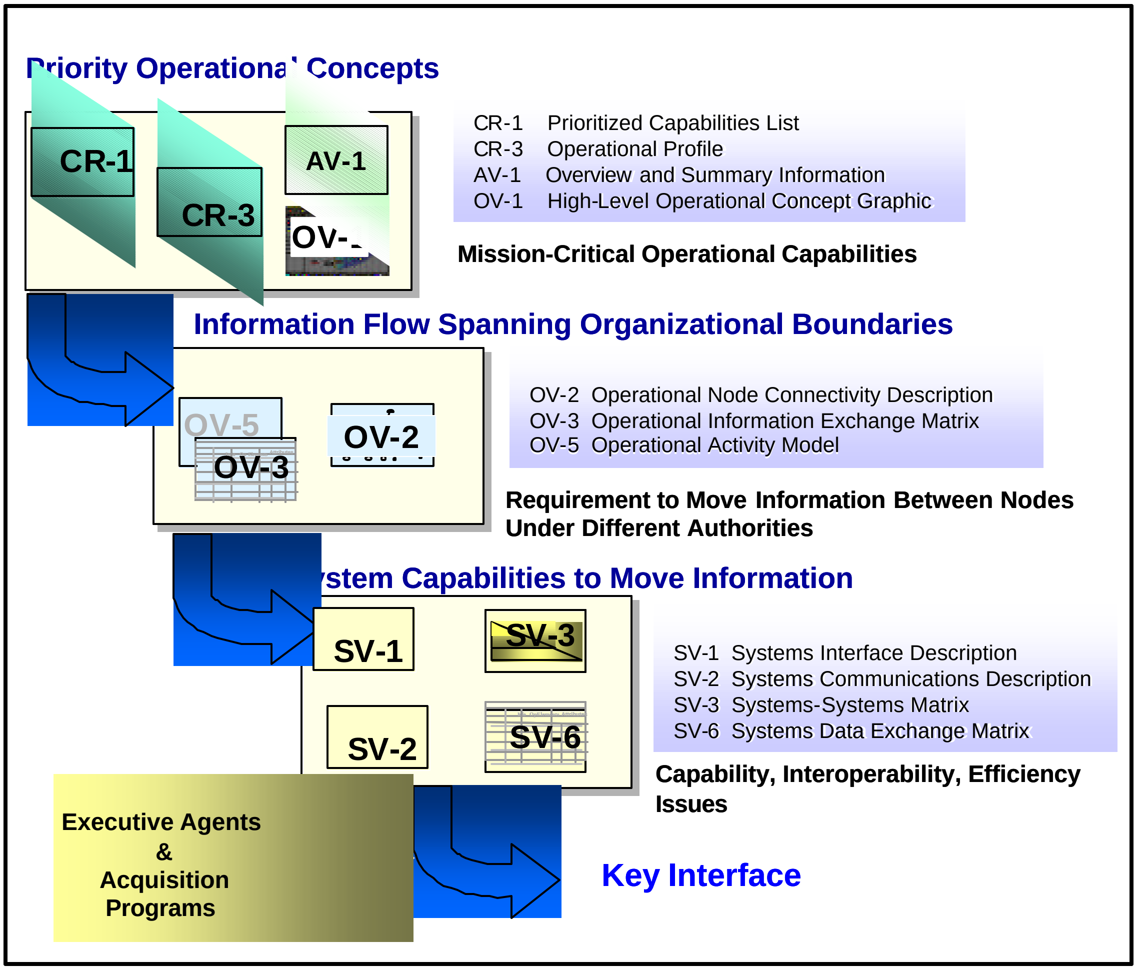

Priority Operational Concepts
CR-1
CR-3
AV-1
OV-1
CR-1 Prioritized Capabilities List
CR-3 Operational Profile
AV-1 Overview and Summary Information
OV-1 High-Level Operational Concept Graphic
Mission-Critical Operational Capabilities
Information Flow Spanning Organizational Boundaries
OV-5
OV-3
OV-2
OV-2 Operational Node Connectivity Description
OV-3 Operational Information Exchange Matrix
OV-5 Operational Activity Model
Requirement to Move Information Between Nodes Under Different Authorities
System Capabilities to Move Information
SV-1
SV-3
SV-2
SV-6
SV-1 Systems Interface Description
SV-2 Systems Communications Description
SV-3 Systems-Systems Matrix
SV-6 Systems Data Exchange Matrix
Capability, Interoperability, Efficiency Issues
Executive Agents & Acquisition Programs
Key Interface

Figure 3.4-2. Using Architecture Products to Identify a Key Interface

3.4.4 Developing a Key Interface Profile

A KIP describes the interface in terms of required operational and systems functionality and technical specifications. Issues associated with the interface, when identified in the parent architecture,[2] should be resolved in the KIP. For example, if there is a lack of interoperability between applications, the KIP should provide the specifications for implementing the required interoperability. If multiple communications systems interface to a common point but require system-unique interfaces, the KIP should define a more efficient interface.

- Operational View (OV) - provides a description of the required operational characteristics for the interface to include connectivity and information exchange requirements. The OV products for the KIP are based on portions of the parent Operational View that relate to the interface. The KIP OV products refine and expand on the information provided in the parent architecture to provide additional depth for aspects relevant to the interface.

[2] The term “parent architecture” as used in this section refers to the architecture that was used to identify the Key Interface.

- Systems View (SV) - provides a description of the IT mechanisms (hardware and software) relating to the interface and includes system capability and interoperability requirements. The SV products for the KIP are based on portions of the parent System View that relate to the interface. The KIP SV products refine and expand on the information provided in the parent architecture to provide additional depth for aspects relevant to the interface.
- Interface Control Document (ICD) per MIL HDBK 61A - In addition to providing the technical specifications for the KIP, the Interface Control Document should include data characterization and formats as well as the rules, conventions, and criteria that govern the operation of the KIP.
- Standards Profile - contains a TV-1: Technical Standards Profile, TV-2: Technical Standards Forecast and a SV/TV Bridge. TV-1 and TV-2 contain SV/TV Bridge data. The bridge maps defined and emerging standards from the TV-1 and TV-2 against implemented systems as defined in SV-1. The purpose of the SV/TV Bridge report is to demonstrate how the technical specifications support the required systems interoperability.

The KIP documentation above provides a functional and technical description of the interface. In order to appropriately manage the interface, the following is also needed:

- Configuration Management Plan that ensures that any changes in the KIP support necessary functionality and interoperability. Procedures for standards conformance and interoperability testing should be included as part of reference implementations.
- Operational View and Systems View products developed for the KIP address those aspects (activities, nodes, information flow, systems, and communications) of the parent architecture that directly relate to the Key Interface. However, the KIP Operational View and Systems View add more detail to the corresponding segments initially developed in the parent architecture. For example, the KIP OV would include only those activities from the parent architecture (OV-5) that operate through the interface. But, in the KIP Operational View, those activities may be decomposed to a lower level. In the KIP Systems View, only systems relevant to the interface are included, but they are specified in more detail.
- After their development, the KIP Operational View and Systems View are integrated into future versions of the parent architecture. The KIP Standards Profiles become components of the Technical Standards View of the parent architecture. The Interface Control Document, Configuration Management Plan, and Procedures for Standards Conformance and Interoperability Testing are made available to DoD system developers or other DoD personnel.

Figure 3.4-3 provides an overview of the KIP process.

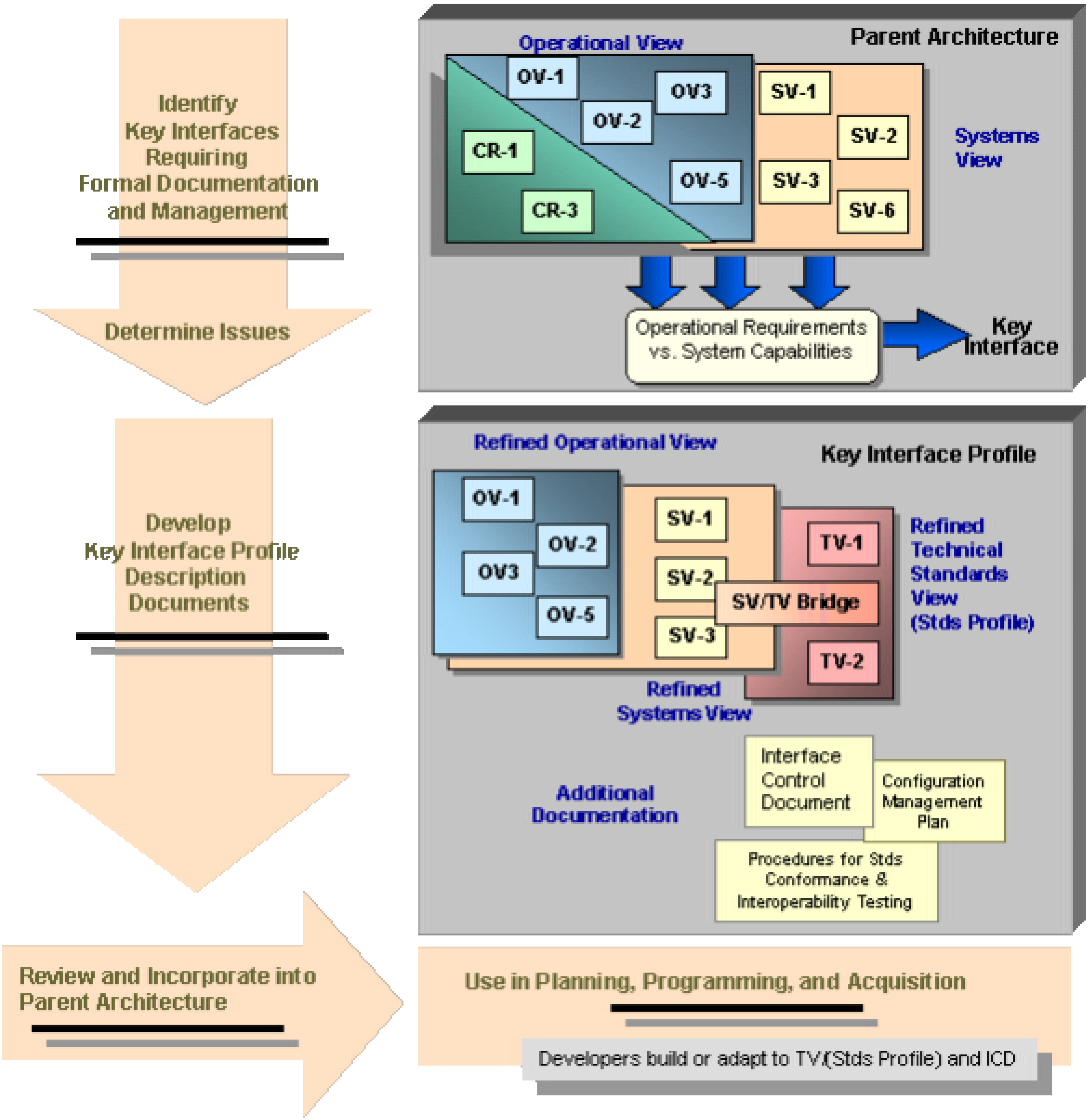


Figure 3.4-3. KIP Process

3.4.5 Uses and Benefits of the KIP Profile

Interoperability issues often arise among systems at the seams between networks and technologies, particularly if those systems were developed by different organizations. As used here, the term *interface* refers to moving data elements between multiple applications as well as passing data transmissions across boundaries between systems. An interface-oriented interoperability management approach is more economical and palatable to subordinate organizations than approaches that depend entirely on commonality or synchronization of internal implementation details. In particular, legacy system owners can often adapt to joint interfaces at the edges of their systems without scrapping the remainder of their implementations.

An interface approach is more economical to implement, since it does not require synchronization of the internals of numerous systems. In addition to being more economical, standardization efforts focused on inter-system interfaces permit more rapid adoption of new technology behind the interfaces. As long as systems continue to comply with appropriate joint Interface Control Documents at their external interfaces, implementation details that are transparent have greater flexibility. It does not matter how well neighboring systems synchronize fielding of system changes, if all systems continue to comply with applicable (and relatively stable) joint Interface Control Documents. This results in far fewer inter-system scheduling and funding dependencies, and therefore a less brittle inter-network of systems.

As discussed in section 3.4.2, and depicted in Figure 3.4-2, architecture information provides the basis for identifying Key Interfaces, where different authorities govern at the boundaries but interoperability is critical for mission accomplishment. The architecture also provides the basis for specifying the interoperability requirements at the interface. One of the objectives of developing the KIP is to more specifically define issues related to the interface and then resolving those issues.

The resultant KIP can be useful in the DoD planning, programming, and acquisition processes. The Profile specifies those changes to the interface that need to be addressed within the planning process. Resources to support the interface, or necessary modifications to the interface, are addressed via programming. Functional and technical specifications for the interface are inputs for acquisition efforts that must use the interface.

3.4.6 References

KIP Working Group, OASD(C3I)/DCIO, *Managing Key Interface Points*, White Paper, April 11, 2002.

Mabry, Roy, OASD(C3I)/DCIO, *GIG Key Interface Point Management*, Progress Report Briefing to the GIG Architecture Working Group, September 12, 2002.

OASD(C3I)/DCIO and OJCS/J6I, *Key Interfaces of the GIG v 1.0.* Status Brief to the GIG Architecture Working Group, December 4, 2002.

OJCS/J6, *GIG Key Interface Point (KIP) Management*, Progress Report and Decision Briefing to the GIG Architecture Integration Panel, January 22, 2002.

3.5 C4I SUPPORT PLANS

3.5.1 Applicable Policy Document

Command, Control, Communications, Computers, and Intelligence (C4I) Support Plans (C4ISPs) are developed to identify and resolve implementation issues related to an acquisition program's Command, Control, Communications, Computers, Intelligence, Surveillance, and Reconnaissance (C4ISR) infrastructure support and information technology (IT) system (including National Security Systems [NSS]) interface requirements. In order to accomplish this, the C4ISP must document an evaluation of the supportability and interoperability of IT and NSS.[1]

A C4ISP includes:[2]

- A system description
- Operational employment concept and employment rates including mission area-focused Operational, Systems, and Technical Standards Views
- C4ISR support requirements derived through analysis from the employment concept/rates, architecture views, and the performance capabilities and characteristics specified by [a requirements document] validated by the requirements authority
- Potential C4ISR shortfalls with proposed solutions or mitigation strategies

Instructions for building a C4ISP are contained in the *Interim Defense Acquisition Guidebook*, dated October 30, 2002.[3] The Deputy Secretary's memorandum, *Defense Acquisition*, dated October 30, 2002, and Attachment 2 to that memorandum, reference a guidebook to accompany the interim guidance. The former DoD 5000.2-R regulation[4] serves as that interim guidebook, while the Defense Acquisition Policy Working Group creates a streamlined guidebook. Note that although the guidebook's title and text use the term "mandatory," the former DoD 5000.2-R is not mandatory at this time. Instead, it provides best practices, lessons learned, and expectations. However, the new, streamlined guidebook will be mandatory when completed.

The format and review process for a C4ISP are provided in Appendix 5 to the interim guidebook, "Command, Control, Communication, Computers, and Intelligence (C4I) Support Plan (C4ISP) Mandatory Procedures and Formats." This section of the Deskbook is not intended to duplicate Appendix 5 (i.e., it is not intended as a full tutorial on how to develop a C4ISP). Rather, it is intended as supplementary advice on ways to approach the various architecture products recommended in Appendix 5. This section refers to Appendix 5 as the "policy document."

[1] Implementing the C4I Support Plan Requirements, Briefing, OASD(C3I), DASD (C3ISR & Space), March 2002.

[2] Interim Defense Acquisition Guidebook, DoD, 30 October 2002 (Formerly DoD Regulation 5000.2-R, *Mandatory Procedures for Major Defense Acquisition Programs (MDAP) and Automated Information Systems (MAIS) Acquisition Programs*, April 5, 2002).

[3] Ibid.

[4] Ibid.

3.5.2 Recommended Framework Products

The purpose of a C4ISP is not to build architectures—its purpose is to use existing architectures to support requirements analysis and evaluation.[5] However, where architecture products dealing with the system(s) under consideration do not exist, they may need to be developed. The products that are useful for inclusion in a C4ISP include many of those shown in Volumes I and II under Acquisition Process uses (see Figure 3-7, Architecture Products by Use in Volume I of the Framework; this table is also repeated as Figure 2-2 in Volume II).

Figure 3.5-1 shows the DoD Architecture Framework (DoDAF) products recommended for inclusion in a C4ISP.

- A light grey cell (░) indicates the product is required in order to have an integrated architecture.
- A dark grey cell (▓) indicates the identified product is specified in policy (i.e., policy indicates that the product should be developed to support the indicated use).
- A solid black circle (●) indicates the product is highly applicable to the indicated use (i.e., the product should be developed, when the architecture is intended to support the indicated use).
- A white circle with a center black dot (⊙) indicates that the product is often or partially applicable (i.e., consideration should be given to developing the designated product, when the architecture is intended to support the indicated use).
- A blank cell indicates that the product is usually not applicable (i.e., there is usually no need to develop the designated product, when the architecture is intended to support the indicated use).

Applicable Architecture Products																					
C4I Support Plan	All View		Operational View (OV)							Systems View (SV)											Tech Stds View	
	1	2	1	2	3	4	5	6	7	1	2	3	4	5	6	7	8	9	10	11	1	2
	●	●	●	●	●		●	●		●			⊙		●	⊙					●	⊙

● = Product is highly applicable

blank = Product is usually not applicable

░ = Product is required for an integrated architecture (includes AV-1, AV-2, OV-2, OV-3, OV-5, SV-1, and TV-1)

⊙ = Product is often or partially applicable

▓ = Product is specifically addressed in policy

Figure 3.5-1. Products Applicable to a C4I Support Plan

For each of the architecture products recommended for inclusion in the C4ISP and others recommended by the DoDAF, the following sections discuss relevant architecture data elements to be included, provide special product tailoring advice, and suggest Universal Reference Resources (URRs), where applicable.

[5] *Implementing the C4I Support Plan Requirements*, Briefing, OASD(C3I), DASD (C3ISR & Space), March 2002.

References to Use in Developing C4ISP-Related Architecture Products

When available, the following provide substantive information relevant to the C4ISP and applicable across multiple architecture products:

- Joint Operating Concepts - An articulation of how a future joint force commander will plan, prepare, deploy, employ, and sustain a joint force against potential adversaries' capabilities or crisis situations within the range of military operations.
- Joint Functional Concepts - An articulation of how a future joint force commander will integrate a set of related military tasks to attain capabilities required across the range of military operations.
- Analysis of Alternatives (AoA) - An analysis to assess the advantages and disadvantages of alternatives being considered to satisfy capabilities.
- Relevant Joint Capabilities Integration and Development System (JCIPS) documentation
  - Initial Capabilities Document (ICD)
  - Capability Development Document (CDD)
  - Capability Production Document (CPD)

3.5.2.1 Overview and Summary Information (AV-1) (Highly Applicable)

The AV-1 is not called out explicitly by the policy document, but the Framework recommends it be included. AV-1 is also one of the products specified by the Framework as required for an integrated architecture. C4ISP guidance emphasizes the need to integrate the proposed system with other systems and operational concepts and, by implication, with relevant architectures. The AV-1 is an ideal place to reference these connections. The format of the C4ISP includes much information that is a natural fit for the AV-1. In addition, if C4ISPs include this information in an AV-1, subsequent C4ISPs will be able to reference these products to more easily determine potential architectural relationships.

Specific Product Data Elements to Include in the AV-1

The following data elements are specified in the policy document and can be included in the AV-1:

- Program name
- Acquisition category
- Approved or validated and draft documents that affect the C4ISR and IT aspects of the system being acquired
- Linkage to the relevant Mission Needs Statement (to be replaced by the ICD) and Operational Requirements Document (to be replaced by the CDD and CPD)
- Availability of support functions/capabilities on which the system must rely
- Status within acquisition cycle

- Stage in review process (Stage 1, Stage 2, Stage 3)
- Organizations that have reviewed the C4ISP as of the current date

The following data elements are specified by the policy document and are also called out in the Framework for inclusion in the AV-1:

- Purpose and scope (of the C4ISP)
- Likely scenarios and operational environment in which the proposed system will operate
- Points of contact for further information

Special Product Tailoring Advice

The information required by the C4ISP guidance can be accommodated by the existing structure of the AV-1.

Suggested Universal Reference Resources to Use in Building the Product

URRs are not directly needed to build AV-1, because AV-1 largely consists of descriptions of information in other architecture products. The URRs are chiefly used in building the other architecture products.

3.5.2.2 Integrated Dictionary (AV-2) (Highly Applicable)

The AV-2 is not called out explicitly by the policy document, but the Framework recommends it be included. AV-2 is also one of the products specified by the Frame work as required for an integrated architecture. C4ISP guidance emphasizes the need to integrate the proposed system with other systems and operational concepts and, by implication, with relevant architectures. AV-2 consists of textual definitions in the form of a glossary, a repository of architecture data, their taxonomies, and their metadata (i.e., data about the architecture data). AV-2 provides a central repository for a given architecture’s data and metadata. AV-2 enables the set of architecture products to stand alone, allowing them to be read and understood with minimal reference to outside resources.

Specific Product Data Elements to Include in the AV-2

All architecture data including a glossary, a repository of architecture data, their taxonomies, and their metadata should be included in an AV-2.

Special Product Tailoring Advice

The information required by the C4ISP guidance can be accommodated by the existing structure of the AV-2.

Suggested Universal Reference Resources to Use in Building AV-2

URRs are needed to build the AV-2, because the architecture data may be identified from authoritative resources and taxonomies.

3.5.2.3 High-Level Operational Concept Graphic (OV-1) (Highly Applicable)

The OV-1 is specified by the policy document.

Specific Product Data Elements to Include in the OV-1

The policy document specifies:

- Capabilities and functions (operational activities) of nodes and interfaces (needlines)
- Identification of critical capabilities and functions (operational activities)

Special Product Tailoring Advice

- The policy document specifies that a separate OV-1 should be developed for each mission/capability area that the proposed system supports, but also states that if the missions/capabilities are very similar, one OV-1 can suffice.
- The document further states that a separate OV-1 should be developed for various time frames if the operational concept is expected to change over time. The recommended minimal set of time frames includes current Program Objective Memorandum (POM) year, last POM year (5th year out), and the Initial Operational Capability (IOC) year.
- The policy document also states that the C4ISP's OV-1 must correlate with the OV-1 included with the relevant capabilities documents.

Suggested Universal Reference Resources to Use in Building the Product

- The Global Information Grid (GIG) Architecture describes selected Joint Mission Areas that may be mapped to the mission areas covered by the proposed system.
- The Universal Joint Task List (UJTL) describes the structure of Joint tasks and can be mapped to the functional areas/tasks/activities associated with the proposed system.

3.5.2.4 Operational Node Connectivity Description (OV-2) (Highly Applicable)

The OV-2 is specified by the policy document. OV-2 is also one of the products specified by the Framework as required for an integrated architecture. A notional OV-2 is provided in **Figure 3.5-2**.

OV-2: 2003 Strike Mission

(Notional)

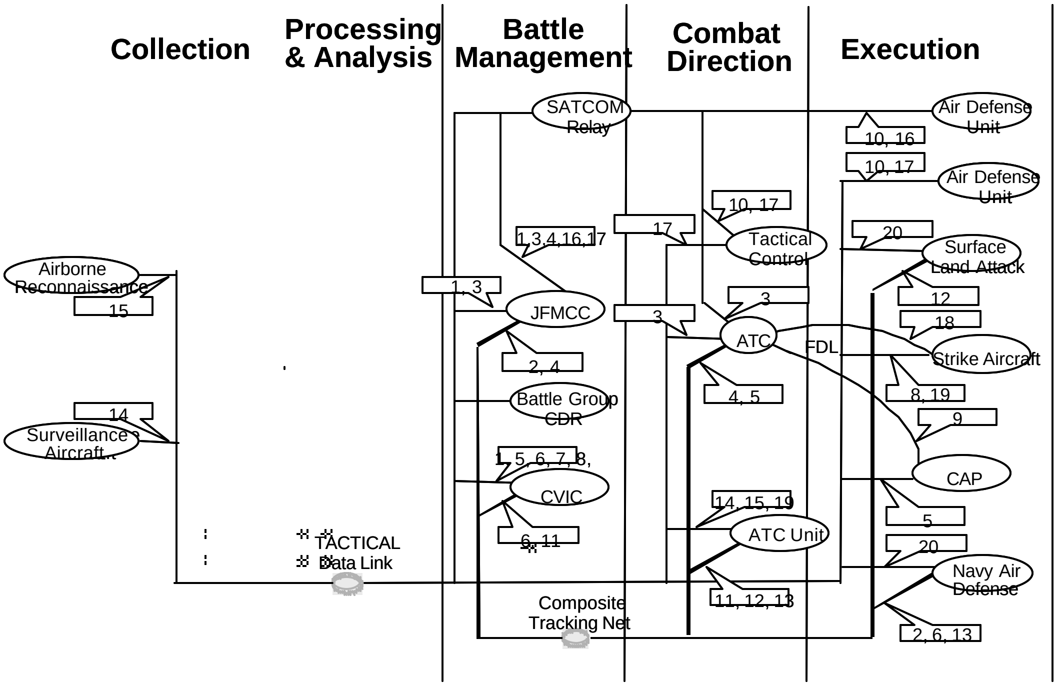


Figure 3.5-2. Notional OV-2

Specific Product Data Elements to Include in the OV-2

The policy document calls for these data elements:

- For each operational node, a description of the node's role
- For each operational node, a description of the critical functions of the node

In addition, the Framework recommends including all of the data elements annotated by an asterisk in the Data Element Definition Table for the OV-2.

The policy document, paragraph 3.1.1 of the Mandatory Format, specifies that the C4ISP's OV-2 should include intra-Service, inter-Service/Joint and combined/coalition C4ISR support and IT interfaces associated with each mission or function supported by the system. Inclusion of C4I support information is consistent with the Framework's description of an OV-2. However, the Framework currently uses the term *interface* to mean the lines shown in the SV-1 that correspond to OV-2 needlines. To clarify the Framework's terminology, the C4ISP's OV-2 should show needlines between operational nodes, and the interfaces between system nodes/systems should be reserved for SV-1. In addition, if useful in the context of a given C4ISP, OV-2 usually shows non-IT needlines as well as IT ones. For example, if some information exchanges are to be done manually and they are important to the context of the C4ISP, then they are usually represented as needlines on the OV-2.

Special Product Tailoring Advice

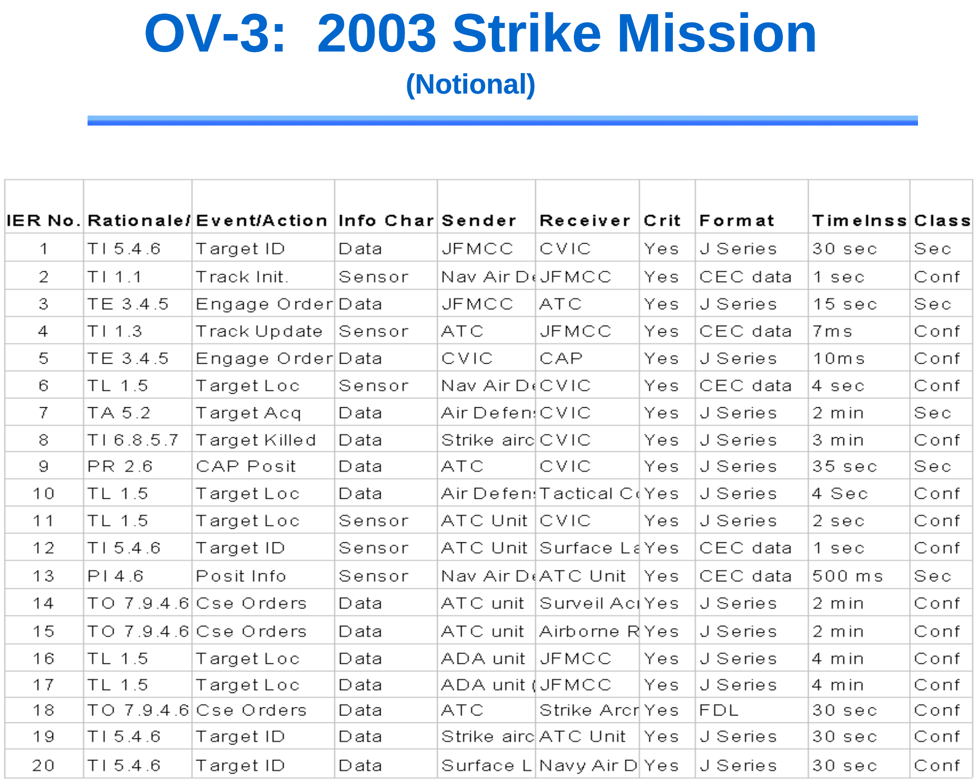
The policy document specifies that a separate OV-2 should be developed for each mission or functional area that the proposed system supports. It further specifies that separate OV-2s should be developed for specific time frames, if the operational concept is expected to change over time.

Suggested Universal Reference Resources to Use in Building the Product

- The GIG Architecture describes selected Joint Mission Areas (JMAs) that may be mapped to the mission areas covered by the proposed system.
- The UJTL describes the structure of Joint tasks and can be mapped to the functional areas/tasks/activities associated with the proposed system.

3.5.2.5 Operational Information Exchange Matrix (OV-3) (Highly Applicable)

OV-3 is specified by the policy document. OV-3 is also one of the products specified by the Framework as required for an integrated architecture. A notional OV-3 is provided in **Figure 3.5-3**.



OV-3: 2003 Strike Mission

(Notional)

IER No.	Rationale/	Event/Action	Info Char	Sender	Receiver	Crit	Format	Timelnss	Class
1	TI 5.4.6	Target ID	Data	JFMCC	CVIC	Yes	J Series	30 sec	Sec
2	TI 1.1	Track Init.	Sensor	Nav Air De	JFMCC	Yes	CEC data	1 sec	Conf
3	TE 3.4.5	Engage Order	Data	JFMCC	ATC	Yes	J Series	15 sec	Sec
4	TI 1.3	Track Update	Sensor	ATC	JFMCC	Yes	CEC data	7ms	Conf
5	TE 3.4.5	Engage Order	Data	CVIC	CAP	Yes	J Series	10ms	Conf
6	TL 1.5	Target Loc	Sensor	Nav Air De	CVIC	Yes	CEC data	4 sec	Conf
7	TA 5.2	Target Acq	Data	Air Defens	CVIC	Yes	J Series	2 min	Sec
8	TI 6.8.5.7	Target Killed	Data	Strike airc	CVIC	Yes	J Series	3 min	Conf
9	PR 2.6	CAP Posit	Data	ATC	CVIC	Yes	J Series	35 sec	Sec
10	TL 1.5	Target Loc	Data	Air Defens	Tactical Co	Yes	J Series	4 Sec	Conf
11	TL 1.5	Target Loc	Sensor	ATC Unit	CVIC	Yes	J Series	2 sec	Conf
12	TI 5.4.6	Target ID	Sensor	ATC Unit	Surface La	Yes	CEC data	1 sec	Conf
13	PI 4.6	Posit Info	Sensor	Nav Air De	ATC Unit	Yes	CEC data	500 ms	Sec
14	TO 7.9.4.6	Cse Orders	Data	ATC unit	Surveil Acr	Yes	J Series	2 min	Conf
15	TO 7.9.4.6	Cse Orders	Data	ATC unit	Airborne R	Yes	J Series	2 min	Conf
16	TL 1.5	Target Loc	Data	ADA unit	JFMCC	Yes	J Series	4 min	Conf
17	TL 1.5	Target Loc	Data	ADA unit (	JFMCC	Yes	J Series	4 min	Conf
18	TO 7.9.4.6	Cse Orders	Data	ATC	Strike Arcr	Yes	FDL	30 sec	Conf
19	TI 5.4.6	Target ID	Data	Strike airc	ATC Unit	Yes	J Series	30 sec	Conf
20	TI 5.4.6	Target ID	Data	Surface L	Navy Air D	Yes	J Series	30 sec	Conf

Figure 3.5-3. Notional OV-3

Specific Product Data Elements to Include in the OV-3

The policy document calls for these data elements:

- Interoperability Key Performance Parameter of each information exchange, with threshold and objective values
- Criticality of the information exchange
- For all information exchanges, all architecture data elements that are required by the Chairman of the Joint Chiefs of Staff Instruction (CJCSI) 6212.01B
- For all information exchanges, the architecture data elements from the Systems Data Exchange Matrix (SV-6) (i.e., combine OV-3 with SV-6)
- In addition, the Framework recommends including all of the architecture data elements annotated by an asterisk in the Data Element Definition Table for OV-3
- The relationship between the views is critical to the analysis required in chapter 4 of the C4ISP. To make this analysis more visible, the numbering of information exchanges and needlines is critical. OV-3 should include a column that has a unique number for each information exchange that relates to a needline on the OV-2

Special Product Tailoring Advice

- The policy document requires that the information exchange requirements (IERs) from the relevant requirements documents be included in the C4ISP.
- The policy document requires that information exchanges shown on OV-3 be at the level of IERs specified in requirements documents. (The Framework itself allows aggregation of information exchanges in OV-3 when that is appropriate for the purpose of the architecture.)
- The policy document requires that SV-6 be appended to OV-3. However, the Framework recommends that these products be developed separately. SV-6 architecture data are usually not available at the time of development of an OV-3, because SV-6 is based on, and derived from, architecture data identified in OV-3, SV-1, and SV-4.

Suggested Universal Reference Resources to Use in Building the Product

- The GIG Architecture describes information exchanges associated with selected JMAs and may be applicable to the information exchanges of the C4ISP.
- The UJTL describes the structure of Joint tasks and can be mapped to the functional areas/tasks/activities associated with the sending and receiving nodes of the information exchanges.

3.5.2.6 Operational Activity Model (OV-5) (Highly Applicable)

OV-5 is not required by the policy document, but the Framework recommends it. OV-5 is also one of the products specified by the Framework as required for an integrated architecture. The policy document requires that derived requirements be obtained through "hierarchical decomposition of the operational tasks performed by the system being developed." Interpreting the term *system* broadly, this kind of information is part of an activity model. In addition, an OV-5 provides the basis for activities to be referenced in OV-2 and OV-3. Additionally, OV-5 forms the basis for identifying the activity relationships used in developing OV-6c, which is required by the policy document. When used in conjunction with OV-6c, it describes capabilities required.

Specific Product Data Elements to Include in the OV-5

The Framework recommends including at least the data elements that are annotated with an asterisk in the Data Element Definition Table for OV-5.

Special Product Tailoring Advice

The Framework recommends the development of both an overarching activity model that covers all of the missions or functional areas associated with the C4ISP and the development of individual mission area-specific or functional area-specific models as variants on that overall model. This allows the analyst to see commonality among, and differences between, relevant missions/functional areas.

Suggested Universal Reference Resources to Use in Building the Product

- The GIG Architecture describes selected JMAs that may be mapped to the mission areas covered by the proposed system.
- The UJTL describes the structure of Joint tasks and can be mapped to the functional areas/tasks/activities associated with the proposed system.

3.5.2.7 Operational Event-Trace Description (OV-6c) (Highly Applicable)

OV-6c is recommended by the policy document if it appears to be needed for a given C4ISP (i.e., when it is needed to clarify the time-critical nature of information for each mission). The Framework also lists OV-6c as highly applicable for use in developing C4ISPs. OV-6c depicts the dynamic behavior of the mission process with timing and sequencing attributes (i.e., it depicts operational threads, and can contribute to establishing operational performance requirements).

Specific Product Data Elements to Include in OV-6c

The Framework recommends including at least the data elements that are annotated with an asterisk in the Data Element Definition Table for OV-6c.

Special Product Tailoring Advice

- If scenarios are identified in OV-3 of a given C4ISP, the Framework recommends that, at the least, an OV-6c be developed for each of these scenarios. This will help relate the two products and clearly delineate the timing and sequencing aspects of the information exchanges.
- Alternatively, a number of OV-6c's can be developed that show specific paths through the activities of OV-5.

Suggested Universal Reference Resources to Use in Building the Product

- The GIG Architecture can be referenced for a description of relevant mission areas that can serve as input to the OV-6c sequences.
- The UJTL can be referenced for mapping to the appropriate operational activities that form the foundation for an OV-6c.

3.5.2.8 Systems Interface Description (SV-1) (Highly Applicable)

The SV-1 is required by the policy document. SV-1 is also one of the products specified by the Framework as required for an integrated architecture and is listed as highly applicable for use in developing C4ISPs. A notional SV-1 is provided in **Figure 3.5-4**.

Specific Product Data Elements to Include in SV-1

The Framework recommends including at least the data elements annotated with an asterisk in the Data Element Definition Table for SV-1. It is especially important in a C4ISP to relate operational needlines (from an OV-2) to their corresponding SV-1 interface. The Framework recommends that the Key Interface designation and Key Interface rationale be documented for the applicable SV-1 interfaces. In addition, the Framework recommends that supported operational activities and supported operational capability also be related to system functions and systems, respectively.

Special Product Tailoring Advice

The policy document states that the C4ISP is intended to be a living document that increases in detail as the proposed system moves through the milestones. The SV-1 should start out with the internodal version (node edge-to-node edge), and should evolve through the internodal version (system interconnections), the intranodal version, and the intrasystem version, as appropriate, to the nature of the systems.

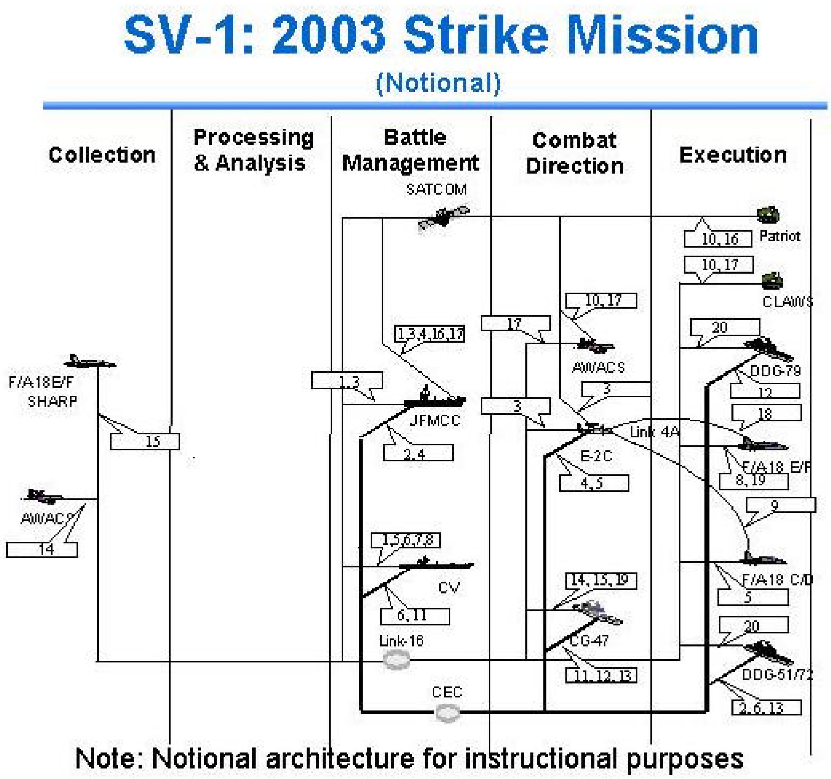


Figure 3.5-4. Notional SV-1

Suggested Universal Reference Resources to Use in Building the Product

- SV-1 should be compliant with the systems shown in the GIG Architecture.
- SV-1 system standards should be consistent with the Joint Technical Architecture (JTA).
- SV-1 systems and software should be consistent with the Common Operating Environment (COE).

3.5.2.9 Systems Functionality Description (SV-4) (Often or Partially Applicable)

SV-4 is not required by the policy document, but the Framework lists it as often or partially applicable for use in developing C4ISPs. SV-4 supports identification of a hierarchy of required system functions. System functions provide a basis for assessing various approaches for achieving a capability via a materiel approach. SV-4 can also be used to identify and document required system data elements whose exchange attributes are described in SV-6.

Specific Product Data Elements to Include in SV-4

The Framework recommends including the system functions and system data flows between them. Data flows form the systems data elements that are documented in SV-6 system data exchanges. The Framework also recommends including the data elements that are annotated with an asterisk in the Data Element Definition Table for SV-4.

Special Product Tailoring Advice

N/A

Suggested Universal Reference Resources to Use in Building the Product

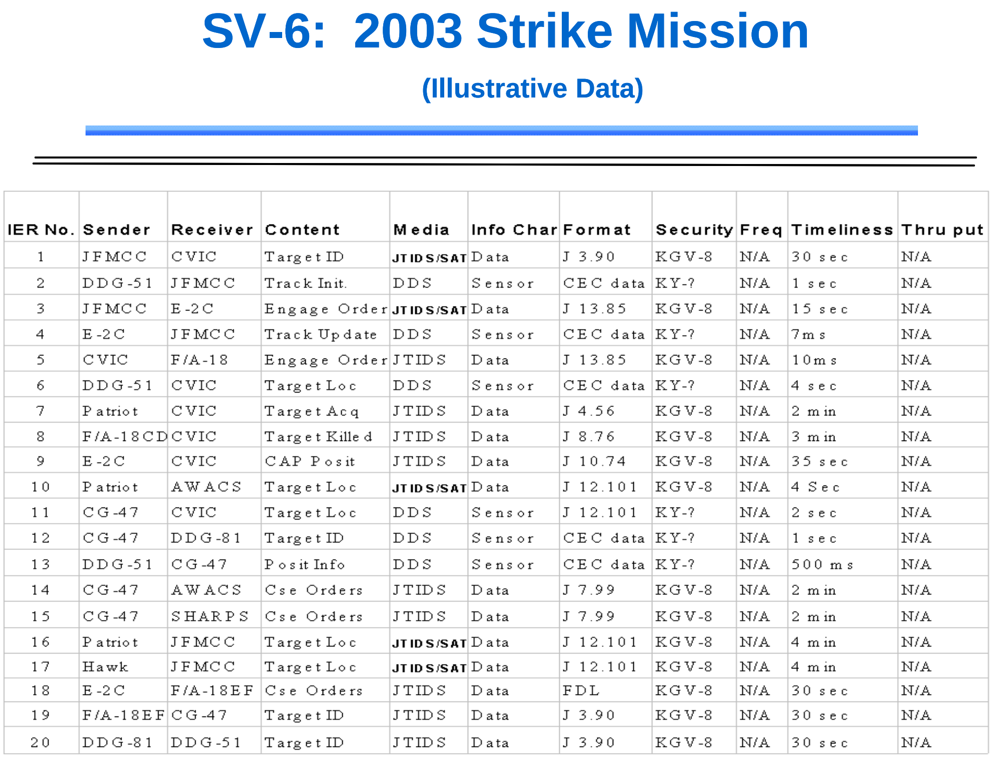
- SV-4 system functions should be consistent with any related system functions shown in the GIG Architecture.
- SV-4 human computer interface (HCI) and graphical user interface (GUI) function standards should be consistent with the JTA.

