DEPARTMENT OF DEFENSE
6000 DEFENSE PENTAGON
WASHINGTON, DC 20301-6000

CHIEF INFORMATION OFFICER

MAY 28 2009

MEMORANDUM FOR SECRETARIES OF THE MILITARY DEPARTMENTS
CHAIRMAN OF THE JOINT CHIEFS OF STAFF
UNDER SECRETARIES OF DEFENSE
COMMANDERS OF THE COMBATANT COMMANDS
ASSISTANT SECRETARIES OF DEFENSE
GENERAL COUNSEL OF THE DEPARTMENT OF DEFENSE
DIRECTOR, OPERATIONAL TEST AND EVALUATION
INSPECTOR GENERAL, DEPARTMENT OF DEFENSE
ASSISTANTS TO THE SECRETARY OF DEFENSE
DIRECTOR, ADMINISTRATION AND MANAGEMENT
DIRECTOR, PROGRAM ANALYSIS AND EVALUATION
DIRECTOR, NET ASSESSMENT
DIRECTORS OF THE DEFENSE AGENCIES
DIRECTORS OF DOD FIELD ACTIVITIES

SUBJECT: The Department of Defense Architecture Framework (DoDAF) Version 2.0

The DoD Architecture Framework (DoDAF) Version 2.0 is approved for immediate use. Version 2.0, which supersedes Version 1.5 released 23 April 2007, is the prescribed framework for all Department architectures, and represents a substantial shift in approach. It places emphasis upon a disciplined process of defining the purpose, scope and information requirements of the architecture up-front, followed by collection of data in accordance with a standard vocabulary. Data collected through the architectural process is delivered to the customer in either standard models or "Fit for Purpose" presentations.

DoDAF Version 2.0 accommodates artifacts and viewpoints created under version 1.5 and includes new Viewpoints to meet user requirements. While DODAF is the prescribed means of representing architecture content, the specific models developed are Selected by the user and defined by the processes which they support. DODAF Version 2.0 provides a richer, yet leaner methodology to document essential architectural content. Architectures shall comply with Version 2.0 in their next major release. DODAF version 2.0 is available at https://www.us.army.mil/suite/page/454707.

Version 2.0 consists of three volumes and a Journal:

- Volume 1 (Manager's Guide - Introduction, Overview, and Concepts) introduces DoD architecture concepts and provides general guidance for development, use, and management of DoD architectures.
- Volume 2 (Architect's Guide – Architectural Data and Models) describes the Meta-model data groups, and their associated models from a technical viewpoint.
- Volume 3 (Developer's Guide - DoDAF Meta-model Physical Exchange Specification) relates the Conceptual Data Model structure, Logical Data Model relationships, associations, and business rules to introduce the Physical Exchange Specification which provides the constructs needed to enable exchange of data and derived information among users and Communities of Interest.
- The DoDAF Journal provides a place for submitting future change requests to DoDAF or the DoDAF Meta-model, and provides the examples referenced in the various DoDAF volumes. The DoDAF Journal also contains supplementary "how to" information relating to architecture, architecture best practices, lessons learned, and reference documents.

Our future plans include the development of a "virtual DoDAF", that will allow for incremental changes based upon user feedback and DoDAF Core Management Group adjudication. The release of the "virtual DoDAF" will be announced via the DoDAF website referenced above. My point of contact for the DoDAF is Mr. Michael L. Wayson, (703) 607-0482, michael.wayson@osd.mil.

David M. Wennergren
Performing the Duties of the
ASD(NII)/DoD CIO

FINAL

DoD Architecture Framework
Version 2.0

Volume 2: Architectural Data and Models

Architect's Guide

28 May 2009

This page left intentionally blank

TABLE OF CONTENTS

SECTION PAGE

TABLE OF CONTENTS (CONT)

SECTION **PAGE**

LIST OF FIGURES

FIGURE **PAGE**

LIST OF TABLES

TABLE PAGE

LIST OF TABLES (CONT)

TABLE **PAGE**

This page left intentionally blank

1. INTRODUCTION

Department of Defense Architecture Framework (DoDAF) V2.0 serves as the overarching, comprehensive framework and conceptual model enabling the development of architectures to facilitate Department of Defense (DoD) managers at all levels to use architectures developed under the DoDAF in support of more effective decision-making through organized information sharing across the Department, Joint Capability Areas (JCAs), Components, and Program boundaries. DoDAF V2.0 focuses on architectural data and information required by key DoD decision-makers, rather than on developing individual products.

DoDAF Volume 2 describes the technical aspects of data collection and presentation, organized through the DoDAF Meta-model (DM2), enabling the requirements of architecture stakeholders and their viewpoints to be realized through both federation of efforts, and data sharing, as described in Volume 1.

DoDAF also serves as the principal guide for development of *integrated architectures.* DoD Instruction 4630[1] defines an integrated architecture as, *"An architecture consisting of multiple views facilitating integration and promoting interoperability across capabilities and among integrated architectures."* For the purposes of architecture development, the term integrated means that data required in more than one of the architectural models is commonly defined and understood across those models. Integrated architectures are a property or design principle for architectures at all levels: Capability, Component, Solution, and Enterprise (in the context of the DoD Enterprise Architecture (EA) being a federation architectures).

The Department has adopted a *federated approach* oriented toward distributed architectural data collection, organization, and management among the Components and Community of Interests (COIs). The DoD EA is comprised of DoD architecture policy, tools, and standards, DoD-level architectural descriptions like the DoD Information Enterprise Architecture (DoD IEA), DoD-level Capability architectural descriptions, and Component architectural descriptions. Solution architectural descriptions must conform to the DoD EA. This approach provides increased flexibility while retaining significant oversight and quality management services at the Departmental level.

Federating the DoD EA involves making the architectural descriptions described above both visible and accessible. Architectural Descriptions will register overview and summary metadata in a format based on the DoD Discovery Metadata Specification (DDMS) with extensions. The metadata will include a mapping against the capabilities areas contained within the JCA Taxonomy which will provide the overarching and organizing construct for the federation. Additional tasks for federation will involve providing proof of alignment with other Architectural Descriptions based on a variety of metrics that will evolve over time (e.g. business rules, technical standards, use of the Enterprise Vocabulary). This will result in a robust search capability for visibility. Publication of the Architectural Descriptions will provide accessibility to this important information. A governance process for the creation and management of the DoD EA will be provided by the DoD Chief Information Officer (CIO).

---

[1] DoD Instruction 4630.8, Procedures for interoperability and Supportability of Information technology (IT) and National Security Systems (NSS) 30 June 2004, Office of the Assistant Secretary of Defense (NII).

DoDAF V2.0 is intended to enable the sharing and reuse of architectural data. DoDAF accommodates various approaches, DoDAF-described Models, and definitions for communicating and facilitating the presentation of key architecture information (i.e., architecture vision, principles, guidance, processes) required for the development of Architectural Descriptions. It establishes a common foundation for understanding, comparing, and federating architectures and as such provides the overarching guidance for developing DoD Architectural Descriptions.

The DM2 is a data model that provides information needed to collect and organize data in a way easily understood. The development process for the DM2 is described in Section 2. The presentation descriptions and DoDAF-described Models in Volumes 1 and 2 provide guidance on how to develop graphical representations of that data that will be useful for decision-makers in analyzing and presenting alternatives for adoption, funding, and/or implementation.

DM2 is a complete replacement for the Core Architecture Data Model (CADM) that supported architecture development efforts under previous versions of DoDAF. CADM should not be used for new architecture development, except as it may be applicable to legacy architectures maintained under previous versions. DoDAF does not define the database design; but defines an exchange format for exchanging data. To aid any needed migrations forward, a mapping of CADM's independent entities to the DM2 data elements is provided in Appendix B.

Version 1.0 and 1.5 of the DoDAF used the term product or products to describe the visualizations of architectural data. In this volume, the term DoDAF-described Model is generally used, unless there is a specific reference to the products of earlier versions. DoDAF-described Models that have been populated or created with data for an architecture, the term Views will be used or Fit-for-Purpose Views will be used where the DoDAF-described Models are customized or combined for the decision-maker's need.

In addition, to align to International Standards Organization (ISO) 15074, ISO 14439, and ISO 42010/IEEE1471 terminology where appropriate, Views, in DoDAF V1.0 and 1.5, will be changed to Viewpoints in DoDAF V2.0 (e.g., from Operational View to Operational Viewpoint or System View to System Viewpoint.)

The Models described in DoDAF, including those that are legacy products from previous versions of DoDAF, are provided as pre-defined examples that can be used when developing presentations of architectural data.

DoDAF is prescribed for the use and development of Architectural Descriptions in the Department. Specific DoDAF-described Models for a particular purpose are prescribed by process-owners. All the DoDAF-described Models do not have to be created. DoDAF V2.0 is "Fit-for-Purpose", based on the decision-maker needs. DoDAF does not prescribe any particular models, but instead concentrates on data as the necessary ingredient for architecture development. 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. However, regulations and instructions from both DoD and Chairman of the Joint Chiefs of Staff (CJCS) have particular presentation view requirements. The architectural data described in DoDAF V2.0 can support many model and view requirements and the regulations and instructions should be consulted for specific model and view requirements.

Within DoDAF, the reference to "data" refers to the "architectural data" that an Architectural Description needs to capture. As an exception, in Volume 2, Section 2.3, "Information and Data" and Volume 2, Section 3.1.3, "Data and Information Viewpoint", the discussions describes the architectural data and the data that is being captured to populate the models for the solution. The "architectural data" may be the resource flows, but the "solution data" is the specific attributes of an instance of a resource flow for a given solution, e.g., the information that needs to capture the Latitude within a Cursor on Target message.

1.1 How This Volume is Organized

This volume contains information for the architect, and technical professionals that develop architectures. Section 1 is the Introduction to this volume. Section 2 presents the DoDAF Meta-model Data Groups**,** those categories of data that serve as the building blocks of architecture development. There are 12 categories of data presented. These are:

- Performers (2.1).
- Resource Flows (2.2).
- Information and Data (2.3).
- Activities (2.4).
- Training/Skill/Education (2.5).
- Capability (2.6).
- Services (2.7).
- Projects (2.8).
- Goals (2.9).
- Rules (2.10).
- Measures (2.11).
- Locations (2.12).

Within each of the categories, descriptions of the data are provided through:

- Introductory information which provides an overview of the Data Group.
- Data Entities that comprise the Data Group, and the relevant part of the DoDAF Meta-model, which provides the associations and relationships that characterize the data.
- Suggested Method(s) for collecting the data.
- Primary uses of the data.

As described previously in Volume 1, the concepts presented in DoDAF V2.0 are data-centric in nature, rather than product-centric as in previous versions of the DoDAF. Federation and sharing are facilitated by the use of common data as described and defined in this volume and the DoDAF Meta-model. Methods of collecting data, use, and graphical presentation are all suggested rather than mandated. Organizations can tailor their presentations or documents to suit the culture requirements of their own organization.

Section 3 describes the viewpoints of DoDAF V2.0. These viewpoints are the major categories of data, arranged into useful grouping to facilitate their use.

The appendices to the document are:

- Appendix A: Acronyms
- Appendix B: Mapping to DM2 Concepts (Maps the DM2 Concepts, Associations, and Attributes Mapping to DoDAF-described Models and the CADM Independent Entities to DM2 Data Elements)
- Appendix C: How Does DoDAF Represent Security
- Appendix D: References

In addition, the DoDAF Journal contains the following referenced files:

- DoDAF V2.0 Architecture Development Process for the DoDAF-described Models – A non-prescriptive set of tasks to develop DoDAF-described Models.
- DoDAF Product Development Questionnaire Analysis Report.
- DoDAF V2.0 Meta-model Data Dictionary.

The DoDAF Journal has two locations:

- A public DoDAF Journal website at *http://www.defenselink.mil/cio-nii/*.

- On Defense Knowledge Online DoDAF Journal at *https://www.us.army.mil/suite/page/454707*.

1.2 Presentations (Fit-for-Purpose Views) and Documents

Effective presentation of business information is necessary for architects to convey the data in the Architectural Description in a way meaningful to stakeholders. Since the purpose of the enterprise architecture discipline is to collect and store all relevant information about an enterprise, it can be assumed that the majority of information needed by an organization's decision-makers is contained somewhere in the architectural data. Presentations, or Fit-for-Purpose Views, are always dependent on the quality of the architecture information collected through the rigor of architecture methods. Many of the existing architecture methods, or DoDAF-described Models, are valuable for organizing architecture information, but less valuable for communicating that information to stakeholders. As **Figure 1.2-1** illustrates, presentation techniques pull from the architecture information store and display the data to stakeholders.

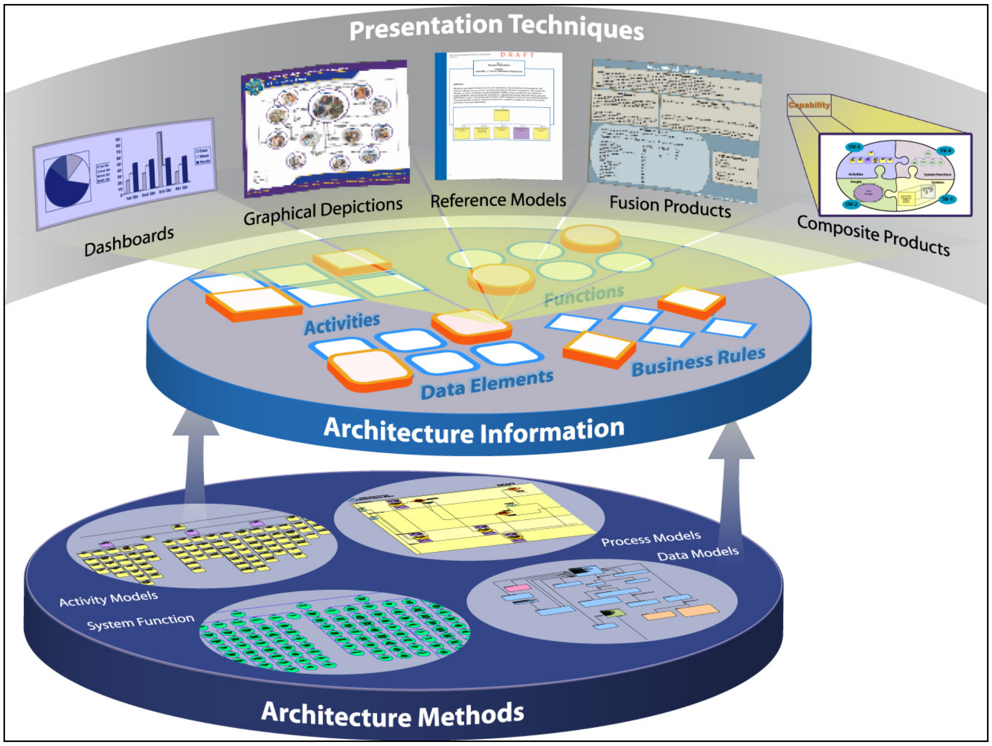


Figure 1.2-1: Relationship of Architecture Methods, Data, and Presentation Techniques

Presentation techniques allow for the communication of many complex or disparate concepts in a context that is meaningful and useful for viewers. Displaying complex information in an effective way can be difficult, but enables the communication and analysis of information. If designed well, a single presentation, a Fit-for-Purpose View, can replace 20 individual documents and display the information with purpose, geared to the targeted stakeholder. This knowledge visualization is accomplished through the use of various techniques, which are each described below.

Information is generally presented in textual documents, with associated, imbedded graphical representations. Specific presentation types are educational syllabi; instruction modules; dashboards on accomplishments or status; and graphical charts, such as pie charts, or bar charts.

It is imperative to realize that when choosing how to present data sets, there is no limit on which presentations (Fit-for-Purpose Views) to use. There are countless ways to display information to decision-makers, and it is up to the presentation developer to determine the most effective way to accomplish this task. The remainder of this document will give a base of presentations to start from, each created to serve its own unique purpose. Details are provided on five different presentation techniques that have proven to be useful in engaging various audiences, and a more comprehensive treatment of presentations will be found online in the DoDAF Journal. The five techniques are as follows:

- **Composite Presentations**: Display multiple pieces of architecture in formats that are relevant to a specific decision-maker.
- **Dashboards**: Integrate abstracted architecture information for a given business context.

- **Fusion Presentations**: Display multiple pieces of architecture and incorporate disparate pieces of information that are not captured within the architecture.
- **Graphics**: Visually represent manipulated data.
- **Reference Models**: Capture the elements of the architecture and translate those elements into text.

The DoDAF-described Models that are available in DoDAF V2.0 are listed in Table 1.2-1. The list provides the possible models and is not prescriptive. The decision-maker and process owners will determine the DoDAF-described Models that are required for their purposes. The DoDAF-described Models are grouped into the following viewpoints:

- All Viewpoint (AV).
- Capability Viewpoint (CV).
- Data and Information Viewpoint (DIV).
- Operational Viewpoint (OV).
- Project Viewpoint (PV).
- Services Viewpoint (SvcV).
- Standard Viewpoint (StdV).
- Systems Viewpoint (SV).

Table 1.2-1: DoDAF V2.0 Models

Models	Descriptions
AV-1: Overview and Summary Information	Describes a 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: 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 Organizational 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 Operational Activities Mapping	A mapping between the capabilities required and the operational activities that those capabilities support.

Table 1.2-1: DoDAF V2.0 Models

Models	Descriptions
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.
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 (usually, very short activities).
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 programs and 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).

Table 1.2-1: DoDAF V2.0 Models

Models	Descriptions
SvcV-4 Services Functionality Description	The functions performed by services and the service data flows among service functions (activities).
SvcV-5 Operational Activity to Services Traceability Matrix	A mapping of services (activities) back to operational activities (activities).
SvcV-6 Services Resource Flow Matrix	It provides details 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 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 Operational Activity to Systems Function Traceability Matrix	A mapping of system functions (activities) back to operational activities (activities).
SV-5b Operational Activity 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).

Table 1.2-1: DoDAF V2.0 Models

Models	Descriptions
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 Description	One of three models used to describe system functionality. It identifies system-specific refinements of critical sequences of events described in the Operational Viewpoint.

Within the DoDAF Meta-model, the elements for the DoDAF-described Models are modeled with time periods (temporal extents) that can be in the future, and the models can be used to describe requirements. A requirement is a two-party agreement, between a requirer and a require-ee. An OV DoDAF-described Model could be used to describe a business process (activity) requirement while an SV DoDAF-described Model might be used to describe a system requirement.

To aid the decision-maker and process owners, the DoDAF-described Models have been categorized into the following types:

- Tabular: Models which present data arranged in rows and columns, which includes structured text as a special case.
- Structural: This category comprises diagrams describing the structural aspects of an architecture.
- Behavioral: This category comprises diagrams describing the behavioral aspects of an architecture.
- Mapping: These models provide matrix (or similar) mappings between two different types of information.
- Ontology: Models which extend the DoDAF ontology for a particular architecture.
- Pictorial: This category is for free-form pictures.
- Timeline: This category comprises diagrams describing the programmatic aspects of an architecture.

DoDAF Architectural Descriptions are expressed in the form of sets of data, expressed as DoDAF-described Models, which can be classified into categories. Table 1.2-2 below provides a summary of how the DoDAF-described Models can be sorted using the categories above and can provide insight for the decision-maker and process owners for the DoDAF-described Models needed.

Table 1.2-2: DoDAF-Described Models Categorized by Type

Category / VP	Tabular	Structural	Behavioral	Mapping	Taxonomy	Pictorial	Timeline
All Viewpoint	AV-1				AV-2		
Capability	CV-1	CV-4		CV-6 CV-7	CV-2		CV-3 CV-5
Operational	OV-3	OV-2 OV-4	OV-6a OV-6b OV-6c		OV-5	OV-1	
System	SV-6 SV-7 SV-9	SV-1 SV-2	SV-4 SV-10a SV-10b SV-10c	SV-3 SV-5a SV-5b			SV-8
Standards	StdV-1 StdV-2						
Data and Information		DIV-1 DIV-2 DIV-3					
Service	SvcV-6 SvcV-7 SvcV-9	SvcV-1 SvcV-2	SvcV-4 SvcV-10a SvcV-10b SvcV-10c	SvcV-3a SvcV-3b SvcV-5			SvcV-8
Project		PV-1		PV-3			PV-2

Some of the DoDAF-described Models above were based on analysis of Ministry of Defence Architecture Framework (MODAF) and North Atlantic Treaty Organization (NATO) Architecture Framework (NAF) views and information requirements provided in the key process workshops [2]by the subject matter experts. In addition, analysis on the DoDAF V1.5 products was performed by the DoDAF V2.0 Presentation Technical Working Group[3]. The objective of the analysis was to determine if any product could be eliminated or if any product was created in every architecture effort. The OV-1 is the most created product at 92 percent of the projects. The SV-7 was the least created product at 5 percent. What is revealing is that there was not a product that could be deleted. The results of the survey are documented in the *DoDAF Product Development Questionnaire Analysis Report.doc* online in the DoDAF Journal.

In addition, based on the level of the architecture effort, the decision-maker and architect need to determine the DoDAF-described Models and Fit-for-Purpose Views needed. To assist,

[2] JCIDS, SE, and Operations workshops were conducted. Other key process workshops, PPBE and Defense Acquisition System (DAS), were not conducted.
[3] The Presentation Technical Working Group reported into the DoDAF Core Management Group and worked with the DoDAF Development Team. The Presentation Technical Working Group focus was on presenting architecture in meaningful ways to the decision-makers.

Table 1.2-3 uses the Zachman Framework[4] with the levels of architecture overlaid for consideration by the decision-maker and architect. Table 1.2-3 is only provided as input; DoDAF is not prescribing DoDAF-described Model or Fit-for-Purpose Views or presentations.

Table 1.2-3: Zachman Framework with Levels of Architecture

Strategic Architectures apply to entire Department | Capability Architectures specific to CPM & Component Tiers | Solution Architecture: Materiel/Non-materiel

DoD EA: Strategic, Capability

	Layer	What (Data)	How (Function)	Where (Network)	Who (People)	When (Time)	Why (Motivation)
1	Scope Context Boundary **(Planner)**	List of things important to the business	List of processes the business performs	List of locations in which the business operates	List of organizations important to the business	List of events significant to the business	List of business goals/ strategies
2	Business Model Concepts **(Owner)**	e.g., Semantic or Entity-relationship Model	e.g., Business Process Model	e.g., Business Logistics System	e.g., Work Flow Model	e.g., Master Schedule	e.g., Business Plan
3	System Model Logic **(Designer)**	e.g., Logical Data Model	e.g., Application Architecture	e.g., Distributed System Architecture	e.g., Human Interface Architecture	e.g., Processing Structure	e.g., Business Rule Model
4	Technology Model Physics **(Builder)**	e.g., Physical Data Model	e.g., System Design	e.g., Technology Architecture	e.g., Presentation Architecture	e.g., Control Structure	e.g., Rule Design
5	Component Configuration **(Implementer)**	e.g., Data Definition	e.g., Program	e.g., Network Architecture	e.g., Security Architecture	e.g., Timing Definition	e.g., Rule Specification
6	Functioning Enterprise Instances **(Worker)**	e.g., Data	e.g., Function	e.g., Network	e.g., Organization	e.g., Schedule	e.g., Strategy

1.2.1 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 questions, *who, what, when, where, why,* and *how*, that facilitate collection and usage of architecture-related data. DoDAF provides a means of answering these interrogatives through the DoDAF Viewpoints and DoDAF-described Models (described further in Volume 2), and the DoDAF Meta-model Data Groups, introduced in Section 9 of Volume 1 as the major parts of the DoDAF Conceptual Data Model (CDM).

Table 1.2.1-1 is a simple matrix that presents the DoDAF Viewpoints and DoDAF-described 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 decisions on how to best represent the data in DoDAF-described Models that are appropriate to the purpose for which the architecture is created.

[4] Zachman, John. Zachman Framework. © Zachman International. The Zachman Framework can be found at the Zachman International Website: http://zachmaninternational.com/index.php/the-zachman-framework/26-articles/13-the-zachman-framework-a-concise-definition

Table 1.2.1-1: Standard Interrogatives Matrix

	What (Date)	How (Function)	Where (Network)	Who (People)	When (Time)	Why (Motivation)
Viewpoint	AV, DIV	OV, SV, SvcV	OV, SV, SvcV	OV	CV, OV, PV, SV, SvcV	AV, CV, OV, StdV, SV, SvcV
DoDAF-described Models	AV-2, DIV-1, DIV-2, DIV-3	OV-5a, OV-5b, OV-6a, b, c, SV-4, SV-10a, b, c, SvcV-10a, b, c	OV-2, SV-2, SvcV-2	OV-2, OV-4	CV-2, CV-4, OV-6c, PV-2, SV-8, SvcV-8, Sv-10c, SvcV-10c	AV-1, CV-1, OV-6a, StdV-1, StdV-2, SV-10a, SvcV-10a
Meta-model group	Information and Data, Project	Activity, Capability, Service, Measures	Location	Performer	All	Rules, Goals

As an example, a decision is required on changing a logistics transaction process (a composite of activities). The process is documented (*how*), to include the measures of performance, services required, and the capability supported by the action (activity). Data required to execute the process (*what*) is collected concurrently. Included in that data collection is the location and other administrative data on the place of process execution (*where*), and the performers of the action (*who*). The time frames required (*when*) and the Rules, Goals, and Expected Results (*why*) are also determined. These interrogatives impact on measures of performance. Each of these interrogatives can be represented by either a DoDAF-described Model or a Fit-for-Purpose View defined by the architectural development team that meets agency requirements. Either way, the models and views needed are created utilizing data defined and derived from the DoDAF Meta-model.

The architecture interrogatives are overlaid on the DM2 Conceptual Data Model in **Figure 1.2.1-2:**

- The Data Description — What (DM2 generalizes to other Resources besides just Data)
- The Function Description — How (and also the Performer that performs the Function, Measures, Rules, and Conditions associated with)
- The Network Description — Where (generalized)
- The People Description — Who (DM2 includes Organizations)
- The Time Description — When
- The Motivation Description — Why (broadened to include Capability requirements)

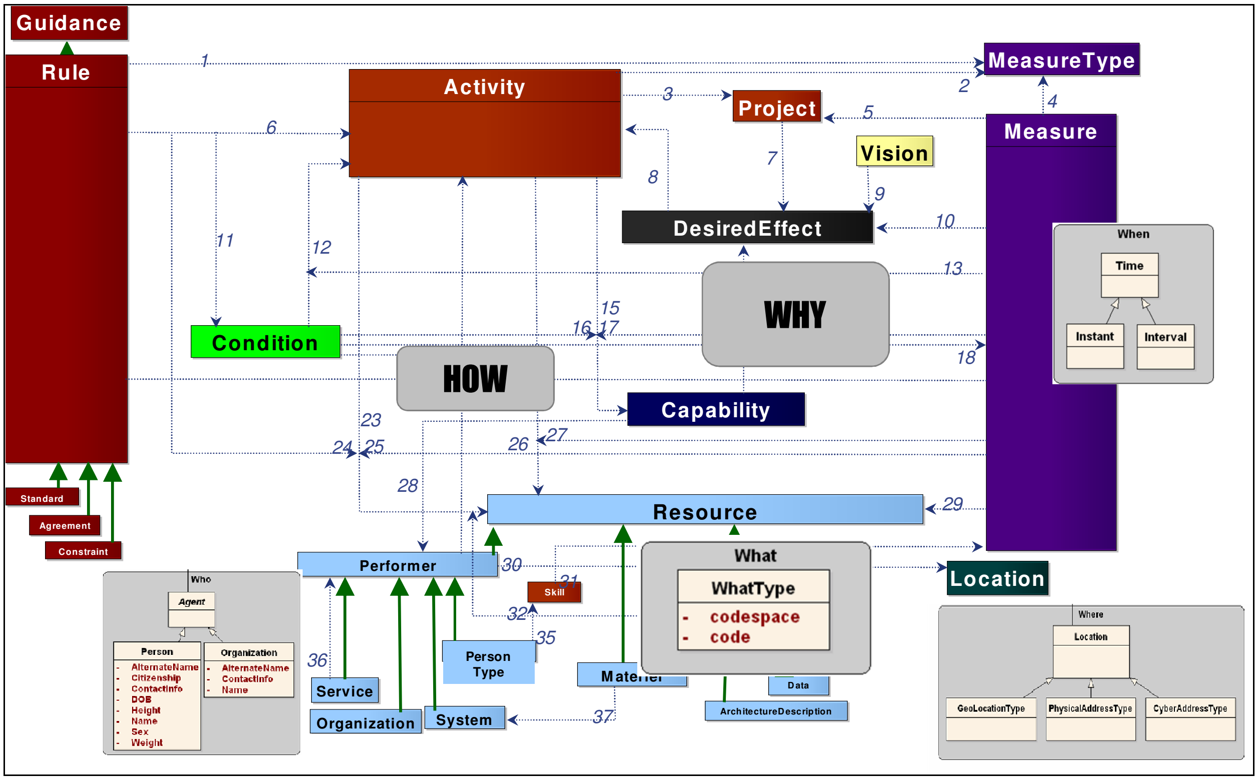


Figure 1.2.1-2: Architecture Interrogative overlay on the DM2 Conceptual Data Model

1.2.2 Architecture Modeling Primitives

Work is presently underway within the Department to ensure uniform representation for the same semantic content within architecture viewing, called Architecture Modeling Primitives. The Architecture Modeling Primitives, hereafter referred to as Primitives, will be a standard set of viewing elements and associated symbols mapped to DM2 concepts and applied to viewing techniques. Use of the Primitives to support the collection of architecture content in concert with the Physical Exchange Specification will aid in generating common understanding and improving communication. As the Primitives concepts are applied to more viewing techniques, they will be updated in the DoDAF Journal and details provided in subsequent releases of DoDAF. When creating an OV-6c in Business Process Modeling Notation (BPMN), the primitives notation may be used. DoD has created the notation and it is in the DoDAF Journal. The full range of Primitives for DoDAF-described Models, as with the current BPMN Primitives, will be coordinated for adoption by architecture tool vendors. Examples of presentations can be viewed online in the public DoDAF Journal.

1.3 What is New for Volume 2

The major changes for DoDAF V2.0 Volume 2 are:

- For the architect, DoDAF V2.0 changes the focus of the Architecture Development Process and is described in Section 1.4, What Does the Architect Need to Do? The basis of the Architecture Development Process is now the Data Meta-model Groups, which are described in Section 2.

- To align with ISO Standards, where appropriate, the terminology has changed from Views to Viewpoint (e.g., the Operational View is now the Operational Viewpoint).
- With the focus on data, DoDAF V2.0 does not have products but has DoDAF-described Models. Rather than the Operational Viewpoint-5 (OV-5) Operational Activity Model Product, there is the Activity Model with the same supporting data. This is shifting the focus of the architecture effort onto the data early in the Architecture Development Process. Volume 2 reflects the shift to the data.
- To support customer requirement and re-organization needs, in Section 3:
  - All the models of data—conceptual, logical, or physical—have been placed into the Data and Information Viewpoint.
  - The Technical Standards Viewpoint has been updated to the Standards Viewpoint and can describe business, commercial, and doctrinal standards, in addition to technical standards.
  - The Operational Viewpoint now can describe rules and constraints for any function (business, intelligence, warfighting, etc.) rather that just those derived from data relationships.
  - Due to the emphasis within the Department on Capability Portfolio Management and feedback from the Acquisition community, the Capability Viewpoint and Project Viewpoint have been added.
- DoDAF can capture the security markings and are documented in Appendix B. In addition, a discussion of the security characteristics mapped to DoDAF Concepts is in Appendix C.
- System has changed from DoDAF V1.5. System is not just computer hardware and computer software. System is now defined in the general sense of an assemblage of components – machine, human – that perform activities (since they are subtypes of Performer) and are interacting or interdependent. This could be anything, i.e., anything from small pieces of equipment that have interacting or interdependent elements, to Family of Systems (FoS) and System of Systems (SoS). Note that Systems are made up of Materiel (e.g., equipment, aircraft, and vessels) and Personnel Types.
- In DoDAF V1.5 and previous versions, Nodes are logical concepts that caused issues in the exchange and discussion of architectures. In one architecture that was reviewed, Operational Nodes mapped to System, Organization, Person Type, Facility, Materiel, and Installation. Within the same architecture, System Node maps to System, Materiel, Organization, and Location. The overlap Organizational and System nodes (System, Organization, Material) illustrates the complexity of trying to define Nodes. The concrete concepts of Node (including Activities, System, Organization, Person Type, Facility, Location, Materiel, and Installation) were incorporated into the DoDAF Meta-model. Since Nodes are logical concepts that could be used to represent the more concrete concepts of activities, systems, organizations, personnel types, facilities, locations, materiels, and installations or combinations of those things, DoDAF V2.0 focuses on those concrete concepts. There will *not* be a mapping of Node to the DoDAF V2.0 Meta-model Groups, concepts, classes, or associations. For the architect, there are some changes in architecture development:
  - When appropriate, DoDAF V1.0 and V1.5 architectures that use the Node concept will need to update the architecture to express the concrete concepts in place of the abstract concept that Node represents. When pre-DoDAF V2.0 architecture is compared with DoDAF V2.0 architecture, the concrete concepts that Node represents must be defined for the newer architecture.

– DoDAF V2.0 architectures will need to express the concrete concepts (activities, systems, organizations, personnel types, facilities, locations, materiels, and installations, etc.).

1.4 What Does the Architect Need to Do?

Using the DoDAF V2.0 Volumes and the DoDAF Journal, the architect needs to perform two key activities:

- Develop the Architectural Description.
- Enable use of the Architectural Description in the solution implementation.

The following subsections describe the architect's activities in more detail.

1.4.1 Develop the Architectural Description

Once the Architectural Description Purpose and Scope are identified, what does the architect need to do? Within the 6-Step Architecture Development Process (described in Volume 1, Section 6.1.1, 6-Step Architecture Development Process), in Step 3 the architect determines the data needed to support the Architectural Description development.

In each step, the Meta-model Groups referred to by the step is that data in the Meta-model Groups in the DoDAF Meta-model contained in this volume. Figure 1.4.1-1 depicts the sub steps that the architect needs to perform within the 6-Step Architecture Development Process. Some of these sub steps are performed in concert with the decision-maker, but the architect has more steps than the decision-maker.

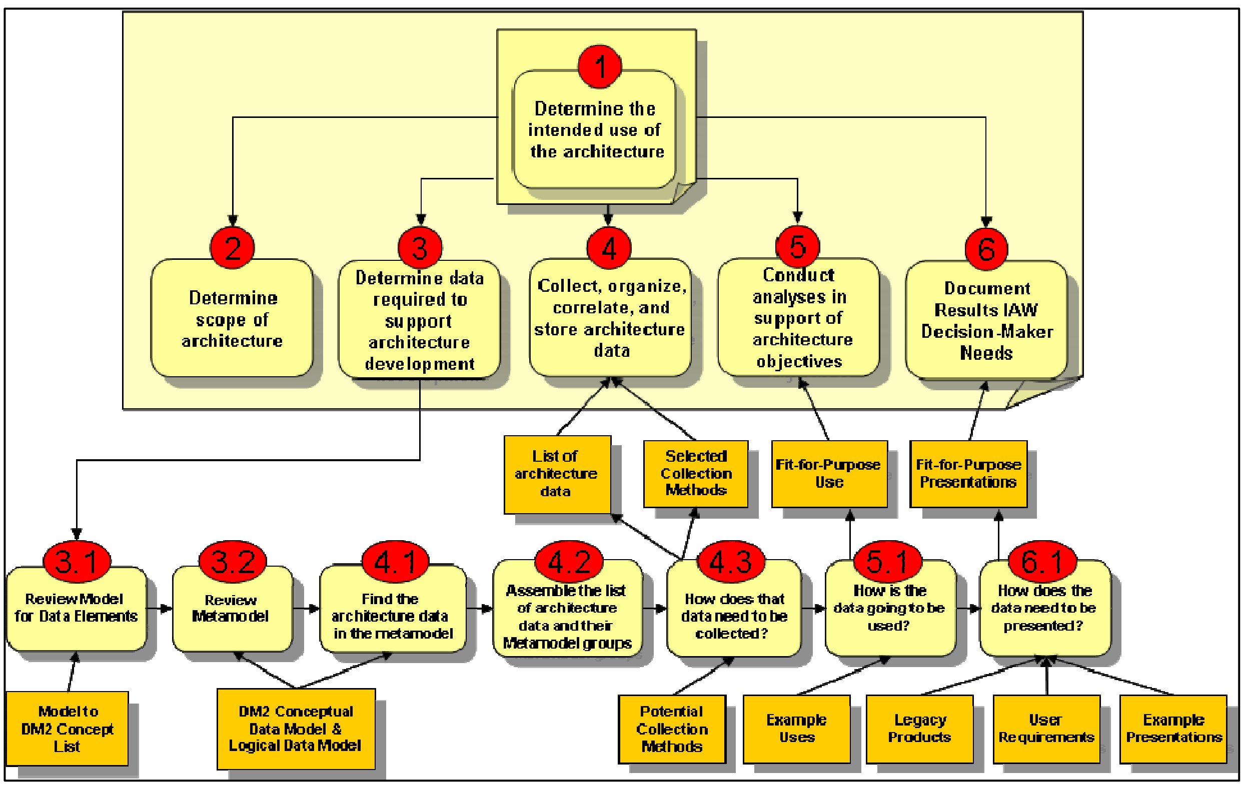


Figure 1.4.1-1: What Does the Architect Need to Do?

The architect's detailed steps, as part of the 6-Step Architecture Development Process are as follows:

- Step 3.1: Using Table B-1, DM2 Concepts, Associations, and Attributes Mapping to DoDAF-described Models in Appendix B, Mappings to DM2 Concepts, the architect determines the DoDAF-described Models needed, based on the concepts required to satisfy the architecture's purpose and scope (from Step 1 and 2 of the 6-Step Architecture Development Process). The architect also determines the Fit-for-Purpose Views needed, also based on the concepts required to satisfy the architecture's purpose and scope.
- Step 3.2: After determining the DoDAF-described Models and Fit-for-Purpose Views required, the architect reviews the:
  - DM2 Conceptual Data Model (described in Volume 1, Section 8.1, The DoDAF Conceptual Data Model)
  - DM2 Logical Data Model (described in Volume 2, Section 2, Meta-model Data Groups)
  - DM2 Concepts, Associations, and Attributes (described in the DoDAF Meta-model Data Dictionary and Table B-1: DM2 Concepts, Associations, and Attributes Mapping to DoDAF-described Models in Appendix B)
- Step 4.1: With the concepts identified in the Architectural Description's Purpose and Scope (from Step 1 and 2 of the 6-Step Architecture Development Process), the required DoDAF-described Models and Fit-for-Purpose Views, the available DM2 metadata, the architect determines the specific architecture DM2 Meta-model Groups, concepts, associations, and attributes that need to be collected for the Architecture Development Process. The tables in the Method subsections of Section 2, Meta-model Data Groups, identify the specific data.
- Step 4.2: The architect assembles the list of required DoDAF-described Models and Fit-for-Purpose Views, DM2 Meta-model Groups, Concepts, Associations, and Attributes. This provides the list of architectural data that needs to be collected, organized, correlated, and stored as part of Step 4 of the 6-Step Architecture Development Process.
- Step 4.3: Using the identified Meta-model Groups in the DM2, the architect determines the method to collect the data. With the specific list of required DoDAF-described Models, Fit-for-Purpose Views, DM2 Meta-model Groups, Concepts, Associations, and Attributes, the architect determines the appropriate collection methods for the "Fit-for-Purpose" needs. Section 2 of this document contains a Method subsection for each Meta-model group which provides potential collection methods. The results of this sub-step should guide the collection methods that will be performed in Step 4 of the 6-Step Architecture Development Process.
- Step 5.1: Using the identified Meta-model Groups in the DM2, the architect determines the usage of the data. With the specific list of required DoDAF-described Models, Fit-for-Purpose Views, DM2 Meta-model Groups, Concepts, Associations, and Attributes, the architect determines the appropriate usage to satisfy the identified "Fit-for-Purpose" needs. Section 2 of this document contains a Use subsection for each of the Meta-model groups which describe uses. The architect needs to determine the "Fit-for-Purpose" use of the architectural data that will meet the decision-maker's purpose and support the decision processes, including the analysis that will need to be performed in Step 5 of the 6-Step Architecture Development Process. The results of this sub step should support the analysis that will be performed in Step 5 of the 6-Step Architecture Development Process. Architectural Description analysis is key to proper use of an architecture by its stakeholders.

Such analysis should be the joint responsibility of the stakeholders and the architect to ensure it answers the stakeholders' questions.

- Step 6.1: Using the identified Meta-model Groups in the DM2, the architect and decision-maker determines the presentations of the data.

  With the specific list of required:

  - DoDAF-described Models
  - Fit-for-Purpose Views
  - DM2 Meta-model Groups
  - Concepts, Associations, and Attributes

  along with the:

  - Legacy Products
  - User Requirements
  - Example Presentations

  The architect and decision-maker determines the appropriate presentations (Fit-for-Purpose Views) and data for the identified "Fit-for-Purpose" needs that will meet the decision-maker's purpose and support their decision processes.

  The results of this sub-step should support the presentations (Fit-for-Purpose Views) that will be created in Step 6 of the 6-Step Architecture Development Process. The DoDAF V2.0 Architecture Development Process for the DoDAF-described Models in the DoDAF Journal presents a non-prescriptive set of tasks to develop DoDAF-described Models in a Microsoft Project Plan.

1.4.2 Using Architectural Metadata

In addition, as the architecture is being developed, architecture metadata can be used (and updated) to support various processes and to populate architecture resources for implementation. One of the Net-Centric Data Strategy goals supported is to enable the architecture to be Discoverable, as a reusable Architecture Resource, mentioned in Section 3.5 in Volume 1. Figure 1.4.2-1 illustrates the potential uses of architecture metadata for the processes they can support and the architecture resources that can be populated from the metadata captured in an architecture repository. It is important to note that architecture metadata can be used throughout the development process, not just at the end of the architecture effort.

The architecture metadata can support:

- Defense Acquisition System process with Project metadata.
- Planning, Programming, Budgeting, and Execution (PPBE) process with Cost metadata
- Information Support Plan (ISP) process with Capability metadata.
- Systems Design and Systems Engineering processes with various metadata, e.g., capability, activity, processes, systems, services, cost, project, data, and taxonomies.
- Service description, service port, and service Resource Flow metadata is used to populate a Service Registry.
- AV-2 metadata is used to create DDMS data catalog entries for authoritative sources.
- Resource Flow and Physical Schema metadata is used to populate the Metadata Registry.
- DoD Information Technology Portfolio Repository (DITPR) population with System data.

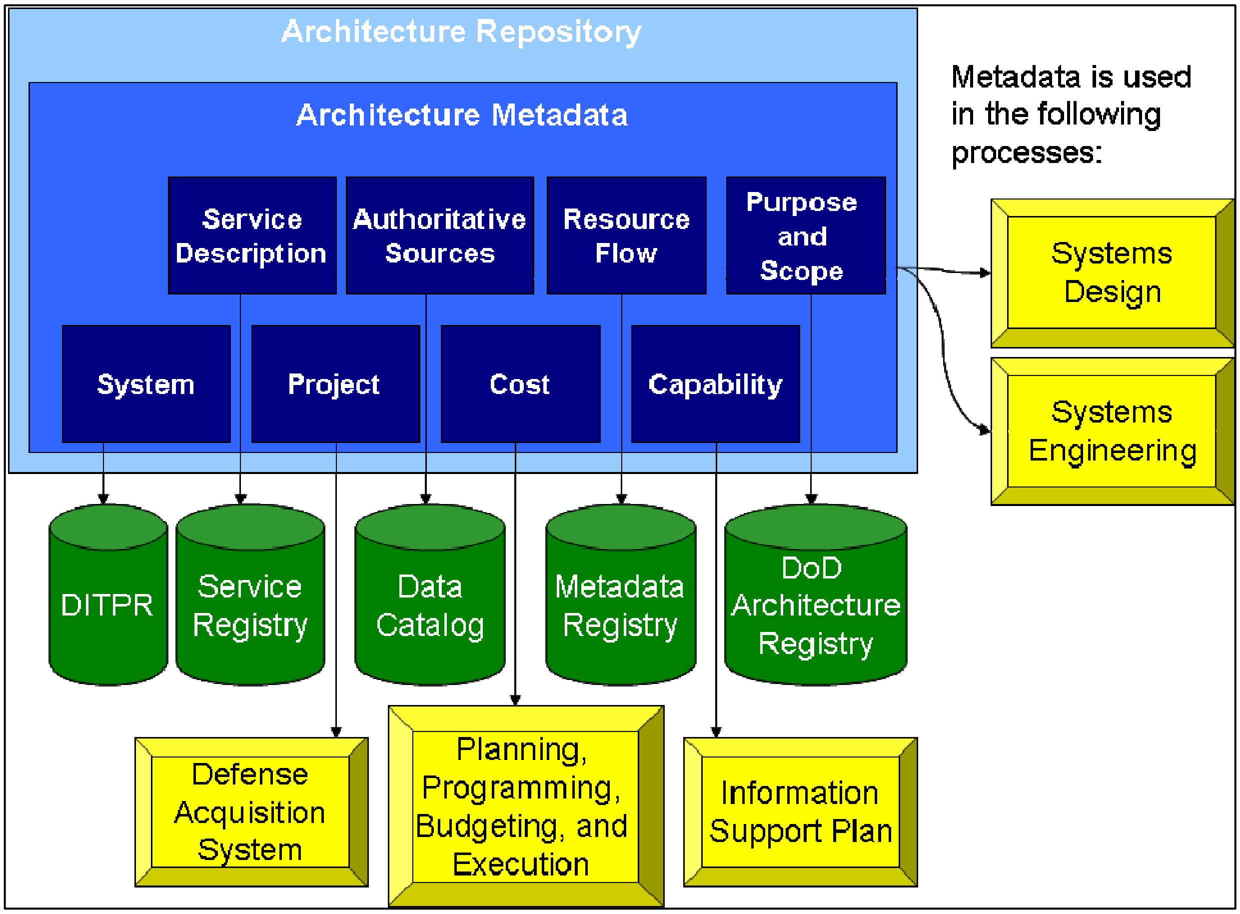


Figure 1.4.2-1: Architectural Metadata Supports Implementation

1.5 What Does the DoD Manager (Decision-maker, Process-Owners, Executive, or Stakeholder) Need to Do?

The DoD Manager identifies the Purpose and Scope for the Architectural Description and gains agreement with the architect. Within the 6-Step Architecture Development Process (described in Volume 1, Section 7.1.1, 6-Step Architecture Development Process), the DoD Manager needs to be involved in the entire process to support the Architectural Description development.

