



FMBT
Financial Management
Business Transformation



ADG | Architecture Development Guide

Version 1.0

2/5/2018

VA



U.S. Department of Veterans Affairs

Office of Finance
Office of Financial Solutions

Document Control

Document Identification

Document Name:	ADG Architecture Development Guide
Abstract:	This document provides a guidance on how to develop a FEAF + DoDAF compliant architecture through a comprehensive collection of model descriptions and criteria.
Document Owner:	USDA/VA FMBT Program
Document Author:	FMBT Solution Architecture & Integration Team
Document Version:	1.0.0
Publish Date:	2/5/2018

Revision History

02/05/2018	Adrian Carter, Director FMBT Technology Workstream	Version 1.0.0 – First Draft.
<MM/DD/YYYY>	<Name, Title>	<Version #.#.# – Change Summary>
<MM/DD/YYYY>	<Name, Title>	<Version #.#.# – Change Summary>

Contents

PART I – INTRODUCTION	8
1. OVERVIEW	9
1.1. DEFINITION & PURPOSE	9
1.2. DOCUMENT OUTLINE	9
PART II – FRAMEWORK.....	11
2. FEDERAL ENTERPRISE ARCHITECTURE FRAMEWORK (FEAF)	12
2.1. STRATEGY DOMAIN	13
2.2. BUSINESS DOMAIN	14
2.3. DATA DOMAIN	15
2.4. APPLICATION DOMAIN	16
2.5. INFRASTRUCTURE DOMAIN	16
2.6. SECURITY DOMAIN	17
3. DOD ARCHITECTURE FRAMEWORK (DODAF).....	18
3.1. DODAF DESCRIPTION	18
3.2. DoDAF VIEWPOINTS, VIEWS, & ELEMENTS	19
3.3. ARCHITECTURE INTERROGATIVES.....	25
PART III – STYLE	27
4. DESIGN PRINCIPLES	28
5. THEME	28
5.1. NOTATION.....	28
5.2. PROPORTION	29
5.3. FONT.....	29
5.4. CONVENTION	30
5.5. COLOR.....	30
6. MODEL TEMPLATES.....	33
PART IV – DESIGN	34
7. ALL VIEWPOINT (AV)	35
7.1. AV-1: OVERVIEW AND SUMMARY INFORMATION	35
7.1.1. Description	35
7.1.2. Checklist	35
7.2. AV-2: INTEGRATED DICTIONARY.....	36
7.2.1. Description	36
7.2.2. Checklist	37
8. CAPABILITY VIEWPOINT (CV).....	37
8.1. CV-1: CAPABILITY VISION	38

8.1.1. Description	38
8.1.2. Checklist	38
8.2. CV-2: CAPABILITY TAXONOMY	38
8.2.1. Description	38
8.2.2. Checklist	39
8.3. CV-3: CAPABILITY PHASING.....	39
8.3.1. Description	39
8.3.2. Checklist	40
8.4. CV-4: CAPABILITY DEPENDENCIES.....	40
8.4.1. Description	40
8.4.2. Checklist	41
8.5. CV-5: CAPABILITY TO ORGANIZATIONAL DEVELOPMENT MAPPING	41
8.5.1. Description	41
8.5.2. Checklist	42
8.6. CV-6: CAPABILITY TO OPERATIONAL ACTIVITIES MAPPING	42
8.6.1. Description	42
8.6.2. Checklist	43
8.7. CV-7: CAPABILITY TO SERVICES MAPPING	43
8.7.1. Description	43
8.7.2. Checklist	44
9. DATA AND INFORMATION VIEWPOINT (DIV)	44
9.1. DIV-1: CONCEPTUAL DATA MODEL	44
9.1.1. Description	44
9.1.2. Checklist	44
9.2. DIV-2: LOGICAL DATA MODEL.....	45
9.2.1. Description	45
9.2.2. Checklist	45
9.3. DIV-3: PHYSICAL DATA MODEL	47
9.3.1. Description	47
9.3.2. Checklist	48
10. OPERATIONAL VIEWPOINT (OV)	48
10.1. OV-1: HIGH-LEVEL OPERATIONAL CONCEPT GRAPHIC.....	48
10.1.1. Description	48
10.1.2. Checklist	49
10.2. OV-2: OPERATIONAL RESOURCE FLOW DESCRIPTION.....	50
10.2.1. Description	50
10.2.2. Checklist	51
10.3. OV-3: OPERATIONAL RESOURCE FLOW MATRIX.....	52
10.3.1. Description	52
10.3.2. Checklist	52
10.4. OV-4: ORGANIZATIONAL RELATIONSHIPS CHART	53
10.4.1. Description	53

10.4.2. Checklist	54
10.5. OV-5A: OPERATIONAL ACTIVITY DECOMPOSITION TREE	55
10.5.1. Description	55
10.5.2. Checklist	55
10.6. OV-5B: OPERATIONAL ACTIVITY MODEL.....	56
10.6.1. Description	56
10.6.2. Checklist	57
10.7. OV-6A: OPERATIONAL RULES MODEL	58
10.7.1. Description	58
10.7.2. Checklist	58
10.8. OV-6B: STATE TRANSITION DESCRIPTION.....	60
10.8.1. Description	60
10.8.2. Checklist	61
10.9. OV-6C: EVENT-TRACE DESCRIPTION	61
10.9.1. Description	61
10.9.2. Checklist	61
11. PROJECT VIEWPOINT (PV)	63
11.1. PV-1: PROJECT PORTFOLIO RELATIONSHIPS.....	63
11.1.1. Description	63
11.1.2. Checklist	63
11.2. PV-2: PROJECT TIMELINES	64
11.2.1. Description	64
11.2.2. Checklist	64
11.3. PV-3: PROJECT TO CAPABILITY MAPPING.....	65
11.3.1. Description	65
11.3.2. Checklist	65
12. STANDARDS VIEWPOINT (STDV)	66
12.1. STDV-1: STANDARDS PROFILE.....	66
12.1.1. Description	66
12.1.2. Checklist	66
12.2. STDV-2: STANDARDS FORECAST	67
12.2.1. Description	67
12.2.2. Checklist	68
13. SYSTEMS VIEWPOINT (SV)	68
13.1. SV-1: SYSTEMS INTERFACE DESCRIPTION	68
13.1.1. Description	68
13.1.2. Checklist	69
13.2. SV-2: SYSTEMS RESOURCE FLOW DESCRIPTION.....	70
13.2.1. Description	70
13.2.2. Checklist	70
13.3. SV-3: SYSTEMS-SYSTEMS MATRIX	71

13.3.1. Description	71
13.3.2. Checklist	72
13.4. SV-4: SYSTEMS FUNCTIONALITY DESCRIPTION (+ DATA FLOW DIAGRAM)	72
13.4.1. Description	72
13.4.2. Checklist	73
13.5. SV-5A: OPERATIONAL ACTIVITY TO SYSTEMS FUNCTION TRACEABILITY MATRIX.....	74
13.5.1. Description	74
13.5.2. Checklist	74
13.6. SV-5B: OPERATIONAL ACTIVITY TO SYSTEMS TRACEABILITY MATRIX.....	75
13.6.1. Description	75
13.6.2. Checklist	75
13.7. SV-6: SYSTEMS RESOURCE FLOW MATRIX	76
13.7.1. Description	76
13.7.2. Checklist	76
13.8. SV-7: SYSTEMS MEASURES MATRIX.....	77
13.8.1. Description	77
13.8.2. Checklist	78
13.9. SV-8: SYSTEMS EVOLUTION DESCRIPTION	78
13.9.1. Description	78
13.9.2. Checklist	79
13.10. SV-9: SYSTEMS TECHNOLOGY & SKILLS FORECAST	80
13.10.1. Description	80
13.10.2. Checklist	80
13.11. SV-10A: SYSTEMS RULES MODEL.....	81
13.11.1. Description	81
13.11.2. Checklist	81
13.12. SV-10B: SYSTEMS STATE TRANSITION DESCRIPTION.....	82
13.12.1. Description	82
13.12.2. Checklist	82
13.13. SV-10C: SYSTEMS EVENT-TRACE DESCRIPTION	83
13.13.1. Description	83
13.13.2. Checklist	83

Index of Tables

TABLE 1. ADG OUTLINE & STRUCTURE.....	10
TABLE 2. DODAF VIEWPOINTS	19
TABLE 3. DODAF VIEWS	21
TABLE 4. DODAF ELEMENTS.....	25
TABLE 5. DODAF INTERROGATIVES.....	26
TABLE 6. GENERAL ARCHITECTURE DESIGN PRINCIPLES.....	28
TABLE 7. RGB CODES	32
TABLE 8. AV-1 CHECKLIST	35
TABLE 9. AV-2 CHECKLIST	37
TABLE 10. CV-1 CHECKLIST	38
TABLE 11. CV-2 CHECKLIST	39
TABLE 12. CV-3 CHECKLIST	40
TABLE 13. CV-4 CHECKLIST	41
TABLE 14. CV-5 CHECKLIST	42
TABLE 15. CV-6 CHECKLIST	43
TABLE 16. CV-7 CHECKLIST	44
TABLE 17. DIV-1 CHECKLIST	44
TABLE 18. DIV-2 CHECKLIST	45
TABLE 19. DIV-3 CHECKLIST	48
TABLE 20. OV-1 CHECKLIST	49
TABLE 21. OV-2 CHECKLIST	51
TABLE 22. OV-3 CHECKLIST	52
TABLE 23. OV-4 CHECKLIST	54
TABLE 24. OV-5A CHECKLIST	55
TABLE 25. OV-5B CHECKLIST	57
TABLE 26. OV-6A CHECKLIST	58
TABLE 27. OV-6B CHECKLIST	61
TABLE 28. OV-6C CHECKLIST	61
TABLE 29. PV-1 CHECKLIST	63
TABLE 30. PV-2 CHECKLIST	64

TABLE 31. PV-3 CHECKLIST	65
TABLE 32. STDV-1 CHECKLIST	66
TABLE 33. STDV-2 CHECKLIST	68
TABLE 34. SV-1 CHECKLIST	69
TABLE 35. SV-2 CHECKLIST	70
TABLE 36. SV-3 CHECKLIST	72
TABLE 37. SV-4 CHECKLIST	73
TABLE 38. SV-5A CHECKLIST	74
TABLE 39. SV-5B CHECKLIST	75
TABLE 40. SV-6 CHECKLIST	76
TABLE 41. SV-7 CHECKLIST	78
TABLE 42. SV-8 CHECKLIST	79
TABLE 43. SV-9 CHECKLIST	80
TABLE 44. SV-10A CHECKLIST	81
TABLE 45. SV10-B CHECKLIST	82
TABLE 46. SV-10C CHECKLIST	83

Index of Figures

FIGURE 1. EXAMPLE OF BUSINESS PROCESS MODEL NOTATION (BPMN)	29
FIGURE 2. AN EXAMPLE DEMONSTRATING GOOD USE OF PROPORTION	29
FIGURE 3. AN EXAMPLE DEMONSTRATING GOOD USE OF FONT	30
FIGURE 4. AN EXAMPLE DEMONSTRATING GOOD APPLICATION OF CONVENTIONS	30
FIGURE 5. AN EXAMPLE DEMONSTRATING GOOD APPLICATION OF COLOR SCHEME	31

Part I – Introduction



1. OVERVIEW

1.1. Definition & Purpose

The Architecture Development Guide (ADG) is a reference document that provides practitioners with a standard set of instructions to support the design, development, and maintenance of an architecture. The ADG is structured with a comprehensive assembly of example architecture models, checklists and criteria, style guidance, tools and techniques, and general instructions related to the development of an architecture that meets both **compliance** and **conformance** standards.

- ✓ **Compliance** is determined by assessing the degree to which a given architecture meets the criteria outlined within this document. A fully compliant architecture can be *quantified* through use of criteria checklists.
- ✓ **Conformance** is determined by assessing the degree to which a given architecture meets the style, theme, and usability requirements found herein and determined by the architecture's consumers. A fully conformed architecture can be *qualified* through formal acceptance and use of the consumers.

Note: Variance is determined by assessing the degree to which a given architecture deviates from the compliance criteria and/or conformance requirements.

Architectures that have a high degree of variance from the compliance and conformance standards outlined in this ADG are considered Fit-For-Purpose (FFP). FFP architectures become necessary when there are unique requirements at play or unforeseen changes to a variety of influential factors. Some examples include delays in project schedule or shifts architecture scope which ultimately lead to a need for fit-for-purpose solutions. Another example is when a program is developing new innovative capabilities that cannot be represented using traditional architecture elements. Both examples are sufficient in justifying an architecture's variation of compliance and/or conformance.

The primary purpose of the ADG is to promote effectiveness and efficiency throughout the architecture effort. It promotes effectiveness through the production of tangible/usable results and efficiency through greater consistency within the architecture. In general, it ensures:

- Consistency in the development of high-quality models in accordance with the frameworks' direction and intent through (1) established stylistic guidance and detailed content guidance for each model, (2) specific quality measures for each model, and (3) alignment with prescribed modeling and element conventions.
- Use of authoritative sources for model development and common conventions and notations for architecture model integration through (1) identified relationships between related models, their processes, and analyses, and (2) specific guidance on the relationships that shall exist between data elements of specific models for them to be considered part of an integrated architecture.
- Use of common conventions for architecture model federation to ensure that solution architectures provide the necessary information and data to support their federation with segment and enterprise architectures.

1.2. Document Outline

The ADG contains a broad range of content related to a fully compliant and conformed architecture. The guidance can be used for enterprise, segment, and solution architectures alike. It can also be used for the development of architectures at any perspectives defined by the Zachman Framework:

1. **Contextual level** (Executive)
2. **Conceptual level** (Manager)
3. **Logical level** (Architect)
4. **Physical level** (Engineer)
5. **Component level** (Technician)

The guidance and corresponding criteria outlined herein are agnostic to these perspectives; it is thus up to the architect to determine both scope and perspective of the subject architecture. This guidance cannot prescribe nor recommend the way an architect determines scope or perspectives of a subject architecture; however, it does provide general best practices and logical instructions that support such decision-making.

Table 1. ADG Outline & Structure



Introduction – provides a general definition and contextual description of the ADG.



Framework – identifies, defines, and describes the applicable architecture frameworks.



Style – outlines the conformance requirements through various design best practices.



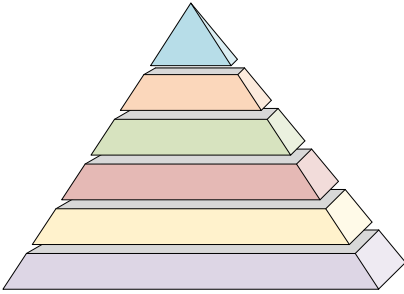
Design – provides architects with design guidance, model criteria, and mock architectures.

The vast majority of the content contained within the ADG is observed within the Design section. The Design section provides a comprehensive collection of instructions and mock architecture models. It is the beating heart of this document and is intended to be used as a tool for architects throughout architecture design and development efforts.

Part II – Framework



2. Federal Enterprise Architecture Framework (FEAF)



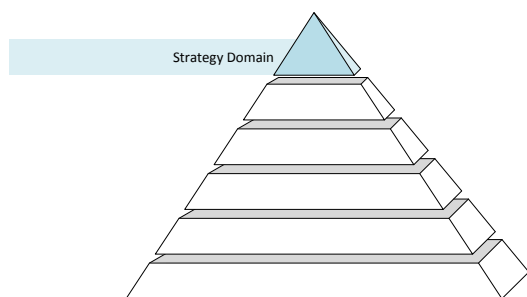
The Federal Enterprise Architecture Framework (FEAF) v2.0 supports planning and decision-making through documentation and information that provides an abstracted view of an enterprise at various levels of scope and detail. The Common Approach to Federal Enterprise Architecture, released in May 2012 as part of the Federal CIO's policy guidance and management tools for increasing shared approaches to IT service delivery, presents an overall approach to developing and using Enterprise Architecture in the Federal Government. The Common Approach promotes increased levels of mission effectiveness by standardizing the development and use of architectures within and between Federal Agencies. This

includes principles for using EA to help agencies eliminate waste and duplication, increase shared services, close performance gaps, and promote engagement among government, industry, and citizens.

FEAF describes a suite of tools to help government planners implement the Common Approach. At its core is the Consolidated Reference Model (CRM), which equips OMB and Federal agencies with a common language and framework to describe and analyze investments. Collectively, the reference models—hereafter referred to as “domains”—comprise a framework for describing important elements of federal agency operations in a common and consistent way. Using the FEAF and its vocabulary, IT portfolios and programs can be better managed and leveraged across the federal government. As such, the six key domains by which the FEAF institutes its structure are as follows:

- **The Strategy Domain**—also referred to as “The Performance Reference Model” (PRM)—links agency strategy, internal business components, and investments to provide a means of measuring the impact of those investments on strategic outcomes.
- **The Business Domain**—also referred to as “The Business Reference Model” (BRM)—describes an organization through a taxonomy of common mission and support service areas instead of through a stove-piped organizational view, thereby promoting intra- and inter-agency collaboration.
- **The Data Domain**—also referred to as “The Data Reference Model” (DRM)—facilitates discovery of existing data holdings residing in “silos” and enables understanding the meaning of the data, how to access it, and how to leverage it to support performance results.
- **The Application Domain**—also referred to as “The Application Reference Model” (ARM)—categorizes the system- and application-related standards and technologies that support the delivery of service capabilities, allowing agencies to share and reuse common solutions and benefit from economies of scale.
- **The Infrastructure Domain**—also referred to as “The Infrastructure Reference Model” (IRM)—categorizes the network/cloud related standards and technologies to support and enable the delivery of voice, data, video, and mobile service components and capabilities.
- **The Security Domain**—also referred to as “The Security Reference Model” (SRM)—provides a common language and methodology for discussing security and privacy in the context of federal agencies’ business and performance goals.

2.1. Strategy Domain



The Strategy Domain (also known as the Performance Reference Model (PRM)) is designed to provide linkage between investments or activities and the strategic vision established by agencies and the Federal government. Historically, linking information management investments and activities has been anecdotal due to a lack of standard approach to describing Agency and cross agency performance attributes. The GPRA Modernization Act of 2010 requires the government to publish performance information through a central web site and make strategic plans and performance reports available in machine readable formats. This

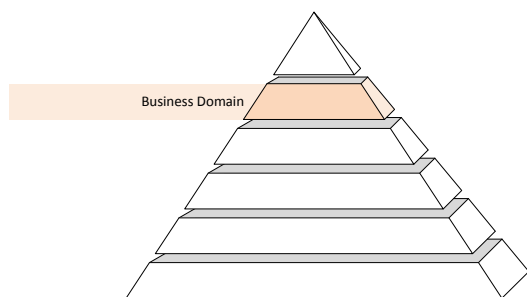
advancement enables more comprehensive and consistent linking of investments and activities to Agency strategic goals and objectives, Agency priority Goals, Cross Agency Priority goals and management areas of focus. The PRM leverages the requirements of the GPRA Modernization Act to establish mechanisms to link directly to the authoritative performance elements published in compliance with the law and provides the means for use of future developments in the mandated central performance website Performance.gov.