3.5.2.10 Systems Data Exchange Matrix (SV-6) (Highly Applicable)

The SV-6 is required by the policy document. SV-6 is also listed by the Framework as highly applicable for use in developing C4ISPs. A notional SV-6 is provided in **Figure 3.5-5**.

Specific Product Data Elements to Include in SV-6

The Framework recommends including at least the data elements annotated with an asterisk in the Data Element Definition Table for SV-6. It is especially important to indicate the system interface (from SV-1) that denotes the system data exchange on SV-1. In this way, a system data exchange is linked, through the system interface, back to the operational needline.



SV-6: 2003 Strike Mission

(Illustrative Data)

IER No.	Sender	Receiver	Content	Media	Info Char	Format	Security	Freq	Timeliness	Thru put
1	JFMCC	CVIC	Target ID	JTIDS/SAT	Data	J 3.90	KGV-8	N/A	30 sec	N/A
2	DDG-51	JFMCC	Track Init.	DDS	Sensor	CEC data	KY-?	N/A	1 sec	N/A
3	JFMCC	E-2C	Engage Order	JTIDS/SAT	Data	J 13.85	KGV-8	N/A	15 sec	N/A
4	E-2C	JFMCC	Track Update	DDS	Sensor	CEC data	KY-?	N/A	7ms	N/A
5	CVIC	F/A-18	Engage Order	JTIDS	Data	J 13.85	KGV-8	N/A	10ms	N/A
6	DDG-51	CVIC	Target Loc	DDS	Sensor	CEC data	KY-?	N/A	4 sec	N/A
7	Patriot	CVIC	Target Acq	JTIDS	Data	J 4.56	KGV-8	N/A	2 min	N/A
8	F/A-18CD	CVIC	Target Killed	JTIDS	Data	J 8.76	KGV-8	N/A	3 min	N/A
9	E-2C	CVIC	CAP Posit	JTIDS	Data	J 10.74	KGV-8	N/A	35 sec	N/A
10	Patriot	AWACS	Target Loc	JTIDS/SAT	Data	J 12.101	KGV-8	N/A	4 Sec	N/A
11	CG-47	CVIC	Target Loc	DDS	Sensor	J 12.101	KY-?	N/A	2 sec	N/A
12	CG-47	DDG-81	Target ID	DDS	Sensor	CEC data	KY-?	N/A	1 sec	N/A
13	DDG-51	CG-47	Posit Info	DDS	Sensor	CEC data	KY-?	N/A	500 ms	N/A
14	CG-47	AWACS	Cse Orders	JTIDS	Data	J 7.99	KGV-8	N/A	2 min	N/A
15	CG-47	SHARPS	Cse Orders	JTIDS	Data	J 7.99	KGV-8	N/A	2 min	N/A
16	Patriot	JFMCC	Target Loc	JTIDS/SAT	Data	J 12.101	KGV-8	N/A	4 min	N/A
17	Hawk	JFMCC	Target Loc	JTIDS/SAT	Data	J 12.101	KGV-8	N/A	4 min	N/A
18	E-2C	F/A-18EF	Cse Orders	JTIDS	Data	FDL	KGV-8	N/A	30 sec	N/A
19	F/A-18EF	CG-47	Target ID	JTIDS	Data	J 3.90	KGV-8	N/A	30 sec	N/A
20	DDG-81	DDG-51	Target ID	JTIDS	Data	J 3.90	KGV-8	N/A	30 sec	N/A

Note format differences between OV-3 and SV-6

Figure 3.5-5. Notional SV-6

Special Product Tailoring Advice

- The policy document requires that SV-6 be appended to OV-3. However, the Framework recommends that these products be developed separately. SV-6 architecture data are usually not available at the time of development of an OV-3, because SV-6 is based on, and derived from, architecture data identified in OV-3, SV-1, and SV-4.
- The relationship between the views is critical to the analysis required in chapter 4 of the C4ISP. To make this analysis more visible, the numbering of information exchanges and needlines is critical. SV-6 should include a column that has a unique number for each information exchange that relates to an interface on SV-1.

Suggested Universal Reference Resources to Use in Building the Product

- SV-6 should be consistent with the systems nodes and systems shown in the GIG Architecture.
- SV-6 data exchange standards should be consistent with the JTA.
- SV-6 "interoperability level achievable" can be expressed in terms of the levels described in Levels of Information System Interoperability (LISI).

3.5.2.11 Systems Performance Parameters Matrix (SV-7) (Often Applicable)

SV-7 is not required by the policy document, but the Framework recommends it as often or partially applicable for use in developing C4ISPs. The policy document requires "relevant specific system and component performance parameters such as reliability, maintainability, and availability." The Framework calls for this information in SV-7.

Specific Product Data Elements to Include in SV-7

The Framework recommends considering at least the data elements annotated with an asterisk in the Data Element Definition Table for SV-7. However, the actual parameters selected will depend on the nature of the system for which the C4ISP is built.

Special Product Tailoring Advice

N/A

Suggested Universal Reference Resources to Use in Building the Product

N/A

3.5.2.12 Technical Standards Profile (TV-1) (Highly Applicable)

TV-1 is specified by the policy document. TV-1 is also one of the products specified by the Framework as required for an integrated architecture and is listed as highly applicable for use in developing C4ISPs. A notional TV-1 is provided in **Figure 3.5-6**.

Specific Product Data Elements to Include in the TV-1

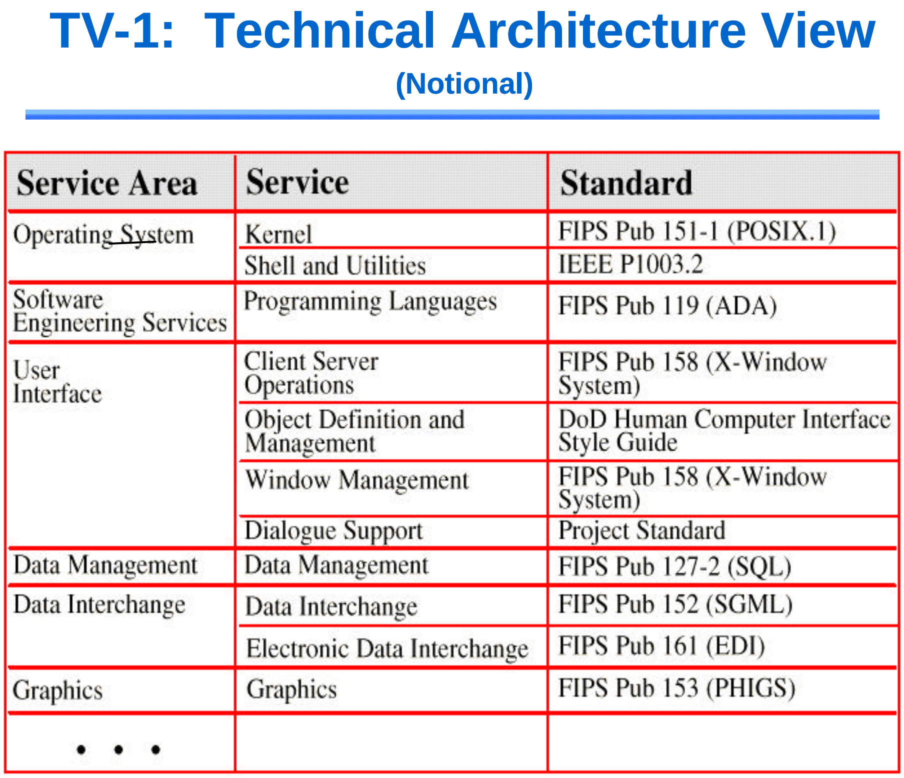
The Framework recommends considering at least the data elements annotated with an asterisk in the Data Element Definition Table for the TV-1.

Special Product Tailoring Advice

For multiyear C4ISPs, information from a Technical Standards Forecast (TV-2) should be added to show the expected evolution of standards coincident with the development of the system.

Suggested Universal Reference Resources to Use in Building the Product

The JTA can be referenced for applicable standards.



TV-1: Technical Architecture View

(Notional)

Service Area	Service	Standard
Operating System	Kernel	FIPS Pub 151-1 (POSIX.1)
	Shell and Utilities	IEEE P1003.2
Software Engineering Services	Programming Languages	FIPS Pub 119 (ADA)
User Interface	Client Server Operations	FIPS Pub 158 (X-Window System)
	Object Definition and Management	DoD Human Computer Interface Style Guide
	Window Management	FIPS Pub 158 (X-Window System)
	Dialogue Support	Project Standard
Data Management	Data Management	FIPS Pub 127-2 (SQL)
Data Interchange	Data Interchange	FIPS Pub 152 (SGML)
	Electronic Data Interchange	FIPS Pub 161 (EDI)
Graphics	Graphics	FIPS Pub 153 (PHIGS)
• • •		

Figure 3.5-6. Notional TV-1

3.5.2.13 Technical Standards Forecast (TV-2) (Often or Partially Applicable)

TV-2 is not required by the policy document, but the Framework recommends it as often or partially applicable for use in developing C4ISPs (applicable for multi-year C4ISPs). See discussion of TV-1.

3.5.2.14 Relationship of Architecture Products for C4ISP Analysis

Figure 3.5-7 depicts the relationship between architecture products required in the policy document for C4ISPs.

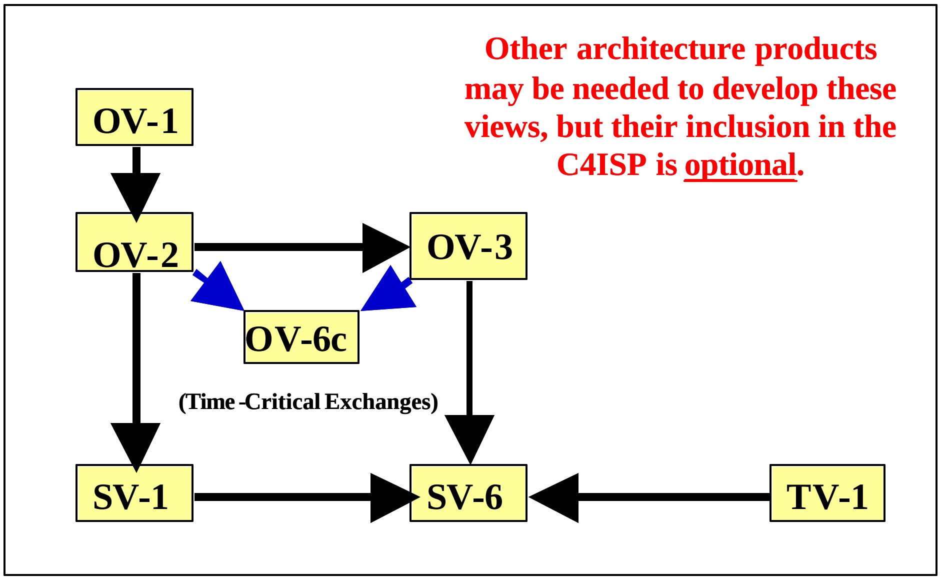

Other architecture products may be needed to develop these views, but their inclusion in the C4ISP is optional.
OV-1
OV-2
OV-3
OV-6c
(Time -Critical Exchanges)
SV-1
SV-6
TV-1

Figure 3.5-7. Relation of Architecture Products for C4ISP Analysis

3.5.3 References

Chairman, Joint Chiefs of Staff Instruction, CJCSI 6212.01B, *Interoperability and Supportability of National Systems and Information Technology Systems*, May 8, 2000.

Chairman, Joint Chiefs of Staff Manual, CJCSM 3500.04C, *Universal Joint Task List*, July 1, 2002.

Defense Information Systems Agency, *Joint Technical Architecture*, Version 4.0, July 17, 2002.

Office of the Assistant Secretary of Defense for Command, Control, Communications and Intelligence (OASD[C3I]), Deputy Assistant Secretary of Defense for C3I Surveillance and Space (DASD[C3ISR and Space]), *Implementing the C4I Support Plan Requirements*, Briefing, March 2002.

DoD, *Interim Defense Acquisition Guidebook*, October 30, 2002 (Formerly DoD Regulation 5000.2-R, *Mandatory Procedures for Major Defense Acquisition Programs [MDAP] and Automated Information Systems [MAIS] Acquisition Programs*, April 5, 2002).

3.6 THE ROLE OF ARCHITECTURES IN CAPITAL PLANNING AND INVESTMENT CONTROL (CPIC)

3.6.1 Introduction

The Clinger-Cohen Act (CCA)[1] requires the head of each executive agency to design and implement a process for maximizing the value, and assessing and managing the risks of information technology (IT) acquisitions and the development of the IT Architecture. Today, the process has evolved into the Capital Planning and Investment Control (CPIC) process and the IT Architecture has evolved into the enterprise architecture (EA) as amplified by OMB Circular A-130[2] and other guidance. The EA is, in part, a management tool to guide investment decisions.

3.6.2 Overview of CPIC

The goal of a CPIC process is to link mission needs, information, and IT effectively and efficiently. A basic CPIC process, as described in A-130, has three phases: Select, Control, and Evaluate.

- **Select** investments with demonstrated return-on-investment (ROI), benefit-cost analysis, proper planned oversight mechanisms, maximum usefulness, and other qualities aligned with the enterprise architecture and strategic plans as part of a managed investment portfolio. An executive management investment review board makes the selections.
- **Control** or manage the investments through their development and implementation or acquisition. Achieve control by measuring and monitoring actual performance against expected performance, meeting milestones and user requirements and expectations, providing security protection, managing risks, following enterprise architecture procedures, having periodic oversight reviews, and otherwise controlling the investment implementation. Make a continue/modify/terminate decision at each milestone review.
- **Evaluate** the results of the investment after implementation to assess the benefit-cost achieved compared to the benefit-cost expected, evaluate the ROI to make a continue/modify/terminate decision on continuing with the investment, document lessons learned, redesign processes where needed, reassess the business case, technical compliance, and EA compliance, and update the EA and CPIC processes.

The Select, Control, and Evaluate Phases form a continuous cycle with assessments from each phase feeding the next phase as shown in **Figure 3.6-1**.

Some agencies have added other phases such as mission assessment or pre-selections that prepare information for the selection process and a steady state that continuously monitors each investment.

[1] Information Technology Management Report Act (*aka* Clinger-Cohen Act) (Public Law 104-106) February 10, 1996.

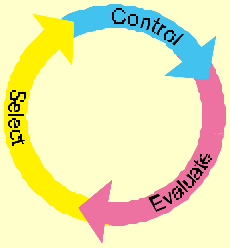
[2] OMB Circular A-130, *Management of Federal Information Resources*, November 30, 2000.

Select

1. Support core government mission functions
2. No private sector alternative
3. Work processes redesigned
4. Avoid custom components
5. Demonstrate = or better ROI
6. Have Benefit Cost Analysis (BCA)
7. Have IT investment portfolio
8. Consistency with EAs (federal, agency bureau)
9. No duplication of IT capability
10. Max usefulness, min public burden, preserve integrity, usability, availability, confidentiality
11. Oversight mechanisms
12. Not restrict state, local, tribal governments
13. Facilitate accessibility for disabled

Control

1. Performance measures and monitor actual against expected
2. Periodic oversight review for changed requirements, results, performance, interoperability, maintenance
3. Proceed timely, agreed milestones, in life cycle, meet expectations, deliver benefits, meet user requirements, provide security protection
4. Risk mitigation strategy
5. Financial Management Systems conform to OMBA-127
6. Provide management controls for the disposition of records
7. Follow Enterprise Architecture procedures



Evaluate

1. Post implementation benefits cost assessment, document effective management practices
2. Evaluate systems for ROI, continue/modify/terminate decision
3. Document lessons learned and redesign processes and performance levels
4. Re-assess business case, technical compliance, and EA compliance
5. Update EA and IT Capital Planning processes

Figure 3.6-1. CPIC Has Select, Control, and Evaluate Phases, Goal – Link Mission Needs, Information, and IT Effectively and Efficiently

3.6.3 Overview of Enterprise Architecture

OMB A-130 defines *enterprise architecture* as "the explicit description and documentation of the current and desired relationships among business and management processes and information technology." The EA includes principles, an EA framework, a standards profile, current and target architectures, and a transition strategy to move from the current to target architecture (see **Figure 3.6-2**).

The EA defines principles and goals and sets direction on key issues. An EA framework organizes architecture information areas. The EA framework can also identify the product types needed to document the EA and show how to portray linkages between mission needs, business processes, and IT functionalities. (The DoD Architecture Framework [DoDAF] products can be used to describe the EA as suggested below.) Having a framework with associated product types helps structure and manage the EA effort. Using the same framework and product specifications across different but related EAs increases the comparability of the EAs and facilitates communications among the architects working on the different EAs. The DoDAF is an example of an EA framework.

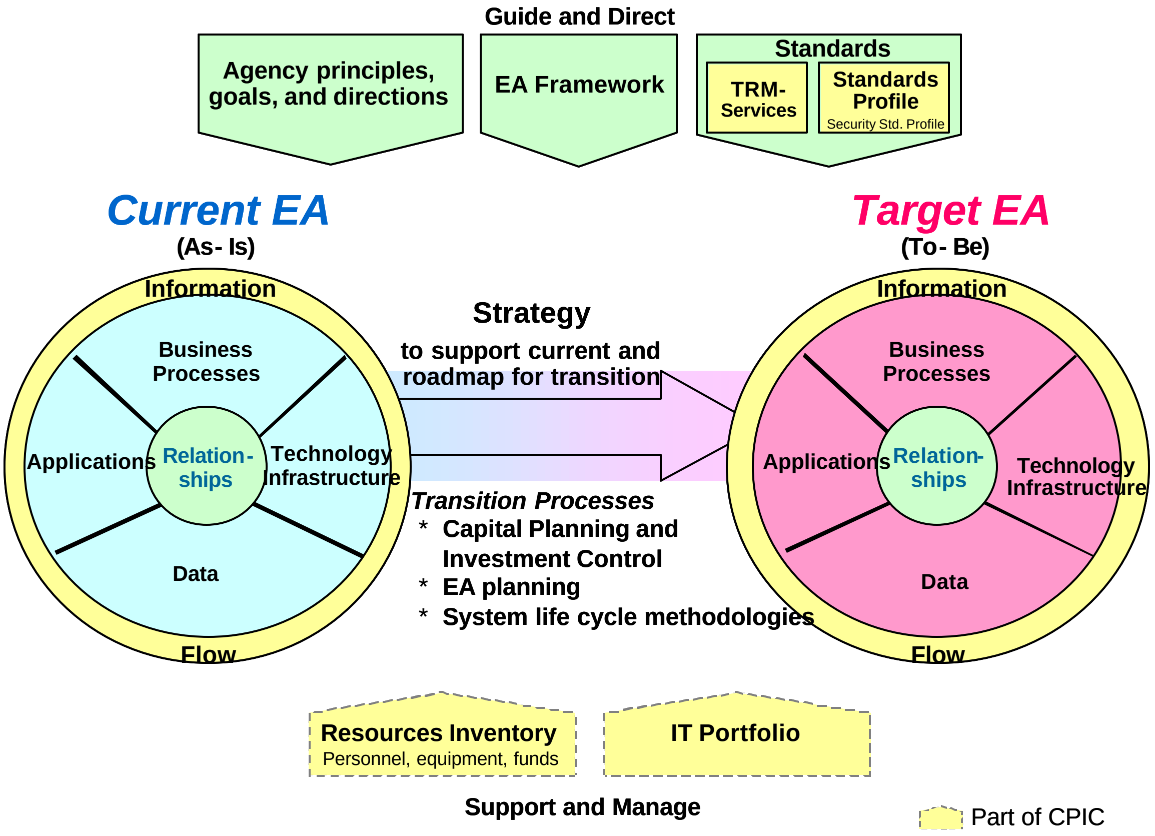


Figure 3.6-2. OMB A-130 Enterprise Architecture Elements

To describe the information services used throughout the agency, the EA includes a Technical Reference Model (TRM). The associated Standards Profile, including its Security Standards Profile, defines the set of IT standards that support the services articulated in the TRM. The Standards Profile guides the implementation of the EA by defining standards to be used.

The EA includes a current and target architecture (including "the rules and standards and systems life-cycle information to optimize and maintain the environment that the agency wishes to create and maintain by managing its IT portfolio"). The current and target architectures include business processes, information flow and relationships, applications, data descriptions and relationships, and technology infrastructure. The target architecture should support the strategic goals and lead to the vision.

The EA also includes a transition strategy to enable support for the current environment and provide a roadmap for transition to the target environment. The transition strategy phasing addresses not only the creation of new functionalities and systems, but retiring existing old systems. To effectively achieve the transition, the agency must have CPIC processes, EA planning processes, and systems life-cycle methodologies in place.

To support the EA, an agency must have an inventory of information resources (e.g., personnel, equipment, funds) devoted to IT. To create the target environment, the agency must have and manage, through its CPIC selection and control processes, a portfolio of major information systems.

3.6.4 Relating DoD Architecture Framework Products to the EA

The business processes, data, applications, and infrastructure of the current or target enterprise architecture can be described using several DoDAF products. For example, the Operational Activity Model (OV-5) provides a way to present business processes and relate them to information flows and the organizations and systems as mechanisms that support them. The Operational Node Connectivity Description (OV-2) can relate operational roles, activities performed at a node, and the need for and types of information passing among nodes. The Systems Interface Description (SV-1) provides the high-level view of the computing systems, while other products, such as the Systems Functionality Description (SV-4), provide more detailed descriptions. The Technical Standards Profile (TV-1) provides the standards with which new implementations should comply. **Figure 3.6-3** shows a partial mapping of DoDAF products to the elements of an EA required by OMB A-130.

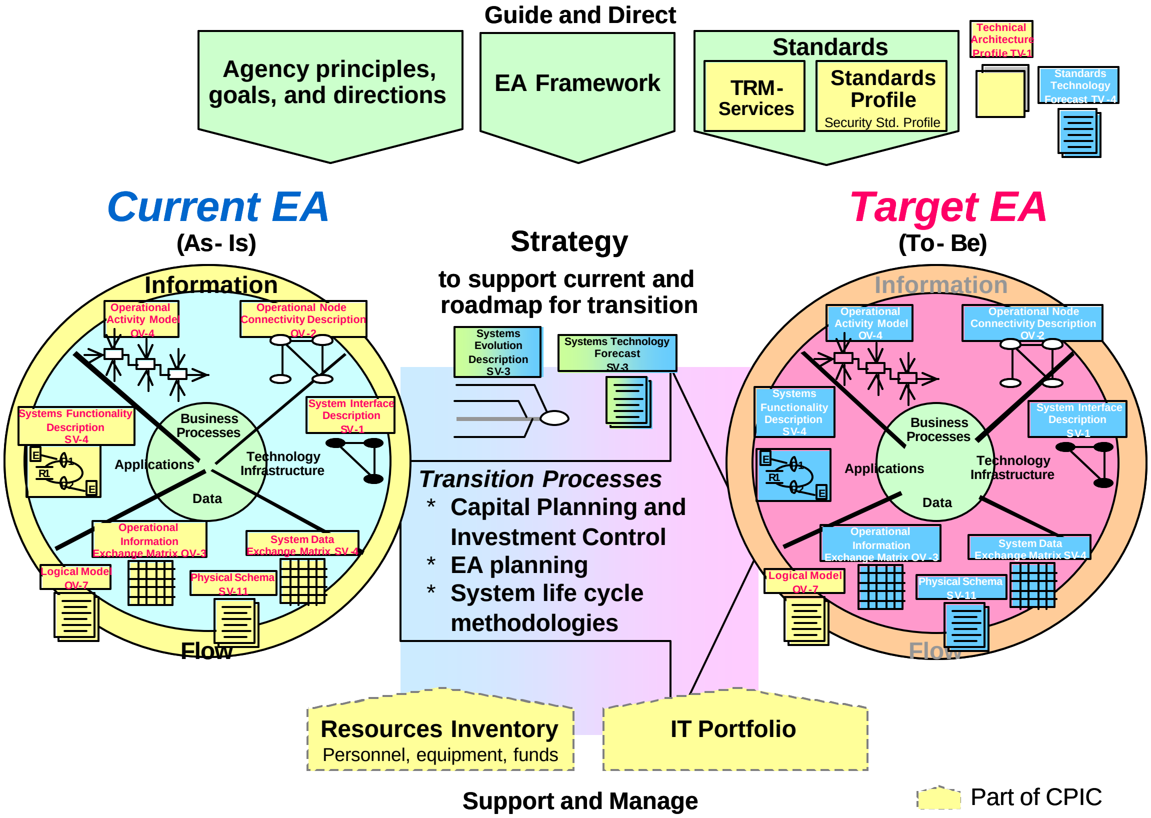

Guide and Direct
Agency principles, goals, and directions
EA Framework
Standards
TRM-Services
Standards Profile
Security Std. Profile
Technical Architecture Profile TV-1
Standards Technology Forecast TV-4
Current EA
(As- Is)
Information
Operational Activity Model OV-4
Operational Node Connectivity Description OV-2
Systems Functionality Description SV-4
System Interface Description SV-1
Business Processes
Applications
Technology Infrastructure
Data
E
R1
E
Operational Information Exchange Matrix OV-3
System Data Exchange Matrix SV-4
Logical Model OV-7
Physical Schema SV-11
Flow
Strategy
to support current and roadmap for transition
Systems Evolution Description SV-3
Systems Technology Forecast SV-3
Transition Processes
* Capital Planning and Investment Control
* EA planning
* System life cycle methodologies
Target EA
(To- Be)
Information
Operational Activity Model OV-4
Operational Node Connectivity Description OV-2
Systems Functionality Description SV-4
System Interface Description SV-1
Business Processes
Applications
Technology Infrastructure
Data
E
R1
E
Operational Information Exchange Matrix OV-3
System Data Exchange Matrix SV-4
Logical Model OV-7
Physical Schema SV-11
Flow
Resources Inventory
Personnel, equipment, funds
IT Portfolio
Support and Manage
Part of CPIC

Figure 3.6-3. OMB A-130 Enterprise Architecture Elements with Applicable Framework Products Overlay

3.6.5 Using the Enterprise Architecture in the CPIC Process

An EA and investment decisions are driven by the mission and vision statement and strategic plans of the enterprise. The mission and vision statement describes where the agency wants to be in the future. The vision guides the definition of the business processes, data, and IT systems needed to achieve the vision (i.e., the vision guides) and is the realization of the target architecture. To get from the existing, or current, business processes and IT systems (i.e., the current architecture) to the target, the transition strategy identifies a sequence or phases of new functionalities and systems and retirement of old systems that move toward the target.

Implementing the target requires investments consistent with the transition strategy. The selection of the best investments to realize the target architecture and thus the goals of the strategic plan is one of the objectives of CPIC.

3.6.6 Using the EA and Framework Products During the Select Phase

Fundamental to the selection process is the creation of a sound business case that describes the investment, explains how it produces effectiveness and efficiency gains, and contains ROI, benefit-cost, risk, and performance measures information. However, a decision to invest in a project is not an isolated decision based only on ROI or benefit-cost. Investment decisions consider strategic goals, project dependencies, competition for the same resources, and other factors. As part of the selection process, an agency develops a set of selection criteria. Compliance with the EA is one of the criteria required for CPIC.

There are several ways in which a proposed investment must comply with the EA. The first question to ask is, "Is the business need for the investment clear?" This question can be answered by using the High-Level Operational Concept Graphic (OV-1), OV-5, SV-4, and the Operational Activity to Systems Function Traceability Matrix (SV-5). OV-1 describes the enterprise's business missions, processes, and organizations at a high level. OV-5 details the business processes and the information flows between the organizations that conduct these business processes. SV-4 documents system functional hierarchies and system functions that support the business processes documented in OV-5. The mapping between systems and business processes is explicitly documented in SV-5.

The proposed investment should comply with the principles and standards of the EA. Complying with principles such as use of component-based architectures, maximizing use of commercial-off-the-shelf (COTS), contracting out services wherever possible, single data capture, data residency requirements, user access functionalities, and security and privacy considerations affect the approach to and design of new business processes and systems. The technical standards are documented in TV-1, and the compliance with standards can be demonstrated by using various Systems View (SV) products including SV-1, the Systems Communication Description (SV-2), SV-4, the Systems Data Exchange Matrix (SV-6), and the Physical Schema (SV-11). The compliance with system and system component related standards are documented using SV-1. Compliance with the communications system and network related standards are documented using the SV-2. Compliance with system function related standards is documented using SV-4. Compliance with data exchange related standards is documented using SV-6. Compliance with physical data schema-related standards is documented in SV-11.

Interoperability requirements for new operational capabilities that lead to new system functionality should be identified. The boundaries and interfaces should align well with those identified in the EA to facilitate the transitions needed as systems come on board or are retired. The systems' (or new system functionalities) interoperability requirements can be identified by using SV-1 and SV-6. SV-1 describes the interfaces between systems nodes that support the necessary interactions between key players described in OV-2 in order to conduct the business processes described in OV-5. SV-6 specifies the characteristics and requirements of the data exchanges between systems that automate the information exchanges between key players described in the Operational Information Exchange Matrix (OV-3). OV-3 details each information exchange between the key players, describing who exchanges what information with whom, why the information exchange is necessary, and how the information exchange must occur.

The proposed investment should appear as a need or project in the EA transition strategy and be linked to achieving some part of the strategic plan and associated goals. The investment should be made in a sequence consistent with the EA transition plan. All new system functionality should be funded and should progress with reasonable management effectiveness and risk factors. For example, future investment strategy or future critical capabilities may be dependent on funding certain system functionality in the present. This future investment need or dependency on current funding is a factor to be considered in a transition strategy. If the investment replaces an existing legacy investment, plans for the retirement of the legacy system should appear in the EA transition strategy, be funded, and be timed to match the needs of the proposed investment. The EA transition strategy and plan can be identified by using the Systems Evolution Description (SV-8) and Systems Technology Forecast (SV-9). SV-8 describes how systems will evolve from the current architecture to the target architecture over a period of time with milestones and timelines. Funding and evolution goals associated with these milestones and timelines are dependent on the availability of future technology as forecasted in SV-9.

If the investment being considered affects organizations outside the scope of the EA, consistency with the EA of the other organizations needs to be considered. For example, the U.S. Marine Corps EA and investments must be consistent with the Navy EA. Using the same DoDAF products to document related EAs increases the comparability of the EAs, thus allowing consistency among the related EAs to be verified easily.

The executive management investment review board uses EA-related and other investment selection criteria to make the final investment decisions. They rank the candidates in an investment portfolio and identify those that will be funded.

3.6.7 Using the EA and Framework Products During the Control Phase

During the CPIC Control Phase, executives systematically examine the management and progress of IT investment projects consistent with the Systems Development Life Cycle (SDLC) or acquisition life cycle, the milestone reviews of the project, and other management practices. Early in the project, the plans and procedures for the project are examined. The schedule, budget, delivery, and risks are continually reviewed for their agreement with plans and measures in the business case. At each milestone review, a decision to continue, modify, or terminate the project is made. To continue, a project must be within acceptable budget, schedule, benefit, and risk parameters, continue to meet strategic needs, align with the EA, and meet other criteria specified by service requirements or the SDLC process.

The EA compliance criteria identified for the Select Phase still apply at the Control Phase. If the strategic goals and vision have changed, the continuing need for the project should be verified. As the approach, design, and implementation of the project become available in more detail, they should be compared for consistency with the principles, standards, boundaries, and interfaces of the EA. If changes to the EA have been made, the project should be reviewed for compliance with the changes. The usage of the Framework products to support the Select Phase, as described in the previous section, also applies in support of the Control Phase.

The progress of any upstream or downstream projects related to the project being reviewed should be assessed to be certain the needed functionalities will be available at the proper time. Any delay, business process, or technical design impacts from changes in preceding or successor projects should be evaluated.

3.6.8 Updating the EA and Framework Products During the Evaluate Phase

The Evaluate Phase of CPIC compares the actual results achieved with the expected results described in the business case to assess the investment made in the project and to improve the methodologies used to project results and perform CPIC and EA. The new system or functionality needs to be operational for several months before the assessment can be made to allow collection of data on the system performance, costs, and benefits. The review, like other reviews, considers whether the system is still aligned with strategy and compliant with the EA. The review also assesses whether the operational system is delivering the benefits and has the operational costs expected. Based on these and other considerations, the decision could be made to terminate or modify even a new system if it is not performing as expected, or the need for it has changed.

Part of the Evaluate Phase is to document lessons learned and effective management practices, and to modify processes if needed. The assessment of the actual performance, cost, and benefit data compared to business case projections may indicate a need to revise the projection methods used in future business cases. The analysis may also indicate a need for change in the CPIC process such as in investment selection criteria, risk considerations, or milestone review processes.

The EA should be updated to indicate the new system or functionality in the current architecture, and to address any dependency or other issues in the transition strategy. The EA process should be revised based on any lessons learned from the project. The changes in business missions and processes should be reflected by updating the Operational View (OV) products including but not limited to OV-1, OV-2, OV-3, and OV-5. The new system functionality and their compliance with standards in the current architecture should be reflected by updating the SV products including but not limited to SV-1, SV-4, SV-5, and SV-6. The changes in the transition strategy and plan due to changes in the business environment, changes in technologies, and lessons learned should be reflected by updating SV-8 and SV-9. The addition of newly required technical standards and retirement of obsolete standards due to changes in technologies should be reflected by updating TV-1.

3.6.9 References

Buck, Kevin, and Kahn, Susan, MITRE Corporation, *IT Investment Management Framework*, Draft, Available: http://www.whitehouse.gov/omb/circulars/a130/a130trans4.html, January 2003.

Federal CIO Council, *Smart Practices in Capital Planning*, Available: http://www.cio.gov/Documents/smart_practices_book.pdf, October 2000.

McVay, William H, Office of Management and Budget, *Critical Success Factors for Effectively Using Clinger Cohen Act (CPIC, EA, Project Management),* Available: http://www.cio.gov/Documents/OMBA11_CSF_0410.ppt, 2001.

Office of Management and Budget, *Evaluating Information Technology Investments*, Available: http://www.cio.gov/documents/omb_evaluating_it_investments.html, January 20, 2001.

Office of Management and Budget Circular A-130, *Management of Federal Information Resources*, November 30, 2000.

Division E of Public Law 104-106, *Information Technology Management Reform Act of 1996*, August 8, 1996.

U.S. Department of Agriculture (USDA), Office of the Chief Information Officer, *USDA Information Technology Capital Planning and Investment Control Guide*, Available: www.ocio.usda.gov, April 2002.

An expanded list of references and their description is available at:

U.S. Department of Health and Human Services (HHS), *HHS Information Resource Management (IRM) Policy for Capital Planning and Investment Control*, (HHS-IRM-2000-0001), Available: http://www.hhs.gov/read/irmpolicy/0001.html, January 8, 2001.

4 ADDITIONAL INFORMATION

4.1 INTRODUCTION

This section provides additional material on topics relevant for developing architecture descriptions. These topics include the representation of architecture concepts in the All DoD Core Architecture Data Model (CADM), an approach for assessing architecture modeling and repository tools, an overview of Federal Enterprise Architecture (FEA) Reference Models (RMs), and a discussion of Universal Reference Resources (URRs). URRs are reference models and information standards that serve as sources for guidelines and attributes that should be consulted while building architecture products.

4.2 ARCHITECTURAL CONCEPTS AND CADM

4.2.1 Introduction

The All-DoD Core Architecture Data Model (CADM) provides a common approach for organizing and portraying the structure of architecture information. The CADM was initially published in 1997 as a logical data model for architecture data. It was revised in 1998 to meet all the requirements of the C4ISR Architecture Framework Version 2.0.[1] As a logical data model, the initial CADM provided a conceptual view of how architecture information is organized. It identified and defined entities, attributes, and relations. The CADM has evolved since 1998, so that it now has a physical view providing the data types, abbreviated physical names, and domain values that are needed for a database implementation. Because the CADM is also a physical data model, it constitutes a database design and can be used to automatically generate databases. Implementations of the CADM for Microsoft SQL Server 2000® and Access 2000®, as well as Oracle 9i®, have been created in this way. Many other implementations (such as for the Army Systems Architecture) provide configuration management for a separate physical schema that merges some of the CADM entities for reasons of improved performance.

This section introduces some basic architectural data concepts, so that users and developers can gain insight into the concepts (not logical structures) underlying the CADM. The basic architectural data elements identified and discussed in this section are intentionally chosen from the points of view of users and developers and are not in one-to-one correspondence with the entities of the current All-DoD CADM.[2] Section 4.2.5 provides a table of the correspondence between data concepts discussed in this section and CADM entities.

Note that the concepts and basic architecture data elements described below are not in one-to-one correspondence with the data elements noted in Volume II of the DoD Architecture Framework (DoDAF) or with the CADM. Therefore, this paper should be used as a tutorial for identifying and illustrating architecture concepts underlying the CADM. The focus here is on understanding the CADM.

[1] Documented in the two volumes of the C4ISR Core Architecture Data Model (CADM) Version 2.0 (CADM 2.0), Final Report (ASD[C3I]), 1998.

[2] The All-DoD CADM will be documented in a three-volume report, All-DoD Core Architecture Data Model (All-CADM) for DoD Architecture Framework Version 1.0, Volume 1, Overview Description; Volume 2, Technical Specification; and Volume 3, Annexes.

4.2.2 Basic Architectural Elements

An architecture data repository responsive to the architecture products of the DoDAF contains information on basic architectural elements such as the following:

- Operational nodes may be organizations, organization types, and operational (human) roles. (A role may be a skill, occupation, occupational specialty, or position.).
- Operational activities including tasks defined in the Universal Joint Task List (UJTL).
- Information and data refers to information provided by domain databases and other information asset sources (which may be network centric) and systems data that implement that information. These information sources and systems data may define information exchanges or details for system interfaces.
- Systems nodes refers to nodes associated with physical entities as well as systems and may be facilities, platforms, units,[3] or locations.
- Systems include families of systems (FOSs) and systems of systems (SOSs) and contain software and hardware equipment items.
- System functions are required by operational activities and are performed by one or more systems.
- Performance refers to performance characteristics of systems, system functions, links (i.e., physical links), computer networks, and system data exchanges.
- Standards are associated with technologies, systems, systems nodes, and data, and refer to technical standards for information processing, information transfer, data, security, and human computer interface.
- Technologies include future technologies and relates to systems and emerging standards concerning the use of such technologies.

Conceptually, these are related as shown in **Figure 4.2-1**.

[3] In this context unit refers to “any military element whose structure is prescribed by competent authority, such as a table of organization and equipment; specifically, part of an organization.” (Joint Publication 1-02. DoD *Dictionary of Military and Associated Terms*, 12 April 2001).

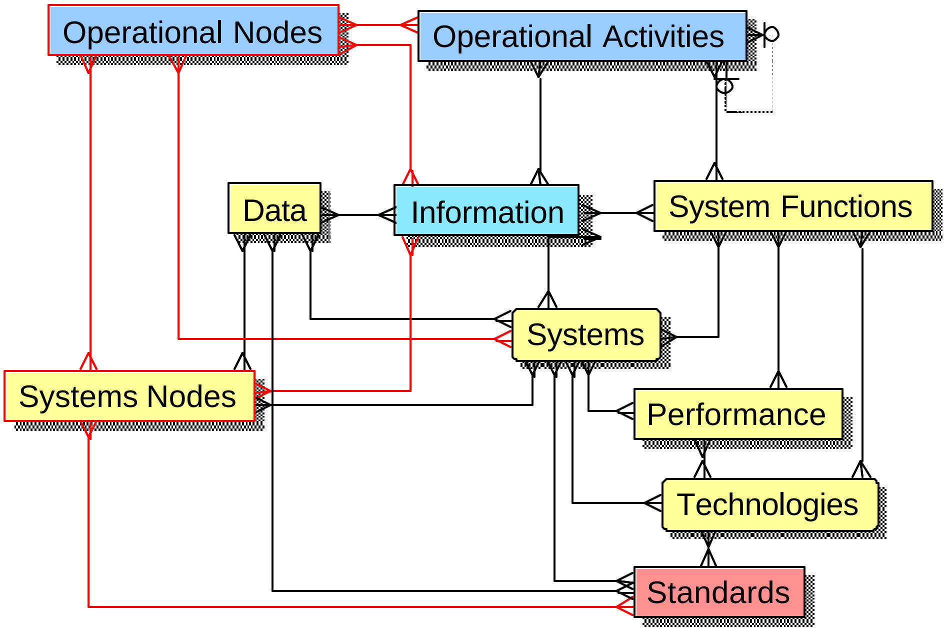


Figure 4.2-1. Architecture Concepts Model

The depicted (conceptual) relationships shown in this diagram include the following (among many others):

- Operational nodes perform many operational activities.
- Operational nodes require information.
- Information are related to systems and implemented as data, which is associated with standards.
- Systems perform system functions.
- Systems have performance characteristics; both systems and performance may relate to a system function being performed.

With these relationships, many types of architectural and related information can be represented such as networks, information flows, information requirements, interfaces, and so forth.

4.2.3 Conceptual Descriptions of Basic Architecture Concepts

4.2.3.1 Operational Nodes

Operational nodes have three major types, as depicted in **Figure 4.2-2**. Operational nodes are independent of materiel considerations; indeed, they exist to fulfill the missions of the enterprise and to perform its tasks and activities (processes, procedures, and operational functions). Use of operational nodes supports analysis and design by separating business process modeling and information requirements from the materiel solutions that support them. Similarly, tasks and activities are organized and communities of interest are defined to suit the mission and process requirements. The materiel is flexibly and automatically configurable to support the operational processes. However, an Operational View (OV) often has materiel constraints and requirements that must be addressed. Where appropriate, systems or physical nodes that constitute the location of an Operational Node may augment the description of an Operational Node. These are often taken as recommendations or boundaries for further Systems View (SV) details.

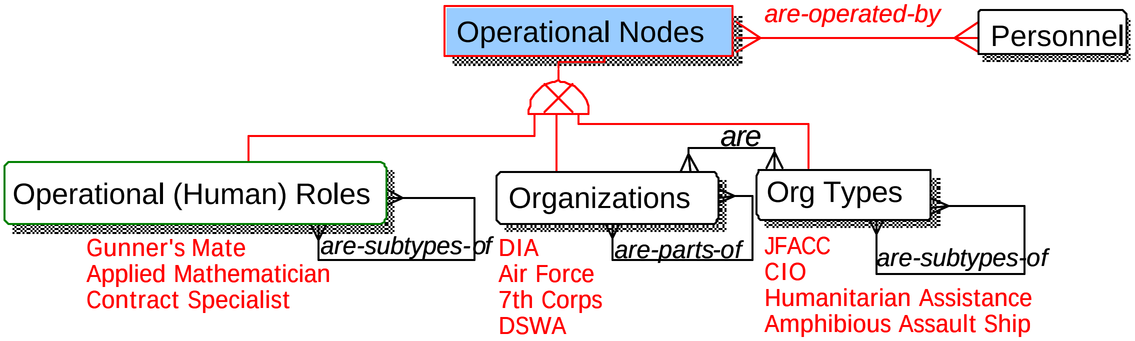


Figure 4.2-2. Operational Nodes Concept Model

Figure 4.2-2 illustrates the relationships between operational nodes and organizations (e.g., Defense Intelligence Agency [DIA], U.S. Air Force [USAF], U.S. Army 7th Corps), organization types (e.g., Joint Force Air Component Commander [JFACC], Chief Information Officer [CIO], Non-Government Humanitarian Assistance Agency), and operational (or human) roles (e.g., gunner's mate, applied mathematician, contract specialist). An operational role may be a skill, occupation, occupational specialty, or position (not shown in this diagram). The relationships indicate that operational roles may be subsets (part of, subtype of) of other operational roles; organizations may be subsets of other organizations, and organization types may be subsets of other organization types. An organization may be of one or more organization types.