Figure 1.5-1 depicts the sub-steps that the DoD Manager needs to perform in coordination with the architect within the 6-Step Architecture Development Process.

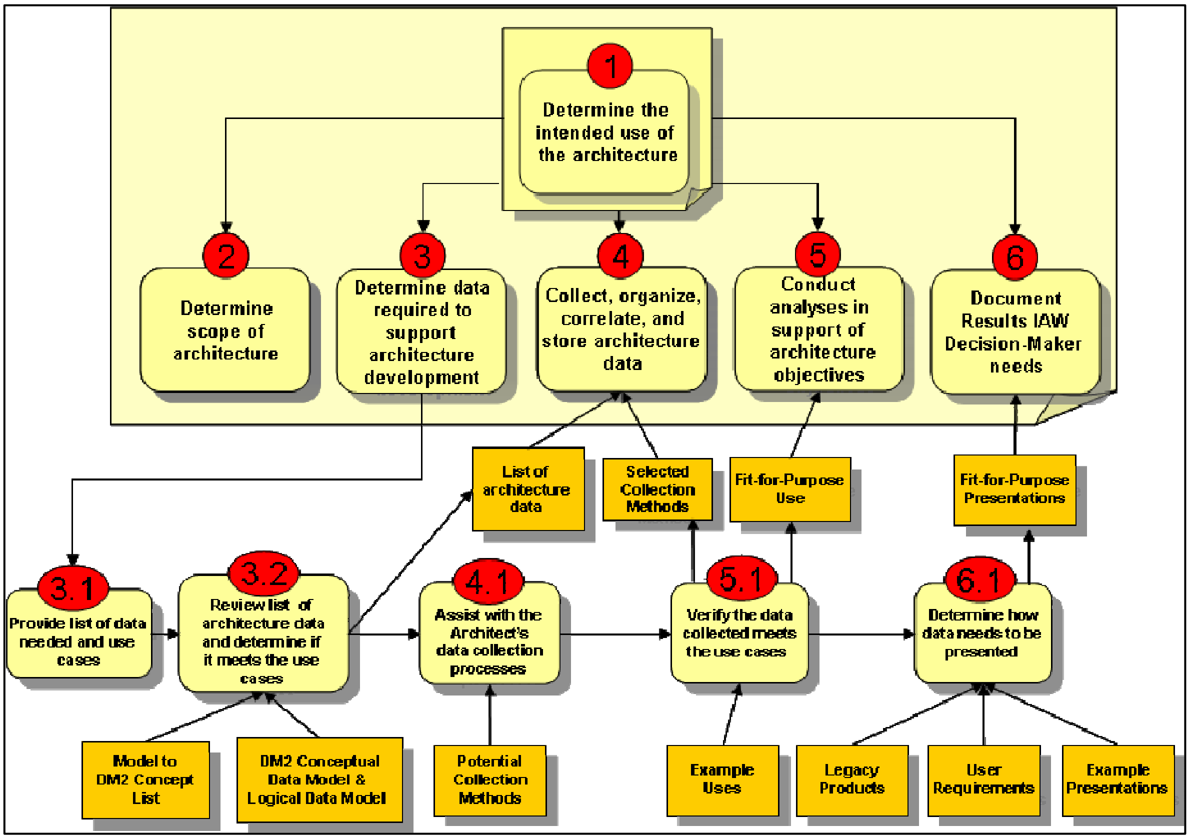


Figure 1.5-1: What Does the Decision-Maker Need To Do?

The detailed steps are:

- Step 3.1: After the DoD Manager has determined the Purpose and Scope as part of Steps 1 and 2 of the Architecture Development Process, the DoD Manager needs to review the Purpose and Scope with the architect. In order for the architecture to be “Fit-for-Purpose”, the DoD Manager needs to provide the list of data needed and the usage of the data (use-cases) to the architect. The DoD Manager, not the architect, is the subject matter expert. The DoD Manager, in concert with the architect, will determine the problem to be solved, the decision to be made, or the data and information to be captured and analyzed. Determining the data needed and the uses is an important responsibility for the DoD Manager and can not be delegated to the architect.
- Step 3.2: The DoD Manager reviews the DoDAF-described Models and Fit-for-Purpose Views, Concepts, Associations, and Attributes that, according to the architect, meet the data requirements and use-cases. The Models, Views, Concepts, Associations, and Attributes required are determined in the architect’s detailed process (Step 3.2) described in Section 1.6 of Volume 2.
- Step 4.1: From the architect’s detailed process (Step 3.5) described in Section 1.6 of Volume 2, the architect determined the appropriate collection methods for the “Fit-for-Purpose” needs. Section 2 of Volume 2 contains a Method subsection for each of the Meta-model groups which provide potential collection methods. The DoD Manager needs to assist or provide the data needed using the architecture collection method.

- Step 5.1: The architect has determined the architectural data that will meet the DoD Manager's purpose ("Fit-for-Purpose") and support their decision processes (use-cases). Section 2 of Volume 2 contains a Use subsection for each of the Meta-model groups which describe example uses. The DoD Manager needs to verify that the data collected meets their needs (use-cases) to support the analysis that will be performed in Step 5 of the 6-Step Architecture Development Process.
- Step 6.1: Based on data collected in Step 4 and the Use-cases, the DoD Manager needs to determine the appropriate presentations for the "Fit-for-Purpose" needs and to support their decision processes. This step should support the presentations that will be created in Step 6 of the 6-Step Architecture Development Process.

1.6 DoDAF Development Guidelines

DoDAF V2.0 provides comprehensive and practical guidance for the creation of architectures that contributes added value for decision-making at whatever level of the DoD they are produced. To this end, DoDAF offers guiding principles in the development of architectures that transcend the tier, level or purpose of the architecture development, and a logical method for executing architecture development for supporting decisions within DoD. DoDAF also offers flexibility in approach, toolset utilization, and techniques (such as structured analysis, object-oriented, and service-oriented).

1.6.1 Guiding Principles

Guiding principles are high-level concepts, which provide a general roadmap for success in architecture development under DoDAF V2.0 as defined in Volume 1, Section 3.5.1. The principles are:

- Architectural Descriptions are built to clearly support the stated objective(s) ("Fit-for-Purpose"). DoDAF offers general direction in the development of architectures so that they can support decisions within DoD. While DoDAF V2.0 describes a number of models and architectural data, diligent scoping of a project and any guiding regulations, instructions, or standard procedures will determine the visualization requirements for a particular architectural effort.
- Architectural Descriptions should be simple and straightforward to achieve their stated purpose. However, the architecture needs reflect the level of complexity required for the stated purpose. Architectural Descriptions should defined by the purpose for their creation. Scoping of a project, as described in Volume 1, Section 7.0 Methodologies, will ensure that the resulting architectural data and derived information, and the models created are consistent with their original purpose.
- Architectural Descriptions should facilitate, not impede communications in decision processes and execution. Architectural Description creation is meant to support decision processes and facilitate improvement of procedures and/or technology in the enterprise. Collection of architectural data and creation of DoDAF-described Models and Fit-for-Purpose Views is intended to support decision-making and to explain critical choices to technical and non-technical managerial staff.
- Architectural Descriptions should be relatable, comparable, and capable of facilitating cross-architecture analysis. Most architectures, except perhaps those at the highest levels of DoD or

an organization, relate on their boundaries to other external processes and operations. When several processes and/or operations are evaluated, compared, or cross-referenced, it should be clear how, where and why data passes among them in similar form.

- Architectural Description should articulate how data interoperability is achieved among federated architectures. To enable federation, DoDAF will provide structures to ensure that horizontal touch-points can be compared for consistency across architecture boundaries. Other mechanisms will ensure that higher tiers have access to data from lower tiers in a form that supports their decision needs. DoDAF utilizes the DoDAF Meta-model, and particularly the Physical Exchange Specification described in Volume 3, as a resource for interoperability.
- Architectural Descriptions should be data centric and tool-agnostic. DoDAF assists in the design of structures that meet specific needs depending on the priorities of individual organizations. In particular, DoDAF calls for the development of integrated, searchable, structured architectural data sets that support analysis targeted to decisions to be made.
- Architectural data should be organized, reusable, and decomposed sufficiently for use by architecture development teams. Collecting and organizing architectural data for use in decision processes should not be over done. The depth and breadth of data collected should be sufficient to capture the major processes actions and not be so broad that the original intent of the architecture project becomes clouded. Whenever possible, data common to other architectures should be used. New data should be created utilizing the structures described in Section 2 and Volume 3 so that, when stored in the DoD Metadata Registry, it becomes available to others with similar requirements.
- Architectural Description development should be guided by the principles and practices of net-centricity to facilitate and support the Net-Centric Strategy of the Department.

Architectural guiding principles enable and facilitate validation and verification activities that will determine the success of the project, and the ability of the resulting architecture to serve the purpose for which it was created. Guiding principles support the more specific goals and objectives of a project as a roadmap.

1.6.2 Multiple Techniques and Toolsets, including Structured and Object Oriented Analysis

DoDAF allows architects to select techniques and toolsets to meet specific needs. While DoDAF describes examples of the application of both Structured Analysis and Design Technique (SADT) and Object-Oriented Analysis and Design (OOAD) technique, it mandates neither. DoDAF explicitly permits any technique that meets the needs of the organization, provides the appropriate architectural data, adheres to the architectural data requirements of parent tiers described in Volume 1, Section 3, and is capable of producing data that can be shared in a federated environment. A brief section on essential toolset attributes desirable for creation of architectures utilizing DoDAF is contained in Volume 1, Section 3.5.3.

2. META-MODEL DATA GROUPS

An overview of the DM2 is contained in Volume 1. This section of Volume 2 presents the logical model -- concepts, attributes, and relationships – that, 1) form the vocabulary for description and discourse about DoDAF-described Models and 2) is the basis for generation of the physical exchange specification for exchange of data between architecture tools and databases.

There are three underlying concepts that were followed in the Development of the DoDAF Meta-model: Principles, grouping of semantically related concepts, and foundation ontology where the properties are inherited by all the DoDAF Concepts. These concepts are discussed below.

The first underlying concept is the DM2 was developed in accordance with the following principles:

- The DM2 models Core Process (PPBE, Defense Acquisition System [DAS], Joint Capabilities Integration and Development System [JCIDS], Capability Portfolio Management [CPM], Systems Engineering [SE], Ops) business objects
- Terms enter the model via thorough semantic research:
  - Assignment to a researcher
  - Collection of authoritative definitions, documenting source
  - Assessment of redundant (alias) or composite terms
  - Formulation/selection of definition based on authoritative definitions
  - Examples
  - Outbrief to team
  - Recording of research and decision rationale
- No need to distinguish or label concepts that differ only in level of aggregation – e.g., subfunction – function. Whole-part relationship covers the need without different names for different types of wholes and parts. When a user has a need to label, the naming pattern accommodates.
- Relationships (associations) should be typed using the foundation.
- There is no commitment to an implementation type. The DM2 should logically support Relational Database Management System (RDBMS), eXtensible Markup Language (XML) Schema Definition (XSD), Java, etc.
- The DM2 is a core that can be extended by user communities; it does not try to cover all user detail. Extenders should be careful to not create redundant representations.
- The model will enter a Configuration Management (CM) process.

Extensions (subtypes (e.g., Unified Modeling Language (UML) specializations), additional attribution, and concepts beyond scope of DM2) to the DM2 are expected and can be done by architecture development efforts. If an extension becomes widespread, it may be appropriate to submit a change request to the DoDAF so that it can be considered by the DoDAF Change Control Board and the Data Technical Working Group for inclusion in the baseline DM2.

The second underlying concept is the grouping of semantically related concepts into the following clusters:

- **Goals.** How goals, visions, objectives, and effects relate and bear on architectures.
- **Capability.** Models of what is needed to perform a set of activities under certain conditions and standards to achieve desired effects and the way in which those needs are satisfied.
- **Activities.** Activities are work that transforms (changes) inputs into outputs or changes their state.
- **Performer.** Things that perform activities such as service performers, systems, personnel, and organizations.
- **Services.** Business and software services, what they do for what effects, by what measures and rules, how they are described for discovery and use, and how and where they can be accomplished.
- **Resource Flows.** The interaction between Activities (which are performed by Performers) that is both temporal and results in the flow or exchange of objects such as information, data, materiel, and performers.
- **Information and Data.** Representations (descriptions) of things of interest and necessary for the conduct of activities.
- **Project.** All forms of planned activities that are responsive to visions, goals, and objectives that aim to change the state of some situation.
- **Training/Skill/Education.** Definitions, descriptions, and the promulgation of training requirements, skills sets required for specific capabilities and operations, and the formal education required
- **Rules.** How rules, standards, agreements, constraints, and regulations and are relevant to architectures.
- **Measures.** All form of measures (metrics) applicable to architectures including needs satisfaction measures, performance measures, interoperability measures, organizational measures, and resource measures.
- **Locations.** All forms of locations including points, lines, areas, volumes, regions, installations, facilities, and addresses including electronic addresses (e.g., Uniform Resource Locators [URLs]) and physical (e.g., postal.)

The data groups are related, as illustrated in **Figure 2-1,** conceptually as is described in the Conceptual Data Model description in Volume 1. They can be roughly grouped as, 1) goals and desired effects (Goals and Capabilities); 2) the actual mission configurations (Activities, Performers, Services, Resource Flows, and Information and Data); 3) the means by which the end items are put in place (Projects and Training / Skills / Education), and 4) the characteristics of the end items (Rules they comply with, Measures associated with them, and where they are Located).

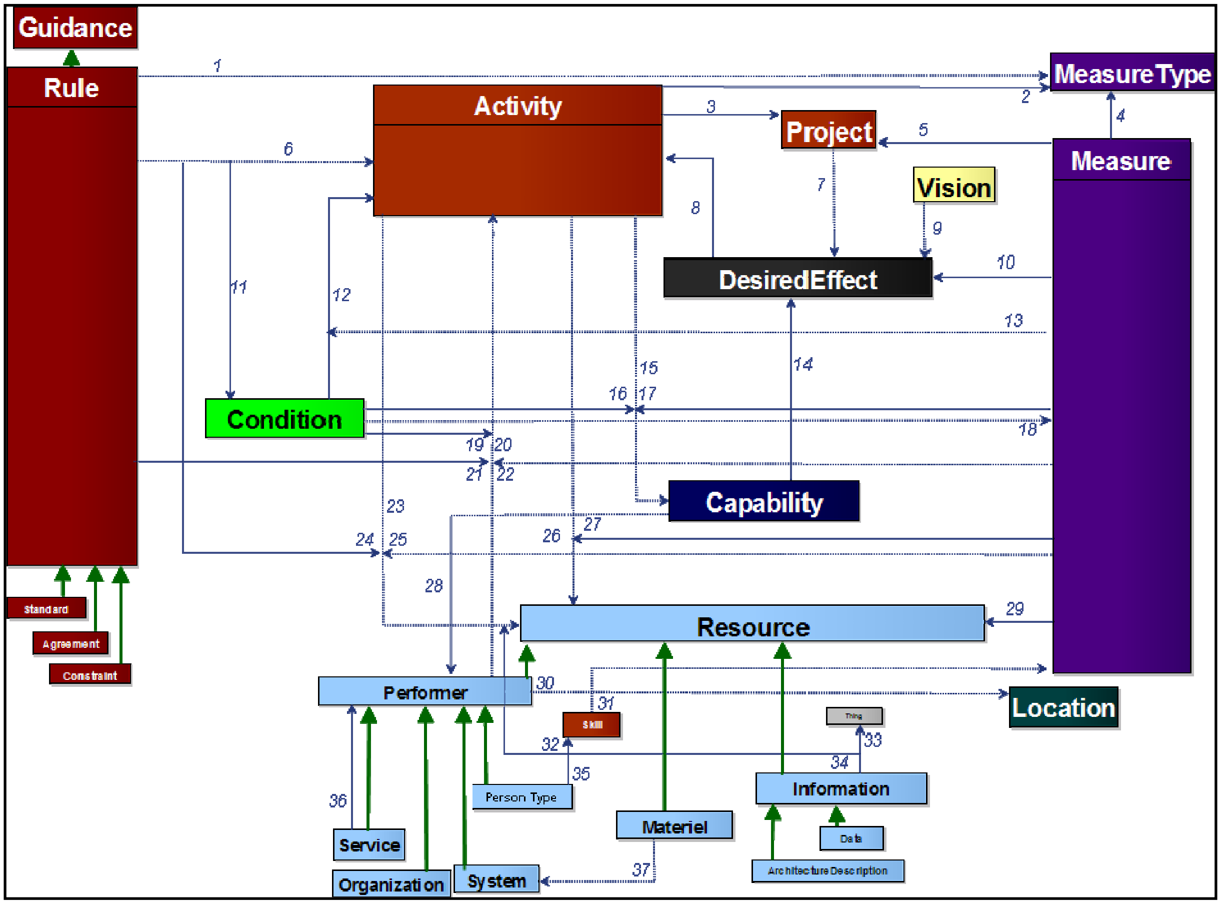


Figure 2-1: DM2 Overview, Showing High-Level Interrelationships Among the Data Groups

The third underlying concept is the root foundation from the International Defence Enterprise Architecture Specification (IDEAS)[5], from which all DoDAF concepts inherit several important properties. None of these foundation properties are unusual; they are all used in reasoning everyday:

- Individuals, things that exist in 3D space and time, i.e., have spatial-temporal extent.
- Types, sets of things.
- Tuples, ordered relations between things, e.g., ordered pairs in 2D analytic geometry, rows in relational database tables, and subject-verb-object triples in Resource Description Framework.
- Whole-part; e.g., components of a service or system, parts of the data, materiel parts, subdivisions of an activity, and elements of a measure.
- Temporal whole-part; e.g., the states or phases of a performer, the increments of a capability or projects, the sequence of a process (activity).

[5] http://www.ideasgroup.org

- Super-subtype; e.g., a type of system or service, capability, materiel, organization, or condition.
- Interface; e.g., an overlap between two things.

The foundation is usually called a formal ontology. It is a formal, higher-order, 4D, based on four dimensionalism[6,7]. It is extensional (see Extension [metaphysics]), using physical existence as its criterion for identity. In practical terms, this means the ontology is well suited to managing change-over time and identifying elements with a degree of precision that is not possible using names alone. The methodology for defining the ontology is very precise about criteria for identity by grounding reasoning about whether two things are the same using something that can be accurately identified. So, comparing two individuals, if they occupy precisely the same space at the same time, they are the same. Clearly this only works for individuals, but the principle can be used to compare types too. For two types to be the same, they must have the same members. If those members are individuals, their physical extents can be compared. If the members are types, then the analysis continues until individuals are reached, then they can be compared. The advantage of this methodology is that names are separated from things and so there is no possibility of confusion about what is being discussed. The upper foundation is shown in **Figure 2-2.**

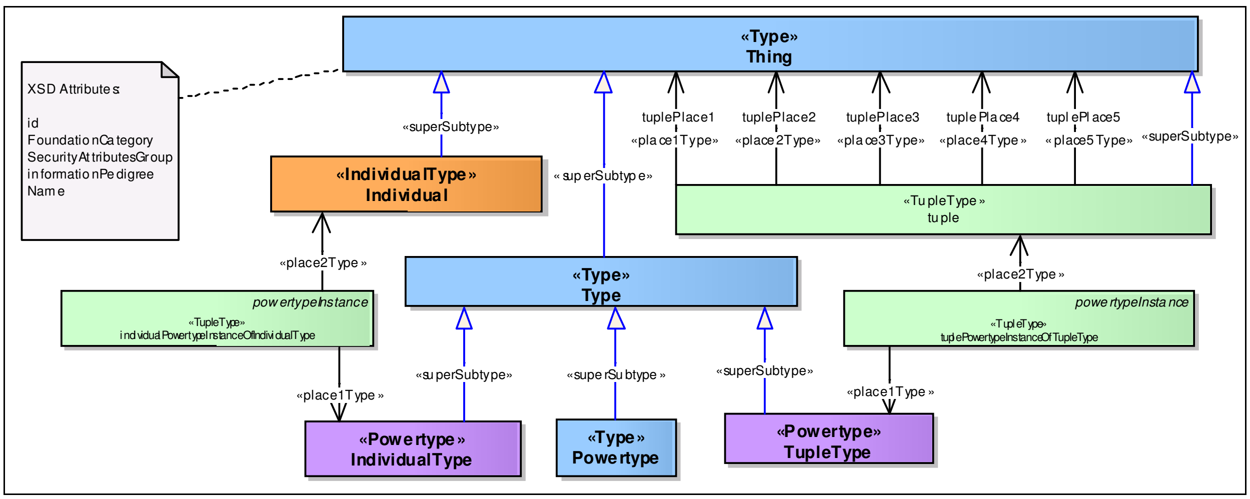


Figure 2-2: Foundation Top-Level

Several items are notable:

- There are three subtypes of Thing: 1) Individuals meaning Things that have spatio-temporal extent, i.e., that exist in space and time – can be kicked; 2) Types or sets of Things; and 3) Tuples or ordered relations between Things.
- Types include sets of Tuples and sets of sets.
- Tuples can have other Tuples in their tuple places.

[6] Rea, M. C., "Four Dimensionalism" in The Oxford Handbook for Metaphysics. Oxford Univ. Press
[7] Sider, Theodore. "Four Dimensionalism". Philosophical Review. pp. 197-231

- There are three subtypes of Type: 1) Individual Type, sets whose members are Individuals (Things with spatio-temporal extent); Power Types, sets whose members are generated from a powerset on some other set; and 3) Tuples, sets of ordered relations between Things.
- The participants in a super-subtype relationship can be from the same class, e.g., the supertype can be an instance of Measure Type as well as the subtype. This allows for representation of as much of a super-subtype taxonomy as is needed.
- Power Type members are generated from some Type by taking all the possible subsets of the members of the Type. For example consider the Type whose members are a, b, c. The powerset would be:

  $$\{a,b,c\},\{a,b\},\{a,c\},\{b,c\},\{a\},\{b\},\{c\},\{\varnothing\}$$

- For example, take the Individual Type AIRCRAFT, whose members include all the aircraft of the world. The powerset generated from this set would have:

  $$\{a_1,a_2,...,a_n\},\{\varnothing\}$$

  $$\{\text{F-15}_1,\text{F-15}_2,...,\text{F-15}_{lastF-15built}\}$$

  $$\{\text{F-15}_1,747_1,...,\text{Cessna}_1\}$$

- Some of these subsets are not used by anyone, e.g., the full set, the null set, or just some random subset. However, the second one, which might be name F-15 Type, is quite useful. The last example is not useful to most unless you are interested in the first (assuming the subscript 1 means first) of any particular aircraft type, e.g., if you were doing a study of first-off-the-line aircraft production lessons-learned. This is the usefulness of Power Types and why they are employed in DM2: they allow for multiple categorization schemes, according to someone else's use, yet traceability back to the common elements so that the relationships between multiple categorization schemes can be understood. This was a DM2 requirement – multiple categorization schemes or taxonomies – because across a large enterprise it is not possible to employ a single categorization scheme; rather schemes vary depending on function. For example, a weaponeer's classifies ordnance is naturally different from a logistician's, the former concerned with delivery means, lethality, etc. and the latter with weight, size, and other transportation issues.
- Note also that a powerset can then be taken of the powerset. This allows for build up of what is often called a taxonomic hierarchy. These are quite useful in enterprise Architectural Descriptions.

The DM2 utilizes the formal ontology of IDEAS because it provides:

- Mathematical rigor needed for precision Architectural Descriptions that can be analyzed and used in detailed processes such as Systems Engineering and Operations Planning.
- Reuse of common patterns to economize the model and implementations.
- Improved interoperation with Unified Profile for DoDAF and MODAF (UPDM)-SysML tools which are following IDEAS concepts.

- Improved opportunities for Coalition and NATO data exchange since MODAF is following IDEAS and NAF is interested in following IDEAS.

The re-use patterns useful to Architectural Descriptions are shown in **Figure 2-3.**

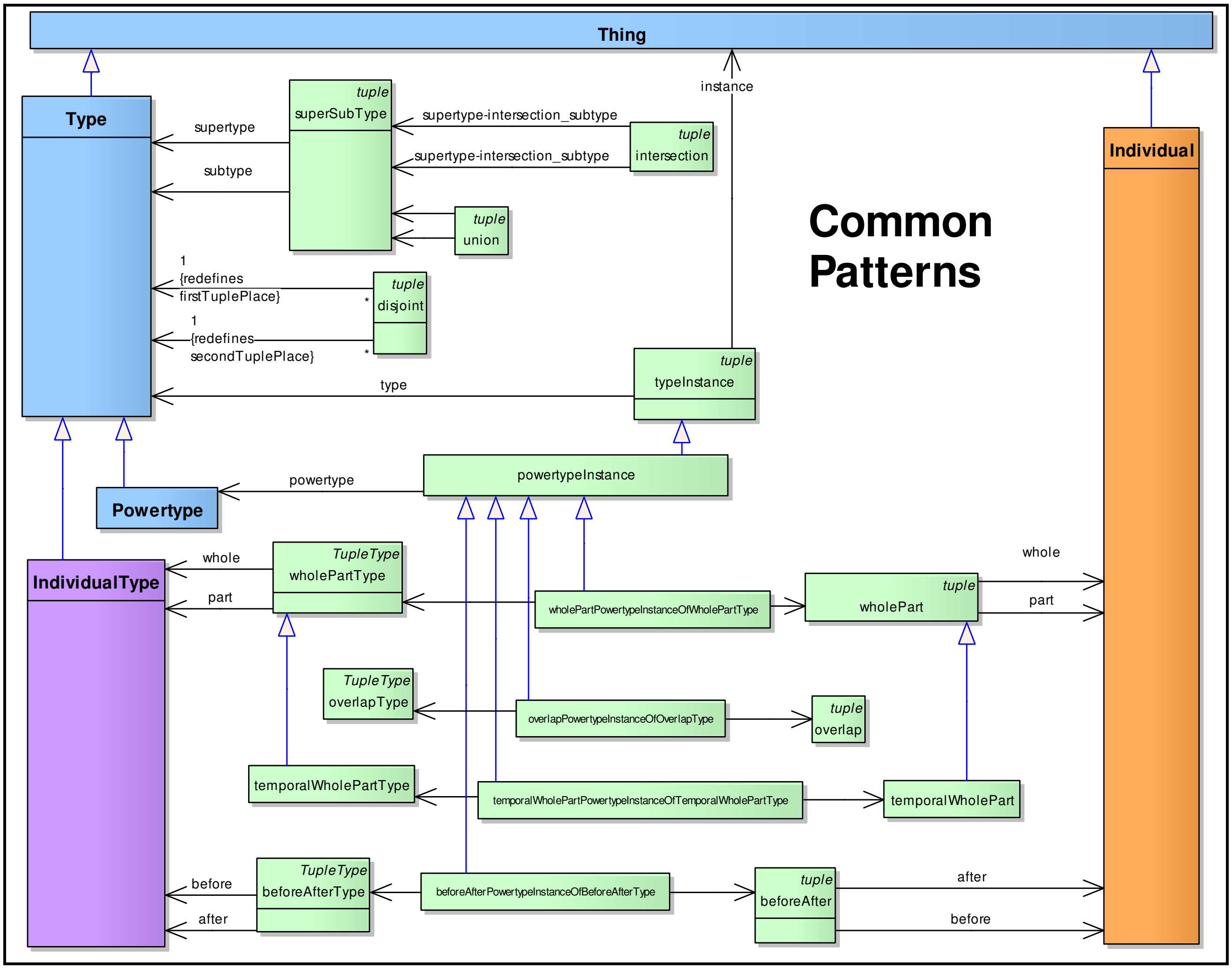


Figure 2-3: DM2 Common Patterns

The DM2 made some ease-of-use modifications to the formalism and naming convention in IDEAS as follows:

- In DM2, all Individuals (Things with spatial and temporal extent – things you can kick) and their Types are States, i.e., the whole-life Individual is just a special state case, that is, where the temporal extent is the Individual's start and end times. The names of the concepts do not include the word State because in all cases where it is applicable, it is implied.
- Since most architectural concern is with types of things, rather than specific individual things (e.g., not a specific President or System), the IDEAS convention of appending Type to the name was left off. In cases where both specific (individual) things and types are useful in DoDAF architectures, an appendage of Individual or Type is made to the less prevalent case.

- Detailed formal modeling of Tuple Types, Numerals, and Symbols is assumed. This detail is proper formalism but, once worked out, does not need to be included in the domain modeling of the DM2.
- Several names were changed due to familiarity in the United States (U.S.) DoD. This was expected in IDEAS and is one use of the Naming pattern. An example is Agent, which the DM2 Technical Working Group (TWG) felt should be called Performer. These are all simple aliases. National aliasing was understood as a requirement at the start of IDEAS; the naming pattern was developed in part to satisfy that requirement. Using the naming pattern, simple aliases are easily accommodated.
- IDEAS Proper Overlap required a cardinality constraint, that is, two overlap Part tuples were required. One represented the part of individual A that overlapped with individual B; the other represented the part of individual B that overlapped with individual A. In addition, it was required the two parts (the part of A, the part of B) equal. For DM2, it was simplified this by removing the unenforceable constraints by re-modeling overlap as a couple of couples where each couple is a whole-Part, one of Individual A and its part, the other of Individual B and its part. This is easily interoperable with IDEAS but is simpler to implement since there are no informal constraints.
- Security classification and information pedigree were added a core attributes, to apply to any element of data. This was done to follow DoD’s Net-Centric Data Strategy.
- Some IDEAS concepts are left out because their exact mathematical meaning has not yet been modeled by the IDEAS Group.
- Agent Capable of Responsibility. Although both the IDEAS Group and the DM2 TWG feel there is a sense of distinction between Agents (Performers) in general and Agents capable of responsibility, the actual mathematical distinction has not yet been modeled in IDEAS. Both groups believe a mathematical distinction exists but it involves more research in the nature of responsibility to complete.

The IDEAS foundation concepts, common to all data groups are shown in **Table 2-1**. It is important to remember that even though these are not repeated in the descriptions of the data groups, they are nevertheless present in the model and apply to the data group concepts according to the Doman Class Hierarchy shown in Figure 2-4.

Table 2-1: IDEAS Foundation Concepts Applicable to all DoDAF Data Groups

IDEAS Concept	Definition
Classes	
endBoundary	The maximum time value of a temporal extent.
endBoundaryType	The maximum value of a temporal extent taken over a Type, i.e., the maximum time value taken over all it's members.
Individual	A Thing that has spatio-temporal extent. Note - this may be something that existed in the past, exists now, or may exist in some future possible world.
IndividualType	The powertype of Individual.

Table 2-1: IDEAS Foundation Concepts Applicable to all DoDAF Data Groups

IDEAS Concept	Definition
Information	Information is the state of a something of interest that is materialized -- in any medium or form -- and communicated or received.
InformationType	Category or type of information
Name	The type of all utterances of a given name for a Thing. The exemplarText provides a written example of the uttered name.
NamingScheme	A Type whose members are Names. What kind of name the name is.
Powertype	A Type that is the set (i.e., Type) of all subsets (i.e., subTypes) that can be taken over the some Type.
startBoundary	The beginning of a temporalBoundary.
temporalBoundary	The start and end times for the spatio-temporal extent of an Individual
temporalBoundaryType	The start and end times for the Individual members of a Type.
Thing	The union of Individual, Type, and tuple.
TupleType	The powertype of tuple that provides the stereotype for tuples of Types.
Type	A set (or class) of Things. Note1: Types are identified by their members (i.e. all the things of that type). Note2: The IDEAS Foundation is a higher-order ontology, so Types may have members that are also Types.
Associations	
beforeAfter	A couple that represents that the temporal extent end time for the individual in place 1 is less than temporal extent start time for the individual in place 2.
beforeAfterPowertypeInstanceOfBeforeAfterType	beforeAfter is a member of BeforeAfterType
beforeAfterType	An association between two Individual Types signifying that the temporal end of all the Individuals of one Individual Type is before the temporal start of all the Individuals of the other Individual Type.
couple	An ordered relationship (tuple) between two Things, i.e., that has two place positions.
couplePowertypeInstanceOfCoupleType	couple is a member of CoupleType
coupleType	A couple in which the places are taken by Types only.
describedBy	A tuple that asserts that Information describes a Thing.
disjoint	Asserts that two Types define disjoint sets (i.e. they share no common members).
endBoundaryPowertypeInstanceOfEndBoundaryType	endBoundary is a member of EndBoundaryType
endBoundaryTypeInstanceOfMeasure	endBoundary is a member of Measure
endBoundaryTypeTypeInstanceOfMeasure	endBoundaryType is a member of Measure
individualPowertypeInstanceOfIndividualType	individual is a member of IndividualType
informationPowertypeInstanceOfInformationType	information is a member of InformationType
intersection	A couple of couples where each constituent couple represents the subset that is common to both sets.

Table 2-1: IDEAS Foundation Concepts Applicable to all DoDAF Data Groups

IDEAS Concept	Definition
namedBy	A couple that asserts that a Name describes a Thing.
namePowertypeInstanceOfNamingScheme	Name is a member of NameType
overlap	A couple of wholePart couples where the part in each couple is the same.
overlapPowertypeInstanceOfOverlapType	overlap is a member of OverlapType
overlapType	An overlap in which the places are taken by Types only.
powertypeInstance	An association that between of the sets within the powerType and the powerType. A special form of typeInstance.
startBoundaryPowertypeInstanceOfStartBoundaryType	startBoundary is a member of startBoundaryType
startBoundaryType	The beginning of a temporalBoundaryType.
startBoundaryTypeInstanceOfMeasure	startBoundary is a member of Measure
startBoundaryTypeTypeInstanceOfMeasure	startBoundaryType is a member of Measure
superSubType	An association in which one Type (the subtype) is a subset of the other Type (supertype).
temporalBoundaryPowertypeInstanceOfTemporalBoundaryType	temporalBoundary is a member of temporalBoundaryType
temporalWholePart	A wholePart that asserts the spatial extent of the (whole) individual is co-extensive with the spatial extent of the (part) individual for a particular period of time.
temporalWholePartPowertypeInstanceOfTemporalWholePartType	temporalWholePart is a member of temporalWholePartType
temporalWholePartType	A couple between two Individual Types where for each member of the whole set, there is a corresponding member of the part set for which a wholePart relationship exists, and conversely
tuple	A relationship between two or more things. Note: Tuples are identified by their places (i.e. the ends of the relationship).
tuplePowertypeInstanceOfTupleType	tuple is a member of TupleType
typeInstance	A Thing can be an instance of a Type - i.e. set membership. Note that IDEAS is a higher-order model, hence Types may be instances of Types.
union	A couple of couples where each constituent couple represents the superset union over the unioned sets.
wholePart	A couple that asserts one (part) Individual is part of another (whole) Individual.
wholePartPowertypeInstanceOfWholePartType	wholePart is a member of wholePartType
wholePartType	A coupleType that asserts one Type (the part) has members that have a whole-part relation with a member of the other Type (whole).

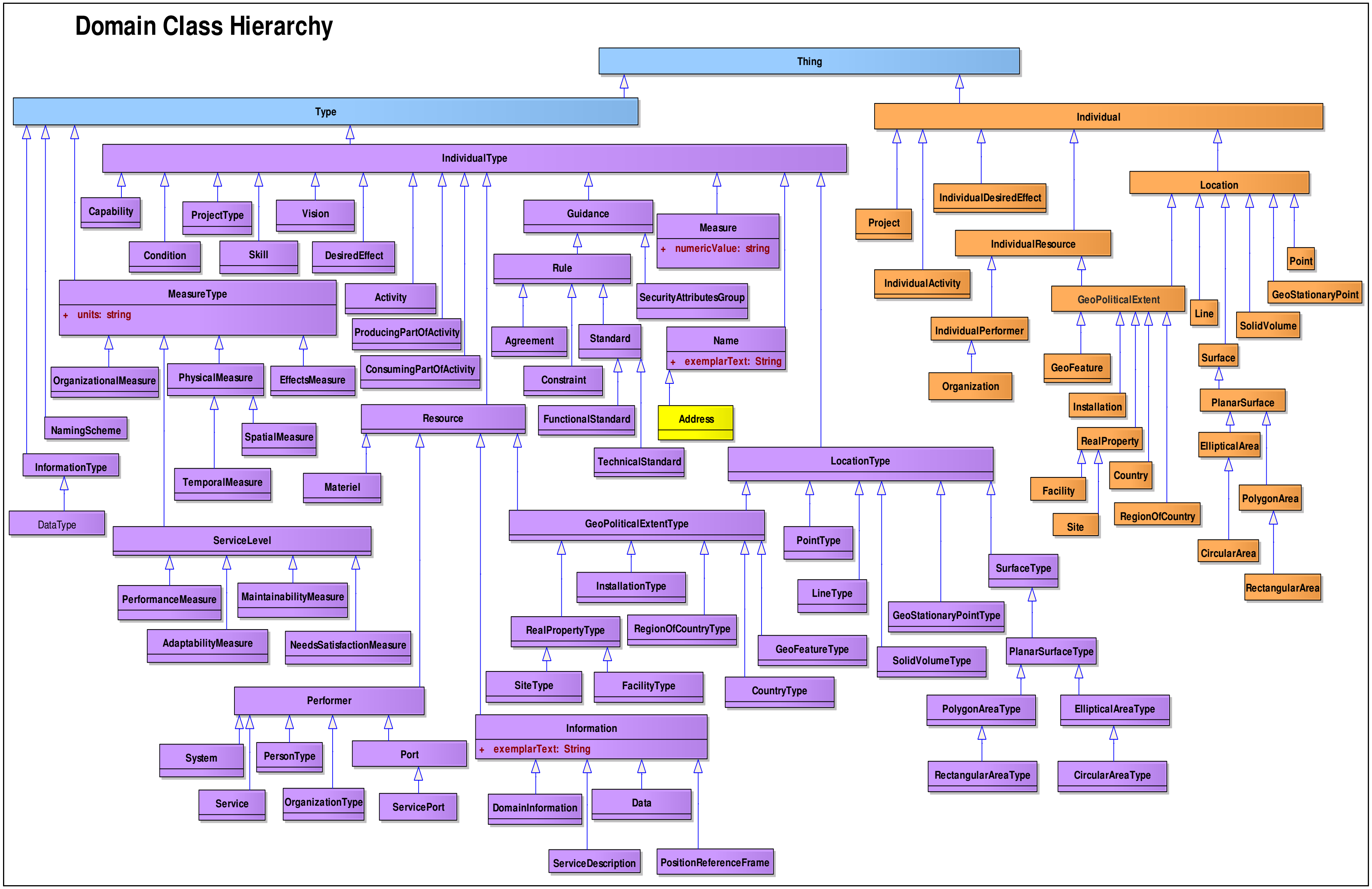


Figure 2-4: DM2 Domain Class Hierarchy

The IDEAS Model is represented in UML. The UML language is not ideally suited to ontology specification in its native form. The UML language can be extended through the use of profiles. The IDEAS Model has been developed using a UML Profile - any UML elements that are not stereotyped by one of the IDEAS foundation elements will not be considered part of an IDEAS ontology. The IDEAS Foundation specifies the fundamental types that define the profile stereotypes. The super-subtype structure in IDEAS is quite comprehensive, and showing the super-type relationships on some diagrams can result in a number of crossed lines. In these cases, supertypes of a given type will be listed in italic text in the top-right-hand corner of the UML element box.

The stereotype of an element in an IDEAS UML model is shorthand for the element being an instance of the type referred to by the Stereotype, though the type must be one that has been defined in the root package of the foundation. Hence, if the stereotype is <<Individual>> then the element is an instance of an Individual. The following stereotyped classes, with their color-coding are used in the model:

1. <<Individual>> An instance of an Individual - something with spatio-temporal extent [Color Name: Grey(80%), Color Codes: R40 G40 B40]
2. <<Type>> The specification of a Type [Color Name: Pale Blue, Color Code: R153 G204 B255]
3. <<IndividualType>> The specification of a Type whose members are Individuals [Color Name: Light Orange, Color Codes: R255 G173 B91]
4. <<TupleType>> The specification of a Type whose members are tuples [Color Name: Light Green, Color Codes: R204 G255 B204]
5. <<Powertype>> The specification of a Type that is the set of all subsets of a given Type [Color Name: Lavender, Color Codes: R204 G153 B255]
6. <<Name>> The specification of a name, with the examplar text provided as a tagged value [Color Name: Tan, Color Codes: R255 G254 B153]
7. <<NamingScheme>> The specification of a Type whose members are names [Color Name: Yellow, Color Codes: R255 G255 B0]

The following stereotyped relationships are used in the model:

1. <<typeInstance>> a relationship between a type and one of its instances (UML:Dependency) [Color Name: Red, Color Codes: R255 G0 B0]

2. <<powertypeInstance>> a relationship between a type and its powerset (UML:Dependency) [Color Name: Red, Color Codes: R255 G0 B0]

3. <<nameTypeInstance>> a relationship between a name and its NameType (UML:Dependency) [Color Name: Red, Color Codes: R255 G0 B0]

4. <<super-subtype>> a relationship between a type and one of its subtypes (UML:Generalisation) [Color Name: Blue, Color Codes: R0 G0 B255]

5. <<wholePart>> a relationship between an individual and one of its parts (UML:Aggregation) [Color Name: Green, Color Codes: R0 G147 B0]

6. <<namedBy>> a relationship between a name and the thing it names [Color Name: Black, Color Codes: R0 G0 B0]

7. <<tuple>>/<<couple> a relationship between a things (UML:n-ary relationship diamond) [Color Name: Grey(80%), Color Codes: R40 G40 B40]

Some examples are depicted in [Figure 2-5](#):

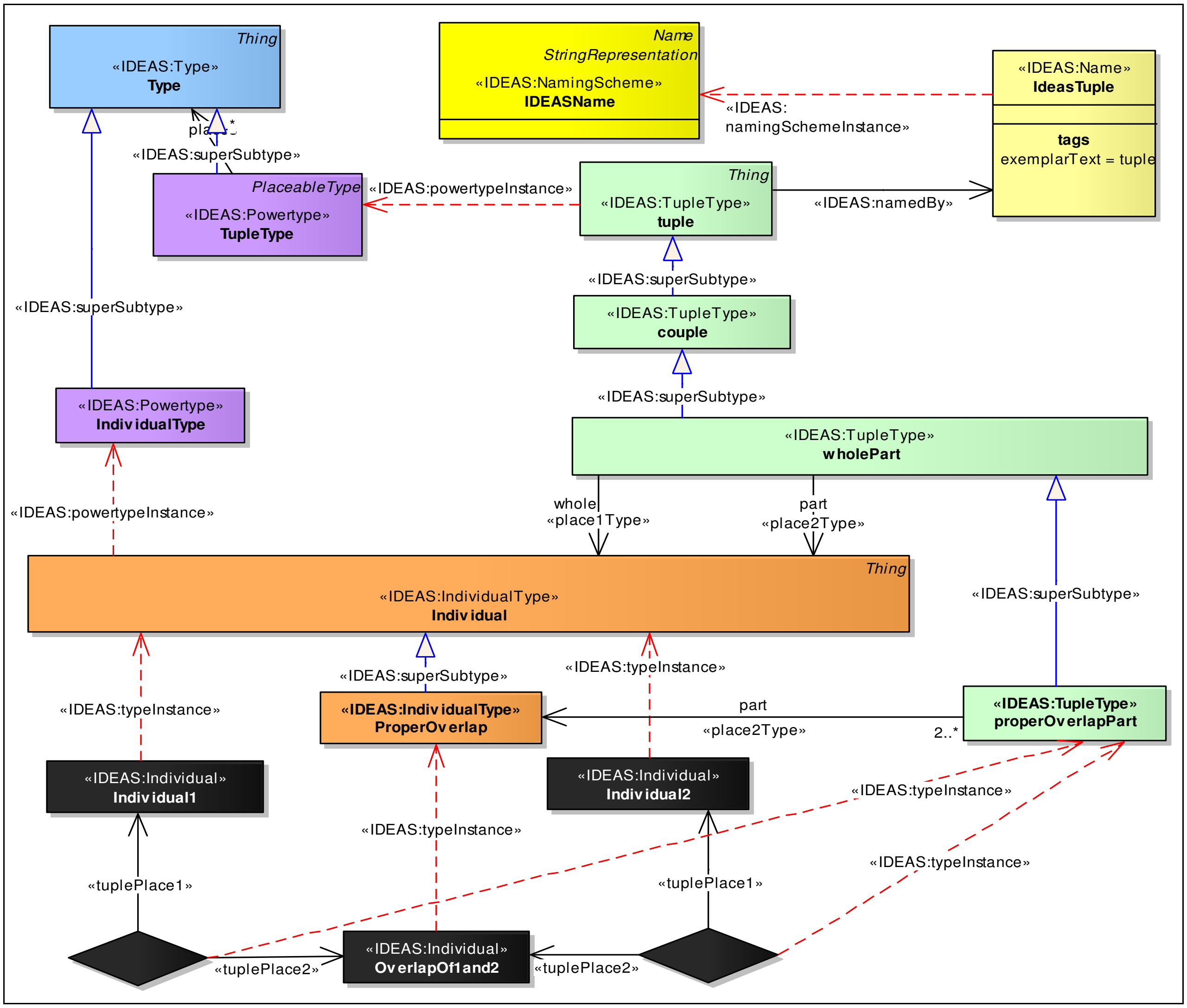


Figure 2-5: UML Examples with Color-Coding

The naming convention for classes, attributes, and association names is camel case as follows:

- Class names start with uppercase.
- Attributes and association names start with lowercase.
- Acronyms are all uppercase. Acronyms in the middle of a name are avoided because of the concatenation of the acronym uppercase and the succeeding string leading uppercase.

Note that the size of the icons is not indicative of their importance; the sizes are adjusted to reduce line crossings and bends to make the diagrams easier to understand.

The following subparagraphs describe each of the data groups, how such data is collected and put together, and how it can be used.

2.1 Performers

Performer is a class of entities that are central to the description of architecture. It is the *Who* in the Architectural Development Process. The *How,* tasks, activities, and processes (composite of activities), are assigned to Performers to accomplish the desired outcome. Performers are further subdivided and allocated to organizations, personnel and mechanization. Rules, locations and measures are then applied to organizations, personnel and mechanization. Within this assignment and allocation process there are many major tradeoff opportunities. Automation (mechanization versus people) tradeoffs, analysis for items such as performance and cost/benefit are involved in the process. When these tradeoffs and associated decisions are sufficiently mature, an allocated baseline can be declared and initial work breakdown structures refined.