There are three areas to the Performance Reference Model. The first is the Goal. This enables grouping of investments and activities through a common and authoritative framework established by agencies in compliance with OMB direction and the GPRA Modernization Act. It allows the identification of common performance elements across investments or activities, and in the future will enable cross-platform information linkages between systems such as Performance.gov and the IT Dashboard. This linkage provides the logical relationships necessary to consistently provide much richer insights into details of the supported performance areas than previously feasible.

The second area of the Performance Reference Model is the Measurement Area. This describes the way the investment or activity supports the achievement of the supported performance element identified by the Agency Goal. Measurement Areas apply to the more detailed performance indicators associated with the investment of activity rather than the functions of the investment or activity. Investment or activity performance indicators should have a clear linkage to the activities, of course, but it is important to recognize that investments or activities may align to multiple measurement areas.

The third area, Measurement Category, refines the Measurement Area. Any Measurement Category may be applied to any Goal. The PRM, like all other reference models, is intended to work in concert with other reference models. The combined descriptive qualities of the multiple perspectives afforded by assigning different reference model perspectives to investments or activities can provide rich insights into what, why and how the investments or activities are undertaken. Previous versions of the PRM included mission function characteristics that were redundant to the BRM. In this version of the PRM the Measurement Category codes have been streamlined to better identify the means by which performance is achieved. Including BRM and PRM mappings with an investment or activity provides information about the strategic basis (why) through the Agency Goal, the means (how) through the measurement category, and the mission functions involved (what) through the BRM taxonomy. Additional mappings to other reference models provide further context for the investment or activity with the SRM providing information about risk, the DRM about the information involved and the ARM and IRM providing the technical details about the implementation.

2.2. Business Domain



The Business Reference Model (BRM) is a classification taxonomy used to describe the type of business functions and services that are performed in the Federal Government. By describing the Federal Government using standard business functions rather than an organizational view, the BRM promotes cross-government collaboration. It enables business and IT leaders to discover opportunities for cost savings and new business capabilities that help to achieve strategic objectives. The BRM describes the “What we do” of the Federal enterprise through the definition of outcome-oriented and measurable functions and services.

While the BRM provides a standardized way of classifying government functions, it is only a model; its true utility and value is realized when it is applied and effectively used in business analysis, design and decision support that help to improve the performance of an agency, bureau or program. The BRM taxonomy is structured as a three-layer hierarchy representing Executive Branch Mission Sectors, Business Functions and Services.

- **Mission Sector** – Identifies the ten business areas of the Federal Government in the Common Approach to EA
- **Business Function** – Describes what the Federal government does at an aggregated level, using the budget function classification codes provided in OMB Circular A-11
- **Service** – Further describes what the Federal government does at a secondary or component level

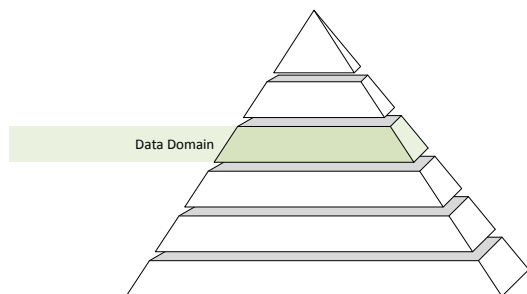
The choice of these three layers for the taxonomy enables aggregation and analysis of IT investments and applications for a variety of different purposes. Including budget function classification codes in the reference model enables detailed analysis of IT investments along the same lines as budget analysis is performed for all other government investments. Including services in the reference model facilitates the search for reusable or sharable applications or components to reduce redundancy and costs for providing services to agencies and citizens.

The BRM benefits the agency at all organizational levels, from executives to developers.

- **Executives and Managers:** Use of a standardized business taxonomy such as the BRM enables executives and managers to see the gaps and redundancies within their enterprise. These gaps and redundancies are opportunities for cost savings and new business capabilities that help achieve the organization’s strategic objectives.
- **Portfolio Managers:** Use of the BRM as a framework for IT portfolio management ensures proper alignment of IT projects and investments to the business needs of the organization. It will also help guide the development of business cases to request and justify funding for future development and maintenance of programs, systems, and applications.
- **Project Managers:** During the concept and planning phase of a project, the BRM allows project managers to identify current business capabilities and determine if or how the proposed project fits into the existing architecture. Project managers can also use the BRM to streamline common business processes to reduce or avoid cost, improve cycle time, and improve customer satisfaction and value. Additionally, application performance may be enhanced by finding better ways of doing business, such as sharing data sources, and developing common data retrieval and storage services.

- **Developers:** From a development perspective, the BRM will enhance the ability for project teams to work towards a common, shareable solution for satisfying business needs. The costs associated with maintaining duplicative applications and services can be reduced by developing sharable services that can be used by more than one application or organization. Integrated service delivery approaches can also reduce the burden on the public by collecting data once and sharing it among systems, thereby reducing the burden on users of those systems.

2.3. Data Domain



The Data Reference Model's (DRM) primary purpose is to promote the common identification, use, and appropriate sharing of data/information across the federal government. The DRM is a flexible and standards-based framework to enable information sharing and reuse via the standard description and discovery of common data and the promotion of uniform data management practices. The DRM provides a standard means by which data may be described, categorized, and shared, and it facilitates discovery and exchange of core information across organizational boundaries.

As a reference model, the DRM is presented as an abstract framework from which concrete implementations may be derived. The DRM's abstract nature will enable agencies to use multiple implementation approaches, methodologies and technologies while remaining consistent with the foundational principles of the DRM. The Data Reference Model taxonomy is defined by a hierarchy in three layers:

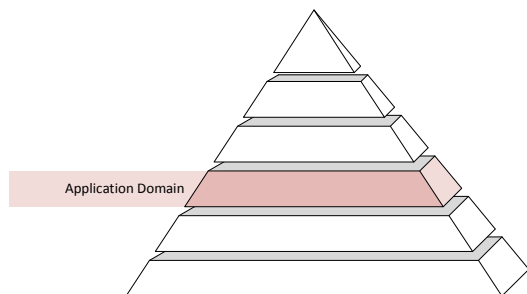
1. Data Domain
2. Data Subject
3. Data Topic

The top rank of the hierarchy consists of four Domains. The middle layer of the hierarchy contains twenty-two Subject elements and the lowest rank of the hierarchy includes one hundred and forty-four Topic elements. The DRM provides a structure and vocabulary for agencies to form a consensus as to how, at a Federal level, to categorize, describe, and share data. There are two primary goals of the DRM:

- **GOAL 1:** Improve the quality and depth of information available for mission performance.
 - **METHOD:** Comparison of data sources containing similar data, though possibly used for a different purpose.
 - **CHALLENGE:** Finding data sources that are worth comparing and the identification of common information between data sources, and thus usable for correlation.
- **GOAL 2:** Facilitate a standardized information exchange across a Community of Interest.
 - **METHOD:** Model the exchange and build exchange schemas using available data standards, such as NIEM.
 - **CHALLENGE:** Federal, state, local, and tribal organizations typically use different data definitions and structures in the storage and exchange of like data across a community of interest; also, as data is exchanged between organizations within or across these domains, transformations or interfaces must be created for each new data source.

How the DRM Helps: The DRM taxonomy identifies data categories, regardless of usage context. Used in concert with the Business Reference Model (BRM) taxonomy, the DRM taxonomy can help classify the data that is managed in a given data source by the mission or business context in which that data is used. Classifying a set of data sources by the DRM and BRM taxonomies produces a data set that can be searched to determine, for example, which data sources contain a common data class but use it for different business contexts. For a large set of data sources, that search capability saves considerable time over manually examining each data source to see if it contains what is required.

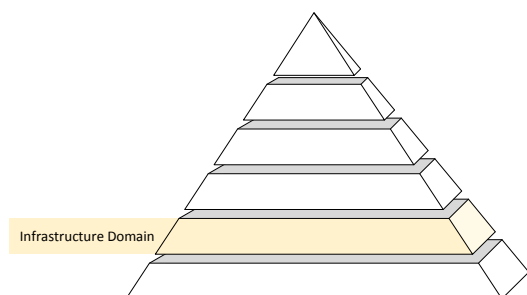
2.4. Application Domain



The purpose of the Application Reference Model (ARM) is to provide the basis for categorizing applications and their components. As agencies map their current and planned Information Systems to the ARM categories, gaps and redundancies will become evident, which will aid in identifying opportunities for sharing, reuse, and consolidation or renegotiation of licenses. This information may be used in conjunction with the other Reference Models to identify these opportunities.

For the purposes of the CRM, Application is defined as: Software components (including websites, databases, email, and other supporting software) resting on Infrastructure that, when aggregated and managed, may be used to create, use, share, and store data and information to enable support of a business function. The ARM is a categorization of different types of software, components and interfaces. It categorizes software that supports or may be customized to support business. It does not include operating systems or software that is used to operate hardware (e.g. firmware) because these are contained in the IRM. It also does not contain mission-specific categorizations for systems because that information can be obtained from mappings to the BRM. The ARM consists of three levels: Systems, Application Components, and Interfaces.

- **Systems** are discrete sets of information technology, data, and related resources, organized for the collection, processing, maintenance, use, sharing, dissemination or disposition of information in support of a specific business process. The ARM Systems category does not include mission-specific systems.
- **Application Components** are self-contained software that can be aggregated or configured to support, or contribute to achieving, many different business objectives. For example, workflow management, document management, records management and many other types of components can support multiple IT Systems and business processes.
- **Interfaces** are protocols used to transfer information from system to system.



2.5. Infrastructure Domain

The Infrastructure Reference Model (IRM) is the taxonomy based reference model for categorizing IT infrastructure and the facilities and network that host the IT infrastructure. The IRM supports definition of infrastructure technology items and best practice guidance to promote positive outcomes across technology implementations.

For the purposes of the CRM, Infrastructure is defined as: The generic (underlying) platform consisting of hardware, software and delivery platform upon which specific/customized

capabilities (solutions, applications) may be deployed. The IRM implementation enables sharing and reuse of infrastructure to reduce costs, increase interoperability across the government and its partners, support efficient acquisition and deployment, and enable greater access to information across enterprises.

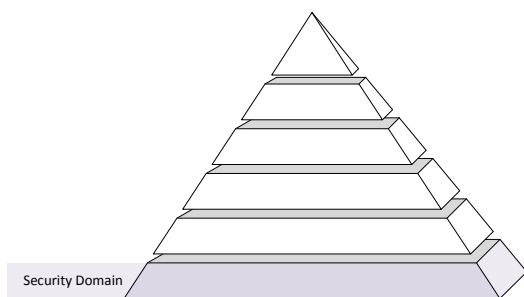
In addition to providing a categorization schema for IT infrastructure assets, the IRM enables analysis of IT infrastructure assets at a Department or Agency level as well as at a Federal Government level. In the Federal context, the IRM is adopted and used to conduct Government-wide analysis of IT infrastructure assets and to identify consolidation initiatives. In the Department or Agency context, the IRM is used to drive good IT infrastructure asset management practices such as identifying end-of-life assets before they affect the mission of an organization and to identify opportunities for sharing and consolidating infrastructure.

The IRM taxonomy is intended to provide a categorization scheme for physical IT assets, the operating systems and firmware that run them, and the locations or facilities that host the physical IT assets. The IRM is divided into three levels:

- Level 1 of the hierarchy, called “Domain”, consists of three entities, Platform, Network and Facility, which are linked and related to each other to enable analysis of IT assets across the three dimensions.
- Level 2 of the hierarchy, called “Area”, consists of 13 total Areas (for example, “Hardware”) linked to the three Domains in Level 1.
- Level 3 of the hierarchy, called “Category”, consists of 90 total Categories (for example, “Personal Computer – Laptop”) linked to the 13 Areas in Level 2.

The adaptive and loosely coupled approach of the IRM supports multiple levels of executive management, capital planning and architecture stakeholders and their analytical needs.

2.6. Security Domain



Security is integral to all architectural domains and at all levels of an organization. As a result, the Security Reference Model (SRM) must be woven into all the sub-architectures of the overarching EA across all the other reference models and it must be considered up and down the different levels of the Enterprise. Enterprise Architecture Governance is the perfect place for security standards, policies, and norms to be developed and followed, since it is an enforcement point for Information Technology investments.

The SRM allows architects to classify or categorize security architecture at all scope levels of the Federal Architecture: International, National, Federal, Sector, Agency, Segment, System and Application. At the highest levels, the SRM is used to transform federal laws, regulations, and publications into specific policies. At the segment level, the SRM is used to transform department specific policies into security controls and measurements. At the system level, it is used to transform segment controls into system specific designs or requirements. Each level of the SRM is critical to the overall security posture and health of an organization and/or system.

The Federal Security Reference Model (SRM) has three areas: Purpose, Risk, and Controls; these are divided into six total subareas (see figure below). Each one of these subareas must be addressed at the enterprise, agency, and system level.

The SRM uses the information from the purpose and risk at each level of the enterprise to find and classify the correct controls to secure the environment.

3. DoD Architecture Framework (DoDAF)



3.1. DoDAF Description

The Department of Defense Architecture Framework (DoDAF) Version 2.0 is the overarching, comprehensive framework and conceptual model enabling the development of architectures to facilitate the ability of executives, managers, and architects to make key decisions more effectively through organized information sharing across enterprises. DoDAF serves as one of the principal pillars supporting Chief Information Officers (CIO) in their responsibilities for development and maintenance of architectures required under the Clinger-Cohen Act.

Executives, managers, and architects are all responsible for the specification of requirements supporting the development of architectures within their areas of authority. They select an architect and an architecture development team to create the architecture in accordance with the requirements they define. DoDAF Conformance ensures reuse of information and that architecture artifacts, models, and viewpoints can be shared with common understanding. DoDAF v2.0 focuses on architectural *metadata* rather than on developing individual *products* as described in previous versions. In general, data can be collected, organized, and stored by a wide range of architecture tools developed by commercial sources. DoDAF v2.0 provides a Data Capture Method for each data group of the DoDAF-Meta-Model (DM2) to guide architects in collecting and organizing the necessary architectural data.

The DoDAF enables architectural content that is "Fit-for-Purpose" as an architectural description consistent with specific project or mission objectives. Because the techniques of architectural description can be applied at myriad levels of an enterprise, the purpose or use of an architectural description at each level will be different in content, structure, and level of detail. Tailoring the architectural description development to address specific, well-articulated, and understood purposes, will help ensure the necessary data is collected at the appropriate level of detail to support specific decisions or objectives. Visualizing architectural data is accomplished through models (e.g., the products described in previous versions of DoDAF). Models can be documents, spreadsheets, dashboards, or other graphical representations and serve as a template for organizing and displaying data in a more easily understood format. When data is collected and presented as a "filled-in" model, the result is called a view. Organized collections of views (often representing processes, systems, services, standards, etc.) are referred to as viewpoints, and with appropriate definitions are collectively called the Architectural Description.

- DoDAF-described Models (also referred to as Models) are created from the subset of data for a particular purpose. Once the DoDAF-described Models are populated with data, these "views" are useful as examples for presentation purposes, and can be used as described, modified, or tailored as needed.
- Fit-for-Purpose Views are user-defined views of a subset of architectural data created for some specific purpose (i.e., "Fit-for-Purpose"). While these views are not described or defined in DoDAF, they can be created, as needed, to ensure that presentation of architectural data is easily understood. This enables organizations to use their own established presentation preferences in their deliberations.

The models described in DoDAF, including those that are legacies from previous versions of the Framework, are provided as pre-defined examples that can be used when developing presentations of architectural data. Specific DoDAF-described Models for a purpose are prescribed by process-owners. All the DoDAF-described Models do not have

to be created. If an activity model is created, a necessary set of data for the activity model is required. Key process owners will decide what architectural data is required, generally through DoDAF-described Models or Fit-for-Purpose Views. The architect and stakeholders select views to ensure that the Architectural Descriptions will support current and future states of the process or activity under review. Selecting Architecture Viewpoints carefully ensures that the views adequately frame concerns, e.g., by explaining the requirements and proposed solutions, in ways that enhance audience understanding.

The DM2 provides information needed to collect, organize, and store data in a way easily understood. The DM2 replaces the Core Architecture Data Model (CADM) which supported previous versions of the DoDAF. DM2 is a data construct that facilitates reader understanding of the use of data within an architecture document. CADM can continue to be used in support of architectures created in previous versions of DoDAF. NOTE: DoDAF v2.0 does NOT prescribe a Physical Data Model (PDM), leaving that task to software developers who will implement the principles and practices of DoDAF in their own software offerings. The core of DoDAF v2.0 is a data-centric approach where the creation of architectures to support decision-making is secondary to the collection, storage, and maintenance of data needed to make efficient and effective decisions. The architect and stakeholders select views to ensure that architectures will explain current and future states of the process or activity under review. Selecting architectural views carefully ensures that they adequately explain the requirement and proposed solution in ways that will enhance audience understanding.

DoDAF v2.0 also provides, but does not require, a methodology in architecture development. It provides guidance and suggestions on how to ensure that other proposed methods can be adapted as needed to meet the requirements for data collection and storage. Similarly, the views presented in DoDAF are examples, intended to serve as a possible visualization of a particular view. DoDAF v2.0 also continues providing support for views (i.e., 'products' developed in previous versions of the Framework). These views do not require any graphical design by toolset vendors.

3.2. DoDAF Viewpoints, Views, & Elements

DoDAF v2.0 is structured using decomposition of *Viewpoints*, *Views*, and *Elements*. A *Viewpoint* is an architecture category and logical grouping of *Views*. A *View* is a tangible product that depicts the architecture and describes actuality. *Views* are produced by the assembly and logical arrangement of architecture *Elements*. The *Elements* found within a DoDAF-compliant architecture make up the DoDAF-Meta-Model (DM2) and are used as the basis to support a fully integrated architecture.

Table 2. DoDAF Viewpoints

ID	Viewpoint	Viewpoint Description
AV	All Viewpoint	Describes the overarching aspects of architecture context that relates to all viewpoints.
CV	Capability Viewpoint	Articulates the capability requirements, the delivery timing, and the deployed capability.
DIV	Data & Information Viewpoint	Articulates the data relationships and alignment structures in the architecture content for the capability and operational requirements, system engineering processes, and systems and services.
OV	Operational Viewpoint	Includes the operational scenarios, activities, and requirements that support capabilities.
PV	Project Viewpoint	Describes the relationships between operational and capability requirements and the various projects being implemented.
SvcV	Services Viewpoint	Details the design for solutions articulating the Performers, Activities, Services, and their Exchanges, providing for or supporting operational and capability functions.
StdV	Standards Viewpoint	Articulates the applicable operational, business, technical, and industry policies, standards, guidance, constraints, and forecasts.
SV	Systems Viewpoint	Details the design for solutions articulating the systems, their composition, interconnectivity, and context providing for or supporting operational and capability functions.