4.2.3.2 Operational Activities (and Tasks)

Operational activities have many-to-many relationships with operational nodes, reflecting the fact (requirement) that operational nodes perform many operational activities and that more than one operational node can perform an operational activity. Similarly, there is a traceability relationship between operational activities and formal (often preplanned) tasks such as those defined in the Universal Joint Task List (UJTL).[4] Operational activities are supported by system functions and are performed at operational nodes. The relationships indicate that operational activities may by subsets of (i.e., part of, subtype of, just below in a hierarchy of) other operational activities; and tasks may be subsets (i.e., part of, subtype of, just below in a hierarchy of) other tasks. These relationships are shown in **Figure 4.2-3**.

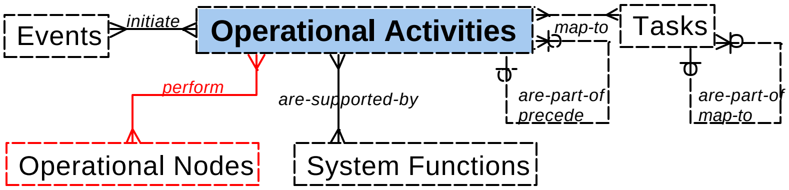


Figure 4.2-3. Operational Activities Concept Model

4.2.3.3 Information, Data, and Data Sources

Information is processed (produced and/or consumed) by operational nodes while carrying out operational activities as defined in the Operational View. They define what is being exchanged in an information exchange. Automated information are those subsets of information exchanges that are implemented as system function data flows and system data exchanges in the SV. System data exchanges are specified via data stored in databases and other data structures (e.g., XML). System data exchanges are implemented using standards, such as data standards and message standards of the Joint Technical Architecture (JTA). System data exchanges are also characterized by the entities, attributes, and relationships of a Data Model. Data can be populated (sourced) from data dictionaries to provide support to various systems nodes. Where possible, standard (authoritative) data sources are used to populate data resources. A conceptual model of these relationships is depicted in **Figure 4.2-4**.

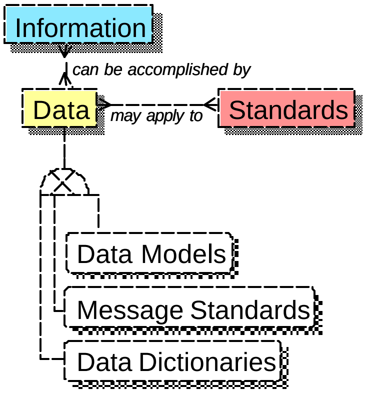


Figure 4.2-4. Information and Data Concept Model

[4] CJCSM 3500.04C, *Universal Joint Task List (UJTL)*, Version 5.0, 1 July 2002.

4.2.3.4 Information, Data, Information and Systems Data Exchanges, and Interfaces

The requirement for, and communication of, information and data is one of the most important architecture constructs. Operational nodes exchange information via activities' input/output (I/O) flows. Two or more systems have an Interface. An Interface represents the content or systems data that is exchanged via the Interface. The description of system function data flows exchanged between two systems is in terms of system data exchanges that have traceability to operational activity I/O flows, and information exchanges (e.g., system data exchanges implement operational information exchanges).

As can be seen in **Figure 4.2-5** at the system level, one describes the interface (system interface) in terms of system data exchanges. Because there is a relationship between system data exchanges and operational information exchanges, the operational information exchange requirements can be correlated to the system data exchanges that implement them.

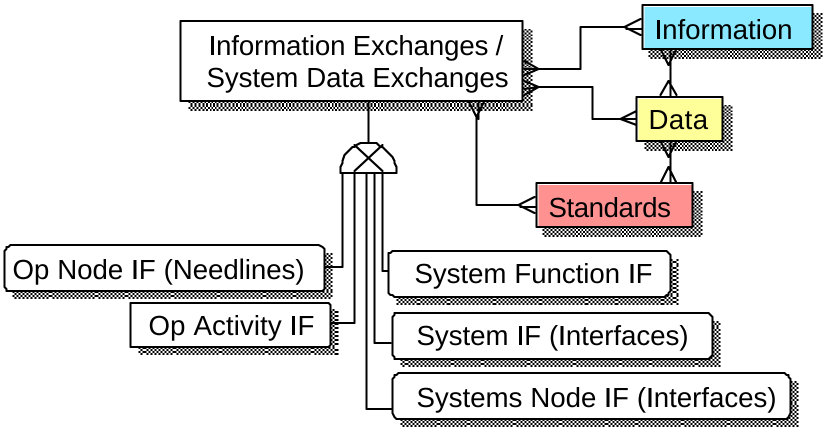


Figure 4.2-5. Interfaces Concept Model

4.2.3.5 Systems or Physical Nodes

Figure 4.2-6 depicts four major conceptual categories of systems nodes: (1) platforms such as ships, aircraft, missile, and vehicles; (2) units such as a military element; (3) (geographical) locations; and (4) facilities such as a post, base, airfield, depot, or fort. All have the property that systems and equipment can be installed in, or assigned to, them. Such equipment establishments are critical to the cost and performance of a candidate architecture.

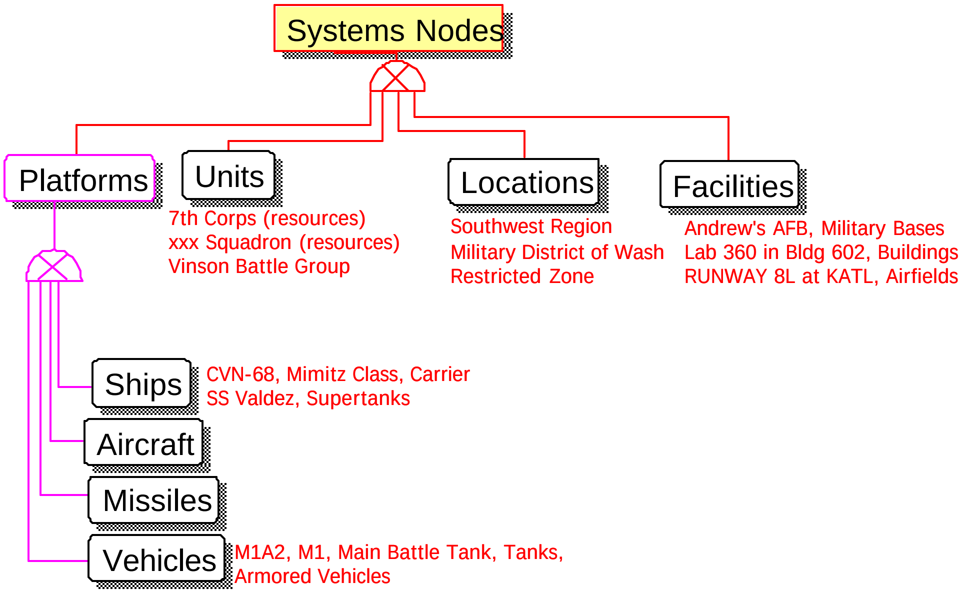


Figure 4.2-6. Systems Nodes Concept Model

Platforms allow for multiple typing of aircraft, so that it is not necessary to specify a single type, a problem for multi-mission aircraft such as the F/A-18. These concepts encourage a distinction between operational node (described, perhaps, by the administrative collection of tasks or process activities with a mission) and units. This distinction allows the OV to address the business process (operational nodes and activities) independently of materiel considerations.

4.2.3.6 Systems and System Items (Software and Hardware Equipment)

The concept "system" has a very general meaning. The term *system* in the Framework is used to denote a family of systems (FoS), a system of systems (SoS), a nomenclatured system, or a subsystem. As shown in **Figure 4.2-7**, a system is composed of hardware (equipment) and software. A systems node can be both the "host" (e.g., platform) in which systems are installed (e.g., the Advanced Synthetic Aperture Radar System [ASARS] in the U-2 platform) and also the system itself (e.g., the U-2 is made up of subsystems, software, and hardware items and may be considered a system). As a systems node, the U-2 is a materiel platform into which systems are installed, while, as a system, it is an arrangement of system items that are used by an operational node. Since architects must be able to think in either or both terms, both points of view are modeled but with separate defined meanings and separate defined inter-relationships.

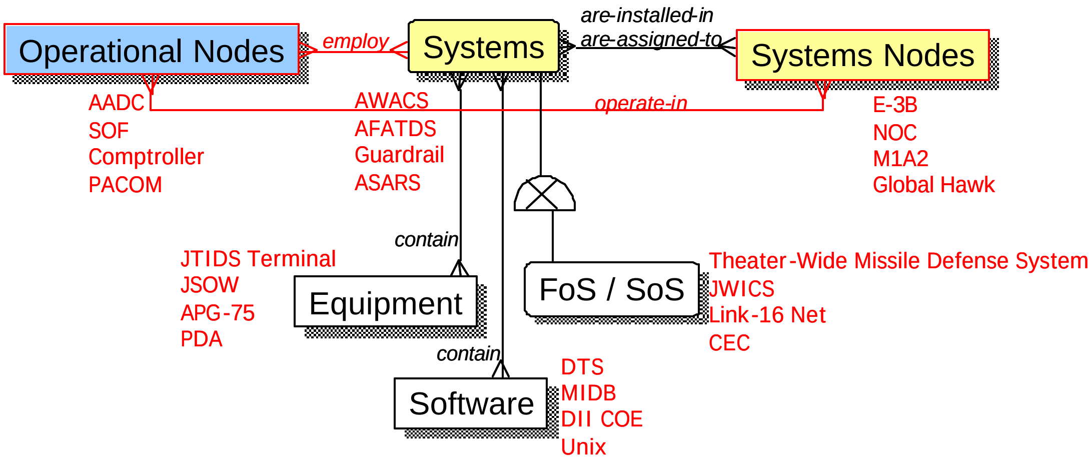


Figure 4.2-7. Systems Concept Model

4.2.3.7 Networks and Physical Links

Computer networks are accomplished by collections of systems, physical links, and hardware equipment (routers, switches, cable, receivers/transmitters, antennae) arranged with standards and software/firmware so as to accomplish a communications function (of a system), as shown in **Figure 4.2-8**. Communications systems form an important subclass of systems (not shown) for characterizing requirements for, and in support of, computer networks.

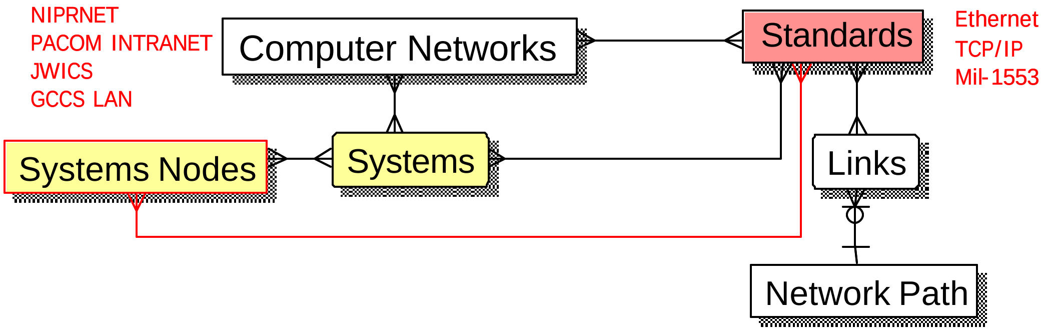


Figure 4.2-8. Networks Concept Model

Two or more systems have an interface. Physical links describe the physical means by which the system interface is achieved. Attributes of physical links include type (such as communications media) and communications protocols. Since systems contains hardware equipment, this basic structure can be used to capture the most detailed communications system. As shown in Figure 4.2-8, an interface between two systems can have multiple physical means or physical links; the links can have multiple communications protocols; and a collection of links makes up a network path. A collection of network paths and communications systems makes up a network.

4.2.3.8 Performance Characteristics

Architectural analysis and development considers the performance of systems, system functions, physical (communications) links, system data exchanges, and computer networks as shown in **Figure 4.2-9**. Performance may also be subject to conditions.

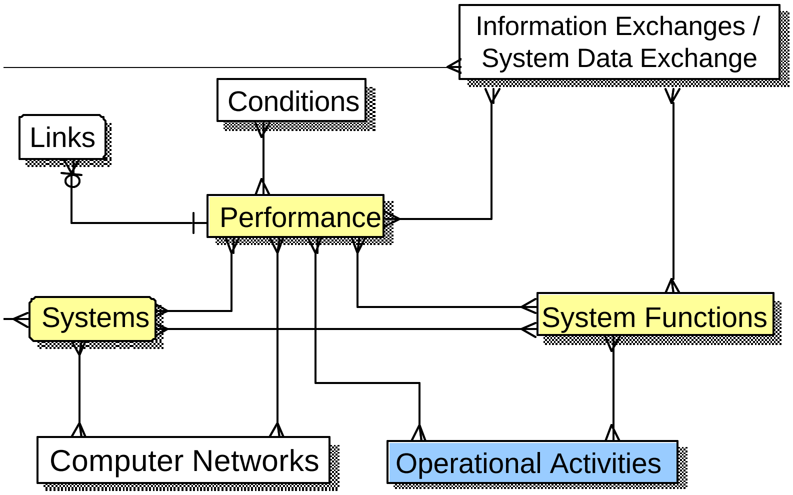


Figure 4.2-9. Performance Concepts Example

4.2.3.9 Technical Standards for Information Processing, Information Transfer, Data/Information, Security, and Human Computer Interface

Information technology (IT) standards are related to architectural concepts as shown in **Figure 4.2-10**. Information technology standards include the following groups:

- Message (information exchange) standards
- Data standards, to include standard data elements, standard prime words, and standard generic elements (many from the DoD Data Dictionary System, populated through data standardization under DoD 8320)[5]
- Other standards such as information processing and security standards, as detailed in the Joint Technical Architecture (JTA)
- Standards for design including human computer interface (HCI)
- Reference models such as the DoD Technical Reference Model (TRM)

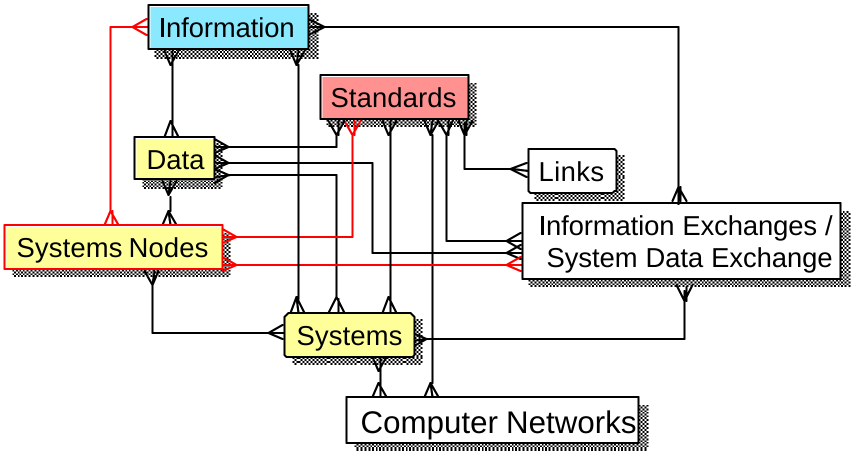


Figure 4.2-10. Standards Concepts Example

[5] Almost all of the All-DoD CADM has been submitted for data standardization; about 95 percent of the CADM entities and attributes (and relationships for DoD Data Model) are approved as DoD data standards.

4.2.3.10 Allocations and Assignments

Allocations and assignments express the relationships between architectural data elements such as:

- Operational activities are performed by operational nodes
- Operational activities are supported by system functions
- System functions are performed by systems
- Systems are installed in or assigned to facilities, platforms, locations, and units
- Facilities, platforms, locations, and units are employed by operational nodes
- Systems are employed by operational nodes
- Technical standards are implemented by systems
- Technical standards are implemented by facilities, platforms, and units
- Performance attributes are associated with systems, and system functions are performed by systems
- Communications protocols (i.e., standards) are applicable to links or to a network of links

In the CADM, allocations and assignments are implemented as associative entities between entities (two or three). There are often amplifying data to express performance values, caveat support, set time periods, and others.

4.2.3.11 Missions, Mission Areas, Mission Capabilities, and Functional Areas

Architecture can pertain to one or more missions, mission areas, mission capabilities, functional (operational) areas, etc. An architecture is described by a collection of architecture products (one such integrated set of architecture products is described in Volume II and comprises the DoD set of standard architecture products). Missions, mission areas, mission capabilities, functional areas, etc., can be addressed by architecture products, as shown in **Figure 4.2-11**. This figure shows that there is a many-to-many relationship between architectures and architecture products that compose basic architectural data elements and are subject to reuse. Examples of basic architectural data elements include operational nodes, operational activities, information and data exchanges, system functions, systems nodes, performance, technologies, and standards. Reusing architecture products composed of standard architecture data elements saves redundant data element definition, increases the integrity of those data elements within and across architectures, reduces independent product reconciliation, and supports cross-architecture interoperability, performance, and capability assessment. As noted above, the intent is to populate these products with standard (authoritative) data sources, wherever they can be found.

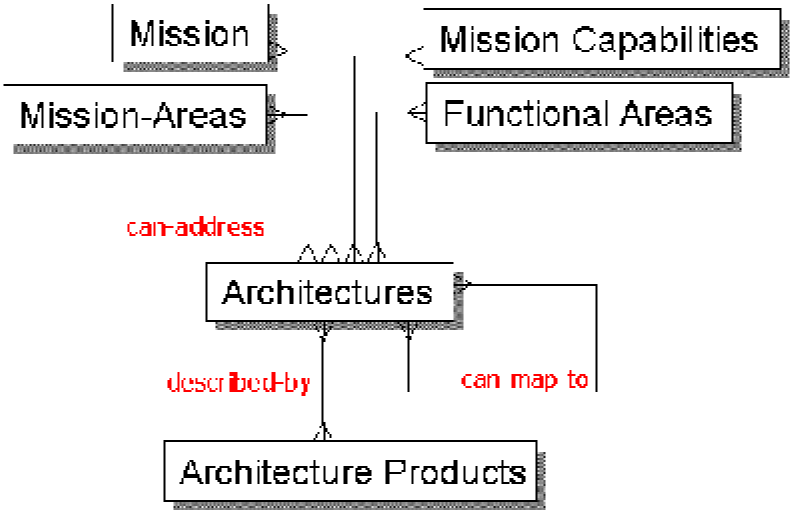


Figure 4.2-11. Architectures Concept

4.2.3.12 Requirements

Requirements, including conditions and scenario, as related to architectures, are of many types, as shown in **Figure 4.2-12**. Examples of requirements are guidance directed by an authority. Other classes of guidance commonly referenced in architectures are goals, visions, doctrine, directives, policy, strategy, mission statement, operational rule, and operational condition.

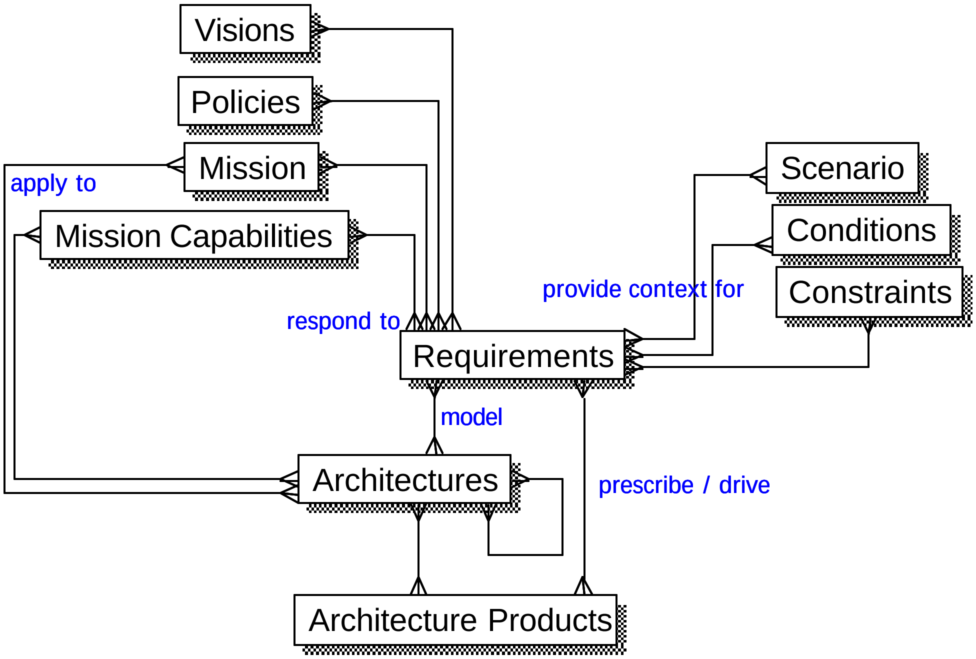


Figure 4.2-12. Architecture and Requirements Concept Examples

Four types of IT requirements are currently supported in the CADM (with appropriate relationships among them): (1) information requirement (the information element being exchanged); (2) exchange needline requirement (nodes with whom the exchange occurs); (3) information exchange requirement (what conditions are imposed on the size, speed, and other aspects of the exchange); (4) and process activity exchange requirement (which operational activities are supported).

4.2.3.13 Schedules for Requirements, Resources, and Acquisition

Architectural analysis and development addresses the phasing, development, installation, and other time period concepts, as shown in **Figure 4.2-13**. Relating basic architectural elements to time periods enables a consistent specification of what is required and what is provided for each time period applicable to an architecture. Time period also allows architectures to use the same time frames for ease of cross-reference, comparison, and analysis. The time period construct, in conjunction with other related architecture constructs, can also be used to document a transition plan for moving from an existing architecture configuration to a future one.

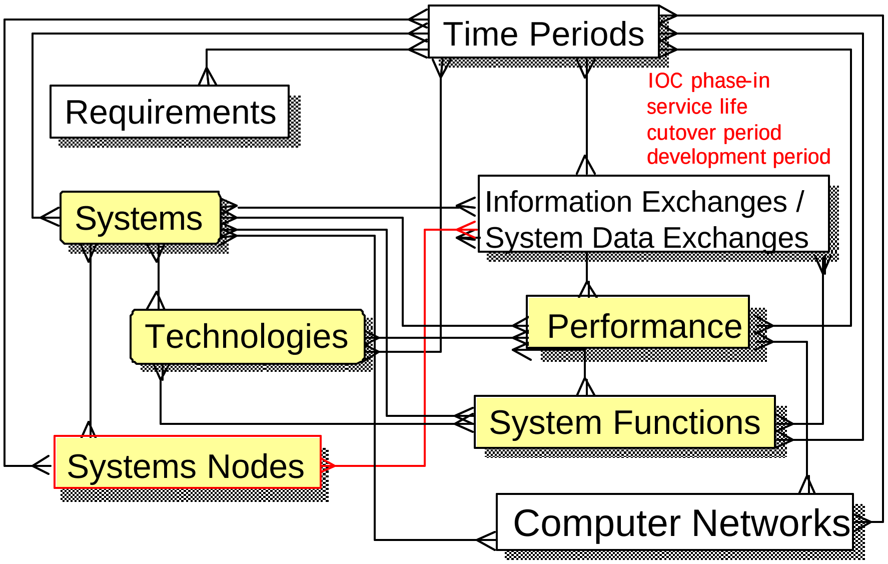


Figure 4.2-13. Schedules Concept Examples

4.2.3.14 Technologies for Systems and Information Technology Standards

Architectural development addresses the needs and plans for future technologies, as shown in **Figure 4.2-14**. The architecture reveals and justifies the need for research and development and lays out a high-level plan for its transition for each applicable time period.

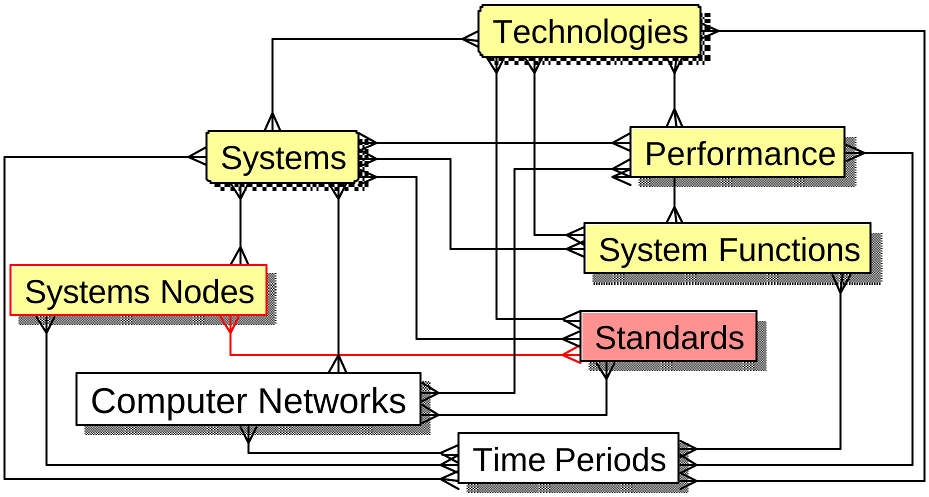


Figure 4.2-14. Technologies Concept Example

4.2.3.15 Costs and Programmatics

Architectures provide a systematic means of analyzing cost, programs (and associated funds), risks, and requirements, as suggested by **Figure 4.2-15**. The rigorous modeling of these relationships in CADM is future, but the concepts are shown here because of this important role for architectures and the imminent realization and codification of architecture analysis processes as defined, for example, in the Mission Capabilities Package section of this Deskbook.

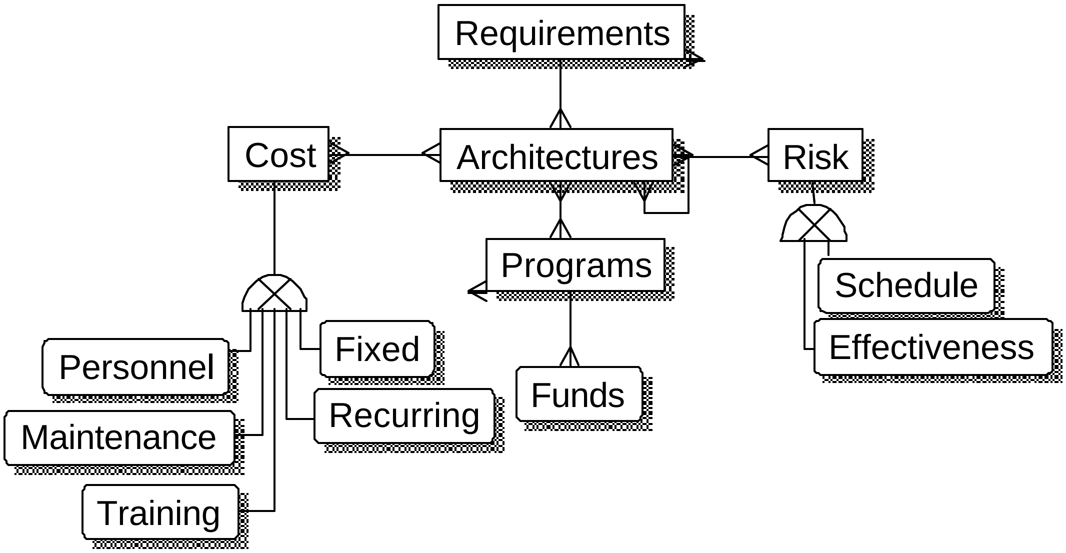


Figure 4.2-15. Costs and Programmatics Concept Model

4.2.4 Using CADM

Some key features of the CADM are:

- Use of existing DoD data standards where possible. Sometimes this did not result in the most optimized solution for architectures, but the compromise was worthwhile to improve interoperability with other databases. For example, CADM overlaps the Command and Control (C2) Core Data Model and other data models used to specify DoD data standards. CADM is identical with those data models where they overlap. Consistency among such models greatly facilitates data sharing.
- Use of subtypes. As in many of the Defense Data Architecture (DDA) models, CADM employs high-level entities that then have subtypes. Sometimes the subtyping has many levels (e.g., Materiel-Item has subtypes for Ship and Aircraft). More specific subtypes can be added later, as architects need those specific properties in their architectures.
- Use of associative entities that allow architecture data elements to participate in multiple products and multiple architectures.
- Use of independent tables (e.g., ORGANIZATION-TYPE) with double-associative entities (e.g., ORGANIZATION-TYPE-ASSOCIATION) that represent the enterprise taxonomies and hierarchies.
- Properly generalizing the data elements for architecture products described in Volume II, so that data underlying different products is consistent. Because many of the same elements are re-used across products, this supports the data administration and management goal of develop-once, use-many, and inter-product consistency.

4.2.4.1 Product Subviews

The CADM comes with built-in subviews for each architecture product as well as some other subviews. When a product subview is selected, the model shows only the entities/tables that comprise that product. Using this, it is possible to see all the data that can be input or accessed from a CADM repository for any particular product. This is the detailed specification of the products provided in Volume II. It is possible to color code entities/tables or to create local subviews for a product to show the data elements that will be developed or used in a particular architecture project. This can be helpful in setting up a data development, collection, interface, and authoritative source plan for a project.

It is also possible to use a data modeling tool's report option to generate a report on the entities, tables, attributes, fields, definitions, domain values, data types, etc., for any pre-defined or user-defined subview. These can be useful as implementation notes for modeling or assessment tool developers as well as for architecture developers to understand the full range of data associated with a particular product.

4.2.4.2 Domain Values

Often an attribute/field will have standard specified values that it can take on (e.g., Country Code). These are in the CADM logical model in the "Notes" tab. They also can be output in a data modeling tool's report generator.

4.2.4.3 Example Values and Implementation Notes

The CADM documentation provides sample values for many tables as an implementation guide for repository, modeling, simulation, and data developers. Often, there appear to be multiple locations in which the same type of data could be stored in a CADM-based database. The implementation notes provide the preferred storage entity that will make subsequent data crossing and sharing less costly and more effective.

4.2.4.4 Change Control Process and Data Standardization

CADM constantly evolves as new architecture data requirements (e.g., after the finalization of a version of the Framework) are identified that are common to multiple architecture domains (thus part of a core). CADM data elements are registered as standard data elements under the DoD Data Administration policy. When users need extensions or discover problems or deficiencies with CADM, there is a configuration management process that begins with the Office of the Assistant Secretary of Defense for Networks and Information Integration (OASD[NII]). If the affected data elements fall within the authority of other functional data administrators, coordination with those organizations is necessary. Since many implemented databases and tools could be affected by a change, changes should have strong rationale.

4.2.4.5 Extensible Markup Language (XML) Tags for Architecture Data

The XML for CADM was generated from the CADM and has been registered with the DISA XML repository. The definitions and data types are the same as those in the CADM.

4.2.4.6 Conformance

CADM may be implemented in multiple target environments, e.g., implementations exist in MS Access, SQL Server 2000, and Oracle DBMS. A strong concept of conformance is needed to ensure fully faithful information transfer among databases, which cannot happen if the primary keys of one database have no correlation to the primary keys of another database for the same entity. CADM conformance means the following:

- Conforming model is to be based on a subset of the CADM (not all the entities nor all attributes of selected entities are required).
- Extensions of that subset are expected (but should not be redundant with elements of the CADM itself); extensions that could apply to the CADM for general use should be proposed.
- Agreed data types and coded domains should be used.
- Points of contact should be identified and consulted when generating instances of keys (to avoid redundancy and non-uniqueness).

- Primary key attributes for entities taken from the CADM should be identical with or directly derivable from the primary key attributes specified in CADM (alternate keys may be used but CADM keys need to be preserved).
- Keys for authoritative data source instances should be retained to enable effective updates from those sources. The goal of CADM conformance is to ensure fully faithful information transfer among databases, which cannot happen if the primary keys of one database have no correlation to the primary keys of another database for the same entity.

The following suggest the use and value of conformance to the CADM:

- Conformance can be determined by inspection and analysis of tools or repositories that are proposed as CADM compliant. This inspection should be of the logical and physical data models, the actual populated database, and the interaction of the tools with the database.
- Conformance to CADM enables comparison between architecture data repositories and sharing of architecture data across architecture repositories and databases. Non-conforming repositories require translation and data correlation and reconciliation.
- Translation losses and infeasible reconciliations can occur. Translation, data correlation, and reconciliation costs and impacts are typically underestimated. For these reasons, development of architecture data in non-conforming data repositories, databases, or tools should be carefully considered and avoided, whenever possible.

4.2.5 Relating Conceptual Basic Architectural Elements to Entities of the CADM

Table 4.2-1 associates the basic architectural elements noted above and the actual structures of the CADM at the entity level.

Table 4.2-1. Relation of CADM Entities to Basic Architectural Elements

Architectural Element	Component	CADM Specification (Entity Names)
Operational Nodes	[Nodes]	NODE, NODE-ASSOCIATION, NODE-ASSOCIATION-DOCUMENT, NODE-DETAIL, NODE-HIERARCHY, NODE-MISSION-AREA, NODE-ICON, NODE-TREE, NODE-TREE-NODE-HIERARCHY
	Organizations	ORGANIZATION, NODE-ORGANIZATION, ORGANIZATION-ASSOCIATION, ORGANIZATION-ASSOCIATION, ORGANIZATION-CAPABILITY-ESTIMATE, ORGANIZATION-DOCUMENT, ORGANIZATION-GUIDANCE, ORGANIZATION-MISSION-AREA, ORGANIZATION-NAME, ORGANIZATION-PROCESS-ACTIVITY
	Organization Types	ORGANIZATION-TYPE, NODE-ORGANIZATION-TYPE, ORGANIZATION-TYPE-ASSOCIATION, ORGANIZATION-TYPE-CAPABILITY-NORM, ORGANIZATION-TYPE-DOCUMENT, ORGANIZATION-TYPE-ORGANIZATION, ORGANIZATION-TYPE-PROCESS-ACTIVITY; OPERATIONAL-FACILITY, OPERATIONAL-ELEMENT
	Operational Role	OPERATIONAL-ROLE, NODE-ORGANIZATIONAL-ROLE; SKILL, PERSON-TYPE, OCCUPATION, OCCUPATIONAL-SPECIALTY, POSITION
Operational Activities (and Tasks)	Operational Activity	PROCESS-ACTIVITY, PROCESS-ACTIVITY-ASSOCIATION, ACTIVITY-MODEL, ACTIVITY-MODEL-PROCESS-ACTIVITY, NODE-PROCESS-ACTIVITY, ACTIVITY-MODEL-THREAD, NODE-ACTIVITY-MODEL-THREAD, OPERATIONAL-MISSION-THREAD,
	Tasks	TASK, TASK-ASSOCIATION, TASK-MEASURE, TASK-MISSION-AREA, PROCESS-ACTIVITY-TASK, OPERATIONAL-CAPABILITY-TASK, MISSION-ESSENTIAL-TASK, MISSION-ESSENTIAL-TASK-LIST, NODE-TASK, MISSION-ESSENTIAL-TASK-STANDARD
Information, Data, and Data Sources	Information	INFORMATION-ELEMENT, INFORMATION-ELEMENT-ASSOCIATION, INFORMATION-ELEMENT-ACTIVITY-MODEL-ROLE, NODE-ACTIVITY-MODEL-INFORMATION-ELEMENT-ROLE
	Data	DATA-STANDARD, STANDARD-DATA-ELEMENT, STANDARD-PRIME-WORD, DATA-DICTIONARY, DATA-DOMAIN, INFORMATION-ELEMENT-DATA-DICTIONARY-ELEMENT
		DATA-ATTRIBUTE, DATA-ENTITY, DATA-ENTITY-RELATIONSHIP
Information, Data, Information and System Data Exchanges, and Interfaces	[Guidance]	GUIDANCE, GUIDANCE-ASSOCIATION, GUIDANCE-DOCUMENT, DIRECTED-CONSTRAINT
	Info Req	INFORMATION-ELEMENT, INFORMATION-REQUIREMENT, INFORMATION-ELEMENT-ASSOCIATION
	Info Flows	EXCHANGE-NEED-LINE-REQUIREMENT, EXCHANGE-RELATIONSHIP-TYPE, INFORMATION-EXCHANGE-REQUIREMENT, INFORMATION-EXCHANGE-REQUIREMENT-ASSURANCE, INFORMATION-EXCHANGE-REQUIREMENT-ELEMENT, INFORMATION-EXCHANGE-REQUIREMENT-ELEMENT-DEPLOYMENT-MISSION-TYPE, INFORMATION-EXCHANGE-REQUIREMENT-ELEMENT-DEPLOYMENT-PHASE,

Architectural Element	Component	CADM Specification (Entity Names)
		INFORMATION-EXCHANGE-REQUIREMENT-ELEMENT-METHOD, INFORMATION-EXCHANGE-REQUIREMENT-ELEMENT-PRODUCT, INFORMATION-EXCHANGE-REQUIREMENT-FAILURE-IMPACT-DETAIL, INFORMATION-EXCHANGE-REQUIREMENT-TRIGGER, INFORMATION-EXCHANGE-REQUIREMENT-TRIGGER-OPERATIONAL-RULE
	Interfaces	TECHNICAL-INTERFACE, TECHNICAL-INTERFACE-INFORMATION-TECHNOLOGY-REQUIREMENT, TECHNICAL-INTERFACE-ORGANIZATION, TECHNICAL-INTERFACE-STANDARD-TRANSACTION, TECHNICAL-INTERFACE-TYPE, SYSTEM-INTERFACE-DESCRIPTION {SV-1}, SYSTEM-INTERFACE-DESCRIPTION-ELEMENT
Systems (Physical) Nodes	[Nodes]	NODE, NODE-ASSOCIATION, NODE-ASSOCIATION-INFORMATION-TECHNOLOGY-REQUIREMENT, NODE-ASSOCIATION-NETWORK, NODE-ASSOCIATION-SYSTEM-ASSOCIATION, NODE-COMMUNICATION-MEDIUM, NODE-INFORMATION-ASSET, NODE-PORT, ICON-CATALOG, ICON-CATALOG-ASSOCIATION, ICON-CATALOG-INSPECTOR
	Systems Node	SYSTEM, NODE-SYSTEM, NODE-SYSTEM-ASSOCIATION, NODE-SYSTEM-ASSET-OWNERSHIP, NODE-SYSTEM-COST-MANAGEMENT, NODE-SYSTEM-SOFTWARE-ITEM, NODE-SYSTEM-TRANSMISSION
	Platforms	MATERIEL, NODE-MATERIEL; SATELLITE; MILITARY-PLATFORM; NODE-MILITARY-PLATFORM, MILITARY-PLATFORM-ASSOCIATION, MILITARY-PLATFORM-COMMUNICATION-SYSTEM, MILITARY-PLATFORM-SENSOR-SYSTEM, MILITARY-PLATFORM-WEAPON-SYSTEM, SHIP, SHIP-TYPE
	Units	ORGANIZATION, NODE-ORGANIZATION; ORGANIZATION-TYPE, NODE-ORGANIZATION-TYPE, OPERATIONAL-FACILITY, OPERATIONAL-ELEMENT, OPERATIONAL-NETWORK-NODE
	Locations	NODE, NODE-DETAIL, LOCATION, POINT, FACILITY-POINT, ORGANIZATION-POINT, MATERIEL-LOCATOR, LINE, GEOMETRIC-SURFACE, GEOMETRIC-VOLUME, CONE-VOLUME, MEASURED-ELEVATION-POINT
	Facilities	FACILITY, NODE-FACILITY, FACILITY-TYPE, FACILITY-ASSOCIATION, FACILITY-BASIC-CATEGORY, FACILITY-CATEGORY, FACILITY-CLASS, FACILITY-USE, AREA-FACILITY, STRUCTURE-FACILITY, UTILITY-SYSTEM (FACILITY), BUILDING, FACILITY-PARTITION, ROOM, ROOM-TYPE, ROOM-ASSOCIATION, FACILITY-INFRASTRUCTURE-IMPROVEMENT, FACILITY-IMPROVEMENT-ACTIVITY, TELECOMMUNICATION-DISTRIBUTION-FACILITY, FACILITY-TELECOMMUNICATION-REQUIREMENT, FACILITY-TELECOMMUNICATIONS-INFRASTRUCTURE-IMPROVEMENT