2.1.1 Data

The DoDAF Meta-model for the data comprising Performers is shown in **Figure 2.1.1-1**. The figure may be hard to read in a hardcopy printout but a version at full-resolution which can be zoomed in is published in the online DoDAF V2.0 Meta-model Data Dictionary. Definitions for the model terms are in **Table 2.1.1-1** along with summary of aliases and composite term definitions in **Table 2.1.1-2**. Authoritative Source definitions, aliases, and rationale are provided in the DoDAF V2.0 Meta-model Data Dictionary. Note that foundational classes are generally not shown on data group diagrams; this foundational material is in Section 2. This includes super-subtype, whole-part, temporal-whole-part, overlap, type-instance (member-of), and before-after patterns. Also not shown are the data structures for classification marking of architecture information at the whole and element (portion) levels using the Intelligence Community – Intelligence Standard Markings (IC-ISM). The schema for the IC-ISM is in Volume 3. Lastly, note that the size of the icons is not indicative of their importance; sizes are adjusted to reduce line crossings and bends to make the diagrams easier to understand.

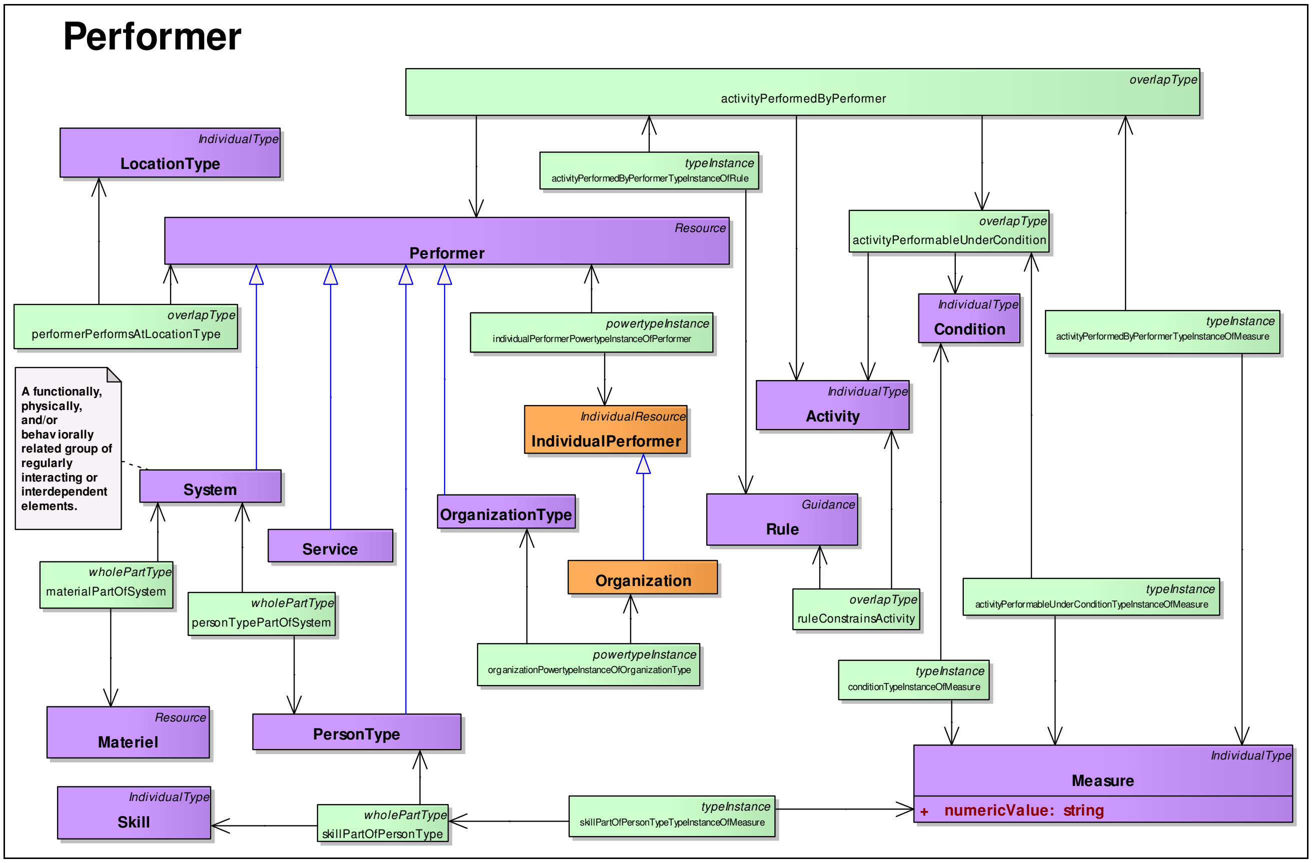


Figure 2.1.1-1: DoDAF Meta-model for Performers

a. The first thing to note about Performer is that it can represent:

   1) A Personnel Type such as described by the Amy's Military Occupational Specialties (MOS). MOS describe Skills and their measurement (not shown in this diagram).

   2) An Organization (type or actual Individual Organization) meaning a mission chartered organization, not limited to just collections of people or locations, e.g., the Federal Bureau of Investigation (FBI) has a chartered mission and it chooses the locations, people, etc., to accomplish such.

   3) A System in the general sense of an assemblage of components – machine, human – that accomplish a function, i.e., anything from small pieces of equipment to FoS and SoS. Note that Systems are made up of Materiel (e.g., equipment, aircraft, and vessels) and Personnel Types, and organizational elements.

   4) A Service, from a software service to a business service such as Search and Rescue.

   5) Any combination of the above.

b. The performance of an Activity by a Performer occurs in physical space and time. That is, at some place and time, the Activity is conducted. This is referred to as a spatial-temporal overlap, simply meaning that the Activity and Performer overlap in space and time. There are two ways in which a Performer spatial-temporally overlaps an Activity:

   1) In the act of performing the Activity. This relationship is sometimes called assigned to for the purposes of traceability.

2) The other way is as part of a larger process (aggregated Activity). This is sometimes called *allocated to* and forms the initial stages of system or process decomposition. Allocated Performer elements (parts of Performers) are assigned Activities (or processes, tasks) in the initial stages of Performer definition.

c. A standard (Rule) constrains an Activity in general. Some of those constraints might also apply to the performance of the Activity by a Performer.

d. A Performer may have Measures associated with the performance of an Activity (e.g., target tracking accuracy.) It may also have Measures associated with the Performer overall (e.g., operational condition.)

e. Performers perform at Locations that can be specific positions or areas, regions, or installations, sites, or facilities. Location type requirements/capabilities of a Performer are captured/expressed via the Activities that are performed under certain Conditions (e.g., must be able to perform Maneuver under Desert Conditions.)

Table 2.1.1-1: DoDAF Meta-model Definitions for Performers

Technical Term	Composite Definition	Potentially Related Terms or Aliases
	Classes	
Activity	Work, not specific to a single organization, weapon system or individual that transforms inputs (Resources) into outputs (Resources) or changes their state.	Action, Process Activity, Process, Function, System Function, Operation, Task, Plan, Project
Condition	The state of an environment or situation in which a Performer performs.	
IndividualPerformer	A specific thing that can perform an action	
LocationType	The powertype of Location	
Materiel	Equipment, apparatus or supplies that are of interest, without distinction as to its application for administrative or combat purposes.	
Measure	The magnitude of some attribute of an individual.	
Organization	A specific real-world assemblage of people and other resources organized for an on-going purpose.	Department, Agency, Enterprise
OrganizationType	A type of Organization	
Performer	Any entity - human, automated, or any aggregation of human and/or automated - that performs an activity and provides a capability.	Actor, Agent, Capability Configuration (MODAF)
PersonType	A category of persons defined by the role or roles they share that are relevant to an architecture.	Role

Table 2.1.1-1: DoDAF Meta-model Definitions for Performers

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Rule	A principle or condition that governs behavior; a prescribed guide for conduct or action	
Service	A mechanism to enable access to a set of one or more capabilities , where the access is provided using a prescribed interface and is exercised consistent with constraints and policies as specified by the service description. The mechanism is a Performer. The "capabilities" accessed are Resources -- Information, Data, Materiel, Performers, and Geo-political Extents.	
Skill	The ability, coming from one's knowledge, practice, aptitude, etc., to do something well.	Training, Knowledge, Ability
System	A functionally, physically, and/or behaviorally related group of regularly interacting or interdependent elements.	
Associations		
activityPerformableUnderCondition	Represents that an activity was / is / can-be/ must-be conducted under certain conditions with a spatiotemporal overlap of the activity with the condition.	
activityPerformableUnderConditionTypeInstanceOfMeasure	activityPerformableUnderCondition is a member of Measure	
activityPerformedByPerformer	An overlap between a Performer and an Activity that is non-specific as to whether: 1. the Activity is solely performed by the Performer 2. the Activity is performed by several Performers 3. the Performer performs only this Activity 4. the Performer performs other Activities	
activityPerformedByPerformerTypeInstanceOfMeasure	activityPerformerOverlap is a member of Measure	
activityPerformedByPerformerTypeInstanceOfRule	activityPerformerOverlap is a member of Rule	
conditionTypeInstanceOf	Condition is a member of Measure	
individualPerformerPowertypeInstanceOfPerformer	IndividualPerformer is a member of Performer	
materialPartOfSystem	A whole-part association between a System (whole) and the Materiel parts of the System. (A System can have Personnel	

Table 2.1.1-1: DoDAF Meta-model Definitions for Performers

Technical Term	Composite Definition	Potentially Related Terms or Aliases
	Type and Organizational components.)	
organizationPowertypeInstanceOfOrganizationType	Organization is a member of OrganizationType	
performerPerformsAtLocationType	The relationship that describes the location of a performer or type of performer	
personTypePartOfSystem	A overlap between a Personnel Type and a System in which it performs	
ruleConstrainsActivity	An overlap between a Rule and the Activities it allows	
skillPartOfPersonType	An overlap between a Personnel Type and the Skills it entails	
skillPartOfPersonTypeTypeInstanceOfMeasure	skillPartOfPersonType is a member of Measure	

*The Black Text indicates a Concept in DM2, the Red Text indicates an Association in DM2.

Table 2.1.1-2: Aliases and Composite Terms Related to Performers

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Ability	The quality of being able to perform	
Actor	A performer that is external to and invokes the performer to be architected.	User, customer, agent, performer
Business Process	A functionally or temporally linked collection of structured activities/ tasks aimed at producing specific services and products for an end-user.	Activity, Process, Function, Job, Chore, Assignment.
Capability Configuration	A combination of organizational aspects (with their competencies) and equipment that combine to provide a capability.	aggregated Performer
Data Dependency	Resource consumed by Performer	1. Resource consumed by Performer 2. dataAssociation
Enterprise	An umbrella term for the management systems, information systems and computer systems within an organization.	System
Federation	A union comprising a number of partially self-governing states or regions united by a central ("federal") government	A type of Performer interaction (overlap of their Activities).

Table 2.1.1-2: Aliases and Composite Terms Related to Performers

Technical Term	Composite Definition	Potentially Related Terms or Aliases
FoS	A set of systems that provide similar capabilities through different approaches to achieve similar or complementary effects. For instance, the warfighter may need the capability to track moving targets. The FoS that provides this capability could include unmanned or manned aerial vehicles with appropriate sensors, a space-based sensor platform, or a special operations capability. Each can provide the ability to track moving targets but with differing characteristics of persistence, accuracy, timeliness, etc.	Systems with similar Capability overlaps.
Function	The action for which a person or thing is specially designed, fitted, used or intended to accomplish or execute.	Activity, Process, Job, Chore, Assignment.
Functional Dependency	A constraint on or dependence of, a function on one or more outside influences, conditions, functions, triggers or events.	Composite of Activity with Constraint or dependence on one or more Conditions, Activities, triggers (composite of Activity and Event), Events.
Mechanism	An instrument or a process, physical or mental, by which something is done or comes into being.	Performer
Network	An interconnected or interrelated chain, group, or system	System, group of systems, chain of systems
Operational Condition	A statement of the values or states needed for the execution of actions within the processes and transactions of an enterprise.	Condition
Performer Role	Any entity - human, automated, or any aggregation of human and/or automated - that performs a function, activity, or role, or provides a capability.	1. Composite of Performer (and its parts in the case of an aggregate), the Activities it performs, the processes (Activities) it is within (overlaps), and the Capabilities in provides. 2. Alias with function (Activity)
Performer Supporting Activity	A type of Activity - Performer overlap between a Performer and those Activities which may not necessarily be carried out by the Performer but which are necessary for the performance of the Activity	ActivityPerformerOverlap of the Activities actually performed by the Performer and then Activity overlaps between them and the supported Activity

Table 2.1.1-2: Aliases and Composite Terms Related to Performers

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Physical Asset	Covered by the Real Property and Materiel concepts	Real Property, Materiel
Platform	A set of subsystems/technologies that provide a coherent set of functionality through interfaces and specified usage patterns that any subsystem that depends on the platform can use without concern for the details of how the functionality provided by the platform is implemented.	System
Process	A logical, systematic sequence of activities, triggered by an event, producing a meaningful output.	Activity, Process, Function, Job, Chore, Assignment.
Responsibility	Answerable or accountable, as for something within one's power, control, or management	Association between a Performer and an Activity by or under an Agreement between an authority Performer and a performing Performer that performing Performer perform Activities in accordance with certain Metrics, Rules, Conditions, and Locations.
Role	A set of similar or otherwise logically related activities, implying a set of skills or capabilities, to which a performer may be assigned.	Performer, Activity, and their overlap
ServiceFunction	White box implementation of the Activities of the Service.	Activity known to be a Service Function when it is performed by a Service
SoS	A set or arrangement of interdependent systems that are related or connected to provide a given capability. The loss of any part of the system could significantly degrade the performance or capabilities of the whole. The development of an SoS solution will involve trade space between the systems as well as within an individual system performance.	Systems that have interface overlaps necessary to achieve Capabilities.
System Function	A function that is performed by a system. Although commonly used to refer to the automation of activities, data transformation or information exchanges within IT systems, it also refers to the delivery of military capabilities.	Activity, Process, Function, Job, Chore, Assignment.

Table 2.1.1-2: Aliases and Composite Terms Related to Performers

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Task	A action, activity or undertaking enabling missions, activities or functions to be performed or accomplished.	Activity, Process, Function, Job, Chore, Assignment.
Technical Dependency	A Constraint on an Activity related to Performer(s) or Resource(s) needed.	Rule to Performer Resource - Performer overlap Resource consumed by Performer
Unit	Any military element whose structure is prescribed by competent authority, such as a table of organization and equipment; specifically, part of an organization.	Organization
User	Any actor (as defined above) that invokes an automated performer.	Actor

2.1.2 Method

Methods for collecting and viewing Performer data are as follows:

2.1.2.1 Performer Modeling and Core Usage. In a typical modeling methodology, an event (contextually, a short activity) initiates an action (single-step activity) within (part of) an activity (multiple steps) to form (aggregation) a process (multiple activities) which accomplishes a defined outcome. Activities and composition activities (processes) can be serial or parallel. Activities are assigned to Performers (organizational, human, materiel, or some combination thereof). Capabilities or lower-level derived capabilities, measures, conditions, constraints and other expressions of requirements are assigned to the various levels of Performer decomposition. Allocation occurs from level-to-level as part of the structural design decomposition.

Allocation is the term used by architects and engineers to denote the organized cross-association (mapping) of elements within the various structures or hierarchies of a user view regardless of modeling convention or standard. The concept of allocation requires flexibility suitable for abstract system specification, rather than a particular constrained method of system or software design. System modelers often associate various elements in abstract, preliminary, and sometimes tentative ways. Allocations can be used early in the design as a precursor to more detailed rigorous specifications and implementations. As the requirements definition stage gives way to the design stage and actual components become visible, it becomes important to distinguish between *allocated to* and *assigned to*.

Some types of performers under configuration control called system Configuration Items (CIs). Software Configuration items are termed Computer Software Configuration Items (CSCIs) or Software Configuration items (SCIs) in MIL-HDBK-881A. Hardware Configuration items may follow the Mil-STD-161E[8] taxonomy (Central, Center, System, Subsystem, Set, Group, Unit.)

[8] MIL-STD-196E, 17 February 1998, Joint Electronics Type Designation System.

MIL-HDBK-881A[9], which guides DoD Work Breakdown Structures (WBS), defines software only by levels (e.g., 1, 2, 3, etc.)

2.1.2.2 System Functions. Activities performed by a System are defined as system or service functions (i.e., activities and/or processes performed by a system). System or service functions (activities) are allocated to hardware, software, firmware or personnel (when the person is considered integral to the system).

2.1.2.3 Personnel Activities. Personnel processes are typically termed Tactics, Techniques and Procedures (TTP) in DoD. Procedures are allocated sets of activities and/or processes, where Tactics and Techniques, typically, are made up of the procedures as influenced by rules, doctrine, paradigms, etc. acquired through skill development during the education and training process.

2.1.2.4 Performer Data Capture Method. A method to capture Performer data is described in Table 2.1.2.4-1.

Table 2.1.2.4-1: Performer Data Capture Method Description

Methodology Description	Capture Data for Architectural Description of Performer
Definition:	Define a process by which architectural information relative to the Performer entity within the DoDAF Meta-model can be captured and structured to enable it to support the major decision processes of the Department (e.g., PPBE, PfM, and JCIDS). A Performer can be one of several actors/mechanisms that execute a function, activity or process. Within the context of DoDAF V2.0, a Performer can be a person, organization, service or system.
Input:	• Concepts of Operations documentation • Organization Charts • Operational Roles • Human Resources (HR)/Personnel Data/Documentation • Systems Documentation • Services Documentation • Requirements Documentation
Method:	DoDAFV 2.0 is intended to be methodology agnostic. Therefore, structured analysis and object-oriented analysis techniques can be used to capture the information that constitutes a Performer. The following process can be used to capture the architectural information relative to Performer. 1. For the purpose identified as driving the architecture effort, identify the business functions required to support the purpose. 2. Identify the capabilities required to support the functions. 3. Identify the organizations and organizational roles that are responsible for executing the functions and/or delivering the capabilities. a. For any organizational roles identified as required for the function or capability, identify the requisite skills for the role. b. Associate the roles to the skills.

[9] MIL-HDBK-881A, 30 July 2005, Department of Defense Handbook, Work Breakdown Structures for Defense Materiel Items

Table 2.1.2.4-1: Performer Data Capture Method Description

Methodology Description	Capture Data for Architectural Description of Performer
	c. In some cases there may be levels of skill required to fulfill a role or roles. Associate the requisite skill levels to the appropriate roles. 4. Identify any services either in place or planned to support the functions and capabilities. 5. Identify any systems either in place or planned to support the functions and capabilities. 6. If identifying and defining processes to support the functions, identify the roles responsible for executing the steps of the process. a. If defining a process at a level of granularity to support automation, identify roles, and services and systems responsible for executing the process. 7. The roles that have been previously identified can now be used as mechanisms on an activity model, swim lanes in a process model, or as actors in a use-case model.
Primary Output:	Types of Persons/Roles, Skills or Skill Sets, Services, Systems, Organizations
Secondary Output:	Skill Levels (i.e., measures), Personnel
Disciplines:	Structured Analysis, Object-Oriented Analysis, Business Process Analysis

2.1.3 Use

Data for Performer are used in the following ways:

MIL-HDBK-881A, 30 July 2005 and DoDD 5000.1, in providing fundamental guidance for specifications, WBS, Statement of Works (SOWs) of the DAS all require the identification of the Performers and their component parts and types as fundamental elements.

In typical uses, the Activities are represented by **verbs** and Performers are represented by **nouns**. This distinguishes the *how* from the *who*. In a typical specification process allocation to performers can take place at varying levels of detail depending on the design maturity or the intended degree of design constraint.

Performers are represented in many places and stages in the detailed architecture. It should be noted that a pure Requirements Architectural Description may not show allocations or performer. This may be left to later stages of the design process. Further, not all architecture modeling standards explicitly provide for allocation. For example, the Systems Modeling Language (SysML) extensions to the UML modeling standard have added this feature.

2.2 Resource Flows

This section is oriented toward the use and methods associated with Resource Flows that are typically used to model the behavioral aspects of activities (processes, tasks, etc.) and performers. Resource Flows should be used to model the flow of material, information or personnel. Resource Flows are extensively used as a key technique in systems engineering, process improvement, work flow, mission planning and many other disciplines. Resource Flow models and associated analysis techniques reveal behavior such as:

- The connectivity between resources.
- The content of the information flowing between resources (e.g., interface definition).
- The order or sequential behavior (parallel or serial) of the resources in relation to one another (e.g., project task execution and critical path).
- The behavior of Resource Flow between or within organizations (e.g., work flow, information flow, etc.).
- The changes in state during the spatial and/or temporal existence of the resource.
- The rules that modify the behavior of the Resource Flow (e.g., business rules, controls, decisions, etc.).
- The measures that define the quality, constraints, timing, etc. of the Resource Flow (e.g., Quality of Service (QoS), measures of performance, measures of effectiveness, etc.).
- The flow of control orchestrating the behavior of the Resource Flow.

These techniques apply to the flow of material, personnel, and information; this section will focus on the Information Flow between activities and performers. Resource flow representing flow of material and/or personnel should also be represented using the same techniques. Activity Resource Flows should be used for process improvement analysis including automation tradeoffs. Performer Resource Flows should be used in disciplines, such as system engineering, interface definition, and organizational work flow planning. The Resource Flows should be directly traceable to the capability and/or upper-level process defining the root need or requirement. Operations utilizing information flows should be technology independent. However, operations and their relationships may be influenced by new technologies where process improvements instituted before policy can reflect the new procedures. There may be some cases in which it is necessary to document the way activities are performed to examine ways in which new systems could facilitate streamlining the activities. In such cases, information Resource Flows may have technology constraints and requirements.

Figure 2.2-1 represents a dated example of an Enterprise-level View of Resource Flow depicting high-level connectivity between resources, high-level mission and goals, and net-centric architectural concepts. This type of Resource Flow is typically used as a high-level operational concept graphic with lower-level models detailing the Resource Flows.

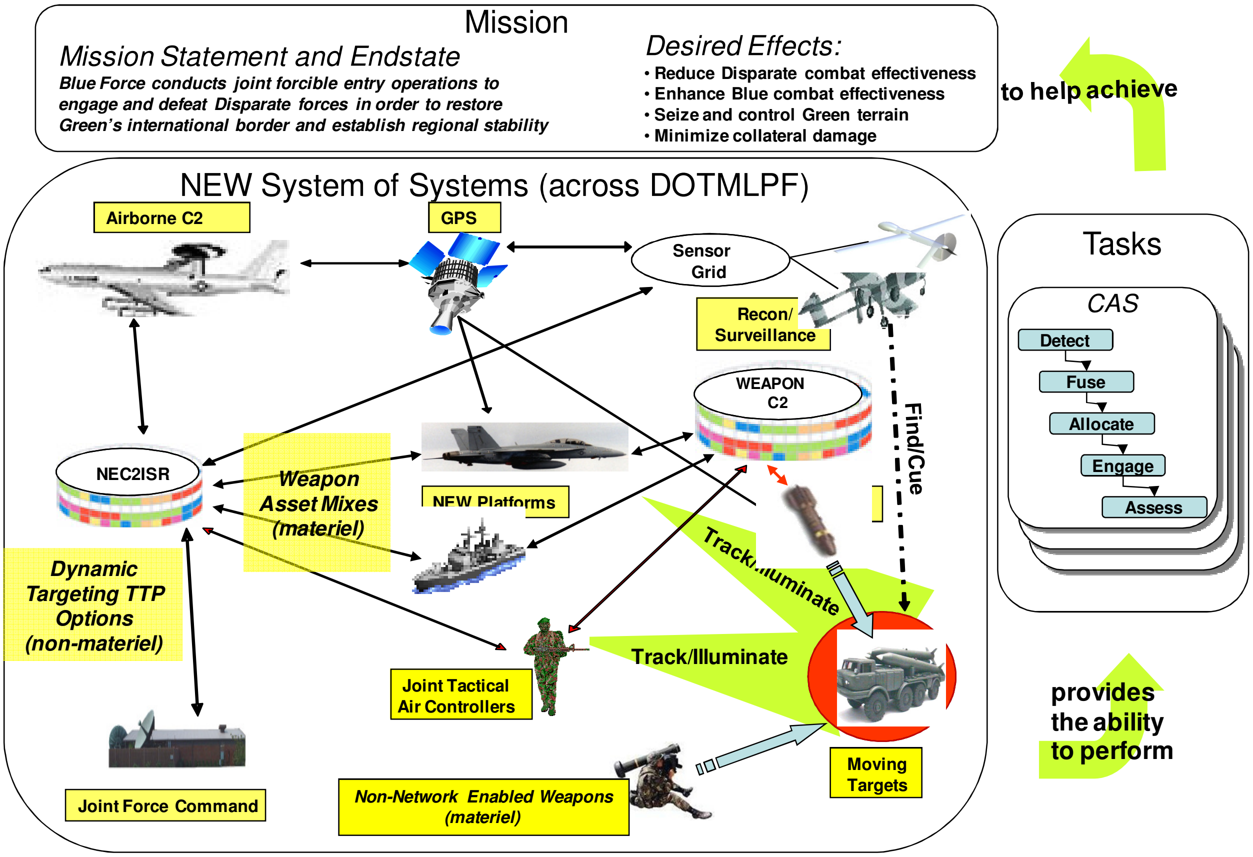


Figure 2.2-1: A Dated Example Diagram Illustrating Resource Flow

2.2.1 Data

The DoDAF Meta-model for the data comprising Resource Flow is shown in [Figure 2.2.1-1](). The figure may be hard to read in a hardcopy printout but a version at full-resolution which can be zoomed in, is published in the online DoDAF V2.0 Meta-model Data Dictionary. Definitions for the model terms are in [Table 2.2.1-1](). Alias and composite terms related to Resource Flows are shown in [Table 2.2.1-2](). Authoritative Source definitions, aliases, and rationale are provided in the DoDAF V2.0 Meta-model Data Dictionary. Note that foundational classes are generally not shown on data group diagrams; this foundational material is in Section 2. This includes super-subtype, whole-part, temporal-whole-part, overlap, type-instance (member-of), and before-after patterns. Also not shown are the data structures for classification marking of architecture information at the whole and element (portion) levels using the IC-ISM. The schema for the IC-ISM is in Volume 3. Lastly, the size of the icons is not indicative of their importance; sizes are adjusted to reduce line crossings and bends to make the diagrams easier to understand.

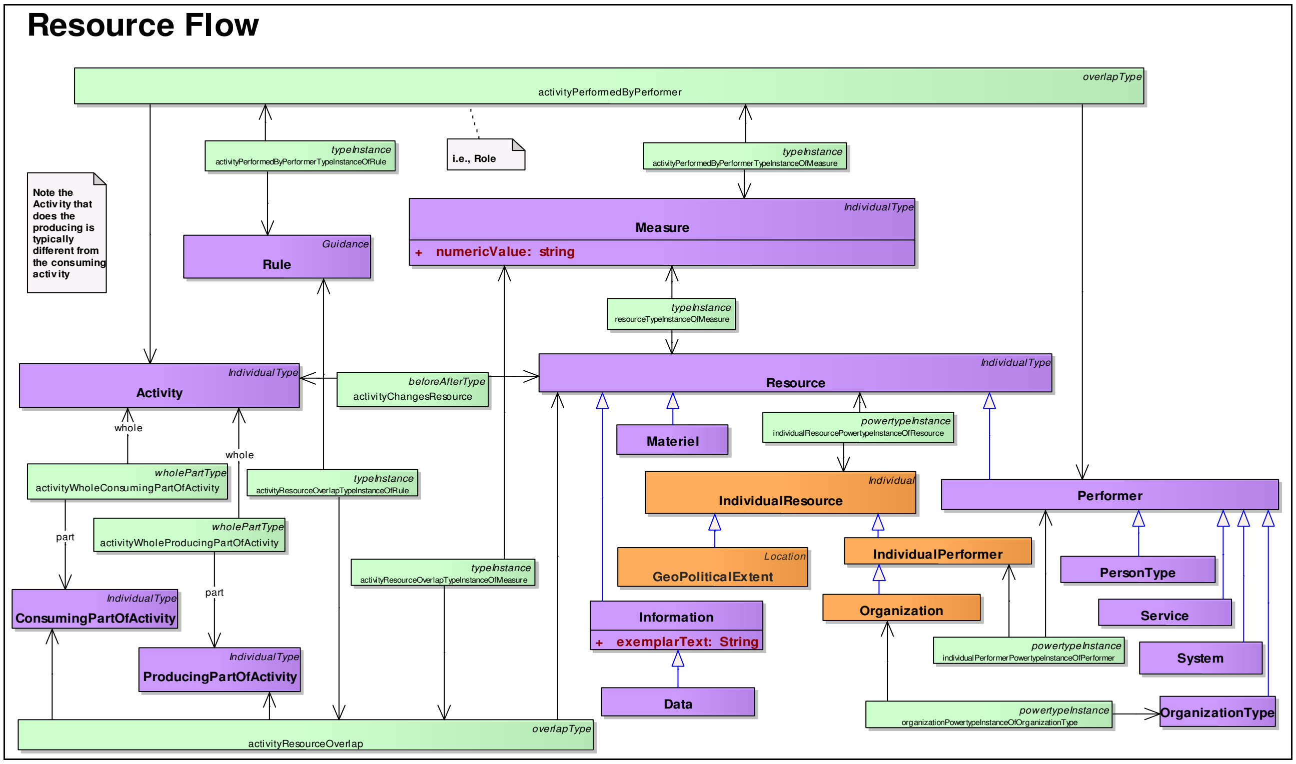


Figure 2.2.1-1: DoDAF Meta-model for Resource Flow

The Resource Flow Meta-model describes the resources that can flow between activities, tasks performed by performers. Activity-based Resource Flows are typically modeling techniques that define and describe operations. Performer based Resource Flows should be used to define and describe solutions. Resources in Resource Flows can be Personnel, Materiel, Data or Information. Rules and Measures are applied to specific Activities and their Performers. Activities, Systems and Personnel can be assigned to Locations and further can be assigned Conditions and Constraints. Resource Flows are key modeling techniques used in the definition of Interfaces and assurance of Interoperability between Activities and their performing Performers (e.g., Systems and Personnel.)

a. Whereas prior versions of DoDAF modeled only information and data exchanges and flows, this version also allows modeling of other flows, such as:

1) Materiel flows such as ammunition, fuel, etc. important for modeling the fire rate, logistics, etc., aspects of a Capability solution so it can be compared with other alternative solutions.

2) Personnel Types such as Military Occupational Specialty (MOS) that allow representation of the Training and Education pipeline aspects of Doctrine, Organization, Training, Material, Leadership and Education, Personnel, and Facilities (DOTMLPF).

3) Performers such as Services, Systems, or Organizations that might be the output or result of a Project’s design and production process (activities). This allows modeling of, for instance, an acquisition project.

b. Another difference from prior versions of DoDAF is that all exchanges and flows are by virtue of a producing or consuming Activity. That is, a Performer can only provide or consume by conducting an activity of production or consumption. For instance, publication and subscription are modeled as an interaction between the publishing Activity, the subscribing Activity, and the information or data Resource. Note that publication is typically not at the same time as subscription but the subscriber does have to go to the publication place to retrieve the Resource. For example, data might be published at 2:00 GMT on a server located at some URL and the subscriber may not overlap until 10:00 GMT. Also note in the diagram the overlap is a triple – the producing Activity, the Consuming Activity, and the Resource.

c. The exchange or flow triple may have standards (Rules) associated with it such as Information Assurance (IA)/Security rules or, for data publication or subscription, data COI and web services standards.

d. The exchange or flow triple may have Measures associated with it such as timeliness, throughput, reliability, or QoS.

Table 2.2.1-1: DoDAF Meta-model Definitions for Resource Flow

Technical Term	Composite Definition	Potentially Related Terms or Aliases
	Classes	
Activity	Work, not specific to a single organization, weapon system or individual that transforms inputs (Resources) into outputs (Resources) or changes their state.	Action, Process Operational Activity, Processes, Function, System Function, Operation, Task, Plan, Project
Data	Representation of information in a formalized manner suitable for communication, interpretation, or processing by humans or by automatic means. Examples could be whole models, packages, entities, attributes, classes, domain values, enumeration values, records, tables, rows, columns, and fields.	
IndividualPerformer	A specific thing that can perform an action	
IndividualResource	Any specific physical or virtual entity of limited availability	
Materiel	Equipment, apparatus or supplies that are of interest, without distinction as to its application for administrative or combat purposes.	
Measure	The magnitude of some attribute of an individual.	
Organization	A specific real-world assemblage of people and other resources organized for an on-going purpose.	Department, Agency, Enterprise
OrganizationType	A type of Organization	

Table 2.2.1-1: DoDAF Meta-model Definitions for Resource Flow

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Performer	Any entity - human, automated, or any aggregation of human and/or automated - that performs an activity and provides a capability.	Actor, Agent, Capability Configuration (MODAF)
PersonType	A category of persons defined by the role or roles they share that are relevant to an architecture.	Role
Resource	Data, Information, Performers, Materiel, or Personnel Types that are produced or consumed.	
Rule	A principle or condition that governs behavior; a prescribed guide for conduct or action	
Service	A mechanism to enable access to a set of one or more capabilities , where the access is provided using a prescribed interface and is exercised consistent with constraints and policies as specified by the service description. The mechanism is a Performer. The "capabilities" accessed are Resources -- Information, Data, Materiel, Performers, and Geo-political Extents.	
System	A functionally, physically, and/or behaviorally related group of regularly interacting or interdependent elements.	
Associations		
activityChangesResource	Represents that an activity was / is / will-be the cause of change in the effected object with a before-after relationship.	
activityPerformedByPerformer	An overlap between a Performer and an Activity that is non-specific as to whether: 1. the Activity is solely performed by the Performer 2. the Activity is performed by several Performers 3. the Performer performs only this Activity 4. the Performer performs other Activities	
activityPerformedByPerformerTypeInstanceOfMeasure	activityPerformerOverlap is a member of Measure	

Table 2.2.1-1: DoDAF Meta-model Definitions for Resource Flow

Technical Term	Composite Definition	Potentially Related Terms or Aliases
activityPerformedByPerformerTypeInstanceOfRule	activityPerformerOverlap is a member of Rule	
activityResourceOverlap	An overlap of an Activity with a Resource, in particular a consuming or producing Activity that expresses an input, output, consumption, or production Activity of the Resource.	output, produce
activityResourceOverlapTypeInstanceOfMeasure	activityResourceOverlap is a member of Measure	
activityResourceOverlapTypeInstanceOfRule	activityResourceOverlap is a member of Rule	
activityWholeConsumingPartOfActivity	A whole - part association between an Activity and the part of it that consumes a Resource.	input, consume
activityWholeProducingPartOfActivity	A whole - part association between an Activity and the part of it that produces a Resource.	proceeds, succeeds
ConsumingPartOfActivity	A part of an Activity that consumes a Resource	
GeoPoliticalExtent	A geospatial extent whose boundaries are by declaration or agreement by political parties.	
individualPerformerPowertypeInstanceOfPerformer	IndividualPerformer is a member of Performer	
individualResourcePowertypeInstanceOfResource	IndividualResource is a member of Resource	
organizationPowertypeInstanceOfOrganizationType	Organization is a member of OrganizationType	
ProducingPartOfActivity	A part of an Activity that produces a Resource	
resourceTypeInstanceOfMeasure	ResourceType is a member of Measure	

*The Black Text indicates a Concept in DM2, the Red Text indicates an Association in DM2.

Table 2.2.1-2: Aliases and Composite Terms Related to Resource Flows

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Behavior	The manner in which an individual, group or machine functions, operates or reacts/responds to stimuli.	Composite of Performer and its Activities and the Events that Trigger them and the Performer State changes and/or Activities and outputs (Exchange Objects) or trigger Events resulting from those Activities.
Data Dependency	Resource consumed by Performer	1. Resource consumed by Performer 2. dataAssociation
Event	Something that happens at an instant in the world, i.e., a zero-duration process (Activity).	Milestone, Trigger, Activity
Needline	An information technology requirement that is the logical expression of the need to transfer information among performers	composite term accomplished by ActivityOverlap and two PerformerActivityOverlaps
Trigger	Something that happens at an instant in the world, i.e., a zero-duration process (Activity).	Event or composite of Event and the Activity it triggers

2.2.2 Method

Methods for collecting and modeling Resource Flow data are as follows:

2.2.2.1 Resource Flow Modeling and Core Usage. The Resource Flow models represent the activities and their performers that either publish or subscribe to the resource containing the information. Activities are assigned to performers in defining and describing how the transition occurs when moving from operational or capability position to those describing solutions. These assignments are a result of various tradeoffs and should be maintained for traceability. Mechanization or automation trades will reveal the performer subtypes (organizations, systems, etc.) and the activities that are assigned define the functionality of the performer subtypes. Detailed design will further detail the whole-part taxonomies associated with the subtype portions of the automated aspects of the performer. It may be desirable to standardize these taxonomies for particular communities of interest (e.g., common components, common system functions [activities], common service functions [activities [, etc.). Note: The Joint Common System Function List (JCSFL) is representative of initiatives in this area. Non-automated performer subtypes (e.g., organization, personnel or procedures) maintain traceability to their root activity and form the basis for the definition of lower-level TTP. Individual communities of interest typically standardize these procedures and processes as Doctrine or policy and as such become the focus of process improvement.

It should be noted that information inputs and outputs between resources for some levels of decomposition may be at a higher-level of abstraction than the information characteristics represented in the matrix. This is commonly done to simplify graphical representations of information flow or in the initial definition stages where the characteristics are still unknown. In this case, multiple information exchanges will map to a single resource input or output. Similarly, the information inputs and outputs between resources at a low-level of decomposition may be at a higher-level of detail than the information exchanges in the matrix, and multiple information inputs and outputs may map to a single information exchange. In these cases, to provide the necessary clarity and precision, an ontological or taxonomic structure of information aggregation should be developed for use in each level of decomposition of the Resource Flow models (e.g., The Navy Common Information Exchange List [CIEL] represents initiatives showing taxonomic structure or levels of aggregation).

The upper-level aggregations have been termed *need lines* in previous versions DoDAF. Other terminology expressing levels of aggregation are used depending on the community of interest (e.g., The SysML modeling standard uses *lifeline*).

The Resource Flow model provides a key tool for engineering operational and solutions-oriented DoDAF-described Models. **Table 2.2.2.1-1** show examples of analysis considerations that should be included in trade methods employed in the analysis Resource Flows.

Table 2.2.2.1-1: Resource Flow Model Analysis Considerations

Operations Models	Solutions Models
• What are the activities of the Enterprise? • What are the primary activities of concern? • What capability limitations are associated with the processes? • What are the issues associated with these processes?	• What activities or portions of activities are currently automated and by what means? (Current baseline). • View the current activities and automation (automated performers) at the level of detail appropriate to address areas of concern.
• What process improvements are needed? • What are the specific objectives associated with the improvements?	• Define activity and system assumptions and constraints.
• Is the activity as efficient as required?	• Apply process streamlining analysis techniques (e.g., Lean Six Sigma or similar techniques).
• What are the missing or unnecessary steps? • Where are the process bottlenecks?	• Define new process change alternatives. • Define alternative for eliminating bottlenecks.
• Will the activities benefit substantially from new or modified automation/mechanization? • Define the <u>Automation opportunities</u> and expected benefits.	• Identify new <u>automation possibilities</u> afforded from new technology and associated material performers. • Evaluate cost/benefit.

Table 2.2.2.1-1: Resource Flow Model Analysis Considerations

Operations Models	Solutions Models
• Are improvements needed in TTP? • Are TTP improvements adequate versus developing new automation?	• Define candidate TTP changes. • Evaluate personnel and training impact.
• Prioritize Automation opportunities? • Prioritize TTP changes?	• Identify requirements for new performers (technology components, building blocks, etc.) and performance characteristics. • Identify new system or service, functions (activities), components and modifications required.
• Do we need to integrate among other related Service and Mission areas, and system efforts?	• Identify new system integration requirements. • Identify new Resource Flow requirements.
• Are the activities and procedures interoperable?	• Identify new and emerging systems interoperability requirements. • Identification of the need for Application of new standards.

Specific automation or mechanization trades (e.g., analysis of automation opportunities and possibilities) could initially be described from the operational or capability position and then iterated as part of a proposed solution as part of the tradeoff space.

Various methods can be employed in modeling and analyzing Resource Flow. Both structured and object-oriented techniques should be used where appropriate. Typically structured methods are useful in representing requirements traceability, testing, and decomposition of detailed procedures dealing with Resource Flow. Object-oriented techniques can be used in the gathering of user needs and the design of software. Typically structured analysis emphasizes process and functions, while object-oriented analysis emphasizes system behavior using objects. Resource flow can use both techniques to adequately represent the behavior in both Operational and Solutions-related Viewpoints and DoDAF-described Models. Careful consideration should be given to where and when to apply the appropriate methods. Typical modeling methodologies are illustrated in **Figures 2.2.2.1-1** and **2.2.2.1-2**. In the structured design approach, performers, activities, resources, rules, conditions, and measures have whole-part (spatial, temporal) and super-subtype relationships that allow successive refinement of the model.

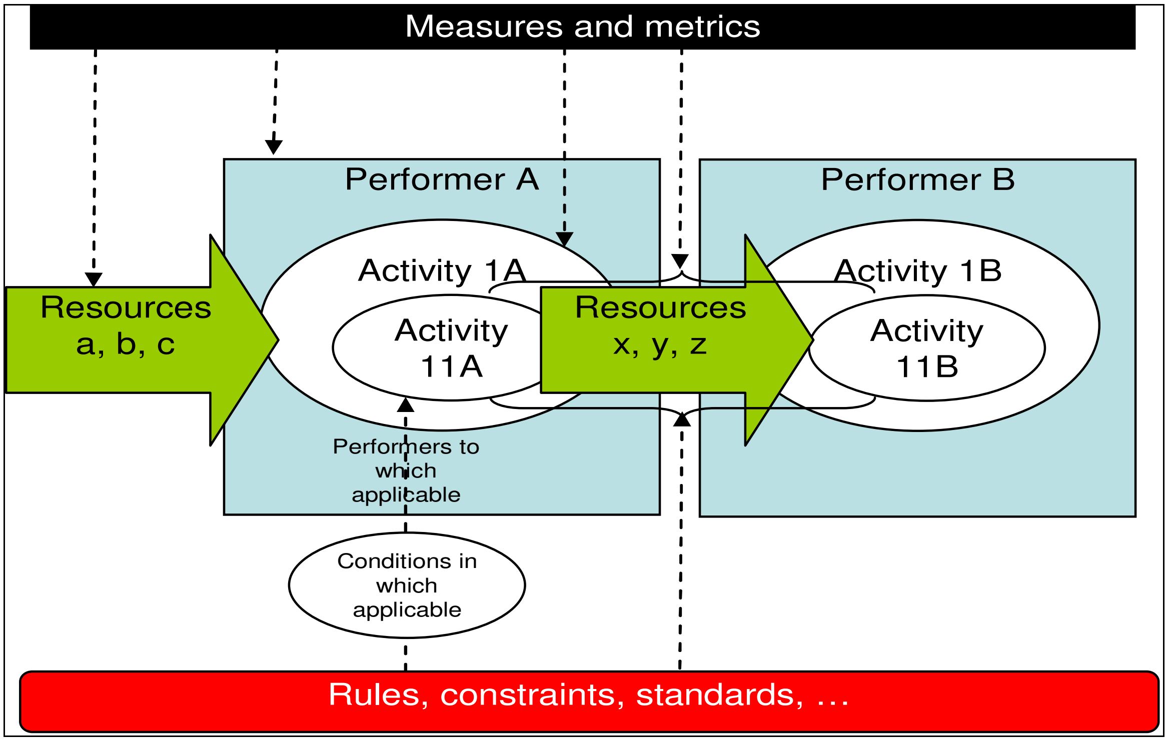


Figure 2.2.2.1-1: Non-Prescriptive, Illustrative Structured Design Technique Example

The Resource Flow also provides a key tool for engineering the interfaces needed to define and describe Operational and Solution-related Viewpoints and DoDAF-described Models. Interfaces can be considered at varying levels of the enterprise and their granularity of definition depends on the purpose. Interface identification, explicit definition and control are essential in every enterprise. These interfaces, for the purpose of this document, can be considered to be any interconnection or interaction between producing and consuming activities and their performers. The focus in Solution-related Viewpoints and DoDAF-described Models should be on interfaces within and between equipment, subsystems, systems, an SoS, or other technology driven aspects of an enterprise. Attention to this area is critical to cost effective acquisition and development under the DAS. Human and organizational interactions typically are the focus of the Operational Viewpoint and DoDAF-described Models except when human beings are considered an integral part of the system's operation and functionality (e.g., system operator versus system user).

Interfaces are generally documented in interface documentation representing the agreements of the responsible parties in charge of each end of the interface (both information supplier and information consumer). This, in no way implies a point-to-point interface. Interfaces implemented with an enterprise service bus, for example, are defined with appropriate publish/subscribe documentation formalized, if necessary, with contractual agreements between information supplier and consumer.