Table 3. DoDAF Views

ID	View	View Description
AV-1	Overview and Summary Information	A description of the Project's Visions, Goals, Objectives, Plans, Activities, Events, Conditions, Measures, Effects (Outcomes), and produced objects.
AV-2	Integrated Dictionary	An architectural data repository with definitions of all terms used throughout the architectural data and presentations.
CV-1	Capability Vision	The overall vision for transformational endeavors, which provides a strategic context for the capabilities described and a high-level scope.
CV-2	Capability Taxonomy	A hierarchy of capabilities which specifies all the capabilities that are referenced throughout one or more Architectural Descriptions.
CV-3	Capability Phasing	The planned achievement of capability at different points in time or during specific periods of time. The CV-3 shows the capability phasing in terms of the activities, conditions, desired effects, rules complied with, resource consumption and production, and measures, without regard to the performer and location solutions.
CV-4	Capability Dependencies	The dependencies between planned capabilities and the definition of logical groupings of capabilities.
CV-5	Capability to Org. Development Mapping	The fulfillment of capability requirements shows the planned capability deployment and interconnection for a particular Capability Phase. The CV-5 shows the planned solution for the phase in terms of performers and locations and their associated concepts.
CV-6	Capability to OA Mapping	A mapping between the capabilities required and the operational activities that those capabilities support.
CV-7	Capability to Services Mapping	A mapping between the capabilities and the services that these capabilities enable.
DIV-1	Conceptual Data Model	The required high-level data concepts and their relationships.
DIV-2	Logical Data Model	The documentation of the data requirements and structural business process (activity) rules. In DoDAF V1.5, this was the OV-7.
DIV-3	Physical Data Model	The physical implementation format of the Logical Data Model entities, e.g., message formats, file structures, physical schema. In DoDAF V1.5, this was the SV-11.
OV-1	High-Level Operational Concept Graphic	The high-level graphical/textual description of the operational concept.
OV-2	Operational Resource Flow Description	A description of the Resource Flows exchanged between operational activities.
OV-3	Operational Resource Flow Matrix	A description of the resources exchanged and the relevant attributes of the exchanges.
OV-4	Organizational Relationships Chart	The organizational context, role or other relationships among organizations.

ID	View	View Description
OV-5a	Operational Activity Decomposition Tree	The capabilities and activities (operational activities) organized in a hierarchal structure.
OV-5b	Operational Activity Model	The context of capabilities and activities (operational activities) and their relationships among activities, inputs, and outputs; Additional data can show cost, performers, or other pertinent information.
OV-6a	Operational Rules Model	One of three models used to describe activity (operational activity). It identifies business rules that constrain operations.
OV-6b	State Transition Description	One of three models used to describe operational activity (activity). It identifies business process (activity) responses to events.
OV-6c	Event-Trace Description	One of three models used to describe activity (operational activity). It traces actions in a scenario or sequence of events.
PV-1	Project Portfolio Relationships	It describes the dependency relationships between the organizations and projects and the organizational structures needed to manage a portfolio of projects.
PV-2	Project Timelines	A timeline perspective on programs or projects, with the key milestones and interdependencies.
PV-3	Project to Capability Mapping	A mapping of projects to capabilities to show how the specific projects and program elements help to achieve a capability.
SvcV-1	Services Context Description	The identification of services, service items, and their interconnections.
SvcV-2	Services Resource Flow Description	A description of Resource Flows exchanged between services.
SvcV-3a	Systems-Services Matrix	The relationships among or between systems and services in a given Architectural Description.
SvcV-3b	Services-Services Matrix	The relationships among services in a given Architectural Description. It can be designed to show relationships of interest, (e.g., service-type interfaces, planned vs. existing interfaces).
SvcV-4	Services Functionality Description	The functions performed by services and the service data flows among service functions (activities).
SvcV-5	OA to Services Traceability Matrix	A mapping of services (activities) back to operational activities (activities).
SvcV-6	Services Resource Flow Matrix	A documentation of service Resource Flow elements being exchanged between services and the attributes of that exchange.
SvcV-7	Services Measures Matrix	The measures (metrics) of Services Model elements for the appropriate time frame(s).
SvcV-8	Services Evolution Description	The planned incremental steps toward migrating a suite of services to a more efficient suite or toward evolving current services to a future implementation.
SvcV-9	Services Technology & Skills Forecast	The emerging technologies, software/hardware products, and skills that are expected to be available in a given set of time frames and that will affect future service development.
SvcV-10a	Services Rules Model	One of three models used to describe service functionality. It identifies constraints that are imposed on systems functionality due

ID	View	View Description
		to some aspect of system design or implementation.
SvcV-10b	Services State Transition Description	One of three models used to describe service functionality. It identifies responses of services to events.
SvcV-10c	Services Event-Trace Description	One of three models used to describe service functionality. It identifies service-specific refinements of critical sequences of events described in the Operational Viewpoint.
StdV-1	Standards Profile	The listing of standards that apply to solution elements.
StdV-2	Standards Forecast	The description of emerging standards and potential impact on current solution elements, within a set of time frames.
SV-1	Systems Interface Description	The identification of systems, system items, and their interconnections.
SV-2	Systems Resource Flow Description	A description of Resource Flows exchanged between systems.
SV-3	Systems-Systems Matrix	The relationships among systems in a given Architectural Description. It can be designed to show relationships of interest, (e.g., system-type interfaces, planned vs. existing interfaces).
SV-4	Systems Functionality Description	The functions (activities) performed by systems and the system data flows among system functions (activities).
SV-5a	OA to SF Traceability Matrix	A mapping of system functions (activities) back to operational activities (activities).
SV-5b	OA to Systems Traceability Matrix	A mapping of systems back to capabilities or operational activities (activities).
SV-6	Systems Resource Flow Matrix	Provides details of system resource flow elements being exchanged between systems and the attributes of that exchange.
SV-7	Systems Measures Matrix	The measures (metrics) of Systems Model elements for the appropriate timeframe(s).
SV-8	Systems Evolution Description	The planned incremental steps toward migrating a suite of systems to a more efficient suite, or toward evolving a current system to a future implementation.
SV-9	Systems Technology & Skills Forecast	The emerging technologies, software/hardware products, and skills that are expected to be available in a given set of time frames and that will affect future system development.
SV-10a	Systems Rules Model	One of three models used to describe system functionality. It identifies constraints that are imposed on systems functionality due to some aspect of system design or implementation.
SV-10b	Systems State Transition Description	One of three models used to describe system functionality. It identifies responses of systems to events.
SV-10c	Systems Event-Trace	One of three models used to describe system functionality. It identifies system-specific refinements of critical sequences of

ID	View	View Description
	Description	events described in the Operational Viewpoint.

Table 4. DoDAF Elements

Element	Description	View
Capabilities	Capabilities are defined as the required capacity, materiel, or expertise needed to effectively support or perform core business functions.	CV-2 CV-4 CV-6
Information Exchanges	An Information Exchange is the passing of data/information between resources (e.g., Systems or Organizations that transmit messages).	OV-2 OV-3
ICOMs	Inputs, Controls, Outputs, and Mechanisms (ICOMs) are characteristics of an Operational Activity. Inputs are comprised of some type of data/information that are required to execute an Operational Activity. Controls consist of those aspects that constrain the Operational Activity (e.g., laws or regulations). Outputs are derived data/information resulting from an Operational Activity. Lastly, Mechanisms are the tools necessary to perform the Operational Activity (e.g., a hosting environment or a software tool).	OV-5b OV-6c
Operational Activities	Operational Activities are those tasks that pertain to an Organization's core business activities. Operational Activities are often enabled and performed manually, or by systems and system functions.	CV-6 OV-2 OV-5a OV-5b SV-5a
Organizations (Performer)	Nodes define the locations at a starting point or ending point of an Information Exchange. They describe physical locations that are in scope of a given system, and can send and/or receive data via varying resources.	AV-2 OV-2 OV-3 SV-1
People (Performer)	People, for the purposes of an architecture, can be thought of as either an end-user of a system, or an individual that carries out a specific manual task.	OV-2 OV-5b OV-6c
Services (Performer)	Services are software functions that are provided via web-based means. Services are a form of web-based communication, usually between proprietary interoperable systems.	SvcV-1
Systems (Performer)	Systems refer to information systems (usually hardware/software platforms) that perform system functions to: 1) automate otherwise manual processes, 2) establish interfaces and workflows, 3) provide an application for fulfilling a specific function.	SV-1 SV-2
System Functions	System functions, which are performed by a given information system, are those processes and computational tasks that effectively run the given system. The result of a System Function is an output.	SV-4 SV-5a

3.3. Architecture Interrogatives

A critical part of defining an architecture is answering what is known as the set of standard interrogatives, which are the set of “who, what, when, where, why, and how” questions that facilitate collection and usage of architecture-related data. DoDAF v2.0 provides a means of answering these interrogatives through its viewpoints and associated models, and the DoDAF Meta-Model Data Groups, as the major parts of the DoDAF Conceptual Data Model (CDM).

The table below provides a simple matrix that presents the DoDAF v2.0 Viewpoints and Models as they relate to the DoDAF Meta-Model Groups, and how these viewpoints, models, and groups answer the standard interrogatives. When architecture is required to support decision-making, the matrix is useful in both data collection and deciding how to best represent the data in DoDAF-described Models that are appropriate to the purpose for which the architecture is created. This table has been modified from the version in DoDAF v2.0 to address only those Viewpoints and Models discussed in this guide.

Table 5. DoDAF Interrogatives

	What Definition	Why Motivation	When Transition	Where Location	Who Mechanism	How Function
Viewpoint	AV, DIV	AV, CV, OV,	CV, OV, SV,	OV, SV, SvcV	OV	OV, SV, SvcV
View	AV-2, DIV-2, DIV-3	StdV, SV, SvcV	SvcV	OV-2, SV-1,	OV-2, OV-4, SV- 1, SvcV-1	OV-5a, OV-5b,
Element	Information and Data	AV-1, CV-1, OV- 1	CV-2, CV-4,	SvcV-1, SV-2,	Performer	OV-6abc, SV- 4, SV-10abc, SvcV- 10abc

Part III – Style



4. Design Principles

To assert compliance and conform to specific style requirements, architects must ensure that their architecture adheres to a number of general design principles. These design principles span across all architecture models respective to the defined scope boundaries. The following table identifies and defines these as such.

Table 6. General Architecture Design Principles

Principle	Definition
Accurate	The architecture model content is factually correct and faithfully represents the intended subject.
Complete	The architecture model contains all required elements/attributes necessary to meet the model's purpose.
Consistent	The architecture model conforms to an established style (theme), scope (boundary), and perspective (detail).
Integrated	The architecture model contains elements that are synchronized and non-conflicting across the viewpoints.
Aligned	The architecture model is designed and asserted with respect to the bigger (or smaller) picture.
Relevant	The architecture model is useful, necessary, and informative to the intended consumer.

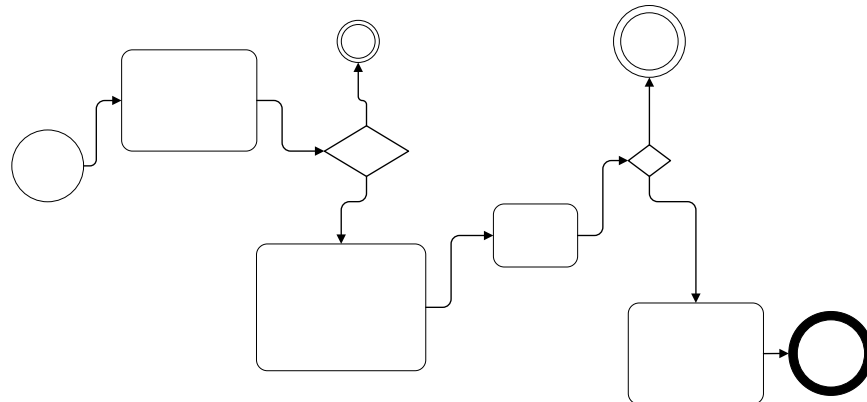
5. Theme

An architecture *theme* is a collection of standards that specify how an architecture is represented. The theme of an architecture must be established prior to the actual design and build phases. Conformance to an agreed-upon theme yields consistency throughout the entirety of an architecture. In determining a theme, the following sub-sections contain considerable style components and best practices.

5.1. Notation

A notation is a series or system of written symbols used to represent people, activities, decisions, or events. Notation provides the architect with a standardized set of tools to display architecture elements using common shapes, icons, logos, or graphics. The Business Process Model & Notation (BPMN) is a great example of a standard notation. BPMN is commonly used in process modeling for its intuitiveness. An example of this notation might look something like this:

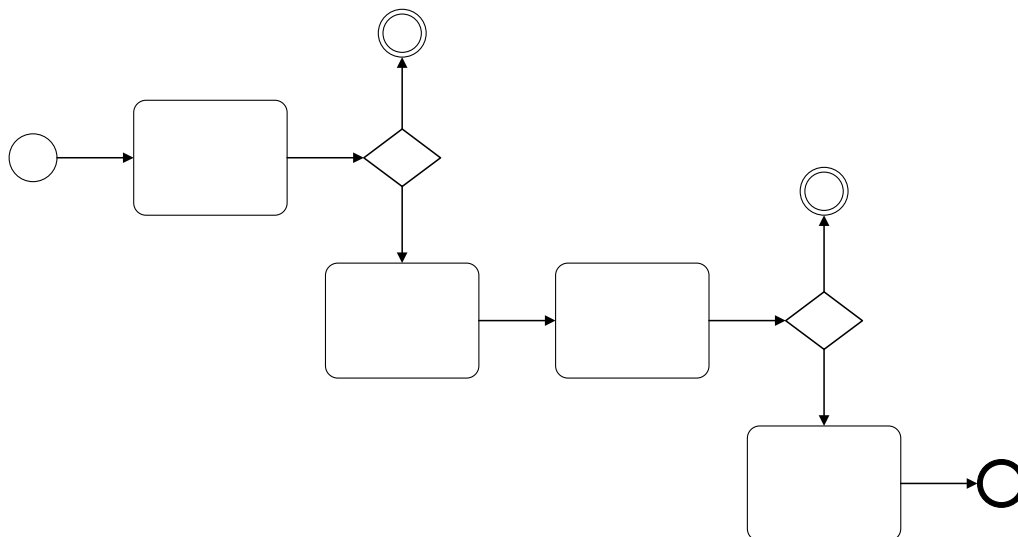
Figure 1. Example of Business Process Model Notation (BPMN)



5.2. Proportion

Although the model above is technically compliant with the BPMN notation, it proves a bit difficult to follow. Applying a sense of proportion to any architecture model using respectively sized and spaced elements (e.g. using the golden ratio) turns otherwise chaotic assemblies of elements into better organized and naturally attractive renderings.

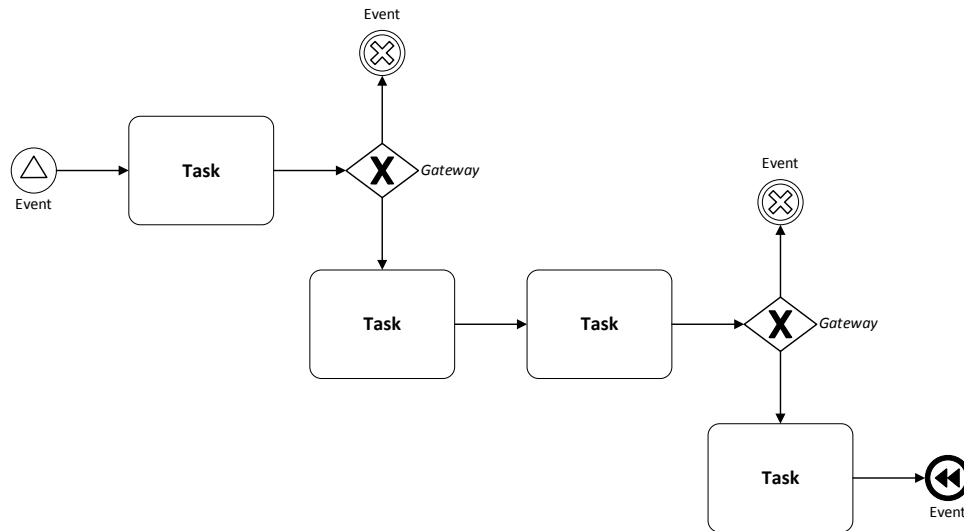
Figure 2. An example demonstrating good use of Proportion



5.3. Font

The use of conformed font types and sizes (with respect to the aforementioned proportion rules) reinforces the chosen architecture theme and adds a layer of specificity or clarity to the architecture.

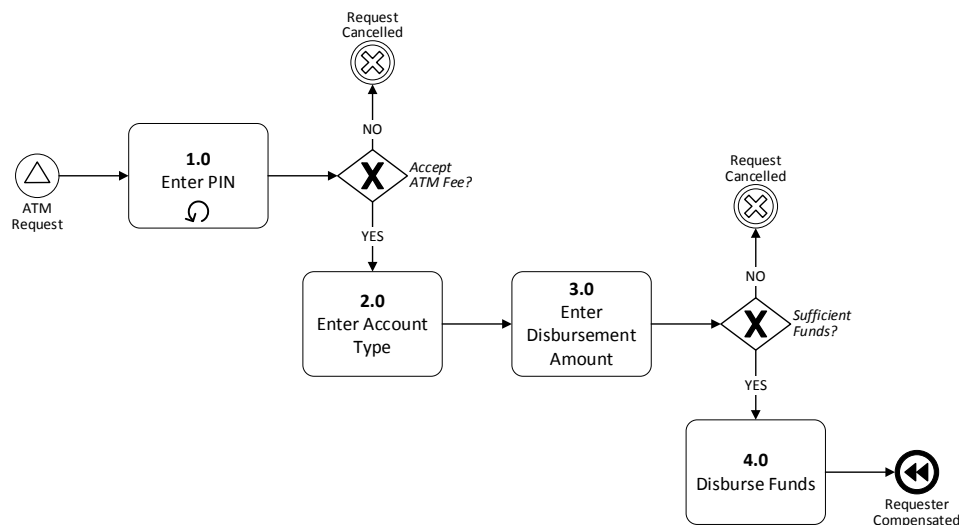
Figure 3. An example demonstrating good use of Font



5.4. Convention

Convention defines the narrative standards associated with the labeling and identification of architecture elements. A typical naming convention for a BPMN-compliant “Task” box requires the architect to use action verbs to indicate an action being performed. In the event of an ATM withdrawal (observed below), all tasks are labeled with a number and a specific action verb to indicate when the actions are occurring in the sequence.