Architectural Element	Component	CADM Specification (Entity Names)
Systems and System Items (Software, and Hardware Equipment)	Materiel	MATERIEL, MATERIEL-ITEM; MATERIEL-ASSOCIATION, MATERIEL-CUSTODY, MATERIEL-FIELDING, MATERIEL-HOLDING-MATERIEL-ITEM; MATERIEL-ITEM-ASSOCIATION, MATERIEL-ITEM-CAPABILITY-NORM, MATERIEL-ITEM-COST, MATERIEL-ITEM-DOCUMENT, CIRCUIT-SWITCH-MATERIEL
	Systems	SYSTEM, SYSTEM-TYPE, SYSTEM-DETAIL, SYSTEM-ASSOCIATION, SYSTEM-TYPE-ASSOCIATION, SYSTEM-TRANSMISSION; SYSTEM-DIRECTED CONSTRAINT, SYSTEM-ASSOCIATION-DIRECTED-CONSTRAINT; SYSTEM-ELEMENT, PLATFORM-ELEMENT, PLATFORM-APPLICATION-SOFTWARE-ELEMENT, AUTOMATED-INFORMATION-SYSTEM
	Equipment	MATERIEL-ITEM, EQUIPMENT-TYPE, RADIO-TYPE, WEAPON-SYSTEM, SENSOR-SYSTEM, SYSTEM-EQUIPMENT-TYPE, ANTENNA-TYPE
	Software	SOFTWARE-ITEM, SOFTWARE-ITEM-ASSOCIATION, SYSTEM-SOFTWARE-ITEM, SOFTWARE-ITEM-USE, NODE-SYSTEM-SOFTWARE-ITEM, CONVENTIONAL-SOFTWARE-ITEM
	System Function	SYSTEM-FUNCTION, SYSTEM-FUNCTIONALITY-DESCRIPTION {SV-4}, SYSTEM-FUNCTION-TRACEABILITY-MATRIX {SV-5}, SYSTEM-FUNCTION-TRACEABILITY-MATRIX-ELEMENT
Networks and Physical Links	Networks	NETWORK, NETWORK-SYSTEM, COMMUNICATION-SYSTEM, NETWORK-NODE, NETWORK-ORGANIZATION, NETWORK-PATH, NETWORK-PATH-LINK, NETWORK-TYPE NETWORK-ASSOCIATION, NETWORK-CAPABILITY, NETWORK-COMMUNICATION-MEDIUM, NETWORK-CONTROLLER-TYPE, NETWORK-DEMARCATION-POINT, NETWORK-DETAIL, NETWORK-DEVICE-MATERIEL, NETWORK-DEVICE-MATERIEL-INFORMATION-TECHNOLOGY-STANDARD, NETWORK-DEVICE-MATERIEL-INTERNET-ADDRESS, NETWORK-DEVICE-MATERIEL-WORKSTATION, NETWORK-DEVICE-MATERIEL-WORKSTATION-SOFTWARE-ITEM, NETWORK-ECHELON, NETWORK-INFORMATION-TECHNOLOGY-STANDARD, NETWORK-INTERNET-ADDRESSING
	Channels	COMMUNICATION-CHANNEL
	Circuits	COMMUNICATION-CIRCUIT, COMMUNICATION-CIRCUIT-INFORMATION-TECHNOLOGY-REQUIREMENT, COMMUNICATION-CIRCUIT-THREAD-ELEMENT, COMMUNICATION-CIRCUIT-TYPE
	Links	COMMUNICATION-LINK, COMMUNICATION-LINK-INFORMATION-TECHNOLOGY-REQUIREMENT, COMMUNICATION-LINK-TYPE, NODE-CONNECTIVITY-DESCRIPTION {OV-2}, NODE-CONNECTIVITY-DESCRIPTION-ELEMENT, NODE-LINK, NODE-LINK-ASSOCIATION, NODE-LINK-CAPABILITY, NODE-LINK-COMMUNICATION-MEDIUM, NODE-LINK-COMMUNICATION-ROUTE-SEGMENT, NODE-LINK-INFORMATION-TECHNOLOGY-STANDARD

Architectural Element	Component	CADM Specification (Entity Names)
	Means	COMMUNICATION-MEANS, COMMUNICATION-MEANS-ROUTE-SEGMENT, COMMUNICATION-MEDIUM
Allocations and Assignments	Allocations and Assignments	EQUIPMENT-TYPE-SOFTWARE-ITEM FACILITY-HOLDING-MATERIEL-ITEM INFORMATION-ASSET, INFORMATION-ASSET-AGREEMENT, INFORMATION-ASSET-DOCUMENT, INFORMATION-ASSET-GUIDANCE, INFORMATION-ASSET-INFORMATION-ELEMENT, INFORMATION-ASSET-RELATION MATERIEL-ORGANIZATION MATERIEL-ITEM-ESTABLISHMENT, MATERIEL-ITEM-ESTABLISHMENT-MATERIEL-ITEM-DETAIL, MATERIEL-MATERIEL-ITEM-ESTABLISHMENT-MATERIEL-ITEM-DETAIL NETWORK-ORGANIZATION-TYPE-ESTABLISHMENT-MATERIEL-ITEM-DETAIL ORGANIZATION-FACILITY, ORGANIZATION-GUIDANCE, ORGANIZATION-HOLDING-MATERIEL-ITEM, ORGANIZATION-HOLDING-ORGANIZATION-TYPE, ORGANIZATION-ORGANIZATION-TYPE-ESTABLISHMENT, ORGANIZATION-AGREEMENT ORGANIZATION-TYPE-ESTABLISHMENT, ORGANIZATION-TYPE-ESTABLISHMENT-CROSS-REFERENCE-ASSOCIATION, ORGANIZATION-TYPE-ESTABLISHMENT-FORCE-STRUCTURE, ORGANIZATION-TYPE-ESTABLISHMENT-MATERIEL-ITEM-DETAIL, ORGANIZATION-TYPE-ESTABLISHMENT-ORGANIZATION-TYPE-DETAIL, ORGANIZATION-TYPE-ESTABLISHMENT-ORGANIZATION-TYPE-DETAIL-ELEMENT, ORGANIZATION-TYPE-ESTABLISHMENT-PERSON-TYPE-DETAIL, ORGANIZATION-TYPE-ESTABLISHMENT-POSITION-DETAIL, ORGANIZATION-TYPE-ESTABLISHMENT-SYSTEM-DETAIL, ORGANIZATION-TYPE-ASSIGNED-MATERIEL-ITEM-DETAIL ORGANIZATION-TYPE-MISSION-AREA PLAN, PLAN-ASSOCIATION, PLAN-DOCUMENT, PLAN-GUIDANCE, PLANNED-ACTION, PLAN-ORGANIZATION SATELLITE-ANTENNA-TYPE, SATELLITE-ASSOCIATION SYSTEM-SATELLITE, SYSTEM-PROCESS-ACTIVITY, SYSTEM-PROCESS-ACTIVITY-STANDARD, SYSTEM-SYSTEM-TYPE, SYSTEM-ORGANIZATION-TYPE-ESTABLISHMENT-MATERIEL-ITEM-DETAIL, SYSTEM-MISSION-AREA, SYSTEM-OPERATIONAL-CAPABILITY-TASK, SYSTEM-ORGANIZATION, SYSTEM-ORGANIZATION-TYPE, SYSTEM-INFORMATION-ASSET, SYSTEM-DOCUMENT, SYSTEM-CAVEATED-SECURITY-CLASSIFICATION, SYSTEM-CAPABILITY

Architectural Element	Component	CADM Specification (Entity Names)
		TASK-MATERIEL-ITEM,
Performance Characteristics	Perf Char	CAPABILITY, CAPABILITY-ASSOCIATION TECHNICAL-CRITERIA-DOCUMENT, TECHNICAL-CRITERION, TECHNICAL-CRITERION-PROFILE, TECHNICAL-CRITERION-PROFILE-AGREEMENT SYSTEM-PERFORMANCE-PARAMETER-MATRIX {SV-7}, SYSTEM-PERFORMANCE-PARAMETER-MATRIX-ELEMENT SYSTEM-CRITERIA-PROFILE SYSTEM-CAPABILITY, SYSTEM-ASSOCIATION-MEANS, SYSTEM-ASSOCIATION-MIGRATION,
Technical Standards for Information Processing, Information Transfer, Data/Information, Security, and Human Computer Interface	IT Std	INFORMATION-TECHNOLOGY-STANDARD, INFORMATION-TECHNOLOGY-STANDARD-CATEGORY, INFORMATION-TECHNOLOGY-STANDARD-COST-MANAGEMEN T, INFORMATION-TECHNOLOGY-STANDARD-OPTION, INFORMATION-TECHNOLOGY-STANDARD-PARAMETER, INFORMATION-TECHNOLOGY-STANDARD-PROFILE, INFORMATION-TECHNOLOGY-STANDARD-TECHNICAL-SERVICE, INFORMATION-TECHNOLOGY-STANDARD-TECHNICAL-SERVICE-AREA
	Transfer Std	MESSAGE-STANDARD, MESSAGE-STANDARD-INFORMATION-ELEMENT, MESSAGE-STANDARD-POINT-OF-CONTACT APPLICATION-PROGRAM-INTERFACE-STANDARD
Missions, Mission Areas, Mission Capabilities, and Functional Areas	Mission	MISSION, MISSION-ASSOCIATION, MISSION-GUIDANCE, MISSION-ORGANIZATION, MISSION-TASK, MISSION-TASK-OPERATIONAL-CONDITION
	Mission Area	MISSION-AREA, MISSION-AREA -MISSION, MISSION-AREA -PROCESS-ACTIVITY
	Functional Area	FUNCTIONAL-AREA, MISSION-AREA -FUNCTIONAL-AREA, MISSION-FUNCTIONAL-AREA
	Mission Capabilities	MISSION-MILITARY-PLATFORM, MISSION-MILITARY-PLATFORM -CAPABILITY, MISSION-MILITARY-PLATFORM-SENSOR-SYSTEM, MISSION-MILITARY-PLATFORM-SENSOR-SYSTEM-CAPABILITY,
Requirements	IT Req	INFORMATION-TECHNOLOGY-REQUIREMENT, INFORMATION-TECHNOLOGY-REQUIREMENT-COMMUNICATION-MEDIUM, INFORMATION-TECHNOLOGY-REQUIREMENT-INFORMATION-ASSET, INFORMATION-TECHNOLOGY-REQUIREMENT-MATERIEL-ITEM, INFORMATION-TECHNOLOGY-REQUIREMENT-MISSION-AREA, INFORMATION-TECHNOLOGY-REQUIREMENT-NETWORK-NODE, INFORMATION-TECHNOLOGY-REQUIREMENT-SYSTEM, INFORMATION-TECHNOLOGY-REQUIREMENT-TASK

Architectural Element	**Component**	**CADM Specification (Entity Names)**
	Conditions	OPERATIONAL-CONDITION, OPERATIONAL-CONDITION-ASSOCIATION, OPERATIONAL-CONDITION-DESCRIPTOR, OPERATIONAL-DEPLOYMENT-MISSION-TYPE, OPERATIONAL-DEPLOYMENT-PHASE,
	Scenario	OPERATIONAL-SCENARIO, OPERATIONAL-SCENARIO-MISSION, OPERATIONAL-SCENARIO-OPERATIONAL-CONDITION
Schedules for Requirements, Resources, and Acquisition	Schedules	SYSTEM-PROPONENT, SYSTEM-PROCUREMENT-STATUS SYSTEM-STATUS, SYSTEM-STATUS-DEPENDENCY, SYSTEM-STATUS-TYPE, SYSTEM-STATUS-TYPE-SYSTEM, SYSTEM-SYSTEM-ARCHITECTURE SYSTEM-EVOLUTION-DESCRIPTION {SV-8}, SYSTEM-MIGRATION-EVOLUTION, SYSTEM-IMPLEMENTATION-TIME-FRAME
Technologies for Systems and Information Technology Standards	Technologies	TECHNOLOGY, TECHNOLOGY-ASSOCIATION, TECHNOLOGY-COUNTERMEASURE, TECHNOLOGY-FORECAST, TECHNOLOGY-ISSUE TECHNICAL-SERVICE, TECHNICAL-SERVICE-AREA SYSTEM-TECHNICAL-INTERFACE-TYPE, SYSTEM-TECHNOLOGY-FORECAST {SV-9}, SYSTEM-TECHNOLOGY-FORECAST-PROFILE TECHNICAL-STANDARD-PROFILE {TV-1}, TECHNICAL-STANDARD-FORECAST {TV-2} TECHNICAL-GUIDELINE, TECHNICAL-GUIDELINE-ELEMENT, TECHNICAL-GUIDELINE-ELEMENT-INFORMATION-TECHNOLOGY-STANDARD TECHNICAL-ARCHITECTURE, TECHNICAL-ARCHITECTURE-PROFILE-ELEMENT TECHNICAL-ARCHITECTURE-STANDARD REQUIRED-INFORMATION-TECHNOLOGY-CAPABILITY, REQUIRED-REFERENCE-MODEL-SERVICE
Costs and Programmatics	Program	IMPLEMENTATION-TIME-FRAME, PERIOD
	Costs	INFLATION-FACTOR, REGIONAL-COST-FACTOR, COST-BASIS, INFORMATION-TECHNOLOGY-STANDARD-COST-MANAGEMENT, MATERIEL-ITEM-COST, NODE-SYSTEM-COST-MANAGEMENT

4.2.6 References

Assistant Secretary of Defense for Command, Control, Communications, Computers, and Intelligence (C4I), *C4ISR Core Architecture Data Model (CADM) Version 2.0*, Final Report, 1998.

The CADM is available for download at http://www.aitcnet.org/dodfw/

The CADM also may be downloaded from the Department of Navy Web site. The file is in Erwin format and requires Erwin data modeling software Version 2.5 or higher. The URL is: http://www.don-imit.navy.mil/adpm/ADPMFiles/ ArchitectureDevelopmentProcessModel.htm#Downloadable

4.3 ARCHITECTURE MODELING AND REPOSITORY TOOLS ASSESSMENT CRITERIA AND APPROACH

4.3.1 Introduction

This purpose of this section is to provide criteria for evaluating architecture modeling tools and architecture data repository tools. The goal is to provide assessment criteria for evaluating architecture modeling and repository tools with respect to support for DoD Architecture Framework (DoDAF) products and DoD processes. Tools are also to be evaluated with respect to an integrated approach for dealing with architecture data elements and architecture design and modeling efforts.

4.3.2 Scope

The scope of the evaluation criteria is modeling tools for producing architecture products and repository tools that store data and their metadata. **Figure 4.3-1** illustrates this scope. Tools for various purposes and uses are illustrated against the system development processes. The scope of this report is limited to the architecture modeling and repository tools shown in a block box and does not include other tools (such as acquisition tools or decision support tools for example).

4.3.3 Uses of an Architecture Tool Set

An architecture tool set may be used by architects to build architectures and by managers to:

- Serve as a centralized repository to effect communication
- Organize, integrate, and roll up architecture information across organizations
- Identify information technology (IT) systems and standards, and associate them with architecture information
- Include capabilities for configuration and change management
- Facilitate identifying, organizing, and disseminating
  - The mission or joint vision
  - The operational processes
- Facilitate integrating architecture development within an organization
- Facilitate collaboration, information sharing, and information reuse
- Provide decision makers with better, more consistent information and tools
- Facilitate linking important program milestones and resource decisions to architecture activities

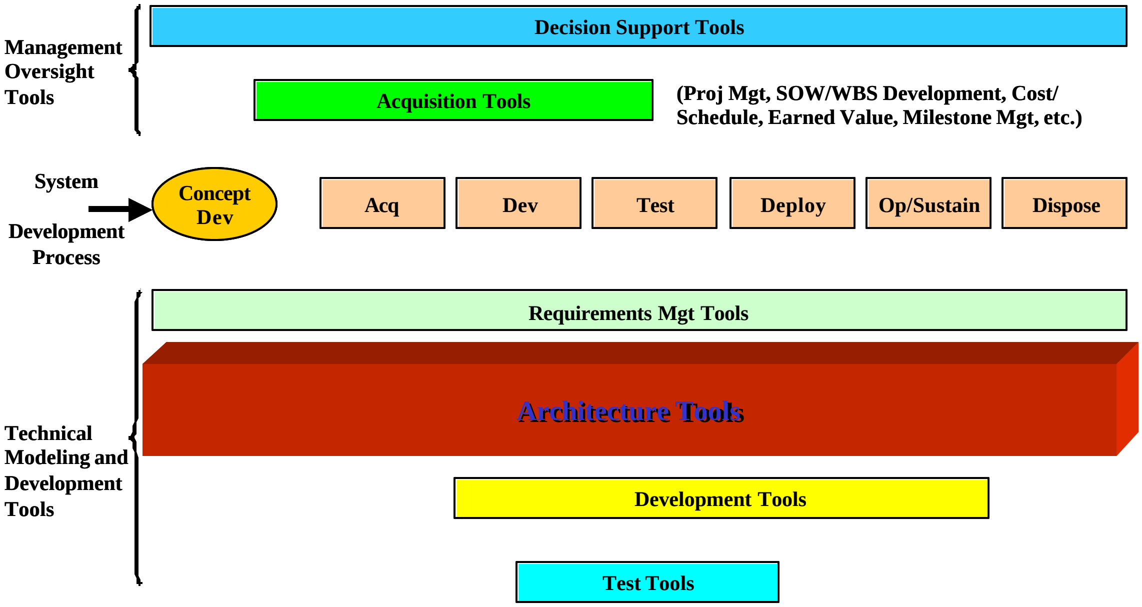

Management Oversight Tools
Decision Support Tools
Acquisition Tools
(Proj Mgt, SOW/WBS Development, Cost/ Schedule, Earned Value, Milestone Mgt, etc.)
System Development Process
Concept Dev
Acq
Dev
Test
Deploy
Op/Sustain
Dispose
Technical Modeling and Development Tools
Requirements Mgt Tools
Architecture Tools
Development Tools
Test Tools

Figure 4.3-1. Scope of Architecture Tools With Respect to Other Tools

4.3.4 Tool User Categories

A variety of users may need to use architecture tools to access architecture information. The following are categories of users:

- **Architecture Designers and Developers**: Require direct support through modeling, modeling standards, and customization capabilities
- **Architects**: Need to maintain, update, and oversee the architectural data elements, and work products across the organization
- **Planners, Stakeholders, and Management**: Need to run analysis, obtain guidance, and evaluate baseline and current models
- **Browsers**: Need specific views and perspectives of the architecture via technologies such as the HTML

Several user characteristics influence the choice of architecture modeling tools. Users:

- May be in several locations
- Have a variety of IT platforms
- Require numerous mechanisms to access the information
- Can view relatively static information on Web pages
- Have interactive access to components and relationships

4.3.5 Tools Assessment Criteria

To aid tool users in evaluating and deciding on a tool or tool set for their organization, the following sets of criteria have been developed based on industry best practices and current research on architecture modeling and repository tools. Architecture modeling and repository tools may be grouped into several sets depending on their use in the organization. **Figure 4.3-2** illustrates these sets, ranging from repository (relational or object-oriented [OO]) tools or database management systems (DBMSs) and development tools that form the foundation for constructing, storing, and manipulating architecture data, and ending with the web viewer tools and report generation tools that present the finished enterprise architecture (EA) models and architecture data to the architecture users who do not need to be expert architects or expert tools users to access and utilize the architecture data to aid them in making decisions.

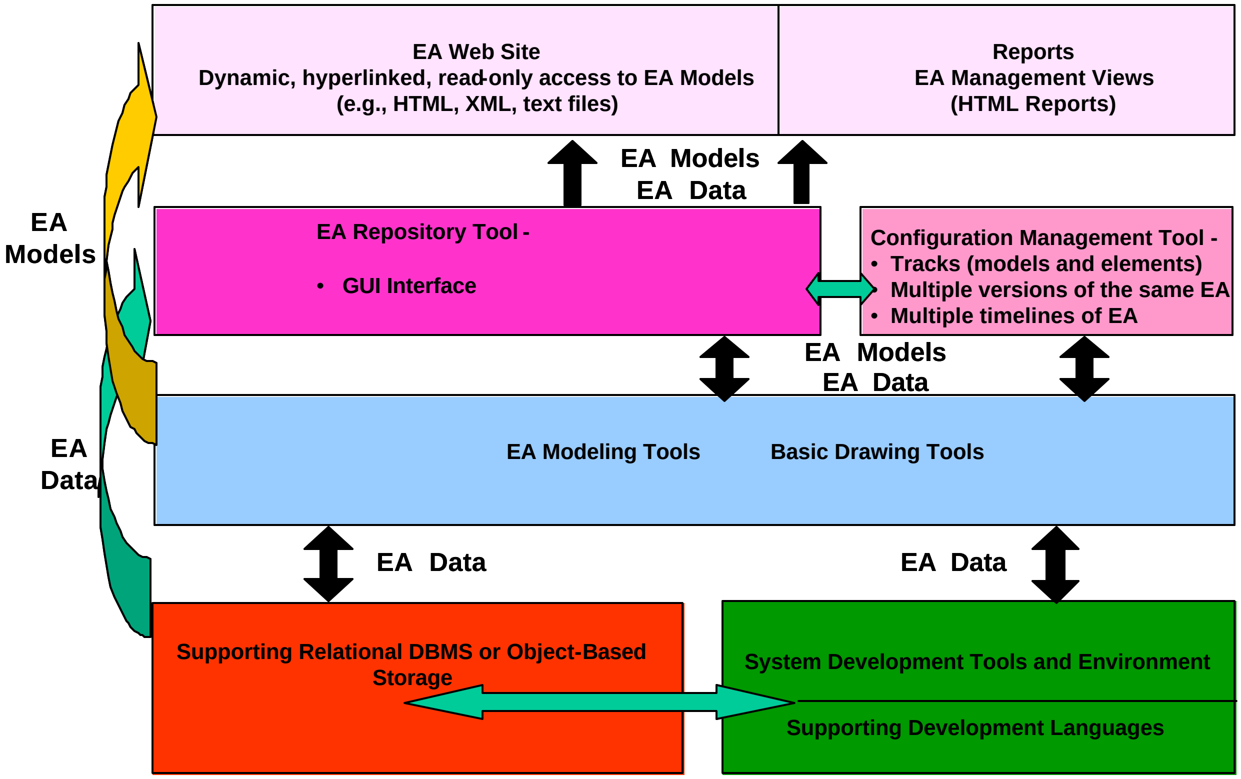

EA Web Site
Dynamic, hyperlinked, read-only access to EA Models
(e.g., HTML, XML, text files)
Reports
EA Management Views
(HTML Reports)
EA Models
EA Data
EA Models
EA Repository Tool -
• GUI Interface
Configuration Management Tool -
• Tracks (models and elements)
• Multiple versions of the same EA
• Multiple timelines of EA
EA Models
EA Data
EA Data
EA Modeling Tools
Basic Drawing Tools
EA Data
EA Data
Supporting Relational DBMS or Object-Based Storage
System Development Tools and Environment
Supporting Development Languages

Figure 4.3-2. Architecture Modeling and Repository Tool Suite

4.3.5.1 Framework Products, Modeling Support — Criteria

The first set of evaluation criteria is for evaluating architecture modeling tools or tools whose purpose is to create architecture models or products.

Architecture modeling tools should meet the following criteria:

- Ability to roll up and describe an organization architecture as a high-level summary for use in planning, budgeting, decision analysis, etc.
- Ability to link cost and budgeting information to architecture elements
- Ability to describe the architecture of complex systems for use in system development
- Ability to build an architecture, as described by the Framework
- Ability to organize Framework products into views that are subsets of the organization information architecture

- Ability to support views of time-based architecture (i.e., current, current+n months/quarter/years, target)
- Ability to customize and enforce robust traversal relationships between Framework products and architecture data elements
- Ability to perform consistency and completeness checks among the various Framework products
- Ability to choose modeling notation and methodology
- The scope of the products encompasses architecture information description for the whole organization
- The products illustrate the essential information flows
- Tool offers a variety of industry accepted modeling standards (e.g., Unified Modeling Language [UML], integration definition for data modeling [IDEF0], etc.)
- Ability to customize data dictionary capability with attributes and relationships, as required by the Framework
- Ability to support simulation

4.3.5.2 EA Repository Tools — Criteria

The second set of evaluation criteria is for evaluating architecture repository tools or tools whose purpose is to create, store, and provide access to architecture data for use in architecture models or products.

EA repository tools should meet the following criteria:

- Ability to maintain architecture data in a repository/database using a non-proprietary, commercial DBMS based on relational technology, persistent object storage, or using XML
- Ability for user customization and manipulation of the data schema or the persistent object attributes
- Ability to generate custom reports
- Ability to create, update, delete, and retrieve data from repository (knowledge) base using a graphical user interface (GUI)
- Ability to use simple queries to generate high-level, summary reports for management from the architecture data that facilitate acquisition, requirements generation/management, or budgeting decisions
- Ability to populate data repository by importing architecture data elements and data from external data sources

4.3.5.3 Customization Support — Criteria

The third set of evaluation criteria is for evaluating the ability of the tool suite to allow customization in support of varying user needs and user environments.

A tool suite should support the following:

- Ability to provide formal graphical modeling symbols
- Ability to create custom symbols
- Ability to import third-party graphical symbols
- Ability to add custom icons to the tool's set of modeling symbols
- Ability to customize diagrams
- Ability to create report templates
- Provide an easily extendable internal structure (e.g., ability to add user defined properties)
- A capability to collect and publish various architecture products (diagrams, tables, and requirements) in standard document templates
- Ability to support queries and custom reports within specific architectures and across groups of architectures

4.3.5.4 Interoperability — Criteria

The fourth set of evaluation criteria is for evaluating the ability of the tool suite to interoperate with other tools.

A tool suite should support the following:

- Ability to integrate with other tools
- Two-way interfaces for architecture models to multiple tools including notation and semantics
- Interface with office automation and productivity tools
- Import/export database information (entities, attributes, and relationships) from other existing DBMSs, or object-based storage using open standards and techniques (e.g., Open Data Base Connectivity [ODBC])
- Ability to support multiple data exchange formats
- Enable data sharing (import/export) with other tools via standard formats (e.g., Comma Separated Values [CSV] file formats, XML)
- Ability to support defined, published import/export interface (e.g., XMI)
- Provide open standard Application Program Interface (API)

4.3.5.5 General Purpose Characteristics — Criteria

The fifth set of evaluation criteria is general purpose criteria that apply to any of the tools in the tool suite.

A tool suite should support the following:

- Configuration management (CM) of model data

- Ability to create, maintain, and compare different versions
- Ability to group versions by architecture and by product within the architecture
- Ability to support other CM functions such as change management and status accounting
- Ability to track ownership of data entered
- Ability to enforce/customize various security standards
- Ability to support a multiuser environment
- Ability to support collaboration among project team members
- Ability to provide read-only Intranet access or ability to generate HTML
- Ability to support direct HTML publishing and/or offer a free viewer
- Ability to support a Web interface (with access to the models or data repository from geographically distributed locations)
- Scalability (to thousands of architectural elements and relationships, and multiple versions of the architecture)
- Adaptability (to new standards, techniques, etc.)
- Support various IT platforms (e.g., Windows, Unix, or both)
- Cost of ownership (initial and ongoing maintenance costs, training costs)
- Usability, refers to the quality of a user's experience when interacting with the EA tool
- Short learning curve, reasonably easy to use
- Ease of use of GUI (e.g., MS Explorer-like interface)
- Ease of use of query capability (e.g., is knowledge of a query language needed?)
- Spell check capability
- Adaptable/customizable user interface

4.3.5.6 Vendor Assessment — Criteria

The sixth set of evaluation criteria is general purpose criteria that apply to tool vendors. Figure 4.3-2 highlights the set of tools covered by these criteria. The criteria are listed below.

These criteria support the following:

- Training: The vendor provides training or training material to help users learn how to use the tool. Kinds of training offered should include:
  - Classroom
  - Computer-based training/tutorial
  - Customized training

- Quick training time (3–5 days)
- Technical support
- Online help
- User manuals and support documentation
- Help-desk response time - quality of vendor support
- Maintenance agreement upgrades
- Vendor Stability: The vendor is a recognized, stable tool vendor
- Customer categories and experience (e.g., Military, Federal, private industry, etc.)
- Target market
- Number of installed licenses
- Number of years in business
- Product development history (“roots”)
- Tool Release Schedule
- Vendor’s future plans for the package

4.3.6 Assessment Approach

The following approach can be used for assessments:

- Weights are assigned to evaluation criteria. For example, each criterion can be assigned a weight on a scale from 4 to 1. The weights reflect the users’ needs:
  - 4 = must have (i.e., tool must satisfy criterion)
  - 3 = important to have
  - 2 = desirable to have
  - 1 = nice to have
- For each criterion, scores are assigned to each tool based on testing results. For example, scores can be based on a 3-point scale:
  - 1 (if tool meets criterion)
  - 0 (if tool does not meet it)
  - 0.5 (if tool only partially meets criterion)
- Measurements are calculated for each criterion per each tool based on the criterion weight multiplied by the tool score for that criterion.
- Totals for each tool are computed by summing up the total measurements for the tool.
- The total obtained for each tool can be compared to totals of other tools, and a final decision can be made based on the totals obtained.

4.3.7 Issues with Choosing a Tool

The following issues exist when dealing with choosing and adopting an architecture and repository tool or tool suite.

- Currently, no one tool(s) meets all criteria. Therefore, users need to choose a tool(s) that currently exists and satisfies immediate needs, and has the potential to meet the criteria in the future.
  - Mitigation: Choose a tool(s) that provides the most open interface to industry-standard data formats and to other industry-standard tools.
- Initial investment costs (i.e., cost, training, learning curve) are incurred when introducing new tools and processes.
  - Mitigation: Weigh long-term cost-benefit analysis against potential cost overruns if automated tools and new processes are not introduced.
- Several groups are responsible for related architectures but are using non-interoperable architecture tools. This results in disjoint architectures that can be readily compared, or integrated.
  - Mitigation: Groups should not be forced to use one "standard" tool.
  - Use tools compliant with industry-standard data formats
  - Use tools that follow a common data model
  - Use integrated repository to:
    - -- Bring together architectures and EA data information
    - -- Enable chief architect to make sound investment decisions

4.3.8 Issues With Organizational Use of Automated Tools

Many types of issues are associated with organizational use of architecture modeling tools and repositories including:

- Programmatic issues
  - How to roll up architecture information - authority
- Architecture issues
  - Limited time and resources to define criteria and assessment approach and choose and customize too
  - Resolution of data naming conflicts
- Policy compliance issues
  - While policy requires use of a common data model, there is no enforcement mechanism.
  - A common data taxonomy is needed for interoperability but currently is not supported by policy.

4.3.9 Recommended Solution

A recommended solution is to follow the CADM as the common data model, and to use tools that allow the direct import and export of architecture data between the chosen tools and the CADM-based data repository. **Figure 4.3-3** illustrates this recommended approach showing one such common data repository.

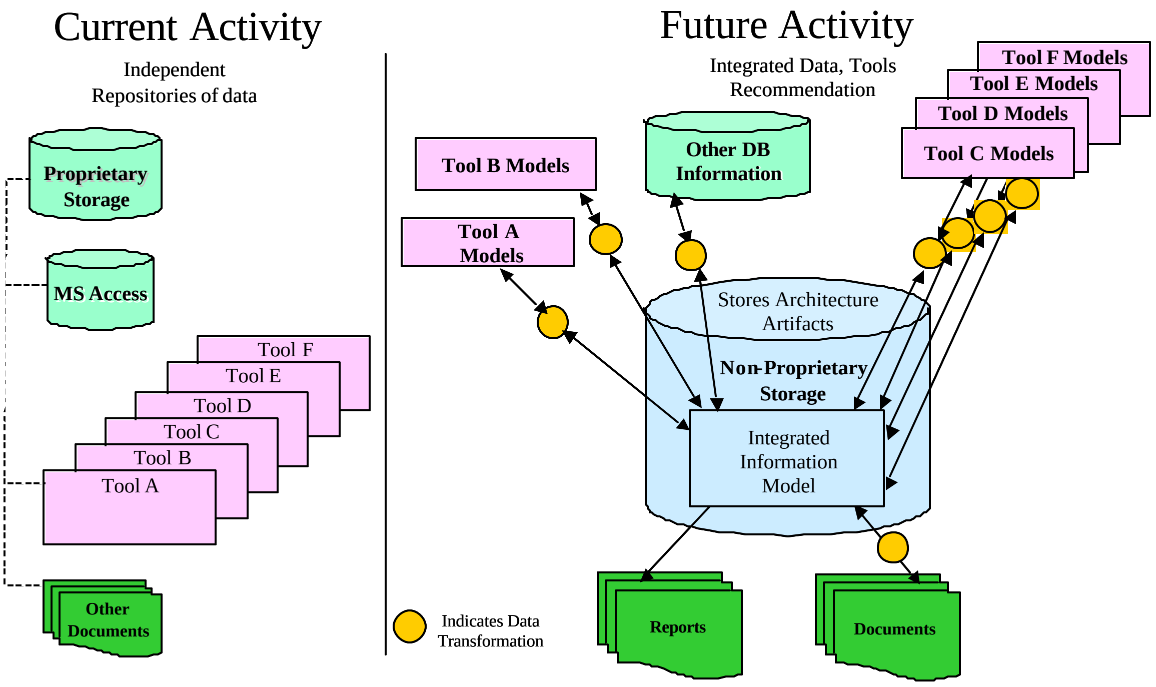


Figure 4.3-3. Approaches to Utilizing Tools in Supporting Architecture Development

4.3.9.1 Conclusion

This section provided criteria for evaluating architecture modeling and repository tools coupled with an assessment approach. The criteria are based on industry best practices and experience.

4.3.10 References

Dandashi, Dr. Fatma, MITRE, *Architecture Modeling and Repository Tools: Assessment Criteria and Approach,* Briefing sponsored for the Federal CIO Council, October 5, 2002.

International Council on Systems Engineering (INCOSE) System Architecture, Tools Survey, Available: http://www.incose.org/tools/tooltaxs.html, September 2001.

OMB Circular A-130, *Management of Federal Information Resources*, Available: http://www.whitehouse.gov/omb/circulars/a130/a130trans4.html, November 30, 2000.

4.4 FEDERAL ENTERPRISE ARCHITECTURE REFERENCE MODELS – RELATIONSHIP TO DOD AND DOD ARCHITECTURE FRAMEWORK

4.4.1 Introduction

This section of the Deskbook provides an overview of the Federal Enterprise Architecture (FEA) Reference Models (RMs), relates them to comparable DoD processes or enablers, and summarizes the key aspects of each. Relationships between the RMs and DoD Architecture Framework (DoDAF) products are also summarized.

4.4.2 Comparison Between DoD and Office of Management and Budget (OMB) Approach to Architecture

4.4.2.1 DoD and Architecture

DoD Components must integrate many business and operational processes, data flows, and infrastructures. An architecture description provides a defined and structured mechanism for depicting operational, systems, and technical standards structures and their inter-relationships to facilitate interoperability analysis and to eliminate redundant efforts.

The DoDAF defines a common approach for describing, presenting, and integrating DoD architectures to ensure that architecture descriptions can be compared and related across organizational boundaries, including Joint and multinational boundaries. The architecture products described in the Framework are an integrated set of models that capture relationships for understanding and analyzing dependencies and impacts. The DoDAF provides guidance for developing architectures that are useful enablers for conducting DoD processes.

4.4.2.2 OMB and Architecture

Outside DoD, economy demands that all elements of an enterprise fit together and work well with minimal investment while taking advantage of reuse and eliminating unnecessary redundant efforts. Elements of an enterprise include the business processes, organizations responsible for them, information and systems data they need to inter-operate, information technology (IT) capabilities, systems, infrastructure, and specific technical standards that facilitate their inter-operation. An enterprise architecture (EA) describes these elements, their structures, and inter-relationships to facilitate capital planning and IT development sequencing.

OMB's predominant mission is to assist the President in overseeing the preparation of the Federal budget and to supervise administration in Executive Branch agencies. To facilitate this mission, OMB Circular A-130 *Management of Federal Information Resources* describes the required content of an EA that must be submitted to OMB. In the creation of an EA, agencies must identify and document business processes, information flow and relationships, applications, data descriptions and relationships, and technology infrastructure and include a technical reference model and standards profile. During 2002, OMB proposed the development of an FEA from a set of RMs. These RMs create a comprehensive government-wide framework to guide agency IT investment activities, identify opportunities for collaboration and consolidation of initiatives, and integrate government activities.

"The Federal Enterprise Architecture is being constructed through a collection of inter-related *reference models* designed to facilitate cross-agency analysis and opportunities for collaboration." [1]

OMB's motivation for defining the RMs is based on the President's E-Government initiatives that identified the lack of an FEA approach as a critical obstacle to implementing the initiatives. OMB plans to enforce the use of RMs through the budget process. The FEA consists of five RMs:

- The Business Reference Model (BRM) lists all lines of business and sub-functions that major IT investment supports.
- The Service-Component Reference Model (SRM) consists of categories of IT components included in major IT investments.
- The Technical Reference Model (TRM) consists of Service Areas, Service Categories, Service Standards, and Service Specifications that collectively describe the technology supporting a major IT investment.
- The Performance Reference Model (PRM) identifies performance measures and will be enforced starting in 2005 and beyond. To be compliant with OMB requirements, an agency must identify performance information that pertains to any major IT investment.
- The Data Reference Model (DRM) has not been defined yet by OMB but may consist of types of data that will be used in a major IT investment project.

Figure 4.4-1 is an OMB diagram that relates the FEA RMs to each other.

[1] FEA-PMO Using the Business and Performance Reference Models to Help Improve Citizen Services, Norman Lorentz, October 7–8, 2002.

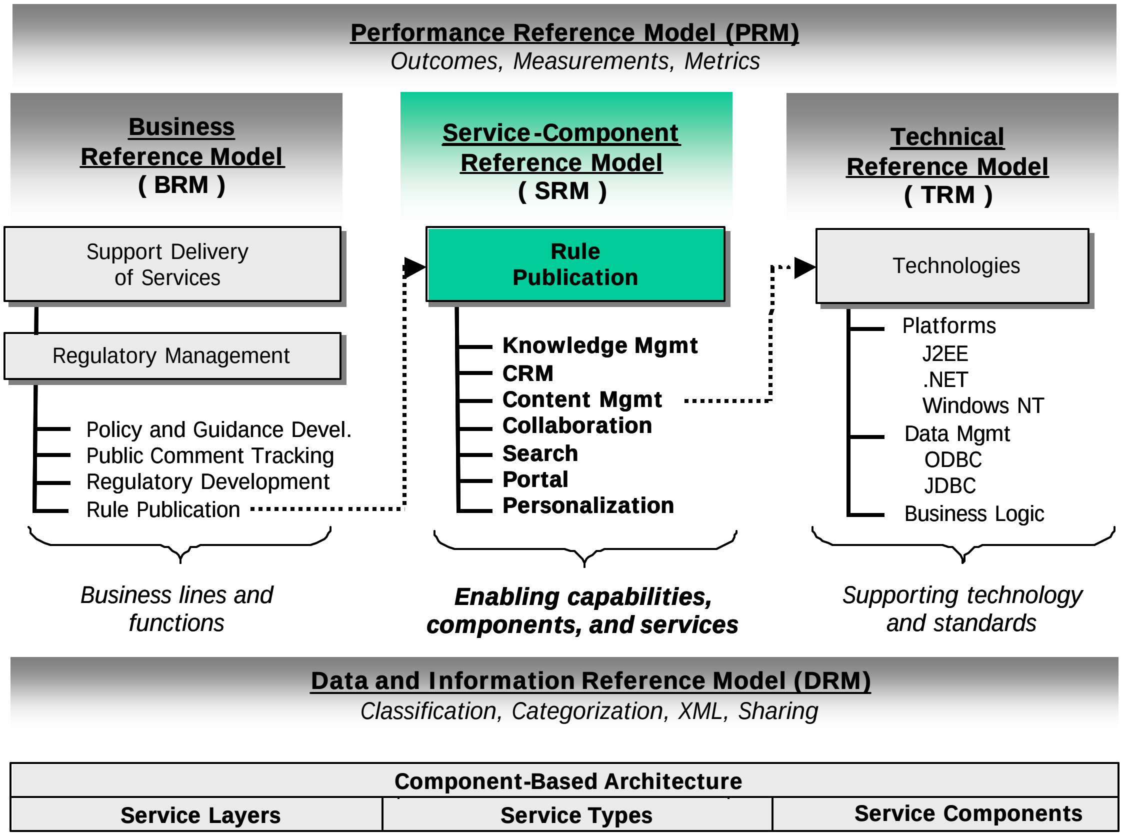

Performance Reference Model (PRM)
Outcomes, Measurements, Metrics
Business Reference Model (BRM)
Support Delivery of Services
Regulatory Management
Policy and Guidance Devel.
Public Comment Tracking
Regulatory Development
Rule Publication
Business lines and functions
Service-Component Reference Model (SRM)
Rule Publication
Knowledge Mgmt
CRM
Content Mgmt
Collaboration
Search
Portal
Personalization
Enabling capabilities, components, and services
Technical Reference Model (TRM)
Technologies
Platforms
J2EE
.NET
Windows NT
Data Mgmt
ODBC
JDBC
Business Logic
Supporting technology and standards
Data and Information Reference Model (DRM)
Classification, Categorization, XML, Sharing
Component-Based Architecture
Service Layers
Service Types
Service Components

Figure 4.4-1. FEA Reference Models (used from FEA PMO Web site)

4.4.2.3 Summary of DoD and OMB Architecture Approaches

DoD and OMB have devised similar approaches to solve some of the same problems.

Since 1995, DoD has mandated the development of integrated architectures (using the DoDAF) as enablers for the decision maker and as facilitators in the execution of DoD processes (e.g., acquisition, capability analysis, etc.). DoD mandates (i.e., directives, instructions, and manuals) that relate to architectures have been discussed in sections 2 and 3 of Volume I. Within DoD, integrated architectures provide a common method for aggregating information and a common basis for capability analysis.

Since 2002, OMB has realized the need for a common business-based structure that facilitates cross-agency analysis, analysis of IT investments, and other capital assets, which can serve as the foundation for budget and performance reporting. OMB has defined the FEA as a mandate to Federal agencies for the development of an EA. Circular A-11, *Preparation, Submission, and Execution of the Budget,* requires agency Capital Asset Plans (also called Exhibit 300 business cases[2], see OMB Circular A-11 Section 300.7) to be mapped against OMB's FEA RMs. OMB will use the FEA to manage the budget. In 2002, the Exhibit 300 had to be related to the BRM. Mappings to other models are now required, since more RMs have been published.

[2] The Exhibit 300 is a format to demonstrate to agency management and OMB that it has employed the disciplines of good project management, represented a strong business case for the investment, and met other administration priorities to define the proposed cost, schedule, and performance goals for the investment if funding approval is obtained.

OMB budgeting objectives are to:

- Relate IT investment budget submissions within and across Federal Government Agencies
- Use common patterns to identify commonality
- Have common performance measures
- Use Federal Enterprise Architecture Management System (FEAMS) to create reports

OMB has mandated that Federal Agencies use FEAMS, a government-developed database management system with a graphical user interface. Agencies are required to capture their information organized by the RM categories and to manually enter it into FEAMS via the Web interface, thereby completing the budget submission process to OMB. OMB then will have the information from all Federal Agencies and will be able to query the information to look for cross-agency initiatives, redundancy, and performance comparisons.