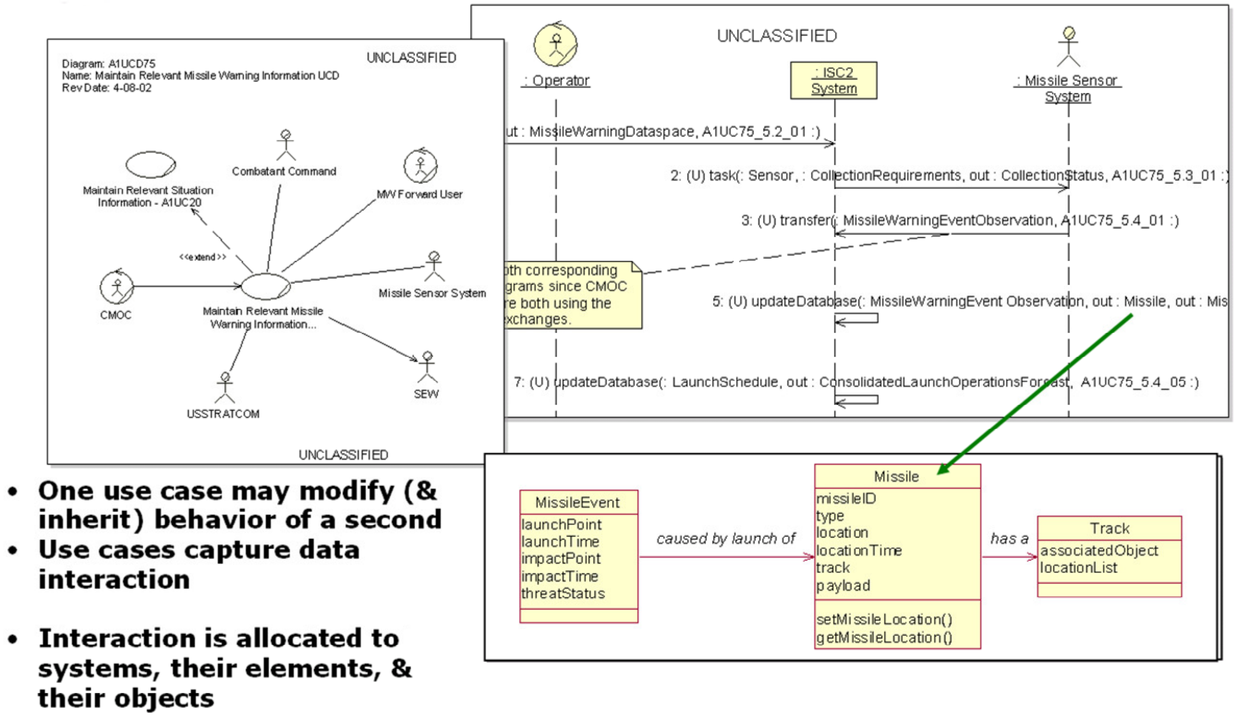


Figure 2.2.2.1-2: Non-prescriptive, Illustrative Object Oriented Design Technique Example

2.2.2.2 Resource Flow Data Capture Method. A method to capture Resource Flow data is described in Table 2.2.2.2-1.

Table 2.2.2.2-1: Resource Flow Data Capture Method Description

Methodology Description	Capture Data for Architectural Description of Resource Flow
Definition:	Define a process by which architectural information relative to the Resource Flow entity within the DoDAF Meta-model can be captured and structured to enable it to support the major decision processes of the Department (e.g., PPBE, PfM, and JCIDS). Per the definition of Resource Flow, it becomes apparent that interfaces are integral to accurately identifying and defining the resources for a particular architecture effort. Within the context of DoDAF V2.0, a resource can be data, information, performer, materiel, or personnel types.
Input:	• Concepts of Operations documentation • Operational Roles • HR/Personnel Data/Documentation • Systems Documentation • Requirements Documents • Services Documentation

Table 2.2.2.2-1: Resource Flow Data Capture Method Description

<table>
<tr><th>Methodology Description</th><th>Capture Data for Architectural Description of Resource Flow</th></tr>
<tr><td>Method:</td><td>
DoDAF V2.0 is intended to be methodology agnostic. Therefore, structured analysis and object-oriented analysis techniques can be used to capture the information that constitutes a Performer. The Performer entity is included here because resources can be transmitted between Performers by virtue of their producing and consuming activities. The following process can be used to capture the architectural information relative to Resource Flow.
The term flow implies that something (e.g., materiel, information) is moving from point A to point B. This means that interfaces must be a focus of the analysis for Resource Flow. DoDAF has identified several entities that would have interfaces that enable exchange of resources. These entities are:

Activities
Performers e.g.:

Services
Systems
Organizations (Operations Department)
Personnel Types (e.g., Commanding Officer)

For the purpose identified as the intended purpose for the architecture, determine the level of granularity needed for things being exchanged or interchanged. (For example, if the purpose of the architecture were to serve as a source of design requirements to constrain system development, the resources need to be identified and defined at the data element-level. If the purpose of the architecture were to support Investment Managers in categorizing systems, the resources may need to be defined only at a categorization-level, such as Sales Reimbursement Information.
For activities, identify and define the objects that are being either consumed or produced by the activity or process.
If being consumed, designate the object as an input to the activity or process. If being produced, designate the object as an output of the activity or process.
To be able to complete the description of Resource Flow, it is imperative that the origination and destination of the resources being exchanged are identified and defined. This creates a logical flow between activities or process steps that can be modeled and analyzed in support of the everyday operations.

For services and systems, the interfaces are integral to definition of Resource Flow.

Identify the services and/or systems that must talk to each other. This implies that there must be an interface between those services or systems.
Identify the data or information that must be exchanged via the interfaces.
As mentioned above, designate whether the exchanged information is being either consumed or produced. This is especially important when accommodating services within the architecture.
Show traceability to the portion of the operational process being automated by the performing system or service.

</td></tr>
</table>

Table 2.2.2.2-1: Resource Flow Data Capture Method Description

Methodology Description	Capture Data for Architectural Description of Resource Flow
Primary Output:	Types of Persons/Roles, Skills or Skill Sets, Services, Systems, Organizations, Data and Information
Secondary Output:	Skill Levels (i.e., measures), Personnel
Disciplines:	Structured Analysis, Object-Oriented Analysis, Business Process Analysis

2.2.3 Use

Resource Flow modeling is a fundamental engineering based technique used in Information Technology (IT) Architecture, System Engineering, Process Re-engineering, Resource Planning and many other disciplines. Resource Flow modeling provides an explicit means to describe the behavior of activities, systems, organizations and their composite effects on the overall enterprise. Resource Flow modeling can be performed at varying levels of detail and fidelity depending on the areas of concern being analyzed and the solutions being sought. Key areas where Resource Flow modeling is used include:

- Process Improvement Analysis including reengineering, and gap/overlap identification.
- System Engineering including architecture, design, testing and production.
- Interface Identification and Definition including interoperability analysis and standardization.
- Project Planning including scheduling and task sequencing.
- Mission Planning including simulation and training.
- Logistics planning.

Examples of detailed use of Resource Flow models in the developing the Operational Viewpoint and DoDAF-described Models are:

- Clearly identify the Activities required to provide a Capability.
- Clearly associate activities with responsible organizational or personnel performers.
- Uncover unnecessary or inefficient operational activities and information flows.
- Evaluate alternative architectures with different connectivity and Resource Flow to maximize capability and minimize automation complexity.
- Provide a necessary foundation for depicting information needs and task sequencing to assist in producing procedures, operational plans and facilitate associated personnel training.
- Identify critical mission threads and operational Resource Flow exchanges by annotating which activities are critical (i.e., identify the activities in the DoDAF-described Model that are critical e.g., Critical Path).
- Identify and prioritize activities that are candidates for automation.
- Identify common activities that can be standardized across capability or mission areas, communities of interest, etc.
- Identify or flag issues, automation opportunities, or changes to activities and information flow that need to be scrutinized further.

- Identify critical connectivity needs and interfaces (or Key Interface Profiles (KIPs) between activities and their performers (organizations and personnel types).

Examples of more detailed use of Resource Flow models in solution-related Viewpoints and DoDAF-described Models are:

- Clearly identify the relationship and information flow between systems and system/services in an SoS or between services in a Service Oriented Architecture (SOA).
- Identify the Interfaces and/or Publish/Subscribe needs between systems and/or services.
- Define Interface details.
- Support configuration management of interfaces.
- Support Analysis of Alternatives (AoA) and other Systems Engineering Analysis.
- Verify the decomposition of the Activities (System Functions or Service Functions).
- Support the various levels of system definition and design.
- Define explicit traceability to needs, capabilities and goals in the Operational Viewpoint and DoDAF-described Models.
- Support functional allocation in a System of Systems or within Systems.
- Evaluate alternative system architectures.
- Support the development of test sequences and procedures.
- Support system design and training documentation.

Among the many uses of Resource Flow modeling is DoD's Enterprise Architecture focus on Interoperability and net-centric goals to improve the interfaces between activities and their performers. In that light some amplification with regard to Interface definition and analysis relationship to the DoD's primary processes of JCIDS, PfM, and the DAS is in order.

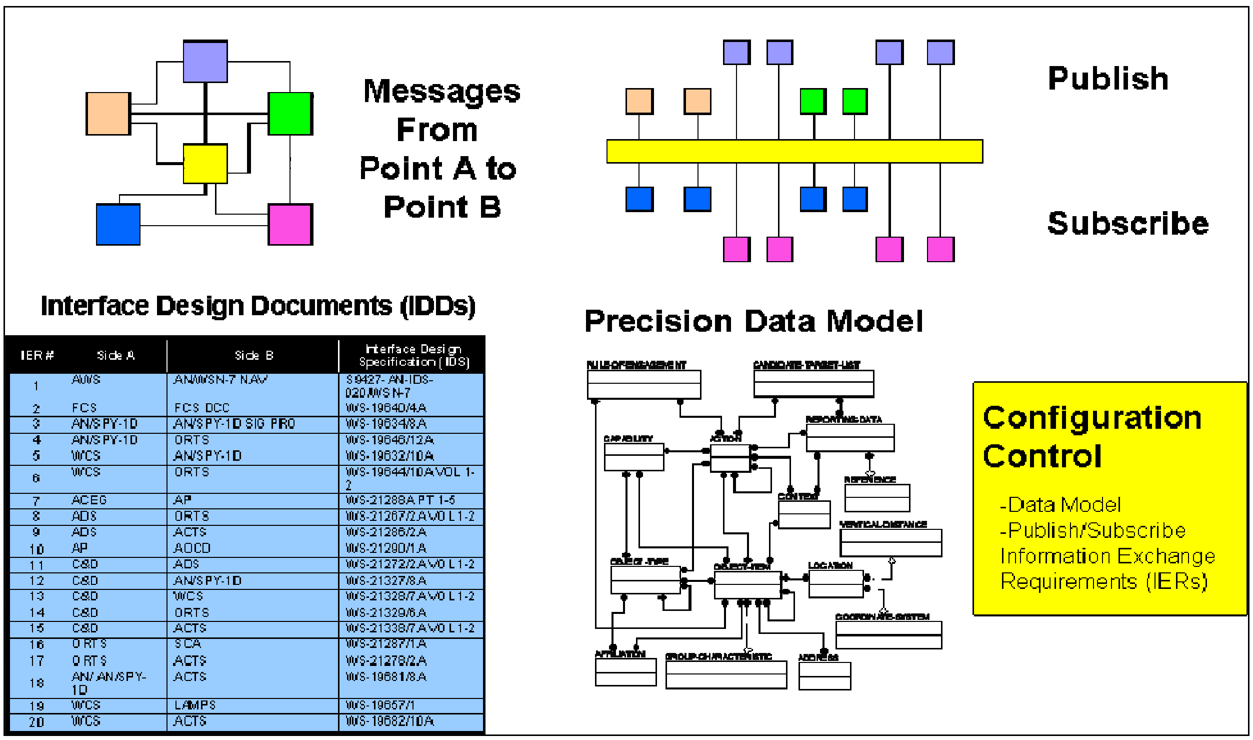
System interfaces reflect and are traceable to information flow needs or requirements identified in the Operational Viewpoint and DoDAF-described Models. Resource Flow descriptions, produced at varying levels of detail, substantially contribute to the quality of this process and aid in the understanding and documentation.

The Details of Resource Flow (materiel, personnel, or data) are generally documented in Interface Control Documents (ICDs), Interface Requirements Specifications (IRSs) and Interface Description Documents (IDDs). This data is typically provided to DoD Investment Review Board (IRB) registry systems for the purpose of milestone reviews and support of acquisition decisions points.

Critical Interfaces are generally documented in formal interface documentation signed by the responsible authorities (both information supplier and information consumer) in charge of each end of the interface. This type of interface may be annotated as a Key Interface (KI). A KI is defined as an interface where one or more of the following criteria are met:

- The interface spans organizational boundaries (may be across instances of the same system, but utilized by different organizations).
- The interface is mission critical.
- The interface is difficult or complex to manage.
- There are capabilities, interoperability, or efficiency issues associated with the interface.

Critical Interfaces should be traceable to the interfaces identified in the JCIDS process. Further, critical interfaces are generally documented in formal interface documentation signed by the responsible authorities (both information supplier and information consumer) in charge of each end of the interface. For legacy point-to-point interfaces this may be in the form of ICDs, Interface Requirement Documents (IRSs), Interface Design Documents (IDDs), etc. In multiple access or common connectivity (radio communications or bus type connectivity) implementations may be in the form of formal agreements (defined herein as a consent among parties regarding the terms and conditions of activities that said parties participate in) detailing the specific set of implementations (e.g., Tactical Digital Information Links [TADILs]) data elements implementation tables or in the case of a SOA, a publish/subscribe implementation document. These agreements are, in general, managed and controlled by the SoS or System Project manager. In new systems, and where possible the interface should be managed and configuration controlled using a common precision data model. Figure 2.2.3-1 illustrates the evolution from configuration control of legacy point-to-point interfaces to a net-centric, distributed processing means of connectivity using carefully managed publish and subscribe agreements and documentation based on formally documented logical and physical data models.



IER #	Side A	Side B	Interface Design Specification (IDS)
1	AWS	AN/WSN-7 NAV	S9427- AN-IDS-020/WSN-7
2	FCS	FCS DCC	WS-19640/4A
3	AN/SPY-1D	AN/SPY-1D SIG PRO	WS-19634/8A
4	AN/SPY-1D	ORTS	WS-19646/12A
5	WCS	AN/SPY-1D	WS-19632/10A
6	WCS	ORTS	WS-19644/10A VOL 1-2
7	ACEG	AP	WS-21288A PT 1-5
8	ADS	ORTS	WS-21267/2A VOL 1-2
9	ADS	ACTS	WS-21286/2A
10	AP	AOCD	WS-21290/1A
11	C&D	ADS	WS-21272/2A VOL 1-2
12	C&D	AN/SPY-1D	WS-21327/8A
13	C&D	WCS	WS-21328/7A VOL 1-2
14	C&D	ORTS	WS-21329/6A
15	C&D	ACTS	WS-21338/7A VOL 1-2
16	ORTS	SCA	WS-21287/1A
17	ORTS	ACTS	WS-21278/2A
18	AN/ AN/SPY-1D	ACTS	WS-19681/8A
19	WCS	LAMPS	WS-19657/1
20	WCS	ACTS	WS-19682/10A

Figure 2.2.3-1: Migrating from Legacy to Data Focused Configuration Management

2.3 Information and Data

Information is the state of a something-of-interest that is materialized, in any medium or form, and communicated or received. In DoDAF V1.0, this took the form of what was called a logical data model which even in DoDAF V1.0 permitted a less structured and formalized description than the computer science definition of a logical data model. In DoDAF V2.0, the emphasis is on the identification and description of the information in a semantic form (what it means) and why it is of interest (who uses it). Although this may entail some formality such as describing relationships between concepts, its purpose is to convey the interests in the operator, executive, or business person's frame of reference.

Data is the representation of information in a formalized manner suitable for communication, interpretation, or processing by humans or by automatic means, and is concerned with the encoding of information for repeatability, meaning, and proceduralized use. While information descriptions are useful in understanding requirements, e.g., inter-federate information sharing requirements or intra-federate representation strategies, data descriptions are important in responsive implementation of those requirements and assurances of interoperable data sharing within and between federates.

2.3.1 Data

The DoDAF Meta-model for the data, comprising Information and Data is shown in **Figure 2.3.1-1.** The figure may be hard to read in a hardcopy printout but a version at full-resolution which can be zoomed in is published in the online DoDAF V2.0 Meta-model Data Dictionary. Definitions for the model terms are in **Table 2.3.1-1.** Aliases and composite terms related to Information and Data are shown in Table 2.3.1-2.

Authoritative Source definitions, aliases, and rationale are provided in the DoDAF V2.0 Meta-model Data Dictionary. Note that foundational classes are generally not shown on data group diagrams; this foundational material is in Section 2. This includes super-subtype, whole-part, temporal-whole-part, overlap, type-instance (member-of), and before-after patterns. Also not shown are the data structures for classification marking of architecture information at the whole and element (portion) levels using the IC-ISM. The schema for the IC-ISM is in Volume 3. The size of the icons is not indicative of their importance; sizes are adjusted to reduce line crossings and bends to make the diagrams easier to understand.

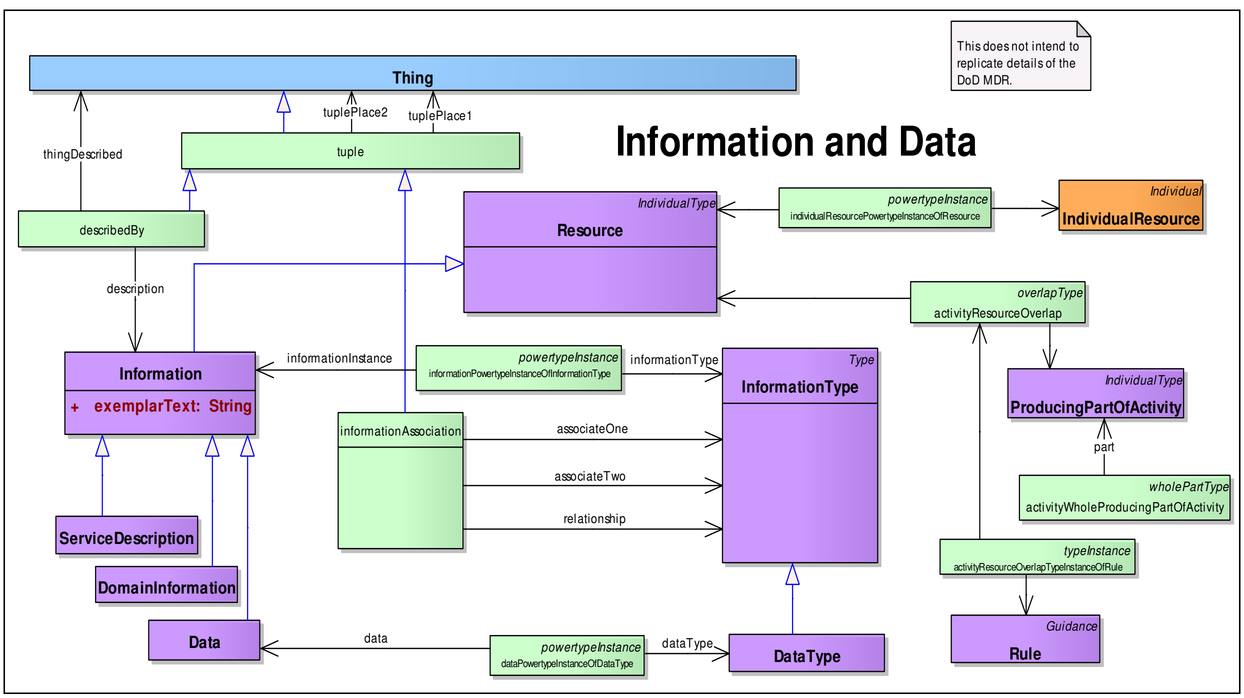


Figure 2.3.1-1: Information and Data Model Diagram

Items of note:

- The key concept in this model is that Information describes some Thing – material, temporal, or even abstract, such as a relationship (Tuple) or set (Type).
- Since Information is a Thing, Information can describe other Information, e.g., metadata.
- A Name is a type of Information in that it describes a Thing. A Name may be short or long – there is no restriction. So a textual description can be thought of a just a long Name. Information is more general than text strings and could be structured, formalized, or include other manners of description such as diagrams or images.
- Information, as a Resource Type, inherits whole-part, super-subtype, and before-after relationships.
- If Information is processable by humans or machines in a repeatable way, it is called proceduralized. Not all proceduralized information is necessarily computerized; forms are examples of data proceduralized for human repeatable processing.
- Data to be proceduralized has associations such as parts and types as well as other application specific associations. So for an Entity-Relationship model, Attributes are *has* associations with Entities and Entities are related according to verb phrases and cardinalities. In the physical schema, the fields are associated to datatypes.
- The representation for Data is not intended to cover all the details of, for instance, a relational data base management system (DBMS) underlying Meta-model, but just those aspects necessary to support the decision-making of the core processes.
- Architectural Descriptions describes architectures. An Activity Model is an example of an Architectural Description. Two subtypes of Architectural Description are called out – the AV-1 and the Manifest – because of their importance in discovery and exchange, respectively. Note that the AV-1 information can also be provided in a structured manner, using the Project data group to describe the architecture project's goals, timeline, activities, resources, productions, rules, measures, etc.

Table 2.3.1-1: DoDAF Meta-model Definitions for Information and Data

Technical Term	Composite Definition	Potentially Related Terms or Aliases
	Classes	
Data	Representation of information in a formalized manner suitable for communication, interpretation, or processing by humans or by automatic means. Examples could be whole models, packages, entities, attributes, classes, domain values, enumeration values, records, tables, rows, columns, and fields.	
DataType	Powertype of Data	
DomainInformation	Types of information within the scope or domain of the architecture.	

Technical Term	Composite Definition	Potentially Related Terms or Aliases
IndividualResource	Any specific physical or virtual entity of limited availability	
Resource	Data, Information, Performers, Materiel, or Personnel Types that are produced or consumed.	
Rule	A principle or condition that governs behavior; a prescribed guide for conduct or action	
ServiceDescription	Information necessary to interact with the service in such terms as the service inputs, outputs, and associated semantics. The service description also conveys what is accomplished when the service is invoked and the conditions for using the service.	Service Interface Description (UPDM)
Associations		
activityResourceOverlap	An overlap of an Activity with a Resource, in particular a consuming or producing Activity that expresses an input, output, consumption, or production Activity of the Resource.	output, produce
activityResourceOverlapTypeInstanceOfRule	activityResourceOverlap is a member of Rule	
activityWholeProducingPartOfActivity	A whole - part association between an Activity and the part of it that produces a Resource.	proceeds, succeeds
dataAssociation	A relationship or association between two elements of proceduralized information.	
dataPowertypeInstanceOfDataType	Data is a member of DataType	
individualResourcePowertypeInstanceOfResource	IndividualResource is a member of Resource	
ProducingPartOfActivity	A part of an Activity that produces a Resource	
activityResourceOverlap	An overlap of an Activity with a Resource, in particular a consuming or producing Activity that expresses an input, output, consumption, or production Activity of the Resource.	output, produce
activityResourceOverlapTypeInstanceOfRule	activityResourceOverlap is a member of Rule	
activityWholeProducingPartOfActivity	A whole - part association between an Activity and the part of it that produces a Resource.	proceeds, succeeds

*The Black Text indicates a Concept in DM2, the Red Text indicates an Association in DM2.

Table 2.3.1-2: DoDAF Meta-model Aliases and Composite Terms Related to Information and Data

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Architecture Description	(DoDAF V 1.5): "The Framework products portray the basic architecture data elements and relationships that constitute an Architecture Description", therefore Architecture Description: architecture data elements and relationships that make up an architecture model or product. Hence, and "Architecture Description" is an architecture model or product.	A type of Information
Definition	A statement conveying fundamental character	Description
Manual	A small reference book, especially one giving instructions.	Information
Metadata	Information about information	Thing describedBy Information where the Thing is Information
Source	One, such as a person or document, that supplies information	pedigree model
Term	A word or group of words having a particular meaning	Name
Used In	Put into service	Description whole part

2.3.2 Method

Methods for collecting and constructing models of Information and Data vary. They are taught in academic and vocational curricula. There is considerable literature, such as books, professional journals, conference proceedings, and professional magazines, on best practices, experiences, and theory. Figure 2.3.2-1 illustrates some of the basic methods for model creation.

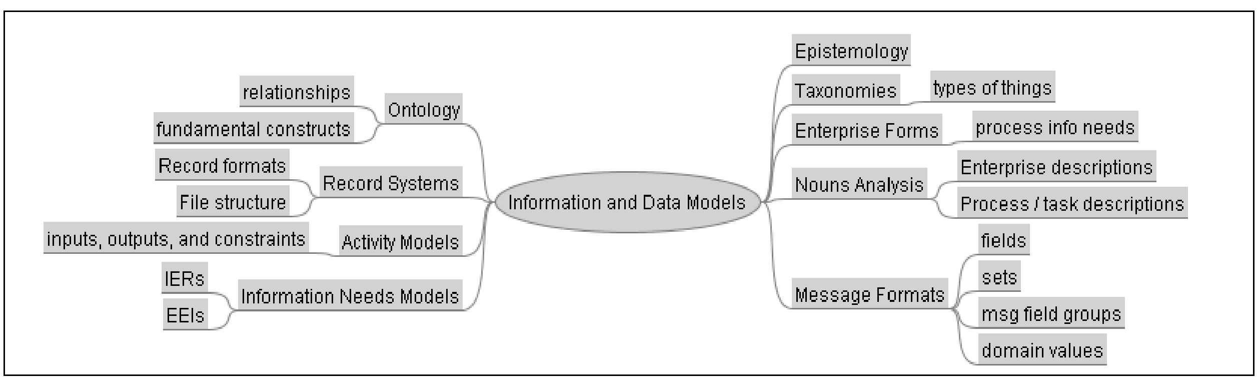


Figure 2.3.2-1: Some of the Ways Information and Data Models are Constructed

It should be noted that all methods, even the most philosophical and methodical, involve the ingestion of some record of the enterprise's processes, legacy information-keeping systems, and descriptions of what types of things it thinks it deals with. Upon collection of this raw data, terms within it are then:

- **Identified.** This is done by noting recurring or key terms.
- **Understood.** Definitions of terms are sought and researched. In most cases, there are multiple authoritative definitions. Definitions selected should be appropriate for the context of use of the term within the enterprise activities.
- **Collated and correlated**. This is done by grouping seemingly similar or related terms.
- **Harmonized.** In this step, aliases, near-aliases, and composite terms are identified. A consensus definition is formulated from the authoritative source definitions. Often super-subtype and whole-part relationships begin to emerge.

The next step is to relate the harmonized terms. Some of the relationships are implicit in the definitions and these definitions may contribute to the relationship description. At this point, the formality can vary. A formal ontological approach will type all relationships to foundational concepts such as whole-part and super-subtype. However, there are many metaphysical challenges with such an approach and it is not necessary for many applications. This constitutes the conceptual-level of modeling, defined and related terms, now considered concepts because the definitions and relationships lend a meaning to the terms. The conceptual model should be understandable by anyone knowledgeable about the enterprise. Super-subtype and whole-part relationships can provide cognitive economy. Conceptual models can be done in Entity-Relationship or UML Class model style although any format that documents definitions and relationships is functionally equivalent. Note that the subtype concept in UML generally results in the subclass inheriting properties from the supertype while in Entity-Relationship (E-R) modeling only the identifying keys are inherited directly; the other supertype properties are available after a join operation.

At the logical-level, relationships may have cardinalities or other rules added that indicate how many of one instance of something relates to an instance of something else, the necessity of such relations, and so on. The concepts may also be attributed, meaning they will be said to have some other concept, e.g., the concept of eye has the concept of color. Often at the logical-level, the relationships are reified or made concrete or explicit. At the logical-level, this is done in case there is something additional that needs to be stated about the relationship, e.g., the quantity of some part of something or the classification of the related information, which may be different from the classification of the individual elements. There may also be considerations of normalization, meaning that the database structure is modified for general-purpose querying and is free of certain undesirable characteristics during insertion, update, and deletion operations that could lead to a loss of data integrity. The benefits of normalization are to uncover additional business rules that might have been overlooked without the analytical rigor of normalization and ensure the precise capture of business logic. The logical model, though having more parts than the conceptual model, should still be understandable by enterprise experts. At the logical-level, some sort of modeling style is normally used such as Entity-Relationship or UML Class modeling.

At the physical-level, the exact means by which the information is to be exchanged, stored, and processed is determined. At this level, we are talking about data. The efficiency, reliability, and assured repeatability of the data use are considered. The datatypes, the exact format in which the data is stored are determined. The datatype needs to accommodate all the data that is permissible to store or exchange yet be efficient and disallow formats that are not permissible. The entities may be de-normalized for efficiency so that join operations don't have to be performed. Logical associations may be replaced with identifiers (e.g., as associative entities or foreign or migrated keys in Entity Relationship Diagrams [ERDs] or explicit identifier attributes or association classes in class models). Keys, identifiers, and other means of lookup are setup. Indexes, hashes, and other mechanisms may be setup to allow data access in accordance with requirements. The physical target may be any of the following:

- Database – relational, object, or flat file.
- Message exchange format – document (e.g., XML), binary (e.g., Interface Definition Language (IDL)).
- Cybernetic (human – machine), e.g., print or screen formats, such as forms.

2.3.3 Use

Information and Data models are used in the following ways:

- Information models materialize for enterprise participants what things are important to the enterprise and how they are related.
- Information models can serve as a basis for standardization of terminology and concept inter-relationships for human, machine, and human-machine communications.
- Information models can provide cognitive compactness for an enterprise's personnel through the use of taxonomies and other relationship structures. This can improve clarity, efficiency, accuracy, and interoperability of action within the enterprise.
- Information models document the scope of things the enterprise is concerned with in a form that allows comparison with other communities of interest to reveal common interests.
- Data models can be used to generate persistent storage of information such as in databases.
- Data models can be used to generate formats for exchanging data between machines, humans, and machine-to-human. For example, an XSD is a physical data model that is generally an exchange format. Web services can be used with relational DBMS' to generate XML for exchange in the format of the data model implemented in the DBMS. The underlying data models (the physical data model and the exchange data format) do not have to be the same; a translator or mediator may be invoked to translate during the exchange.
- Data models can be used to compare whether Performers are compatible for data exchange.
- Data and information models can be used to determine if components of a portfolio have:
  - Overlapping data or information production (an indication of potential unwanted redundancy).
  - Interdependent data or information needs.
- Data and information models can be used to determine if a proposed capability will interoperate, be redundant with, or fill gaps in conjunction with other capabilities.
- Data and information models can be used during milestone reviews to verify interoperability, non-redundancy, and sufficiency of the solution.
- Information models are useful in initial discovery of a service, to know what sorts of information it may provide access to or its accessed capabilities need. An information model is part of a service description.

- Data models are useful in knowing how to interact with a service and the capabilities it provides and for establishing the service contract. A data model is part of a service description and service contract.
- COI coordination and harmonization.
- Data assets management.
- Database/sources consolidation and migration.
- Authoritative sources identification and management.
- Mediation and cross-COI sharing.
- Standards definition and establishment.

2.4 Activities

An Activity is work, not specific to a single organization, weapon system, or individual, that transforms inputs into outputs or changes their state. Activity has been a central concept in architectures since the early DoDAF definitions. At that time the focus was on:

- Business activities and how they could be re-engineered or streamlined.
- Strategic, theater, operational, and tactical tasks.
- Activities (System Functions) performed by Systems.
- Operational activities performed by organizations (and their Types) and in the course of conducting an operational role.

The concept remains central in net-centric, service-oriented, Capabilities-focused, and Project-aligned architectures, as well as Goal-responsive architectures, such as:

- The Activities involved in the service mechanism and the Capabilities thereby accessed.
- As a part of a Service description.
- Part of a Capability.
- The core of a Project.
- The response to a Goal.

2.4.1 Data

The DoDAF Meta-model for the data comprising Activities is shown in **Figure 2.4.1-1**. The figure may be hard to read in a hardcopy printout but a version at full-resolution which can be zoomed in is published in the online DoDAF V2.0 Meta-model Data Dictionary. Definitions for the model terms are in **Table 2.4.1-1**. Aliases and composite terms related to Activities are shown in **Table 2.4.1-2**. Authoritative Source definitions and rationale are provided in the DoDAF V2.0 Meta-model Data Dictionary. Note that foundational classes are generally not shown on data group diagrams; this foundational material is in Section 2. This includes super-subtype, whole-part, temporal-whole-part, overlap, type-instance (member-of), and before-after patterns. Also not shown are the data structures for classification marking of architecture information at the whole and element (portion) levels using the IC-ISM. The schema for the IC-ISM is in Volume 3. Lastly, note that the size of the icons is not indicative of their importance; sizes are adjusted to reduce line crossings and bends to make the diagrams easier to understand.

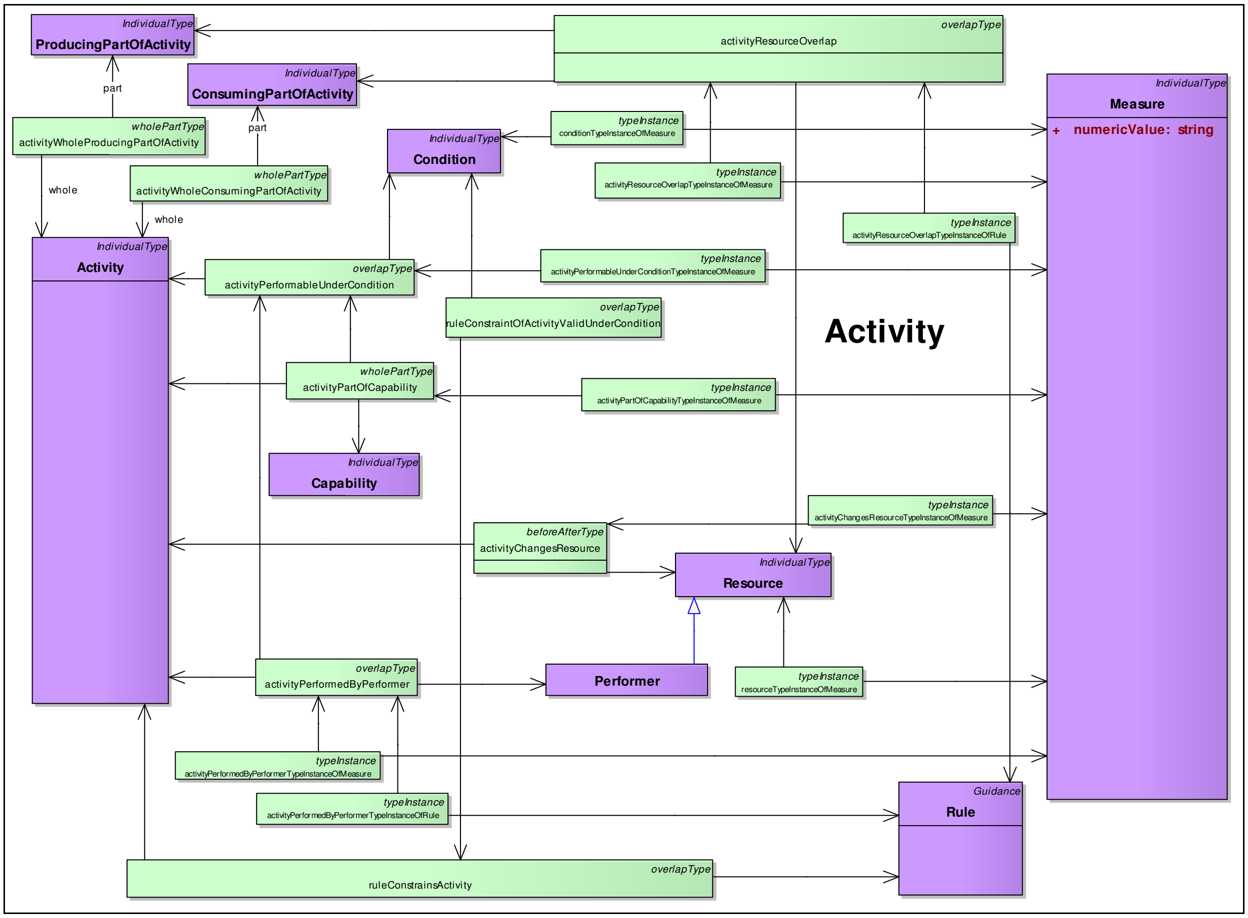


Figure 2.4.1-1: DoDAF Meta-model for Activities

Table 2.4.1-1: DoDAF Meta-model Definitions for Activities

Technical Term	Composite Definition	Potentially Related Terms or Aliases
	Classes	
Activity	Work, not specific to a single organization, weapon system or individual that transforms inputs (Resources) into outputs (Resources) or changes their state.	Action, Process Operational Activity, Processes, Function, System Function, Operation, Task, Plan, Project
Capability	The ability to achieve a Desired Effect under specified [performance] standards and conditions through combinations of ways and means [activities and resources] to perform a set of activities.	
Condition	The state of an environment or situation in which a Performer performs.	

Table 2.4.1-1: DoDAF Meta-model Definitions for Activities

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Measure	The magnitude of some attribute of an individual.	
Performer	Any entity - human, automated, or any aggregation of human and/or automated - that performs an activity and provides a capability.	Actor, Agent, Capability Configuration (MODAF)
Resource	Data, Information, Performers, Materiel, or Personnel Types that are produced or consumed.	
Rule	A principle or condition that governs behavior; a prescribed guide for conduct or action	
Associations		
activityChangesResource	Represents that an activity was / is / will-be the cause of change in the effected object with a before-after relationship.	
activityChangesResourceTypeInstanceOfMeasure	activityChangesResource is a member of Measure	
activityPartOfCapability	A disposition to manifest an Activity. An Activity to be performed to achieve a desired effect under specified [performance] standards and conditions through combinations of ways and means.	
activityPartOfCapabilityTypeInstanceOfMeasure	activityPartOfCapability is a member of Measure	
activityPerformableUnderCondition	Represents that an activity was / is / can-be/ must-be conducted under certain conditions with a spatiotemporal overlap of the activity with the condition.	
activityPerformableUnderConditionTypeInstanceOfMeasure	activityPerformableUnderCondition is a member of Measure	
activityPerformedByPerformer	An overlap between a Performer and an Activity that is non-specific as to whether: 1. the Activity is solely performed by the Performer 2. the Activity is performed by several Performers 3. the Performer performs only this Activity 4. the Performer performs other Activities	
activityPerformedByPerformerTypeInstanceOfMeasure	activityPerformerOverlap is a member of Measure	

Table 2.4.1-1: DoDAF Meta-model Definitions for Activities

Technical Term	Composite Definition	Potentially Related Terms or Aliases
activityPerformedByPerformerTypeInstanceOfRule	activityPerformerOverlap is a member of Rule	
activityResourceOverlap	An overlap of an Activity with a Resource, in particular a consuming or producing Activity that expresses an input, output, consumption, or production Activity of the Resource.	output, produce
activityResourceOverlapTypeInstanceOfMeasure	activityResourceOverlap is a member of Measure	
activityResourceOverlapTypeInstanceOfRule	activityResourceOverlap is a member of Rule	
activityWholeConsumingPartOfActivity	A whole - part association between an Activity and the part of it that consumes a Resource.	input, consume
activityWholeProducingPartOfActivity	A whole - part association between an Activity and the part of it that produces a Resource.	proceeds, succeeds
conditionTypeInstanceOfMeasure	Condition is a member of Measure	
ConsumingPartOfActivity	A part of an Activity that consumes a Resource	
ProducingPartOfActivity	A part of an Activity that produces a Resource	
resourceTypeInstanceOfMeasure	ResourceType is a member of Measure	
ruleConstrainsActivity	An overlap between a Rule and the Activities it allows	
ruleConstraintOfActivityValidUnderCondition	An overlap between the Activities constrained by a Rule and the Conditions under which the Rule applies	

*The Black Text indicates a Concept in DM2, the Red Text indicates an Association in DM2.

Table 2.4.1-2: Aliases and Composite Terms Related to Activities

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Actor	A performer that is external to and invokes the performer to be architected.	User, customer, agent, performer
Business Process	A functionally or temporally linked collection of structured activities/ tasks aimed at producing specific services and products for an end-user.	Activity, Process, Function, Job, Chore, Assignment.

Table 2.4.1-2: Aliases and Composite Terms Related to Activities

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Concept of Operations	A general idea derived or inferred from specific instances or occurrences of major planning and operating functions	Activity
Course of Action	A path towards a goal	Mission, strategy, plan
Doctrine	The body of principles by which an enterprise seeks to guide its activities.	Composite of Activities and their structure, sequencing, state transitions, their assignment to Organizations, Organization structure, and Rules
Effect	The result, outcome, or consequence of an action.	A change in the state of a Resource as a result of some Activity. Goal, Objective, Desired Result, Outcome, Consequence, Effect Object
Enduring Task	A continuing function to be performed	Activity
Event	Something that happens at an instant in the world, i.e., a zero-duration process (Activity).	Milestone, Trigger, Activity
Function	The action for which a person or thing is specially designed, fitted, used or intended to accomplish or execute.	Activity, Process, Job, Chore, Assignment.
Functional Dependency	A constraint on, or dependence of, a function on one or more outside influences, conditions, functions, triggers or events.	Composite of Activity with Constraint or dependence on one or more Conditions, Activities, triggers (composite of Activity and Event), Events.
Operational Activity	An activity is an action performed in conducting the business of an enterprise. It is a general term that does not imply a placement in a hierarchy (e.g., it could be a process or a task as defined in other documents and it could be at any level of the hierarchy of the Operational Activity Model). It is used to portray operational actions not hardware/software system functions. (DoDAF)	Activity

Table 2.4.1-2: Aliases and Composite Terms Related to Activities

Technical Term	Composite Definition	Potentially Related Terms or Aliases
PerformerSupportingActivity	A type of Activity - Performer overlap between a Performer and those Activities which may not necessarily be carried out by the Performer but which are necessary for the performance of the Activity	ActivityPerformerOverlap of the Activities actually performed by the Performer and then Activity overlaps between them and the supported Activity
Plan	A set of Activities that result in a Goal, Desired Effect, outcome, or objective.	Course of Action, Activity aggregate (temporal or otherwise)
Process	A logical, systematic sequence of activities, triggered by an event, producing a meaningful output.	Activity, Process, Function, Job, Chore, Assignment.
Role	A set of similar or otherwise logically related activities, implying a set of skills or capabilities, to which a performer may be assigned.	Performer, Activity, and their overlap
System Function	A function that is performed by a system. Although commonly used to refer to the automation of activities, data transformation or information exchanges within IT systems, it also refers to the delivery of military capabilities.	Activity, Process, Function, Job, Chore, Assignment.
Tactics, Techniques, and Procedures (TTP)	The actions and methods that implement doctrine and describe how forces will be employed in operations	Activity, Rule, Organization and their inter-relationships
Task	A action, activity or undertaking enabling missions, activities or functions to be performed or accomplished.	Activity, Process, Function, Job, Chore, Assignment.

Considerations with the use of aliases:

- Because of the inheritance from the foundation of whole-part (so some Activities are parts of others), temporal whole-part, and before-after (some some Activities happen before others), there is no longer a need to try to use different terms (e.g., Task, Process, Function) to distinguish larger Activities or sequences of Activities.
- The performance of an Activity has been deconstructed from the concept of Activity so there is no need to use different terms for Activities performed by Systems (e.g., System Functions) from those whose performer is unspecified (e.g., Operational Activities).

2.4.2 Method

A method to capture Activity data is described in **Table 2.4.2-1**.

Table 2.4.2-1: Activity Data Capture Method Description

Methodology Description	Capture Data for Architectural Description of Activity
Definition:	Define a method by which activities can be defined and architected in a manner that enables them to be used in composing the major decision processes of the DoD. The Activity Method includes characteristics used to ensure proper definition of activities as well as a process by which architectural information relative to activities can be captured and structured to enable it to support the major decision processes of the Department (e.g., PPBE, PfM, and JCIDS).
Input:	• Enterprise/Component/Program vision documentation • Enterprise/Component/Program strategic documentation • Mission Statements • Directives • Objectives and goals documentation • Concept of operations documentation • Doctrine
Method:	This method is described in two sections. The first section describes the attributes of an activity. The second section describes steps that can be taken to architect an activity. **Attributes of a Well-Defined Activity** A well-defined activity consists of: • resource inputs • resource outputs • activity production and consumption relationships • rules that constrain the activity as performed by certain performers • rules that constrain the resource production and consumption (rules about resource production and consumption, e.g., resource exchange IA rules) • conditions under which those rules apply • conditions under which the activity is to be performed • measures associated with the activity • measures associated with the production and consumption of resources and performers To clarify some of the terms: • Inputs are the triggers that cause an activity to occur are other activities or events (zero duration activities). • Outputs are the results of activity performance. These can be outputs of products, services, or requirements for further action, or outcomes (i.e., demonstration that an action has produced a desired change). • Rules include doctrine, regulations, or other documents that prescribe how an activity is to take place, what course the activity must follow, and, what form or format is expected/required for the result. • Resources are those things that assist in performance of the activity. These can be physical, logical, technological, or human resources. Resources are inputs and outputs of activities performed by performers. Attributes of a well-defined activity also include quality, focus, granularity and modularity. Quality: A high quality activity is a modular representation of the specific steps taken to perform the action being described, along with its sub-activities, services

Table 2.4.2-1: Activity Data Capture Method Description

Methodology Description	Capture Data for Architectural Description of Activity
	and systems used. An activity can be created and described in either a baseline or future (i.e., "To-Be") model. Focus: Well-focused activities are both necessary and sufficient (as a group) to achieve the desired action. Granularity: Activities should be defined at a level of granularity that is: • meaningful and consistent in an operations context • appropriate for intended use by the stakeholders • consistent with approved taxonomies to be used to help architecturally define the activity • consistent with the DoD EA Reference Models to support federation Modularity: Each Activity should describe a complete action. **Minimum steps for architecting activities** • Define the activity. • Provide a name for the activity (Each activity should have a unique identifier). • Define the triggers (inputs) that cause activity performance • Identify the steps taken to perform the activity, to include linkages to other activities (i.e., inputs from other actions that trigger the activity being described). • Identify the rules, requirements, and limitations on the activity. • Identify the expected results and outputs of activity performance.
Primary Output:	Information, physical products, inputs to other activities and their performers.
Secondary Output:	Personnel, Roles, Services, Systems, Rules, Organizations that relate to the activity.
Disciplines:	Structured Analysis, Object-Oriented Analysis, Business Process Analysis. Activity modeling, functional decomposition

2.4.3 Use

Data for Activities are used as follows:

Data for activity is used to describe how an activity is or will be performed, and often when it is performed as a part of some larger process. In general, data on activity describes work being performed for some purpose. The data describes how the input (i.e., trigger or other artifact that causes an action to occur) interacts through business rules to perform the requested activity, and produce the desired output.

2.5 Training/Skill/Education

The Training/Skill/Education data group provides information on the identification of data and information used to define, describe, and promulgate training requirements, skills sets required for specific capabilities and operations, and the formal education required for commissioned and non-commissioned officers of all grades.

Training provides an understanding of military procedures. Skill Sets are those sets of personal capabilities and competencies required to perform a designated military task. Education is the knowledge or skill obtained or developed by an organized learning process that provides a specified kind or level of information.