Figure 4. An example demonstrating good application of Conventions



5.5. Color

Choosing a color scheme is a crucial step in the determination of an architecture theme. Although there is no right or wrong answer when selecting a color scheme, it is commonplace to use colors in accordance with *color theory*. Color theory provides guidance around naturally occurring colors and/or color combinations (e.g. analogous colors and complementary colors). When used in accordance with color theory, an architecture will radiate with *color harmony* and

color context throughout its underlying design. This will overlay a sense of embedded behavior in the architecture and promote consistency with the chosen theme.

Figure 5. An example demonstrating good application of color scheme

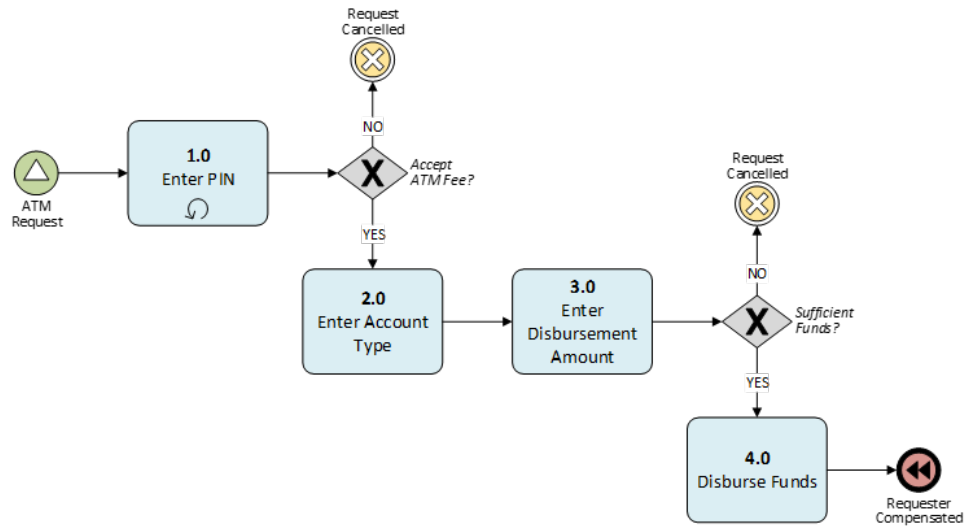




Table 7. RGB Codes


Blue	Light	Mid	Dark	Darker	Darkest
	R: 219	183	146	49	32
	G: 238	221	205	133	88
	B: 243	232	220	155	103
Orange	Light	Mid	Dark	Darker	Darkest
	R: 252	251	249	234	156
	G: 235	215	196	112	74
	B: 221	187	153	13	9
Green	Light	Mid	Dark	Darker	Darkest
	R: 235	215	196	120	80
	G: 241	227	214	148	99
	B: 223	191	160	64	42
Red	Light	Mid	Dark	Darker	Darkest
	R: 242	229	217	146	97
	G: 220	185	149	57	38
	B: 218	181	143	49	33
Yellow	Light	Mid	Dark	Darker	Darkest
	R: 255	254	255	191	127
	G: 242	229	217	144	96
	B: 204	153	101	0	0
Purple	Light	Mid	Dark	Darker	Darkest
	R: 238	221	204	126	84
	G: 234	214	194	100	66
	B: 242	229	217	158	106
Gray	Light	Mid	Dark	Darker	Darkest
	R: 242	216	191	165	127
	G: 242	216	191	165	127
	B: 242	216	191	165	127


6. Model Templates


 0-Model_Template_G
eneric.vsdX

 1-Model_Template_S
trategy_Domain.vsdX

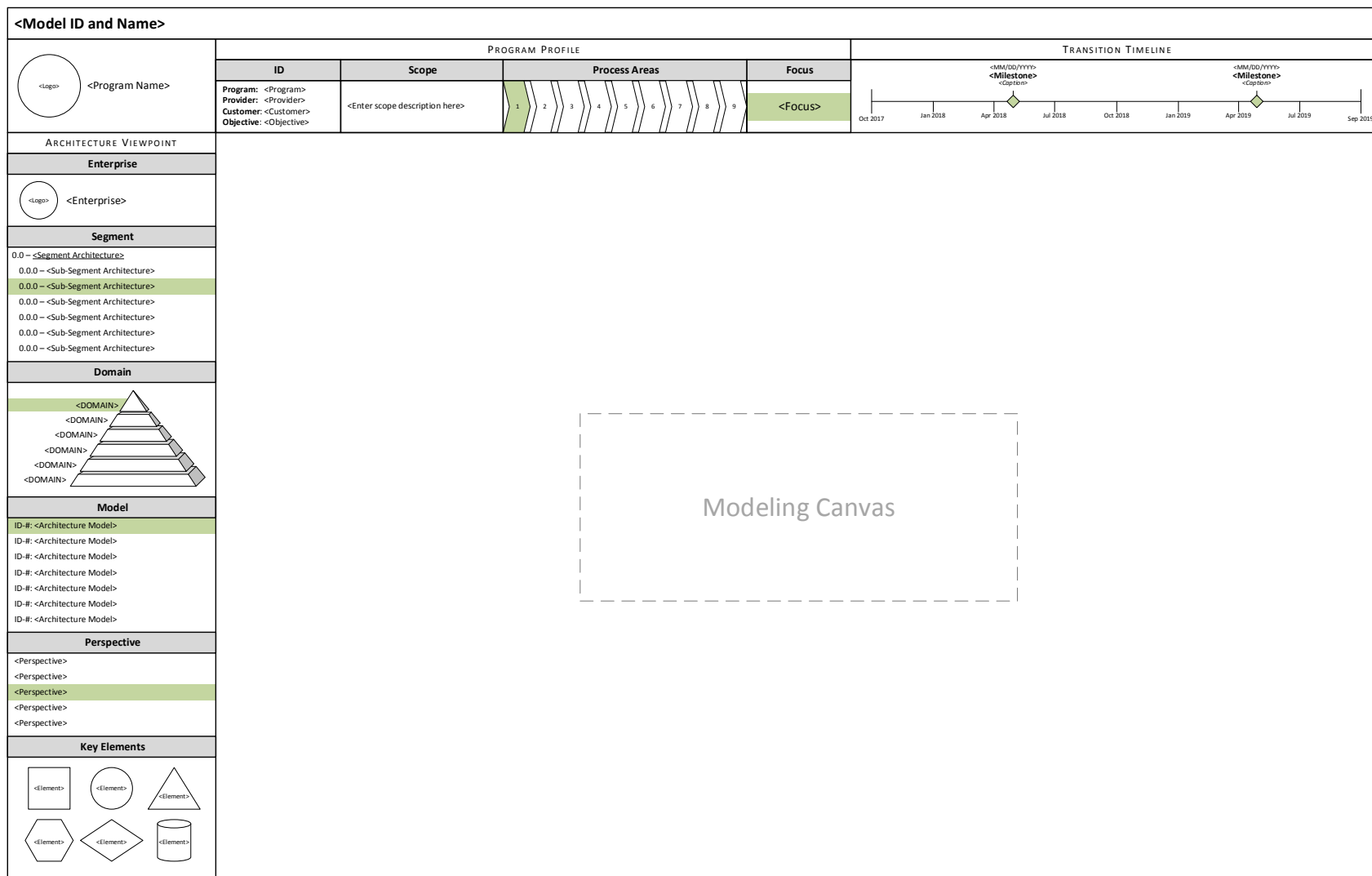
 2-Model_Template_B
usiness_Domain.vsdX

 3-Model_Template_D
ata_Domain.vsdX

 4-Model_Template_A
pplication_Domain.vsdX

 5-Model_Template_In
frastructure_Domain.vsdX

 6-Model_Template_S
ecurity_Domain.vsdX



Part IV – Design



7. All Viewpoint (AV)

7.1. AV-1: Overview and Summary Information

7.1.1. Description

The overview and summary information contained within the AV-1 provides executive-level summary information in a consistent form that allows quick reference and comparison between Architectural Descriptions. The written content of the AV-1 describes the concepts contained in the pictorial representation of the OV-1.

The AV-1 frames the context for the Architectural Description. The AV-1 includes assumptions, constraints, and limitations that may affect high-level decisions relating to an architecture-based work program. It should contain sufficient information to enable a reader to select a single Architectural Description from among many to read in more detail. The AV-1 serves two additional purposes:

1. In the initial phases of architecture development, it serves as a planning guide
2. When the architecture is built, the AV-1 provides summary information concerning who, what, when, why, and how of the plan as well as a navigation aid to the models that have been created

The usage of the AV-1 is to:

- Scope the architecture effort
- Provide context to the architecture effort
- Define the architecture effort
- Summarize the findings from the architecture effort
- Assist search within an architecture repository

7.1.2. Checklist

Table 8. AV-1 Checklist

ID	Type	Priority	Criteria	Checklist
AV-1-01	General	Must	The AV-1 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
AV-1-02	General	Must	The AV-1 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
AV-1-03	General	Must	The AV-1 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
AV-1-04	General	Must	The AV-1 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
AV-1-05	General	Must	The AV-1 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>

ID	Type	Priority	Criteria	Checklist
AV-1-06	General	Must	The AV-1 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
AV-1-07	Design	Must	The AV-1 answers the six interrogative questions: “Architecture Identification (<i>What</i>),” “Architecture Justification (<i>Why</i>),” “Architecture Transition (<i>When</i>),” “Architecture Location (<i>Where</i>),” “Architecture Instantiation (<i>How</i>),” and “Architecture Distribution (<i>Who</i>).”	<input type="checkbox"/>
AV-1-08	Design	Must	The AV-1 effectively defines and describes the scope and boundaries of the architecture.	<input type="checkbox"/>
AV-1-09	Design	Must	The AV-1 effectively defines and describes the baseline perspective (i.e. level of detail) of the architecture.	<input type="checkbox"/>
AV-1-10	Design	Could	The AV-1 shall include: assumptions, constraints, approval authority, completion date, level of effort and costs (projected and actual) required to develop the architecture.	<input type="checkbox"/>
AV-1-11	Design	Could	The AV-1 shall include: a list of tools used for architecture model creation along with the file names and formats for the Architectural Models, if appropriate.	<input type="checkbox"/>
AV-1-12	Design	Could	The AV-1 shall Include all findings resulting from architecture analysis consistent with the architecture effort.	<input type="checkbox"/>

7.2. AV-2: Integrated Dictionary

7.2.1. Description

The AV-2 presents all the metadata used in an architecture. An AV-2 presents all the data as a hierarchy, provides a text definition for each one and references the source of the element (e.g., DoDAF Meta-model, IDEAS, a published document or policy). An AV-2 shows elements from the DoDAF Meta-Model that have been described in the Architectural Description and new elements (i.e., not in the DM2) that have been introduced by the Architectural Description.

Because of the interrelationship among models and across architecture efforts, it is useful to define common terminology with common definitions (referred to as taxonomies) in the development of the models within the Architectural Description. These taxonomies can be used as building blocks for DoDAF-described Models and Fit-for-Purpose Views within the Architectural Description. The need for standard taxonomies derives from lessons learned from early DoD Architectural Description development issues as well as from federation pilots conducted within the Department. Federation of Architectural Descriptions was made much more difficult because of the use of different terminology to represent the same architectural data. Use of taxonomies to build models for the architecture has the following benefits over free-text labeling:

- Provides consistency across populated views, based on DoDAF-described Models.
- Provides consistency across Architectural Descriptions.
- Facilitates Architectural Description development, validation, maintenance, and reuse.
- Traces architectural data to authoritative data sources.

Architectural Descriptions can often introduce new terms - possibly because the architecture is covering new technology or business activities. The purpose of the AV-2 is to provide a means to explain the terms and abbreviations used in building the architecture and, as necessary, submit them for review and inclusion into authoritative vocabularies developed by COIs that are pertinent to the Architectural Description content.

7.2.2. Checklist

Table 9. AV-2 Checklist

ID	Type	Priority	Criteria	Checklist
AV-2-01	General	Must	The AV-2 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
AV-2-02	General	Must	The AV-2 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
AV-2-03	General	Must	The AV-2 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
AV-2-04	General	Must	The AV-2 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
AV-2-05	General	Must	The AV-2 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
AV-2-06	General	Must	The AV-2 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
AV-2-07	Design	Must	The AV-2 shall include definitions of all terms used in all architecture models. Each labeled item (e.g., icon, box, or connecting line) in the graphical representation shall have a corresponding entry in AV-2. Each item from a textual representation of an architectural model should also have a corresponding entry. Ensure the use of standard terms for architecture elements, such as those found in FEA Reference Models, the JCSFL, or any listed in Architecture Resources. Provide the definition and reference the source to ensure there is a mapping to standard definitions, where possible.	<input type="checkbox"/>
AV-2-08	Design	Must	The AV-2 shall include, as a minimum, the Document Block, Element Type, Element Name, Element Definition, Element Sub-type (if applicable), and Authoritative Source.	<input type="checkbox"/>
AV-2-09	Design	Must	Changes to any of the models must be reflected in the AV-2.	<input type="checkbox"/>
AV-2-10	Style	Recommended	Each architecture element name and definition shall comply with established conventions.	<input type="checkbox"/>
AV-2-11	Style	Recommended	The AV-2 shall make use of the relevant operational community vocabulary, to include Acronyms and Glossary (repository data, taxonomy, and metadata) and notes that have been included on any unique definitions.	<input type="checkbox"/>

8. Capability Viewpoint (CV)

8.1. CV-1: Capability Vision

8.1.1. Description

The CV-1 addresses the enterprise concerns associated with the overall vision for transformational endeavors. The purpose of a CV-1 is to provide a strategic context for the capabilities described in the Architectural Description. It also provides a high-level scope for the Architectural Description which is more general than the scenario-based scope defined in an OV-1. The intended usage is communication of the strategic vision regarding capability development.

8.1.2. Checklist

Table 10. CV-1 Checklist

ID	Type	Priority	Criteria	Checklist
CV-1-01	General	Must	The CV-1 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
CV-1-02	General	Must	The CV-1 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
CV-1-03	General	Must	The CV-1 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
CV-1-04	General	Must	The CV-1 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
CV-1-05	General	Must	The CV-1 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
CV-1-06	General	Must	The CV-1 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
CV-1-07	Design	Must	The vision for the Enterprise shall align with the capabilities identified in the Capability Taxonomy.	<input type="checkbox"/>

8.2. CV-2: Capability Taxonomy

8.2.1. Description

The CV-2 captures capability taxonomies. The model presents a hierarchy of capabilities. These capabilities may be presented in the context of a timeline - i.e., it can show the required capabilities for current and future states. The CV-2 specifies all the capabilities that are referenced throughout one or more architectures. In addition, it can be used as a source document for the development of high-level use cases and user requirements.

The intended usage of the CV-2 includes:

- Identification of capability requirements
- Capability planning (capability taxonomy)
- Codifying required capability elements

- Capability audit
- Capability gap analysis
- Source for the derivation of cohesive sets of user requirements
- Providing reference capabilities for architectures

8.2.2. Checklist

Table 11. CV-2 Checklist

ID	Type	Priority	Criteria	Checklist
CV-2-01	General	Must	The CV-2 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
CV-2-02	General	Must	The CV-2 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
CV-2-03	General	Must	The CV-2 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
CV-2-04	General	Must	The CV-2 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
CV-2-05	General	Must	The CV-2 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
CV-2-06	General	Must	The CV-2 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
CV-2-07	Design	Must	The Capability Taxonomy shall align with its associated Operational Activity Hierarchy.	<input type="checkbox"/>
CV-2-08	Design	Must	Each Capability must be defined in the AV-2.	<input type="checkbox"/>
CV-2-09	Design	Must	Capabilities shall map to the appropriate Operational Activities in the OV-5a.	<input type="checkbox"/>
CV-2-10	Style	Recommended	Parent Capabilities are decomposed into at least two, but not more than nine child Capabilities.	<input type="checkbox"/>

8.3. CV-3: Capability Phasing

8.3.1. Description

The CV-3 addresses the planned achievement of a capability at different points in time or during specific periods of time, i.e., capability phasing. The CV-3 supports the capability audit process and similar processes used across the different COIs by providing a method to identify gaps or duplication in capability provision. The CV-3 indicates capability increments, which should be associated with delivery milestones within acquisition projects (when the increments are associated with capability deliveries).

The intended usage of the CV-3 includes:

- Capability planning (capability phasing)
- Capability integration planning
- Capability gap analysis

8.3.2. Checklist

Table 12. CV-3 Checklist

ID	Type	Priority	Criteria	Checklist
CV-3-01	General	Must	The CV-3 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
CV-3-02	General	Must	The CV-3 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
CV-3-03	General	Must	The CV-3 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
CV-3-04	General	Must	The CV-3 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
CV-3-05	General	Must	The CV-3 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
CV-3-06	General	Must	The CV-3 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
CV-3-07	Design	Must	The Capability Phasing should align with the evolution of Systems or Services depicted in the SV- 8/SvcV-8.	<input type="checkbox"/>
CV-3-08	Style	Recommended	The Capability Phase shall be represented as an 'X' or a color code	<input type="checkbox"/>

8.4. CV-4: Capability Dependencies

8.4.1. Description

The CV-4 describes the dependencies between planned capabilities. It also defines logical groupings of capabilities.

The CV-4 is intended to provide a means of analyzing the dependencies between capabilities. The groupings of capabilities are logical, and the purpose of the groupings is to guide enterprise management. In particular, the dependencies and groupings may suggest specific interactions between acquisition projects to achieve the overall capability.

The intended usage of the CV-4 includes:

- Identification of capability dependencies

- Capability management (impact analysis for options, disposal etc.)

8.4.2. Checklist

Table 13. CV-4 Checklist

ID	Type	Priority	Criteria	Checklist
CV-4-01	General	Must	The CV-4 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
CV-4-02	General	Must	The CV-4 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
CV-4-03	General	Must	The CV-4 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
CV-4-04	General	Must	The CV-4 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
CV-4-05	General	Must	The CV-4 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
CV-4-06	General	Must	The CV-4 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
CV-4-07	Design	Must	Capability Dependencies should align with the capabilities listed in the Capability Taxonomy.	<input type="checkbox"/>
CV-4-08	Style	Recommended	Capability dependency lines shall indicate the relationships which asserts that a capability ('to' capability) is dependent on another ('from' capability) capability. The line may be a dashed line or color-coded to address a specific type of dependency.	<input type="checkbox"/>

8.5. CV-5: Capability to Organizational Development Mapping

8.5.1. Description

The CV-5 addresses the fulfillment of capability requirements. This model shows the planned capability deployment and interconnection for a particular phase. It should also provide a more detailed dependency analysis than is possible using the CV-3 Capability Phasing model. The CV-5 is used to support the capability management process and assist the planning of fielding.