4.4.3 Comparison Between DoD Enablers and Equivalent FEA RMs

4.4.3.1 DoD Budget Titles and FEA Business RM

DoD Budget Titles: DoD categorizes its budget by the following set of budget titles:

- Military Personnel
- Operations and Maintenance
- Other Related Agencies
- Research, Development, Test, and Evaluation
- Procurement
- Military Construction
- Family Housing
- Revolving and Management Funds
- Base Realignment and Closure

In addition to budget titles, DoD Business and Military Functions can be divided into two major areas: (1) Military Operations, which are further categorized under Joint Mission Areas and Universal Joint Task Lists (UJTLs); and (2) Military Operations Other Than War (MOOTW), which include other functions that DoD conducts to support its main mission of defending the nation. Some of these functions include those documented by the Business Management Modernization Program's Business Enterprise Architecture areas, such as:

- Accept Real Property Strategic Plan
- Accept Real Property Work
- Accumulate Cost Detail Information

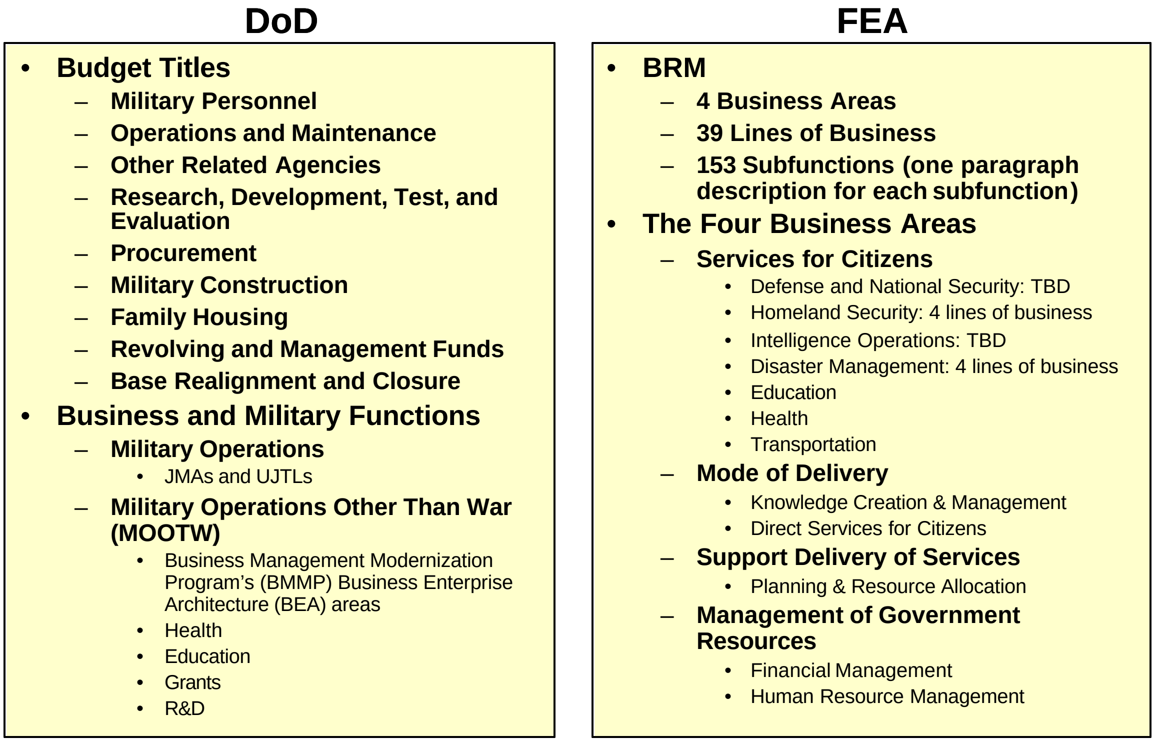
- Accumulate General Ledger Data
- Accumulate Non-Financial Data

The DoD UJTLs were developed to communicate requirements for training (stated as *task*). However, they can be used as a common basis for defining warfighting activities. The intent is to integrate UJTLs to form a complete picture of the operations. UJTLs have several levels of decomposition, and they are associated with performance metrics that state the required performance needed to accomplish a task.

FEA BRM: OMB requires that all budget submissions requesting a major investment in an IT system be categorized as belonging to one of the BRM business areas or subcategories.

- **Purpose**: Define and communicate high-level view of how—in business terms—the Federal Government achieves its various missions
- **Content**: Business functions
- **Organization**: Hierarchical division of three Business Areas (Services to Citizens to include Mode of Delivery, Support Delivery of Services, and Management of Government Resources) into Lines of Business, then 137 subfunctions

Summary Comparison Between DoD Budget Titles and FEA BRM: A summary comparison to the two approaches to classify business areas appears in **Figure 4.4-2**.



DoD

- **Budget Titles**
  - **Military Personnel**
  - **Operations and Maintenance**
  - **Other Related Agencies**
  - **Research, Development, Test, and Evaluation**
  - **Procurement**
  - **Military Construction**
  - **Family Housing**
  - **Revolving and Management Funds**
  - **Base Realignment and Closure**
- **Business and Military Functions**
  - **Military Operations**
    - JMAs and UJTLs
  - **Military Operations Other Than War (MOOTW)**
    - Business Management Modernization Program's (BMMP) Business Enterprise Architecture (BEA) areas
    - Health
    - Education
    - Grants
    - R&D

FEA

- **BRM**
  - **4 Business Areas**
  - **39 Lines of Business**
  - **153 Subfunctions (one paragraph description for each subfunction)**
- **The Four Business Areas**
  - **Services for Citizens**
    - Defense and National Security: TBD
    - Homeland Security: 4 lines of business
    - Intelligence Operations: TBD
    - Disaster Management: 4 lines of business
    - Education
    - Health
    - Transportation
  - **Mode of Delivery**
    - Knowledge Creation & Management
    - Direct Services for Citizens
  - **Support Delivery of Services**
    - Planning & Resource Allocation
  - **Management of Government Resources**
    - Financial Management
    - Human Resource Management

Figure 4.4-2. Comparison Between DoD Budget Titles and FEA BRM Structure

4.4.3.2 DoD Standard Service-Components and FEA SRM

DoD Net-Centric Enterprise Services (NCES): The objective of NCES is to provide timely, secure access to decision-quality information to all DoD users via the Global Integrated Grid. NCES defines common DoD services that enable other programs to operate in a net-centric environment. NCES includes:

- Enterprise Service Management
- Messaging Services
- Discovery Services
- Mediation Services
- Collaboration Services
- User Assistant Services
- Application Services
- Security Services
- Storage Services

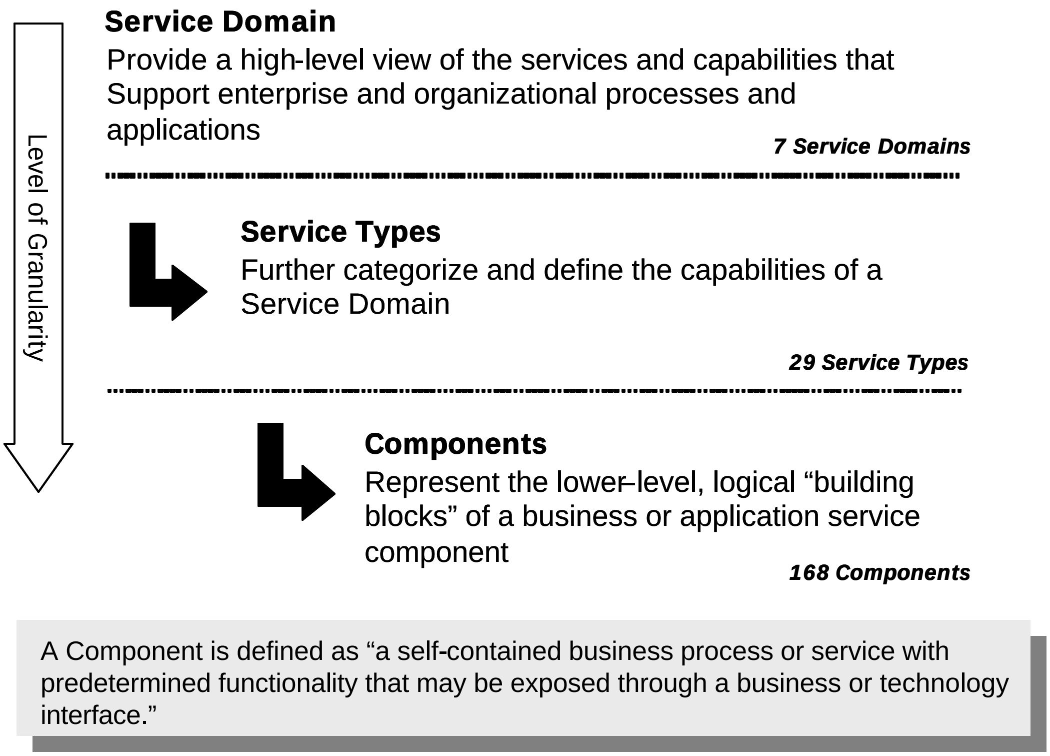
Definitions of each of these services and additional information on NCES are provided in Section 2.7: An Architecture Perspective on NCOW and Section 4.5: Universal Reference Resources.

FEA SRM: The SRM is a component-based framework intended to provide—independent of business function—a leverageable foundation for reuse of applications, application capabilities, components, and business services.

- **Purpose**: Aid in recommending service capabilities to support the reuse of business components and services across the Federal Government
- **Content**: Identify and classify horizontal and vertical service components that support Federal agencies and their IT investments and assets[3]
- **Organization**: Service Domains comprised of Service Types with Service Components

As illustrated in **Figure 4.4-3**, the SRM consists of 7 Service Domains comprised of 29 Service Types that further categorize and define the capabilities of a Service Domain. The 168 Components represent the lower-level, logical "building blocks" of a business or application service component.

[3] In the SRM context, a component is defined as a self-contained business process or service with predetermined functionality that may be exposed through a business or technology interface.



Service Domain

Provide a high-level view of the services and capabilities that Support enterprise and organizational processes and applications

7 Service Domains

Service Types

Further categorize and define the capabilities of a Service Domain

29 Service Types

Components

Represent the lower-level, logical "building blocks" of a business or application service component

168 Components

A Component is defined as "a self-contained business process or service with predetermined functionality that may be exposed through a business or technology interface."

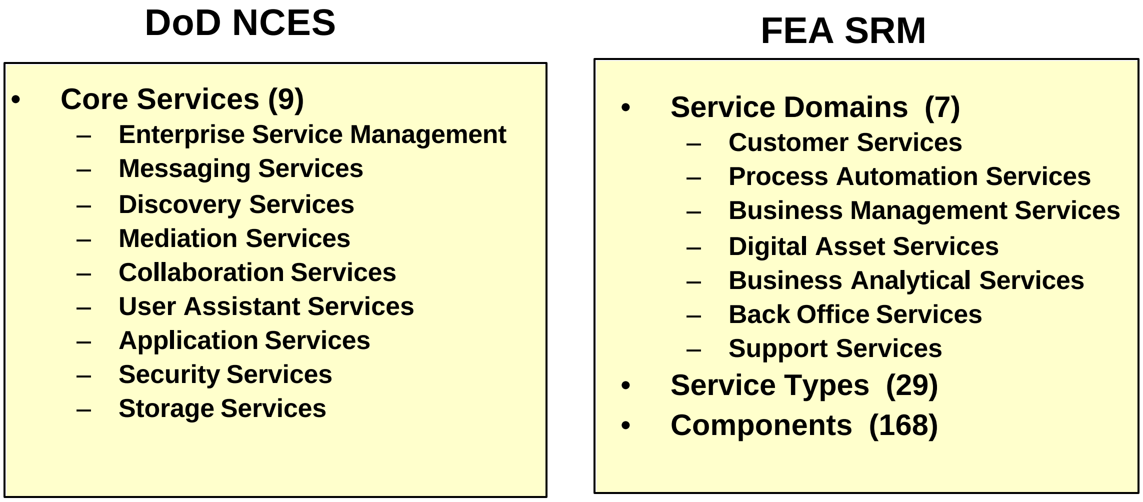
Figure 4.4-3. FEA SRM Categories

The SRM assumes that the services will be provided over the Internet. The seven Service Domains of the SRM are:

- **Customer Services Domain** – Capabilities that are directly related to the end customer, interaction between the business and the customer, and customer-driven activities or functions; consists of 3 Service Types and 21 Components
- **Process Automation Services Domain** – Capabilities that support the automation of process and management activities and assist in effectively managing the business; consists of 2 Service Types and 5 Components
- **Business Management Services Domain** – Capabilities that support the management and execution of business functions and organizational activities that maintain continuity across the business and value-chain participants; consists of 4 Service Types and 20 Components
- **Digital Asset Services Domain** – Capabilities that support the generation, management, and distribution of intellectual capital and electronic media across the business and extended enterprise; consists of 4 Service Types and 25 Components

- **Business Analytical Services Domain** – Capabilities that support the extraction, aggregation, and presentation of information to facilitate decision analysis and business evaluation; consists of 4 Service Types and 19 Components
- **Back Office Services Domain** – Capabilities that support the management of enterprise planning transactional-based functions; consists of 6 Service Types and 47 Components
- **Support Services Domain** – Cross-functional capabilities that can be leveraged independent of Service Domain objective or mission; consists of 6 Service Types and 31 Components

<u>Summary Comparison Between DoD's Standard Service-Components and FEA SRM</u>: **Figure 4.4-4** below illustrates the NCES and SRM categories.



DoD NCES

- Core Services (9)
  - Enterprise Service Management
  - Messaging Services
  - Discovery Services
  - Mediation Services
  - Collaboration Services
  - User Assistant Services
  - Application Services
  - Security Services
  - Storage Services

FEA SRM

- Service Domains (7)
  - Customer Services
  - Process Automation Services
  - Business Management Services
  - Digital Asset Services
  - Business Analytical Services
  - Back Office Services
  - Support Services
- Service Types (29)
- Components (168)

Figure 4.4-4. Comparison Between DoD NCES and FEA SRM Structure

4.4.3.3 DoD TRM/Joint Technical Architecture (JTA) and FEA TRM

<u>DoD TRM and JTA</u>: The DoD TRM contains a comprehensive set of service and interface definitions to support emerging and legacy information systems and IT applications. The DoD TRM is the foundation of the JTA as well as of other initiatives (such as the Common Operating Environment [COE]). The JTA defines Service Areas, Interfaces, and Standards applicable to all DoD systems. The TRM is the key source of service and interface definitions that provide part of the JTA document structure and the accompanying service descriptions. The DoD TRM and JTA are discussed in Section 4.5: Universal Reference Resources.

JTA service areas are:

- Information-Technology Standards
- Information-Processing Standards
- Information-Transfer Standards
- Information Modeling, Metadata, and Information Exchange Standards
- Human Computer Interface (HCI)
- Information Security Standards
- Physical Services Standards

Each service within the service areas contains a listing of acceptable standards for use throughout DoD.

FEA TRM: The TRM outlines standards, specifications, and technologies that support the Federal Government's IT transition towards interoperable E-Government solutions. FEA defines the TRM as:

- **Purpose**: Compliment and guide agency, E-Government TRMs; focus on Internet, Component-based architectures; support trade-off analysis
- **Content**: Standards and specifications for service components (e.g., Palm Pilot, XML Schema, and SNMP)
- **Organization**: Divided into four core Service Areas (Service Areas group the standards, specifications, and technologies into lower-level functional areas.)

Figure 4.4-5 illustrates TRM Service Areas and their associated categories within an IT environment.

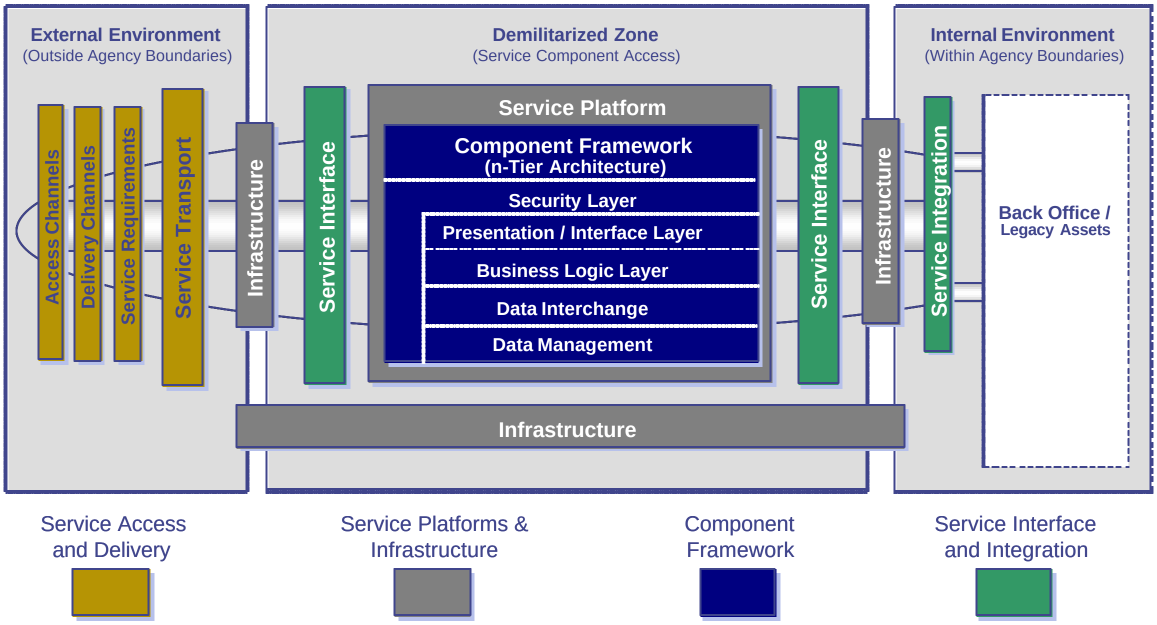


Figure 4.4-5. FEA TRM

The following defines the Service Areas, Service Categories, and Standards Categories for the FEA TRM.

Service Area is a technical tier that supports the secure construction, exchange, and delivery of business or service components. Each Service Area groups the requirements of component-based architectures within the Federal Government into *functional* areas. Each Service Area aggregates and groups the standards, specifications, and technologies into lower-level functional areas. There are four Service Areas within the TRM, which are defined as follows:

- ***Service Access and Delivery*** refers to the collection of standards and specifications to support external access, exchange, and delivery of Service Components or capabilities. This area also includes the Legislative and Regulatory requirements governing the access and usage of the specific Service Component.
- ***Service Interface and Integration*** refers to the collection of technologies, methodologies standards, and specifications that govern how agencies will interface (both internally and externally) with a Service Component. This area also defines the methods by which components will interface and integrate with back office/legacy assets.
- ***Component Framework*** refers to the underlying foundation, technologies, standards, and specifications by which Service Components are built, exchanged, and deployed across Component-based, Distributed, or Service-orientated Architectures.
- ***Service Platform & Infrastructure*** refers to the collection of delivery and support platforms, infrastructure capabilities, and hardware requirements to support the construction, maintenance, and availability of a Service Component or capabilities.

Service Category a sub-tier of the Service Area to classify lower levels of technologies, standards, and specifications in respect to the business or technology function they serve. **Figure 4.4-6** depicts the relation between the Service Areas and Service Categories.

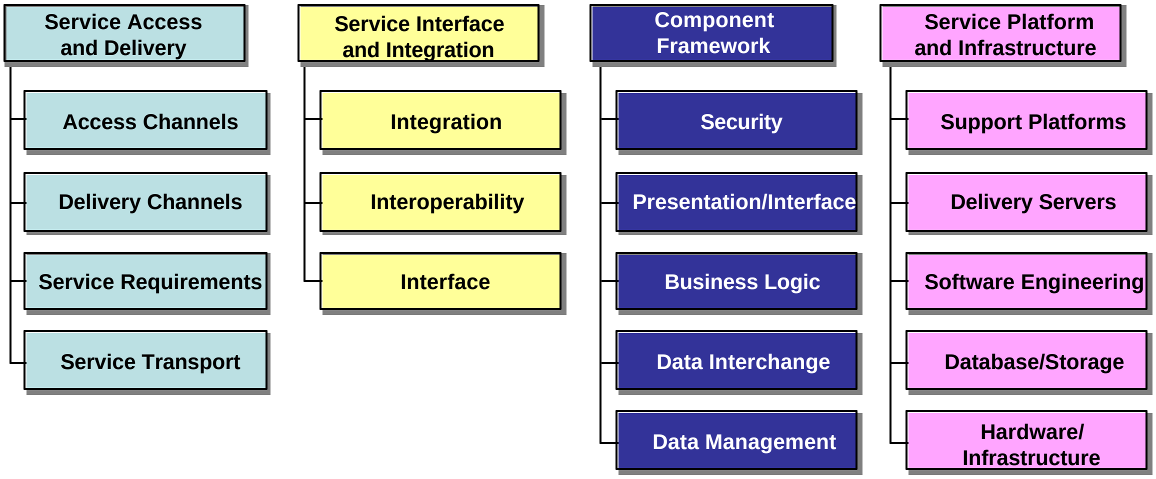
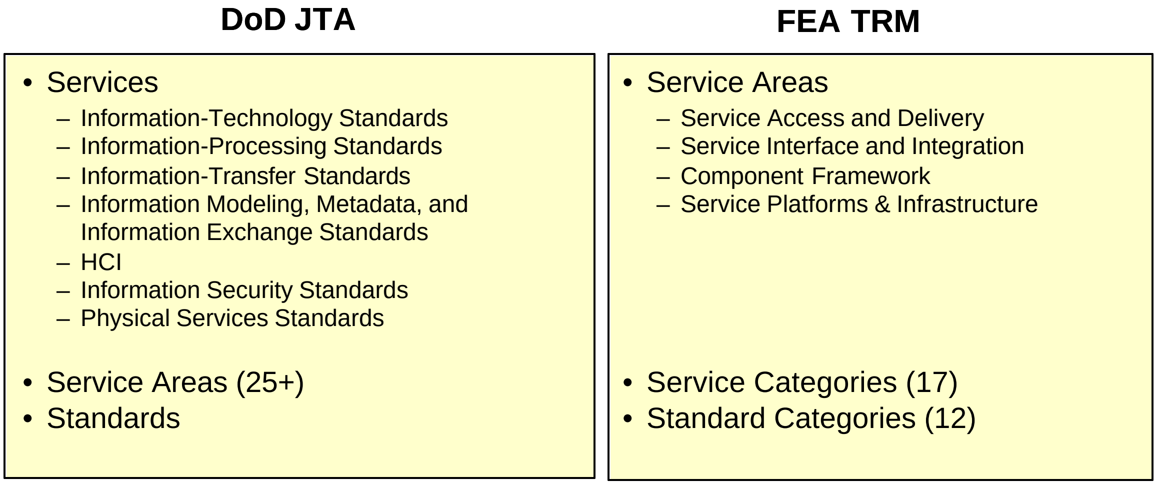


Figure 4.4-6. FEA TRM Service Areas and Categories

Standard includes hardware, software, or specifications that are widely used and accepted (de facto), or are sanctioned by a standards organization (de jure). Standards are typically categorized as follows:

- Programming Language Standards
- Character Code Standards
- Hardware Interface Standards
- Storage Media Standards
- Operating System Standards
- Communication and Networking Standards
- Machine Language Standards
- File System Management Standards
- Database Management System Standards
- Text Systems Standards
- Graphic Systems Standards
- Internet Standards

Summary Comparison Between DoD TRM and FEA TRM: **Figure 4.4-7** compares DoD JTA Service Areas to FEA TRM Service Areas.



DoD JTA

- Services
  - Information-Technology Standards
  - Information-Processing Standards
  - Information-Transfer Standards
  - Information Modeling, Metadata, and Information Exchange Standards
  - HCI
  - Information Security Standards
  - Physical Services Standards
- Service Areas (25+)
- Standards

FEA TRM

- Service Areas
  - Service Access and Delivery
  - Service Interface and Integration
  - Component Framework
  - Service Platforms & Infrastructure
- Service Categories (17)
- Standard Categories (12)

Figure 4.4-7. Comparison Between DoD TRM and FEA TRM Structure

4.4.3.4 DoD Performance-Based Budgeting Process and FEA PRM

DoD Performance-Based Budgeting Process: There are three primary sources for or inputs to a typical budgeting process—plans, performance, and people.

- **Plans**: An organization's plans and priorities should be an important driver to the budgeting process. Budgets should reflect the planned change initiatives of management, the costs of those initiatives, and the expected results.

- **Performance**: An organization's past and current performance, as well as the performance of like organizations, should contribute to the budget preparation process. However, careful consideration needs to be given to uncontrollable changes in the outside world that could dramatically affect the operation and its results. For example, many organizations, in both the public and private sectors, had to significantly change their projected forecasts in late 2000 due to rapidly rising energy costs and the slowdown in the economy.
- **People**: Good intra- and inter-organizational communications are essential to developing both good plans and good budgets. From customers to suppliers to internal personnel, the higher the quality of information, thought, and input into the process, the more likely a more realistic budget will result.

Under the DoD Planning, Programming, Budgeting, and Execution process, DoD evolves from an annual program objective memorandum and budget estimate submission (BES) cycle to a biennial (2-year) cycle starting with an abbreviated review and amendment cycle for FY05. The department will formulate 2-year budgets and use the off-year to focus on fiscal execution and program performance.

DoD uses budget change proposals (BCPs) instead of a BES during the off-year. BCPs accommodate fact-of-life changes (e.g., cost increases, schedule delays, management reform savings, workload changes, etc.) as well as changes resulting from congressional actions.

The FY05 execution reviews provide the opportunity to make assessments concerning current and previous resource allocations and whether DoD achieved its planned performance goals. Performance metrics, including the program assessment rating tool, will be the analytical underpinning to ascertain whether an appropriate allocation of resources exists in current budgets. To the extent performance goals of an existing program are not being met, recommendations may be made to replace that program with alternative solutions or to make appropriate funding adjustments to correct resource imbalances.

FEA PRM: The intent of the PRM is to provide a common and consistent framework for IT performance measurements.

OMB states that agencies will define the performance measures for each BRM function, and that these measures will be drawn from Federal Agencies, Balanced Scorecard, Baldrige Criteria, and private sector best practices and principles. The PRM will be applied during the FY05 budget formulation process.

- **Purpose**: Measure the ability of an agency to meet stated mission
- **Content**: Measures of IT performance (non-process specific) (e.g., customer satisfaction, cost effectiveness, and security)
- **Organization**: Measurement Areas with associated Measurement Categories and Generic Measurement Indicators

PRM organizational categories are defined as:

- **Measurement Areas** contain the high-level organizing framework of the PRM that captures aspects of performance at the input, output, and outcome levels. The draft PRM includes six measurement areas:

  - Mission and Business Results
  - Customer Results
  - Processes and Activities
  - Human Capital
  - Technology
  - Other Fixed Assets

  (Human Capital and Other Fixed Assets will not be used in FY05 budget formulation.)

- **Measurement Categories** are groupings within each Measurement Area that describe the attribute or characteristic to be measured. For example, the Mission and Business Results Measurement Area includes four Measurement Categories:
  - Lines of Business in Services for Citizens
  - Lines of Business in Support Delivery of Services
  - Lines of Business in Management of Government Resources
  - Lines of Business in Finance

- **Generic Measurement Indicators** are generic indicators (e.g., delivery time) that agencies can "operationalize" for their specific environments.

Figure 4.4-8 shows the six Measurement Areas with associated Measurement Categories and describes how they are designed to capture the relationships among inputs, outputs, and outcomes.

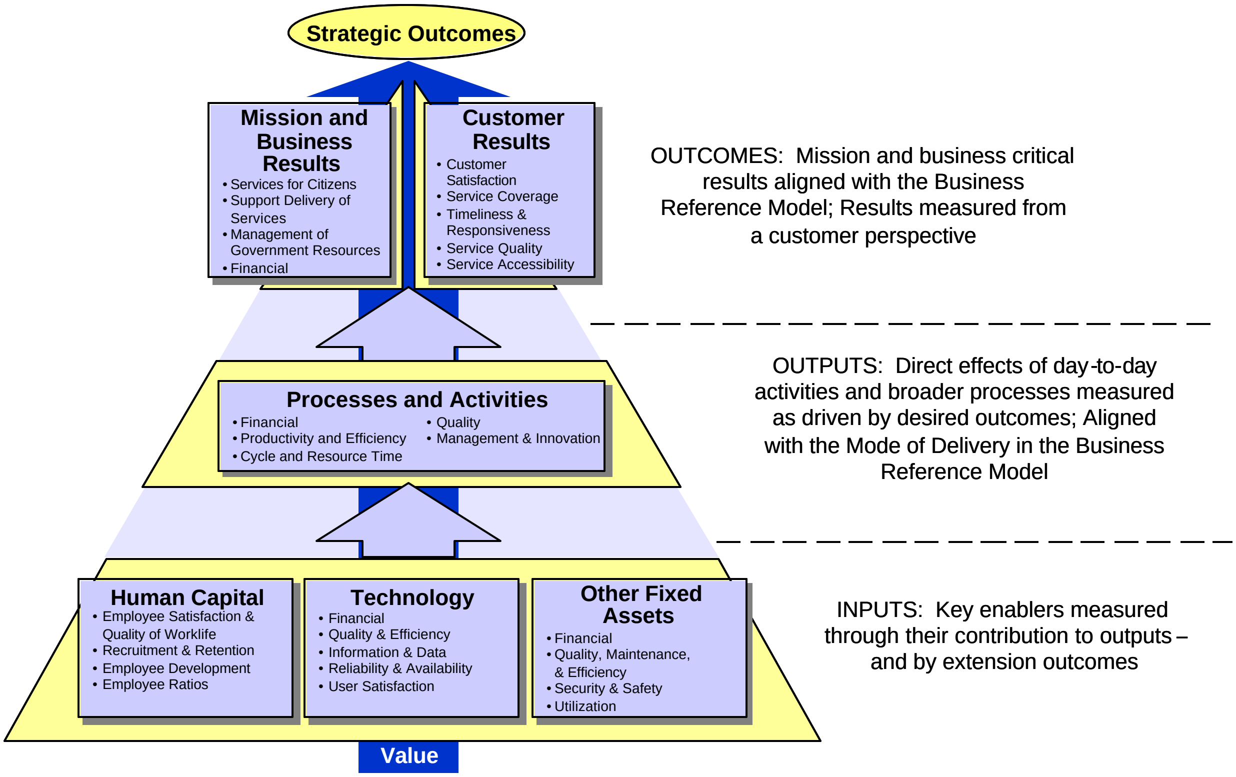

Strategic Outcomes
Mission and Business Results
• Services for Citizens
• Support Delivery of Services
• Management of Government Resources
• Financial
Customer Results
• Customer Satisfaction
• Service Coverage
• Timeliness & Responsiveness
• Service Quality
• Service Accessibility
OUTCOMES: Mission and business critical results aligned with the Business Reference Model; Results measured from a customer perspective
Processes and Activities
• Financial
• Productivity and Efficiency
• Cycle and Resource Time
• Quality
• Management & Innovation
OUTPUTS: Direct effects of day-to-day activities and broader processes measured as driven by desired outcomes; Aligned with the Mode of Delivery in the Business Reference Model
Human Capital
• Employee Satisfaction & Quality of Worklife
• Recruitment & Retention
• Employee Development
• Employee Ratios
Technology
• Financial
• Quality & Efficiency
• Information & Data
• Reliability & Availability
• User Satisfaction
Other Fixed Assets
• Financial
• Quality, Maintenance, & Efficiency
• Security & Safety
• Utilization
INPUTS: Key enablers measured through their contribution to outputs – and by extension outcomes
Value

Figure 4.4-8. Six Measurement Areas of the PRM Capture the Relationship Among Inputs, Outputs, and Outcomes

The PRM is designed to serve three main purposes:

- Help produce enhanced IT performance information to improve strategic and daily decision making
- Improve the alignment and better articulate the contribution of IT to business outputs and outcomes, thereby creating a clear "line of sight" to desired results
- Identify performance improvement opportunities that span traditional organizational structures and boundaries

FEA PRM Example Metrics: The following are examples of FEA PRM metrics:

- Operational Measures of Effectiveness for the Processes and Activities Measurement Area
  - Financial – Achieving financial measures, direct and indirect total and per unit costs of producing products and services, and costs saved or avoided
  - Productivity & Efficiency – Amount of work accomplished per relevant units of time and resources applied
  - Cycle Time & Timeliness – Time required to produce products or services
  - Quality – Error rates and complaints related to products or services

  - Security – Extent to which security is improved
  - Management & Innovation – Management policies and procedures, compliance with applicable requirements, capabilities in risk mitigation, knowledge management, and continuous improvement

- Operational Measures of Effectiveness for the Technology Measurement Area
  - Financial – Technology-related costs and costs avoided through reducing or eliminating IT redundancies
  - Quality – Extent to which technology satisfies functionality or capability requirements or best practices, and complies with standards
  - Efficiency – System or application performance in terms of response time, interoperability, user accessibility, and improvement in technical capabilities or characteristics
  - Information & Data – Data or information sharing, standardization, reliability and quality, and storage capacity
  - Reliability & Availability – System or application capacity, availability to users, and system or application failures
  - Effectiveness – Extent to which users are satisfied with the relevant application or system, whether it meets user requirements, and its impact on the performance of the process(es) it enables and the mission results to which it contributes

Table 4.4-1 list example measurements sorted by Measurement Category and Generic Measurement Indicator Grouping.

Table 4.4-1. Example of Measurement Indicators Sorted by Measurement Category

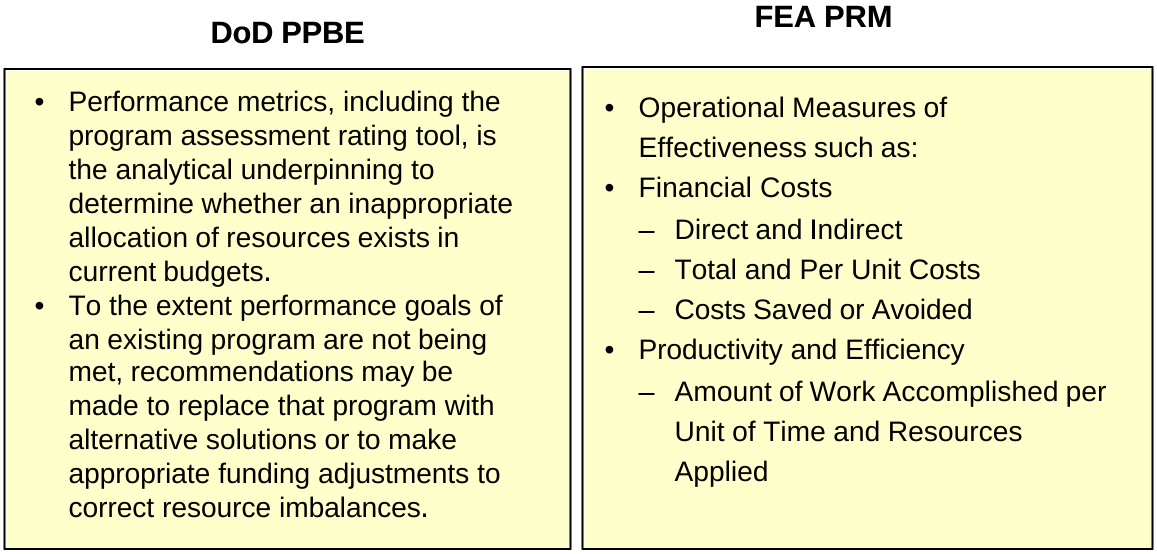
Measurement Category	Generic Measurement Indicator Grouping	Examples of "Operationalized" Measurement Indicators
ADMINISTRATIVE MANAGEMENT	Facilities, Fleet, and Equipment Management	Percent of government-owned assets with return on investment of at least 6 percent
ADMINISTRATIVE MANAGEMENT	Travel	Number of travel arrangements fully completed in the consolidated, fully integrated e-travel
HUMAN RESOURCE MANAGEMENT	Benefits Management	User/customer satisfaction

The FY05 A-11 requires agencies to use the PRM for each new Exhibit 300 they submit. The key PRM requirement is to align performance information for development, modernization, and enhancement IT investments with the PRM in Exhibit 300, Section I.C "Performance Goals and Measures." For FY05, the Performance Goals and Measures section will have two tables. **Table 4.4-2** is to be used for all development, modernization, and enhancement projects for FY05.

Table 4.4-2. PRM Alignment Table

Fiscal Year	Measurement Area	Measurement Category	Measurement Indicator	Baseline	Planned Improvements to the Baseline	Actual Results
2005						
2005						
2005						
2005						
2006						
2006						
2006						
2006						

Summary Comparison Between DoD Performance–Based Budgeting Process and FEA PRM: **Figure 4.4-9** provides a comparison of DoD's approach to performance-based budgeting (using measurements such as cost increases, schedule delays, management reform savings, workload changes, and changes resulting from congressional actions) and the similar approach that OMB has adopted to evaluate budget submissions by various agencies based on their performance (using measurements drawn from Federal agencies, Balanced Scorecard, Baldrige Criteria, and the private sector).



DoD PPBE

- Performance metrics, including the program assessment rating tool, is the analytical underpinning to determine whether an inappropriate allocation of resources exists in current budgets.
- To the extent performance goals of an existing program are not being met, recommendations may be made to replace that program with alternative solutions or to make appropriate funding adjustments to correct resource imbalances.

FEA PRM

- Operational Measures of Effectiveness such as:
- Financial Costs
  – Direct and Indirect
  – Total and Per Unit Costs
  – Costs Saved or Avoided
- Productivity and Efficiency
  – Amount of Work Accomplished per Unit of Time and Resources Applied

Figure 4.4-9. Comparison Between DoD Performance-Based Budget and FEA PRM

4.4.4 The DoD Architecture Framework and the FEA RMs

The focus of the FEA RMs is to allow OMB to compare investments, performance, and components for reuse. The focus of DoD, through the use of integrated architecture, is to relate capabilities and interoperability to systems acquisition and to support major DoD processes. Both approaches are needed, and each serves the intended audience well.

Through the DoDAF, DoD has defined a rigorous mechanism for describing the enterprise (the DoD), its components, their operational capabilities (current or future), and the systems they utilize to enable these capabilities. The relationships between DoD-specific guidance, major DoD processes, and applicable DoDAF architecture products have been addressed in Volume I and other sections in this Deskbook.

Relationships between DoDAF products and the FEA RMs include:

- The Operational Activity Model (OV-5) can portray FEA BRM's business subfunction refinements and relate the subfunctions to information flows.
- Systems View products can portray the FEA SRM's Service Components and relate the system functions or services to data flows. Specifically, the Systems Functionality Description (SV-4) and possibly the Systems Interface Description (SV-1) and Systems Communications Description (SV-2) can be used.
- Business subfunctions, the organizations responsible for them, the information flows, the service components supporting them, the systems data, and the standards are related via DoDAF products such as the Operational Activity to Systems Function Traceability Matrix (SV-5), Systems Data Exchange Matrix (SV-6), and Technical Standards Profile (TV-1).
- The DoD TRM, JTA, and a given architecture's TV-1 can be used to align to the FEA TRM and to relate standards to DoD Services and Service Areas and to TRM Service Areas and Service Categories.
- Measures of effectiveness and measures of performance documented in the following architecture products can be related to the FEA PRM: Operational Information Exchange Matrix (OV-3), Systems Data Exchange Matrix (SV-6), Systems Performance Parameters Matrix (SV-7), and Operational Activity Model (OV-5) controls; and in Operational Activity Sequence and Timing Descriptions (OV-6) rules and conditions. These products allow a quick assessment of the ability of the architecture and the systems to meet effectiveness and performance measures.
- Through the built-in relationships among the DoDAF architecture elements, the business processes, responsible organizations, associated information and systems data, IT systems, communications, and technical standards are described and inter-related. This description meets OMB's criteria and can be automatically extracted from architecture modeling tools and repositories for submission to OMB through FEAMS.

Figure 4.4-10 compares the DoD approach to architecture and the FEA RMs.

Architecture Presented in DoDAF Products

- Business functions integrated as part of process with flow
- Architecture identifies
  - Functions of service components
  - Allocation of functions to systems
  - How related to other service components, business processes, and to data
  - Standards presented in JTA
  - Standards categories defined by DoD TRM
  - Standards Applicable to Architecture presented in Technical Standards Profile
- Data related to
  - Business processes
  - Where produced and used
  - Related security issues, etc.
  - Data may be defined at entity and attribute level depending on need
- Performance – At systems levels

Reference Models

- BRM – High-level distinct business functions
- SRM – Service Components, general classes
- TRM – Standards presented in TRM
- DRM – FUTURE – UNCERTAIN – Data may be high-level categories or may include XML schemas for exchange
- PRM – Performance Measures defined in the PRM

Figure 4.4-10. Comparison Between DoDAF Architectures and Reference Models

Integrated architectures support DoD's requirements, budgeting, and acquisition processes. DoDAF-compliant architectures provide information for decision makers to support capabilities (outcomes based), prioritize requirements, and detect dependencies. The Operational View provides mechanisms for describing existing and future capabilities. The Systems View provides a vehicle for sound systems engineering that ties systems design decisions to operational needs and capabilities. DoDAF-compliant architectures can provide the basis for the analysis required to formulate DoD budget submissions to OMB.

The FEA RMs use categories that are general for the Federal Government and not based on business rules, legislation, missions, etc. that are specified for various Federal agencies and departments.

Table 4.4-3 shows a listing of the DoDAF core data elements (see taxonomy table in Volume II) and the FEA reference models. The table illustrates areas where the RMs do not describe certain aspects and core data elements that are essential for an integrated architecture. The complex relationships between data elements, which have been explicitly specified in DoDAF, are also not reflected in the FEA as the relationships between categories from one FEA RM to another FEA RM are that of simple associations. In summary, the RMs do not provide mechanisms to describe the whole enterprise by documenting the business processes, organizations responsible for them, information and systems data they need to inter-operate, IT capabilities, systems, and infrastructure, and specific technical standards that facilitate their inter-operation.

Table 4.4-3. Summary of Relationships Between FEA RMS and DoDAF Core Data Elements

FEA Reference Models	Taxonomy Types
Business Reference Model	Operational Activities and Tasks
Service Component Reference Model	System Functions
Technical Reference Model	Technical Standards *Information Processing, Information Transfer, Data, Security, and Human Factors* Technology Areas *Systems and Standards*
Performance Reference Model	Performance Parameters
Data Reference Model (Not Yet Defined)	Information Elements *Including mappings to System Data Elements*
Areas Not Part of FEA	**Taxonomy Types**
Locations	Systems Nodes *Facilities, Platforms, Units, and Locations*
Service Component Aggregations	Systems *Family of Systems, System of Systems, Networks, Applications, Software, and Equipment*
Other	Triggers/Events

4.4.5 Conclusion

OMB's objective with the FEA RMs is to enable budget analysts to quickly recognize areas of similarity across Federal agencies and departments. The ability to recognize these areas of similarity will enable OMB to make funding decisions based on reuse and collaboration across agencies, so that no duplication of IT investment takes place. Relevant points are:

- Enforcement of compliance will take place through the budget process. (Funding will be approved or denied based on compliance.)
- The BRM is intended to separate and identify similar things.
- The SRM drives the departments toward a component-based architecture and speeds arrival at an architecture solution by using e-business patterns.[4]
- The TRM specifies standard Service Areas and categories similar to the DoD JTA.

[4] Adams, J., Koushik, S., Vasudeva, G., & Galambos, G., *Patterns for e-business, A Strategy for Reuse*, First Edition, Fourth Printing, IBM Press, Double Oak, Texas, March 2003.

- The PRM is in draft release (dated July 2003) and is a major goal of OMB. OMB wants to use the BRM Business Areas and subcategories to tie outcomes to performance measurements defined in the PRM. Funding decisions will be based on outcomes.
- The DRM has not been yet been defined by OMB.

4.4.6 References

Brozen, OMB, *The Federal Enterprise Architecture: Accomplishments and Next Steps*, Presented to CISA Worldwide Conference 2003, June 9, 2003.