2.5.1 Data

The DoDAF Meta-model for the data comprising Training/Skill/Education is shown in **Figure 2.5.1-1.** The figure may be hard to read in a hardcopy printout but a version at full-resolution which can be zoomed in is published in the online DoDAF V2.0 Meta-model Data Dictionary. Definitions for the model terms are in **Table 2.5.1-1** and **Table 2.5.1-2.** Authoritative source definitions, aliases, and rationale are provided in the DoDAF V2.0 Meta-model Data Dictionary. Note that foundational classes are generally not shown on data group diagrams; this foundational material is in Section 2. This includes super-subtype, whole-part, temporal-whole-part, overlap, type-instance (member-of), and before-after patterns. Also not shown are the data structures for classification marking of architecture information at the whole and element (portion) levels using the IC-ISM. The schema for the IC-ISM is in Volume 3. Lastly, note that the size of the icons is not indicative of their importance; sizes are adjusted to reduce line crossings and bends to make the diagrams easier to understand.

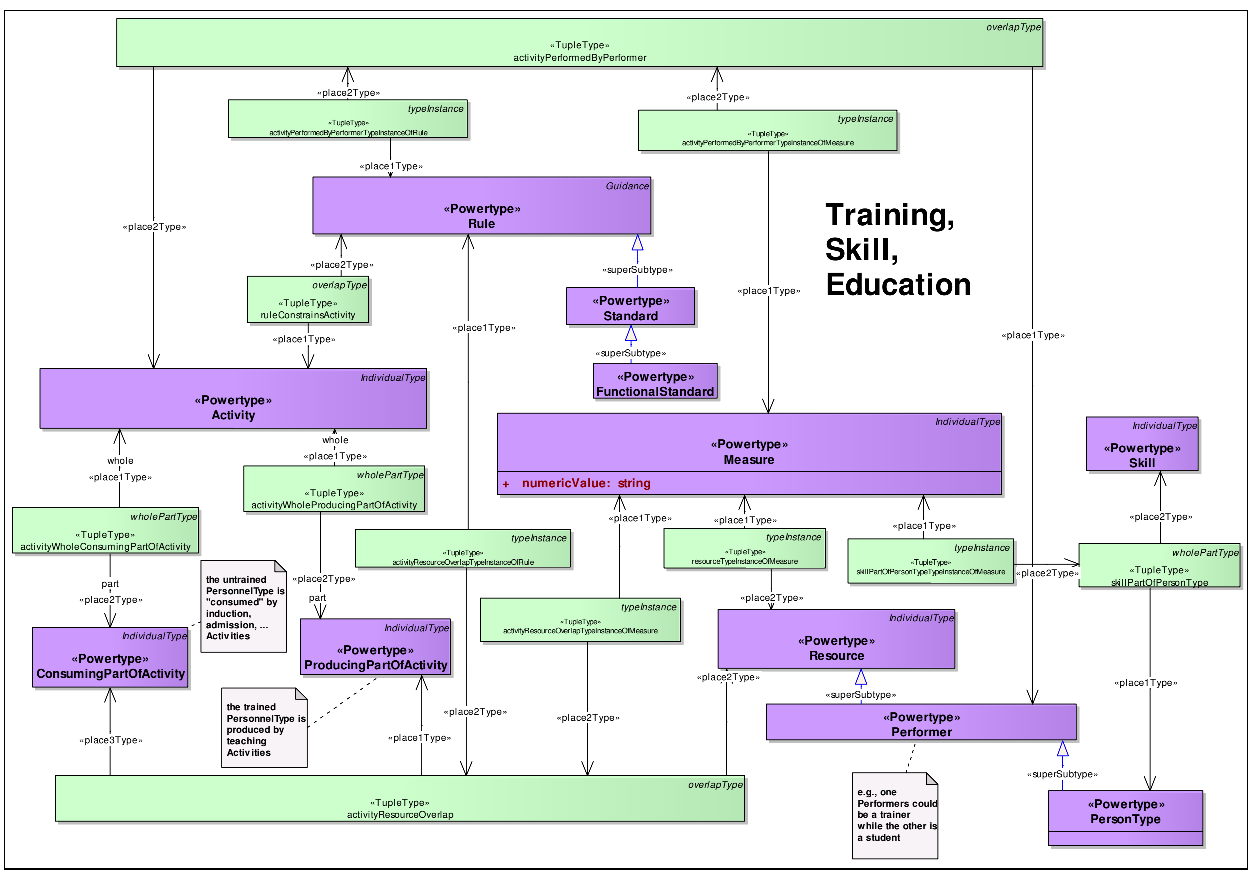


Figure 2.5.1-1: DoDAF Meta-model for Training/Skill/Education

Table 2.5.1-1 below provides the DoDAF Meta-model definitions for the Training/Skill/ Education data group presented in the model in Figure 2.5.1-1, above.

Table 2.5.1-1: DoDAF Meta-model Definitions for Training/Skill/Education

Technical Term	Composite Definition	Potentially Related Terms or Aliases
	Concepts	
Activity	Work, not specific to a single organization, weapon system or individual that transforms inputs (Resources) into outputs (Resources) or changes their state.	Action, Process Operational Activity, Processes, Function, System Function, Operation, Task, Plan, Project
FunctionalStandard	Functional standards set forth rules, conditions, guidelines, and characteristics.	
Measure	The magnitude of some attribute of an individual.	
Performer	Any entity - human, automated, or any aggregation of human and/or automated - that performs an activity and provides a capability.	Actor, Agent, Capability Configuration (MODAF)
PersonType	A category of persons defined by the role or roles they share that are relevant to an architecture.	Role
Resource	Data, Information, Performers, Materiel, or Personnel Types that are produced or consumed.	
Rule	A principle or condition that governs behavior; a prescribed guide for conduct or action	
Skill	The ability, coming from one's knowledge, practice, aptitude, etc., to do something well.	Training, Knowledge, Ability
Standard	A formal agreement documenting generally accepted specifications or criteria for products, processes, procedures, policies, systems, and/or personnel.	

Associations		
activityPerformedByPerformer	An overlap between a Performer and an Activity that is non-specific as to whether: 1. the Activity is solely performed by the Performer 2. the Activity is performed by several Performers 3. the Performer performs only this Activity 4. the Performer performs other Activities	
activityPerformedByPerformerTypeInstanceOfMeasure	activityPerformerOverlap is a member of Measure	
activityPerformedByPerformerTypeInstanceOfRule	activityPerformerOverlap is a member of Rule	
activityResourceOverlap	An overlap of an Activity with a Resource, in particular a consuming or producing Activity that expresses an input, output, consumption, or production Activity of the Resource.	output, produce
activityResourceOverlapTypeInstanceOfMeasure	activityResourceOverlap is a member of Measure	
activityResourceOverlapTypeInstanceOfRule	activityResourceOverlap is a member of Rule	
activityWholeConsumingPartOfActivity	A whole - part association between an Activity and the part of it that consumes a Resource.	input, consume
activityWholeProducingPartOfActivity	A whole - part association between an Activity and the part of it that produces a Resource.	proceeds, succeeds
ConsumingPartOfActivity	A part of an Activity that consumes a Resource	
ProducingPartOfActivity	A part of an Activity that produces a Resource	
resourceTypeInstanceOfMeasure	ResourceType is a member of Measure	
ruleConstrainsActivity	An overlap between a Rule and the Activities it allows	
skillPartOfPersonType	An overlap between a Personnel Type and the Skills it entails	
skillPartOfPersonTypeTypeInstanceOfMeasure	skillPartOfPersonType is a member of Measure	

*The Black Text indicates a Concept in DM2, the Red Text indicates an Association in DM2.

Table 2.5.1-2: Aliases and Composite Terms Related to Training/Skill/Education

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Ability	The quality of being able to perform	
Doctrine	The body of principles by which an enterprise seeks to guide its activities.	Composite of Activities and their structure, sequencing, state transitions, their assignment to Organizations, Organization structure, and Rules
Event	Something that happens at an instant in the world, i.e., a zero-duration process (Activity).	Milestone, Trigger, Activity
Instruction	An imparted or acquired item of knowledge	Skill
Means	An action or system by which a result is brought about; a method	Tactics, Strategy, Project, any DOTMLPF elements
OccupationalTraining	To make proficient by instruction and practice in particular knowledge or skills.	Skill

2.5.2 Training/Skill/Education Information Capture Method

A method to capture Training/Skill/Education data is described in **Table 2.5.2-1**.

Table 2.5.2-1: Training/Skill/Education Data Capture Method Description

Methodology Description	Capture Data for Architectural Description of Capability
Definition:	Per the DoDAF V2.0, training/skill/education data provides necessary information needed to determine specific training, skills, and education requirements necessary to execute a particular activity. The following information describes a process by which data associated with training, skills, or education can be captured to support development of an enterprise architecture.
Input:	• Training Information • Training Policy • Training Performance Measures • Training Triggering Events • Skill Information • Education Information • Education Policy • Education Performance Measures • Education Triggering Events

Table 2.5.2-1: Training/Skill/Education Data Capture Method Description

Methodology Description	Capture Data for Architectural Description of Capability
Method:	Training/Skill/Education Data is a type of Information which is collected to determine when specific activities are executed by a performer who executes activities to create, fill, transfer, or adjust positions that execute those activities. Conduct of training or education necessary to acquire necessary skills are provided by a service provider. The following steps can be taken to capture Training/Skill/Education information to support the intended purpose of the architecture: • Identify and capture the operations, business activities and processes requiring training/skill/education. • Describe specific training/skill/education requirements necessary to perform some specific action. • Identify the organization needed to perform the services required to provide the necessary training/skill/education. • Using the Training/Skill/Education Requirement Description, capture the information to be provided by the training/education service and the information required to be produced by the training/education service to provide required skills. • Define and capture the rules applied to the information produced by the training/education service. Also define and capture the rules governing or constraining the use of the training/education service in skill development. • If not captured as part of the previously mentioned rules, define and capture the measures that will be used to gauge the performance of the training/education service as applied to required skills. • Identify and capture other services or systems on which the training/education service is dependent or are dependent on the service.
Primary Output:	Traceability to: • Capabilities • Business activities • Activities • Performance measure
Secondary Output:	Organization responsible for providing the service.
Disciplines:	Structured analysis, Object-oriented Analysis (UML or SysML), BPMN

2.5.3 Use

Training and Education, in their broadest sense, are well-defined ways to ensure that requisite skills are available and can be applied to execute a unit of work that provides a useful result to a consumer. Training and Education to acquire Skills are activities performed by a Service provider (Performer) to achieve desired results for a Service consumer (other Performer). Training and Education Services may utilize web-based technology or functions, although their use in the net-centric environment generally involves the use of web-based, or network-based, resources.

Functionally, a Training and Education Services to enable required Skills are a set of strictly delineated functionalities, restricted to answering the what-question, independent of construction or implementation issues[10].

There are a number of uses for architecture information to support Training and Education:

- First, hierarchical descriptions of activities with increasing levels of decomposition assist training designers when mapping out course content. By understanding the activity, related activities, and sub-activities the trainer can decide what is appropriate for course content and the logical order in which it should be presented. Thorough understanding of the activities to be trained will aid in focusing lesson plan development and measures of student comprehension.
- Second, an appreciation for the complexity of the activities derived from architectural data can provide insight about what knowledge, skills, and abilities are prerequisite for students prior to participation in increasingly advanced training.
- Third, an understanding of composite activities comprised of component that are sequenced over time and the events and triggers that initiate them, can assist in planning a logical flow for training which will provide the student with an understanding of how an overall process or procedure occurs and where they fit in that process.
- Lastly, an understanding of the existing automation that supports or enables the activities being trained, can aid in planning curricula for appropriate levels of training on information technology where and when applicable throughout the Program of Instruction (POI). These concepts and constructs can be applied across a broad educational spectrum from institutional to unit and to individual training and has the same value for classroom or hands-on instruction. Utilizing architectural information in the planning and conduct of training can insure that the correct training is received at the appropriate educational level to produce the desired skills and abilities in the student.

2.6 Capability

The Capability Data Group provides information on the collection and integration of activities that combine to respond to a specific requirement. A capability, as defined here is *"the ability to achieve a desired effect under specified standards and conditions through combinations of means and ways to perform a set of tasks."* This definition is consistent with that contained in the JCIDS Instruction published by the Joint Staff[11].

2.6.1 Data

The DoDAF Meta-model for the data comprising Capability is shown in **Figure 2.6.1-1**. The figure may be hard to read in a hardcopy printout but a version at full-resolution which can be zoomed in is published in the online DoDAF V2.0 Meta-model Data Dictionary. Definitions for the model terms are in **Table 2.6.1-1**. Aliases and composite terms related to Capabilities are shown in **Table 2.6.1-2**. Authoritative Source definitions, aliases, and rationale are provided in the DoDAF V2.0 Meta-model Data Dictionary. Note that foundational classes are generally not shown on data group diagrams; this foundational material is in Section 2. This includes super-subtype, whole-part, temporal-whole-part, overlap, type-instance (member-of), and before-after patterns. Also not shown are the data structures for classification marking of architecture

[10] North Atlantic Treaty Organization (NATO), NATO Architecture Framework, V.3, DRAFT 0.9 14 July 2006.
[11] JCIDS Cite

information at the whole and element (portion) levels using the IC-ISM. The schema for the IC-ISM is in Volume 3. Lastly, note that the size of the icons is not indicative of their importance; sizes are adjusted to reduce line crossings and bends to make the diagrams easier to understand.

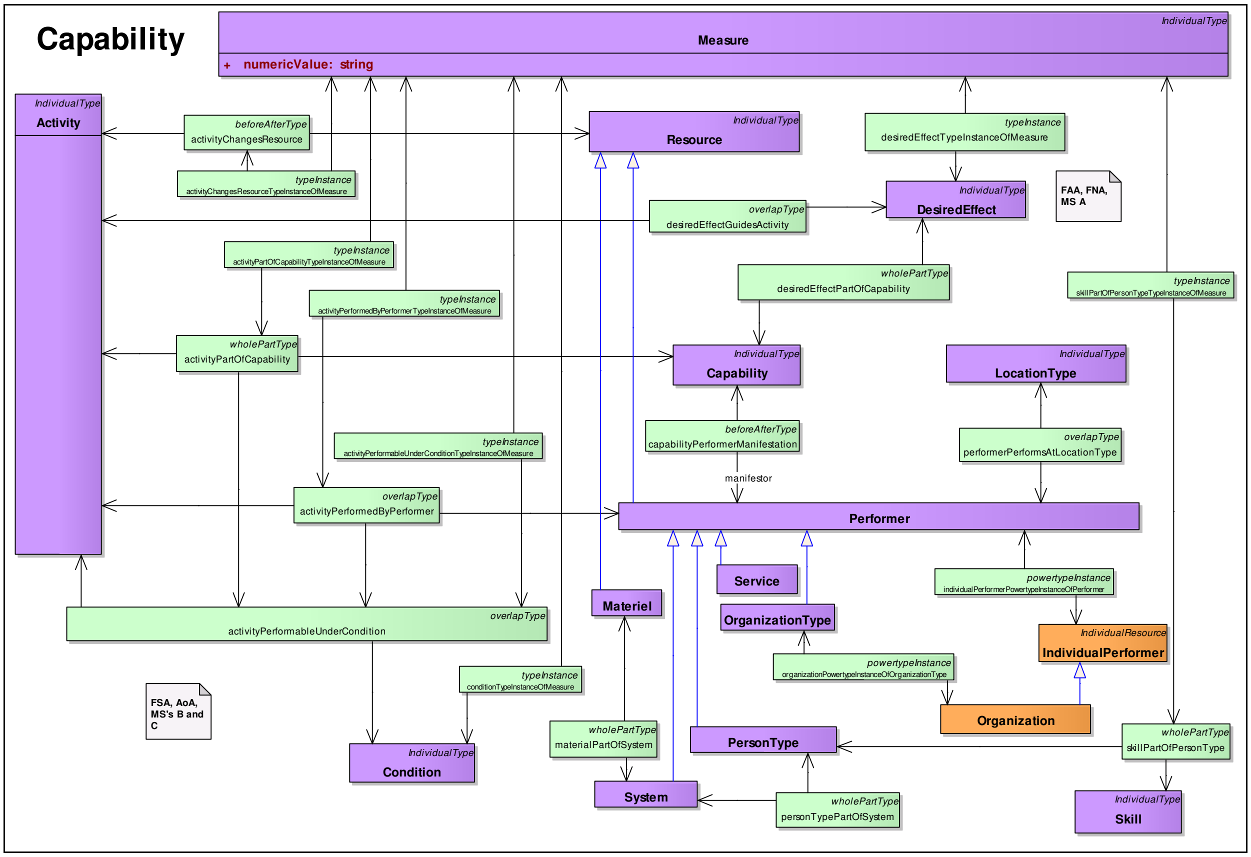


Figure 2.6.1-1: DoDAF Meta-model for Capability

Table 2.6.1-1 below provides the DoDAF Meta-model definitions for the Capability data group presented in the model in Figure 2.6.1-1, above.

Table 2.6.1-1: DoDAF Meta-model Definitions for Capability

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Classes		
Activity	Work, not specific to a single organization, weapon system or individual that transforms inputs (Resources) into outputs (Resources) or changes their state.	Action, Process Operational Activity, Processes, Function, System Function, Operation, Task, Plan, Project
Capability	The ability to achieve a Desired Effect under specified [performance] standards and conditions through combinations of ways and means [activities and resources] to perform a set of activities.	
Condition	The state of an environment or situation in which a Performer performs.	
DesiredEffect	The result, outcome, or consequence of an action [activity].	DesiredEffectType IndividualDesiredEffe ct
IndividualPerformer	A specific thing that can perform an action	
LocationType	The powertype of Location	
Materiel	Equipment, apparatus or supplies that are of interest, without distinction as to its application for administrative or combat purposes.	
Measure	The magnitude of some attribute of an individual.	
Organization	A specific real-world assemblage of people and other resources organized for an on-going purpose.	Department, Agency, Enterprise
OrganizationType	A type of Organization	
Performer	Any entity - human, automated, or any aggregation of human and/or automated - that performs an activity and provides a capability.	Actor, Agent, Capability Configuration (MODAF)
PersonType	A category of persons defined by the role or roles they share that are relevant to an architecture.	Role
Resource	Data, Information, Performers, Materiel, or Personnel Types that are produced or consumed.	

Table 2.6.1-1: DoDAF Meta-model Definitions for Capability

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Service	A mechanism to enable access to a set of one or more capabilities , where the access is provided using a prescribed interface and is exercised consistent with constraints and policies as specified by the service description. The mechanism is a Performer. The "capabilities" accessed are Resources -- Information, Data, Materiel, Performers, and Geo-political Extents.	
Skill	The ability, coming from one's knowledge, practice, aptitude, etc., to do something well.	Training, Knowledge, Ability
System	A functionally, physically, and/or behaviorally related group of regularly interacting or interdependent elements.	
Associations		
activityChangesResource	Represents that an activity was / is / will-be the cause of change in the effected object with a before-after relationship.	
activityChangesResourceTypeInstanceOfMeasure	activityChangesResource is a member of Measure	
activityPartOfCapability	A disposition to manifest an Activity. An Activity to be performed to achieve a desired effect under specified [performance] standards and conditions through combinations of ways and means.	
activityPartOfCapabilityTypeInstanceOfMeasure	activityPartOfCapability is a member of Measure	
activityPerformableUnderCondition	Represents that an activity was / is / can-be/ must-be conducted under certain conditions with a spatiotemporal overlap of the activity with the condition.	
activityPerformableUnderConditionTypeInstanceOfMeasure	activityPerformableUnderCondition is a member of Measure	

Table 2.6.1-1: DoDAF Meta-model Definitions for Capability

Technical Term	Composite Definition	Potentially Related Terms or Aliases
activityPerformedByPerformer	An overlap between a Performer and an Activity that is non-specific as to whether: 1. the Activity is solely performed by the Performer 2. the Activity is performed by several Performers 3. the Performer performs only this Activity 4. the Performer performs other Activities	
activityPerformedByPerformerTypeInstanceOfMeasure	activityPerformerOverlap is a member of Measure	
capabilityPerformerManifestation	A couple that represents the capability that a performer manifests	
conditionTypeInstanceOfMeasure	Condition is a member of Measure	
desiredEffectGuidesActivity	A couple that represents how a desired effect guides an activity	
desiredEffectPartOfCapability	A couple that represents the whole part relationship between a desired effect and a capability	
desiredEffectTypeInstanceOfMeasure	DesiredEffect is a member of Measure	
individualPerformerPowertypeInstanceOfPerformer	IndividualPerformer is a member of Performer	
materialPartOfSystem	A whole-part association between a System (whole) and the Materiel parts of the System. (A System can have Personnel Type and Organizational components.)	
organizationPowertypeInstanceOfOrganizationType	Organization is a member of OrganizationType	
performerPerformsAtLocationType	The relationship that describes the location of a performer or type of performer	
personTypePartOfSystem	A overlap between a Personnel Type and a System in which it performs	
skillPartOfPersonType	An overlap between a Personnel Type and the Skills it entails	
skillPartOfPersonTypeTypeInstanceOfMeasure	skillPartOfPersonType is a member of Measure	

*The Black Text indicates a Concept in DM2, the Red Text indicates an Association in DM2.

Table 2.6.1-2: Aliases and Composite Terms Related to Capability

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Actor	A performer that is external to and invokes the performer to be architected.	User, customer, agent, performer
Business Process	A functionally or temporally linked collection of structured activities/ tasks aimed at producing specific services and products for an end-user.	Activity, Process, Function, Job, Chore, Assignment.
Capability Configuration	A combination of organizational aspects (with their competencies) and equipment that combine to provide a capability.	aggregated Performer
Capability Increment	A capability that can be effectively developed, produced, acquired, deployed and sustained.	composite of Capability temporal part (with time period) - Performer (and its time period)
Desired Result	The wished for result, outcome, or consequence of an action. A desired result may be either a goal or an objective.	desired effect, desired outcome, desired consequence
Doctrine	The body of principles by which an enterprise seeks to guide its activities.	Composite of Activities and their structure, sequencing, state transitions, their assignment to Organizations, Organization structure, and Rules
Effect	The result, outcome, or consequence of an action.	A change in the state of a Resource as a result of some Activity. Goal, Objective, Desired Result, Outcome, Consequence, Effect Object
Function	The action for which a person or thing is specially designed, fitted, used or intended to accomplish or execute.	Activity, Process, Job, Chore, Assignment.
Functional Dependency	A constraint on, or dependence of, a function on one or more outside influences, conditions, functions, triggers or events.	Composite of Activity with Constraint or dependence on one or more Conditions, Activities, triggers (composite of Activity and Event), Events.
Means	An action or system by which a result is brought about; a method	Tactics, Strategy, Project, any DOTMLPF elements

Table 2.6.1-2: Aliases and Composite Terms Related to Capability

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Network	An interconnected or interrelated chain, group, or system	System, group of systems, chain of systems
Operational Condition	A statement of the values or states needed for the execution of actions within the processes and transactions of an enterprise.	Condition
Performer Role	Any entity - human, automated, or any aggregation of human and/or automated - that performs a function, activity, or role, or provides a capability.	1. Composite of Performer (and its parts in the case of an aggregate), the Activities it performs, the processes (Activities) it is within (overlaps), and the Capabilities in provides. 2. Alias with function (Activity)
Phasing/Evolution/Forecast	Phase: A stage in a process of change or development. Evolution: Any process of formation or growth; development. Forecast: To predict a future condition or occurrence	before after relationships, temporal state, time period
Physical Asset	Covered by the Real Property and Materiel concepts	Real Property, Materiel
Process	A logical, systematic sequence of activities, triggered by an event, producing a meaningful output.	Activity, Process, Function, Job, Chore, Assignment.
Requirement	A singular documented need of what a particular product or service should be or do	Rule
Role	A set of similar or otherwise logically related activities, implying a set of skills or capabilities, to which a performer may be assigned.	Performer, Activity, and their overlap
System Function	A function that is performed by a system. Although commonly used to refer to the automation of activities, data transformation or information exchanges within IT systems, it also refers to the delivery of military capabilities.	Activity, Process, Function, Job, Chore, Assignment.
Task	A action, activity or undertaking enabling missions, activities or functions to be performed or accomplished.	Activity, Process, Function, Job, Chore, Assignment.
Unit	Any military element whose structure is prescribed by competent authority, such as a table of organization and equipment; specifically, part of an organization.	Organization
User	Any actor (as defined above) that invokes an automated performer.	Actor

A consideration with the use of aliases:

- Capabilities link to Measures (Metrics) through the Activities they entail and the Desired Effects sought.

2.6.2 Capability Data Capture Method

A method to capture Capability data is described in **Table 2.6.2-1**.

Table 2.6.2-1: Capability Data Capture Method Description

Methodology Description	Capture Data for Architectural Description of Capability
Definition:	Define a method by which capabilities can be defined and architected in a manner that enables them to support the major decision processes of the DoD. The Capability Method includes characteristics used to ensure proper definition of capabilities as well as a process by which architectural information relative to capabilities can be captured and structured to enable it to support the major decision processes of the Department (e.g., PPBE, PfM, and JCIDS).
Input:	• Enterprise/Component/Program Vision Documentation • Enterprise/Component/Program Strategy Documentation • Mission Statements • Directives • Objectives and Goals Documentation • Concept of Operations Documentation • Organization Needs • Compliance Requirements • Material Weaknesses • Strengths, Weaknesses, Opportunities and Threats (SWOT) analysis
Method:	This method is described in two sections. The first section describes the attributes of a well-defined capability as defined within the Business Mission Areas Business Transformation Guidance dated 6 July 2007. The second section describes steps that can be taken to architect a capability. The method described here is done so with the assumption that enterprise priorities have been identified and defined. The assumption is also made that the desired goals and objectives for the enterprise priority have been defined. **Attributes of a Well-Defined Capability** If a new capability is added or an existing capability is being updated, then it must be defined. Attributes of a well-defined capability include quality, focus, granularity and modularity. 1. Quality: A high quality capability is a modular representation of the activities, the conditions under which they are to be performed and the desired effects to be achieved. A high quality capability has minimal overlap with other capabilities. 2. Focus: Well-focused capabilities are both necessary and sufficient (as a group) to achieve the enterprise priority. 3. Granularity: Capabilities should be defined at a level of granularity that is: a. meaningful and consistent in an operations context b. appropriate for intended use by the stakeholders

Table 2.6.2-1: Capability Data Capture Method Description

Methodology Description	Capture Data for Architectural Description of Capability
	c. consistent with approved taxonomies to be used to help architecturally define the capabilities d. consistent with the DoD EA Reference Models to support federation e. defined according to an appropriate level of roles and responsibility such as: i. Governance: setting strategy, prioritizing enterprise efforts, assigning responsibilities and authorities, allocating resources, and communicating a shared vision. ii. Management: focusing on organizing tasks, people, relationships, and technology. iii. Work: Executing the strategy and plans established at a management level. 4. Modularity: Each capability should serve as a unit of transformation a. Cleanly identified with tiered implementation accountability assigned at the appropriate level (Enterprise, Component, Program). b. Developed using one or more solutions that encompass people, activities, and technology. c. Developed to be implementable via various transformation mechanisms such as the PPBE, PfM and Acquisition Processes. **Minimum steps for architecting capabilities:** 1. Define the capability or capability improvement. (The above items serve as guidelines for defining a capability or capability improvement). 2. Provide a name for the capability (Each capability should have a unique identifier). 3. Describe, as discretely as possible the anticipated beneficial outcome(s) in terms of efficiency, effectiveness, or improved responsiveness to warfighter needs, decision-maker requirements, or taxpayer interests. 4. Briefly describe the problems/needs/gaps that this capability or capability improvement addresses. 5. Derive from the enterprise priority a list of questions that this capability or capability improvement addresses. 6. Identify the enterprise priority objectives supported by the capability or capability improvement. 7. Identify activities, services, systems, initiatives that can or will provide the capability or improvement.
Primary Output:	Capabilities, goals, performance measures, milestones, related activities
Secondary Output:	Personnel, Services, Systems, Organizations that relate to the capability
Disciplines:	Structured analysis, activity modeling, functional decomposition

2.6.3 Use

Data for Capabilities are used to describe the capability; define acquisition and development requirements necessary to provide the required capability; facilitate understanding of capability execution; develop/update/improve doctrine and educational materials in support of capability execution; and to facilitate sharing and reuse of data.

The CV captures the enterprise goals associated with the overall vision for executing a specified course of action, or the ability to achieve a desired effect under specific standards and conditions through combinations of means and ways to perform a set of tasks. It provides a strategic context for the capabilities described by an architecture, and an accompanying high-level scope, more general than the scenario-based scope defined in an operational concept diagram. The models within the CV are high-level and describe capabilities using terminology which is easily understood by decision-makers and used for communicating a strategic vision regarding capability evolution.

Factors considered in a Capability Based Analysis are:

- Doctrine
- Organizations
- Training
- Materiel
- Leadership and Education
- Personnel
- Facilities

The following sections document how the Capability Data Group and DM2 support analysis of each of these factors.

2.6.3.1 Doctrine. In Joint Pub 1-02, Dictionary of Military and Associated Terms, **doctrine** is defined as **"**Fundamental principles by which the military forces or elements thereof guide their actions in support of national objectives. It is authoritative but requires judgment in application."

The concept of judgment required in application deals with decision making and cannot be precisely modeled except perhaps as rules affecting the applicability of other rules. The parts of doctrine that can be modeled are included in the capability data group as follows:

- Principles are modeled as Rules.
- Military forces and elements thereof are modeled as types and assemblies of Performers.
- Actions are modeled as Activities.

Thus, doctrine is contained in the specification of certain fundamental Rules, Activities, and Performers and the relationships among them. These relationships are:

- Each Performer must be of one or more Activities.
- Each Activity must be by one or more Performers.
- Each Rule may be a constraint on one or more Activities.
- Each Activity may be constrained by one or more Rules.

- Each Rule may be a constraint on one or more Performers.
- Each Performer may be constrained by one or more Rules.

Thus, since the DM2 contains the entities and relationships listed above it contains the necessary and sufficient set of entities and relationships to permit the modeling of doctrine and a separate data group for Doctrine is not required.

2.6.3.2 Organizations. An organization is a specific real-world assemblage of people and other resources organized for an ongoing purpose. DM2 models Organizations as a type of Performer.

Defining an Organization as an assemblage means that each Organization exhibits a whole/part relationship whereby each Organization may be an assembly of other Organizations and each Organization may also be a component of one or more other Organizations. The following DM2 relationships are involved in the capability based analysis of Organization where each Organization is a type of Performer:

- Each Capability must be the result of one or more Activities.
- Each Activity must be by one or more Performers, where each Performer must be a type of Organization, therefore, each Capability must be provided by one or more Organizations.
- Each Organization must be the Performer of one or more Activities.
- Each Rule may be a constraint on one or more Organizations.
- Each Organization may be constrained by one or more Rules.
- Each Rule may be a constraint on one or more Activities.
- Each Activity may be constrained by one or more Rules.

2.6.3.3 Training. Training is defined as an activity or set of Activities to increase the capacity of one or more performers to perform one or more tasks under specified conditions to specific standards:

- Each Performer may be either an Organization or a Person.
- Each Performer must be of one or more Activities.
- Each Activity must be performed under one or more Conditions.
- Each Activity must be completed to meet one or more Standards.
- Each Standard must be specified by one or more Measures.

2.6.3.4 Materiel. Materiel is a type of Performer and is tracked as an individual Materiel. Like Organization above, each Materiel exhibits a whole/part relationship whereby each Materiel may be an assembly of other Materiels and each Materiel may also be a component of one or more other Materiels.

The following DM2 relationships are involved in the capability based analysis of materiel where each Materiel is a type of Performer:

- Each Materiel must be assigned to one or more Organizations.
- Each Materiel must be used by one or more Persons, where each Person must be the member of only one Organization at any one time.
- Each Capability must be the result of one or more Activities.

- Each Activity must be by one or more Performers, where each Performer must be either an Organization or a Person using a Materiel.
- Each Materiel must be the Performer of one or more Activities.
- Each Rule may be a constraint on one or more Materiels.
- Each Materiel may be constrained by one or more Rules.
- Each Rule may be a constraint on one or more Activities.
- Each Activity may be constrained by one or more Rules

2.6.3.5 Leadership and Education. Joint Pub 1-02 does not define leadership. In the context of the DM2, leadership is defined as the ability to lead. Joint Pub 1-02 defines Military Education as the systematic instruction of individuals in subjects that will enhance their knowledge of the science and art of war. Thus, to a certain extent, leadership is a set of skills that can be taught as part of the science and art of war and a smaller set of skills that can be trained as Activities that must be performed under specified conditions to meet specified standards.

Leadership is about the judgment required in application of doctrine; it deals with decision making and cannot be precisely modeled except perhaps as rules affecting the applicability of other rules.

2.6.3.6 Personnel. Personnel refer to Persons. Each Person is a type of Performer.

The following DM2 relationships are involved in the capability based analysis of materiel where each Person is a type of Performer:

- Each Person must be assigned to only one Organization at any one time.
- Each Person may the user of one or more Materiels.
- Each Materiel must be used by one or more Persons.
- Each Capability must be the result of one or more Activities.
- Each Activity must be by one or more Performers, where each Performer must be either an Organization or a Person using a Materiel.
- Each Person must be the Performer of one or more Activities.
- Each Rule may be a constraint on one or more Persons.
- Each Person may be constrained by one or more Rules.
- Each Rule may be a constraint on one or more Persons.
- Each Activity may be constrained by one or more Rules.

2.6.3.7 Facilities. A Facility is defined as a real property entity consisting of underlying land and one or more of the following: a building, a structure (including linear structures), a utility system, or pavement. Please note that this definition requires that facilities be firmly sited on or beneath the surface of the earth. Things like tents, aircraft, and satellites that are not affixed to a single location on or beneath the surface of the earth are a type of Materiel. Materiel are germane to capability-based analysis through the following relationships:

- Each Facility or Materiel may be the site of one or more Performers.
- Each Performer may be at only one Facility or within a Materiel enclosure at any one time.
- Because a Facility is an Individual, it has a spatial and temporal extent.

- An Individual instance of Materiel has a spatial and temporal extent in contrast to a Type which does not. Generally Architectural Descriptions deal with Types of Materiel, not specific Individuals, e.g., not specific serial-numbered items of equipment. However, the DM2 does represent a Performer at a Location and, consequently, any Materiel that is part of the Performer would also be at the Location.

2.7 Services

The Services Data Group provides those data that support the definition and use of Services within the net-centric environment. Section 2.7.1 identifies and describes the data within the group; Section 2.7.2 provides an example method for collecting data on services; Section 2.7.3 provides illustrative uses of the data, and Section 2.7.4 provides presentation examples for using the Services-related data for presentation to/for management in decision-making.

2.7.1 Data

The DoDAF Meta-model for the data comprising services is shown in **Figure 2.7.1-1**. The figure may be hard to read in a hardcopy printout but a version at full-resolution which can be zoomed in is published in the online DoDAF V2.0 Meta-model Data Dictionary. Definitions for the model terms are in provided in **Table 2.7.1-1**. All of the Types and Individuals are founded on a formal ontology from which they inherit whole-part, super-subtype, before-after, and, in some cases, interface, patterns. Additionally, the Tuples inherit a places pattern. These are shown in the DoDAF V2.0 Meta-model Data Dictionary. Aliases and composite terms related to Services are shown in **Table 2.7.1-2**. Authoritative Source definitions, aliases, and rationale are also provided in the DoDAF V2.0 Meta-model Data Dictionary. Note that foundational classes are generally not shown on data group diagrams; this foundational material is in Section 2. This includes super-subtype, whole-part, temporal-whole-part, overlap, type-instance (member-of), and before-after patterns. Also not shown are the data structures for classification marking of architecture information at the whole and element (portion) levels using the IC-ISM. The schema for the IC-ISM is in Volume 3. Lastly, note that the size of the icons is not indicative of their importance; sizes are adjusted to reduce line crossings and bends to make the diagrams easier to understand.

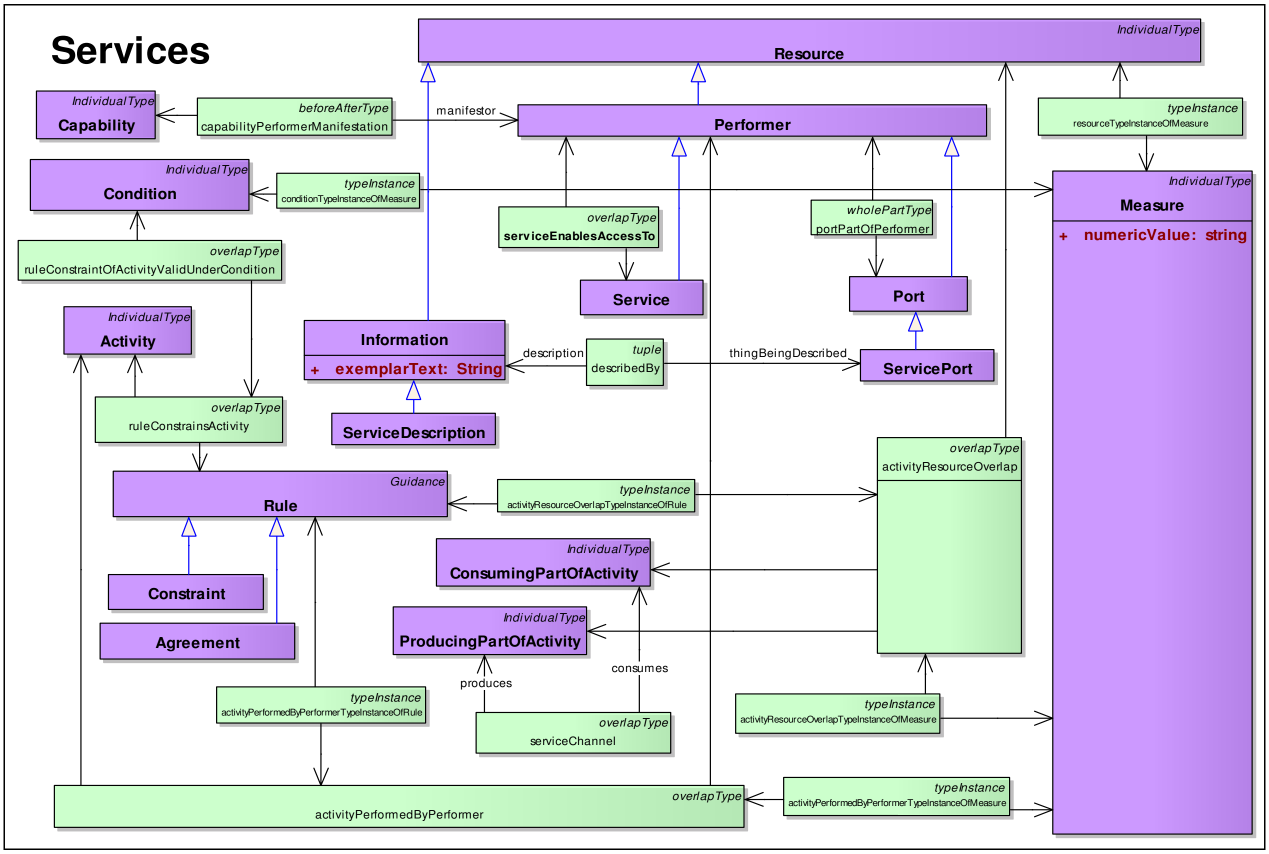


Figure 2.7.1-1: DoDAF Meta-model for Services

Note the following:

- Capabilities and Services are related in two ways. One, the realization or implementation of a Capability by a Performer (usually a configuration of Performers, including Locations) may include within the configuration Services (or Service compositions) to access other Performers within the overall Performer configuration. Conversely, the realization or implementation of a Capability by a Performer (configuration, including Location) may provide the Performers that are accessed by a Service (or Service composition).
- Unlike DoDAF V1.5, Services in DoDAF V2.0 include business services, such as Search and Rescue. This is important to keep in mind because much of the SOA literature is IT-oriented.
- Although, in principle, anything has a description, the importance of self-description for discovery and use of Services merits its call-out as a class. Further, because only a public-facing side is described, the Service description needs to represent that it describes the Service Port, not the entire Service. A Service Port is a special type of Port that is self-describing and visible. The Service Description of the Service Port is of all aspects necessary to utilize the Service and no more. As such, it may include visible functionality, QoS, interface descriptions, data descriptions, references to Standards or other Rules (Service Policy), etc. The inner workings of the Service are not described in a Service Description.
- Since Service inherits whole-part, temporal whole-part (and with it before-after), Service may refer to an orchestrated or choreographed Service, as well as individual Service components.

- Since Service Ports are types of Ports and Ports are types of Performers, they inherit all of Performer's properties, including Measures associated with the Performer, performance of Activities (Service Functions) with associated Measures, and provision of objects (Materiel, Data, Information, Performers, Geopolitical Extents).
- Any Performer that consumes a Service may have a Service Port that is described in the service request. This description indicates how the Service provider should provide or respond back to the Service consumer. That is, Service Ports are parts of Performers that may or may not be Services themselves.
- The Service Port is a special type of Port that is the part of a Performer that provides access to the Performer capabilities. Note that the Performer capabilities provided access to can be an aggregate, e.g., an orchestration, of Performer components. The Service Port is the service consumer facing part of the Service and so has a Service Description, a type of Information.

Table 2.7.1-1: DoDAF Meta-model Definitions for Services

Technical Term	Composite Definition	Potentially Related Terms or Aliases
	Classes	
Activity	Work, not specific to a single organization, weapon system or individual that transforms inputs (Resources) into outputs (Resources) or changes their state.	Action, Process Operational Activity, Processes, Function, System Function, Operation, Task, Plan, Project
Agreement	A consent among parties regarding the terms and conditions of activities that said parties participate in.	
Capability	The ability to achieve a Desired Effect under specified [performance] standards and conditions through combinations of ways and means [activities and resources] to perform a set of activities.	
Condition	The state of an environment or situation in which a Performer performs.	
Constraint	The range of permissible states for an object.	Business Rule, Rule, Restraint, Operational Limitation, Guidance
Measure	The magnitude of some attribute of an individual.	
Performer	Any entity - human, automated, or any aggregation of human and/or automated - that performs an activity and provides a capability.	Actor, Agent, Capability Configuration (MODAF)
Port	An interface (logical or physical) provided by a System.	

Table 2.7.1-1: DoDAF Meta-model Definitions for Services

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Resource	Data, Information, Performers, Materiel, or Personnel Types that are produced or consumed.	
Rule	A principle or condition that governs behavior; a prescribed guide for conduct or action	
Service	A mechanism to enable access to a set of one or more capabilities , where the access is provided using a prescribed interface and is exercised consistent with constraints and policies as specified by the service description. The mechanism is a Performer. The "capabilities" accessed are Resources -- Information, Data, Materiel, Performers, and Geo-political Extents.	
serviceChannel	A logical or physical communication path between requisitions and services.	
ServiceDescription	Information necessary to interact with the service in such terms as the service inputs, outputs, and associated semantics. The service description also conveys what is accomplished when the service is invoked and the conditions for using the service.	Service Interface Description (UPDM)
ServicePort	A part of a Performer that specifics a distinct interaction point through which the Performer intereacts with other Performers. This isolates dependencies between performers to particular interaction points rather than to the performer as a whole.	Mediator (OASIS SOA RA), Service Interface (UPDM)
Associations		
activityPerformedByPerformer	An overlap between a Performer and an Activity that is non-specific as to whether: 1. the Activity is solely performed by the Performer 2. the Activity is performed by several Performers 3. the Performer performs only this Activity 4. the Performer performs other Activities	

Table 2.7.1-1: DoDAF Meta-model Definitions for Services

Technical Term	Composite Definition	Potentially Related Terms or Aliases
activityPerformedByPerformerTypeInstanceOfMeasure	activityPerformerOverlap is a member of Measure	
activityPerformedByPerformerTypeInstanceOfRule	activityPerformerOverlap is a member of Rule	
activityResourceOverlap	An overlap of an Activity with a Resource, in particular a consuming or producing Activity that expresses an input, output, consumption, or production Activity of the Resource.	output, produce
activityResourceOverlapTypeInstanceOfMeasure	activityResourceOverlap is a member of Measure	
activityResourceOverlapTypeInstanceOfRule	activityResourceOverlap is a member of Rule	
capabilityPerformerManifestation	A couple that represents the capability that a performer manifests	
conditionTypeInstanceOfMeasure	Condition is a member of Measure	
ConsumingPartOfActivity	A part of an Activity that consumes a Resource	
portPartOfPerformer	A an association of the whole Performer to its Port that is visible and interfaces with other Performers	
ProducingPartOfActivity	A part of an Activity that produces a Resource	
resourceTypeInstanceOfMeasure	ResourceType is a member of Measure	
ruleConstrainsActivity	An overlap between a Rule and the Activities it allows	
ruleConstraintOfActivityValidUnderCondition	An overlap between the Activities constrained by a Rule and the Conditions under which the Rule applies	
serviceEnablesAccessTo	An overlap between the Service mechanism and the Performer capabilities it provides access to	

*The Black Text indicates a Concept in DM2, the Red Text indicates an Association in DM2.

Table 2.7.1-2: Aliases and Composite Terms Related to Services

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Actor	A performer that is external to and invokes the performer to be architected.	User, customer, agent, performer
Effect	The result, outcome, or consequence of an action.	A change in the state of a Resource as a result of some Activity. Goal, Objective, Desired Result, Outcome, Consequence, Effect Object
Function	The action for which a person or thing is specially designed, fitted, used or intended to accomplish or execute.	Activity, Process, Job, Chore, Assignment.
Performer Role	Any entity - human, automated, or any aggregation of human and/or automated - that performs a function, activity, or role, or provides a capability.	1. Composite of Performer (and its parts in the case of an aggregate), the Activities it performs, the processes (Activities) it is within (overlaps), and the Capabilities in provides. 2. Alias with function (Activity)
Process	A logical, systematic sequence of activities, triggered by an event, producing a meaningful output.	Activity, Process, Function, Job, Chore, Assignment.
Quality of Services	The ability to provide different priority to different applications, users, or data flows, or to guarantee a certain level of performance to a data flow.	activityPerformerOverlapTypeInstanceOfMeasure activityResourceOverlapTypeInstanceOfMeasure
Requirement	A singular documented need of what a particular product or service should be or do	Rule
Role	A set of similar or otherwise logically related activities, implying a set of skills or capabilities, to which a performer may be assigned.	Performer, Activity, and their overlap
Service Level Agreement	Part of a service contract where the level of service is formally defined	Agreement, Constraint
ServiceFunction	White box implementation of the Activities of the Service.	Activity known to be a Service Function when it is performed by a Service
ServicePolicy	An agreement governing one or more Services	Agreement, Constraint

Table 2.7.1-2: Aliases and Composite Terms Related to Services

Technical Term	Composite Definition	Potentially Related Terms or Aliases
SoA Service	A distinct part of the functionality that is provided by a technical system on one side of an interface to a general system on the other side of the interface (Derived from IEEE 1003.0). Characterized by transparency, autonomy, loose coupling, and discovery.	Composite Term
System Function	A function that is performed by a system. Although commonly used to refer to the automation of activities, data transformation or information exchanges within IT systems, it also refers to the delivery of military capabilities.	Activity, Process, Function, Job, Chore, Assignment.
Unit	Any military element whose structure is prescribed by competent authority, such as a table of organization and equipment; specifically, part of an organization.	Organization
Used In	Put into service	Description whole part
User	Any actor (as defined above) that invokes an automated performer.	Actor

2.7.2 Service Data Capture Method

A method to capture Services data is described in Table 2.7.2-1.