The intended usage of the CV-5 includes:

- Fielding planning
- Capability integration planning
- Capability options analysis
- Capability redundancy/overlap/gap analysis

- Identification of deployment level shortfalls

8.5.2. Checklist

Table 14. CV-5 Checklist

ID	Type	Priority	Criteria	Checklist
CV-5-01	General	Must	The CV-5 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
CV-5-02	General	Must	The CV-5 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
CV-5-03	General	Must	The CV-5 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
CV-5-04	General	Must	The CV-5 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
CV-5-05	General	Must	The CV-5 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
CV-5-06	General	Must	The CV-5 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
CV-5-07	Design	Must	Organizations with specific capabilities should be aligned with Performer in the OV-2 and the System/Service depicted in the SV-1/SvcV-1.	<input type="checkbox"/>
CV-5-08	Design	Must	An Organizational Unit from the OV-4 that currently possesses or will have a specified capability shall be listed in the CV-5 matrix. Description of Organizational Units shall include name, Organization type and its purpose.	<input type="checkbox"/>
CV-5-09	Design	Must	The Capability Timeframe shall indicate when an Organizational Unit will have a specified capability. It is represented as a window of time, such as 2010-2020.	<input type="checkbox"/>
CV-5-10	Design	Must	The Capability Resource may be a System, Service, Material, etc. deployed to an Organizational Unit that supports a capability during an established timeframe or phase. Multiple resources may be depicted in the model at the intersection between capabilities and Organizational Units	<input type="checkbox"/>

8.6. CV-6: Capability to Operational Activities Mapping

8.6.1. Description

The CV-6 describes the mapping between the capabilities required and the activities that enable those capabilities.

It is important to ensure that the operational activity matches the required capability. The CV-6 DoDAF-described Model provides a bridge between a capability analyzed using CVs and operational activities analyzed using OVs. Specifically, it identifies how operational activities can be performed using various available capability elements. It is similar in function to the SV-5a Operational Activity to Systems Function Traceability Matrix. The capability to activity mappings may

include both situations where activities fully satisfy the desired capability and those where the activity only partially meets the capability requirement.

The intended usage of the CV-6 includes:

- Tracing capability requirements to operational activities
- Capability audit

8.6.2. Checklist

Table 15. CV-6 Checklist

ID	Type	Priority	Criteria	Checklist
CV-6-01	General	Must	The CV-6 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
CV-6-02	General	Must	The CV-6 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
CV-6-03	General	Must	The CV-6 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
CV-6-04	General	Must	The CV-6 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
CV-6-05	General	Must	The CV-6 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
CV-6-06	General	Must	The CV-6 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
CV-6-07	Design	Must	Capabilities listed in the CV-6 must be aligned to the activities in the OV-5a.	<input type="checkbox"/>
CV-6-08	Style	Recommended	The intersection between a capability and an operational activity may be annotated with an 'X' to indicate a relationship or it may have a special annotation (i.e., Phase) to meet stakeholder needs.	<input type="checkbox"/>

8.7. CV-7: Capability to Services Mapping

8.7.1. Description

The CV-7 describes the mapping between the capabilities required and the services that enable those capabilities. It is important to ensure that the services match the required capability. The CV-7 provides a bridge between capability analyzed using CVs and services analyzed using SvcVs. Specifically, it identifies how services can be performed using various available capability elements. It is similar in function to the SV-5a which maps system functions to operational activities. The capability to service mappings may include both situations where a service fully satisfies the desired capability and those where the service only partially meets the capability requirement.

The intended usage of the CV-7 includes:

- Tracing capability requirements to services
- Capability audit

8.7.2. Checklist

Table 16. CV-7 Checklist

ID	Type	Priority	Criteria	Checklist
CV-7-01	General	Must	The CV-7 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
CV-7-02	General	Must	The CV-7 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
CV-7-03	General	Must	The CV-7 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
CV-7-04	General	Must	The CV-7 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
CV-7-05	General	Must	The CV-7 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
CV-7-06	General	Must	The CV-7 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>

9. Data and Information Viewpoint (DIV)

9.1. DIV-1: Conceptual Data Model

9.1.1. Description

The DIV-1, a new DoDAF-described Model in DoDAF v2.0, addresses the information concepts at a high-level on an operational architecture.

The DIV-1 is used to document the business information requirements and structural business process rules of the architecture. It describes the information that is associated with the information of the architecture. Included are information items, their attributes or characteristics, and their interrelationships.

The intended usage of the DIV-1 includes:

- Information requirements
- Information hierarchy

9.1.2. Checklist

Table 17. DIV-1 Checklist

ID	Type	Priority	Criteria	Checklist
----	------	----------	----------	-----------

DIV-1-01	General	Must	The DIV-1 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
DIV-1-02	General	Must	The DIV-1 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
DIV-1-03	General	Must	The DIV-1 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
DIV-1-04	General	Must	The DIV-1 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
DIV-1-05	General	Must	The DIV-1 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
DIV-1-06	General	Must	The DIV-1 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>

9.2. DIV-2: Logical Data Model

9.2.1. Description

The DIV-2 allows analysis of an architecture's data definition aspect, without consideration of implementation specific or product specific issues.

Another purpose is to provide a common dictionary of data definitions to consistently express models wherever logical-level data elements are included in the descriptions. Data definitions in other models include:

- Data described in a DIV-2 may be related to information in an OV-1 High Level Operational Concept Graphic or an Activity Resource (where the Resource is Data) flow object in an OV-5b Operational Activity Model. This relation may be a simple subtype, where the Data is a proceduralized (structured) way of describing something. Recall that Information describes something. Alternatively, the relation may be complex using Information and Data whole-part (and overlap) relationships.
- The DIV-2 information entities and elements can be constrained and validated by the capture of business requirements in the OV-6a Operational Rules Model. The information entities and elements modeled in the DIV-2 also capture the information content of messages that connect life-lines in an OV-6c Event-Trace Description.
- The DIV-2 may capture elements required due to Standards in the StdV-1 Standards Profile or StdV-2 Standards Forecast.

9.2.2. Checklist

Table 18. DIV-2 Checklist

ID	Type	Priority	Criteria	Checklist
DIV-2-01	General	Must	The DIV-2 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
DIV-2-02	General	Must	The DIV-2 contains all required elements/attributes necessary to meet the model's	<input type="checkbox"/>

			purpose.	
DIV-2-03	General	Must	The DIV-2 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
DIV-2-04	General	Must	The DIV-2 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
DIV-2-05	General	Must	The DIV-2 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
DIV-2-06	General	Must	The DIV-2 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
DIV-2-07	Design	Must	The DIV-2 shall depict the logical operational and business data requirements and rules associated with the architecture domain, mission, or business as a set of entities, with their associated Attributes and Relationships.	<input type="checkbox"/>
DIV-2-08	Design	Must	The DIV-2 Entity shall correlate to the Operational Resource Flows in the OV-3 and OV-5b Inputs, Outputs and Controls.	<input type="checkbox"/>
DIV-2-09	Design	Must	Technical standards in the STDV-1 shall apply to modeling techniques in the DIV-2.	<input type="checkbox"/>
DIV-2-10	Design	Must	Each Entity shall have a Primary Key that is minimal (i.e., removal of any attribute that renders the key not unique) and shall not contain null components.	<input type="checkbox"/>
DIV-2-11	Design	Must	Each Primary Key shall use the natural key when one is available.	<input type="checkbox"/>
DIV-2-12	Design	Must	Each Entity Attribute must have a Primary Key, represent a distinct piece of information, and be functionally dependent on a primary key. They shall not be compounded nor be derived from other entities.	<input type="checkbox"/>
DIV-2-13	Design	Must	All subtypes shall have the same primary key as the supertype (Role-based names are allowed).	<input type="checkbox"/>
DIV-2-14	Design	Must	All subtypes shall have one or more Attributes and/or one or more Relationships to differentiate them from the supertype and the other subtypes.	<input type="checkbox"/>
DIV-2-15	Design	Must	All child Entities shall have one or more non-key Attributes and/or one or more Relationships that differentiate them from their parent Entity.	<input type="checkbox"/>
DIV-2-16	Design	Must	Each Entity shall contain at least one Attribute.	<input type="checkbox"/>
DIV-2-17	Design	Must	Relationship lines on all DIV-2 diagrams shall be displayed properly and not hidden.	<input type="checkbox"/>
DIV-2-18	Design	Must	Logical IDEF1X Categorizations shall have a Name or Discriminator	<input type="checkbox"/>
DIV-2-19	Design	Must	If provided by a Performer linked to their OV-6c Data Objects, DESs shall be linked to Attributes within DIV-2 Entities via Data Elements.	<input type="checkbox"/>
DIV-2-20	Design	Must	Any Data Elements required by the functional area for System certification shall be	<input type="checkbox"/>

			identified.	
DIV-2-21	Style	Recommended	The associated tags of all Relationship lines shall be positioned properly on the diagram.	<input type="checkbox"/>
DIV-2-22	Style	Recommended	All Attribute names must contain at least one Attribute, be unique, be a singular noun or noun with adjective- phrase and be separated with underscores and no special characters. They should not use the possessive form or use names of organizations, computer or information systems, directives, forms, or reports.	<input type="checkbox"/>
DIV-2-23	Style	Recommended	All Entity names will be singular and use upper case alphanumeric characters and will refer to the class of the information and not the occurrence of the class. They will not utilize any abbreviations or acronyms.	<input type="checkbox"/>

9.3. DIV-3: Physical Data Model

9.3.1. Description

The DIV-3 defines the structure of the various kinds of system or service data that are utilized by the systems or services in the Architectural Description. The Physical Schema is one of the models closest to actual system design in DoDAF. DIV-3 is used to describe how the information represented in the DIV-2 Logical Data Model is implemented.

While the mapping between the logical and physical data models is relatively straightforward, the relationship between the components of each model (e.g., entity types in the logical model versus relational tables in the physical model) is frequently one-to-many or many-to-many.

The intended usage of the DIV-3 includes:

- Specifying the system/service data elements exchanged between systems and/or services, thus reducing the risk of interoperability errors
- Definition of physical data structure
- Providing as much detail as possible on data elements exchanged between systems, thus reducing the risk of interoperability problems
- Providing data structures for use in the system design process, if necessary
- Providing a common dictionary of data implementation elements (e.g., tables and records in a relational database schema) to consistently express models wherever physical-level data elements are included in the descriptions
- Providing as much detail as possible on the system or service data elements exchanged between systems, thus reducing the risk of interfacing errors
- Providing system and service data structures for use in the system and service design process, if necessary

9.3.2. Checklist

Table 19. DIV-3 Checklist

ID	Type	Priority	Criteria	Checklist
DIV-3-01	General	Must	The DIV-3 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
DIV-3-02	General	Must	The DIV-3 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
DIV-3-03	General	Must	The DIV-3 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
DIV-3-04	General	Must	The DIV-3 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
DIV-3-05	General	Must	The DIV-3 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
DIV-3-06	General	Must	The DIV-3 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
DIV-3-07	Design	Must	The DIV-3 shall describe the physical implementation of the information requirements represented in the DIV-2.	<input type="checkbox"/>
DIV-3-08	Design	Must	Entities shall represent System/Service Data Flows depicted in the SV-4/SvcV-4.	<input type="checkbox"/>
DIV-3-09	Design	Must	All schemas, database instances, table spaces and databases shall be represented in the model.	<input type="checkbox"/>
DIV-3-10	Design	Must	Relationships among schemas, database instances, table spaces and databases shall be represented in the model.	<input type="checkbox"/>
DIV-3-11	Design	Must	The physical organization of the data shall be consistent with the logical data model (DIV-2).	<input type="checkbox"/>

10. Operational Viewpoint (OV)

10.1. OV-1: High-Level Operational Concept Graphic

10.1.1. Description

The OV-1 describes a mission, class of mission, or scenario. It shows the main operational concepts and interesting or unique aspects of operations. It describes the interactions between the subject architecture and its environment, and between the architecture and external systems. The OV-1 is the pictorial representation of the written content of the AV-1 Overview and Summary Information. Graphics alone are not sufficient for capturing the necessary architectural data.

The OV-1 provides a graphical depiction of what the architecture is about and an idea of the players and operations involved. An OV-1 can be used to orient and focus detailed discussions. Its main use is to aid human communication, and it is intended for presentation to high-level decision-makers.

The intended usage of the OV-1 includes:

- Putting an operational situation or scenario into context
- Providing a tool for discussion and presentation; for example, aids industry engagement in acquisition
- Providing an aggregate illustration of the details within the published high-level organization of more detailed information in published architectures

10.1.2. Checklist

Table 20. OV-1 Checklist

ID	Type	Priority	Criteria	Checklist
OV-1-01	General	Must	The OV-1 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
OV-1-02	General	Must	The OV-1 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
OV-1-03	General	Must	The OV-1 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
OV-1-04	General	Must	The OV-1 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
OV-1-05	General	Must	The OV-1 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
OV-1-06	General	Must	The OV-1 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
OV-1-07	Design	Must	The OV-1 shall include details regarding the business processes or missions, high-level operations, organizations and geographical distribution of assets.	<input type="checkbox"/>
OV-1-08	Design	Must	The OV-1 shall "tell the story" of the operation being described. It should highlight key capability requirements. This includes the key interactions between performers and for the flow of relevant information. The readers should get a sense of a sequence of events depicted in the OV-1 to aid in understanding the domain. The required explanatory text should assist in this understanding.	<input type="checkbox"/>
OV-1-09	Design	Must	Organizations, Organization types, and/or roles shall be traceable to Operational Performers in the OV-2.	<input type="checkbox"/>
OV-1-10	Design	Must	Relationships in the OV-1 shall trace to Needlines in OV-2.	<input type="checkbox"/>
OV-1-11	Style	Recommended	The OV-1 shall consist of one or more (possibly animated) graphic images, or motion videos.	<input type="checkbox"/>

OV-1-12	Design	Must	The OV-1 shall convey what the architecture is about by describing the interactions between the subject architecture and its environment, and between the architecture and external Systems/Services.	<input type="checkbox"/>
OV-1-13	Technique	Optional	The depiction of systems and/or services may be included to convey the concept graphically to the customer of the OV model.	<input type="checkbox"/>
OV-1-14	Technique	Optional	A legend may be provided if necessary to enhance understanding of the graphic.	<input type="checkbox"/>
OV-1-15	Design	Must	The OV-1 shall frame the operational concept (what happens, who does what, in what order, to accomplish what goal) and highlight interactions to the environment and other external capabilities.	<input type="checkbox"/>
OV-1-16	Design	Must	The scope of the OV-1 shall be consistent with the AV-1.	<input type="checkbox"/>

10.2. OV-2: Operational Resource Flow Description

10.2.1. Description

The OV-2 DoDAF-described Model applies the context of the operational capability to a community of anticipated users. The primary purpose of the OV-2 is to define capability requirements within an operational context. The OV-2 may also be used to express a capability boundary.

New to DoDAF v2.0, the OV-2 can be used to show flows of funding, personnel and material in addition to information. A specific application of the OV-2 is to describe a logical pattern of resource (information, funding, personnel, or material) flows. The logical pattern need not correspond to specific organizations, systems or locations, allowing Resource Flows to be established without prescribing the way that the Resource Flows are handled and without prescribing solutions.

The intended usage of the OV-2 includes:

- Definition of operational concepts
- Elaboration of capability requirements
- Definition of collaboration needs
- Applying a local context to a capability
- Problem space definition
- Operational planning
- Supply chain analysis
- Allocation of activities to resources

10.2.2. Checklist

Table 21. OV-2 Checklist

ID	Type	Priority	Criteria	Checklist
OV-2-01	General	Must	The OV-2 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
OV-2-02	General	Must	The OV-2 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
OV-2-03	General	Must	The OV-2 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
OV-2-04	General	Must	The OV-2 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
OV-2-05	General	Must	The OV-2 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
OV-2-06	General	Must	The OV-2 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
OV-2-07	Design	Must	The OV-2 shall have a textual description of the operation depicted in the graphic provided on the graphic or attached.	<input type="checkbox"/>
OV-2-08	Design	Must	Each Performer shall be reflected in at least one OV-2 Model diagram.	<input type="checkbox"/>
OV-2-09	Design	Must	A single Needline shall be used to represent the interactions of all Operational Resource Flows that have a common source and destination between a pair of Performers.	<input type="checkbox"/>
OV-2-10	Design	Must	The Performers in the OV-2 shall be in alignment with the Organizational Entities/Roles depicted in the OV-4	<input type="checkbox"/>
OV-2-11	Design	Must	Organizations, Organization types, and/or roles depicted in OV-1 shall be traceable to Operational Performers in the OV-2.	<input type="checkbox"/>
OV-2-12	Design	Must	Relationships in the OV-1 must trace to Needlines in OV-2.	<input type="checkbox"/>
OV-2-13	Design	Must	Each Performer shall have a unique identifier.	<input type="checkbox"/>
OV-2-14	Design	Must	Each internal Performer shall have one or more leaf-level activities mapped to it.	<input type="checkbox"/>
OV-2-15	Design	Must	Each Needline in the model shall have a unique identifier or phrase that describes the type of Operational Resource Flow involved. The Needline shall represent the logical flow of resources between Performers, not the physical path the information will be exchanged over.	<input type="checkbox"/>
OV-2-16	Design	Must	Needlines shall represent Operational Resource Flow requirements between Performers not reporting relationships. Ensure the OV-2 shows the need to exchange resources between Organizations, Organization types, or roles.	<input type="checkbox"/>

OV-2-17	Design	Must	If multiple OV-2 model diagrams are developed, there shall be only Performers that Interface with the center Performer shown on each diagram.	<input type="checkbox"/>
OV-2-18	Design	Must	Every Needline shall have at least one Operational Resource Flow assigned. All Performers shall have at least one Needline.	<input type="checkbox"/>
OV-2-19	Design	Must	All existing Needlines shall be used on at least one/but no more than one OV-2 Model diagrams.	<input type="checkbox"/>
OV-2-20	Design	Must	A net-centric OV-2 shall depict Service Functionality Providers, Service Consumers, and unanticipated Users.	<input type="checkbox"/>
OV-2-21	Design	Must	The OV-2 shall capture the appropriate organizations and their Operational Resource Flow requirements as shown in the OV-1.	<input type="checkbox"/>
OV-2-22	Design	Must	Performers and Needlines in the OV-2 intended to be automated, can trace to elements in the SV-1 and SvcV-1. This can be a one-to-many relationship.	<input type="checkbox"/>
OV-2-23	Design	Must	OV-2 Performers shall match OV-6c Pools and Swimlanes developed in BPMN.	<input type="checkbox"/>
OV-2-24	Style	Recommended	When feasible, annotate Performers with Operational Activities from the OV-5a.	<input type="checkbox"/>
OV-2-25	Style	Recommended	There shall be at least one Performer-centric OV-2 diagram plus, if desired, an integrated Enterprise-level OV-2 representing the sum of all developed OV-2s.	<input type="checkbox"/>
OV-2-26	Style	Recommended	The names of Performer must be legible when printed.	<input type="checkbox"/>
OV-2-27	Style	Recommended	The number of Needline intersections shall be kept to a minimum.	<input type="checkbox"/>

10.3. OV-3: Operational Resource Flow Matrix

10.3.1. Description

The OV-3 addresses operational Resource Flows exchanged between Operational Activities and locations.