Defenselink, *DoD Moves to Streamline Programming and Budgeting Process*, No. 353-03, May 22, 2003, Available: http://www.defenselink.mil/news/May2003/b05222003_bt353-03.html

DoD Financial Management Regulation Web Site, http://www.dod.mil/comptroller/fmr/

DoD Comptroller iCenter, The Budget Process Web Site, http://www.dod.mil/comptroller/icenter/budget/budgetintro.htm

Federal Enterprise Architecture (FEA) Program Management Office (PMO), *The Performance Reference Model, A Standardized Approach to IT Performance*, Volume I: Version 1.0 Release Document, July 2003.

Federal Enterprise Architecture (FEA) Program Management Office (PMO) Web Site, www.feapmo.gov

Federal Enterprise Architecture (FEA) Program Management Office (PMO), *The Service Component Reference Model (SRM)*, Version 1.0, June 2003.

Federal Enterprise Architecture (FEA) Program Management Office (PMO), *Technical Reference Model (TRM)*, Version 1.0, OSD, June 2003.

Office of Management and Budget's Circulars Web Site, http://www.whitehouse.gov/omb/circulars/index.html

OMB Circular No. A-11 Web Site, http://www.whitehouse.gov/omb/circulars/a11/00toc.html

OMB, SF133 Report on Budget Execution and Budgetary Resources Web Site, http://www.whitehouse.gov/omb/reports/sf133/

4.5 UNIVERSAL REFERENCE RESOURCES

4.5.1 Introduction

A number of reference models and information standards provide guidelines and attributes that should be consulted when building architecture descriptions. Each is defined and described in its own document; however, some of the more prominent of these references are briefly described in the paragraphs that follow. This listing is not meant to be exhaustive. **Table 4.5-1** categorizes selected Universal Reference Resources (URRs).

Table 4.5-1. Universal Reference Resources

Subject	Universal Reference Resource	General Description
Missions and Military Functions	Universal Joint Task List (UJTL)	Hierarchical listing of the tasks that can be performed by a joint and multinational force
Data Environment and Standards	All-DoD Core Architecture Data Model (All-CADM)	Logical Data model of architecture data elements used to describe and build DoD architectures
	Defense Data Dictionary System (DDDS)	The primary tool to support the DoD Data Administration in developing and managing standard data per Directive 8320.1-M-1 (1998)
	SHAred Data Engineering (SHADE)	Strategy and mechanism for data sharing in the context of COE-compliant systems
Technical Implementation Criteria	Technical Reference Model (TRM)	Common Conceptual Model and vocabulary encompassing a representation of the information system domain
	Joint Technical Architecture (JTA)	IT standards and guidelines
Enterprise Capabilities and Services	Global Information Grid (GIG)	Enterprise architecture for DoD
	GIG Reference Model	Common lexicon for NCOW concepts and terminology supported by architecture descriptions
	Net-Centric Enterprise Services (NCES)	Core services available to all users of the GIG
Maturity Models and Transition Guidance	Common Operating Environment (COE)	Environment for systems development encompassing systems architecture standards, software reuse, sharable data, interoperability, and automated integration
	Levels of Information Systems Interoperability (LISI)	Reference model of interoperability levels and operational, systems, and technical architecture associations
	Intelligence Community Information System Capability Maturity Roadmap	Reference model of capability levels that facilitate an integrated plan for maturing capability across an entire domain; assists in moving from As-Is to To-Be architectures
	NATO Degrees of Interoperability	Degrees of systems interoperability and data exchange interoperability in use in NATO

4.5.2 Missions and Functions: The Universal Joint Task List (UJTL)

The CJCS Manual 3500.04C, *Universal Joint Task List (UJTL),* is a comprehensive hierarchical listing of the functional tasks that can be performed by a joint military force. The UJTL is applicable to Joint Staff, Services, combatant commands and components, activities, joint organizations, and combat support activities responsive to the CJCS. In addition to the task list, the manual provides conditions, measures, and criteria of performance.

The UJTL serves as a common language and common reference system for joint force commanders, combat support agencies, operational planners, combat developers, and trainers to communicate mission requirements. It is the basic language for development of a Joint Mission Essential Task List (JMETL) that identifies required capabilities for mission success.

The UJTL is organized into four separate parts by level of war. Each task is individually indexed to reflect its placement in the structure and coded as follows:

- Strategic level - National military tasks (prefix SN)
- Strategic level - Theater tasks (prefix ST)
- Operational level tasks (prefix OP)
- Tactical level tasks (prefix TA) include joint/interoperability tactical tasks and the applicable Service tasks

The UJTL manual includes the following:

- UJTL tasks defined at the SN, ST, OP, and TA levels
- Linkage to Service tasks (Service tasks are published separately)
- Application of UJTL tasks to JMETL/AMETL development
- Application of UJTL tasks to the development of training requirements
- Measures and criteria, and how they are used to create standards for tasks

In developing architectures that depict joint forces, the UJTL should be used to the maximum extent possible as a basis for defining joint force activities. By doing this, the UJTL can form the basis for a common activity set that can greatly facilitate integration of architectures and provide for a common understanding. However, the UJTL is not all-inclusive and does not cover all tasks accomplished within DoD. Service components are capable of tasks beyond those listed. Also, since the UJTL depicts the tasks of joint forces, the UJTL tasks generally do not lend themselves to depict the tasks of the Office of the Secretary of Defense (OSD).

The document is updated periodically and can be found at: http://www.dtic.mil/doctrine/jel/cjcsd/cjcsm/m350004c.pdf

4.5.3 Data Environment and Standards

4.5.3.1 All-DoD Core Architecture Data Model (CADM)

The *All-DoD Core Architecture Data Model* (CADM) is designed to provide a common approach for organizing and portraying the structure of architecture data. The CADM is detailed in section 4.2 of this Deskbook. CADM Version 1.0 may be found at
http://www.defenselink.mil/c3i/org/cio/i3/AWG_Digital_Library
A later version is available at http://www.aitcnet.org/dodfw/

4.5.3.2 Defense Data Dictionary System (DDDS)

The Defense Data Dictionary (DDDS) is the DoD repository containing standard data, definitions, and structure. The DDDS supports development, approval, and maintenance of metadata for DoD data standards. The current DDDS release 4.0 is a windows-style graphical user interface (GUI) with update, delete, query, print screen, and reporting capabilities. Online users manual and installation instructions are available. DISA, as the lead agency, is responsible for executing the policy and procedures and making DoD data standards available to the community.

DoD Directive 8320.1 authorizes the establishment of and assigns responsibilities for DoD data administration to plan, manage, and regulate data within DoD.[1] DISA is responsible for executing the policy and procedures and making DoD data standards available to the community.

Data administration facilitates common methods and techniques in the development and use of data standards. For data administrators, system developers, and planners to manage data as a resource, DISA provides information, products, and services. Some of them are:

- Procedures and techniques in planning, data engineering, and data quality
- Review, approval, and maintenance of data standards for the DoD community
- Technical requirements for sharing data in a common operating environment

DISA also provides a forum for functional and component data administrators to discuss projects and issues related to subjects such as data items, data migration, and data element review procedures.

DISA provides a repository for the centralized management of the DoD data standards and related information. The DDDS is the primary tool to support the DoD data administration in developing and managing standard data per Directive 8320.1. It provides a mechanism for defining metadata, cross-referencing, and consistency checking, and supports the standardization of data element names, definitions, and relationships.

Directive 8320.1 applies to all DoD component automated information systems (AISs). Several DoD component organizations are in the process of migrating their dictionary applications and data to the DDDS and the software needs to be enhanced to support their requirements. Data administration improves interoperability among AISs, facilitates data

[1] DoD Directive 8320.1, *DoD Data Administration*, September 26, 1991.

exchange, provides a means for data sharing and redundancy control, minimizes data handling, and improves data integrity. The DDDS encourages horizontal as well as vertical sharing of data in DoD with other Government agencies and the private sector.

The DDDS homepage is at http://www-datadmn.itsi.disa.mil/ddds/ddds40.html
Approved data standards are at http://www-datadmn.itsi.disa.mil/proposals/closed/apts.html
DoD Data Architecture (DDA) Views can be found at
http://www-datadmn.itsi.disa.mil/datadmn/dda/ddmhmpg.html

4.5.3.3 SHAred Data Environment (SHADE)

SHAred Data Engineering (SHADE) is a component within DoD's Common Operating Environment (COE)[2] intended to increase data interoperability for key COE systems. It includes data sharing approaches, data storage and access architectures, reusable software and data components, development guidelines, and standards for data service developers. The COE is discussed in section 4.5.6.1. SHADE's overall objective is to enable migration from many redundant, dissimilar, but overlapping, data stores to standardized COE-compliant data services built from "plug-and-play" components that blend multiple data technologies. To do this, the COE Data Engineering organization provides engineering support services for system developers and administrators that are intended to reduce barriers to interoperability, development costs, and schedules. These services include the organization and publication of existing components to encourage reuse. These services also encourage migration away from application-centric data stores to data servers built from common components and extended to meet application-specific requirements.

The COE provides guidance for transforming information systems software to be more open, portable, multi-tier, and interoperable. One of the minimal COE objectives (compliance level 5) is the separation of data from applications software to allow them to be managed and used independently. Application developers should use data related services provided by the COE rather than re-implementing them for each application. The COE uses commercial off-the-shelf (COTS) products, primarily relational database management systems (RDBMS), to provide mainline data services. SHADE emphasizes coordinated management of the sharable data structures and semantics employed within RDBMS product frameworks.

Key SHADE objectives accomplish the following:

- Leverage investments in existing databases, data structures, and data values
- Promote interoperability through their reuse
- Provide a foundation for data fusion

A prerequisite to achieving these objectives is a common representation of battlefield data. The common representation provides To-Be migration objectives, a common understanding of warrior data, plus agreement on core objects, their identifiers, and valid domain values. Additionally, it constitutes the core set of battlefield data that mission applications extend, as required. The common representation is maintained as a logical model, but it is manifested in multiple physical forms (e.g., Informix, Sybase or Oracle databases, Extensible Markup Language (XML) documents, flat files, OODBMS, etc.). The common representation is

[2] The COE was formerly known as the Defense Information Infrastructure (DII) COE. The terms are interchangeable.

being evolved by the COE Chief Engineer's Data Engineering team from the existing Command and Control (C2) Core Data Model, a subset of the Defense Data Model (DDM) and from data structures/semantics used by key C3I systems. It is being made available as COE component database segments, XML tags/metadata, reference set code values, and other forms, as required. These and other COE data products can be located via the COE's Data Emporium at http://diides.ncr.disa.mil/shade/index.cfm

The present COE supports a multi-tier architecture that also applies to database services. Developers must preserve the independence of their applications, functioning as Database Management System (DBMS) clients, from the data servers. Specifically, applications that access databases must not be built so that they have to reside on the data server in order to work correctly. It cannot be assumed that all operational sites will have a local data server. Further, where sites have a local data server, it may be on a separate machine that is dedicated to the DBMS, or the server may be collocated with the application on a single machine acting as the application server and the data server. Therefore, to maintain independence and support the client/server architecture, applications cannot assume they reside on the data server.

From the SHADE perspective, all data servers are shared assets, whether they are common or not, because they are accessed by multiple concurrent users. They are also dynamic because their data changes, even if their structure remains static. Databases within the data server may be interdependent (see **Figure 4.5-1**). Databases can be accessed by applications other than those written by the database developer. The function of data servers, regardless of the specific set of COTS DBMS and database segments, remain the same:

- Support independent, evolutionary implementation of databases and applications accessing databases
- Manage concurrent access to multiple, independent, and autonomous databases
- Maintain integrity of data stored in the data server
- Provide discretionary access to multiple databases
- Sustain client/server connections independent of the client application's and data server's hosts
- Support distribution of databases across multiple hosts with replicated data and with distributed updates
- Provide maintainability of users' access rights and permissions
- Support backup and recovery of data in the databases

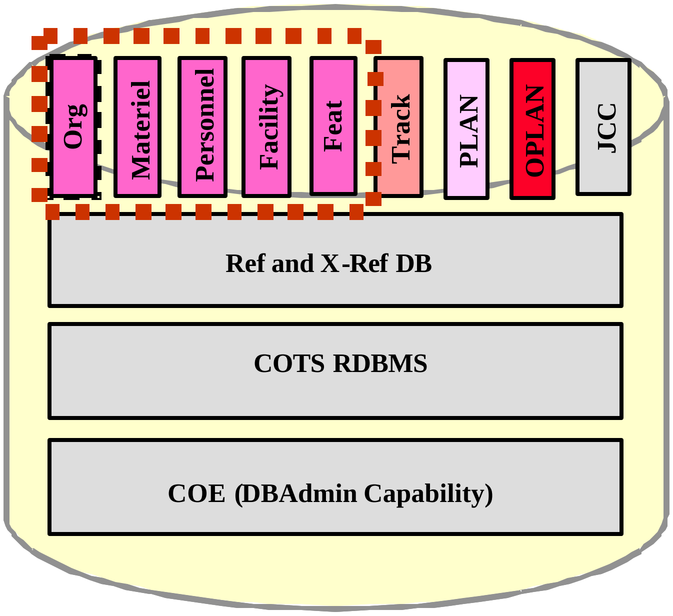


Figure 4.5-1. Shared Data Server Architecture

As part of the common data representation development, the DoD SHADE group has produced a gallery of XML tags. The tags describe DoD's common battlefield objects such as organization, materiel, personnel, and facility.

More information on the DoD XML Registry v3.1.0.1 may be found at http://diides.ncr.disa.mil/xmlreg/user/index.cfm

A briefing on the DII COE XML Effort and SHADE may be found at http://diides.ncr.disa.mil/shade/briefings/COE_XML/COE_XML_files/v3_document.htm

More information on SHADE may be found at http://diides.ncr.disa.mil/shade

4.5.4 Technical Implementation Criteria

4.5.4.1 DoD Technical Reference Model (TRM)

The DoD Technical Reference Model (TRM) provides a common conceptual model and define a common vocabulary, so that the diverse components within DoD can better coordinate acquisition, development, interoperability, and support of DoD information systems.

The DoD TRM provides guidance to developers, system architects, and individuals in using and developing systems and technical architectures. The model promotes open system design but is not a system architecture. The TRM establishes a common vocabulary and defines a set of services and interfaces common to DoD systems. The reference model provides the foundation for the organization and structure for technical architectures. The reference model and technical architecture support the operational architecture, and are the key drivers for the development of systems architecture.

The use of the DoD TRM can:

- Facilitate and enable interoperability
- Enable portability and scalability
- Support open systems concepts
- Promote product independence and software reuse
- Facilitate manageability

Figure 4.5-2 illustrates the TRM detailing the Services View and the Interface View.

Relationship to Joint Technical Architecture (JTA): The DoD TRM is the foundation of the JTA as well as of other initiatives such as the COE. The TRM is the key source of service and interface definitions that provide part of the JTA document structure and the accompanying service descriptions. Some parts of the JTA (e.g., domain annexes) prefer to use and reference the DoD TRM interface views in specifying their requirements. The ability of the DoD TRM to support different types of system requirements and different views is illustrated by the varied use of the model within the DoD community and within the JTA document.

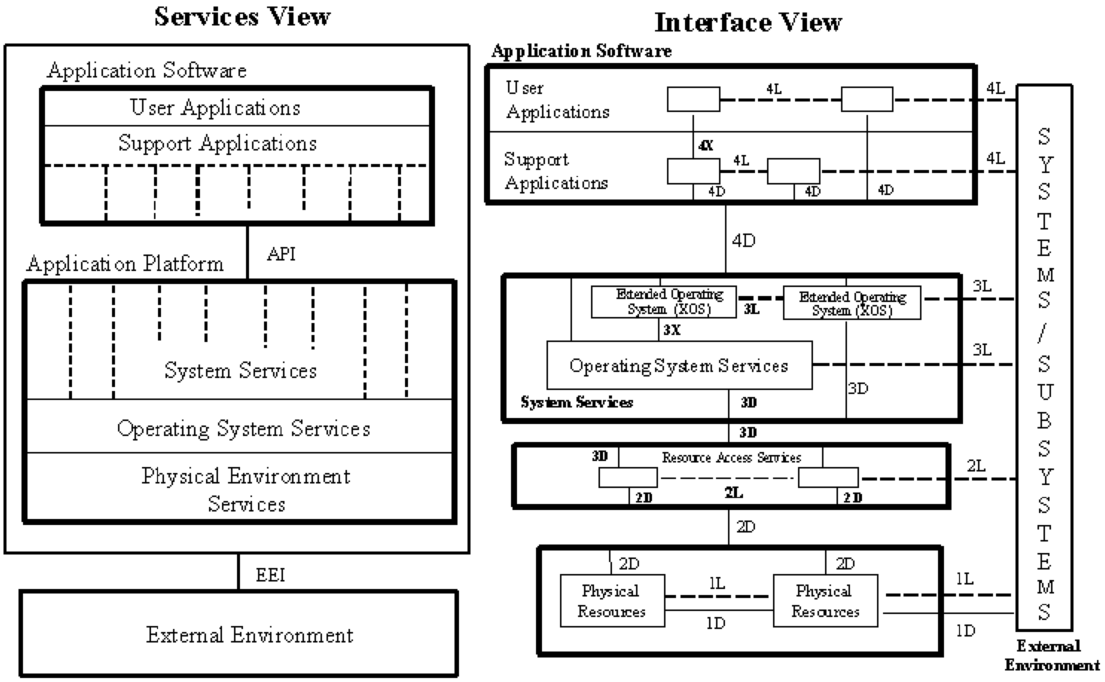


Figure 4.5-2. DoD Technical Reference Model

The DoD TRM and JTA have been expanded to accommodate real-time embedded weapons and avionics system domains, as well as modeling and simulation domains. These domains traditionally have systems or components that require carefully engineered, certifiable, real-time performance requirements. In order to keep the DoD TRM current, a number of different views within the same model are required from which a number of more specific domain-oriented representations can be derived. These representations are also capable of supporting real-time system development concerns more effectively. The Web site is at http://www-trm.itsi.disa.mil

Additional information can be obtained from the TRM document: DoD TRM, Version 2.0, dated April 9, 2001. The document may be downloaded from http://www-trm.itsi.disa.mil/dev/htdocs/trmv2_oct_18_01.pdf

A new TRM V3.0 document that includes a net-centric warfare requirement will soon be available at the TRM Web site www-trm.itsi.disa.mil

4.5.4.2 Joint Technical Architecture

The DoD JTA provides the minimum set of standards that, when implemented, facilitates the flow of information in support of the warfighter. The JTA standards promote:

- A distributed information processing environment in which applications are integrated
- Applications and data independent of hardware to achieve true integration
- Information transfer capabilities to ensure seamless communications within and across diverse media
- Information in a common format with a common meaning
- Common human computer interfaces for users and effective means to protect the information

The current JTA concept is focused on the interoperability and standardization of information technology (IT).

The JTA improves and facilitates the ability of our systems to support joint and combined operations in an overall investment strategy.

The JTA:

- Provides the foundation for interoperability among all tactical, strategic, and combat support systems
- Mandates IT standards and guidelines for DoD system development and acquisition that will facilitate interoperability in joint and coalition force operations. These standards are to be applied in concert with DoD standards reform
- Communicates to industry DoD's preference for open system, standards-based products and implementations
- Acknowledges the direction of industry's standards-based development

The JTA is considered a living document and will be updated periodically as a collaborative effort among the DoD Components (Commands, Services, and Agencies) to leverage technology advancements, standards maturity, open systems, commercial product availability, and changing requirements.

The JTA is critical to achieving the envisioned objective of a cost-effective, seamlessly integrated environment. Achieving and maintaining this vision requires interoperability in the following:

- Within a Joint Task Force (JTF)/Combatant Commander (CC) Area of Responsibility (AOR)
- Across CC AOR boundaries
- Between strategic and tactical systems
- Within and across Services and Agencies
- From the battlefield to the sustaining base
- Among U.S., Allied, and Coalition forces
- Across current and future systems

The JTA currently mandates the minimum set of standards and guidelines for the acquisition of all DoD systems that produce, use, or exchange information. The applicable mandated standards in the JTA are the starting set of standards for a system, *and additional standards may be used to meet requirements, if they are not in conflict with standards mandated in the JTA.* The JTA is used by anyone involved in the management, development, or acquisition of new or improved systems within DoD. Specific guidance for implementing the JTA is provided in separate DoD Component JTA implementation plans. Operational requirements developers are cognizant of the JTA in developing requirements and functional descriptions. System developers use the JTA to facilitate the achievements of interoperability for new and upgraded systems (and the interfaces to such systems). System integrators use it to foster the integration of existing and new systems.

The JTA is updated periodically with continued DoD Component participation. The document may be downloaded from the JTA Web page at http://www-jta.itsi.disa.mil

4.5.5 Enterprise Capabilities and Services

4.5.5.1 The Global Information Grid (GIG)

The Global Information Grid (GIG) is “the globally interconnected, end-to-end set of information capabilities, associated processes, and personnel for collecting, processing, storing, disseminating, and managing information on demand to warfighters, policy makers, and support personnel. The GIG includes all owned and leased communications and computing systems and services, software (including applications), data, security services, and other associated services necessary to achieve Information Superiority. It also includes National Security Systems (NSS), as defined in section 5142 of the Clinger-Cohen Act of 1996. The GIG supports all DoD, National Security, and related Intelligence Community missions and functions (strategic, operational, tactical, and business) in war and in peace. The GIG provides capabilities from all operating locations (bases, posts, camps, stations, facilities, mobile platforms, and deployed sites). The GIG provides interfaces to coalition, allied, and non-DoD users and systems.”[3]

Concerns about interoperability and end-to-end integration of automated information systems led to the concept of a GIG. The requirement for Information Superiority and Decision Superiority to achieve Full Spectrum Dominance, as expressed in *Joint Vision 2020,* has driven the demand for a GIG. GIG provides the enabling foundation for Net-Centric Operations and

[3] DoD Directive 8100.01, “Global Information Grid Overarching Policy,” September 19, 2002.

Warfare (NCOW), Information Superiority, Decision Superiority, and Full Spectrum Dominance.[4]

The GIG Architecture is the architectural description of the GIG. As such, it provides an integrated operational, system, and technical description of the DoD. The GIG Architecture serves as the DoD Enterprise IT Architecture required by the Clinger-Cohen Act of 1996 and OMB Circular A-130. The GIG Architecture is intended to support the DoD Chief Information Officer's decisions and recommendations concerning IT requirements, planning and programming, acquisition, and policy.[5]

GIG Architecture developers use the GIG Systems Reference Model shown in **Figure 4.5-3** as the basis for building the GIG Architecture Systems View.

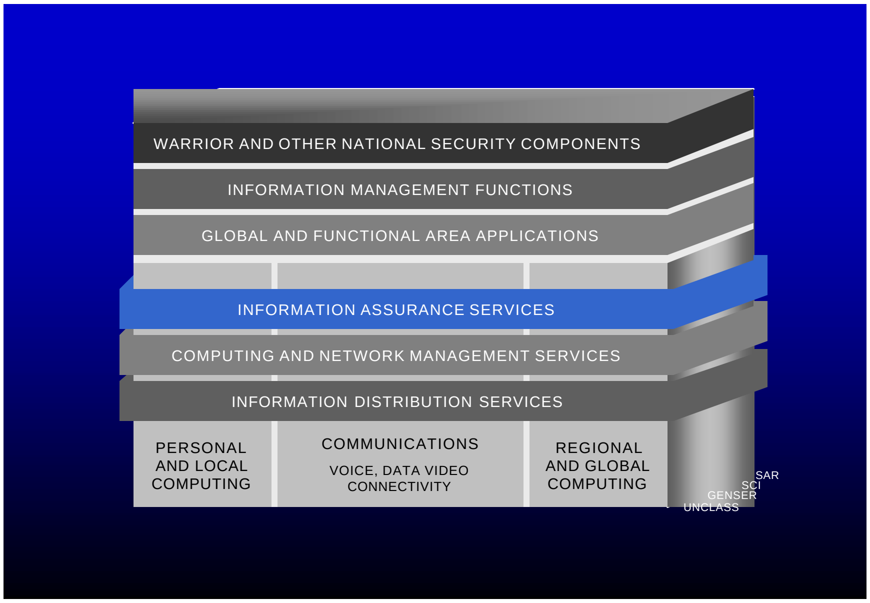

WARRIOR AND OTHER NATIONAL SECURITY COMPONENTS
INFORMATION MANAGEMENT FUNCTIONS
GLOBAL AND FUNCTIONAL AREA APPLICATIONS
INFORMATION ASSURANCE SERVICES
COMPUTING AND NETWORK MANAGEMENT SERVICES
INFORMATION DISTRIBUTION SERVICES
PERSONAL AND LOCAL COMPUTING
COMMUNICATIONS
VOICE, DATA VIDEO CONNECTIVITY
REGIONAL AND GLOBAL COMPUTING
SAR
SCI
GENSER
UNCLASS

Figure 4.5-3. GIG Systems Reference Model

GIG Architecture Version 1, August 2001, addressed the four operational concepts articulated in *Joint Vision 2010*—Dominant Maneuver, Precision Engagement, Full Dimensional Protection, and Focused Logistics. GIG Version 1 describes, for a limited scenario, how a Joint Task Force (JTF) would conduct a specific notional operation. The architecture defines the activities and associated information exchanges required to complete the operation and the system capabilities that support such an operation. In addition, it addresses the activities, information exchanges, and system capabilities of selected Principal Staff Assistants when supporting the given JTF. To the extent possible, the Operational and Systems Views were built by integrating existing architecture products. Architecture segments representative of seven Joint Mission Areas and five PSAs were included.

GIG Architecture Version 2 is being developed as an objective architecture focusing on the enterprise aspects of Net-Centric Operations and Warfare (NCOW). As an objective architecture, it describes what "should be" to conduct operations at multiple levels. A GIG

[4] *GIG Capstone Requirements Document*, JROCM 1314-01, 30 August 2001, p. 1.

[5] GIG Architecture Master Plan, draft, 4 September 2002, p.v.

supporting NCOW can be understood in terms of capabilities and services available across the grid. Enterprise capabilities, services, and example architecture products from GIG v2 are provided in section 2.7 of this Deskbook.

A DoD capability or program architecture must be compliant with the GIG Architecture. The following is required for GIG-compliance:

- Architecture products comply with product definitions in the DoDAF
- Architecture data is provided in database form in conformance to the CADM
- The architecture derives all IT/NSS standards from the DoD JTA or presents the case for new or unique standards as necessary
- The architecture conforms to the NCOW Reference Model (RM)

4.5.5.2 NCOW Reference Model

The goal of the NCOW RM is to provide a common lexicon for NCOW concepts and terminology supported by architectural descriptions. The NCOW RM is based on the premise that a net-centric information environment is a business-neutral, common feature of all GIG Architecture use cases. The NCOW RM is being developed as a DoD community reference model under the leadership of Assistant Secretary of Defense for Networks and Information Integration (ASD[NII])/DoD CIO. As this Deskbook was finalized, Version 0.9 of the NCOW RM had been provided for community review and coordination. Version 0.9 contains Overview and Summary Information (AV-1), Integrated Dictionary (AV-2), High-Level Operational Concept Graphic (OV-1), Operational Activity Model (OV-5), and Technical Standards View (TV). The OV-1 is provided in **Figure 4.5-4**. The OV-5 is discussed in Section 2.7: An Architecture Perspective on NCOW.

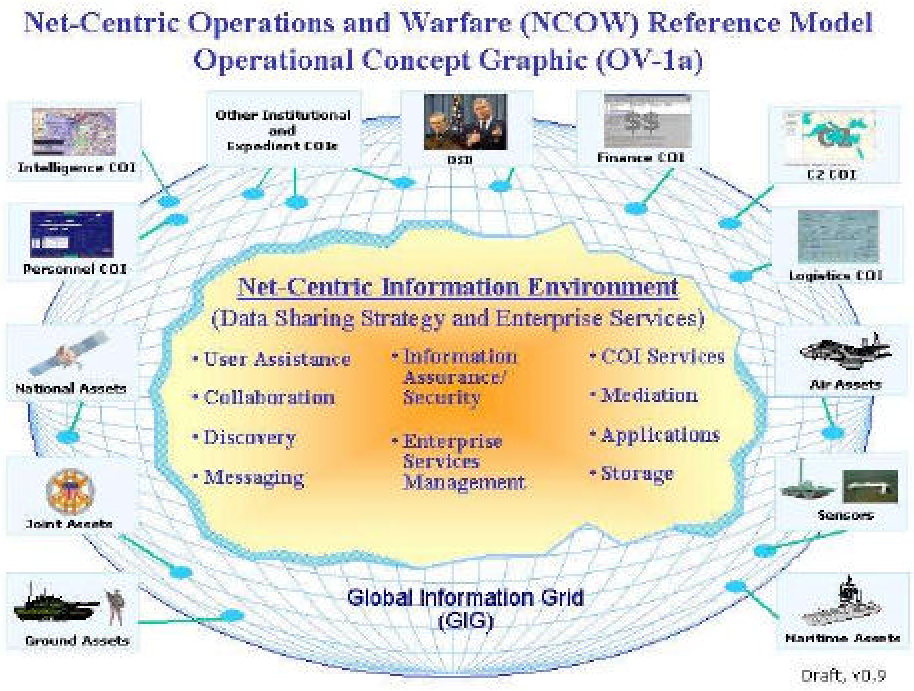

Net-Centric Operations and Warfare (NCOW) Reference Model
Operational Concept Graphic (OV-1a)
Intelligence COI
Other Institutional and Expedient COIs
OSD
Finance COI
C2 COI
Personnel COI
Logistics COI
Net-Centric Information Environment
(Data Sharing Strategy and Enterprise Services)
• User Assistance
• Collaboration
• Discovery
• Messaging
• Information Assurance/ Security
• Enterprise Services Management
• COI Services
• Mediation
• Applications
• Storage
National Assets
Air Assets
Joint Assets
Sensors
Ground Assets
Maritime Assets
Global Information Grid (GIG)
Draft, v0.9

Figure 4.5-4. NCOW RM High-Level Concept Graphic (OV-1)

The NCOW Target Technical Standards View provides a description of 16 emerging protocols and 5 advanced technologies needed to achieve DoD's goals of net-centricity. **Figure 4.5-5** identifies 17 emerging standards that are included in the NCOW Target Technical Standards View.

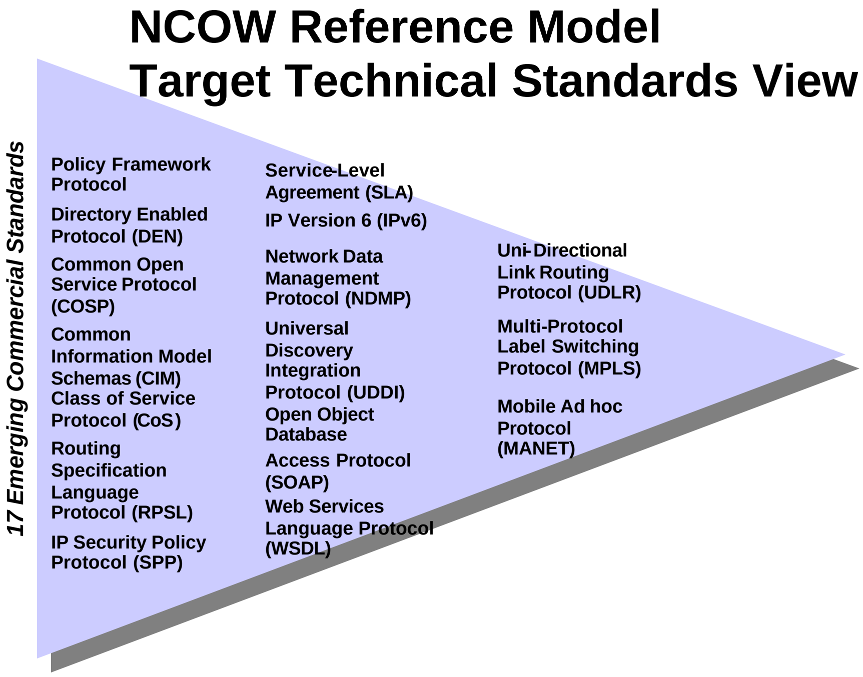

NCOW Reference Model
Target Technical Standards View
17 Emerging Commercial Standards
Policy Framework Protocol
Directory Enabled Protocol (DEN)
Common Open Service Protocol (COSP)
Common Information Model Schemas (CIM)
Class of Service Protocol (CoS)
Routing Specification Language Protocol (RPSL)
IP Security Policy Protocol (SPP)
Service-Level Agreement (SLA)
IP Version 6 (IPv6)
Network Data Management Protocol (NDMP)
Universal Discovery Integration Protocol (UDDI)
Open Object Database Access Protocol (SOAP)
Web Services Language Protocol (WSDL)
Uni-Directional Link Routing Protocol (UDLR)
Multi-Protocol Label Switching Protocol (MPLS)
Mobile Ad hoc Protocol (MANET)

Figure 4.5-5. NCOW RM Emerging Commercial Standards

Compliance with the NCOW RM is one criteria for GIG compliance. An architecture portraying net-centric concepts must:

- Use the definitions and vocabulary from the NCOW RM AV-2
- Show how the capabilities and services defined in the NCOW RM OV-5 are instantiated and utilized by the materiel solution
- Incorporate the IT/NSS standards from NCOW RM Technical Standards View in the TV products developed for the materiel solution

4.5.5.3 Net-Centric Enterprise Services (NCES)

Net-Centric Enterprise Services (NCES) establishes the foundation for operating in a net-centric environment. NCES defines common services that will be available across the GIG and will enable other programs to operate in a net-centric environment. The intent is to provide all DoD users with improved access to decision quality information by:

- Providing new capabilities such as enterprise data discovery and collaboration
- Providing robust security and management of netted information resources (i.e., data posted to shared vs. private space)
- Changing cultures (i.e., publish before process)

GIG Enterprise Services address the GIG Architecture requirements for common capabilities to (1) task, post, process, use, store, manage, and protect information resources *on demand* for warriors, policy makers, and support personnel and (2) facilitate information sharing across systems. As depicted in **Figure 4.5-6**, GIG Enterprise Services encompasses NCES and also includes domain- and community of interest (COI)-specific services. Domain and COI services are specific to a user community such as logistics or intelligence.

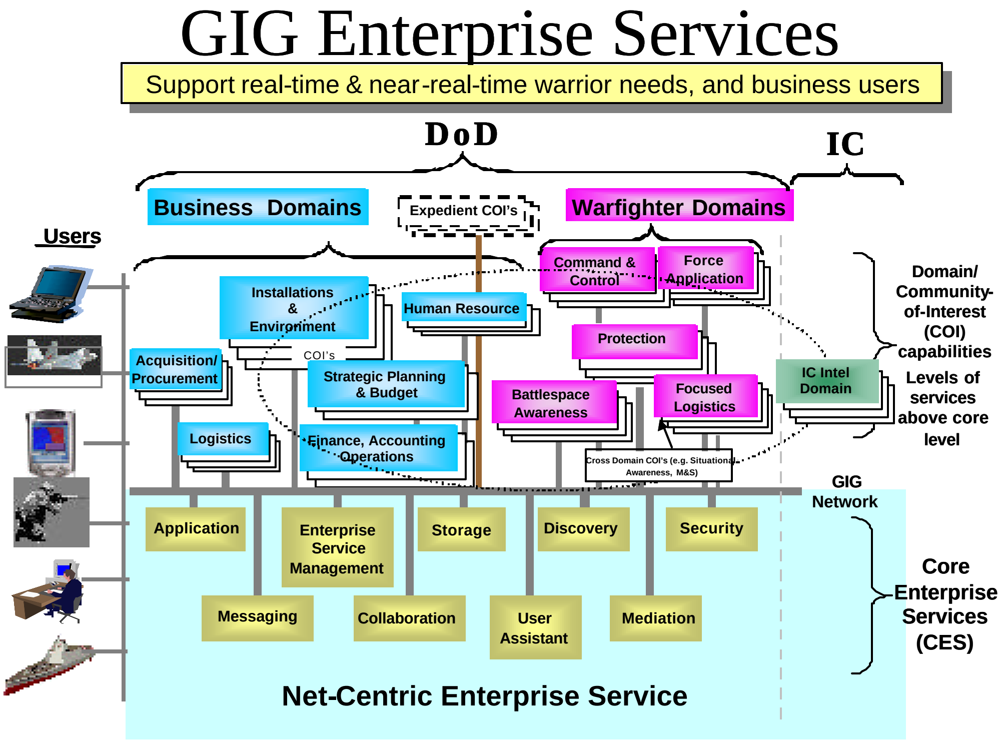

GIG Enterprise Services
Support real-time & near-real-time warrior needs, and business users
DoD
IC
Business Domains
Expedient COI's
Warfighter Domains
Users
Installations & Environment
Human Resource
Acquisition/ Procurement
COI's
Strategic Planning & Budget
Logistics
Finance, Accounting Operations
Command & Control
Force Application
Protection
Battlespace Awareness
Focused Logistics
Cross Domain COI's (e.g. Situational Awareness, M&S)
IC Intel Domain
Domain/ Community-of-Interest (COI) capabilities
Levels of services above core level
GIG Network
Application
Enterprise Service Management
Storage
Discovery
Security
Messaging
Collaboration
User Assistant
Mediation
Core Enterprise Services (CES)
Net-Centric Enterprise Service

Figure 4.5-6. NCES are Core Services within the GIG

NCES is composed of nine core services that would be available to all users of the GIG. These core services are:

- **Enterprise Service Management** – Provides end-to-end GIG performance monitoring, configuration management, and problem detection/resolution as well as enterprise IT resource accounting and addressing (e.g., for users, systems, devices)
- **Messaging Services** – Ability to exchange information among users or applications on the enterprise infrastructure (e.g., E-mail, Defense Message Service, Variable Message Format, U.S. Message Text Format, Tactical Digital Information Link, Message Oriented Middleware, America-On-Line instant messenger, Wireless Services, Alert Services)
- **Discovery Services** – Processes for discovery of information content or services that exploit metadata descriptions of network resources stored in Directories, Registries, and Catalogs (includes search engines)
- **Mediation Services** – Services that help broker, translate, aggregate, fuse, or integrate data

- **Collaboration Services –** Allows users to work together and jointly use selected capabilities on the network (i.e., chat, online meetings, work group software, etc.)
- **User Assistant Services –** Automated "helper" capabilities that reduce effort required to perform manpower intensive tasks
- **Application Services –** Infrastructure to host and organize distributed online processing capabilities
- **Security Services –** Capabilities that addresses vulnerabilities in networks, services, capabilities, or systems
- **Storage Services –** Physical and virtual places to host data on the network with varying degrees of persistence (e.g., archiving, Continuity of Operations, content staging)

4.5.6 Maturity Models and Transition Guidance

4.5.6.1 Common Operating Environment (COE)

The Common Operating Environment (COE)[6] provides a framework for developing and fielding DoD systems that meet the needs of the warfighter in a global information environment. As indicated in the Command, Control, Communications, Computers, and Intelligence (C4I) for the Warrior concept, "the warrior needs a fused, real-time true-picture of the battlespace and the ability to order, respond, and coordinate vertically and horizontally to the degree necessary to prosecute the mission in that battlespace." DoD relies on the COE to provide the degree of system integration and interoperability required to achieve this vision.

The COE addresses systems in the C4I and combat support domains within DoD. The C4I domain includes systems that facilitate the command and control of forces by the tactical commander, while the combat support domain includes systems that support logistics, transportation, base support, personnel, and health affairs functions. The Global Command and Control System (GCCS) and the Global Combat Support System (GCSS) are examples of C4I and combat support systems, respectively, that are based on the COE and support the joint warfighter.

The COE provides a client-server architecture for developing reusable, interoperable software from which systems tailored to the specific needs of a user community can be built. A COE-based system is composed of software components, called segments, contributed by different sources and maintained in a segment repository. Some segments are part of the COE, because they perform common functions required by most systems, while other segments perform mission-specific functions that are targeted to particular operational communities. Software is included in the segment repository only if it conforms to strict standards and specifications that are required to support "plug and play" integration across a range of hardware platforms.

It is critical to the overall usability of a system that the software in the segment repository provide a user interface with a common appearance and behavior, so users can interact

[6] The concept of the Defense Information Infrastructure (DII) has been superceded by that of the Global Information Grid, and what was previously known as the DII COE is now referred to as the Common Operating Environment. As a result, references to the term "DII" have been removed from both the title and content of this release of the document.

effectively with any system built from this software. User interface standardization is particularly important, as users are provided the capability to interact with a variety of complex, multi-windowed applications within a single system. The benefits to be gained from standardization are increased user productivity, reduced training requirements, improved system reliability, reduced maintenance costs, and increased efficiency in the development of individual applications as well as entire systems.

The purpose of the COE is to ensure that software developed for the COE exhibits commonality in "look and feel," because commonality is a key element of usability as well as a requirement of the runtime environment defined by the COE. Compliance with COE style specifications is mandated for all software in the segment repository because the specifications define the "rules" for a well-behaved application[7] to operate predictably in a standard runtime environment. Compliance is especially important, since the applications in a system can be built from multiple segments, each produced by a different organization.

A common look and feel provides consistency in the appearance and behavior of user interface objects while allowing flexibility for addressing operational requirements. Implementing a common look and feel enables users to identify, remember, and predict the rules and organization of a system. By building consistency in the user interface, users can develop an effective and efficient model of how the system works and can generalize this knowledge to other systems.

Further details on COE are available at http://diicoe.disa.mil/coe/.

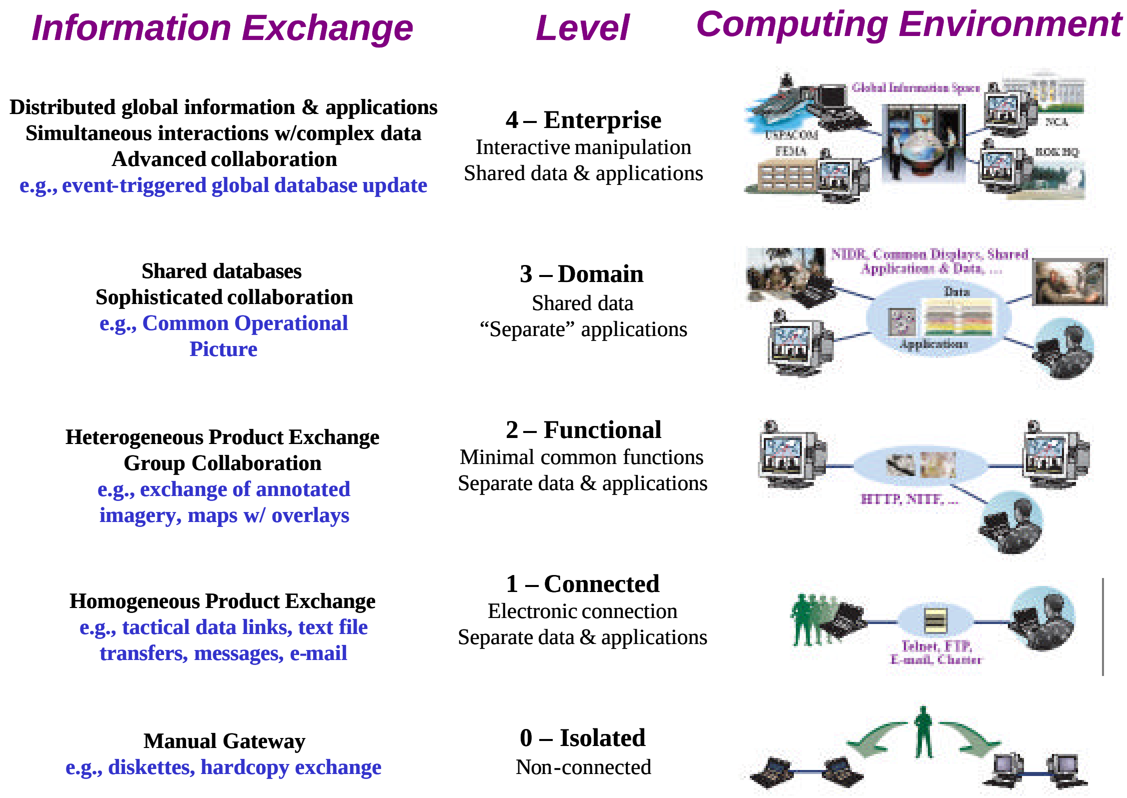
4.5.6.2 Levels of Information Systems Interoperability (LISI)

DoD and its component organizations place a premium on the ability to access, manipulate, and exchange information between multiple disparate systems. The quality that describes how information systems can exchange information and services is generally referred to as interoperability.