Table 2.7.2-1: Service Data Capture Method Description

Methodology Description	Capture Data for Architectural Description of Capability
Definition:	Per the DoDAF V2.0, a Service provides access to a capability through a prescribed interface and has certain constraints and policies applied to it. The following information describes a process by which data associated to a service can be captured to support development of an enterprise architecture.
Input[12]:	• Service Description • Service Policy • Performance Measures • Conditional Events
Method:	A Service is a type of Performer which means that it executes an activity and provides a capability. When analyzing the DM2, the information associated to a Service is very much akin to that related to a system. There is a description, interfaces and constraints that support its definition. The following steps can be taken to capture Services information to support the intended purpose of the architecture: • Identify and capture the capabilities supported or provided by the services. • Identify and capture the operations, business functions and activities supported or automated by the service. • Identify and capture the Organization responsible for providing the services. • Using the Service Description, capture the information to be consumed by the service and the information that is being produced by the service. • Define and capture the logical and/or physical interfaces required by the services. • Define and capture the rules applied to the information consumed and produced by the service. Also define and capture the rules governing or constraining the use of the service. • If not captured as part of the previously mentioned rules, define and capture the measures that will be used to gauge the performance of the service. • Identify and capture other services or systems on which the service is dependent or are dependent on the service.
Primary Output:	Traceability to: • Capabilities • Business functions • Activities Interface requirements, Input to Service Level Agreement Performance measures
Secondary Output:	Organization responsible for providing the service
Disciplines:	Structured analysis, Object-oriented Analysis (UML or SysML), Business Process Model (BPM)

[12] Inputs and Output sources and descriptions may be dependent upon the focus of the architecture efforts. For "To-Be" architectures, Inputs and Outputs may include resource flows between activities.

2.7.3 Use

A Service, in its broadest sense, is a well-defined way to provide a unit of work, through which a provider provides a useful result to a consumer. Services are activities done by a Service provider (Performer) to achieve desired results for a Service consumer (other Performer). Services do not necessarily equate to web-based technology or functions, although their use in the net-centric environment generally involves the use of web-based, or network-based, resources.

Functionally, *a Service is a set of strictly delineated functionalities, restricted to answering the what-question, independent of construction or implementation issues*[13]. Services form a layer, decoupling operational activities from organizational arrangements of resources, such as people and information systems. Finally, Services form a pool that can be orchestrated in support of operational activities, and the operational activities define the level of quality at which the Services are offered.

The Services Data Group described in Section 2.7.2 capture service requirements for supporting capabilities and operational activities, particularly the core processes (PPBE, DAS, JCIDS, SE, CPM, and Operations [Ops]). DoD processes include warfighting, business, intelligence, and Network Operations functions. The Services data are linkable to architecture artifacts in the Operational, Capability, and Project Viewpoints. Service functions (activities) and resources support operational requirements and facilitate the exchange of information among Performers.

2.8 Project

A Project is a temporary endeavor undertaken to create Resources of Desired Effects. Projects form the major elements of the DAS and are the primary focus of the DoD PPBE system.

The Primary Construct of the PPBE system is the Program Element (PE). The PE is defined as:

> ***Program Element***: *The program element is the basic building block of the Future Years Defense Program. The PE describes the program mission and identifies the organization responsible to perform the mission. A PE may consist of forces, manpower, materiel (both real and personal property), services, and associated costs, as applicable.*
>
> *(MIL-HDBK-881A, 30 July 2005)*

The key architectural construct within Project and the Program Element is the Work Breakdown Structure (WBS) subject to DoD Instruction 5000.2. The WBS is the primary instrument connecting an Architectural Description to the Defense Acquisitions System and the PPBE processes. The Work Breakdown Structure (WBS) is defined as:

> ***Work Breakdown Structure***: "*A product-oriented family tree composed of hardware, software, services, data, and facilities. The family tree results from systems engineering efforts during the acquisition of a defense materiel item". (MIL-HDBK-881A, 30 July 2005)*

MIL-HDBK-881A provides guidance for constructing the WBS applicable to programs subject to DoD Instruction 5000.2. The WBS is the process necessary for subdividing the major product deliverables and project work into smaller more manageable components and it serves as a

[13] North Atlantic Treaty Organization (NATO), NATO Architecture Framework, v.3, DRAFT 0.9 14 July 2006.

valuable framework for the technical objectives, and therefore it is product-oriented. Its elements should represent identifiable work products, whether they are equipment, data, or related service products. A WBS is a product structure, not an organizational structure, providing the complete definition of the work to be performed by all participants and the required interfaces between them.

Hardware, software, services, data, and facilities are Resources in the DM2. The information captured in the project administrative tool/techniques (e.g., Project Management Body of Knowledge [PMBOK] 2004) provides the basis for resource information in the DM2. The WBS forms the basis of reporting structures used for contracts requiring compliance with ANSI/EIA 748 Earned Value Management System (EVMS) Guidelines and reports placed on contract such as Contractor Cost Data Reporting (CCDR), Software Resource Data Report (SRDR), Contract Performance Reports (CPR), and Contract Funds Status Reports (CFSR).

MIL-HDBK-881A states: "*…the Program WBS and Contract WBS help document architectural products in a system life cycle. The DoD Architecture Framework (DoDAF) V1.0 defines a common approach for DoD Architecture Description development, presentation, and integration for warfighting operations and business operations and processes*."

Just as the system is defined and developed throughout its lifecycle, so is the WBS. In the early Project phases of concept refinement, system architecture, and technology development, the program WBS is usually in an early stage of development. The results of the Analysis of Material Approaches and the Analysis of Alternatives (AoA) provide the basis for the evolution of the WBS at all stages of Project evolution. As the architectural design of the project's product or service matures, so should the WBS. The WBS is a primary tool in maintaining efficient and cost effective developments of products and services. Figure 2.8-1 illustrates the evolution of the WBS during the lifecycle of Project.

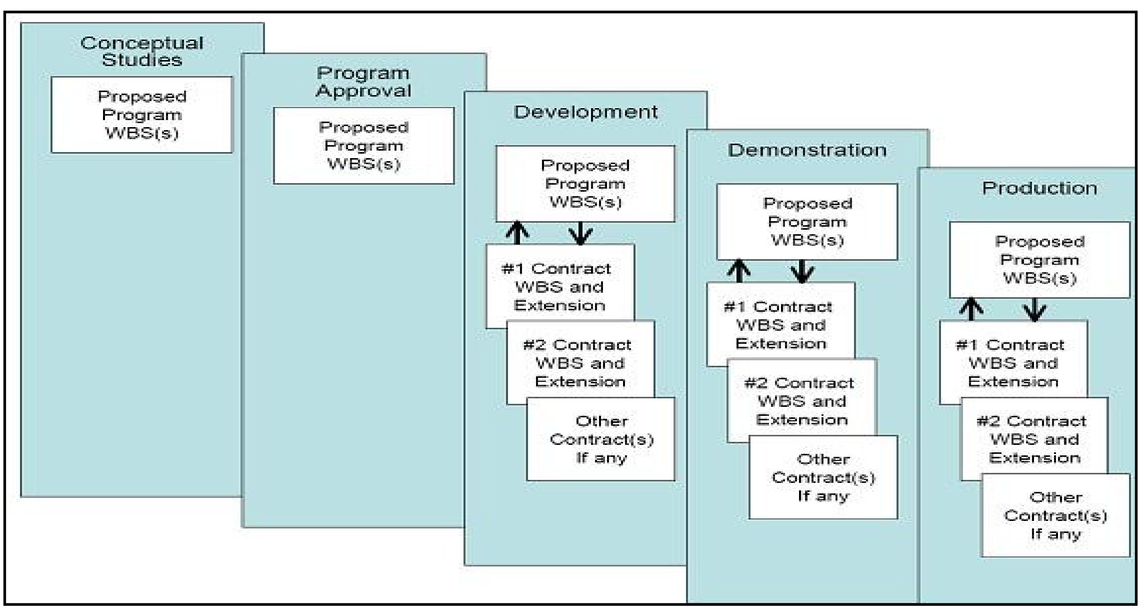


Figure 2.8-1: Evolution of the Project WBS

The following sections describe the DoDAF V2.0 Meta-model elements of Activities, Performers, Organizations, Objectives, Constraints, etc., that form the essential elements of the

WBS Project definition and how their ontological and taxonomic structures are derived from Architectural Description.

It should be noted that the same ontological and taxonomic structures also directly apply and should be traceable to architecture and classical specifications, such as the Statement of Objectives (SOO), and the SOW.

2.8.1 Data

The DoDAF Meta-model for the data comprising Project is shown in **Figure 2.8.1-1**. The figure may be hard to read in a hardcopy printout but a version at full-resolution which can be zoomed in is published in the online DoDAF V2.0 Meta-model Data Dictionary. Definitions for the model terms are in **Table 2.8.1-1**. It is important to be aware that all of the Types and Individuals are founded on a formal ontology from which they inherit whole-part, super-subtype, before-after, and, in some cases, interface, patterns. Additionally, the Tuples inherit a places pattern. These are shown in the DoDAF V2.0 Meta-model Data Dictionary. Aliases and composite terms related to Projects are shown in **Table 2.8.1-2**. Authoritative Source definitions, aliases, and rationale are provided in the DoDAF V2.0 Meta-model Data Dictionary. Note that foundational classes are generally not shown on data group diagrams; this foundational material is in Section 2. This includes super-subtype, whole-part, temporal-whole-part, overlap, type-instance (member-of), and before-after patterns. Also not shown are the data structures for classification marking of architecture information at the whole and element (portion) levels using the IC-ISM. The schema for the IC-ISM is in Volume 3. Lastly, note that the size of the icons is not indicative of their importance; sizes are adjusted to reduce line crossings and bends to make the diagrams easier to understand.

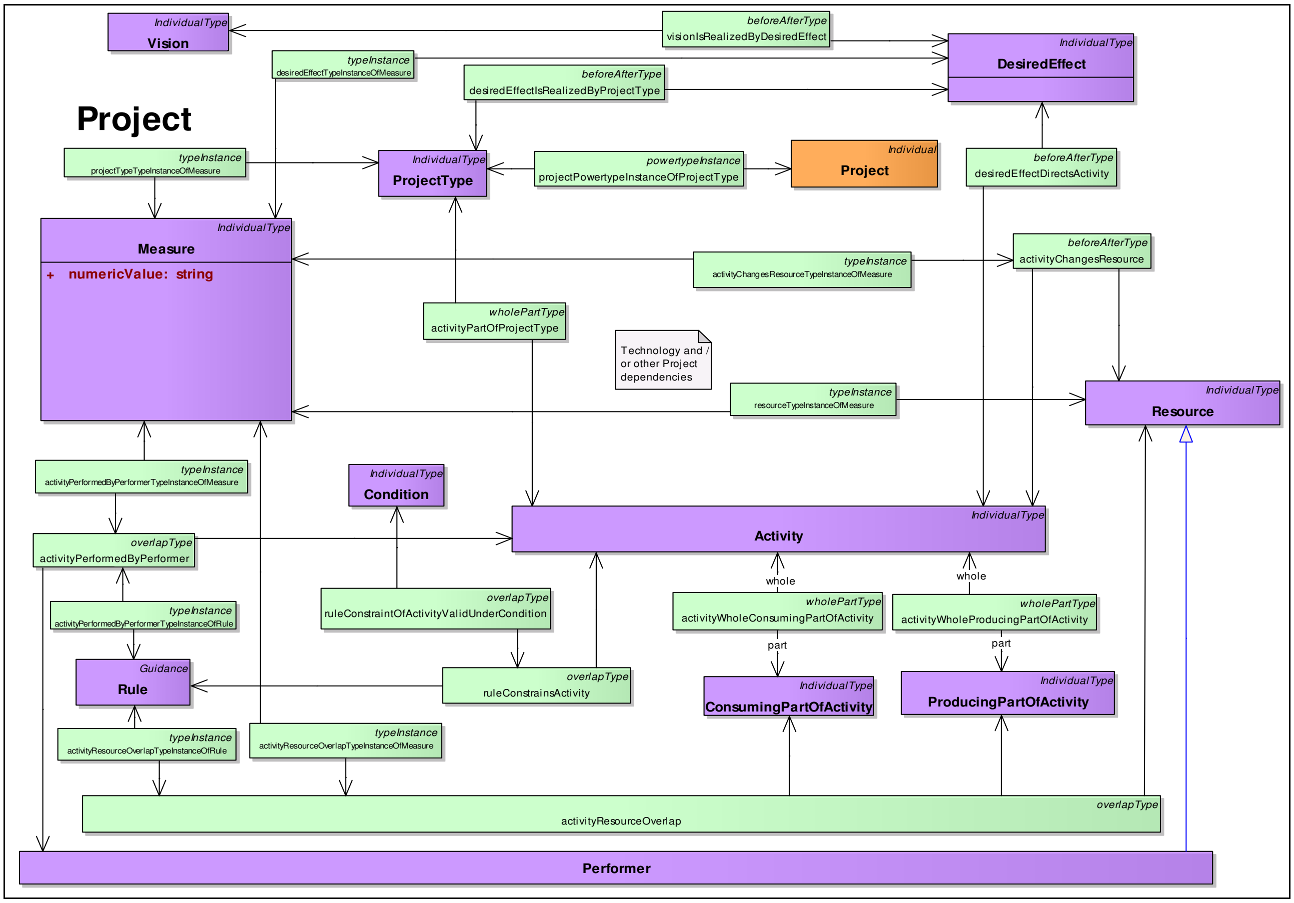


Figure 2.8.1-1: DoDAF Meta-model for Project

Table 2.8.1-1: DoDAF Meta-model Definitions for Project

Technical Term	Composite Definition	Potentially Related Terms or Aliases
	Classes	
Activity	Work, not specific to a single organization, weapon system or individual that transforms inputs (Resources) into outputs (Resources) or changes their state.	Action, Process Operational Activity, Processes, Function, System Function, Operation, Task, Plan, Project
Condition	The state of an environment or situation in which a Performer performs.	
DesiredEffect	The result, outcome, or consequence of an action [activity].	DesiredEffectType IndividualDesired Effect

Table 2.8.1-1: DoDAF Meta-model Definitions for Project

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Measure	The magnitude of some attribute of an individual.	
Performer	Any entity - human, automated, or any aggregation of human and/or automated - that performs an activity and provides a capability.	Actor, Agent, Capability Configuration (MODAF)
Project	A temporary endeavor undertaken to create Resources or Desired Effects.	Plan, Tactic, Strategy, Activity
ProjectType	The powertype of Project	
Resource	Data, Information, Performers, Materiel, or Personnel Types that are produced or consumed.	
Rule	A principle or condition that governs behavior; a prescribed guide for conduct or action	
Vision	An end that describes the future state of the enterprise, without regard to how it is to be achieved; a mental image of what the future will or could be like	
Associations		
activityChangesResource	Represents that an activity was / is / will-be the cause of change in the effected object with a before-after relationship.	
activityChangesResourceTypeInstanceOfMeasure	activityChangesResource is a member of Measure	
activityPartOfProjectType	A wholePart relationship between a Project and an Activity (Task) that is part of the Project	
activityPerformedByPerformer	An overlap between a Performer and an Activity that is non-specific as to whether: 1. the Activity is solely performed by the Performer 2. the Activity is performed by several Performers 3. the Performer performs only this Activity 4. the Performer performs other Activities	
activityPerformedByPerformerTypeInstanceOfMeasure	activityPerformerOverlap is a member of Measure	
activityPerformedByPerformerTypeInstanceOfRule	activityPerformerOverlap is a member of Rule	

Table 2.8.1-1: DoDAF Meta-model Definitions for Project

Technical Term	Composite Definition	Potentially Related Terms or Aliases
activityResourceOverlap	An overlap of an Activity with a Resource, in particular a consuming or producing Activity that expresses an input, output, consumption, or production Activity of the Resource.	output, produce
activityResourceOverlapTypeInstanceOfMeasure	activityResourceOverlap is a member of Measure	
activityResourceOverlapTypeInstanceOfRule	activityResourceOverlap is a member of Rule	
activityWholeConsumingPartOfActivity	A whole - part association between an Activity and the part of it that consumes a Resource.	input, consume
activityWholeProducingPartOfActivity	A whole - part association between an Activity and the part of it that produces a Resource.	proceeds, succeeds
ConsumingPartOfActivity	A part of an Activity that consumes a Resource	
desiredEffectDirectsActivity	The couple that represents how a desired effect directs an activity	
desiredEffectIsRealizedByProjectType	The couple that represents how a desired effect is realized by a project type	
desiredEffectTypeInstanceOfMeasure	DesiredEffect is a member of Measure	
ProducingPartOfActivity	A part of an Activity that produces a Resource	
projectPowertypeInstanceOfProjectType	Project is a member of ProjectType	
projectTypeTypeInstanceOfMeasure	ProjectType is a member of Measure	
resourceTypeInstanceOfMeasure	ResourceType is a member of Measure	
ruleConstrainsActivity	An overlap between a Rule and the Activities it allows	
ruleConstraintOfActivityValidUnderCondition	An overlap between the Activities constrained by a Rule and the Conditions under which the Rule applies	
visionIsRealizedByDesiredEffect	The relationship that exists between a vision and the specific desired effect that realised it	

*The Black Text indicates a Concept in DM2, the Red Text indicates an Association in DM2.

Table 2.8.1-2: Aliases and Composite Terms Related to Projects

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Capability Increment	A capability that can be effectively developed, produced, acquired, deployed and sustained.	composite of Capability temporal part (with time period) - Performer (and its time period)
Course of Action	A path towards a goal	Mission, strategy, plan
Desired Result	The wished for result, outcome, or consequence of an action. A desired result may be either a goal or an objective.	desired effect, desired outcome, desired consequence
Event	Something that happens at an instant in the world, i.e., a zero-duration process (Activity).	Milestone, Trigger, Activity
Goal	A desired change in the state of a Effect Object as a result of some activity.	Desired Result, Effect, Outcome, Consequence
Milestone	Something that happens at an instant in the world, i.e., a zero-duration process (Activity).	Activity, Event
Objective	A clearly defined, decisive, and attainable end toward which every operation is directed. An objective is a specific, time-targeted, measurable, and attainable target that an enterprise seeks to meet in order to achieve its goals.	Desired Result, Effect, Outcome, Consequence
Phasing/Evolution/Forecast	Phase: A stage in a process of change or development. Evolution: Any process of formation or growth; development. Forecast: To predict a future condition or occurrence	before after relationships, temporal state, time period
Plan	A set of Activities that result in a Goal, Desired Effect, outcome, or objective.	Course of Action, Activity aggregate (temporal or otherwise)
Program	A directed funded effort that provides a new, improved, or continuing materiel, weapon or information system or service capability in response to an approved need.	Project
Requirement	A singular documented need of what a particular product or service should be or do	Rule
Schedule Dependency	Schedule dependencies deal with Resources that an Activity requires in order to proceed.	Before after relationships between Activities and Resources

Table 2.8.1-2: Aliases and Composite Terms Related to Projects

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Technology	The application of science to meet one or more objectives.	type of Project
Unit	Any military element whose structure is prescribed by competent authority, such as a table of organization and equipment; specifically, part of an organization.	Organization

The DoDAF Meta-model contains the essential data required for defining a Project. Projects are defined in a Project Plan and supported by a System Engineering Plan. The Project Plan contains the project WBS (including Tasks and responsible Organizations). The Systems Engineering Plan (SEP) identifies the DoDAF-described Models to be produced and it defines the Project adoption and extensions (e.g., standard super-subtypes, whole-parts, and other architecture and engineering conventions) of DoDAF elements required by the specific Project. Further, the plans should define the project's primary areas of concern, as represented by Vision, Goals, and Objectives (VGOs). The VGOs should be directly traceable to the ICD, Capstone Requirements Document (CRD), Key Performance Parameter (KPP), and Capability Production Document (CPD) required by the JCIDS process. These VGOs should then be translated (e.g., requirements derived from the VGOs), to the Activities, Performers, Rules, and Measures in the Project. The Tasks and Performers form the essential elements of the project's WBS. The use of both Tasks and Performers focusing on products to be delivered (e.g., System, Service, Function, Component, etc.) in the WBS is the essential premise of the product-oriented WBS defined in MIL-HDBK-881A. Measures and Rules can be assigned at all levels of the Project decomposition. Top-level Measures and Rules (Conditions and Constraints) should be assigned to the VGOs. Lower-level Measures and Rules can then be derived and assigned to compliance and test criteria. When part of a legal contract, policy, or directive, the DoDAF Meta-model element (e.g., Activities (System Functions or Service Functions), Measures, and Rules) constitute a principle portion of the requirements for the Project. Any element of the DoDAF Meta-model may constitute a requirement if it is invoked by policy, directive, formal agreement, or contract instrument. Table 2.8.1-3 contains examples of requirements and their relationship to the DoDAF Meta-model terminology.

There are several items to note regarding this model:

- Like all concepts in the DM2, Project has whole-part, temporal whole-part, and super-subtype relationships so that major Projects can have minor Projects within them, Projects can have time phases, and Projects can be categorized.
- Because a Project involves execution of a plan composed of Activities (Tasks), there is a flow of resources into the project's activities and a flow of products out of them, as described by the Resource Flow data group. So this model can describe a Project that results in a System, a Service, Personnel Types (i.e., Training), Organizations (i.e., organizational development), Materiel, or Locations (e.g., Facilities, Installations).

- Because technology is part of the Project, this group models the analog of the DoDAF V 1.0 and V1.5 SV-9 (System and Services Technology Forecast) and SV-8 (System and Services Evolution Description).
- Many kinds of measures may be associated with a Project – needs, satisfaction, performance, interoperability, organizational, and cost.

Table 2.8.1-3: Requirements Related to the DoDAF Meta-model

Types of Requirements	
Requirement Type	**Criterion**
State/Mode	States the required states and/or modes of the item, or the required transition between one state and another state, one mode and another mode, made in one state to mode in another state. A state is a condition of something. A mode is a related group of functionality for a purpose.
Functional (Activity, Process, Performer)	States what the item is to do.
Performance (Measures and Rules)	For a given function, states how well that function is to be performed.
External Interface (Derived from Resource Flow)	States the required characteristics at a point or region of connection of the item to the outside world (e.g., location, geometry, inputs and outputs by name and specification, allocation of signals to pins, etc).
Environmental (Conditions and Constraints)	Limits the effect that the external environment (natural or induced) is to have on the item, and the effect that the item is to have on the external environment.
Resource (Conditions and Constraints)	Limits the usage or consumption by the item of an externally provided resource.
Physical (Conditions and Constraints)	States the required physical characteristics of the item as a whole (e.g., mass, dimension, volume).
Other Quality	States any other required quality that is not one of the above types, nor is a design requirement.
Design	Directs the design (internals), by inclusion (build it this way), or exclusion (don't build it this way).
Note: The same Types apply also to Visions, Goals, and Objectives	

2.8.2 Method

Methods for collecting and modeling Project data are as follows:

2.8.2.1 Project Modeling and Core Usage. The WBS is a system management tool very commonly used by program managers and industry. Created early in the life of a program, the WBS identifies deliverable work products (such as Products, Work Packages, Activities, Tasks, etc.). These work products are then further subdivided into successively smaller units until individual tasks can be assigned to people or organizations. This enables the responsibility to be assigned for individual tasks and provides traceability from low-level tasks to high-level work products.

Products and organizations are represented in the DoDAF V2.0 Meta-model as a taxonomic breakdown of the root architectural element Performer. These engineering decomposition methods are described in the Performer and Resource Flow sections of this volume. Figure 2.8.2.1-1 illustrates how taxonomic structure can be used to partition the Project into manageable subprojects, identify where common off-the-shelf-building blocks (Reuse) can be utilized, and identify all components of the system. In the acquisition stages, the level of breakdown of this decomposition is dependent on perspective (e.g., SoS, Enterprise, System, etc.) and acquisition strategy.

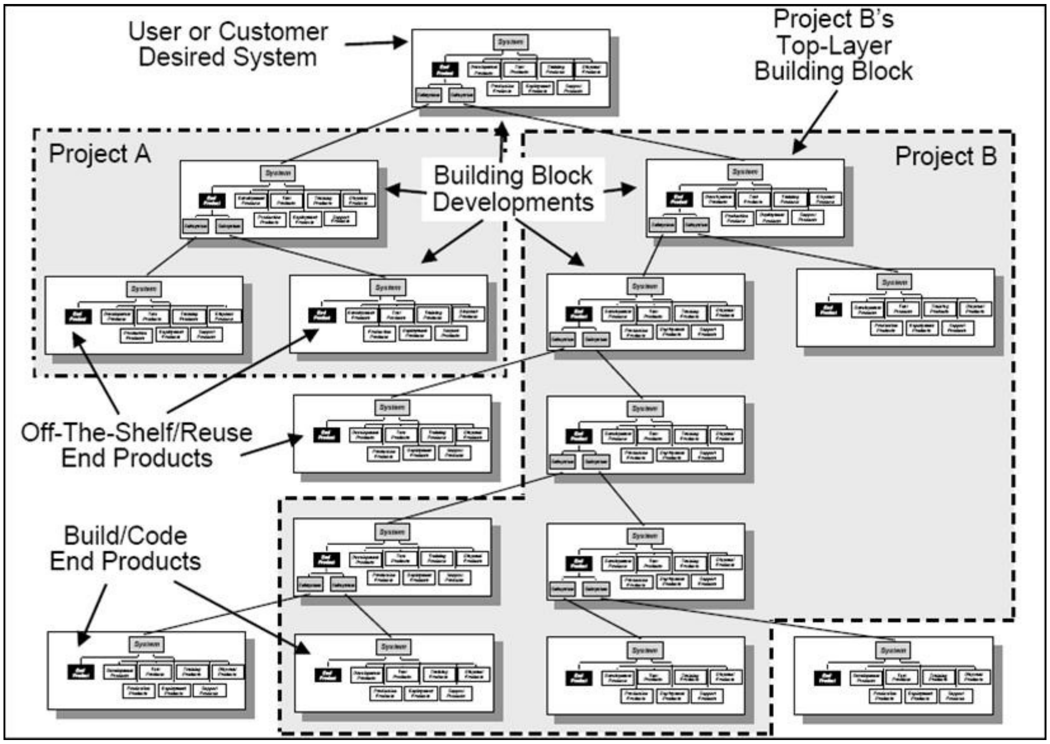


Figure 2.8.2.1-1: Non-prescriptive, Illustrative Example of System Taxonomy Used to Develop the Product Portion of the WBS

As stated in MIL-HDBK-881A, the WBS is a continually evolving instrument from Project conception to lifecycle management. This tracks closely with the evolution of the architecture. As key Activities are refined into primary Activities and assigned to or allocated to Performers, the WBS should mature and the project definition can gain additional focus. Early Project WBSs may contain high-level Activities (Tasks, Processes, System Functions, or Service Functions). As the Project matures, the WBS identifies the system components, such as subsystems and software configuration items (SCIs). The SCIs can be software services or individually testable and deliverable packages of software. Depending on the acquisition strategy, all or part of the Project WBS and, depending an acquisition strategy, may become the Contract WBS and form the basic outline of the requirements in a statement of work and the project Statement of Objectives (SOO) or Specification. Figure 2.8.2.1-2 illustrates this method.

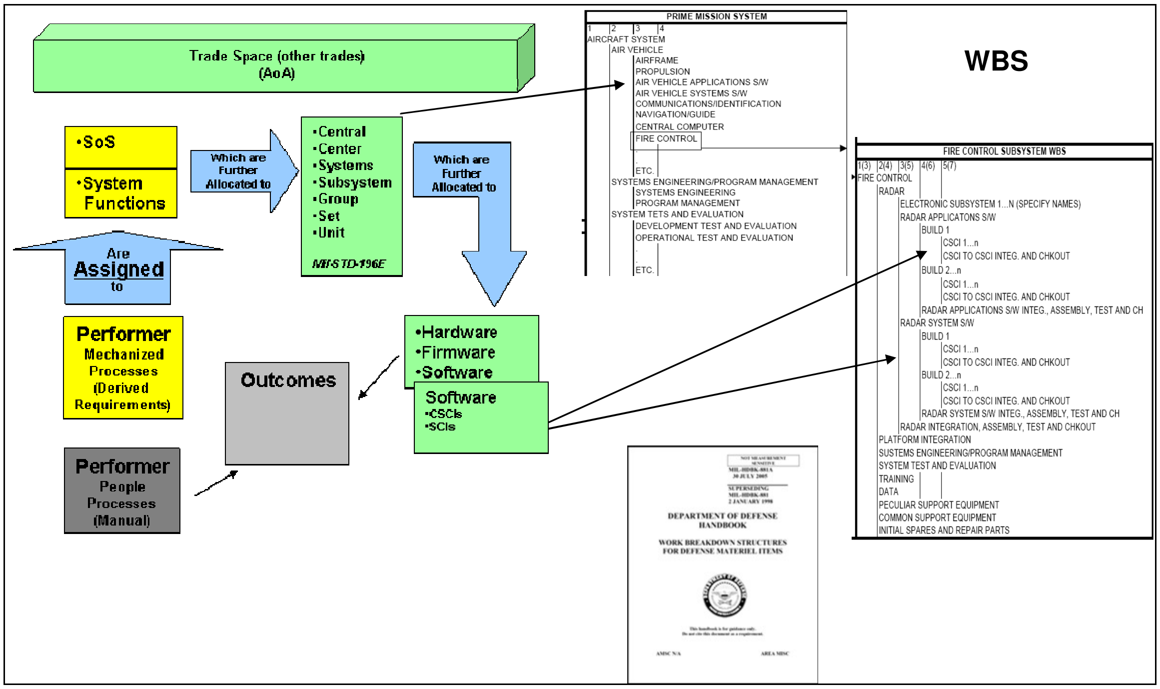


Figure 2.8.2.1-2: Derivation of the Materiel Portion of the WBS

The other, non-materiel portions of the WBS (Work Packages, Task and Activities) are derived in a similar fashion, i.e., Activities assigned to or allocated to Performers that are, in turn, assigned to Organizations, Personnel and Facilities.

2.8.2.2 Project Data Capture Method. A method to capture Project data is described in Table 2.8.2.2-1.

Table 2.8.2.2-1: Project Data Capture Method Description

Methodology Description	Capture Data for Architectural Description of Project
Definition:	Programs are accountable for implementing and managing their respective solutions to achieve priorities. Programs are responsible for reporting progress through performance measures that quantify and qualify achievement of program goals. (e.g., IRB reviews, Defense Business Systems Management Committee [DBSMC] reviews and critical milestones) within the acquisition management process are checkpoints to measure progress.
Input:	• Program Plan, System Engineering Plan, Specifications, etc. containing: – Captured to be Vision/Goals – Work Breakdown Structure – Performance Measures – Scope – Program Requirements – Conditional Events – Program Baseline
Method:	Plans and initiatives to coordinate transition planning in a documented program baseline, show critical success factors, milestones, measures, deliverables, and periodic program reviews.

Table 2.8.2.2-1: Project Data Capture Method Description

Methodology Description	Capture Data for Architectural Description of Project
	• There is a vision of the end result of the transformation that succinctly describes the changed conditions or environment. • Goals should be specific, detailed enough, and expressed in a way that DoD leadership can unambiguously assess whether and how it has been met. • Goals should have a focused, clearly defined scope that makes it possible to know when the capability has truly been achieved ensuring effectiveness. • A plan is then produced including activities with conditions and events that document the blueprint for desired outcomes and the roadmap for how to achieve those outcomes. • In this step, information from previous steps is leveraged to create or modify executable programs and begins the work to deliver improvements. Programs are defined through engagement in the existing requirements and acquisition management processes of the Department. • Transformation is then measured through performance measures that quantify and qualify achievement of program goals. The Execute and Evaluate step includes managing execution, transforming via implementation (testing and deployment) of designated programs, and evaluating and assessing progress using performance measures and other DoD process checkpoints.
Primary Output:	Refined Vision, Defined Goals, Scope, Program Effectiveness, Transition Plan, WBS
Secondary Output:	Personnel, Services, Systems, Organizations that relate to the capability
Disciplines:	Structured analysis, Performance Assessment

2.8.3 Use

Data for Projects are used in the following ways:

The data derived from Architectural Descriptions directly support the definition and structuring of Projects. The architectural data elements are used in the WBS, Architectural, and Classical Specifications and the SOW essential in the DAS. The architectural process augments classical System Engineering techniques by emphasizing the taxonomic structures (hierarchies) and ontological relationships (e.g., the federation with other needs, Systems, and Projects) between them. As shown in **Figure 2.8.3-1**, the Operational Viewpoint and DoDAF-described Models establish the needs typically used (depending on detail and purpose of the architecture) in defining the system requirements' baseline established at the Systems Requirements Review (SRR). Here the operational needs, as described in the Capabilities Description Document (CDD,) are translated into structured, engineerable requirements. Depending upon acquisition strategy, contracting may commence at this point, if assistance is required to establish Solution-related Viewpoints, DoDAF-described Models and associated baselines.

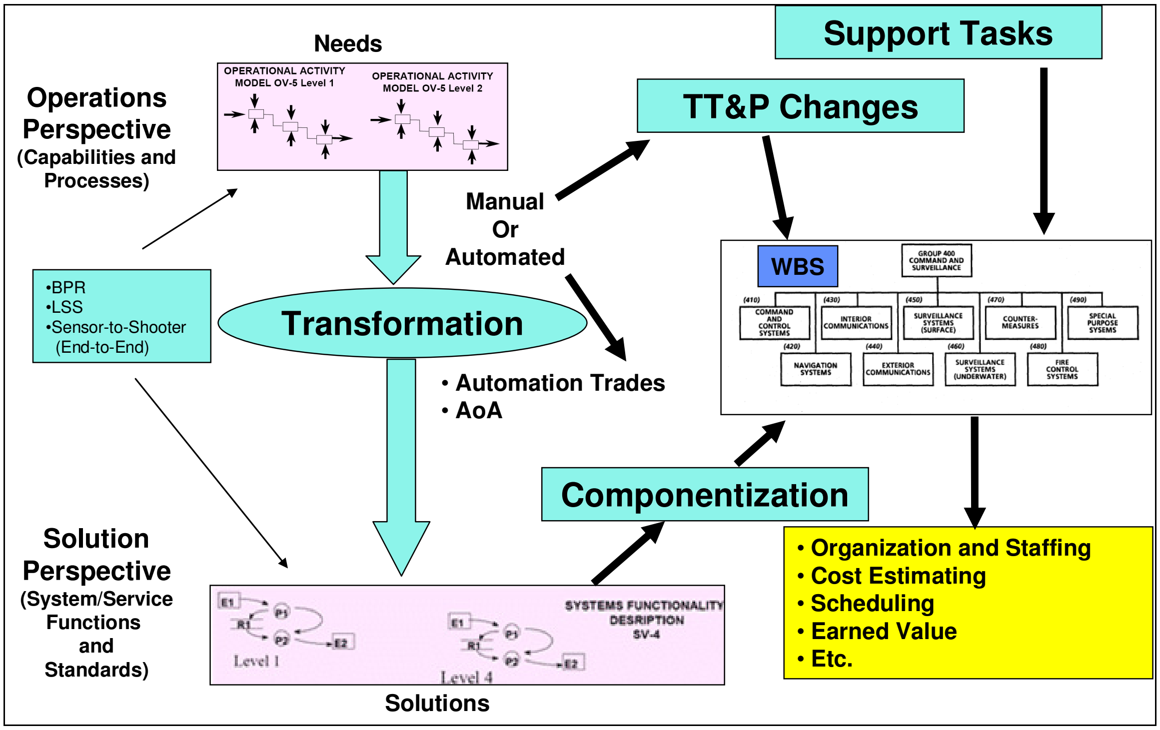


Figure 2.8.3-1: Architectural Description Usage in Forming Project Structure

Needs are transformed into Solutions through automation tradeoffs and AoA.

Various alternatives are iterated through the Operational Viewpoint and DoDAF-described Models to meet the required performance, cost, and schedule constraints. From here, Functional and Allocated baselines can be established. As increased detail is added to the architecture, classical systems engineering and design techniques are increasingly applied.

2.9 Goals

The Goals Data Group defines and describes the high-level data related to the establishment of goals, at some level, in the organization. Goals data are defined to represent the desired effect or outcome, or level of achievement, in operational processes, projects, or special programs. Goals data can be expressed as enterprise goals—high-level strategic goals that apply to the entire organization—or as more specific operational goals that define desired outcomes of the work process. Section 2.9.1 defines and describes goal-related data.

2.9.1 Data

The DoDAF Meta-model for the data comprising Goals are shown in Figure 2.9.1-1. The figure may be hard to read in a hardcopy printout but a version at full-resolution which can be zoomed in is published in the online DoDAF V2.0 Meta-model Data Dictionary. Definitions for the model terms are in Table 2.9.1-1. It is important to understand that all of the Types and Individuals are founded on a formal ontology from which they inherit whole-part, super-subtype, before-after, and, in some cases, interface, patterns. Additionally, the Tuples inherit a places pattern. These are shown in the DoDAF V2.0 Meta-model Data Dictionary. Aliases and composite terms are in Table 2.9.1-2. Authoritative Source definitions, aliases, and rationale are provided in the DoDAF V2.0 Meta-model Data Dictionary. Note that foundational classes are generally not shown on data group diagrams; this foundational material is in Section 2. This includes super-subtype, whole-part, temporal-whole-part, overlap, type-instance (member-of), and before-after patterns. Also not shown are the data structures for classification marking of architecture information at the whole and element (portion) levels using the IC-ISM. The schema for the IC-ISM is in Volume 3. Lastly, note that the size of the icons is not indicative of their importance; sizes are adjusted to reduce line crossings and bends to make the diagrams easier to understand.

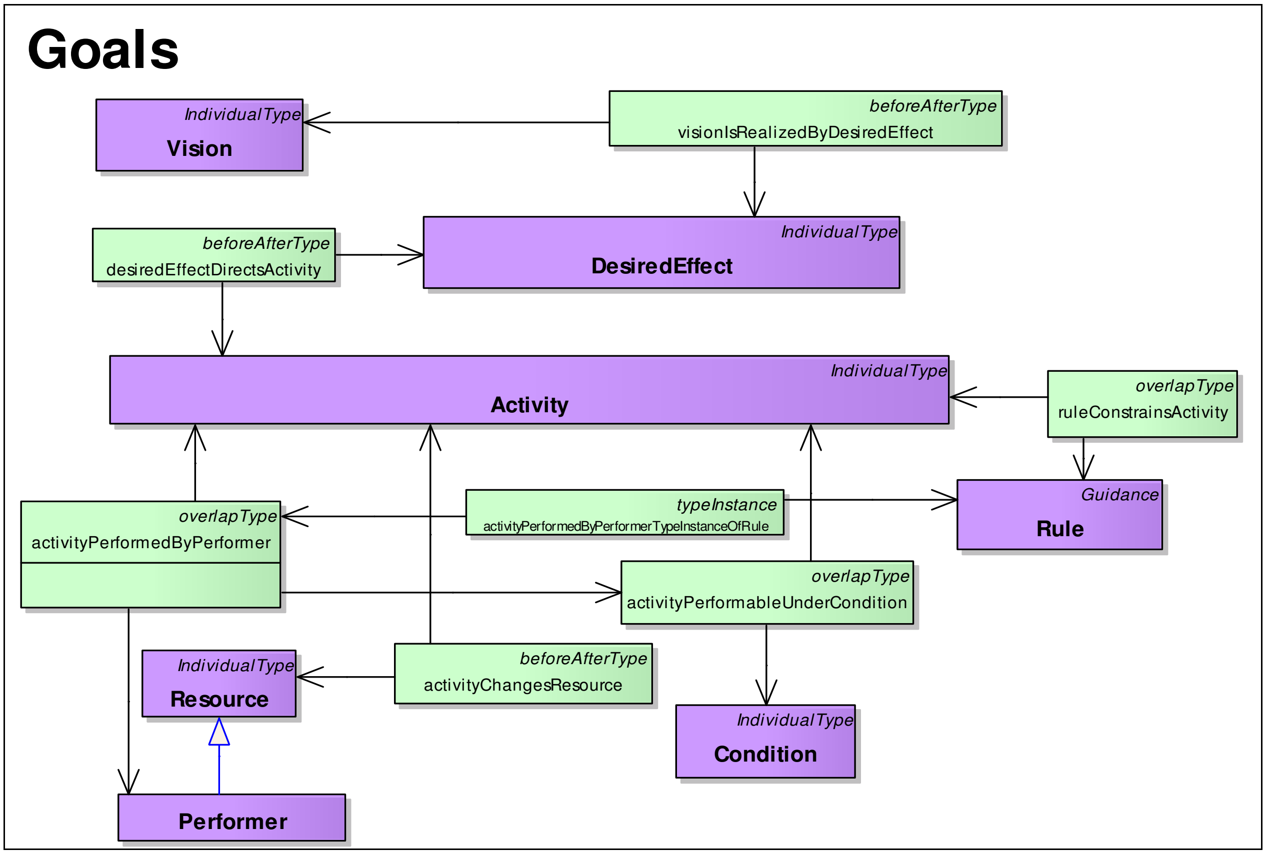


Figure 2.9.1-1: DoDAF Meta-model for Goals

The following should be noted about the Goals Data Group:

- Although the language sounds different, the meaning of a desired effect (a change in state of some object) is the same as the meaning of goal.
- A desired change in the state of some object leads to activities constrained by Rules that are performed by Performers. Some Activities are considered Events because they are of near-zero duration with respect to the frame of discernment of the Vision, Performers, etc.

Table 2.9.1-1: DoDAF Meta-model Definitions for Goals

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Classes		
Activity	Work, not specific to a single organization, weapon system or individual that transforms inputs (Resources) into outputs (Resources) or changes their state.	Action, Process Operational Activity, Processes, Function, System Function, Operation, Task, Plan, Project
Condition	The state of an environment or situation in which a Performer performs.	
DesiredEffect	The result, outcome, or consequence of an action [activity].	DesiredEffectType IndividualDesiredEffect
Performer	Any entity - human, automated, or any aggregation of human and/or automated - that performs an activity and provides a capability.	Actor, Agent, Capability Configuration (MODAF)
Resource	Data, Information, Performers, Materiel, or Personnel Types that are produced or consumed.	
Rule	A principle or condition that governs behavior; a prescribed guide for conduct or action	
Vision	An end that describes the future state of the enterprise, without regard to how it is to be achieved; a mental image of what the future will or could be like	
Associations		
activityChangesResource	Represents that an activity was / is / will-be the cause of change in the effected object with a before-after relationship.	
activityPerformableUnderCondition	Represents that an activity was / is / can-be/ must-be conducted under certain conditions with a spatiotemporal overlap of the activity with the condition.	
activityPerformedByPerformer	An overlap between a Performer and an Activity that is non-specific as to whether: 1. the Activity is solely performed by the Performer 2. the Activity is performed by several Performers 3. the Performer performs only this Activity 4. the Performer performs other Activities	

Table 2.9.1-1: DoDAF Meta-model Definitions for Goals

Technical Term	Composite Definition	Potentially Related Terms or Aliases
activityPerformedByPerformerTypeInstanceOfRule	activityPerformerOverlap is a member of Rule	
desiredEffectDirectsActivity	The couple that represents how a desired effect directs an activity	
ruleConstrainsActivity	An overlap between a Rule and the Activities it allows	
visionIsRealizedByDesiredEffect	The relationship that exists between a vision and the specific desired effect that realised it	

*The Black Text indicates a Concept in DM2, the Red Text indicates an Association in DM2.

Table 2.9.1-2: DoDAF Meta-model Aliases and Composite Terms Related to Goals

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Desired Result	The wished for result, outcome, or consequence of an action. A desired result may be either a goal or an objective.	desired effect, desired outcome, desired consequence
Effect	The result, outcome, or consequence of an action.	A change in the state of a Resource as a result of some Activity. Goal, Objective, Desired Result, Outcome, Consequence, Effect Object
End	an outcome worked toward especially with forethought, deliberate planning, and organized effort	effect, outcome, result
Event	Something that happens at an instant in the world, i.e., a zero-duration process (Activity).	Milestone, Trigger, Activity
Goal	A desired change in the state of a Effect Object as a result of some activity.	Desired Result, Effect, Outcome, Consequence
Mission	The task, together with the purpose [DesiredEffect], that clearly indicates the action [Activity] to be taken and the reason [DesiredEffect]; a duty [Activity] assigned to an individual [Personnel Type] or unit [Organization].	Task (=Activity) and DesiredEffect associated with it (them); Activity and ActivityPerformerOverlap where Performer = PersonnelType or Organization.

Table 2.9.1-2: DoDAF Meta-model Aliases and Composite Terms Related to Goals

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Objective	A clearly defined, decisive, and attainable end toward which every operation is directed. An objective is a specific, time-targeted, measurable, and attainable target that an enterprise seeks to meet in order to achieve its goals.	Desired Result, Effect, Outcome, Consequence
Outcome	An end result; a consequence.	desired effect, desired outcome, desired consequence
Requirement	A singular documented need of what a particular product or service should be or do	Rule
Strategy	A long-term plan to achieve pre-set goals	Plan, project
Tactic	A short-term action used to accomplish a strategy	Plan, project

2.9.2 Goals Data Capture Method

A method to capture Goals data is described in **Table 2.9.2-1**.