Resource Flows provide further detail of the interoperability requirements associated with the operational capability of interest. The focus is on Resource Flows that cross the capability boundary.

The intended usage of the OV-3 includes:

- Definition of interoperability requirements

10.3.2. Checklist

Table 22. OV-3 Checklist

ID	Type	Priority	Criteria	Checklist
----	------	----------	----------	-----------

OV-3-01	General	Must	The OV-3 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
OV-3-02	General	Must	The OV-3 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
OV-3-03	General	Must	The OV-3 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
OV-3-04	General	Must	The OV-3 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
OV-3-05	General	Must	The OV-3 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
OV-3-06	General	Must	The OV-3 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
OV-3-07	Design	Must	All fields in each column must be filled in.	<input type="checkbox"/>
OV-3-08	Design	Must	The "Operational Resource Flow Description" column shall contain the Operational Resource Flow definition.	<input type="checkbox"/>
OV-3-09	Design	Must	Each Operational Resource Flow shall include all the required attributes as specified in this guide.	<input type="checkbox"/>
OV-3-10	Design	Must	Each Operational Resource Flow Attribute value shall be drawn from the options provided in this guide. Exceptions shall be noted in the description of the matrix.	<input type="checkbox"/>
OV-3-11	Design	Must	Each Operational Resource Flow (information) shall be traceable to one or more Entities in the DIV-2.	<input type="checkbox"/>
OV-3-12	Design	Must	Operational Resource Flows in the OV-3 shall be related to the leaf-level Operational Activity (from the Operational Activity Model) that produces or consumes it.	<input type="checkbox"/>
OV-3-13	Design	Must	Needlines in the OV-2 shall map to one or more Operational Resource Flows in the OV-3.	<input type="checkbox"/>
OV-3-14	Design	Must	Operational Resource Flows must be related to an ICOM in the OV-5b.	<input type="checkbox"/>

10.4. OV-4: Organizational Relationships Chart

10.4.1. Description

The OV-4 shows organizational structures and interactions. The OV-4 exists in two forms; role-based (e.g., a typical brigade command structure) and actual (e.g., an organization chart for a department or agency).

A role-based OV-4 shows the possible relationships between organizational resources. The key relationship is composition, i.e., one organizational resource being part of a parent organization. In addition to this, the architect may show the roles each organizational resource has, and the interactions between those roles, i.e., the roles represent the functional aspects of organizational resources. There are no prescribed resource interactions in DoDAF v2.0: the

architect should select an appropriate interaction type from the DM2 or add a new one. Interactions illustrate the fundamental roles and management responsibilities, such as supervisory reporting, Command and Control (C2) relationships, collaboration and so on.

An actual OV-4 shows the structure of a real organization at a point in time, and is used to provide context to other parts of the architecture such as AV-1 and the CVs.

The intended usage of the role-based OV-4 includes:

- Organizational analysis
- Definition of human roles
- Operational analysis

10.4.2. Checklist

Table 23. OV-4 Checklist

ID	Type	Priority	Criteria	Checklist
OV-4-01	General	Must	The OV-4 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
OV-4-02	General	Must	The OV-4 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
OV-4-03	General	Must	The OV-4 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
OV-4-04	General	Must	The OV-4 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
OV-4-05	General	Must	The OV-4 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
OV-4-06	General	Must	The OV-4 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
OV-4-07	Design	Must	The OV-4 shall include a textual description of the organization depicted.	<input type="checkbox"/>
OV-4-08	Design	Must	The OV-4 shall depict the relationships among Organizations, Organizational types, and/or roles that are the key participants within the architecture.	<input type="checkbox"/>
OV-4-09	Design	Must	The Organizations, Organization types, and/or roles shall correspond to Performers in the OV- 2.	<input type="checkbox"/>
OV-4-10	Style	Recommended	The model may have a legend to explain any unique features, to include the use of color codes.	<input type="checkbox"/>
OV-4-11	Design	Must	Command relationships shall be depicted using a solid 'Black' line and Coordination relationships shall be depicted as a dashed 'Black' line.	<input type="checkbox"/>

10.5. OV-5a: Operational Activity Decomposition Tree

10.5.1. Description

The OV-5a and the OV-5b describe the operations that are normally conducted in the course of achieving a mission or a business goal. It describes operational activities (or tasks); Input/Output flows between activities, and to/from activities that are outside the scope of the Architectural Description.

The OV-5a and OV-5b describes the operational activities that are being conducted within the mission or scenario. The OV-5a and OV-5b can be used to:

- Clearly delineate lines of responsibility for activities when coupled with OV-2
- Uncover unnecessary Operational Activity redundancy
- Make decisions about streamlining, combining, or omitting activities
- Define or flag issues, opportunities, or operational activities and their interaction
- Information flows among the activities that need to be scrutinized further
- Provide a necessary foundation for depicting activity sequencing and timing in the OV-6a Operational Rules Model, the OV-6b State Transition Description, and the OV-6c Event-Trace Description

10.5.2. Checklist

Table 24. OV-5a Checklist

ID	Type	Priority	Criteria	Checklist
OV-5a-01	General	Must	The OV-5a content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
OV-5a-02	General	Must	The OV-5a contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
OV-5a-03	General	Must	The OV-5a conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
OV-5a-04	General	Must	The OV-5a contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
OV-5a-05	General	Must	The OV-5a is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
OV-5a-06	General	Must	The OV-5a is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
OV-5a-07	Design	Must	The OV-5a shall have a textual description.	<input type="checkbox"/>

OV-5a-08	Design	Must	Activity boxes shall be numbered sequentially, relative to position and match corresponding Activity numbering on the Activity Diagram. Activity numbers must be prefaced with a capital “A” and shown at the beginning of the Activity box label.	<input type="checkbox"/>
OV-5a-09	Style	Recommended	Parent Activities shall be decomposed into at least two, but not more than nine, child Activities. The optimum range is from 3 to 6 child activities for one parent activity.	<input type="checkbox"/>
OV-5a-10	Design	Must	Activities shall be decomposed to a level low enough to support the specified Capability.	<input type="checkbox"/>
OV-5a-11	Style	Recommended	Operational Activity shall be named as a Verb-Noun, using an active verb phrase (for example, Allocate Resource).	<input type="checkbox"/>
OV-5a-12	Design	Must	The OV-5a hierarchy must match that used when developing the OV-5b.	<input type="checkbox"/>

10.6. OV-5b: Operational Activity Model

10.6.1. Description

The OV-5b describes the operational and business portions of the activities associated with the Architectural Description, as well as the:

- Relationships or dependencies among the activities
- Resources exchanged between activities
- External interchanges (from/to business activities that are outside the scope of the model).

An Operational Activity is what work is required, specified independently of how it is carried out. To maintain this independence from implementation, logical activities and locations in OV-2 Operational Resource Flow Description are used to represent the structure which carries out the Operational Activities. Operational Activities are realized as System Functions (described in SV-4 Systems Functionality Description) or Service Functions (described in SvcV-4 Services Functionality Description) which are the “how” to the Operational Activities “what”, i.e., they are specified in terms of the resources that carry them out.

The intended usage of the OV-5a and OV-5b includes:

- Description of activities and workflows
- Requirements capture
- Definition of roles and responsibilities
- Support task analysis to determine training needs
- Problem space definition
- Operational planning
- Logistic support analysis

- Information flow analysis

10.6.2. Checklist

Table 25. OV-5b Checklist

ID	Type	Priority	Criteria	Checklist
OV-5b-01	General	Must	The OV-5b content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
OV-5b-02	General	Must	The OV-5b contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
OV-5b-03	General	Must	The OV-5b conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
OV-5b-04	General	Must	The OV-5b contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
OV-5b-05	General	Must	The OV-5b is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
OV-5b-06	General	Must	The OV-5b is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
OV-5b-07	Design	Must	Activities used in the OV-5b must be the same as those depicted in the OV-5a.	<input type="checkbox"/>
OV-5b-08	Design	Must	The Context-Level Diagram must include a description of the model purpose and viewpoint.	<input type="checkbox"/>
OV-5b-09	Design	Must	The Context-Level Diagram must include Operational Activities performed by external Operational Performers.	<input type="checkbox"/>
OV-5b-10	Style	Recommended	IDEF0 must be followed when developing the OV-5b.	<input type="checkbox"/>
OV-5b-11	Design	Must	Each Operational Activity shall have at least one (1) Input and one (1) Output; placeholder Activities excluded.	<input type="checkbox"/>
OV-5b-12	Design	Must	Each Activity shall have one or more Mechanism(s), unless Activity- Based Methodology is followed.	<input type="checkbox"/>
OV-5b-13	Design	Must	Each Activity shall have zero, one or more Controls, unless Activity- Based Methodology is followed.	<input type="checkbox"/>
OV-5b-14	Design	Must	Each ICOM definition shall be consistent with the definition of the Activity that produces or consumes it and is consistently decomposed with the Activity.	<input type="checkbox"/>
OV-5b-15	Style	Recommended	The number of Controls shall be limited to a maximum of 8 per Operational Activity. Controls shall be stair-stepped (tallest to the left) with labels placed to the top right side of the Control.	<input type="checkbox"/>
OV-5b-16	Style	Recommended	Mechanism shall be related to Operational Performers. There is no specified maximum number per Operational Activity. The Mechanisms shall normally be arranged in stair	<input type="checkbox"/>

			step fashion (tallest to the left) with labels placed at the bottom right side of the arrow.	
OV-5b-17	Design	Must	Input ICOMs cannot be represented as Outputs for the same Activity. Output ICOMs cannot be represented as Inputs for the same Activity.	<input type="checkbox"/>
OV-5b-18	Design	Must	ICOMs must be balanced. Input/output ICOMs on a parent Activity must be consistent with Inputs/Outputs on its child diagram and vice versa.	<input type="checkbox"/>
OV-5b-19	Design	Must	Each leaf-level Activity Input and Output must have a corresponding Resource Flow, with the same name, definition and linked to the same Operational Performer. Controls and Mechanisms are not traceable to Operational Resource Flows.	<input type="checkbox"/>
OV-5b-20	Design	Must	All ICOMs must be physically connected to a given Activity and that there should be minimal crossings of ICOM arrows on any given diagram.	<input type="checkbox"/>

10.7. OV-6a: Operational Rules Model

10.7.1. Description

An OV-6a specifies operational or business rules that are constraints on the way that business is done in the enterprise. At a top-level, rules should at least embody the concepts of operations defined in OV-1 High Level Operational Concept Graphic and provide guidelines for the development and definition of more detailed rules and behavioral definitions that should occur later in the Architectural definition process.

The intended usage of the OV-6a includes:

- Definition of doctrinally correct operational procedures
- Definition of business rules
- Identification of operational constraints

10.7.2. Checklist

Table 26. OV-6a Checklist

ID	Type	Priority	Criteria	Checklist
OV-6a-01	General	Must	The OV-6a content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
OV-6a-02	General	Must	The OV-6a contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
OV-6a-03	General	Must	The OV-6a conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
OV-6a-04	General	Must	The OV-6a contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>

OV-6a-05	General	Must	The OV-6a is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
OV-6a-06	General	Must	The OV-6a is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
OV-6a-07	Design	Must	A rule may describe conditions that constrain the execution an Operational Activity (OV-5b) in a specific way, or constrain the Organization or role authorized to execute an Operational Activity.	<input type="checkbox"/>
OV-6a-08			A Rule may describe guard conditions for an action in OV-6c.	<input type="checkbox"/>
OV-6a-09	Style	Recommended	The rules in OV-6a may reference the elements of DIV-2 to constrain their structure and validity.	<input type="checkbox"/>
OV-6a-10	Style	Recommended	A Rule should be readily understood by any business party and is always interpreted the same.	<input type="checkbox"/>
OV-6a-11	Style	Recommended	A Rule is atomic.	<input type="checkbox"/>
OV-6a-12	Style	Recommended	A Rule is unambiguous.	<input type="checkbox"/>
OV-6a-13	Style	Recommended	A Rule is in declarative form – no reference to how, where, when, or who.	<input type="checkbox"/>
OV-6a-19	Style	Recommended	A Rule is not procedural (use of “else” and “if”).	<input type="checkbox"/>
OV-6a-20	Design	Must	Rule constrains (or alternatively permits).	<input type="checkbox"/>
OV-6a-21	Style	Recommended	A Rule contains one of the key rule words such as “is,” “may,” “shall,” “no,” “not,” “shall,” “should,” “will,” or “only if.”	<input type="checkbox"/>
OV-6a-22	Design	Must	Words such as “can” are not used.	<input type="checkbox"/>
OV-6a-23	Style	Recommended	A Rule shall use standard terminology such as the common language from the data model.	<input type="checkbox"/>
OV-6a-24	Design	Must	Facts are explicitly expressed in the rule (no hidden facts or computations).	<input type="checkbox"/>
OV-6a-25	Style	Recommended	A Rule is written in “RuleSpeak” formal language.	<input type="checkbox"/>
OV-6a-26	Style	Recommended	The rule does not have a plural subject.	<input type="checkbox"/>
OV-6a-27	Style	Recommended	Plural elements in the sentence should be avoided.	<input type="checkbox"/>
OV-6a-28	Design	Must	A time element is not the subject.	<input type="checkbox"/>
OV-6a-29	Design	Must	A Rule has an explicit subject.	<input type="checkbox"/>
OV-6a-30	Style	Recommended	Computations are the subject of the rule.	<input type="checkbox"/>

OV-6a-31	Design	Must	All required fields shall be filled in properly from load.	<input type="checkbox"/>
OV-6a-32	Design	Must	All Action Assertion and Derivation Business Rules are linked to the appropriate process step or Gateway.	<input type="checkbox"/>
OV-6a-33	Design	Must	All Derivation rules are linked by a Data Object, Process, or Gateway.	<input type="checkbox"/>
OV-6a-34	Design	Must	All Structural Assertion rules shall be linked to a Data Element.	<input type="checkbox"/>
OV-6a-35	Design	Must	All rules with a source type of “compliance requirement” shall have at least one requirement source listed.	<input type="checkbox"/>
OV-6a-36	Design	Must	All rules shall have one Rule category: Action Assertion, Structural Assertion, or Derivation.	<input type="checkbox"/>
OV-6a-37	Design	Must	All rules shall have one valid source type: compliance requirement, derived requirement, or process.	<input type="checkbox"/>
OV-6a-38	Design	Must	All rules shall have unique rule number associated with it.	<input type="checkbox"/>
OV-6a-39	Design	Must	All rules shall have one valid Business Rule level: automated, operational, and conceptual.	<input type="checkbox"/>

10.8. OV-6b: State Transition Description

10.8.1. Description

The OV-6b is a graphical method of describing how an Operational Activity responds to various events by changing its state. The diagram represents the sets of events to which the Activities respond (by taking an action to move to a new state) as a function of its current state. Each transition specifies an event and an action.

An OV-6b can be used to describe the detailed sequencing of activities or work flow in the business process. The OV-6b is particularly useful for describing critical sequencing of behaviors and timing of operational activities that cannot be adequately described in the OV-5b Operational Activity Model. The OV-6b relates events and states. A change of state is called a transition. Actions may be associated with a given state or with the transition between states in response to stimuli (e.g., triggers and events).

The intended usage of the OV-6b includes:

- Analysis of business events
- Behavioral analysis
- Identification of constraints

10.8.2. Checklist

Table 27. OV-6b Checklist

ID	Type	Priority	Criteria	Checklist
OV-6b-01	General	Must	The OV-6b content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
OV-6b-02	General	Must	The OV-6b contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
OV-6b-03	General	Must	The OV-6b conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
OV-6b-04	General	Must	The OV-6b contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
OV-6b-05	General	Must	The OV-6b is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
OV-6b-06	General	Must	The OV-6b is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
OV-6b-07	Design	Must	The state diagram must account for all behavior of objects, impacting events, actions and associated transitions, and resulting states being considered.	<input type="checkbox"/>

10.9. OV-6c: Event-Trace Description

10.9.1. Description

The OV-6c provides a time-ordered examination of the Resource Flows as a result of a particular scenario. Each event-trace diagram should have an accompanying description that defines the scenario or situation. Operational Event/Trace Descriptions, sometimes called sequence diagrams, event scenarios, or timing diagrams, allow the tracing of actions in a scenario or critical sequence of events. The OV-6c can be used by itself or in conjunction with an OV-6b State Transition Description to describe the dynamic behavior of activities.

The intended usage of the OV-6c includes:

- Analysis of operational events
- Behavioral analysis
- Identification of non-functional user requirements
- Operational test scenarios

10.9.2. Checklist

Table 28. OV-6c Checklist

ID	Type	Priority	Criteria	Checklist
----	------	----------	----------	-----------

OV-6c-01	General	Must	The OV-6c content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
OV-6c-02	General	Must	The OV-6c contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
OV-6c-03	General	Must	The OV-6c conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
OV-6c-04	General	Must	The OV-6c contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
OV-6c-05	General	Must	The OV-6c is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
OV-6c-06	General	Must	The OV-6c is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
OV-6c-07	Design	Must	The OV-6c shall include an accompanying description that describes the scenario or situation.	<input type="checkbox"/>
OV-6c-08	Design	Must	The OV-6c shall include Swimlanes with names and descriptions. (BPMN Only)	<input type="checkbox"/>
OV-6c-09	Design	Must	The OV-6c shall include Event Arrows, including name, description, and originating and terminating Operational Performer names.	<input type="checkbox"/>
OV-6c-10	Design	Must	The OV-6c shall include an Event timeline that shows sequencing of time. A unit-graduated timeline is preferred. If unit timing is not able to be determined, a time flow is acceptable.	<input type="checkbox"/>
OV-6c-11	Design	Must	Swimlanes or lifelines in the OV-6c shall map to Operational Performers in the OV-2.	<input type="checkbox"/>
OV-6c-12	Design	Must	The OV-6c will align with the applicable Joint Mission Threads when provided.	<input type="checkbox"/>
OV-6c-13	Design	Must	Events in OV-6c shall map to triggering Events in OV-3. They should not reference themselves.	<input type="checkbox"/>
OV-6c-14	Design	Must	Events in OV-6c shall map to Inputs and Outputs of Operational Activities in the OV-5b.	<input type="checkbox"/>
OV-6c-15	Design	Must	Data Objects shall be associated with a process step (as an Input and/or Output) or linked to a Sequence Flow or Message Flow. (BPMN Only)	<input type="checkbox"/>
OV-6c-16	Design	Must	Message Flows shall be used between Pools and Sequence Flows shall be within Pools. (BPMN Only)	<input type="checkbox"/>
OV-6c-17	Design	Must	Each Process shall be traceable to one or more Activities in the OV-5b. (BPMN Only)	<input type="checkbox"/>
OV-6c-18	Design	Must	BPMN gateways and conditional sequence flows shall be traceable to OV-6a Business Rules when applicable.	<input type="checkbox"/>
OV-6c-19	Style	Recommended	A legend shall be used to explain specific elements of the diagram to facilitate	<input type="checkbox"/>

			understanding by the reviewer.	
OV-6c-20	Style	Recommended	Data Elements that contain data element synonyms (DES) must reference one or more Data Elements.	<input type="checkbox"/>
OV-6c-21	Style	Recommended	Each diagram shall have at least two events and two processes.	<input type="checkbox"/>

11. Project Viewpoint (PV)

11.1. PV-1: Project Portfolio Relationships

11.1.1. Description

The PV-1 represents an organizational perspective on programs, projects, portfolios, or initiatives.