The purpose of LISI is to provide DoD with a maturity model and a process for determining joint interoperability needs, assessing the ability of information systems to meet those needs, and selecting pragmatic solutions and a transition path for achieving higher states of capability and interoperability. LISI is a process for defining, evaluating, measuring, and assessing information systems interoperability. LISI uses a common frame of reference and measure of performance.

LISI is organized into maturity levels that represent increasingly sophisticated user capabilities and the associated computing environments that support them. See **Figure 4.5-7**. Within each of these maturity levels, however, many additional factors influence the ability of information systems to interoperate. LISI categorizes these factors into four key attributes—*Procedures*, *Applications*, *Infrastructures*, and *Data.* These attributes, collectively referred to as ***PAID***, are broad enough by definition to encompass the full range of interoperability considerations.

[7] In this document, the term "application" is used to refer to a user application, i.e., the software with which users interact to perform one or more related operational tasks. In the COE, the tasks in an application can be performed by software taken from different sources. As a result, an application may contain one or more segments, and a single segment may be present in one or more applications.



Information Exchange	Level	Computing Environment
Distributed global information & applications **Simultaneous interactions w/complex data** **Advanced collaboration** **e.g., event-triggered global database update**	**4 – Enterprise** Interactive manipulation Shared data & applications	
Shared databases **Sophisticated collaboration** **e.g., Common Operational Picture**	**3 – Domain** Shared data "Separate" applications	
Heterogeneous Product Exchange **Group Collaboration** **e.g., exchange of annotated imagery, maps w/ overlays**	**2 – Functional** Minimal common functions Separate data & applications	
Homogeneous Product Exchange **e.g., tactical data links, text file transfers, messages, e-mail**	**1 – Connected** Electronic connection Separate data & applications	
Manual Gateway **e.g., diskettes, hardcopy exchange**	**0 – Isolated** Non-connected	

Figure 4.5-7. LISI Levels of Interoperability Sophistication

LISI addresses increasing levels of sophistication regarding the ability of systems to exchange information with each other. In this respect, LISI has a systems and technical focus. Although LISI does define each level in terms that serve to characterize the nature of the information needlines captured in an architecture's Operational View, LISI does not address "operational" interoperability in terms of Joint warfighting levels of interoperation. The reader is referred to ongoing work by the National Security Space Architect and its *Mission Information Management Initial Report*[8] that addresses operational interoperability.

Interoperability Assessments of Information Technology Architectures: It is important to have a common understanding of architectures and the interoperability aspects of those architectures. The Framework establishes a common way to represent architectures. LISI provides a common way to measure and represent system interoperability. By incorporating this LISI representation into the architecture process, an understanding of the interoperability aspects of architectures is enabled. LISI defines a process for identifying interoperability problems, gaps, and shortfalls within any information technology architecture, as well as for assessing and reporting discrete interoperability performance measures as required by Federal Government legislation.

Portraying LISI Metrics as Architecture Overlays: The set of nodes or entities involved (organizations and systems) in a mission operation or business process are defined and described with respect to their valid information exchange requirements. These entities and their relationships are then captured in some form of architecture product (e.g., the Operational Node

[8] Mission Information Management Initial Report, National Security Space Architect, 1999.

Connectivity Description [OV-2] discussed above). The architecture's System Interface Description (SV-1), in accordance with the *LISI Capabilities Model* and ***PAID*** attributes, then identifies the existing or postulated information systems and their capabilities and implementations earmarked for supporting the requirements. LISI is then employed to derive each system's *generic* level of interoperability and to commence the architecture assessment process.

After assigning generic levels, the expected levels of interoperability are determined for each system pair at both ends of each architecture needline. The expected level represents the generic level of both systems if they are equal or the lower generic level of the two systems if they are not equal. The implementation options of both systems are then examined and compared to determine the specific level of interoperability between each system pair. **Table 4.5-2** describes how LISI can be used to relate to DoDAF products.

Table 4.5-2. Contributions of LISI to Selected C4ISR Architecture Framework Products

Applicable View	Product Reference	Framework Product	LISI Contributions
Operational	OV-2	Operational Node Connectivity Description	LISI provides the interoperability maturity model and definitions in accordance with the fundamental nature of information exchanges, including the levels metric, for identifying level required for each information needline
Operational	OV-3	Operational Information Exchange Matrix	The information used by LISI to determine the "data" attribute of *PAID* provides for the creation of the "Potential Input/Output Matrix" for registered systems. This LISI product, initially derived in system-to-system format, easily rolls up to the operational node-to-node representation for this view.
Systems	SV-1	Systems Interface Description	LISI defines the prescribed *PAID* capabilities that must be accommodated by systems on both ends of each needline identified in OV-2. Establishes the basis for individual and pair-wise systems interoperability assessments.
Systems	SV-2	Systems Communications Description	The *PAID Infrastructure* attribute of LISI captures key capabilities and implementation choices of the registered systems to include the form and type of communication exchange needed to satisfy each needline. Mapping the "level" of interoperability to each system-to-system link can assist in early identification of needs and gaps during the architecture analysis process.
Systems	SV-3	Systems-Systems Matrix	When this matrix is used to focus on system-to-system interoperability relationships – current and postulated – all aspects of LISI can be used to construct and assess this architecture product, to any degree of depth (level required, capabilities needed, implementations, and improvement strategies).
Systems	SV-6	Systems Data Exchange Matrix	All four attributes of *PAID* are integral to the preparation of this product. LISI's "Potential Input/Output Matrix," "Interconnection Requirements Matrix," et al., all contribute to the development of this product. This product also maps into OV-3 as a result of summing the matrix information across systems at each node.
Systems	SV-8	Systems Evolution Description	The *LISI Maturity Model* and related capabilities and options vehicles combine to facilitate the development of an evolutionary path for achieving higher states of interoperability over time (for a system or suite of systems).

Applicable View	Product Reference	Framework Product	LISI Contributions
Systems	SV-9	Systems Technology Forecast	LISI contributes to this product by providing information about what choices developers are making and what options are emerging from industry. As more and more developers include what was "leading-edge" technology from prior forecasts, LISI provides insight into how these technologies translate into viable implementation choices. LISI also captures the implementation choices that have been selected or programmed that may not have been listed previously—this aids in updating forecasts by drawing attention to these activities.
Technical	TV-1	Technical Standards Profile	LISI relates the appropriate prevailing standards to the specific *PAID* capabilities that the *LISI Capabilities Model* prescribes for the interoperability maturity level to be achieved, thereby creating the interoperability technical architecture profile for any system and/or enterprise.

InspeQtor: InspeQtor 1.0 is a Web-based tool for capturing, manipulating, and analyzing IT system characteristics in context with any x-y coordinate-based reference model (e.g., LISI Capabilities Model, COE Runtime Environment Compliance Levels, International Standards Organization [ISO], OSI Protocol Stacks).

InspeQtor receives inputs via a system survey questionnaire. Users register system characteristics by selecting the appropriate responses to the questions. Answers are stored in a table from which data can be used to create a set of reports. InspeQtor generates reports that reflect the information captured in the surveys. Reports are available to describe single systems and support comparisons between multiple systems.

Levels of Information System Interoperability, dated March 30, 1998, is available at http://www.defenselink.mil/c3i/org/cio/i3/lisirpt.pdf

4.5.6.3 Intelligence Community Information Systems (IC IS) Capability Roadmap

The IC Chief Information Officer (CIO) and the IC CIO Executive Council have endorsed the *Intelligence Community (IC) Information Systems (IS) Capability Roadmap*. The IC is utilizing the *Roadmap* for prioritizing IT requirements and developing investment strategies for maturing enterprise IT capabilities, guiding resource and policy decisions, coordinating executive agent activities, and tracking progress. The *Roadmap* is potentially useful for the rest of DoD as well. The *Roadmap* is referenced in the *Strategic Direction for Intelligence Community Information Systems*, published by the IC CIO, April 2000. The complete *Roadmap* documentation, in briefing format, is available through the IC CIO.

IC IS Roadmap Components – The *Roadmap* is structured into 16 components that constitute the IT capabilities of the IC enterprise. The 16 components are partitioned into the 3 categories of Process, Knowledge Management, and Infrastructure.

The "Process" category includes the *Roadmap* components that are concerned with enterprise IT governance, resource provisioning, training, and fielding. The "Knowledge Management" category includes the *Roadmap* components that are responsible for building and evolving the shared information space of the enterprise, and for equipping IC users and mission customers with the means of expediently acquiring the information they need. The "Infrastructure" category includes the backbone components of the enterprise that support the Knowledge Management capabilities and services.

There are many ways that an enterprise's IT capability can be segmented. However, the *Roadmap's* structure is the component breakout that resulted from many iterations with community CIOs, was readily accepted, and has stood the test of time.

The *IC IS Capability Roadmap* can be easily adapted to any domain of interest (e.g., the component entitled "Intelligence Applications" can be modified to reflect the mission applications of any domain of interest, such as logistics, C4ISR, or virtually any Government or commercial industry domain). However, any modification of the *Roadmap*'s component labels or text that may be necessary to convert the *Roadmap* to DoD or other Federal Government domains should not require any fundamental changes to the basic functional capability described for each level of any component. **Figure 4.5-8** presents the structure of Roadmap components.

IC IS Roadmap Standard Capability Maturity Scale – A standard capability maturity scale was employed to describe the maturity model for each of the 16 components of the *Roadmap*. The standard scale itself is a generic adaptation from the *Software Engineering Institute's Capability Maturity Model.* The standard scale addresses five generic levels of increasing capability. The levels progress from an "ad hoc" state (wherein each organization of the enterprise acts autonomously), to an "optimizing" state (wherein all member organizations of the enterprise—and the global partners of the enterprise—experience the benefits of interoperability and resource sharing).

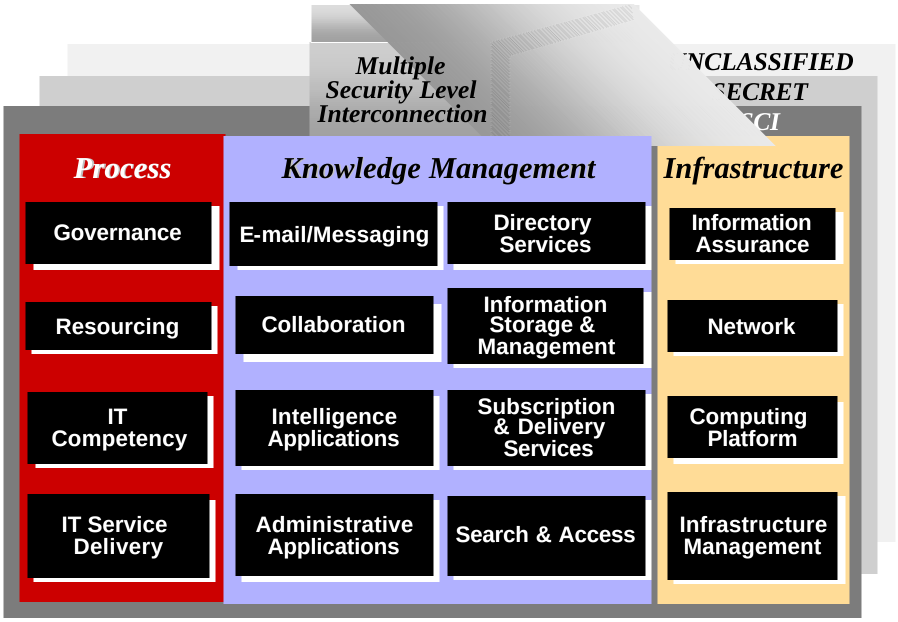


Figure 4.5-8. Components (Focus Areas) of the *IC IS Capability Roadmap*

There are two main dimensions of enterprise capability that change as a *Roadmap* component progresses from Level 1 through Level 5: breadth of "outreach" or global participation, and sophistication of capability. Increase in outreach is fundamentally enabled by cultural and policy changes. Sophistication of capability is heavily influenced by technology evolution and cultural assimilation.

Figure 4.5-9 defines the *Roadmap's* standard capability maturity scale that governs each of the 16 component models. The term "IC" can be replaced by the name of any domain of interest.

Sequence of Information Provided for Each *Roadmap* Component – Each of the 16 components of the *Roadmap* is presented via a sequence of five descriptive charts.

The first chart of the sequence provides a definition of the scope of the *Roadmap* component to ensure common understanding across the enterprise.

The next two charts of the sequence provide two perspectives of the component's capability maturity model in conformance with the standard scale. The first perspective provides an executive-level view, with "traffic lights" depicting where the enterprise currently stands. The second perspective provides a more detailed description of the component's capability maturity levels in terms that translate more directly into functional specifications.

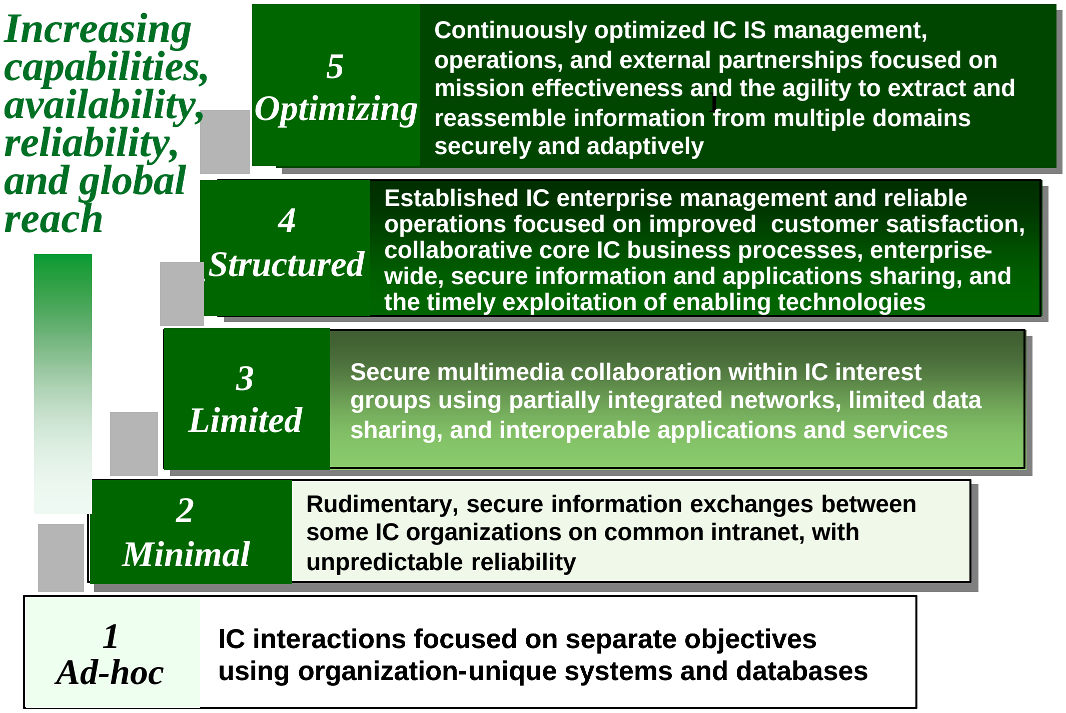

Increasing capabilities, availability, reliability, and global reach
5 Optimizing
Continuously optimized IC IS management, operations, and external partnerships focused on mission effectiveness and the agility to extract and reassemble information from multiple domains securely and adaptively
4 Structured
Established IC enterprise management and reliable operations focused on improved customer satisfaction, collaborative core IC business processes, enterprise-wide, secure information and applications sharing, and the timely exploitation of enabling technologies
3 Limited
Secure multimedia collaboration within IC interest groups using partially integrated networks, limited data sharing, and interoperable applications and services
2 Minimal
Rudimentary, secure information exchanges between some IC organizations on common intranet, with unpredictable reliability
1 Ad-hoc
IC interactions focused on separate objectives using organization-unique systems and databases

Figure 4.5-9. The Standard Capability Maturity Scale

The two charts describing each component's capability maturity levels are focused on increasing functional capability and do not prescribe specific technologies or IT solutions. Because of this intentional divorce from system solutions, the *Roadmap*'s capability maturity models should not change dynamically over time, a fact that is important for long-range evolutionary planning.

The fourth and fifth charts of each of the *Roadmap* component's descriptive sequence are time-sensitive and will change as new technology becomes available and/or as new programs surface.

The fourth chart of each *Roadmap* component's descriptive sequence provides an initial identification of the major policies (including standards) that must be defined or reviewed and modified to permit each of the five capability maturity levels to be achieved. This chart also identifies any known off-the-shelf and emerging technologies to be regarded by the selected systems engineer or executive agent as possible solutions for enabling each of the capability levels to be achieved.

The final chart of each *Roadmap* component's descriptive sequence identifies funded programs that are developing capabilities related to those capabilities targeted for each level of the *Roadmap* component's capability maturity model. The purpose of this information is to identify potential enterprise partners or executive agents who might be provided incentives for extending or adapting their plans to meet the needs of the enterprise.

Figure 4.5-10 summarizes the major elements of the *IC IS Capability Roadmap* that are provided in sequence for each of the *Roadmap's* 16 components in order to facilitate enterprise decision making.

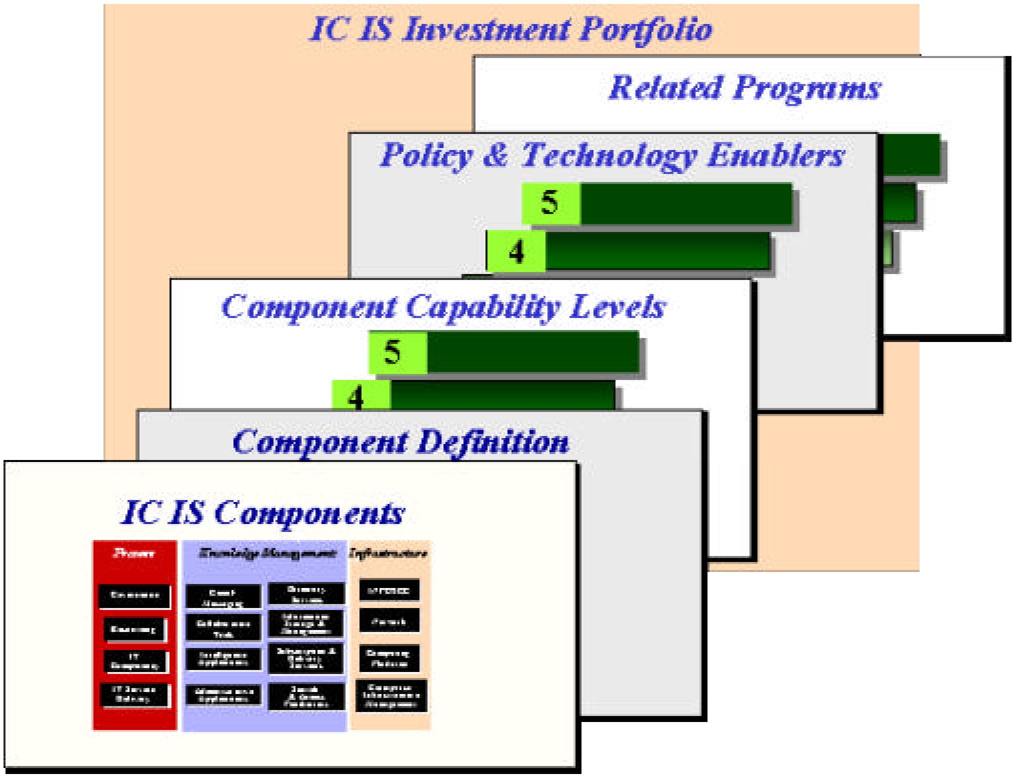


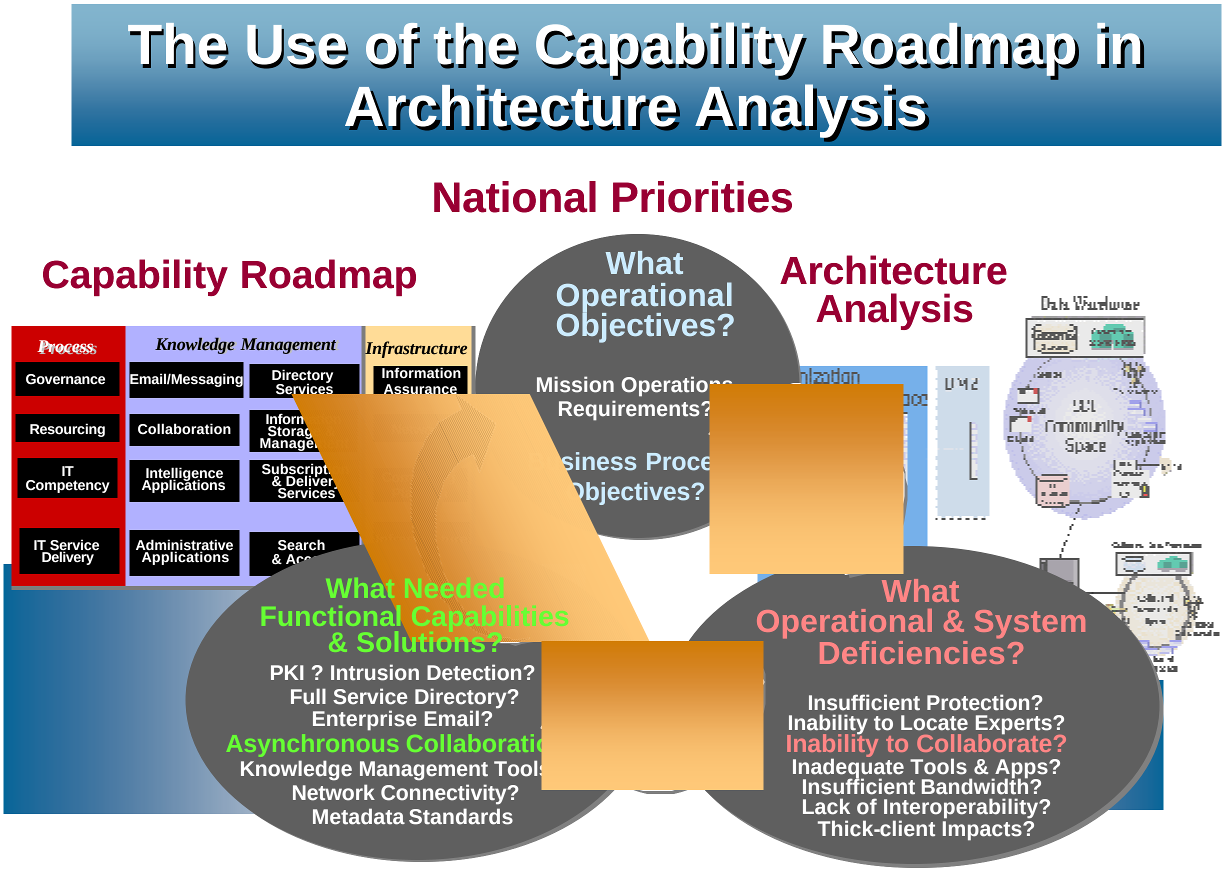
Figure 4.5-10. Details Provided for Each Component of the *IC IS Capability Roadmap*

By utilizing a roadmap structure and the information provided, an architect can derive a tailored Capability Maturity Profile of the enterprise roadmap to:

- Depict the scope of analysis and/or acquisition responsibility
- Identify the systems baseline in terms of specific IT-enabled capability levels
- Assess baseline shortfalls in context with the requirements captured in the Operational, Systems, and Technical Standards Views of the architecture of interest, and
- Identify the relevant component(s) and target level(s) of the *Roadmap* that represent the focus for To-Be improvements.

The additional supporting roadmap information (policies, technology enablers, and related programs) can be used to examine solution options and to flesh out cost and risk implications that are necessary inputs to an investment strategy.

To understand the utility of the *Capability Roadmap* in architecture analysis, one should view the *Roadmap* as a bridge that helps translate operational needs into system specifications, and subsequently, investment options. Most operational deficiencies that are assessed by comparing an architecture's Operational and Systems Views are explicitly related to functional inadequacies. For example, an assessment is likely to identify areas where there is an inability to perform certain required functions *at all*, or the inability to perform certain critical functions *adequately*. Because the *Roadmap* addresses the *functional* aspects of IT-enabled capability, including levels of sophistication, the *Roadmap* inherently provides a categorical and systematic reference model for profiling operational needs in terms that readily translate to system and policy solutions. **Figure 4.5-11** illustrates these relationships.



The Use of the Capability Roadmap in Architecture Analysis

National Priorities

Capability Roadmap

Architecture Analysis

What Operational Objectives?

Mission Operations Requirements?

Business Process Objectives?

What Needed Functional Capabilities & Solutions?

PKI ? Intrusion Detection?
Full Service Directory?
Enterprise Email?
Asynchronous Collaboration
Knowledge Management Tools
Network Connectivity?
Metadata Standards

What Operational & System Deficiencies?

Insufficient Protection?
Inability to Locate Experts?
Inability to Collaborate?
Inadequate Tools & Apps?
Insufficient Bandwidth?
Lack of Interoperability?
Thick-client Impacts?

Figure 4.5-11. Using the Roadmap in Architecture Analysis to Link Investments to Mission Effectiveness

4.5.6.4 North Atlantic Treaty Organization (NATO) Degrees of Interoperability

The NATO definition of interoperability is "the ability of systems, units, or forces to provide services to, and accept services from, other systems, units, or forces, and to use the services so exchanged to enable them to operate effectively together." The degrees of interoperability are intended to classify how structuring and automating the exchange and interpretation of data can enhance operational effectiveness. NATO defines five degrees of Consultation, Command and Control (C3) systems interoperability (Degree 0 through Degree 4) and four degrees of data exchange interoperability (NATO Degree 1 through NATO Degree 4).

NATO defines C3 Systems Interoperability in the *NATO C3 Systems Interoperability Directive (NID)*[9]. The degrees of C3 systems interoperability mandated in NATO are defined below:

- **Degree 0**: **Isolated Interoperability in a Manual Environment**. The key feature of Level 0 is human intervention to provide interoperability where systems are isolated from each other.
- **Degree 1**: **Connected Interoperability in a Peer-to-Peer Environment**. The key feature of Degree 1 is physical connectivity providing direct interaction between systems.
- **Degree 2**: **Functional Interoperability in a Distributed Environment.** The key feature of Degree 2 is the ability of independent applications to exchange and use independent data components in a direct or distributed manner among systems.
- **Degree 3**: **Domain Interoperability in an Integrated Environment.** The key feature of Degree 3 is a domain perspective that includes domain data models and procedures where data is shared among the independent applications, which may begin to work together in an integrated fashion.
- **Degree 4**: **Enterprise Interoperability in a Universal Environment**. The key feature of Degree 4 is a top-level perspective that includes enterprise data models and procedures, where data is seamlessly shared among the applications that work together across domains in a universal access environment.

In the *NATO C3 Technical Architecture*, NATO also defines four degrees of data exchange interoperability.[10] The four degrees of interoperability are:

- **NATO Degree 1**: **Unstructured Data Exchange**. Involves the exchange of human-interpretable unstructured data such as the free text found in operational estimates, analysis, and papers.
- **NATO Degree 2**: **Structured Data Exchange**. Involved the exchange of human-interpretable structured data intended for manual and/of automated handling, but requires manual compilation, receipt, and/or message dispatch.
- **NATO Degree 3**: **Seamless Sharing of Data**. Involves the automated sharing of data amongst systems based on a common exchange model.
- **NATO Degree 4**: **Seamless Sharing of Information** An extension of Degree 3 to the universal interpretation of information through data processing based on co-operating applications.

[9] NATO C3 Board (NC3B)/ Interoperability Sub-Committee (ISC), NATO C3 Systems Interoperability Directive (NID), draft, May 19, 2003

[10] Allied Data Publication 34, *NATO C3 Technical Architecture*, Volume 4: NC3 Common Standards Profile (NCSP), Version 4.0, March 7, 2003, p. 2.

Because the definition of the data exchange interoperability degrees are too course to support selection of standards, NATO has refined each degree of interoperability into functionally oriented sub-degrees that identify specific interoperability services. For example, Degree 1 has three sub-degrees: Network Connectivity, Basic Document Exchange, and Basic Information Message Exchange. Sub-degrees are defined in the *NATO C3 Technical Architecture, Volume 2*.[11]

The *NATO C3 Technical Architecture* is available at http://194.7.79.15/

4.5.6.5 Maturity Model Relationships

Sections 4.5.6.1 through 4.5.6.4 discuss five models that define levels of increasing sophistication or capability. All five of the maturity constructs are related, in that each addresses aspects of information technology and information systems implementation and interoperation. However, each construct has a logical thrust and a scope of factors considered that are unique and complementary when compared with the others.

The benefit of exploiting and coordinating the disciplines and relationships in these models will help assure that:

- Evolving enterprise capabilities will be based on sound investment strategies for systematic system evolution (a roadmap)
- The various implementations of enterprise IT will be well coordinated and interoperable (LISI)
- The evolution of enterprise systems will support the portability of applications across platforms of different types (COE)
- DoD IT Operational and Systems architectural relationships with our Partners for Peace are well coordinated, and permit interoperable access to information, where there is a determined need to know (NATO Degrees).

[11] Allied Data Publication 34, *NATO C3 Technical Architecture*, Volume 2: Architectural Descriptions and Models, Version 4.0, March 7, 2003, pp. 35–38.

ANNEX A
GLOSSARY

A

AADC	Area Air Defense Command
ACTD	Advanced Concept Technology Demonstration
ADPM	Architecture Development Process Model
ADSI	Advanced Distribution System Interface
AESA	Active Electronically Scanned Area (Radar)
AFATDS	Advanced Field Artillery Tactical Data System
AIS	Automated Information System
ALL-CADM	All-DoD Core Architecture Data Model
AMETL	Agency Mission Essential Task List
AoA	Analysis of Alternatives
AOC	Air Operating Center
AOR	Area of Responsibility
API	Application Program Interface
APS	Asynchronous Protocol Specification
ASAS	All Source Analysis System
ASD (NII)	Assistant Secretary of Defense for Networks and Information Integration
ASM	Application System Management
ATC	Air Tactical Center
ATFLIR	Advanced Targeting Forward Looking Infrared
ATO	Air Tasking Order

B

BCL	Battlefield Coordination Line
BDA	Battle Damage Assessment
BDI	Battle Damage Indications
BEA	Business Enterprise Architecture
BMMP	Business Management Modernization Program
BPR	Business Process Reengineering
BRM	Business Reference Model

C

C2	Command and Control
C2S	Command and Control Support
C3	Consultation, Command, and Control
C4I	Command, Control, Communications, Computers, and Intelligence
C4ISP	Command, Control, Communications, Computers, and Intelligence Support Plan
C4ISR	Command, Control, Communications, Computers, Intelligence, Surveillance, and Reconnaissance
CAP	Crises Action Planning
CAPCO	Controlled Access Program Coordinating Office
CBT	Combat

CCA	Clinger-Cohen Act of 1996 (also referred to as ITRMA)
CCIC2S	Combatant Commanders Integrated Command and Control System
CDD	Capability Development Document
CDL	Common Data Link
CED	Capability Evolution Description
CFO	Chief Financial Officer
CHENG	Chief Engineer
CIC	Combat Intelligence Center
CID	Combat Intelligence Division
CIGSS	Common Imagery Ground/Surface System
CIM	Common Information Model Schemas
CIO	Chief Information Officer
CJCSI	Chairman of the Joint Chiefs of Staff Instruction
CM	Configuration Management
CMOC	Cheyenne Mountain Operations Center
CNO	Chief of Naval Operations
COA	Course of Action
COE	Common Operating Environment
COI	Communities of Interest
COID	Combat Operations Intelligence Division
CONOPS	Concept of Operations
COOP	Continuity of Operations
COS	Class of Service
COSP	Common Open Service Protocol
COTS	Commercial Off the Shelf
CPD	Capability Production Document
CPIC	Capital Planning and Investment Control
CPS	Command Performance System
CRT	Cathode Ray Tube
C/S/As	Commands, Services, and Agencies
CSTA	Combatant Commanders Integrated Command and Control System Technical Architecture
CTAPS	Contingency Theater Automated Planning System
CTL	Command Target List
CTO	Chief Technology Officer
CSV	Comma Separated View

D

DB	Database
DBMS	Database Management System
DCCC	Defense Collection Coordination Center
DCGS	Distributed Common Ground Station
DCIO	Deputy Chief Information Officer
DDA	Defense Data Architecture
DDD	Deadline Delivery Date
DDDS	DoD Data Dictionary System
DDM	Defense Data Model

DEN	Directory Enabled Protocol
DIA	Defense Intelligence Agency
DII	Defense Information Infrastructure
DII COE	Defense Information Infrastructure Common Operating Environment
DISA	Defense Information Systems Agency
DJFLCC	Deputy Joint Force Land Component Commander
DNS	Domain Name Service
DOCC	Deep Operations Coordination Center
DoD	Department of Defense
DoDAF	DoD Architecture Framework
DON CIO	Department of the Navy Chief Information Officer
DOTMLPF	Doctrine, Organization, Training, Material, Leadership and education, Personnel, and Facilities
DRM	Data Reference Model
DTD	Document Type Definition

E

EA	Enterprise Architecture
EDB	Electronic Database
EOM	Enterprise Object Model
ESM	Electronic Warfare Support System
EWS	Electronic Warfare Suite
EXW	Expeditionary Warfare

F

F2C2	Friendly Forces Command Center
FEA	Federal Enterprise Architecture
FEAMS	Federal Enterprise Architecture Management System
FoS	Family of Systems
FSCL	Fire Support Coordination Line
FTI	Fixed Target Indicator; Fleet Tactical Imagery
FY	Fiscal Year
FYDP	Future year Defense Program

G

GALE	Generic Area Limitation Environment
GCCS	Global Command and Control System
GCCS-M	Global Command and Control System - Maritime
GCSS	Global Combat Support System
GIG	Global Information Grid
GIS	Geographic Information System
GMRA	Government Management Reform Act (of 1994)
GMTI	Ground Moving Target Indicator
GPRA	Government Performance Results Act
GUI	Graphical User Interface

H

HCI	Human Computer Interface
HITS	Horizontal Integration Training System
HR	Human Resources
HSI	Human Systems Integration
HTI	Horizontal Technology Integration
HTTP	Hyper Text Transfer Protocol (world wide web protocol)

I

IAW	In Accordance With
IC	Intelligence Community
ICD	Interface Control Document
ICOM	Input, Control, Output, and Mechanism
IDEF0	Integration Definition for Data Modeling
IE	Information Exchange
IEE	Institute of Electrical and Electronics Engineers
IER	Information Exchange Requirement
IETF	Internet Engineering Task Force
IM	Information Management
IMEP	ISC2 Master Integrated Evolution Plan
I/O	Input and Output
IOC	Initial Operational Capability
IP	Internet Protocol
IPA/L	Image Product Archive/Library
IPB	Intelligence Preparation of the Battlefield
ISC2	Integrated Space Command and Control
ISO	International Standards Organization
ISR	Intelligence Surveillance and Reconnaissance
IT	Information Technology
ITTE	Interim Terminal Test Environment

J

JBC	Joint Battlefield Center
JCIDS	Joint Capabilities Integration and Development System
JCMT	Joint Collection Management Tool
JDISS	Joint Deployable Intelligence Support System
JFACC	Joint Force Air Component Commander
JFC	Joint Force Commander
JFLCC	Joint Force Land Component Commander
JFMCC	Joint Force Maritime Component Commander
JFSOCC	Joint Force Special Operations Component Commander
JICCENT	Joint Intelligence Center Central Command
JMA	Joint Mission Area
JMETL	Joint Mission Essential Task List
JMCIS	Joint Maritime Command Information System
JNFL	Joint No Flight List

JOC	Joint Operations Center
JOAC	Joint Operations Air Center
JROC	Joint Requirements Oversight Council
JSCP	Joint Strategic Capabilities Plan
JSIPS-N	Joint Service Imagery Processing System - Navy
JSOAC	Joint Special Operations Air Component
JSOTF	Joint Special Operations Task Force
JTA	Joint Technical Architecture
JTAGS	Joint Service Imagery Processing System - Navy
JSTARS GSM	Joint Surveillance Target Attack Radar System Ground Station Module
JTF	Joint Task Force
JTI	Joint Target Indicator
JTL	Joint Target List
JTT	Joint Targeting Toolkit
JTW	Joint Targeting Workstation
JWCA	Joint Warfighting Capability Assessment
JWID	Joint Warrior Interoperability Demonstration

K

KI	Key Interface
KIP	Key Interface Profile
KPP	Key Performance Parameters

L

LED	Light Emitting Diode
LISI	Levels of Information Systems Interoperability

M

MANET	Mobile Ad-hoc Networks
MAW	Marine Aircraft Wing
MCP	Mission Capability Package
MDA	Milestone Decision Authority
MEA	Munitions Effects Assessment
METL	Mission Essential Task List
MIDB	Modernized Integrated Data Base
MISREP	Mission Report
MMA	Multi-Mission Maritime Aircraft
MNS	Mission Need Statement
MOOTW	Military Operations Other Than War
MOU	Memorandum of Understanding
MPLS	Multi-Protocol Label Switching

N

NCD	Node Connectivity Description
NCES	Network-Centric Enterprise Service
NCID	Net-Centric Information Domain
NCOW	Net-Centric Operations and Warfare
NCSA	National Computer Security Association

NDMP	Network Data Management Protocol
NII	Networks and Information Integration
NMJIC	National Military Joint Intelligence Center
NORAD	North American Aerospace Defense Command
NSS	National Security Systems
NTA IRT	Naval Targeting Afloat Integrated Report Team
N/UWSS	NORAD/USSPACECOM Warfighting Support System
NWDC	Navy Warfare Development Command

O

OA	Operational Activity
OASD	Office of the Assistant Secretary of Defense
OB	Order of Battle
ODBC	Open Database Connectivity
OMB	Office of Management and Budget
OO	Object-Oriented
OPEVAL	Operational Evaluation
OPFAC	Operational Facility
OpSit	Operational Situation
ORD	Operational Requirements Document
OSI	Open System Interface
OTH	Over the Horizon
OTS	Operational Trace Sequences
OV	Operational View

P

PC	Personal Computer
PCMCIA	Personal Computer Memory Card International Association (now called PC Cards)
PDA	Personal Digital Assistant
PL	Product Line
PLSM	Product Line Specification Model
PMA	President’s Management Agenda
PMO	Program Management Office
POM	Program Objective Memorandum
PPBE	Planning, Programming, Budgeting, and Execution
PPBS	Planning, Programming, and Budgeting System
PRM	Performance Reference Model
PTW	Precision Targeting Workstation
PVCS	Portable Voice Communications System

R

RAAP	Rapid Application of Air Power
RDA	Research, Development, and Acquisition
RDBMS	Relational Database Management System
RFC	Request For Comments
RJ	Rivet Joint

ROE	Rules of Engagement
ROI	Return On Investment
ROV	Results of Value
RPSL	Routing Specification Language Protocol
RRS	Rapid Response System
RUP SE	Rational Unified Process for Systems Engineering

S

SBU	Sensitive But Unclassified
SHADE	SHAred Data Engineering
SHARP	Shared Reconnaissance Pod
SLA	Service-Level Agreement
SOAP	Simple Object Access Protocol
SoS	System of Systems
SPP	Security Policy Protocol
SQL	Structured Query Language
SR	Surveillance and Reconnaissance
SRM	Service Component Reference Model
SV	Systems View

T

T&E	Test and Evaluation
TA	Technical Architecture
TADIL	Tactical Digital Information Link
TAMD	Theater Air and Missile Defense
TAMPS	Tactical Aircraft Mission Planning System
TBMCS	Theater Battle Management Core System
TC4I	Tactical Command, Control, Communications, Computers, and Intelligence
TCPIP	Transmission Control Protocol/Internet Protocol
TCS	Tactical Control System
TCT	Time Critical Targeting
TERPES	Tactical Electronic Reconnaissance and Evaluation System
TES-N	Tactical Event System - Navy
TESS/NITES	Tactical Environment Support System/ Navy Integrated Tactical Environmental Subsystem
TLAM APS	Tomahawk Land Attack Missile Afloat Pre-positioning Ships
TPPU	Task, Post, Process, and Use
TRDF	Transportable Radio Direction Finder
TRM	Technical Reference Model
TSA	Target System Architecture
TTP	Tactics, Techniques, and Procedures
TV	Technical Standards View

U

UCAV	Unmanned Combat Air Vehicle
UCRD	Use Case Relationship Diagram

UDDI	Universal Discovery Integration Protocol
UDLR	Uni-Directional Link Routing
UJTL	Universal Joint Task List
UML	Unified Modeling Language
URR	Universal Reference Resource
USAF	United States Air Force
USMTF	United States Message Text Format
USSPACECOM	United States Space Command
USW	Undersea Warfare

V

VMF	Variable Message Format
VPN	Virtual Private Network

W

WBS	Work Breakdown Structure
WOC	Wing Operations Center
WSPL	Web Services Language

X

XML	Extensible Markup Language
XSD	XML Schema

ANNEX B

DICTIONARY OF TERMS

The terms included in this Annex are used in some restrictive or special sense. Certain terms are not defined (e.g., event, function) because they have been left as primitives, and the ordinary dictionary usage should be assumed. Where the source for a definition is known, the reference has been provided in parentheses following the definition. Terms that are being used by both the DoD Architecture Framework (DoDAF) and the C4ISR Core Architecture Data Model (CADM) are marked with an asterisk.