Table 2.9.2-1: Goals Data Capture Method Description

Methodology Description	Capture Data for Architectural Description of Goals
Definition:	A method or process by which architectural structural information relative to Goals can be captured to support the products used in the development of an architectural framework.
Input:	• DoD/Mission Area/Component Vision Documentation • DoD/Mission Area/Component Strategic Plan or other Strategic Documentation • DoD Directives • Operational Objectives • Organization Needs • Compliance Requirements • List of Performers (e.g., Roles, Services, Systems, Etc.)
Method:	Goals are used to help guide the Organizations to ensure that everyday operations are aligned to a strategic direction. The following information provides characteristics of well-defined goals. Well-defined goals should be relevant, attainable, timely and measurable. • Relevant means that it directly impacts the fulfillment of a Vision. • Attainable means that the Goal can be achieved given the available resources. • Timely means that the Goal must have a start and end time frame. • Measurable means that progress towards achieving the Goal can be quantified.

Table 2.9.2-1: Goals Data Capture Method Description

Methodology Description	Capture Data for Architectural Description of Goals
	The subsequent information describes steps that can be taken to properly architect goals that can be integrated within an architecture. • Reviewed the enterprise vision to determine desired effects and outcomes (i.e., Goals) that when accomplished will fulfill the Vision. Goals should be expressed in terms of information that is required to direct and manage the fulfillment of a Vision. • Identify and define a list of potential Goals to be reviewed with senior or executive level stakeholders for completeness and correctness. – Using the criteria stated in the previous section, answer the following questions: ▪ What makes this goal relevant? ▪ Is this attainable? ▪ Within timeframe do we desire to accomplish this goal? ▪ What are the measures that will be used to measure progress toward achieving this goal? – Any goal for which the above questions cannot be answered should be removed from the list of potential goals. • From the list of potential Goals, final Goals should be selected and vetted by senior or executive level stakeholders. – Identify any special rules that must be applied during the course of attaining the goal – Identify any special events or triggers that must be accounted for when accomplishing the goal. • An input list of Performers should be reviewed for candidates to be responsible for meeting each of the final Goals. • Performers should be assigned to each of the final Goals. One Performer should be assigned the responsibility to see that a Goal is accomplished. Other Performers may be assigned that have the authority or expertise to perform the any tasks that may be assigned. • The tasks to be performed in support of the goals can be defined as activities or functions. An input list of Activities or functions would be most beneficial and can be reviewed for candidates to be assigned to the Goals. • If the accomplishment of a Goal requires an Activity not in the input list, then a new Activity is appropriately added to the Activity list. • The progress of accomplishing a Goal is captured as an Effect.
Primary Output:	Well-defined Goals, Responsible Performers, Measures.
Secondary Output:	New or Modified Activities, Events and Rules.
Disciplines:	Structured analysis, business process re-engineering, business planning.

2.9.3 Use

Goals are established at all levels of the organization and can be applied to the Enterprise or Solution architecture effort. Goals are particularly useful to teams undertaking an architecture development effort to evaluate the success of the effort and how the effort contributes to achieving higher level goals, mission requirements, capability developments, or development of Services and Systems to support Department or organizational business operations.

Data for Goals are useful for:

- Scoping an activity to ensure that resources utilized in executing an activity contribute to the overall goals and vision of the organization.
- Establishing rules under which activities are executed to create boundaries for efficiency and effectiveness (Constraints) and establishing those circumstances under which an activity is executed (Event).
- Establishing measures and measures to determine the success of an activity, consistent with an established goal.

A goal is an end toward which long-term, ongoing effort is directed. In general, goals are established to provide a description of the intended future state. They are established to provide a target to aim toward, whereby activity is focused on attaining the desired effect (goal). Goals provide participants in activities a sense of direction, and a view of what to expect as activity progresses toward some end point.

Goals are often expressed in terms of Specific, Measurable, Attainable, Relevant, Timely (SMART) qualities, needed for a useful goal.

Specific Goals describe expected effects that are easily understood and capable of being executed. Measurable Goals can be tracked, evaluated against standards, and analyzed for their progress toward a desired objective. Attainable Goals are those that can be successfully achieved, assuming that the means and capabilities to achieve them are present in the organization. Relevant Goals are those goals that have meaning within the context of the project or activity. Timely refers to the established timeframe in which the goals are expected to be achieved, and the ability of the person or team to achieve the goals within that desired timeframe.

Within DoDAF, goals are identified and described to provide direction to Activities and to orient those Activities toward a desired effect. When a Performer executes an Activity, the Performer does so within the limitations prescribed for the Activity (Rules) and aims toward a desired effect. That effect should either meet, or contribute to meeting, established Goals that reflect the vision of the organization.

The key to success in using Goals data is the level of acceptance by other individuals or teams (performers) who will use the data in their efforts.

2.10 Rules

Rules are prescriptive sets of procedures regarding the execution of activities within an enterprise. Rules exist within the enterprise whether or not they are ever written down, talked about, or even part of an organization's consciousness. However, it is fairly common practice for organizations to gather rules in a formal manner for specific purposes.

Business rules are a type of Rule that govern actions and are initially discovered as part of a formal requirement-gathering process during the initial stages of a Project or during activity analysis, or event analysis. In this case, the collecting of the business rules is coincidental to the larger discovery process of determining the workflow of a process. Projects such as the launching of a new system or service that supports a new or changed business operation might lead to a new body of business rules for an organization that would require employees to conceptualize the purpose of the organization in a new way. This practice of coincidental business rule gathering is vulnerable to the creation of inconsistent or even conflicting business rules within different organizational units, or within the same organizational unit over time.

The DoDAF Meta-model provides a set of clear, concise data about rules, as described in Section 2.10.1, that facilitates the creation of rules and enables the sharing of those rules with others requiring similar information.

Creation of rules data must aim toward clear, easily understood, and totally unambiguous statements that define a procedure or function. Several best practices[14] can be adopted to assist in this effort. These are:

- ***The rule must be declarative***. A business rule is a statement of truth about an organization. It is an attempt to *describe* the operations of an organization. That is why business rules are said to be *discovered* or *observed* and not *created*. The prescription of a rule may occur in a future-based timeframe of an architecture, a "*To-Be*" architecture.
- ***The rule must be atomic***. A rule is either completely true or completely false; there are no shades of gray. For example, a rule for an airline that states *passengers may upgrade to first class round-trip tickets if seats are available and they pay the fare increase* does not imply that this deal is available for just one leg of the journey. In other words, conditions apply to rules and rules apply only to certain scope of activities.
- ***The rule must contain distinct, independent constructs***. Business rules should focus on definitions and should be separate from processes (i.e., strategies and tactics). Business Rules should not be complex and should avoid cyclical dependencies.
- ***The rule must be expressed in natural language***. To appeal to the broadest audience, it is almost always best to express business rules in a natural language without the use of a lot of technical jargon. There can be many business rules statements associated with a business rule. The business rule statement should conform to Object Management Group (OMG) specified Semantics of Business Vocabulary and Business Rules (SBVR)[15].
- ***The rule should be clearly understood by those outside the organization***. A company's business rules should not, for example, be foreign to a knowledgeable customer.

A rule is not a process - the two, while related, are very different. A *process* is a transformation that produces new things (outputs) from existing things (inputs), while a *rule* prescribes the exact procedures to be used to ensure that the output is as to be expected in each instance that the process is executed.

2.10.1 Data

The DoDAF Meta-model for the data comprising Rules is shown in **Figure 2.10.1-1**. The figure may be hard to read in a hardcopy printout but a version at full-resolution which can be zoomed in is published in the online DoDAF V2.0 Meta-model Data Dictionary. Definitions for the model terms are in **Table 2.10.1-1**. All the Types and Individuals are founded on a formal ontology from which they inherit whole-part, super-subtype, before-after, and, in some cases, interface, patterns. Additionally, the Tuples inherit a places pattern. These are shown in the

[14] *Business Rule Concepts*, Ron Ross, 2nd Ed, 2005
[15] http://www.omg.org/spec/SBVR/1.0/

DoDAF V2.0 Meta-model Data Dictionary. Aliases and composite terms are in Table 2.10.1-2. Authoritative Source definitions, aliases, and rationale are provided in the DoDAF V2.0 Meta-model Data Dictionary. Note that foundational classes are generally not shown on data group diagrams; this foundational material is in Section 2. This includes super-subtype, whole-part, temporal-whole-part, overlap, type-instance (member-of), and before-after patterns. Also not shown are the data structures for classification marking of architecture information at the whole and element (portion) levels using the IC-ISM. The schema for the IC-ISM is in Volume 3. Lastly, note that the size of the icons is not indicative of their importance; sizes are adjusted to reduce line crossings and bends to make the diagrams easier to understand.

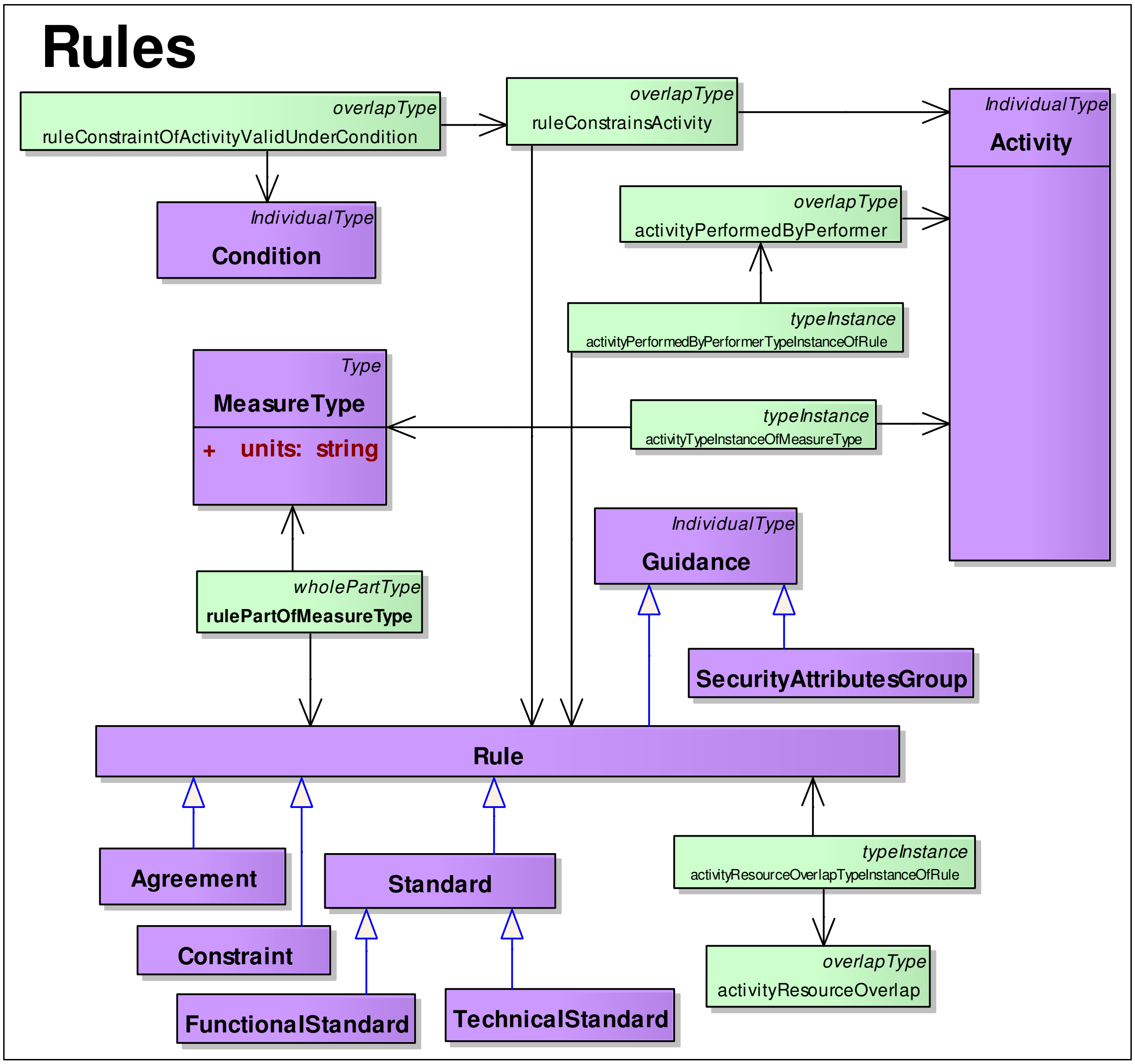


Figure 2.10.1-1: DoDAF Meta-model for Rules

The following should be noted about the Rules Data Group:

- A Rule constrains Activities. For example, a speed limit rule constrains driving activity. Some seemingly static rules have the effect of limiting possible activities. For example, a rule that security fences must be 10 feet high constrains the activity of building security fences. This constraint may apply or vary under certain conditions. For example, speed limits can be lower in poor weather conditions.

- Security classification, security marking, releasability, etc. are types of Guidance. Similarly; a Rule is a stronger form of Guidance.
- An important Constraint type is a Service Policy that constrains access to capability Performers.
- Doctrine, by definition, constrains military action.

Table 2.10.1-1: DoDAF Meta-model Definitions for Rules

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Classes		
Activity	Work, not specific to a single organization, weapon system or individual that transforms inputs (Resources) into outputs (Resources) or changes their state.	Action, Process Operational Activity, Processes, Function, System Function, Operation, Task, Plan, Project
Agreement	A consent among parties regarding the terms and conditions of activities that said parties participate in.	
Condition	The state of an environment or situation in which a Performer performs.	
Constraint	The range of permissible states for an object.	Business Rule, Rule, Restraint, Operational Limitation, Guidance
FunctionalStandard	Functional standards set forth rules, conditions, guidelines, and characteristics.	
Guidance	An authoritative statement intended to lead or steer the execution of actions.	
MeasureType	A category of Measures	
Rule	A principle or condition that governs behavior; a prescribed guide for conduct or action	
Standard	A formal agreement documenting generally accepted specifications or criteria for products, processes, procedures, policies, systems, and/or personnel.	
TechnicalStandard	Technical standards document specific technical methodologies and practices to design and implement.	

Table 2.10.1-1: DoDAF Meta-model Definitions for Rules

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Associations		
activityPerformedByPerformer	An overlap between a Performer and an Activity that is non-specific as to whether: 1. the Activity is solely performed by the Performer 2. the Activity is performed by several Performers 3. the Performer performs only this Activity 4. the Performer performs other Activities	
activityPerformedByPerformerTypeInstanceOfRule	activityPerformerOverlap is a member of Rule	
activityResourceOverlap	An overlap of an Activity with a Resource, in particular a consuming or producing Activity that expresses an input, output, consumption, or production Activity of the Resource.	output, produce
activityResourceOverlapTypeInstanceOfRule	activityResourceOverlap is a member of Rule	
activityTypeInstanceOfMeasureType	activityType is a member of MeasureType	
ruleConstrainsActivity	An overlap between a Rule and the Activities it allows	
ruleConstraintOfActivityValidUnderCondition	An overlap between the Activities constrained by a Rule and the Conditions under which the Rule applies	
rulePartOfMeasureType	A couple that represents the whole part relationship between types of measures and rules	
SecurityAttributesGroup	The group of Information Security Marking attributes in which the use of attributes 'classification' and 'ownerProducer' is required. This group is to be contrasted with group 'SecurityAttributesOptionGroup' in which use of those attributes is optional.	

*The Black Text indicates a Concept in DM2, the Red Text indicates an Association in DM2.

Table 2.10.1-2: DoDAF Meta-model Aliases and Composite Terms Related to Rules

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Directive	An authoritative statement intended to impel actions and the achievement of goals.	Guidance, Agreement
Means	An action or system by which a result is brought about; a method	Tactics, Strategy, Project, any DOTMLPF elements
Policy	A course of action, guiding principle, or procedure considered expedient, prudent, or advantageous	Rule
Requirement	A singular documented need of what a particular product or service should be or do	Rule
Service Level Agreement	Part of a service contract where the level of service is formally defined	Agreement, Constraint
ServicePolicy	An agreement governing one or more Services	Agreement, Constraint
Tactic	A short-term action used to accomplish a strategy	Plan, project
Technical Dependency	A Constraint on an Activity related to Performer(s) or Resource(s) needed.	Rule to Performer Resource - Performer overlap Resource consumed by Performer

2.10.2 Rule Data Capture Method

A method to capture Rules data is described in Table 2.10.2-1.

Table 2.10.2-1: Rule Data Capture Method Description

Methodology Description	Capture Data for Architectural Description of Rule
Definition:	A method or process by which architectural structural information relative to rules can be captured to support the products used in the development of an architectural framework.
Input:	• Rule description notation conventions. • The potential rule statement. • Rule classification, category or type. • The rule trigger or event, if appropriate. • The Activity object constrained by the rule.
Method:	• The input potential rule statement must be reviewed to determine whether the statement can be classified as a rule. Not all statements are rules. The classification, category or type of the input rule is identified as one of the following:

Table 2.10.2-1: Rule Data Capture Method Description

Methodology Description	Capture Data for Architectural Description of Rule
	- Agreement - Guidance - Constraint - Technical Standard - Functional Standard - Means • After the classification, category or type of Rule has been determined, the Activity to be constrained by the input potential rule statement is determined. A Rule must constrain an existing Activity in the architecture otherwise the rule is not required in the architecture. • The classification, category or type of Rule determines the allowable structure and notation of the Rule. A Rule that is a Functional Standard or Technical Standard should use a structured language and notation, be atomic and unambiguous, use a standard vocabulary and be directly enforceable. • A Rule that is not a Functional Standard or Technical Standard generally must be accepted without change because it was created by an entity out side of the architecture being developed; such as Congress. • The input rule statement description is restated, if necessary, to meet the approved Rule description notation conventions. (See Comment 3.) The guidelines for developing an architecture should contain a standard notation for writing Rules. • If required, the Rule trigger or event is evaluated and the Condition is determined. If an existing Condition does not exist, then a new Condition will have to be added. • The Rule is added to the architecture with the designated classification, category or type. • Based on the classification, category or type, the rule is associated with the appropriate Activity and Condition, if required. Functional Rules should be associated with functional or operational Activities and technical Rules should be associated with system Activities.
Primary Output:	A rule that is constructed using the notation standards, is properly classified, and is associated with the appropriate Activity or Activities.
Secondary Output:	Structured lists of Agreements, Guidance, Standards and Means that are the sources of the rules.
Disciplines:	Structured analysis and technical writing.

2.10.3 Use

Rules data are used to create, document, and share rules of all types that support operational activities and/or the execution of capabilities in operational processes (composite activities). These data can include:

- Processes that define transactions where data must be exchanged or passed to execute a specified activity, such as PPBE, CPM, JCIDS, or DAS.
- Rules that define methods of accessing information or services within the net-centric environment, such as Ops, PPBE, CPM, or JCIDS.
- The order of steps that occur in a series of actions that must be performed in a specific order, such as DAS, SE, PPBE, or CPM.

- Rules defining analysis of options or future actions, such as Ops Planning, JCIDS, PPBE or CPM.

2.11 Measures

A measure is the magnitude of some attribute of an object. Measures provide a way to compare objects, whether Projects, Services, Systems, Activities, or Capabilities. The comparisons can be between like objects at a point in time, or the same object over time. For example, a Capability may have different measures when looking at the current baseline and over increments toward some desired end-state.

Measures play a much greater, central role in DoDAF V2.0, compared to earlier versions of DoDAF. This change has multiple drivers, including:

- Core Process use of architectural data. Those management and engineering processes depend on quantification as a means of improving objectivity, accountability, and efficiency of the processes.
- Federal Enterprise Architecture (FEA) Performance Reference Model.

There are many kinds of Measures that are applicable to many architecture elements. These are described in the following paragraph.

2.11.1 Data

The DoDAF Meta-model for the data comprising Measures are depicted in **Figure 2.11.1-1**. The figure may be hard to read in a hardcopy printout but a version at full-resolution which can be zoomed in is published in the online DoDAF V2.0 Meta-model Data Dictionary. Definitions for the model terms are in **Table 2.11.1-1**. Aliases and composite terms related to Measures are shown in Table 2.11.1-2. All the Types and Individuals are founded on a formal ontology from which they inherit whole-part, super-subtype, before-after, and, in some cases, interface, patterns. Additionally, the Tuples inherit a places pattern. These are shown in the DoDAF V2.0 Meta-model Data Dictionary. There currently are no aliases or composite terms for Measures. Authoritative Source definitions, aliases, and rationale are provided in the DoDAF V2.0 Meta-model Data Dictionary. Note that foundational classes are generally not shown on data group diagrams; this foundational material is in Section 2. This includes super-subtype, whole-part, temporal-whole-part, overlap, type-instance (member-of), and before-after patterns. Also not shown are the data structures for classification marking of architecture information at the whole and element (portion) levels using the IC-ISM. The schema for the IC-ISM is in Volume 3. Lastly, note that the size of the icons is not indicative of their importance; sizes are adjusted to reduce line crossings and bends to make the diagrams easier to understand.

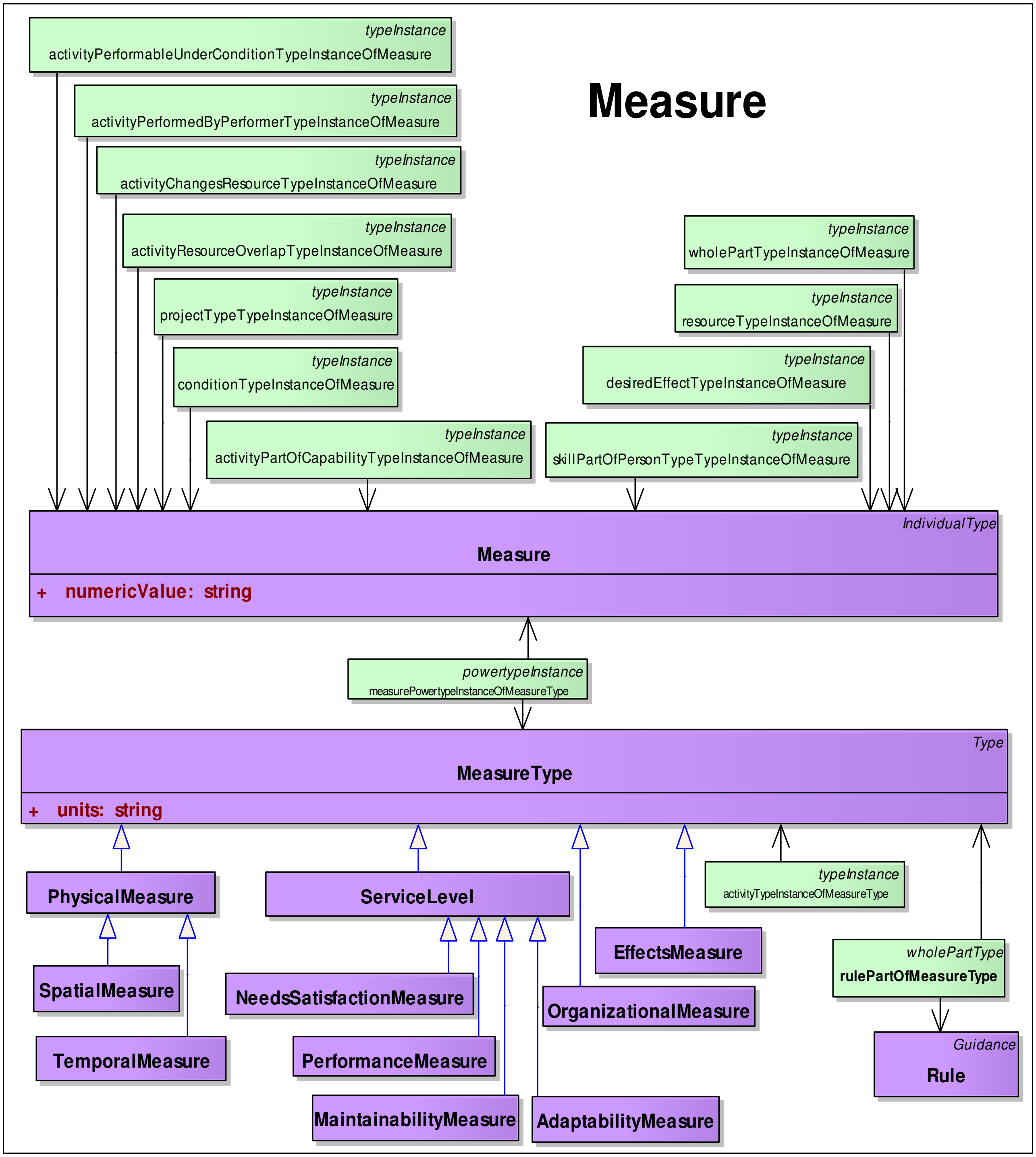


Figure 2.11.1-1: DoDAF Meta-model for Measures

Table 2.11.1-1: DoDAF Meta-model Definitions for Measures

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Classes		
AdaptabilityMeasure	A measure of the ease with which Performers satisfy differing Constraints and Capability and Service needs.	
EffectsMeasure	Category of measures on Effect Objects	
MaintainabilityMeasure	A category of measures of the amount of time a Performer is able to conduct Activities over some time interval.	
Measure	The magnitude of some attribute of an individual.	
MeasureType	A category of Measures	
NeedsSatisfactionMeasure	A category of quality measures that address how well a system meets the user's needs and requirements.	
Rule	A principle or condition that governs behavior; a prescribed guide for conduct or action	
Associations		
activityChangesResourceTypeInstanceOfMeasure	activityChangesResource is a member of Measure	
activityPartOfCapabilityTypeInstanceOfMeasure	activityPartOfCapability is a member of Measure	
activityPerformableUnderConditionTypeInstanceOfMeasure	activityPerformableUnderCondition is a member of Measure	
activityPerformedByPerformerTypeInstanceOfMeasure	activityPerformerOverlap is a member of Measure	
activityResourceOverlapTypeInstanceOfMeasure	activityResourceOverlap is a member of Measure	
activityTypeInstanceOfMeasureType	activityType is a member of MeasureType	
conditionTypeInstanceOfMeasure	Condition is a member of Measure	
desiredEffectTypeInstanceOfMeasure	DesiredEffect is a member of Measure	
measurePowertypeInstanceOfMeasureType	Measure is a member of MeasureType	

Table 2.11.1-1: DoDAF Meta-model Definitions for Measures

Technical Term	Composite Definition	Potentially Related Terms or Aliases
OrganizationalMeasure	A category of quality measures that address how costly a Performer is to operate and maintain.	
PerformanceMeasure	A category of quality measures that address how well a Performer meets Capability needs.	
PhysicalMeasure	A category of measures of spatio-temporal extent of an Individual such as length, mass, energy, velocity, …	
ServiceLevel	A measurement of the performance of a system or service.	
SpatialMeasure	A category of measures of the spatio-temporal location of an Individual.	
TemporalMeasure	A type of measure of time	
projectTypeTypeInstanceOfMeasure	ProjectType is a member of Measure	
resourceTypeInstanceOfMeasure	ResourceType is a member of Measure	
rulePartOfMeasureType	A couple that represents the whole part relationship between types of measures and rules	
skillPartOfPersonTypeTypeInstanceOf Measure	skillPartOfPersonType is a member of Measure	
wholePartTypeInstanceOfMeasure	wholePart is a member of Measure	

*The Black Text indicates a Concept in DM2, the Red Text indicates an Association in DM2.

Table 2.11.1-2: DoDAF Meta-model Aliases and Composite Terms Related to Measures

Technical Term	Composite Definition	Potentially Related Terms or Aliases
AccuracyPrecision	The nearness of a functional goal to the true value	Performance Measure
Capacity	The amount a Performer can hold, receive, or absorb.	Performance Measure
Cost	1. Cost - financial: The price paid to acquire, produce, accomplish, or maintain anything. 2. Cost – general: The expenditure of something, such as time or labor, necessary for the attainment of a goal.	Organizational Measure

Table 2.11.1-2: DoDAF Meta-model Aliases and Composite Terms Related to Measures

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Interoperability	A category of measures of the ability of two or more Performers to exchange Resources and to use the Resources that have been exchanged.	AdaptabilityMeasure
Quality of Services	The ability to provide different priority to different applications, users, or data flows, or to guarantee a certain level of performance to a data flow.	activityPerformerOverlap TypeInstanceOfMeasure activityResourceOverlap TypeInstanceOfMeasure
RateThroughput	The ratio of the effective or useful output to the total input in any system.	Performance Measure
Reliability	A category of measures of the ability of a Performer to perform its required Activities under stated conditions for a specified period of time.	
SecurityMeasure	A measure of the ability of a Performer to manage, protect, and distribute sensitive information.	Performance Measure
Timeliness	The time from the occurrence of an event to the time required action occurs.	Performance Measure
Trustworthiness	A category of measures of the degree to which a Performer avoids compromising, corrupting, or delaying sensitive information.	Performance Measure

The following should be noted about the Measures Data Group:

- The key elements of the Measure Data group are Measure and Measure Type. Measure refers to the actual measure value and units. It relates to a Measure Type that describes what is being measured. Examples of each are shown below in **Table 2.11.1-3**:

Table 2.11.1-3: Non-prescriptive, Illustrative Examples of Measures and Measure Types

Measure	Measure Type
1 year	Timeliness
Mach 3	Rate
99 percent	Reliability
56K	BAUD
3 meters	Target Location Error (TLE) Accuracy
1,000 liters	Capacity
$1M	Cost
Level 3	Capability Maturity Model® Integration (CMMI) Organizational Level

- Formally, a Measure defines membership criteria for a set or class; e.g., the set of all things that has 2 kg mass. The relationship between Measure and Measure Type is that any particular Measure is an instance of all the possible sets that could be taken for a Measure Type.
- The lower part of Figure 2.11.1-1 depicts the upper tiers of a Measure Type taxonomy or classification scheme. It is expected that architects would add more detailed types (make the taxonomy more specialized), as needed, within their federate. Note that Service Level has multiple inheritances because a Service QoS or Service Level Agreement (SLA) could address User Needs, User Satisfaction, Interoperability, or Performance.
- All Measure Types have a Rule that prescribes how the Measure is accomplished; e.g., units, calibration, or procedure. Spatial measures' Rules include coordinate system rules. For example, latitude and longitude are understandable only by reference to a Geodetic coordinate system around the Earth.
- As a Measure Type, cost can be captured in the architecture at different levels, based on the Process-owners requirements
- The upper part of Figure 2.11.1-1 depicts how Measures apply to architecture elements. Note that they apply to relationships between objects; e.g., the Measure applies to a Performer performing an Activity. The Activity has a relationship to Measure Type that says what Measure Types apply to an Activity. This represents Universal Joint Task List (UJTL) tasks and their applicable Measure Types, including Conditions, that is, Condition is quantified by a Measure Type. (The whole-part relationship feature of Condition allows it to be singular.) This is accomplished by Condition's typeInstance association, saying an elementary Condition is a member (instance) of a Measure Type class.

2.11.2 Measures Data Capture Method

A method to capture Measures data is described in **Table 2.11.2-1**.

Table 2.11.2-1: Measures Data Capture Method Description

Methodology Description	Capture Data for Architectural Description of Measures
Definition:	A method or process by which architectural information relative to Measures (or Metrics) can be captured to support the products used in the development of an architectural.
Input:	• Organization Transition Plan • Well-defined Capabilities • Activities or Functions linked to Capabilities • Organization Milestones • Concepts of Operations • Rules or Constraints
Method:	The DoDAF V2.0 has within its Meta-model several architectural constructs to which Measures should be associated. As a rule of thumb, any items against which performance must be measured or progress must be tracked should have Measures assigned to them to enable performance and progress to be gauged. Architectural constructs such as Capabilities, Activities (Functions, Processes, and Tasks), Performers (Persons, Systems, and Services) should have Measures assigned such that performance can be gauged.

Methodology Description	Capture Data for Architectural Description of Measures
	• The Measure must be associated with another object in the architecture including an Activity, Condition or Effect because a Measure defines the value and units of an object. A Measure not associated with another architectural object adds no value to the architecture. • After the associated object has been identified, the name, description, value and units of the Measure are determined. • The Measure Type is determined from the following subtypes: – Needs Satisfaction Measure – Performance Measure 1. Accuracy/Precision 2. Timeliness 3. Rate Throughput 4. Capacity 5. Dependability 6. Trustworthiness 7. Reliability 8. Security – Maintainability Measure – Adaptability Measure 1. Interoperability – Organizational Measure 1. Cost • The Measure is included in the architecture with the appropriate associations to other architectural objects.
Primary Output:	Measures or Metrics, Domain Values for the Measures or Metrics
Secondary Output:	None.
Disciplines:	Structured analysis.

2.11.3 Use

Data for Measures are used in the following ways:

- Planning – adequacy analysis. From an adequacy point of view, Measures that are associated with a Capability (including Capability increment, since Capabilities have whole-part inheritance). Capabilities can be compared with the Measures associated with the Performers to see if the Performer solution(s) are adequate. A set of alternative Performers as part of an Analysis of Alternatives could also be evaluated. Goals or Desired Effects could compare with Measures associated with Performers.
- Planning – overlap analysis. The purpose of an overlap analysis is to determine if there are overlaps, or undesired duplicative capability, in the spending plan, portfolio, capabilities development, or acquisition plan. Similar functionality is often only an indicator of overlapping or duplicative capability. Often Performers with similar functionality operate under different Measures which are not duplicative or overlapping capability. For example, operational-level situation awareness systems may not be as fast or precise as a tactical-level, but they may handle a larger number of objects over a larger area.

- System Engineering/Design. Measures set the design envelope goals, sometimes called performance characteristics or attributes. They can also set the constraints; e.g., cost constraints.
- Performance–Cost Tradeoffs. Measures of performance (e.g., effectiveness) can be compared to different costs to evaluate and make decisions about alternative solutions.
- Requirements. Requirements often have Measure elements.
- Benchmarking. Measures can be used to establish benchmarks of performance, such as for a personnel skill or a radar tracking accuracy test.
- Organizational and Personnel Development. Organizational and personnel goals are often established and then monitored using Measures.
- Capacity Planning. Measures can be used to plan for needed capacity; e.g., for networks, training programs.
- Portfolio Balancing. Measures can be used to balance a portfolio so that it achieves the right mix of goals and constraints.
- Capability Evolution. Measures are part of capability evolution, showing increments of measurable improvement as the capability evolves and allowing monitoring about when the capability is projected to be achieved or has already been achieved.
- Quality of Service (QoS) Description. In SOA, QoS is often expressed as a Measure; e.g., bit loss rate or jitter. These Measures show up in the service description and are part of service discovery, so users can discover access to capabilities that meet their quality requirements.
- Project Constraints. Measures such as cost and risk can be constraints on Projects.
- Goal Setting. Measures are often part of Goals; e.g., production or efficiency Goals.

2.12 Locations

A location is a point or extent in space. The need to specify or describe Locations occurs in some Architectural Descriptions when it is necessary to support decision-making of a core process. Examples of core process analyzes in which locations might have a bearing on the decisions to be made include the following:

- Base Realignment and Closure (BRAC) (SE process).
- Capability for a new regional command (JCIDS).
- Communications or logistics planning in a mission area (Ops process).
- System and equipment installation and Personnel Type assignments to Facilities (Operations and SE processes).

Examples where Locations play little, if any, role in the process are:

- Prioritization of precision engagement programs (PPBE and portfolio management processes).
- Streamlining of a business process (SE process).
- Doctrine development (JCIDS and Operations processes).

The role of Locations in the decision process was implicit in earlier versions of DoDAF. In this version, they are treated explicitly and precisely to allow more rigorous analysis of requirements (e.g., communications or logistics planning) and clearer differentiation among solutions alternatives).

2.12.1 Data

The DoDAF Meta-model for the data comprising Locations is shown in **Figure 2.12.1-1**. The figure may be hard to read in a hardcopy printout but a version at full-resolution which can be zoomed in is published in the online DoDAF V2.0 Meta-model Data Dictionary. Definitions for the model terms are in **Table 2.12.1-1**. It is important to understand that all of the Types and Individuals are founded on a formal ontology from which they inherit whole-part, super-subtype, before-after, and, in some cases, interface patterns. Additionally, the Tuples inherit a places pattern. These are shown in the DoDAF V2.0 Meta-model Data Dictionary. Aliases and composite terms are in **Table 2.12.1-2**. Authoritative Source definitions, aliases, and rationale are provided in the DoDAF V2.0 Meta-model Data Dictionary. Note that foundational classes are generally not shown on data group diagrams; this foundational material is in Section 2. This includes super-subtype, whole-part, temporal-whole-part, overlap, type-instance (member-of), and before-after patterns. Also not shown are the data structures for classification marking of architecture information at the whole and element (portion) levels using the IC-ISM. The schema for the IC-ISM is in Volume 3. Lastly, note that the size of the icons is not indicative of their importance; sizes are adjusted to reduce line crossings and bends to make the diagrams easier to understand.

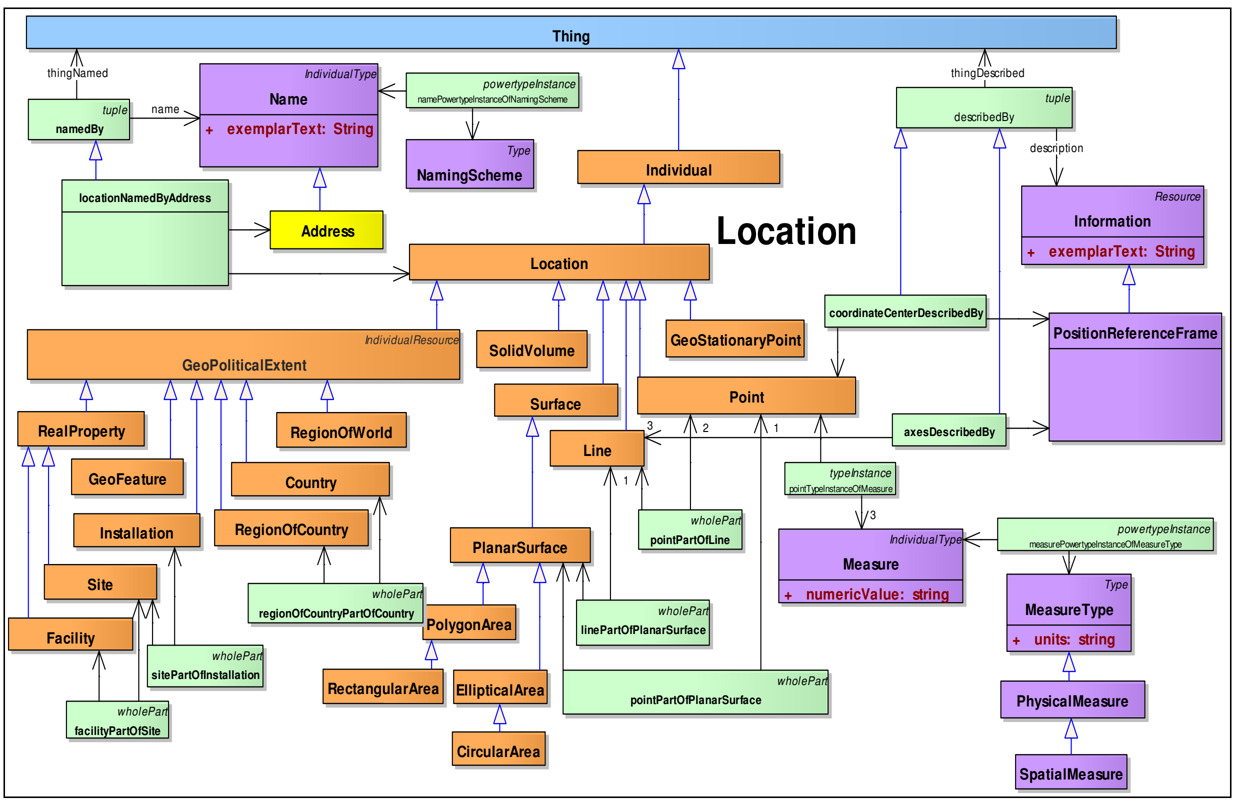


Figure 2.12.1-1: DoDAF Meta-model for Locations

There are several items to note:

- Addresses such as URLs, Universal Resource Names (URNs), postal addresses, datalink addresses, etc. are considered Names for Locations. For example, a postal address is a naming system for the Location of a building. A Universal Resource Locator is a name for a server that is located somewhere on the Web.
- The naming pattern works by identifying the following: 1) the name string, 2) the object being named, and 3) the type of name (e.g., postal address). *Name* here is used in the broadest sense, such that a description is considered a long name.
- The lower left of the diagram is a model of types of Location objects. These can be alternatively named using the naming pattern in the upper left and delineated using the Extent pattern in the lower right.
- Minimal parts of the Spatial Extent (Point, Line, Surface, and Solid Volume) are detailed because of the varying requirements within a federate. That is, member of the federate may need to specialize the Spatial Extents. Some common and simple classes are modeled, such as a Line described by two Points and a Planar Surface defined by a Line and Point.
- Facilities are types of Locations. In prior versions of DoDAF it was not clear if a Facility could be thought of as a system or just a Location. This is now clarified. To describe the functionality of a Facility, the Activities performed by the Performers located at the Facility are described.
- Installation, Site, and Facility follow Army guidance from the Real Property Inventory Requirements (RIPR). Similarly, a Facility can be a linear structure, such as a road or pipeline.
- Geofeatures (called FEATURE in Joint Consultation, Command and Control Information Exchange Data Model (JC3IEDM)) cover man-made control features, as well as geophysical features (including meteorological and oceanographic phenomena).

Table 2.12.1-1: DoDAF Meta-model Definitions for Locations

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Classes		
Address	The name of a location along with the location-finding scheme that allows a location to be found from the name. Examples include postal address, email address, URL, datalink address.	
CircularArea	The space enclosed by a circle.	
Country	A political state or nation or its territory.	
EllipticalArea	The space enclosed by an ellipse.	
Facility	A real property entity consisting of underlying land and one or more of the following: a building, a structure (including linear structures), a utility system, or pavement.	
GeoFeature	An object that encompasses meteorological, geographic, and control features mission significance	
GeoPoliticalExtent	A geospatial extent whose boundaries are by declaration or agreement by political parties.	
GeoStationaryPoint	Unidimensional Individual (dimensionless in space, existant over all time)	
Installation	A base, camp, post, station, yard, center, or other activity, including leased facilities, without regard to the duration of operational control. An installation may include one or more sites.	
Line	A geometric figure formed by a point moving along a fixed direction and the reverse direction.	
Location	A point or extent in space that may be referred to physically or logically.	
Measure	The magnitude of some attribute of an individual.	
MeasureType	A category of Measures	
PhysicalMeasure	A category of measures of spatio-temporal extent of an Individual such as length, mass, energy, velocity, …	
PlanarSurface	A two-dimensional portion of space.	
Point	Unidimensional Individual (dimensionless in space, existant over all time)	
PolygonArea	The space enclosed by a polygon.	

Table 2.12.1-1: DoDAF Meta-model Definitions for Locations

Technical Term	Composite Definition	Potentially Related Terms or Aliases
PositionReferenceFrame	An arbitrary set of axes with reference to which the position or motion of something is described or physical laws are formulated.	
RealProperty	Land and improvements to land (i.e., facilities).	
RectangularArea	The space enclosed by a rectangle.	
RegionOfCountry	A large, usually continuous segment of a political state or nation or its territory.	
RegionOfWorld	A large, usually continuous segment of a surface or space; area.	
Site	Physical (geographic) location that is or was owned by, leased to, or otherwise possessed. Each site is assigned to a single installation. A site may exist in one of three forms: (1) Land only, where there are no facilities present and where the land consists of either a single land parcel or two or more contiguous land parcels. (2) Facility or facilities only, where the underlying land is neither owned nor controlled by the government. A stand-alone facility can be a site. If a facility is not a stand-alone facility, it must be assigned to a site. (3). Land and all the facilities thereon, where the land consists of either a single land parcel or two or more contiguous land parcels.	
Surface	A portion of space having length and breadth but no thickness or regards to time.	
SolidVolume	The amount of space occupied by a three-dimensional object of definite shape; not liquid or gaseous.	
SpatialMeasure	A category of measures of the spatio-temporal location of an Individual.	
	Associations	
axesDescribedBy	A relationship describing the straight lines about which bodies rotate	
coordinateCenterDescribedBy	A relationship describing the mid point of a position reference frame	
facilityPartOfSite	A whole part association between a Facility (part) and the Site (whole) in which it resides.	project goals, objectives, desired outcomes
linePartOfPlanarSurface	A couple that represents the whole part relationship between Line and PlanarSurface	

Table 2.12.1-1: DoDAF Meta-model Definitions for Locations

Technical Term	Composite Definition	Potentially Related Terms or Aliases
locationNamedByAddress	A relationship that represents a location being named by an address	
measurePowertypeInstanceOf MeasureType	Measure is a member of MeasureType	
pointPartOfLine	A couple that represents the whole part relationship between a line and a point	
pointPartOfPlanarSurface	A couple that represents the whole part relationship between a planar surface and a point	
pointTypeInstanceOfMeasure	Point is a member of Measure	
regionOfCountryPartOfCountry	A couple that represents the whole part relationship between a country and a region within it	
sitePartOfInstallation	A whole-part association representing that a Site (the part) is spatio-temporally contained within an Installation (the whole).	

*The Black Text indicates a Concept in DM2, the Red Text indicates an Association in DM2.

Table 2.12.1-2: Aliases and Composite Terms Related to Locations

Technical Term	Composite Definition	Potentially Related Terms or Aliases
Geolocation	A place or site that is either occupied or available for occupancy and is marked by some distinguishing feature.	GeoFeature GeoPoliticalExtent
Physical Asset	Covered by the Real Property and Materiel concepts.	Real Property, Materiel

2.12.2 Method

Methods for collecting and modeling Location data are as follows:

- First, determine the use of the Location data, such as the ones listed in the next paragraph.
- For many architecture applications, a locating scheme is some kind of geometric system because many uses (see next paragraph) require geometric calculations. Named locations (e.g., facility, base, installation, region names) can be applicable since their use may be merely to describe where performance occurs. In addition, the naming pattern basically can use the name as a surrogate for the geometric location, such as postal addresses are rarely applicable to architectures.
- If a geometric system is needed, the coordinate system, reference frame, and units are chosen. The Geospatial Markup Language (GML) defines 26 different kinds of coordinate

systems, including one called user defined. In many cases, a federate may choose reference to GML so issues like handed-ness and orientation don't have to be defined again.

- The accuracy should be determined. For many uses, Locations may not need to be as accurate as some Geospatial system can be, since the use calculation may have many approximations, assumptions, and minor influencing variables that are chosen to be ignored.
- In some cases, there may be need for speed and acceleration ranges. Since these are unusual, they are not part of the core DM2 but would be added as extensions for these kinds of models. The speed could be extended as an attribute or as a trajectory consisting of a set of spatial-temporal points, where the trajectory is a whole and the points are parts.