The PV-1 enables the user to model the organizational structures needed to manage programs, projects, portfolios, or initiatives. It shows dependency relationships between the actual organizations that own the programs, projects, portfolios, or initiatives. This model could be used to represent organization relationships associated with transformation initiatives along with those who are responsible for managing programs, projects, and portfolios. The PV-1 provides a means of analyzing the main dependencies between acquisition elements or transformation elements.

The intended usage of the PV-1 includes, but is not limited to:

- Program management (specified acquisition program structure)
- Project organization
- Cross-cutting initiatives to be tracked across portfolios

11.1.2. Checklist

Table 29. PV-1 Checklist

ID	Type	Priority	Criteria	Checklist
PV-1-01	General	Must	The PV-1 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
PV-1-02	General	Must	The PV-1 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
PV-1-03	General	Must	The PV-1 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
PV-1-04	General	Must	The PV-1 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
PV-1-05	General	Must	The PV-1 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>

PV-1-06	General	Must	The PV-1 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
PV-1-07	Design	Must	The PV-1 shows dependency relationships between organizations and their programs, projects, portfolios, and/or initiatives.	<input type="checkbox"/>
PV-1-08	Design	Could	The PV-1 represents organizational relationships associated with transformation initiatives along with those who are responsible for managing programs, projects, and portfolios.	<input type="checkbox"/>

11.2. PV-2: Project Timelines

11.2.1. Description

The PV-2 provides a timeline perspective on programs. The PV-2 is intended primarily to support the acquisition and fielding processes including the management of dependencies between projects and the integration of DoD 5000.1 Defense Acquisition System policies to achieve a successfully integrated capability. The PV-2 is not limited to the acquisition and fielding processes.

The intended usage of the PV-2 includes:

- Project management and control (including delivery timescales)
- Project dependency risk identification
- Management of dependencies
- Portfolio management

11.2.2. Checklist

Table 30. PV-2 Checklist

ID	Type	Priority	Criteria	Checklist
PV-2-01	General	Must	The PV-2 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
PV-2-02	General	Must	The PV-2 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
PV-2-03	General	Must	The PV-2 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
PV-2-04	General	Must	The PV-2 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
PV-2-05	General	Must	The PV-2 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>

PV-2-06	General	Must	The PV-2 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
PV-2-07	Design	Must	The PV-2 describes the “when” through portrayal of project timelines, relationships, and milestones.	<input type="checkbox"/>

11.3. PV-3: Project to Capability Mapping

11.3.1. Description

The PV-3 supports the acquisition and deployment processes, including the management of dependencies between projects and the integration of all relevant project and program elements to achieve a capability.

The PV-3 maps programs, projects, portfolios, or initiatives to capabilities to show how the specific elements help to achieve a capability. Programs, projects, portfolios, or initiatives are mapped to the capability for a specific timeframe. Programs, projects, portfolios, or initiatives may contribute to multiple capabilities and may mature across time. The analysis can be used to identify capability redundancies and shortfalls, highlight phasing issues, expose organizational or system interoperability problems, and support program decisions, such as when to phase out a legacy system.

The intended usage of the PV-3 includes:

- Tracing capability requirements to projects
- Capability audit

11.3.2. Checklist

Table 31. PV-3 Checklist

ID	Type	Priority	Criteria	Checklist
PV-3-01	General	Must	The PV-3 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
PV-3-02	General	Must	The PV-3 contains all required elements/attributes necessary to meet the model’s purpose.	<input type="checkbox"/>
PV-3-03	General	Must	The PV-3 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
PV-3-04	General	Must	The PV-3 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
PV-3-05	General	Must	The PV-3 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
PV-3-06	General	Must	The PV-3 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
PV-3-07	Design	Must	The PV-3 maps programs, projects, portfolios, or initiatives to capabilities mapped in the Capability Viewpoint (CV).	<input type="checkbox"/>

PV-3-08	Design	Could	The PV-3 is structured in a matrix or table format with capabilities lined across the x-axis and programs, projects, and/or initiatives across the y-axis.	<input type="checkbox"/>
PV-3-09	Design	Could	If the PV-3 is structured as a matrix or table, the axes' intersections (i.e. fields) contain a binary value to represent a fit or a gap (e.g. a value of "0" could be a gap, and a value of "1" could be a fit.)	<input type="checkbox"/>
PV-3-09	Design	Could	If the PV-3 is structured as a matrix or table, the fit/gap mapping only occurs at the lowest respective tier on each axis.	<input type="checkbox"/>

12. Standards Viewpoint (StdV)

12.1. StdV-1: Standards Profile

12.1.1. Description

The StdV-1 defines the technical, operational, and business standards, guidance, and policy applicable to the architecture being described. As well as identifying applicable technical standards, the StdV-1 also documents the policies and standards that apply to the operational or business context. In most cases, building a Standards Profile consists of identifying and listing the applicable portions of existing and emerging documentation. A StdV-1 should identify both existing guidelines, as well as any areas lacking guidance. As with other models, each profile is assigned a specific timescale (e.g., "As-Is", "To-Be", or transitional). Linking the profile to a defined timescale enables the profile to consider both emerging technologies and any current technical standards that are expected to be updated or become obsolete. If more than one emerging standard time-period is applicable to an architecture, then a StdV-2 Standards Forecast should be completed as well as a StdV-1.

The intended usage of the StdV-1 includes:

- Application of standards (informing project strategy)
- Standards compliance

12.1.2. Checklist

Table 32. StdV-1 Checklist

ID	Type	Priority	Criteria	Checklist
StdV-1-01	General	Must	The StdV-1 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
StdV-1-02	General	Must	The StdV-1 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
StdV-1-03	General	Must	The StdV-1 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
StdV-1-04	General	Must	The StdV-1 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>

StdV-1-05	General	Must	The StdV-1 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
StdV-1-06	General	Must	The StdV-1 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
StdV-1-06	Design	Must	Data standards identified in the SV-6/SvcV-6 shall be included in the Std-1 or explained in the matrix description.	<input type="checkbox"/>
StdV-1-06	Design	Must	Proprietary standards in use shall be identified and explained as such within the matrix description.	<input type="checkbox"/>
StdV-1-06	Design	Must	Technical standards shall be applicable to Services, Systems Interfaces, Services, Service Interfaces, Communications Networks and Communications Interfaces on the SV-1/SvcV-1 and SV-2/SvcV-2.	<input type="checkbox"/>
StdV-1-06	Design	Must	Technical standards shall be applicable to System/Service Resource Flows identified in the SV- 6/SvcV-6.	<input type="checkbox"/>
StdV-1-06	Design	Must	Timed technology forecasts in SV-9/SvcV-9 and any movement of a standard to its next version shall be identified and cited.	<input type="checkbox"/>
StdV-1-06	Design	Must	Applicable Organization Unique Standards (OUS) shall be included in the StdV-1.	<input type="checkbox"/>

12.2. StdV-2: Standards Forecast

12.2.1. Description

The StdV-2 contains expected changes in technology-related standards, operational standards, or business standards and conventions, which are documented in the StdV-1 model. The forecast for evolutionary changes in the standards need to be correlated against the time periods mentioned in the SV-8 Systems Evolution Description, SvcV-8 Services Evolution Description, SV-9 Systems Technology & Skills Forecast, and SvcV-9 Services Technology & Skills Forecast models.

A StdV-2 is a detailed description of emerging standards relevant to the systems, operational, and business activities covered by the Architectural Description. The forecast should be tailored to focus on areas that are related to the purpose for which a given Architectural Description is being built, and should identify issues that affect the architecture.

A StdV-2 complements and expands on the StdV-1 Standards Profile model and should be used when more than one emerging standard time-period is applicable to the architecture.

One of the prime purposes of this model is to identify critical technology standards, their fragility, and the impact of these standards on the future development and maintainability of the architecture and its constituent elements.

The intended usage of the StdV-2 includes:

- Forecasting future changes in standards (informing project strategy)

12.2.2. Checklist

Table 33. StdV-2 Checklist

ID	Type	Priority	Criteria	Checklist
StdV-2-01	General	Must	The StdV-2 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
StdV-2-02	General	Must	The StdV-2 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
StdV-2-03	General	Must	The StdV-2 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
StdV-2-04	General	Must	The StdV-2 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
StdV-2-05	General	Must	The StdV-2 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
StdV-2-06	General	Must	The StdV-2 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
StdV-2-07	Design	Must	The matrix, if required, shall contain only emerging standards.	<input type="checkbox"/>

13. Systems Viewpoint (SV)

13.1. SV-1: Systems Interface Description

13.1.1. Description

The SV-1 addresses the composition and interaction of Systems. For DoDAF v2.0, the SV-1 incorporates the human elements as types of Performers - Organizations and Personnel Types.

The SV-1 links together the operational and systems architecture models by depicting how Resources are structured and interact to realize the logical architecture specified in an OV-2 Operational Resource Flow Description. A SV-1 may represent the realization of a requirement specified in an OV-2 Operational Resource Flow Description (i.e., in a "To-Be" architecture), and so there may be many alternative SV models that could realize the operational requirement. Alternatively, in an "As-Is" architecture, the OV-2 Operational Resource Flow Description may simply be a simplified, logical representation of the SV-1 to allow communication of key Resource Flows to non-technical stakeholders.

A System Resource Flow is a simplified representation of a pathway or network pattern, usually depicted graphically as a connector (i.e., a line with possible amplifying information). The SV-1 depicts all System Resource Flows between Systems that are of interest. Note that Resource Flows between Systems may be further specified in detail in SV-2 Systems Resource Flow Description and SV-6 Systems Resource Flow Matrix.

Sub-System assemblies may be identified in SV-1 to any level (i.e., depth) of decomposition the architect sees fit. SV-1 may also identify the Physical Assets (e.g., Platforms) at which Resources are deployed, and optionally overlay Operational Activities and Locations that utilize those Resources. In many cases, an operational activity and location

depicted in an OV-2 Operational Resource Flow Description model may well be the logical representation of the resource that is shown in SV-1.

The intended usage of the SV-1 includes:

- Definition of System concepts
- Definition of System options
- System Resource Flow requirements capture
- Capability integration planning
- System integration management
- Operational planning (capability and performer definition)

The SV-1 is used in two complementary ways:

- Describe the Resource Flows exchanged between resources in the architecture
- Describe a solution, or solution option, in terms of the components of capability and their physical integration on platforms and other facilities

13.1.2. Checklist

Table 34. SV-1 Checklist

ID	Type	Priority	Criteria	Checklist
SV-1-01	General	Must	The SV-1 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
SV-1-02	General	Must	The SV-1 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
SV-1-03	General	Must	The SV-1 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
SV-1-04	General	Must	The SV-1 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
SV-1-05	General	Must	The SV-1 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
SV-1-06	General	Must	The SV-1 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
SV-1-07	Design	Must	The Systems/Services shall map to the related Performers in the OV-2.	<input type="checkbox"/>
SV-1-08	Design	Must	Interfaces must map to one or more Needlines in the OV-2.	<input type="checkbox"/>

SV-1-09	Design	Must	An Interface shall be implemented by Communications Link(s) or communications network(s) in the SV-2/SvcV-2.	<input type="checkbox"/>
SV-1-10	Design	Must	Systems/Services must match Systems/Services in the SV-5a/b and SvcV-5.	<input type="checkbox"/>
SV-1-11	Design	Must	The performance parameters of the SV-7/SvcV-7 shall apply to Systems, Sub-Systems, System hardware/software, or Services in the SV-1/SvcV-1.	<input type="checkbox"/>
SV-1-12	Design	Must	The Systems, Sub-Systems, and System hardware/software or Services of the SV-8/SvcV-8 shall match the corresponding elements in the SV-1/SvcV-1.	<input type="checkbox"/>
SV-1-13	Design	Must	Interfaces must map to System/Service Resource Flows listed in the SV/SvcV-4 and SV/SvcV-6.	<input type="checkbox"/>
SV-1-14	Design	Must	The model shall include external Systems/Services Resources that interact with the subject System/Service Resource.	<input type="checkbox"/>
SV-1-15	Design	Must	The model shall include external Systems/Services that interact with the subject architecture's Systems/Services.	<input type="checkbox"/>
SV-1-16	Design	Must	The model shall depict all the significant subsystems/services that interact with the subject System/Service Resource.	<input type="checkbox"/>
SV-1-17	Design	Must	There shall be at least one interface for each System or Service Resource Pair.	<input type="checkbox"/>

13.2. SV-2: Systems Resource Flow Description

13.2.1. Description

A SV-2 specifies the System Resource Flows between Systems and may also list the protocol stacks used in connections.

A SV-2 DoDAF-described Model is used to give a precise specification of a connection between Systems. This may be an existing connection, or a specification for a connection that is to be made.

The intended usage of the SV-2 includes:

- Resource Flow specification

13.2.2. Checklist

Table 35. SV-2 Checklist

ID	Type	Priority	Criteria	Checklist
SV-2-01	General	Must	The SV-2 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
SV-2-02	General	Must	The SV-2 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>

SV-2-03	General	Must	The SV-2 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
SV-2-04	General	Must	The SV-2 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
SV-2-05	General	Must	The SV-2 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
SV-2-06	General	Must	The SV-2 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
SV-2-07	Design	Must	The SV-2/SvcV-2 Communications Connections shall include correct identifiers, Communications Link(s), and communications type(s).	<input type="checkbox"/>
SV-2-08	Design	Must	The SV-2/SvcV-2 names and identifiers must map back to the OV-2 and SV-1/SvcV-1 (if developed).	<input type="checkbox"/>
SV-2-09	Design	Must	The Communications Link in the SV-2/SvcV-2 must address the physical implementation of an Interface in the SV-1/SvcV-1.	<input type="checkbox"/>
SV-2-10	Design	Must	Systems/Services depicted in the SV-2/SvcV-2 must be the same ones as depicted in the SV-1/SvcV-1.	<input type="checkbox"/>
SV-2-11	Design	Must	The model must include all Systems/Services that interact with the subject System/Service, both internally and externally.	<input type="checkbox"/>
SV-2-12	Style	Recommended	The model should include the pertinent communication system attributes.	<input type="checkbox"/>
SV-2-13	Design	Must	Performance parameters of the SV-7 that deal with communications Systems, Communications Links, and communications networks must map to the corresponding elements in the SV-2/SvcV-2.	<input type="checkbox"/>

13.3. SV-3: Systems-Systems Matrix

13.3.1. Description

A SV-3 enables a quick overview of all the system resource interactions specified in one or more SV-1 Systems Interface Description models. The SV-3 provides a tabular summary of the system interactions specified in the SV-1 Systems Interface Description model for the Architectural Description. The matrix format supports a rapid assessment of potential commonalities and redundancies (or, if fault-tolerance is desired, the lack of redundancies).

The SV-3 can be organized in several ways to emphasize the association of groups of system pairs in context with the architecture's purpose.

The intended usage of the SV-3 includes:

- Summarizing system resource interactions

- Interface management
- Comparing interoperability characteristics of solution options

13.3.2. Checklist

Table 36. SV-3 Checklist

ID	Type	Priority	Criteria	Checklist
SV-3-01	General	Must	The SV-3 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
SV-3-02	General	Must	The SV-3 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
SV-3-03	General	Must	The SV-3 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
SV-3-04	General	Must	The SV-3 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
SV-3-05	General	Must	The SV-3 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
SV-3-06	General	Must	The SV-3 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
SV-3-07	Design	Must	All Systems/Services from the SV-1/SvcV-1 must be represented in the SV-3 or SvcV-3a/b.	<input type="checkbox"/>
SV-3-08	Style	Recommended	An "X" or designated value must be placed at the intersection between the column and the row.	<input type="checkbox"/>

13.4. SV-4: Systems Functionality Description (+ Data Flow Diagram)

13.4.1. Description

The SV-4 addresses human and system functionality.

The primary purposes of SV-4 are to:

- Develop a clear description of the necessary data flows that are input (consumed) by and output (produced) by each resource
- Ensure that the functional connectivity is complete (i.e., that a resource's required inputs are all satisfied)
- Ensure that the functional decomposition reaches an appropriate level of detail
- The Systems Functionality Description provides detailed information regarding the allocation of functions to resources and the flow of resources between functions

The SV-4 is the Systems Viewpoint model counterpart to the OV-5b Activity Model of the Operational Viewpoint.

The intended usage of the SV-4 includes:

- Description of task workflow
- Identification of functional system requirements
- Functional decomposition of systems
- Relate human and system functions

13.4.2. Checklist

Table 37. SV-4 Checklist

ID	Type	Priority	Criteria	Checklist
SV-4-01	General	Must	The SV-4 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
SV-4-02	General	Must	The SV-4 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
SV-4-03	General	Must	The SV-4 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
SV-4-04	General	Must	The SV-4 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
SV-4-05	General	Must	The SV-4 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
SV-4-06	General	Must	The SV-4 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
SV-4-07	Design	Must	Relevant System/Service Functions from the common lists (i.e., JCSFL) must be used to populate the models. Exceptions should be noted.	<input type="checkbox"/>
SV-4-08	Design	Must	The SV-4/SvcV-4 Functional Hierarchy shall be developed to the level of detail necessary to adequately describe functionality.	<input type="checkbox"/>
SV-4-09	Design	Must	The SV-4/SvcV-4 must include System/Service Functions that are executed by Systems/Services depicted in the SV-1/SvcV-1.	<input type="checkbox"/>
SV-4-10	Design	Must	Each System/Service Function must have at least one input and one output on the Data Flow Diagram.	<input type="checkbox"/>
SV-4-11	Design	Must	Data Flows from Systems/Services that interact with the subject System/Services must be indicated in the Data Flow Diagram.	<input type="checkbox"/>
SV-4-12	Design	Must	Relevant Data Stores/Repositories shall be shown on the Data Flow Diagram.	<input type="checkbox"/>

SV-4-13	Design	Must	System/Service Data Flows from SV-4/SvcV-4 must map to entities in DIV-3.	<input type="checkbox"/>
SV-4-14	Design	Must	All System/Service Functions must map one-to-one to System/Service Functions in the SV-5a/b and SvcV-5.	<input type="checkbox"/>
SV-4-15	Design	Must	System/Service Data Flows must map to System/Service Resource Flows listed in the SV-6/SvcV-6.	<input type="checkbox"/>
SV-4-16	Design	Must	If System rules in the SV-10a/SvcV-10c deal with System/Service behavior, they shall reference System/Service Functions in the SV-4/SvcV-4.	<input type="checkbox"/>
SV-4-17	Design	Must	Events in the SV-10c/SvcV-10c must map to System/Service Data Flows in the SV-4/SvcV-4.	<input type="checkbox"/>
SV-4-18	Design	Must	System/Service Functions must map to the performance parameters for System/Service Functions listed in the SV-7/SvcV-7	<input type="checkbox"/>

13.5. SV-5a: Operational Activity to Systems Function Traceability Matrix

13.5.1. Description

The SV-5a addresses the linkage between System Functions described in SV-4 Systems Functionality Description and Operational Activities specified in OV-5a Operational Activity Decomposition Tree or OV-5b Operational Activity Model. The SV-5a depicts the mapping of system functions and, optionally, the capabilities and performers that provide them to operational activities. The SV-5a identifies the transformation of an operational need into a purposeful action performed by a system or solution.