*** Definitions shared between the Framework and CADM documents**

Analysis of Alternatives	The evaluation of operational effectiveness, operational suitability, and estimated costs of alternative systems to meet a mission capability. The analysis assesses the advantages and disadvantages of alternatives being considered to satisfy capabilities, including the sensitivity of each alternative to possible changes in key assumptions or variables. (CJCSI 3170.01C)
Analysis of Materiel Approaches	The JCIDS analysis to determine the best materiel approach or combination of approaches to provide the desired capability or capabilities. Though the AMA is similar to an AoA, it occurs earlier in the analytical process. Subsequent to approval of an ICD, which may lead to a potential ACAT I/IA program, Director Program Analysis & Evaluation provides specific guidance to refine this initial AMA into an AoA. (CJCSI 3170.01C)
Architecture Data Element	One of the data elements that make up the Framework products. Also referred to as architecture data type. (DoDAF)
Attribute	A property or characteristic. (Derived from DATA-ATTRIBUTE, DDDS 4363 (A)) A testable or measurable characteristic that describes an aspect of a system or capability. (CJCSI 3170.01C)
Capability	The ability to execute a specified course of action. (JP 1-02) It is defined by an operational user and expressed in broad operational terms in the format of an initial capabilities document or a DOTMLPF change recommendation. In the case of materiel proposals, the definition will progressively evolve to DOTMLPF performance attributes identified in the CDD and CPD. (CJCSI 3170.01C)
Capability Gaps	Those synergistic resources (DOTMLPF) that are unavailable but potentially attainable to the operational user for effective task execution. (CJCSI 3170.01C)
Capability Development Document	A document that captures the information necessary to develop a proposed program(s), normally using an evolutionary acquisition strategy. The CDD outlines an affordable increment of military useful, logistically supportable, and technically mature capability. (CJCSI 3170.01C)
Capability Production Document	A document that addresses the production elements specific to a single increment of an acquisition program. (CJCSI 3170.01C)
Capstone Requirements Document	A document that contains capability-based requirements that facilitates the development of CDDs and CPDs by providing a common framework and operational concept to guide their development. (CJCSI 3170.01C)
Communications Medium*	A means of data transmission.

Data	A representation of individual facts, concepts, or instructions in a manner suitable for communication, interpretation, or processing by humans or by automatic means. (IEEE 610.12)
Data Model	A representation of the data elements pertinent to an architecture, often including the relationships among the elements and their attributes or characteristics. (DoDAF)
Data-Entity*	The representation of a set of people, objects, places, events or ideas that share the same characteristic relationships. (DDDS 4362 (A))
Defense Acquisition System	The management process by which the Department of Defense provides effective, affordable, and timely systems to the users. (DoDD 5000.1)
DoD Component	The DoD Components consist of the Office of the Secretary of Defense, the Military Departments, the Chairman of the Joint Chiefs of Staff, the combatant commands, the Office of the Inspector General of the Department of Defense, the Defense agencies, the DoD field activities, and all other organizational entities within the Department of Defense. (DoDD 8100.01)
Family of Systems	A set or arrangement of independent systems that can be arranged or interconnected in various ways to provide different capabilities. (DoDD 4630.5)
Format	The arrangement, order, or layout of data/information. (Derived from IEEE 610.5)
Functional Area*	A major area of related activity (e.g., Ballistic Missile Defense, Logistics, or C2 support). (DDDS 4198 (A))
Information	The refinement of data through known conventions and context for purposes of imparting knowledge.
Information Element	Information that is passed from one operational node to another. Associated with an information element are such performance attributes as timeliness, quality, and quantity values. (DoDAF)
Information Exchange	The collection of information elements and their performance attributes such as timeliness, quality, and quantity values. (DoDAF)
Information Exchange Requirement*	A requirement for information that is exchanged between nodes.
Information Technology	Any equipment, or interconnected system or subsystem of equipment, that is used in the automatic acquisition, storage, manipulation, management, movement, control, display, switching, interchange, transmission, or reception of data or information by the executive agency. This includes equipment used by a DoD Component directly, or used by a contractor under a contract with the Component, which (i) requires the use of such equipment, or (ii) requires the use, to a significant extent, of such equipment in the performance of a service or the furnishing of a product. The term "IT" also includes computers, ancillary equipment, software, firmware and similar procedures, services (including support services), and related resources. Notwithstanding the above, the term "IT" does not include any equipment that is acquired by a Federal contractor incidental to a Federal contract. The term "IT" includes National Security Systems (NSS). (DoDD 4630.5)

Initial Capabilities Document	Documents the need for a materiel approach to a specific capability gap derived from an initial analysis of materiel approaches executed by the operational user and, as required, an independent analysis of materiel alternatives. It defines the capability gap in terms of the functional area, the relevant range of military operations, desired effects and time. The ICD summarizes the results of the DOTMLPF analysis and describes why non-materiel changes alone have been judged inadequate in fully providing the capability. (CJCSI 3170.01C)
Integrated Architecture	An architecture consisting of multiple views or perspectives (Operational View, Systems View, and Technical Standards View) that facilitates integration and promotes interoperability across family of systems and system of systems and compatibility among related architectures (DoDD 4630.5) An architecture description that has integrated Operational, Systems, and Technical Standards Views with common points of reference linking the Operational View and the Systems View and also linking the Systems View and the Technical Standards View. An architecture description is defined to be an *integrated architecture* when products and their constituent architecture data elements are developed such that architecture data elements defined in one view are the same (i.e., same names, definitions, and values) as architecture data elements referenced in another view. (DoDAF)
Interoperability	The ability of systems, units, or forces to provide data, information, materiel, and services to and accept the same from other systems, units, or forces and to use the data, information, materiel, and services so exchanged to enable them to operate effectively together. IT and NSS interoperability includes both the technical exchange of information and the end-to-end operational effectiveness of that exchange of information, as required, for mission accomplishment. (DoDD 4630.5)
Joint Capabilities Integrated Development System	Policy and procedures that support the Chairman of the Joint Chiefs of Staff and the Joint Requirements Oversight Council in identifying, assessing, and prioritizing joint military capability needs. (CJCSI 3170.01C)
Key Performance Parameters	Those minimum attributes or characteristics considered most essential for an effective military capability. KPPs are validated by the JROC for JROC interest documents, by the Functional Capabilities Board for Joint Impact documents, and by the DoD Component for Joint Integration or Independent documents. CDD and CPD KPPs are included verbatim in the Acquisition Program Baseline. (CJCSI 3170.01C)
Link	A representation of the physical realization of connectivity between systems nodes.
Mission Area*	The general class to which an operational mission belongs. (DDDS 2305(A)) Note: Within a class, the missions have common objectives.
Mission*	An objective together with the purpose of the intended action. (Extension of DDDS 1(A)) Note: Multiple tasks accomplish a mission. (Space and Naval Warfare Systems Command)
Needline*	A requirement that is the logical expression of the need to transfer information among nodes.
Network*	The joining of two or more nodes for a specific purpose.
Node*	A representation of an element of architecture that produces, consumes, or processes data.

National Security Systems	Telecommunications and information systems operated by the Department of Defense – the functions, operation, or use of which (1) involves intelligence activities, (2) involves cryptologic activities related to national security, (3) involves the command and control of military forces, (4) involves equipment that is an integral part of a weapon or weapons systems, or (5) is critical to the direct fulfillment of military or intelligence missions. Subsection (5) in the preceding sentence does not include procurement of automatic data processing equipment or services to be used for routine administrative and business applications (including payroll, finance, logistics, and personnel management applications). (DoDD 4630.5)
Operational Activity Model	A representation of the actions performed in conducting the business of an enterprise. The model is usually hierarchically decomposed into its actions, and usually portrays the flow of information (and sometimes physical objects) between the actions. The activity model portrays operational actions not hardware/software system functions. (DoDAF)
Operational Activity	An activity is an action performed in conducting the business of an enterprise. It is a general term that does not imply a placement in a hierarchy (e.g., it could be a process or a task as defined in other documents and it could be at any level of the hierarchy of the Operational Activity Model). It is used to portray operational actions not hardware/software system functions. (DoDAF)
Operational Node	A node that performs a role or mission. (DoDAF)
Organization*	An administrative structure with a mission. (DDDS 345 (A))
Planning, Programming, Budgeting, and Execution Process	The primary resource allocation process of the DoD. One of three major decision support systems for defense acquisition, PPBE is a systematic process that guides DoD's strategy development, identification of needs for military capabilities, program planning, resource estimation and allocation, acquisition, and other decision processes.
Platform*	A physical structure that hosts systems or system hardware or software items.
Process	A group of logically related activities required to execute a specific task or group of tasks. (Army Systems Architecture Framework) Note: Multiple activities make up a process. (Space and Naval Warfare Systems Command)
Report	The DoDAF defines a report to be architecture data elements from one or more products combined with additional information. Reports provide a different way of looking at architecture data.
Requirement*	A need or demand. (DDDS 12451/1 (D))
Role	A function or position. (Webster's)
Rule	Statement that defines or constrains some aspect of the enterprise.
Service	A distinct part of the functionality that is provided by a system on one side of an interface to a system on the other side of an interface. (Derived from IEEE 1003.0)
System	Any organized assembly of resources and procedures united and regulated by interaction or interdependence to accomplish a set of specific functions. (DoDAF)
System Data Element	A basic unit of data having a meaning and distinct units and values. (Derived from 8320.1) The architecture data element or type that stores data from the architecture domain (i.e., it has a value) that is produced or consumed by a system function and that has system data exchange attributes as specified in the Systems Data Exchange Matrix. (DoDAF)

System Data Exchange	The collection of System Data Elements and their performance attributes such as timeliness, quality, and quantity values. (DoDAF)
System Function*	A data transform that supports the automation of activities or information elements exchange. (DoDAF)
Systems Node	A node with the identification and allocation of resources (e.g., platforms, units, facilitie s, and locations) required to implement specific roles and missions. (DoDAF)
System of Systems	A set or arrangement of independent systems that are related or connected to provide a given capability. The loss of any part of the system will degrade the performance or capabilities of the whole. (DoDD 4630.5)
Task	A discrete unit of work, not specific to a single organization, weapon system, or individual, that enables missions or functions to be accomplished. (Extension from UJTL, JCSM 3500.04A, 1996). Note: Multiple processes accomplish a task; a single process may support multiple tasks. (Space and Naval Warfare Systems Command)
Universal Reference Resources	Reference models and information standards that serve as sources for guidelines and attributes that must be consulted while building architecture products. (DoDAF)

ANNEX C

DICTIONARY OF UML TERMS

The terms included here are UML terms. They convey some restrictive or special sense in this section. The sources for these definitions are [Booch, 1999] and [Rumbaugh, 1999].

Abstract Class	A class that cannot be directly instantiated. Contrast: concrete class.
Abstraction	1. The act of identifying the essential characteristics of a thing that distinguish it from all other kinds of things. Abstraction involves looking for similarities across sets of things by focusing on their essential common characteristics. An abstraction always involves the perspective and purpose of the viewer; different purposes result in different abstractions for the same things. All modeling involves abstraction, often at many levels for various purposes. 2. A kind of dependency that relates two elements that represent the same concept at different abstraction levels.
Action	The specification of an executable statement that forms an abstraction of a computational procedure. An action typically results in a change in the state of the system and can be realized by sending a message to an object or modifying a link or a value of an attribute.
Action Sequence	An expression that resolves to a sequence of actions.
Action State	A state that represents the execution of an atomic action, typically the invocation of an operation.
Activation	The execution of an action.
Active Class	A class whose instances are active objects. See: active object.
Active Object	An object that owns a thread and can initiate control activity. An instance of active class. See: active class, thread.
Activity Graph	A special case of a state machine that is used to model processes involving one or more classifiers. Contrast: statechart diagram.
Actor [Class]	A coherent set of roles that users of use cases play when interacting with these use cases. An actor has one role for each use case with which it communicates.
Actual Parameter	Synonym: argument.
Adornments	Textual or graphical items that are added to an element's basic notation and are used to visualize details from the element's specification. (one of two annotation mechanisms in UML)
Aggregate [Class]	A class that represents the whole in an aggregation (whole -part) relationship. See: aggregation.
Aggregation	A special form of association that specifies a whole -part relationship between the aggregate (whole) and a component part. See: composition.
Annotation Mechanisms	Annotations of existing items in a UML diagram. The two annotation mechanisms are specifications and adornments.
Architecture	The organizational structure and associated behavior of a system. An architecture can be recursively decomposed into parts that interact through interfaces, relationships that connect parts, and constraints for assembling parts. Parts that interact through interfaces include classes, components, and subsystems.
Artifact	A piece of information that is used or produced by a software development process, such as an external document or a work product. An artifact can be a model, description, or software.

Association	The semantic relationship between two or more classifiers that involves connections among their instances.
Attribute	An attribute is a named property of a class that describes a range of values that instances of the property may hold.
Building Blocks	There are three kinds of building blocks in UML: things, relationships, and diagrams.
Class	A description of a set of objects that share the same attributes, operations, methods, relationships, and semantics. A class may use a set of interfaces to specify collections of operations it provides to its environment. See: interface.
Class Diagram	A diagram that shows a collection of declarative (static) model elements such as classes, types, and their contents and relationships.
Collaboration	The specification of how an operation or classifier, such as a use case, is realized by a set of classifiers and associations playing specific roles used in a specific way. The collaboration defines an interaction. See: interaction.
Collaboration Diagram	A diagram that shows interactions organized around the structure of a model, using either classifiers and associations or instances and links. Unlike a sequence diagram, a collaboration diagram shows the relationships among the instances. Sequence diagrams and collaboration diagrams express similar information, but show it in different ways. See: sequence diagram.
Component	A modular, deployable, and replaceable part of a system that encapsulates implementation and exposes a set of interfaces. A component is typically specified by one or more classifiers (e.g., implementation classes) that reside on it, and may be implemented by one or more artifacts (e.g., binary, executable, or script files). Contrast: artifact.
Component Diagram	A diagram that shows the organizations and dependencies among components.
Concrete Class	A class that can be directly instantiated. Contrast: abstract class.
Constraint	A semantic condition or restriction. Certain constraints are predefined in the UML; others may be user defined. Constraints are one of three extensibility mechanisms in UML. See: tagged value, stereotype.
Container	1. An instance that exists to contain other instances and that provides operations to access or iterate over its contents (e.g., arrays, lists, sets). 2. A component that exists to contain other components.
Containment Hierarchy	A namespace hierarchy consisting of model elements and the containment relationships that exist between them. A containment hierarchy forms a graph.
Context	A view of a set of related modeling elements for a particular purpose, such as specifying an operation.
Dependency	A relationship between two modeling elements, in which a change to one modeling element (the independent element) will affect the other modeling element (the dependent element).
Deployment Diagram	A diagram that shows the configuration of run-time processing nodes and the components, processes, and objects that live on them. Components represent run-time manifestations of code units. See: component diagrams.
Derivation	A relationship between an element and another element that can be computed from it. Derivation is modeled as a stereotype of an abstraction dependency with the keyword Derive.
Derived Element	A [sic] element that can be computed from other elements and is included for clarity or for design purposes even though it adds no semantic information.

Diagram	A graphical presentation of a collection of model elements, most often rendered as a connected graph of arcs (relationships) and vertices (other model elements). UML supports the following diagrams: class diagram, object diagram, use case diagram, sequence diagram, collaboration diagram, state diagram, activity diagram, component diagram, and deployment diagram.
Effect	Specifies an optional procedure to be performed when the transition fires.
Element	An atomic constituent of a model.
Entry action	An action executed upon entering a state in a state machine regardless of the transition taken to reach that state.
Event	The specification of a significant occurrence that has a location in time and space. In the context of state diagrams, an event is an occurrence that can trigger a transition.
Exit Action	An action executed upon exiting a state in a state machine regardless of the transition taken to exit that state.
Extend	A relationship from an extension use case to a base use case, specifying how the behavior defined for the extension use case augments (subject to conditions specified in the extension) the behavior defined for the base use case. The behavior is inserted at the location defined by the extension point in the base use case. The base use case does not depend on performing the behavior of the extension use case. See: extension point, include.
Guard	A Boolean predicate that provides a fine-grained control over the firing of the transition. It must be true for the transition to be fired. It is evaluated at the time the Event is dispatched. There can be at most one guard per transition.
Generalizable Element	A model element that may participate in a generalization relationship. See: generalization.
Generalization	A taxonomic relationship between a more general element and a more specific element. The more specific element is fully consistent with the more general element and contains additional information. An instance of the more specific element may be used where the more general element is allowed. See: inheritance.
Inheritance	The mechanism by which more specific elements incorporate structure and behavior of more general elements related by behavior. See: generalization.
Instance	An individual entity with its own identity and value.
Interaction	A specification of how stimuli are sent between instances to perform a specific task. The interaction is defined in the context of a collaboration. See: collaboration.
Interaction Diagram	A generic term that applies to several types of diagrams that emphasize object interactions. These include collaboration diagrams and sequence diagrams.
Interface	A named set of operations that characterize the behavior of an element.
Link	A semantic connection among a tuple of objects. An instance of an association. See: association.
Link End	An instance of an association end. See: association end.
Message	A specification of the conveyance of information from one instance to another, with the expectation that activity will ensue. A message may specify the raising of a signal or the call of an operation.
Model	A semantically complete abstraction of a system.
Node	A node is a classifier that represents a run-time computational resource, which generally has at least a memory and often processing capability. Run-time objects and components may reside on nodes.

Notes	Notes may contain any combination of text or graphics. A note that renders a comment has no semantic impact; it does not alter the meaning of the model to which it is attached. Notes are used to specify things like requirements, observations, reviews, and explanations, in addition to rendering constraints.
Object	An entity with a well-defined boundary and identity that encapsulates state and behavior. State is represented by attributes and relationships; behavior is represented by operations, methods, and state machines. An object is an instance of a class. See: class, instance.
Object Diagram	A diagram that encompasses objects and their relationships at a point in time. An object diagram may be considered a special case of a class diagram or a collaboration diagram. See: class diagram, collaboration diagram.
Operations	An operation is the implementation of a service that can be requested from any object of the class to affect behavior.
Package	A package is a general-purpose mechanism for organizing elements into groups. Graphically, a package is rendered as a tabbed folder.
Postcondition	A constraint that must be true at the completion of an operation.
Precondition	A constraint that must be true when an operation is invoked.
Realization	The relationship between a specification and its implementation; an indication of the inheritance of behavior without the inheritance of structure.
Refinement	A relationship that represents a fuller specification of something that has already been specified at a certain level of detail. For example, a design class is a refinement of an analysis class.
Relationship	A semantic connection among model elements. Examples of relationships include associations and generalizations.
Relationships	There are four kinds of relationships in the UML: Dependency, Association, Generalization, Realization.
Sequence Diagram	A diagram that shows object interactions arranged in time sequence. In particular, it shows the objects participating in the interaction and the sequence of messages exchanged. Unlike a collaboration diagram, a sequence diagram includes time sequences but does not include object relationships. A sequence diagram can exist in a generic form (describes all possible scenarios) and in an instance form (describes one actual scenario). Sequence diagrams and collaboration diagrams express similar information, but show it in different ways. See: collaboration diagram.
Signal	The specification of an asynchronous stimulus communicated between instances. Signals may have parameters.
Specification	A declarative description of what something is or does. Contrast: implementation (one of two Annotation mechanisms in UML).
Source	Designates the originating state vertex (state or pseudostate) of the transition.
State	A condition or situation during the life of an object during which it satisfies some condition, performs some activity, or waits for some Event. Contrast: state [OMA].
State Machine	A behavior that specifies the sequences of states that an object or an interaction goes through during its life in response to Events, together with its responses and actions.
Statechart Diagram	A diagram that shows a state machine. See: state machine.
Stereotype	A new type of modeling element that extends the semantics of the metamodel. Stereotypes must be based on certain existing types or classes in the metamodel. Stereotypes may extend the semantics, but not the structure of pre-existing types and classes. Certain stereotypes are predefined in the UML, others may be user defined. Stereotypes are one of three extensibility mechanisms in UML. See: constraint, tagged value.

Stimulus	The passing of information from one instance to another, such as raising a signal or invoking an operation. The receipt of a signal is normally considered an Event. See: message.
Swimlane	A partition on an activity diagram for organizing the responsibilities for actions. Swimlanes typically correspond to organizational units in a business model. See: partition.
Tagged Values	Everything in the UML has its own set of properties: classes have names, attributes, and operations, and so on. With stereotypes, you can add new things to the UML; with tagged values, you can add new properties.
Target	Designates the target state vertex that is reached when the transition is taken.
Things	The abstractions that are first-class citizens in a model; relationships tie these things together; diagrams group interesting collections of things. There are four kinds of things in the UML: structural things, behavioral things, grouping things, and annotational things.
Thread [of Control]	A single path of execution through a program, a dynamic model, or some other representation of control flow. Also, a stereotype for the implementation of an active object as lightweight process. See process.
Time Event	An event that denotes the time elapsed since the current state was entered. See: event.
Time Expression	An expression that resolves to an absolute or relative value of time.
Trace	A dependency that indicates a historical or process relationship between two elements that represent the same concept without specific rules for deriving one from the other.
Transient Object	An object that exists only during the execution of the process or thread that created it.
Transition	A relationship between two states indicating that an object in the first state will perform certain specified actions and enter the second state when a specified Event occurs and specified conditions are satisfied. On such a change of state, the transition is said to fire.
Trigger	Specifies the event that fires the transition. There can be at most one trigger per transition.
Type	A stereotyped class that specifies a domain of objects together with the operations applicable to the objects, without defining the physical implementation of those objects. A type may not contain any methods, maintain its own thread of control, or be nested. However, it may have attributes and associations. Although an object may have at most one implementation class, it may conform to multiple different types. See also: implementation class Contrast: interface.
Use Case [Class]	The specification of a sequence of actions, including variants, that a system (or other entity) can perform, interacting with actors of the system. See: use case instances.
Use Case Diagram	A diagram that shows the relationships among actors and use cases within a system.
Use Case Instance	The performance of a sequence of actions being specified in a use case. An instance of a use case. See: use case class.
Use Case Model	A model that describes a system's functional requirements in terms of use cases.

ANNEX D
CADM KEY ENTITY DEFINITIONS

Source: DoD Data Dictionary System (DDDS).

ACTION	(325/1) (A) AN ACTIVITY.
ACTION-VERB	(11373/1) (A) A FUNCTION TO BE PERFORMED.
ACTIVITY-MODEL-INFORMATION-ELEMENT-ROLE	(4182/2) (A) THE ROLE ASSIGNED TO AN INFORMATION-ELEMENT FOR A PROCESS-ACTIVITY IN A SPECIFIC ACTIVITY-MODEL.
ACTIVITY-MODEL-THREAD	(20160/1) (A) A PATH IN AN ACTIVITY-MODEL CONSISTING OF SEQUENTIAL INFORMATION FLOWS FROM ONE PROCESS-ACTIVITY TO ANOTHER.
AGREEMENT	(332/1) (A) AN ARRANGEMENT BETWEEN PARTIES.
ANTENNA-TYPE	(6542/2) (A) THE CLASSIFICATION OF A DEVICE FOR THE COLLECTION OR RADIATION OF ELECTROMAGNETIC SIGNALS.
ARCHITECTURE	(19524/1) (A) THE STRUCTURE OF COMPONENTS, THEIR RELATIONSHIPS, AND THE PRINCIPLES AND GUIDELINES GOVERNING THEIR DESIGN AND EVOLUTION OVER TIME.
ARCHITECTURE-CHANGE-PROPOSAL-REVIEW	(22443/1) (A) THE CHARACTERIZATION OF A CONFIGURATION MANAGEMENT ACTIVITY FOR CHANGES TO ARCHITECTURE.
ARCHITECTURE-ORGANIZATION	(19546/1) (A) THE RELATION OF AN ARCHITECTURE TO A SPECIFIC ORGANIZATION.
AUTOMATED-INFORMATION-SYSTEM	(8020/1) (A) AN INTEGRATED SET OF COMPONENTS USED TO ELECTRONICALLY MANAGE DATA.
BATTLEFIELD-FUNCTIONAL-AREA-PROPONENT	(19563/1) (A) A DISCRETE AREA OF RESPONSIBILITY READILY IDENTIFIABLE BY FUNCTION PERFORMED WHICH CONTRIBUTES DIRECTLY TO BATTLEFIELD MANAGEMENT.
BUSINESS-SUBFUNCTION	(22594/1) (A) THE LOWER-LEVEL SET OF FUNCTIONS PERFORMED BY THE FEDERAL GOVERNMENT FOR A SPECIFIC LINE-OF-BUSINESS.
CAPABILITY	(333/1) (A) AN ABILITY TO ACHIEVE AN OBJECTIVE.
COMMUNICATION-CIRCUIT	(19575/1) (A) A PATH USED FOR TRANSMITTING DATA.
COMMUNICATION-CIRCUIT-TYPE	(19576/1) (A) A KIND OF PATH USED FOR TRANSMITTING DATA.
COMMUNICATION-LINK-TYPE	(19579/1) (A) A GENERIC KIND OF COMMUNICATION-LINK.
COMMUNICATION-MEANS	(19580/1) (A) A PHYSICAL OR ELECTROMAGNETIC INSTANTIATION OF TELECOMMUNICATIONS.
COMMUNICATION-MEDIUM	(19582/1) (A) A MODE OF DATA TRANSMISSION.
COMMUNICATION-SPACE-USE-CLASS	(19585/1) (A) THE SPECIFICATION OF CATEGORIES OF UTILIZATION OF SPACE FOR TELECOMMUNICATION IN BUILDINGS AND OTHER FACILITIES.
COST-BASIS	(19590/1) (A) THE SPECIFICATION USED TO DETERMINE AN UNDERLYING EXPENSE.

COUNTRY	(39/1) (A) A NATION OF THE WORLD.
DATA-ITEM-TYPE	(19595/1) (A) A KIND OF DATA-ITEM.
DECISION-MILESTONE	(20170/1) (A) A DECISION POINT THAT SEPARATES THE PHASES OF A DIRECTED, FUNDED EFFORT THAT IS DESIGNED TO PROVIDE A NEW OR IMPROVED MATERIAL CAPABILITY IN RESPONSE TO A VALIDATED NEED.
DEFENSE-OCCUPATIONAL-SPECIALTY-CROSS-REFERENCE	(22526/1) ® THE RELATIONSHIP OF THE DEPARTMENT OF DEFENSE OCCUPATIONAL CONVERSIONS TO SERVICE-SPECIFIC OCCUPATIONAL SPECIALTIES.
DEPLOYMENT-LOCATION-TYPE	(19596/1) (A) THE CHARACTERIZATION OF A KIND OF GENERIC PLACE FOR DEPLOYED OPERATIONS.
DOCUMENT	(119/1) (A) RECORDED INFORMATION REGARDLESS OF PHYSICAL FORM.
EVENT	(49/1) (A) A SIGNIFICANT OCCURRENCE.
EVENT-NODE-CROSS-LINK	(19978/1) (A) THE SPECIFICATION OF HOW A SPECIFIC EVENT FOR A SPECIFIC ORIGINATOR NODE TEMPORALLY RELATES TO ANOTHER TERMINATOR NODE SUBJECT TO A CONSTRAINT.
EVENT-TYPE	(12341/1) (A) A CATEGORY OF EVENT.
EXCHANGE-RELATIONSHIP -TYPE	(19608/1) (A) THE SPECIFICATION OF A CLASS OF PAIRING FOR INFORMATION EXCHANGE.
FACILITY	(334/1) (A) REAL PROPERTY, HAVING A SPECIFIED USE, THAT IS BUILT OR MAINTAINED BY PEOPLE.
FACILITY-CLASS	(5742/1) (A) THE HIGHEST LEVEL OF REAL PROPERTY CLASSIFICATION BY THE DEPARTMENT OF DEFENSE.
FACILITY-IMPROVEMENT-ACTIVITY	(19541/1) (A) A PROCESS TO IMPROVE CAPABILITIES FOR A SPECIFIC FACILITY.
FACILITY-TYPE	(50/1) (A) A SPECIFIC KIND OF FACILITY.
FEATURE	(4134/2) (A) A SET OF CHARACTERISTICS, STRUCTURES, OR OTHER ENTITIES THAT ARE OF MILITARY SIGNIFICANCE.
FUNCTIONAL-AREA	(4198/2) (A) A MAJOR AREA OF RELATED ACTIVITY.
FUNCTIONAL-PROCESS-FUNCTION	(22044/1) (A) A GENERAL CLASS OF ACTIVITY IN A SPECIFIC FUNCTIONAL-AREA.
GUIDANCE	(336/4) (A) A STATEMENT OF DIRECTION RECEIVED FROM A HIGHER ECHELON.
HAND-RECEIPT	(21353/1) (A) THE SPECIFICATION OF TRANSFER OF PROPERTY RESPONSIBILITY.
ICON-CATALOG	(19625/1) (A) A DIRECTORY OF IMAGES DEPICTED IN GRAPHICAL PRESENTATION SOFTWARE.
ICON-DATA-CATEGORY	(22294/1) (A) A CLASSIFICATION OF ELEMENTS OF INFORMATION THAT APPLY TO ICONS WITHIN AN ICON-CATALOG.
ICON-DATA-REQUIREMENT	(22295/1) (A) THE SPECIFICATION OF WHETHER AN ASSOCIATED ELEMENT OF INFORMATION IS MANDATORY FOR A SPECIFIC ICON.
IDENTIFICATION-FRIEND-FOE	(17031/1) (A) THE RECOGNIZED HOSTILITY CHARACTERIZATION OF A BATTLEFIELD OBJECT.

IMPLEMENTATION-TIME-FRAME	(19731/1) (A) THE SPECIFICATION OF A GENERAL CHRONOLOGICAL PERIOD FOR THE INSTANTIATION OF A CONCEPT, SYSTEM, OR CAPABILITY.
INFLATION-FACTOR	(19732/1) (A) ADJUSTMENTS TO COSTS THAT DEPEND ON FISCAL YEAR.
INFORMATION-ASSET	(4246/3) (A) AN INFORMATION RESOURCE.
INFORMATION-ELEMENT	(4199/2) (A) A FORMALIZED REPRESENTATION OF DATA SUBJECT TO A FUNCTIONAL PROCESS.
INFORMATION-TECHNOLOGY-REGISTRATION	(20501/1) (A) THE IDENTIFICATION OF A MISSION-CRITICAL/MISSION-ESSENTIAL INFORMATION TECHNOLOGY SYSTEM OR OTHER ASSET.
INFORMATION-TECHNOLOGY-STANDARD-CATEGORY	(20513/1) (A) A CLASSIFICATION OF INFORMATION-TECHNOLOGY-STANDARD.
INTERNAL-DATA-MODEL-TYPE	(9289/2) (A) A CLASSIFICATION OF AN INTERNAL-DATA-MODEL.
INTERNET-ADDRESS	(19762/1) (A) THE SPECIFICATION OF A VALUE OR RANGE OF VALUES CONSTITUTING THE LABEL FOR A NODE ON THE INTERNET.
INTEROPERABILITY-DOCUMENT-TYPE	(22390/1) (A) A KIND OF DOCUMENT THAT FOCUSES ON PROPERTIES WHICH ENABLE SYSTEM INTEROPERATION.
LANGUAGE	(2228/1) (A) A MEANS OF COMMUNICATION BASED ON A FORMALIZED SYSTEM OF SOUNDS AND/OR SYMBOLS.
LINE-OF-BUSINESS	(22593/1) (A) THE TOP-LEVEL SET OF FUNCTIONS PERFORMED BY THE FEDERAL GOVERNMENT.
LOCATION	(343/2) (A) A SPECIFIC PLACE.
MATERIEL	(337/1) (A) AN OBJECT OF INTEREST THAT IS NON-HUMAN, MOBILE, AND PHYSICAL.
MATERIEL-ITEM	(787/1) (A) A CHARACTERIZATION OF A MATERIEL ASSET.
MEASURE-UNIT	(2482/2) (A) THE INCREMENT BY WHICH MATTER IS MEASURED.
MILITARY-PLATFORM	(22100/1) (A) AN OBJECT FROM WHICH OR THROUGH WHICH MILITARY TASKS CAN BE CONDUCTED.
MILITARY-TELECOMMUNICATION-USE	(19773/1) (A) THE CHARACTERIZATION OF SPECIFIC USE-DEPENDENT BUT FACILITY-INDEPENDENT PARAMETERS FOR ESTIMATING THE COMMUNICATIONS, WIRING, AND EQUIPMENT REQUIRED BY MILITARY OCCUPANTS OF FACILITIES.
MILITARY-UNIT-LEVEL	(42/2) (A) A MILITARY-UNIT ACCORDING TO A STRATUM, ECHELON, OR POINT WITHIN THE MILITARY COMMAND HIERARCHY AT WHICH CONTROL OR AUTHORITY IS CONCENTRATED.
MISSION	(1/3) (A) THE TASK, TOGETHER WITH THE PURPOSE, THAT CLEARLY INDICATES THE ACTION TO BE TAKEN.
MISSION-AREA	(2305/1) (A) THE GENERAL CLASS TO WHICH AN OPERATIONAL MISSION BELONGS.
MODELING-AND-SIMULATION-JUSTIFICATION	(19776/1) (A) A STATEMENT PROVIDING RATIONALE TO JUSTIFY REQUIREMENTS FROM THE POINT OF VIEW OF MODELING AND SIMULATION.

NETWORK	(10972/1) (A) THE SPECIFICATION FOR THE JOINING OF TWO OR MORE NODES FOR A SPECIFIC PURPOSE.
NETWORK-CONTROLLER-TYPE	(20591/2) (A) THE KIND OF FUNCTIONAL PROPONENT WHO EXERCISES AUTHORITY OVER A NETWORK.
NETWORK-ECHELON	(22486/1) (A) THE NORMAL OPERATIONAL LEVEL SUPPORTED BY A NETWORK.
NETWORK-TYPE	(11570/1) (A) A SPECIFIC KIND OF NETWORK.
NODE	(956/1) (A) A ZERO DIMENSIONAL TOPOLOGICAL PRIMITIVE THAT DEFINES TOPOLOGICAL RELATIONSHIPS.
NODE-SYSTEM-ASSET-OWNERSHIP	(20009/1) (A) THE POSSESSION, IN WHOLE OR PART, OF THE OBJECTS OF VALUE ASSOCIATED TO A SPECIFIC NODE-SYSTEM.
NODE-SYSTEM-COST-MANAGEMENT	(20011/1) (A) THE AMOUNTS ASSOCIATED WITH VARIOUS ASPECTS OF THE MANAGEMENT OF A NODE-SYSTEM.
OCCUPATION	(2009/1) (A) A FIELD OF WORK.
OPERATIONAL-CONDITION	(19589/1) (A) A VARIABLE OF THE OPERATIONAL ENVIRONMENT OR SITUATION IN WHICH A UNIT, SYSTEM, OR INDIVIDUAL IS EXPECTED TO OPERATE THAT MAY AFFECT PERFORMANCE.
OPERATIONAL-DEPLOYMENT-MISSION-TYPE	(19848/1) (A) THE KIND OF HIGH-LEVEL TASKING FOR DEPLOYED OPERATIONS.
OPERATIONAL-DEPLOYMENT-PHASE	(19849/1) (A) A STAGE OF THE OPERATIONAL ACTIVITIES CONDUCTED FOR DEPLOYED OPERATIONS.
OPERATIONAL-FACILITY-ECHELON	(19853/1) (A) A SUBDIVISION OF A HEADQUARTERS (OR) A SEPARATE LEVEL OF COMMAND AS IT APPLIES TO AN OPERATIONAL-FACILITY.
OPERATIONAL-FACILITY-PROPONENT	(19854/2) (A) THE AGENT RESPONSIBLE FOR REQUIREMENTS DEVELOPMENT OF OPERATIONAL FACILITIES.
OPERATIONAL-MISSION-THREAD	(19857/1) (A) AN IDENTIFIED INFORMATION EXCHANGE SEQUENTIAL PROCEDURE TO SUPPORT TASK EXECUTION BY INFORMATION SYSTEMS AND ORGANIZATION-TYPES.
OPERATIONAL-ROLE	(22459/1) (A) THE SPECIFICATION OF A SET OF ABILITIES REQUIRED FOR PERFORMING ASSIGNED ACTIVITIES AND ACHIEVING AN OBJECTIVE.
OPERATIONAL-SCENARIO	(19860/1) (A) A CONCEPT AND SCRIPT FOR POSSIBLE EVENTS AND ACTIONS FOR MILITARY OPERATIONS.
ORGANIZATION	(345/1) (A) AN ADMINISTRATIVE STRUCTURE WITH A MISSION.
ORGANIZATION-TYPE	(892/2) (A) A CLASS OF ORGANIZATIONS.
PERIOD	(1321/1) (A) INTERVAL OF TIME.
PERSON-TYPE	(897/2) (A) A CLASS OF PERSONS.
POINT-OF-CONTACT	(19867/1) (A) A REFERENCE TO A POSITION, PLACE, OFFICE, OR INDIVIDUAL ROLE IDENTIFIED AS A PRIMARY SOURCE FOR OBTAINING INFORMATION.
POINT-OF-CONTACT-TYPE	(22039/1) (A) A KIND OF POINT-OF-CONTACT.
POSITION	(2112/1) (A) A SET OF ESTABLISHED DUTIES.

PROCESS-ACTIVITY	(4204/3) (A) THE REPRESENTATION OF A MEANS BY WHICH A PROCESS ACTS ON SOME INPUT TO PRODUCE A SPECIFIC OUTPUT.
PROCESS-ACTIVITY-FUNCTIONAL-PROCESS	(22043/1) (A) THE MEANS BY WHICH TO CARRY OUT A HIGH-LEVEL FUNCTION.
PROCESS-STATE-VERTEX	(20025/1) (A) THE ABSTRACTION OF AN OBSERVABLE MODE OF BEHAVIOR.
RECORD-TRACKING	(19871/1) (A) INFORMATION REGARDING A SPECIFIC RECORD IN A TABLE OF DATA.
REGIONAL-COST-FACTOR	(19544/1) (A) THE EXPECTED EXPENSE MODIFICATION FOR A GEOGRAPHIC AREA THAT ACCOUNTS FOR SPECIFIC LOCAL COSTS IN RELATION TO A NATIONAL AVERAGE.
RELATION-TYPE	(6515/2) (A) AN ASSOCIATION BETWEEN OBJECTS THAT DEFINES AN INFORMATION ASSET.
ROOM-TYPE	(5605/1) (A) A KIND OF A ROOM.
SATELLITE	(14361/1) (A) A MAN-MADE BODY WHICH REVOLVES AROUND AN ASTROMETRIC-ELEMENT AND WHICH HAS A MOTION PRIMARILY DETERMINED BY THE FORCE OF ATTRACTION OF THAT ASTROMETRIC-ELEMENT.
SECURITY-ACCESS-COMPARTMENT	(16224/2) (A) THE SPECIFICATION OF AN EXCLUSION DOMAIN FOR INFORMATION RELEASED ON A FORMALLY RESTRICTED BASIS (E.G., TO PROTECT SOURCES OR POTENTIAL USE).
SECURITY-CLASSIFICATION	(940/2) (A) THE LEVEL ASSIGNED TO NATIONAL SECURITY INFORMATION AND MATERIAL THAT DENOTES THE DEGREE OF DAMAGE THAT ITS UNAUTHORIZED DISCLOSURE WOULD CAUSE TO NATIONAL DEFENSE OR FOREIGN RELATIONS OF THE UNITED STATES AND THE DEGREE OF PROTECTION REQUIRED.
SKILL	(2226/1) (A) AN ABILITY.
SOFTWARE-LICENSE	(1856/1) (A) THE STIPULATION(S) (AND LEGAL TERMS) BY WHICH THE SOFTWARE MAY BE USED.
SOFTWARE-SERIES	(18977/1) (A) A SET OF SOFTWARE KNOWN BY A SINGLE NAME, BUT COMPRISED OF ONE OR MORE VERSIONS DEVELOPED OVER TIME.
SYSTEM	(326/1) (A) AN ORGANIZED ASSEMBLY OF INTERACTIVE COMPONENTS AND PROCEDURES FORMING A UNIT.
SYSTEM-DETAIL-NODE-TYPE	(22391/1) (A) A KIND OF REPRESENTATION OR DEPICTION APPLICABLE TO SYSTEMS.
SYSTEM-PROPONENT	(22392/1) (A) AN AGENT RESPONSIBLE FOR RESEARCH, DEVELOPMENT, TEST, OR EVALUATION OF SYSTEMS.
SYSTEM-STATUS-TYPE	(22098/1) (A) THE SPECIFICATION OF A KIND OF DEVELOPMENT OR TRANSITION OF ONE OR MORE SYSTEMS.
SYSTEM-TYPE	(9083/2) (A) A SPECIFIC KIND OF SYSTEM.
SYSTEM-USAGE	(22396/1) (A) THE SPECIFICATION OF EMPLOYMENT FOR WHICH SYSTEMS ARE CREATED.
TASK	(290/2) (A) A DIRECTED ACTIVITY.

TECHNICAL-INTERFACE	(21694/1) (A) A GENERIC CONNECTION BETWEEN TWO ELEMENTS THAT IMPLEMENT INFORMATION TECHNOLOGY IN WHICH INFORMATION IS CAPABLE OF BEING TRANSMITTED FROM THE SOURCE ELEMENT TO THE DESTINATION ELEMENT.
TECHNICAL-INTERFACE-TYPE	(19761/1) (A) A KIND OF GENERIC CONNECTION BETWEEN ELEMENTS THAT IMPLEMENT INFORMATION TECHNOLOGY.
TECHNICAL-SERVICE	(19676/1) (A) A DISTINCT PART OF THE SPECIALIZED FUNCTIONALITY THAT IS PROVIDED A SYSTEM ELEMENT ON ONE SIDE OF AN INTERFACE TO A SYSTEM ELEMENT ON THE OTHER SIDE OF AN INTERFACE.
TECHNICAL-SERVICE-AREA	(19677/2) (A) A FIELD OF SPECIALIZED FUNCTIONALITY, USUALLY SPECIFIED BY A REFERENCE-MODEL TO DEFINE INTERFACES.
TECHNOLOGY	(8936/1) (A) THE APPLICATION OF SCIENCE TO MEET ONE OR MORE OBJECTIVES.
TELEPHONE-ADDRESS	(1938/1) (A) AN ELECTRONIC ADDRESS THAT SUPPORTS COMMUNICATION VIA TELEPHONIC MEDIA.
TRANSITION-PROCESS	(20082/1) (A) THE DESCRIPTION OF A METHOD FOR RELATING A “SOURCE” PROCESS-STATE-VERTEX TO A “TARGET” PROCESS-STATE-VERTEX.
UNIFORMED-SERVICE-ORGANIZATION-COMPONENT-TYPE	(2726/2) (A) A SPECIFIC KIND OF SUBDIVISION OF A UNIFORMED-SERVICE-ORGANIZATION.

Note: 115 entities are listed in this table. Source: DoD CADM Baseline 1.0 (18 June 2003)