During requirements definition, the SV-5a plays a particularly important role in tracing the architectural elements associated with system function requirements to those associated with user requirements. The intended usage of the SV-5a includes:

- Tracing functional system requirements to user requirements
- Tracing solution options to requirements
- Identification of overlaps or gaps

13.5.2. Checklist

Table 38. SV-5a Checklist

ID	Type	Priority	Criteria	Checklist
SV-5a-01	General	Must	The SV-5a content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
SV-5a-02	General	Must	The SV-5a contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>

SV-5a-03	General	Must	The SV-5a conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
SV-5a-04	General	Must	The SV-5a contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
SV-5a-05	General	Must	The SV-5a is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
SV-5a-06	General	Must	The SV-5a is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
SV-5a-07	Design	Must	Operational Activities must match those listed in the OV-5a.	<input type="checkbox"/>
SV-5a-08	Design	Must	System/Service Functions must match those listed in the SV-4/SvcV-4.	<input type="checkbox"/>
SV-5a-09	Style	Recommended	An "X" or designated value must be placed at the intersection between the related column and the row.	<input type="checkbox"/>

13.6. SV-5b: Operational Activity to Systems Traceability Matrix

13.6.1. Description

The SV-5b addresses the linkage between described in SV-1 Systems Functionality Description and Operational Activities specified in OV-5a Operational Activity Decomposition Tree or OV-5b Operational Activity Model. The SV-5b depicts the mapping of systems and, optionally, the capabilities and performers that provide them to operational activities. The SV-5b identifies the transformation of an operational need into a purposeful action performed by a system or solution.

During requirements definition, the SV-5b plays a particularly important role in tracing the architectural elements associated with system requirements to those associated with user requirements.

The intended usage of the SV-5b includes:

- Tracing system requirements to user requirements
- Tracing solution options to requirements
- Identification of overlaps or gaps

13.6.2. Checklist

Table 39. SV-5b Checklist

ID	Type	Priority	Criteria	Checklist
SV-5b-01	General	Must	The SV-5b content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
SV-5b-02	General	Must	The SV-5b contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>

SV-5b-03	General	Must	The SV-5b conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
SV-5b-04	General	Must	The SV-5b contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
SV-5b-05	General	Must	The SV-5b is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
SV-5b-06	General	Must	The SV-5b is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
SV-5b-07	Design	Must	Operational Activities must match those listed in the OV-5a.	<input type="checkbox"/>
SV-5b-08	Design	Must	System/Service Functions must match those listed in the SV-4/SvcV-4.	<input type="checkbox"/>
SV-5b-09	Style	Recommended	An "X" or designated value must be placed at the intersection between the related column and the row.	<input type="checkbox"/>

13.7. SV-6: Systems Resource Flow Matrix

13.7.1. Description

The SV-6 specifies the characteristics of the System Resource Flows exchanged between systems with emphasis on resources crossing the system boundary. The SV-6 focuses on the specific aspects of the system Resource Flow and the system Resource Flow content in a tabular format.

The intended usage of the SV-6 includes:

- Detailed definition of Resource Flows

13.7.2. Checklist

Table 40. SV-6 Checklist

ID	Type	Priority	Criteria	Checklist
SV-6-01	General	Must	The SV-6 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
SV-6-02	General	Must	The SV-6 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
SV-6-03	General	Must	The SV-6 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
SV-6-04	General	Must	The SV-6 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
SV-6-05	General	Must	The SV-6 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>

SV-6-06	General	Must	The SV-6 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
SV-6-07	Design	Must	Matrix entries should be taken from the appropriate Operational, System/Service and Standards Viewpoints (e.g., OV-3, SV-1/SvcV-1, SV- 4/SvcV-4, StdV-1, etc.) when available.	<input type="checkbox"/>
SV-6-08	Design	Must	The matrix shall maintain traceability with the OV-3, SV-1/SvcV-1, SV-2/SvcV-2, SV-4/SvcV-4, and StdV-1.	<input type="checkbox"/>
SV-6-09	Design	Must	The matrix shall include a row for each System/Service Resource Flow identifier shown on the SV-1/SvcV-1.	<input type="checkbox"/>
SV-6-10	Design	Must	Standards in the matrix shall be the same as those listed in the StdV-1.	<input type="checkbox"/>
SV-6-11	Design	Must	Systems/Services identifiers and names shall match those in the SV-1/SvcV-1 and SV-2/SvcV-2.	<input type="checkbox"/>
SV-6-12	Design	Must	If any part of an Operational Resource Flow in the OV-3 originates from or flows to an Operational Activity that is to be automated, then that Resource Flow shall map to one or more System/Service Resource Flow in the SV-6/SvcV-6.	<input type="checkbox"/>
SV-6-13	Design	Must	Every System/Service Resource Flow in the matrix shall correspond to a resource flow in the SV-1/ SvcV-1.	<input type="checkbox"/>
SV-6-14	Design	Must	System/Service Resource Flows in the SV-4/SvcV-4 shall map to those appearing in the matrix.	<input type="checkbox"/>
SV-6-15	Design	Must	Events in the SV-10/SvcV-10c shall map to triggering events in the SV-6/SvcV-6.	<input type="checkbox"/>
SV-6-16	Design	Must	Performance parameters in the SV-7 that deal with Interface performance and System Data Exchange capacity requirements trace to System Data Exchanges in the SV-6.	<input type="checkbox"/>
SV-6-17	Style	Recommended	Attributes and their values shall be identified for each System/Service Resource Flow. The list provided in this guide shall be used to the maximum extent possible. Others may be used to help describe each System/Service Resource Flow based on program needs. There shall be no blank fields at the intersection between the columns and rows.	<input type="checkbox"/>

13.8. SV-7: Systems Measures Matrix

13.8.1. Description

The SV-7 depicts the measures (metrics) of resources. The Systems Measures Matrix expands on the information presented in a SV-1 by depicting the characteristics of the resources in the SV-1.

The intended usage of the SV-7 includes:

- Definition of performance characteristics and measures (metrics)

- Identification of non-functional requirements

13.8.2. Checklist

Table 41. SV-7 Checklist

ID	Type	Priority	Criteria	Checklist
SV-7-01	General	Must	The SV-7 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
SV-7-02	General	Must	The SV-7 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
SV-7-03	General	Must	The SV-7 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
SV-7-04	General	Must	The SV-7 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
SV-7-05	General	Must	The SV-7 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
SV-7-06	General	Must	The SV-7 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
SV-7-07	Design	Must	Performance parameters shall apply to Systems/services in the SV-1/SvcV-1.	<input type="checkbox"/>
SV-7-08	Design	Must	Performance parameters that deal with communications Systems/Services, Communications Links, and communications networks shall map to the corresponding elements in the SV-2/SvcV-2.	<input type="checkbox"/>
SV-7-09	Design	Must	Performance parameters that deal with System/Service Resource Flow performance and Data Exchange capacity requirements shall trace back to the System/Service Data Exchanges in the SV-6/SvcV-6.	<input type="checkbox"/>
SV-7-10	Design	Must	If required performance ranges described in the SV-7/SvcV-7 are associated with an overall System/Service evolution or migration plan described in the SV-8/SvcV-8, then the time periods in the SV-7/SvcV-7 should correspond to the milestones in the SV-8/SvcV-8.	<input type="checkbox"/>
SV-7-11	Design	Must	If the future performance expectations (goals) described in the SV-7/SvcV-7 is based on expected technology improvements, then the performance parameters and their time-periods in the SV-7/SvcV-7 should be coordinated with the timed technology forecasts described in the SV- 9/SvcV-9.	<input type="checkbox"/>

13.9. SV-8: Systems Evolution Description

13.9.1. Description

The SV-8 presents a whole lifecycle view of resources (systems), describing how they change over time. It shows the structure of several resources mapped against a timeline.

The intended usage of the SV-8 includes:

- Development of incremental acquisition strategy
- Planning technology insertion

13.9.2. Checklist

Table 42. SV-8 Checklist

ID	Type	Priority	Criteria	Checklist
SV-8-01	General	Must	The SV-8 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
SV-8-02	General	Must	The SV-8 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
SV-8-03	General	Must	The SV-8 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
SV-8-04	General	Must	The SV-8 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
SV-8-05	General	Must	The SV-8 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
SV-8-06	General	Must	The SV-8 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
SV-8-07	Design	Must	The SV-8/SvcV-8 shall convey the program's plan of action and milestones.	<input type="checkbox"/>
SV-8-08	Design	Must	The SV-8/SvcV-8 shall Identify replacement Systems/Services, if applicable.	<input type="checkbox"/>
SV-8-09	Design	Must	The End-of-Service shall be reflected in the SV- 8/SvcV-8, if applicable.	<input type="checkbox"/>
SV-8-10	Design	Must	IPv6 transition shall be depicted in the SV- 8/SvcV-8, if applicable.	<input type="checkbox"/>
SV-8-11	Design	Must	The applicable Fiscal Year timeline shall be correctly shown based on the accompanying text description.	<input type="checkbox"/>
SV-8-12	Design	Must	The Systems/Services in the SV-8/SvcV-8 shall match the corresponding elements in the SV-1/SvcV-1 and SV-2/SvcV-2.	<input type="checkbox"/>
SV-8-13	Design	Must	If the SV-8/SvcV-8 identifies System/Service Functions to be implemented in each phase of a System/Service development, they must match the System/Service Functions in the SV-4/SvcV-4.	<input type="checkbox"/>
SV-8-14	Design	Must	If required performance ranges described in the SV-7/SvcV-7 are associated with an overall System/Service evolution or migration plan described in the SV-8/SvcV-8, then the time-periods in the SV-7/SvcV-7 should correspond to the milestones in the SV-8/SvcV-8.	<input type="checkbox"/>

SV-8-15	Design	Must	The time-periods associated with timelines and milestones in the SV-8/SvcV-8 shall be coordinated with time-periods associated with timed technology forecasts in the SV-9/SvcV-9.	<input type="checkbox"/>
---------	--------	------	--	--------------------------

13.10. SV-9: Systems Technology & Skills Forecast

13.10.1. Description

The SV-9 defines the underlying current and expected supporting technologies and skills. Expected supporting technologies and skills are those that can be reasonably forecast given the current state of technology and skills as well as the expected improvements or trends. New technologies and skills are tied to specific time periods, which can correlate against the time periods used in SV-8 milestones and linked to Capability Phases.

The SV-9 provides a summary of emerging technologies and skills that impact the architecture. The SV-9 provides descriptions of relevant:

- Emerging capabilities
- Industry trends
- Predictions (with associated confidence factors) of the availability and readiness of specific hardware and software systems.
- Current and possible future skills

13.10.2. Checklist

Table 43. SV-9 Checklist

ID	Type	Priority	Criteria	Checklist
SV-9-01	General	Must	The SV-9 content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
SV-9-02	General	Must	The SV-9 contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
SV-9-03	General	Must	The SV-9 conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
SV-9-04	General	Must	The SV-9 contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
SV-9-05	General	Must	The SV-9 is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
SV-9-06	General	Must	The SV-9 is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
SV-9-07	Design	Must	Timed technology forecasts that impact Services, Systems, Sub-Systems, and System	<input type="checkbox"/>

			hardware/software items of SV-1/SvcV-1 shall be identified and cited.	
SV-9-08	Design	Must	Timed technology forecasts that impact communications Services/Systems, Communications Links, and communications networks in the SV-2/SvcV-2 shall be identified and cited.	<input type="checkbox"/>
SV-9-09	Design	Must	If the future performance expectations (goals) described in the SV-7/SvcV-7 is based on expected technology improvements, then the performance parameters and their time-periods in the SV-7/SvcV-7 shall be coordinated with the timed technology forecasts described in the SV- 9/SvcV-9.	<input type="checkbox"/>
SV-9-10	Design	Must	The time-periods associated with timelines and milestones in the SV-8/SvcV-8 shall be coordinated with time-periods associated with timed technology forecasts in the SV-9/SvcV-9.	<input type="checkbox"/>

13.11. SV-10a: Systems Rules Model

13.11.1. Description

The SV-10a specifies functional and non-functional constraints on the implementation aspects of the architecture (i.e., the structural and behavioral elements of the Systems Viewpoint).

The SV-10a DoDAF-described Model describes constraints on the resources, functions, data, and ports that make up the SV physical architecture. The constraints are specified in text and may be functional or structural (i.e., non-functional).

The intended usage of the SV-10a includes:

- Definition of implementation logic
- Identification of resource constraints

13.11.2. Checklist

Table 44. SV-10a Checklist

ID	Type	Priority	Criteria	Checklist
SV-10a-01	General	Must	The SV-10a content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
SV-10a-02	General	Must	The SV-10a contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
SV-10a-03	General	Must	The SV-10a conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
SV-10a-04	General	Must	The SV-10a contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
SV-10a-05	General	Must	The SV-10a is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>

SV-10a-06	General	Must	The SV-10a is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
SV-10a-07	Design	Must	System/Service behavior described in the SV-10a/SvcV-10 should be aligned with the System/Service Function in the SV-4/SvcV-4.	<input type="checkbox"/>
SV-10a-08	Design	Must	Ensure that the rules are clear, deterministic and unambiguous.	<input type="checkbox"/>

13.12. SV-10b: Systems State Transition Description

13.12.1. Description

The SV-10b is a graphical method of describing a resource (or system function) response to various events by changing its state. The diagram basically represents the sets of events to which the resources in the Activities respond (by taking an action to move to a new state) as a function of its current state. Each transition specifies an event and an action.

The explicit time sequencing of service functions in response to external and internal events is not fully expressed in SV-4 Systems Functionality Description. The SV-10b can be used to describe the explicit sequencing of the functions. Alternatively, SV-10b can be used to reflect explicit sequencing of the actions internal to a single function, or the sequencing of system functions with respect to a specific resource.

The intended usage of the SV-10b includes:

- Definition of states, events and state transitions (behavioral modeling)
- Identification of constraints

13.12.2. Checklist

Table 45. SV10-b Checklist

ID	Type	Priority	Criteria	Checklist
SV-10b-01	General	Must	The SV-10b content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
SV-10b-02	General	Must	The SV-10b contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
SV-10b-03	General	Must	The SV-10b conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
SV-10b-04	General	Must	The SV-10b contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
SV-10b-05	General	Must	The SV-10b is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
SV-10b-06	General	Must	The SV-10b is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>

SV-10b-07	Design	Must	The state diagram must account for all behavior of objects, impacting events, actions and associated transitions, and resulting states being considered.	<input type="checkbox"/>
-----------	--------	------	--	--------------------------

13.13. SV-10c: Systems Event-Trace Description

13.13.1. Description

The SV-10c provides a time-ordered examination of the interactions between functional resources. Each event-trace diagram should have an accompanying description that defines the scenario or situation.

The SV-10c is valuable for moving to the next level of detail from the initial solution design, to help define a sequence of functions and system data interfaces, and to ensure that each participating resource or System Port role has the necessary information it needs, at the right time, to perform its assigned functionality.

The intended usage of the SV-10c includes:

- Analysis of resource events impacting operation
- Behavioral analysis
- Identification of non-functional system requirements

13.13.2. Checklist

Table 46. SV-10c Checklist

ID	Type	Priority	Criteria	Checklist
SV-10c-01	General	Must	The SV-10c content is factually correct and faithfully represents the intended subject.	<input type="checkbox"/>
SV-10c-02	General	Must	The SV-10c contains all required elements/attributes necessary to meet the model's purpose.	<input type="checkbox"/>
SV-10c-03	General	Must	The SV-10c conforms to an established style (theme), scope (boundary), and perspective (detail).	<input type="checkbox"/>
SV-10c-04	General	Must	The SV-10c contains elements that are synchronized and non-conflicting across the viewpoints.	<input type="checkbox"/>
SV-10c-05	General	Must	The SV-10c is designed and asserted with respect to the bigger (or smaller) picture.	<input type="checkbox"/>
SV-10c-06	General	Must	The SV-10c is useful, necessary, and informative to the intended consumer.	<input type="checkbox"/>
SV-10c-07	Design	Must	Events in the SV-10c/SvcV-10c shall map to System Resource Flows/Data Flows in the SV-4/SvcV-4.	<input type="checkbox"/>
SV-10c-08	Design	Must	Events in the SV-10c/SvcV-10c must map to triggering Events in the SV-6/SvcV-6.	<input type="checkbox"/>

SV-10c-09	Design	Must	Transitions must be identified within Events.	<input type="checkbox"/>
SV-10c-10	Design	Must	There shall be a sequence captured in a diagram for each use-case flow or scenario identified for the subject System/Service in the context of the Operational Viewpoint models.	<input type="checkbox"/>
SV-10c-11	Design	Must	Messages shall be characterized for each interaction in the flow of events being modeled.	<input type="checkbox"/>
SV-10c-12	Design	Must	Each message in the sequence diagram shall correspond to the resources in the Operational Viewpoint models.	<input type="checkbox"/>
SV-10c-13	Design	Must	Systems/Services depicted in the SV-10c/SvcV-10c must map to those in the SV- 1/SvcV-1.	<input type="checkbox"/>
SV-10c-14	Design	Must	Systems/Services Functions depicted in the SV-10c/SvcV-10c must map to those in the SV-4/SvcV-4.	<input type="checkbox"/>
SV-10c-15	Design	Must	The scenario used in the SV-10c/SvcV-10c shall align with the scenario used in the related OV-6c.	<input type="checkbox"/>