2.12.3 Use

Data for Locations are used to describe where Performers perform. The Location concept supported the system node in DoDAF V1.0 and V1.5. In DoDAF V2.0, it is generalized and more precisely defined. Examples of the uses of the various types of Locations are:

- Facility Locations allow description that certain systems or organizations are located at a specific facility. Note that the function of the Facility is determined by the Activities performed by the Performers located at the Facility; that is, the Facility itself is not a Performer.
- Installation Locations allow descriptions of certain organizations that operate or use an installation.
- Region Locations are used to describe what Performers and Activities are performed in certain regions.
- A Point Location can be used to state when a Performer is located at a specific Point; e.g., latitude and longitude. When the location is not that specific, Regions, Countries, and other geometric shapes can be used.
- Line (set of lines) allows description of Performers located on, beside, or within some enclosing lines. The line could be described mathematically so that it could specify an orbit, e.g., that certain satellites are in some orbit.
- Volume, e.g., that some systems cover a certain volume; e.g., an air defense system.
- Addresses (names for locations) allow descriptions of where something is located using the address scheme; e.g., the URL address scheme allows for looking up the internet protocol (IP) and then the files on the server.

3. DODAF VIEWPOINTS AND MODELS

DoDAF has been designed to meet the specific business and operational needs of the DoD. It defines a way of representing an enterprise architecture that enables stakeholders to focus on specific areas of interests in the enterprise, while retaining sight of the big picture. To assist decision-makers, DoDAF provides the means of abstracting essential information from the underlying complexity and presenting it in a way that maintains coherence and consistency. One of the principal objectives is to present this information in a way that is understandable to the many stakeholder communities involved in developing, delivering, and sustaining capabilities in support of the stakeholder's mission. It does so by dividing the problem space into manageable pieces, according to the stakeholder's viewpoint, further defined as DoDAF-described Models.

Each viewpoint has a particular purpose, and usually presents one or combinations of the following:

- Broad summary information about the whole enterprise (e.g., high-level operational concepts).
- Narrowly focused information for a specialist purpose (e.g., system interface definitions).
- Information about how aspects of the enterprise are connected (e.g., how business or operational activities are supported by a system, or how program management brings together the different aspects of network enabled capability).

However, it should be emphasized that DoDAF is fundamentally about creating a coherent model of the enterprise to enable effective decision-making. The presentational aspects should not overemphasize the pictorial presentation at the expense of the underlying data.

DoDAF organizes the DoDAF-described Models into the following viewpoints:

- The All Viewpoint describes the overarching aspects of architecture context that relate to all viewpoints.
- The Capability Viewpoint articulates the capability requirements, the delivery timing, and the deployed capability.
- The Data and 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.
- The Operational Viewpoint includes the operational scenarios, activities, and requirements that support capabilities.
- The Project Viewpoint describes the relationships between operational and capability requirements and the various projects being implemented. The Project Viewpoint also details dependencies among capability and operational requirements, system engineering processes, systems design, and services design within the Defense Acquisition System process. An example is the Vcharts in Chapter 4 of the Defense Acquisition Guide.
- The Services Viewpoint is the design for solutions articulating the Performers, Activities, Services, and their Exchanges, providing for or supporting operational and capability functions.
- The Standards Viewpoint articulates the applicable operational, business, technical, and industry policies, standards, guidance, constraints, and forecasts that apply to capability and operational requirements, system engineering processes, and systems and services.
- The Systems Viewpoint, for Legacy support, is the design for solutions articulating the systems, their composition, interconnectivity, and context providing for or supporting operational and capability functions.

A presentation of these viewpoints is portrayed in graphic format in **Figure 3-1**. Additional details about these viewpoints are found in Section 3.1.

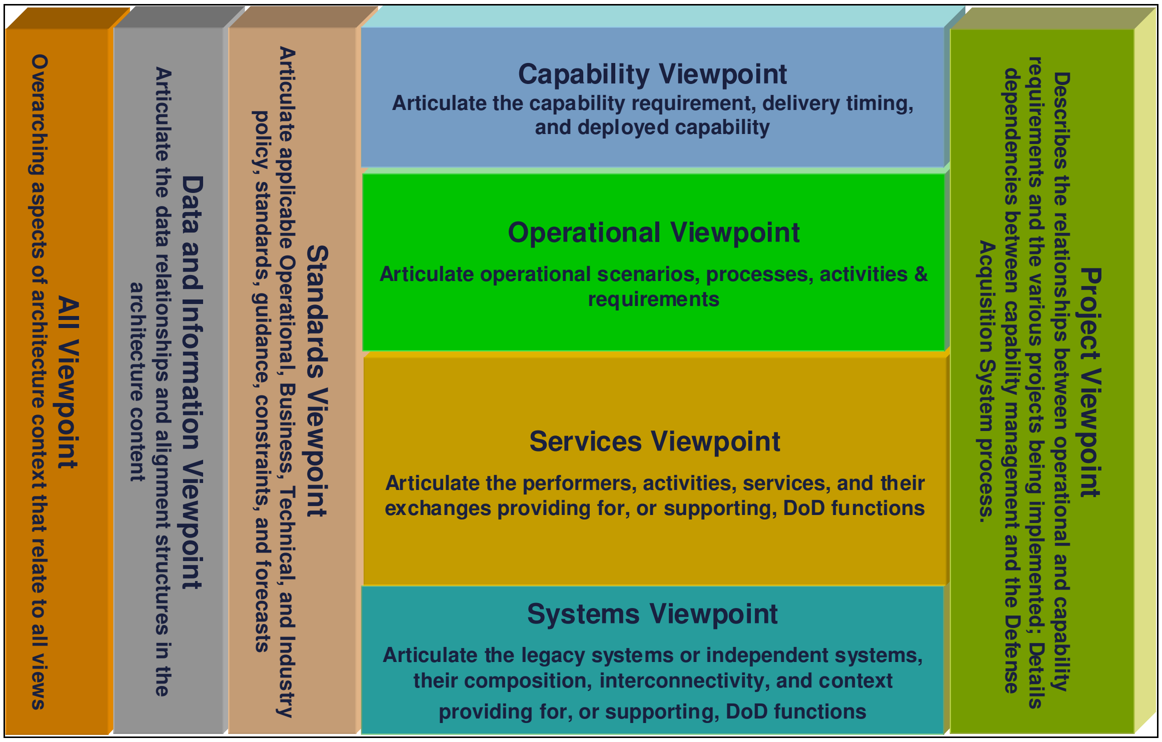


Figure 3-1: DoDAF Viewpoints

DoDAF V2.0 is a more focused approach to supporting decision-makers than prior versions. In the past, decision-makers would look at DoDAF offerings and decide which were appropriate to their decision process. An example is the JCIDS process architecture requirements inside the JCIDS documentation (ICD, CDD, CPD, etc.).

Additionally, older version Architectural Description products were hard-coded in regard to content and how they were visualized. Many times, these design products were not understandable or useful to their intended audience.

DoDAF V2.0, based on process owner input, has increased focus on architectural data, and a new approach for presenting architecture information has addressed the issues. The viewpoints categorize the models as follows:

- As illustrated in Figure 3-2, the original viewpoints (Operational Viewpoint, Systems and Services Viewpoint, Technical Standards Viewpoint, and the All Viewpoint) have had their Models reorganized to better address their purposes. The Services portion of the older Systems and Services Viewpoint is now a Services Viewpoint that addresses in more detail our net-centric or services-oriented implementations.

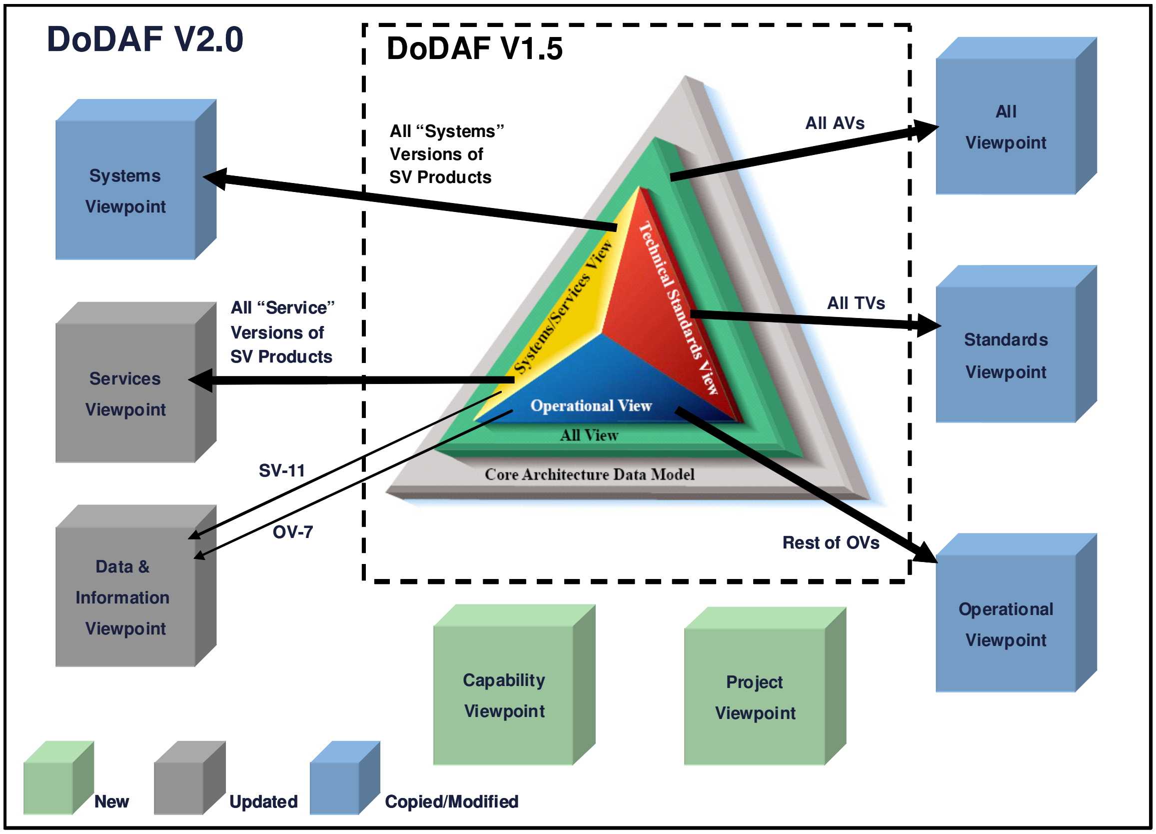


Figure 3-2: DoDAF V1.5 Evolution to DoDAF V2.0

- All the models of data (conceptual, logical, or physical) have been placed into the Data and Information Viewpoint rather than spread throughout the Operational Viewpoint and Systems and Services Viewpoints.
- The Systems Viewpoint accommodates the legacy system descriptions.
- The new Standards Viewpoint can now describe business, commercial, and doctrinal standards, as well as the technical standards applicable to our solutions, which may include systems and services.
- The Operational Viewpoint now can describe rules and constraints for any function (business, intelligence, warfighting, etc.) rather that just those derived from data relationships.
- Due to the emphasis within the Department on Capability Portfolio Management and feedback from the Acquisition community, the Capability Viewpoint and Project Viewpoint have been added through a best-of-breed analysis of the MODAF and NAF constructs.

Workshops have brought the Systems Engineering community and the architecture community closer together in defining the DoDAF architecture content that would be useful to the Systems Engineering process, and this has resulted in an understanding which the entire set of viewpoints and the underlying architectural data can be used in the System Engineering processes. There is not a set of separate System Engineering viewpoint or DoDAF-described Models as the system engineer and system engineering decision-makers can use the existing DoDAF-described Models and their own defined Fit-for-Purpose Views.

The approach to the presentation of Architectural Description moves away from static and rigid one-size-fits-all templates of architecture portrayals for architects. The term we have coined is "Fit-for-Purpose" presentation. Through various techniques and applications, the presentation of Architectural data increases customer understanding and architecture's usefulness to decision-making by putting the data underlying the architectural models into the context of the problem space for each decision-maker.

Details of the DoDAF-described Models, DoDAF V1.5 Support, and the relationships of the DoDAF Meta-model Groups (defined in Section 2) to the DoDAF Viewpoints and DoD Key Processes are shown later in Section 3.

3.1 Viewpoint and DoDAF-described Model Descriptions

The following DoDAF Viewpoints and DoDAF-described Models are discussed below with some details, such as model uses and model descriptions:

- All Viewpoint.
- Capability Viewpoint.
- Data and Information Viewpoint.
- Operational Viewpoint.
- Project Viewpoint.
- Services Viewpoint.
- Standards Viewpoint.
- Systems Viewpoint.

For the DoDAF-described Model descriptions, a major source of material was adapted from MODAF. In addition, a note on system engineering is included.

The Views described in DoDAF, including those that are legacy Views from previous versions of the Framework, are provided as pre-defined examples that can be used when developing presentations of architectural data.

DoDAF is prescribed for the use and development of Architectural Descriptions in the Department. Specific DoDAF-described Models for a particular purpose are prescribed by process-owners. All the DoDAF-described Models do not have to be created. DoDAF V2.0 is "Fit-for-Purpose", based on the decision-maker needs. DoDAF does not prescribe any particular Views, but instead concentrates on data as the necessary ingredient for architecture development. However, other regulations and instructions from both DoD and CJCS may have particular presentation view requirements. These Views are supported by DoDAF 2.0, and should be consulted for specific view requirements.

3.1.1 All Viewpoint

There are some overarching aspects of an Architectural Description that are captured in the AV DoDAF-described Models. The AV DoDAF-described Models provide information pertinent to the entire Architectural Description rather than representing a distinct viewpoint. AV DoDAF-described Models provide an overview of the architecturectural effort including such things as the scope, context, rules, constraints, assumptions, and the derived vocabulary that pertains to the Architectural Description. It captures the intent of the Architectural Description to help ensure its continuity in the face of leadership, organizational, and other changes that can occur over a long development effort.

Names of the models and their descriptions (in **Table 3.1.1-1**) are provided below.

Table 3.1.1-1: All Viewpoint Model Descriptions

Models	Descriptions
AV-1 Overview and Summary Information	Describes a 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.

3.1.1.1 Uses of All Viewpoint DoDAF-described Models. The AV DoDAF-described Models captures the scope of the architecture and where the architecture fits in relationship to other architectures. Another use of the All Viewpoint is for the registration of the architecture to support the net-centric goals of making Architectural Descriptions visible (Discoverable).

Mappings of the All Viewpoint DoDAF-described Models to the DM2 Concepts, Associations, and Attributes are in **Table B-1 DM2 Concepts, Associations, and Attributes Mapping to DoDAF-described Models**. The DM2 Concepts, Associations, and Attributes are described in the DoDAF Meta-model Data Dictionary.

3.1.1.2 Model Descriptions. The All Viewpoint DoDAF-described Models are described below. Examples of these models can be viewed online in the public DoDAF Journal.

3.1.1.2.1 AV-1 Overview and Summary Information. 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 content 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:

- In the initial phases of architecture development, it serves as a planning guide.
- 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.

Detailed Description:

An enterprise has an architecture, which is manifested through an Architectural Description (in this case, a DoDAF described Architectural Description). That Architectural Description consists of a number of populated views each of which is an instance of a specific model or a combination of model. DoDAF consists of a set of viewpoints and these are organized in terms of models. Each model is associated with a specific set of concerns that certain stakeholders have, and which the models constructed are intended to address. The stakeholder groupings tend to align with the model definitions within a viewpoint (so the DoDAF Operational Viewpoint relates to operational stakeholders, i.e., end users). Finally each Architectural Description has a rationale that governs the selection of Models that will be used and the scope of the underlying models. The AV-1 is intended to describe this.

The AV-1 is usually a structured text product. An architecting organization may create a template for the AV-1 that can then be used to create a consistent set of information across different architecture-based projects. While the AV-1 is often dispensed with or “retrofitted” to a finished architecture package, it’s desirable to do it up-front because the AV-1 provides a summary of a given Architectural Description and it documents the following descriptions:

- Architectural Description Identification – Identifies the Architectural Description effort name, the architect, and the organization developing the Architectural Description. It also includes assumptions and constraints, identifies the approving authority and the completion date, and records the level of effort required to develop the Architectural Description.
- Scope – Identifies the Viewpoints, DoDAF-described Models, and Fit-for-Purpose Views that have been selected and developed. The AV-1 should address the temporal nature of the Architectural Description, such as the time frame covered, whether by specific years or by designations such as “current”, “target”, or transitional. Scope also identifies the organizational entities and timelines that fall within the scope of the Architectural Description.
- Purpose and perspective – Explains the need for the Architectural Description, what it will demonstrate, the types of analyses that will be applied to it, who is expected to perform the analysis, what decisions are expected to be made based of each form of analysis, who is expected to make those decisions, and what actions are expected to result. The perspective from which the Architectural Description is developed is identified.

- Context – Describes the setting in which an Architectural Description exists. Context includes such things as: mission, doctrine, relevant goals and vision statements, concepts of operation, scenarios, information assurance context (e.g., types of system or service data to be protected, such as classified or sensitive but unclassified, and expected information threat environment), other threats and environmental conditions, and geographical areas addressed, where applicable. Context also identifies authoritative sources for the standards, rules, criteria, and conventions that are used in the architecture. Any linkages to parallel architecture efforts should be identified.
- Status – Describes the status of the architecture at the time of publication or development of the AV-1 (which might precede the architectural development itself). Status refers to creation, validation and assurance activities.
- Tools and File Formats Used – Identifies the tool suite used to develop the Architectural Description and file names and formats for the Architectural Models if appropriate.
- Assumptions and Constraints.
- Archtecture development schedule including start date, development milestones, date completed, and other key dates. Further details can be reflected in the Project Viewpoint.

If the architecture is used to support an analysis, the AV-1 may be extended to include:

- Findings – States the findings and recommendations that have been developed based on the architectural effort. Examples of findings include: identification of shortfalls, recommended system implementations, and opportunities for technology insertion.
- Costs – the architecture budget, cost projections, or actual costs that have been incurred in developing the architecture and/or undertaking the analysis. This might include integration costs, equipment costs and other costs.

During the course of developing an Architectural Description, several versions of the AV-1 may be produced. An initial version may focus the effort and document its scope, the organizations involved, and so forth. After other Models within an Architectural Description's scope have been developed and verified, another version may be produced to document adjustments to the scope and to other aspects of the Architectural Description that may have been identified. After an Architectural Description has been used for its intended purpose, and the appropriate analysis has been completed, a final version should be produced to summarize these findings for high-level decision-makers. In this version, the AV-1 and a corresponding graphic in the form of an OV-1 serve as an executive summary of the Architectural Description. The AV-1 can be particularly useful as a means of communicating the methods that have been applied to create models and the rationale for grouping these models. Viewing assumptions that have shaped individual models may also be included. In this form, the AV-1 needs to list each individual model and provide a brief commentary.

This could take several forms:

- It could refer to one or more DoDAF-described Models.
- It could refer to the DoDAF Community of Practice.
- It could refer to a focus for the work, e.g., integration or security.
- It could refer to a combination of these.

Finally, each Architectural Description has a rationale that governs the selection of the Models used and the scope of the underlying models as a result of employing the 6-Step Architecture Development Process. The AV-1 DoDAF-described Model is intended to describe the decisions made throughout that process.

3.1.1.2.2 AV-2: Integrated Dictionary. 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.

It is essential that organizations within the DoD use the same terms to refer to a thing. 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 were 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 re-use.
- Traces architectural data to authoritative data sources.

This is facilitated by the DM2. 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.

In the creation of any Architectural Description, reuse of authoritative external taxonomy content, e.g., the FEA Reference Models, or the Joint Common System Function List, are important to aligning the architectural content across many descriptions to increase their understandability, interoperability, Architecture Federation, and compliance. A discussion on the use of taxonomies in the development of the AV-2 and the architecture effort is below.

Detailed Description:

The AV-2 content can be organized by the following areas within the DM2 that can be used to expedite architecture development:

- Capabilities: The taxonomy should minimally consist of names, descriptions, and conditions that may be applicable to performance measures.

- Resource Flow. The taxonomy should minimally consist of names of information elements exchanged, descriptions, decomposition into constituent parts and subtypes, and mapping to system data elements exchanged.
- Activities (Operational Activities or Tasks).[16] The taxonomy should minimally consist of names, descriptions, and decomposition into the constituent parts that comprise an activity.
- Activities (System or Service Functions). The taxonomy should minimally consist of names, descriptions, and decomposition into the constituent parts that comprise a system function.
- Performance Parameters. The taxonomy should minimally consist of names, descriptions, units of measure, and conditions that may be applicable to performance parameters.
- Performers: Performers can be persons, services, systems or organizations. The taxonomy should minimally consist of names, descriptions, breakdowns into constituent parts (e.g., a services comprising other services), and applicable categorizations. Each of the above types of performers is a candidate for a being a taxonomy.
- Skills: The taxonomy should minimally consist of names, descriptions, units of measure, and conditions that may be applicable to performance parameters.
- Standards: The taxonomy should minimally consist of categories of standards (e.g., DoD Information Technology Standards and Profile Registry [DISR]'s Service Areas).
- Triggers/Events: The taxonomy minimally consists of names, descriptions, and breakdown into constituent parts of the event or trigger and categorization of types of events or triggers.

Not all architectural data in a given taxonomy is useful in every case of architectural development. However, given the ongoing evolutionary change in organizations, services, systems, and activities, the value of using established, validated taxonomic structures that can be expanded or contracted as needed becomes obvious. Moreover, the development of new models over time is greatly simplified as understanding of the taxonomies is increased. Standard taxonomies, like DISR Service Categories, become building blocks for more comprehensive, quality architectural DoDAF-described Models and Fit-for-Purpose Views. The DoD Extensible Markup Language Registry and Clearinghouse and the Net-Centric Implementation Document (NCID) are potential sources for taxonomies.

In some cases, a specific community may have its own operational vocabulary. This local operational vocabulary may use the same terms in radically different ways from other operational communities. (For example, the use of the term track refers to very different concepts in the carrier battle group community than in the mine-sweeper community. Yet both of these communities are Navy operational groups and may participate together in littoral warfare task forces.) In these cases, the internal community versions of the models and views within the Architectural Description should use the vocabulary of the local operational community to achieve community cooperation and buy-in. Data elements need to be uniquely identified and consistently used across all viewpoints, models and views within the Architectural Description. These populated views should include notes on any unique definitions used and provide a mapping to standard definitions, where possible.

[16] Operational Activities defined and standardized by the Joint Staff are in the form of Mission Essential Tasks [CJCSM 3500.04E, 25 AUGUST 2008]. Operational Activities are also specified (and sometimes standardized) in the form of process activities arising from process modeling. It is sometimes convenient to merge these sets, either as activities or tasks.

3.1.2 Capability Viewpoint

The Capability Viewpoint and the DoDAF-described Models within the viewpoint are introduced into DoDAF V2.0 to address the concerns of Capability Portfolio Managers. In particular, the Capability Models describe capability taxonomy and capability evolution.

The DoD increasingly employs incremental acquisition to help manage the risks of complex procurements. Consequently, there is a need to provide visualizations of the evolving capabilities so that Portfolio Managers can synchronize the introduction of capability increments across a portfolio of projects. The Capability Models included within DoDAF are based on the program and capability information used by Portfolio Managers to capture the increasingly complex relationships between interdependent projects and capabilities.

Another justification for the Capability Viewpoint is the increasing importance of transformational programs within the DoD (e.g., Global Exchange [GEX], Defense Acquisition Initiative [DAI]). These types of programs are focused on the delivery of capabilities and do not conform to the standard for project management and tend to be benefit-driven rather than capability delivery focused. An ability to view these transformational programs, and their interdependencies, provides a potentially powerful tool for DoD Enterprise Architects.

Names of the models and their descriptions (in **Table 3.1.2-1**) are provided below.

Table 3.1.2-1: Capability Model Descriptions

Model	Description
CV-1: 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 Organizational 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 Operational Activities 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.

Mappings of the Capability Viewpoint DoDAF-described Models to the DM2 Concepts, Associations, and Attributes are in **Table B-1 DM2 Concepts, Associations, and Attributes**

Mapping to DoDAF-described Models. The DM2 Concepts, Associations, and Attributes are described in the DoDAF Meta-model Data Dictionary.

The Capability Viewpoint DoDAF-described Models are described below and are discussed with examples in the DoDAF Product Development Questionnaire Analysis Report. This document can be viewed online in the public DoDAF Journal.

3.1.2.1 Use of Capability Viewpoint Models. The CV DoDAF-described Models listed within this section of the document are intended to provide support to various decision processes within the Department, one of which is portfolio management. Since the DoD has moved toward the delivery of capabilities, these models take on a more important role. Developing an architecture that includes the relationships necessary to enable a capability thread is essential to improving usability of architectures, as well as increasing the value of federation.

In the above context, a capability thread is similar to the result of a query that would be run on a particular capability. For example, if an architecture were to include operational activities, rules, and systems, a capability thread would equate to the specific activities, rules, and systems that are linked to that particular capability. The CV DoDAF-described Models are used to provide the strategic perspective and context for other architectural information.

The concept of capability, as defined by its Meta-model Data Group, within Section 2, allows one to answer questions such as:

- How does a particular capability or capabilities support the overall mission/vision?
- What outcomes are expected to be achieved by a particular capability or set of capabilities?
- What services are required to support a capability?
- What is the functional scope and organizational span of a capability or set of capabilities?
- What is our current set of capabilities that we are managing as part of a portfolio?

3.1.2.2 Model Descriptions. The CV DoDAF-described Models are described below. In addition, examples of models can be viewed online in the public DoDAF Journal.

3.1.2.2.1 CV-1: Vision. The CV-1 addresses the enterprise concerns associated with the overall vision for transformational endeavors and thus defines the strategic context for a group of capabilities. 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.

Detailed Description:

The CV-1 defines the strategic context for a group of capabilities described in the Architectural Description by outlining the vision for a capability area over a bounded period of time. It describes how high-level goals and strategy are to be delivered in capability terms.

A CV-1 may provide the blueprint for a transformational initiative. The CV-1 may be primarily textual descriptions of the overarching objectives of the transformation or change program that the Enterprise is engaged in. Of key importance is the identification of Goals, together with the desired outcomes and measurable benefits associated with these.

3.1.2.2.2 CV-2: Capability Taxonomy. The CV-2 captures capability taxonomies. The model presents a hierarchy of capabilities. These capabilities may be presented in context of a timeline – i.e., it can show the required capabilities for current and future capabilities. 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.

In CV-2, the Capabilities are only described in the abstract – i.e., CV-2 does not specify how a capability is to be implemented. A CV-2 is structured as a hierarchy of capabilities, with the most general at the root and most specific at the leaves. At the leaf-level, capabilities may have a measure specified, along with an environmental condition for the measure.

When capabilities are referenced in operational or systems architectures, it may be that a particular facility, location, or organization or configuration meets more than one level of capability. The CV-2 is used to capture and organize the capability functions – required for the vision set out in the CV-1 Vision.

In contrast to AV-2 Integrated Dictionary, a CV-2 is structured using only one type of specialization relationship between elements: sub-supertype. A sub-supertype relationship is a relationship between two classes with the second being a pure specialization of the first.

In DoDAF V2.0, capabilities exist in space and over time, that is they are intended to provide a framework across the lifetime of the enterprise that is being modeled. This means that it is feasible to develop a capability taxonomy that can apply to all architecture phases.

In addition to the capability nomenclature, appropriate quantitative attributes and measures for that specific capability or function need to be included e.g., the required speed of processing, the rate of advance, the maximum detection range, etc. These attributes and measures will remain associated with the capability whenever it is used across the Architectural Description. The quantitative values expressed may be linked to specific phases (or be “As-Is” values and/or or “To-Be” targets).

The CV-2 has no mandated structure although the architectural data must be able to support the representation of a structured/hierarchal list. This structure may be delivered using textual, tabular or graphical methods. The associated attributes and measures for each capability can either be included on the main CV-2 or in tabular format as an appendix if the inclusion of the attributes and measures would over complicate the presentation of the populated view.

3.1.2.2.3 CV-3: Capability Phasing. The CV-3 addresses the planned achievement of capability at different points in time or during specific periods of time, i.e., capability phasing. The CV-3 supports the capability audit processes 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.

Detailed Description:

The CV-3 provides a representation of the available capability at different points in time or during specific periods of time (associated with the phases – see CV-1 Vision model). A CV-3 can be used to assist in the identification of capability gaps/shortfalls (no fielded capability to fulfill a particular capability function) or capability duplication/overlap (multiple fielded capabilities for a single capability function).

The CV-3 is populated by analyzing programmatic project data to determine when projects providing elements of capability are to be delivered, upgraded and/or withdrawn (this data may be provided in part by a PV-2 Project Timelines model). Then capability increments identified can be structured according to the required capabilities determined in the CV-2 Capability Taxonomy model and the phases. Alternatively, a set of desired capability increments can be viewed and then compared to the project plans. In practice, the population of the model tends to iterate between considering the desired capability and considering what capability is planned to be delivered. The output from this iterative approach can be a table that represents the required capability phasing.

The CV-3 can be presented as a table consisting of rows representing Capabilities (derived from the CV-2 Capability Taxonomy model) and columns representing phases (from CV-1 Vision model).

At each row-column intersection in the CV-3 table, the capability increment that represents the change in Capability within that phase can be displayed. If the availability of the Capability spans multiple periods of time, then this can be indicated by an elongated color-coded bar. If there are no Capabilities planned to satisfy the Capability Requirements in that period of time then a blank space can be left.

A variant CV-3, in which the names of the projects that can deliver the capability increments are included, can identify capability gaps and shortfalls. The essence is the relationship between projects, capabilities and time. The model may be used to envisage the need for interventions in projects (to fulfill a capability gap) or to represent current plans (the availability of capability according to their delivery timescales).

3.1.2.2.4 CV-4: Capability Dependencies. 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.).

Detailed Description:

The CV-4 describes the relationships between capabilities. It also defines logical groupings of capabilities. This contrasts with CV-2 Capability Taxonomy model which also deals with relationships between Capabilities; but CV-2 only addresses specialization-generalization relationship (i.e., capability taxonomy).

A CV-4 shows the capabilities that are of interest to the Architectural Description. It groups those capabilities into logical groupings, based on the need for those elements to be integrated.

An approach for describing a CV-4 is graphical. In some cases, it may be important to distinguish between different types of dependency in the CV-4. Graphically, this can be achieved by color-coding the connecting lines or by using dashed lines. From a data perspective, the CV-4 can make use pre-existing capability dependency types in the DoDAF Meta-model; else new, specific dependency types can be created. The new dependency types need to be recorded the in the AV-2: Integrated Dictionary.

3.1.2.2.5 CV-5: Capability to Organizational Development Mapping. The CV-5 addresses the fulfillment of capability requirements.

This model shows the planned capability deployment and interconnection for a particular phase. and should 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, in particular, 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.

Detailed Description:

The CV-5 shows deployment of Capabilities to specific organizations. CV-5 models are specific to a phase. If a particular Capability is/was used by (or is due to be used by) a specific organization during that phase, it should be shown on the CV-5, mapped to the organization. The CV-5 may also show interactions between them (where these have been previously defined in a

SV-1 Systems Interface Description or SvcV-1 Services Context Description). The CV-5, along with SV-8 Systems Evolution Description, SvcV-8 Services Evolution Description and PV-2 Project Timelines models can be regarded as amplifying the information contained in the CV-3.

To conduct a comprehensive analysis, several CV-5s can be created to represent the different phases. Although the CV-5s are represented separately, Capabilities may exist in more than one model. The information used to create the CV-5 is drawn from other DoDAF-described Models (PV-2 Project Timelines, CV-2 Capability Taxonomy, OV-4 Organizational Relationships Chart, SV-1 Systems Interface Description, SvcV-1 Services Context Description), and the timing is based on PV-2 Project Timelines indicating delivery of Capabilities to actual organizational resources, and also the point at which those organizational resources cease to use a particular Capability.

System interaction (from the SV-1 Systems Interface Description) or Service interaction (from the SvcV-1 Services Context Description) can be shown on a CV-5. In addition, where a Capability or resources is deployed across a number of Organizations, a parent Organization can be created for context purposes, and the Capability or resource stretched across the domain of the parent Organization.

The architect should not overwhelm the diagram with capabilities and organizations. A CV-5 should be seen as a summary of the delivery schedules for capabilities (hence it could be argued that it belongs in the PV Viewpoint). To prevent constraining the solution space, CV-5 should not be produced at the time of developing capability/user requirements, but after the solution is determined. Instead, the CV-5 should be more of an informative from a programmatic standpoint.

The CV-5 is usually based on a tabular representation, with the appropriate Organizational structure represented by one axis, and the capabilities by the other axis. Graphical objects representing Capabilities or resources can be placed in the relevant positions (intersections) relative to these axes.

3.1.2.2.6 CV-6: Capability to Operational Activities Mapping. 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 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.

Detailed Description:

A CV-6 shows which elements of capability may be utilized in support of specific operational activities by means of a mapping matrix. If the CV-6 is created as part of a strategic architecture (i.e., before the creation of supporting operational models), it is recommended that the operational activities described in the CV-6 should be common functions. This model may be used indicate that an operational capability (perhaps reflecting a particular user requirement) does or does not fulfill the requirements for capability for a particular phase.

In principle, there could be a different CV-6 created for each phase of the capability development, or perhaps for different capability phasing scenarios. In most cases, it is considered that a single table can be constructed because the operational activities that are most likely relevant to this model may be relatively high-level. If capabilities associated are generic (see CV-1 Vision model), then they should have a well understood relationship with a standard set of operational activities and this relationship is unlikely to change over time.

This model is analogous to the SV-5a Operational Activity to System Function Traceability Matrix – but provides the interface between Capability and Operational Models rather than Operational to System Models.

The CV-6 can have a tabular presentation. The rows can be the Capabilities and the columns can be the Operational Activities. An X may indicate that the capability may be utilized in support of that activity whereas a blank indicates that it does not. Alternatively, a date or phase can indicate that the capability may support that activity by the date or phase indicated.

3.1.2.2.7 CV-7: Capability to Services Mapping. 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.

Detailed Description:

The CV-7 describes the mapping between capabilities required and the services that those capabilities support. A CV-7 shows which elements of capability may be utilized in support of specific services by means of a mapping matrix. If the CV-7 is created as part of a strategic architecture (i.e., before the creation of supporting service models), it is recommended that the services used as part of the CV-7 are common functions. This model may be used indicate that an operational capability (perhaps reflecting a particular user requirement) does or does not fulfill the requirements for capability for a particular phase.

In principle, there could be a different CV-7 created for each phase of the capability development, or perhaps for different capability phasing scenarios. In most cases, it is considered that a single table can be constructed because the services that are most likely relevant to this model may be relatively high-level. If capabilities associated are generic (see CV-1 Vision model), then they should have a well understood relationship with a standard set of services and this relationship is unlikely to change over time.

This model is analogous to the SV-5a Operational Activity to System Function Traceability Matrix – but provides the interface between Capability and Service Models rather than Operational to System Models.

The CV-7 can have a tabular presentation. The rows can be the Capabilities and the columns can be the services. An X indicates that the capability may be utilized in support of that service whereas a blank indicates that it does not. Alternatively, a date or phase can indicate that the capability may support that service by the date or phase indicated.

3.1.3 Data and Information Viewpoint

DoDAF-described Models within the Data and Information Viewpoint provide a means of portraying the operational and business information requirements and rules that are managed within and used as constraints on the organizations business activities. Experience gained from many enterprise architecture efforts within the DoD led to the identification of several levels of abstraction necessary to accurately communicate the information needs of an organization or enterprise. The appropriate level or levels of abstraction for a given architecture are dependent on the use and the intended users of the architecture. Where appropriate, the data captured in this viewpoint needs to be considered by COIs.

DoDAF V2.0 incorporates three levels of abstraction that correlate to the different levels associated with most data models developed in support of the operations or business. These levels are:

- Conceptual.
- Logical.
- Physical.

Names of the models and their descriptions (in **Table 3.1.3-1**) are provided below.

Table 3.1.3-1: Data and Information Model Descriptions

Models	Descriptions
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.

Mappings of the Data and Information Viewpoint DoDAF-described Models to the DM2 Concepts, Associations, and Attributes are in **Table B-1 DM2 Concepts, Associations, and Attributes Mapping to DoDAF-described Models**. There is traceability between the DIV-1 to the DIV-2 to the DIV3 as follows:

- The information representations in the DIV-1 are same, decomposed into, or factored into the data representations in the DIV-2. The DIV-1 information representations can range in detail from concept lists to structured lists (i.e., whole-part, super-subtype), to inter-related concepts. At the DIV-1 level, any relationships are simply declared and then at the DIV-2 level they are made explicit and attributed. Similarly, attributes (or additional relationships) are added at the DIV-2 level.
- The DIV-3's performance and implementation considerations usually result in standard modifications of the DIV-2 and so it traces quite directly. That is, no new semantics are introduced going from the DIV-2 to the DIV-3.

The DM2 Concepts, Associations, and Attributes are described in the DoDAF Meta-model Data Dictionary.

3.1.3.1 Uses of Data and Information Viewpoint DoDAF-described Models. The DIV DoDAF-described Models provide means of ensuring that only those information items that are important to the organization's operations and business are managed as part of the enterprise. They are also useful foundations for discussion with the various stakeholders of the architecture (e.g., decision-makers, architects, developers). These stakeholders require varying levels of detail to support their roles within the enterprise.

When building an architecture using a structured analysis approach, the items captured as part of the data model can be derived from the inputs and outputs associated to the organizations activities. Building the data model in this manner ties the data being managed within the architecture to the activities that necessitate that data. This provides a valuable construct enabling the information to be traceable to the strategic drivers of the architecture. This also enables the data to be used to map services and systems to the business operations. The conceptual data model would be a good tool to use when discussing this traceability with executive decision-makers and persons at that level.

The logical data model bridges the gap between the conceptual and physical-levels. The logical data model introduces attributes and structural rules that form the data structure. As evidenced by the content, this model provides more detail than the conceptual model and communicates more to the architects and systems analysts types of stakeholders. This is one model that helps bridge the gap between architecture and system development. It provides a valuable tool for generating requirements and test scripts against which services and systems can be tested.

Lastly, the physical data model is the actual data schema representative of the database that provides data to the services and applications using the data. This schema is usually a de-normalized data structure optimized to meet performance parameters. The physical data model usually can be generated from a well-defined logical data model then used by database developers and system developers or it can be developed separately from the logical data model (not the optimum method of development) and optimized by the database and system developers.

This model can be used to develop XML message sets and other physical exchange specifications enabling the exchange of architecture information.

3.1.3.2 Metadata Groups Used to Create Data and Information Models. The previous DoDAF-described Models focused on particular areas within the DoDAF Meta-model from which the majority of the information within the models can be extracted. For example, the Capability Viewpoint DoDAF-described Models are in large part made up of data extracted from the Capability Metadata groups. The same is true for Project, Services and the like. The Data and Information DoDAF-described Models are somewhat different.

The Data and Information DoDAF-described Models contain information extracted from all of the metadata groups. Therefore, any information that an organization is managing using its enterprise architecture, should be captured within the Data and Information Models. As previously stated, there are levels of detail that are not included in all models (e.g., the conceptual data model is usually not fully attributed like the logical and physical) but the information item itself (e.g., capability, activity, service) should be represented in all models. Together, the three types of models help bridge the gap between architecture being used as requirements and architecture being used to support system engineering.

3.1.3.3 Model Descriptions. The Data and Information Viewpoint DoDAF-described Models are described below. In addition, examples of models can be viewed online in the public DoDAF Journal.

3.1.3.3.1 DIV-1: Conceptual Data Model. 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 inter-relationships.

The intended usage of the DIV-1 includes:

- Information requirements
- Information hierarchy

Detailed Description:

The DIV-1 DoDAF-described Model describes the structure of an Architectural Description domain's information types and the structural business process rules (defined in the OV Models).

The Architectural elements for DIV-1 include descriptions of information entity and relationship types. Attributes can be associated with entities and with relationships, depending on the purposes of the Architectural Description.

The intention is that DIV-1 describes information or data of importance to the business (e.g., information products that might be referred to in doctrine, SOPs, etc.) whereas the DIV-3 Physical Data Model describes data relevant at the system-level.

The purpose of a given Architectural Description helps to determine the level of detail needed in this model. This level of detail is driven in particular by the criticality of the interoperability requirements.

Often, different organizations may use the same Entity name to mean very different kinds of information having different internal structure. This situation could pose significant interoperability risks, as the information models may appear to be compatible, e.g., each having a Target Track data Entity, but having different and incompatible interpretations of what Target Track means.

A DIV-1 may be necessary for interoperability when shared information syntax and semantics form the basis for greater degrees of information systems interoperability, or when an information repository is the basis for integration and interoperability among business activities and between capabilities.

The DIV-1 defines the Architectural Description's information classes and the relationships among them. For example, if the architecture effort is describing missile defense, some possible information classes may be trajectory and target with a relationship that associates a target with a certain trajectory. The DIV-1 defines each kind of information classes associated with the Architectural Description scope, mission, or business as its own Entity, with its associated attributes and relationships. These Entity definitions correlate to OV-2 Operational Resource Flow Description information elements and OV-5b Operational Activity Model inputs, outputs, and controls.

The DIV-1 should not be confused with the DoDAF Meta-model. Architectural data types for the DoDAF (i.e., DoDAF-defined architectural data elements and DM2 entities) are things like Performer and Operational Activity. The DM2 does provide a specification of the underlying semantics for DoDAF-described Models such as DIV-1. DIV-1 describes information about a specific Architectural Description scope.

3.1.3.3.2 DIV-2: Logical Data Model. 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 and 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.

Detailed Description:

The DIV-2 is a generalized formal structure in computer science. It directly reflects the paradigm or theory oriented mapping from the DIV-1 Conceptual Data Model to the DIV-2.

Possible Construction Methods: DoDAF does not endorse a specific data modeling methodology. The appropriate way to develop a logical data model depends on the technology chosen as the main design solution (e.g., relational theory or object-orientation). For relational theory, a logical data model seems best described using an entity relationship diagramming technique. For Object-Oriented, a logical data model seems best described using Class and/or Object diagrams.

In either case, attention should be given to quality characteristics for the data model. Definition and acceptance of data model quality measures (not data quality measures) for logical data models are sparse. There is some research, e.g., [17] and [18] and can be an area of expected evolution. Instead, there are best practices [19, 20] with more resources of this type[21, 22, 23]. Framed as a software verification, validation, and quality factors, types of best practices include:

- Validation Factors – Was the Right Model Built?
- Information Requirements Fidelity.
- Conceptual, Logical, and Physical Traceability.
- Adherence to Government and Industry Standards and Best Practices.
- Domain Values.
- Resource Exchange and Other Interoperability Requirements.
- Net-Centric Factors.
  - XML Registration.
  - COI Participation.
  - DDMS Compatibility.
- Identifiers and Labels.
- Verification Factors – Was it Well Built?
- Design Factors.
- Compactness.
- Abstraction and Generalization.
- Ontologic Foundations.
- Semantic Purity.
- Logical and Physical Redundancy.

---

[17] Marcela Genero1, Geert Poels2, and Mario Piattini1; "Defining and Validating Measures for Conceptual Data Model Quality"; CAISE 2002; A. Banks Pidduck et al. (Eds.); Springer-Verlag Berlin Heidelberg 2002

[18] Mario Piattini, Marcela Genero, Coral Calero; Data Model Metrics; Grupo Alarcos, Univery of Castilla La Manch; Ciudad Real, Spain.

[19] Matthew West; Developing High Quality Data Models; European Process Industries STEP Technical Liaison Executive (EPISTLE); 1996.

[20] David C. Hay; "Building Quality Data Models"; Essential Strategies, Inc.; 1994.

[21] http://www.tdan.com/.

[22] http://www.tdwi.org/.

[23] http://www.omg.org/.

- Separation of Concerns.
- Software Quality Factors.
- Documentation.
- Naming Conventions.
- Naming and Business Languages.
- Definitions.
- Completeness.
- Consistency.
- Implementability.
- Enumerations/free text ratio.

An example design factor is normalization– essentially one representation for any particular real-world object. There are degrees of normalization with third normal form (3NF) being commonly used. At 3NF, there are no repeating attributes; instead techniques like lookup tables, super-subtyping to carry the common attributes at the supertype-level, and entity decomposition into smaller attribute groupings are used. For the DIV-2, care should be taken to avoid hidden overlaps, where there is a semantic overlap between concepts with different entity, attribute, or domain value names.

3.1.3.3.3 DIV-3: Physical Data Model. 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 actually 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.

Note that DoDAF talks about information in the Operational Viewpoint and data in the System Viewpoint or Services Viewpoint. The intention of this distinction is that DIV-2 Logical Data

Model describes information of importance to the business, (e.g., information products that might be referred to in doctrine, SOPs etc.) whereas DIV-3 describes data relevant at the system or service-level.

Detailed Description:

The DIV-3 is an implementation-oriented model that is used in the Systems Viewpoint and Services Viewpoint to describe how the information requirements represented in DIV-2 Logical Data Model are actually implemented. Entities represent:

- System Resource flows in SV-4 Systems Functionality Description.
- System Resource elements specified in SV-6 Systems Resource Flow Matrix and SV-10c Systems Event-Trace Description.
- Service Resource flows in SvcV-4 Services Functionality Description.
- Service Resource elements specified in SvcV-6 Services Resource Flow Matrix and SvcV-10c Services Event-Trace Description.
- Triggering events in SV-10b Systems State Transition Description or SvcV-10b Services State Transition Description.
- Events in SV-10c Systems Event-Trace Description or SvcV-10c Services Event-Trace Description.
- Elements required due to Standards in the StdV-1 Standards Profile or StdV-2 Standards Forecast.

For some purposes, an Entity relationship style diagram of the physical database design is sufficient. References to message format standards may be sufficient for message-oriented implementations. Descriptions of file formats may be used when file passing is the model used to exchange information. Interoperating systems may use a variety of techniques to exchange system data and have several distinct partitions in their DIV-3 with each partition using a different form.

Standards associated with entities are also often identified in the development of the DIV-3; these should be recorded in the StdV-1 Standards Profile. Structural Assertions – these involve static aspects of business rules – are best captured in the DIV-3.

Possible Construction Methods: DoDAF does not endorse a specific data modeling methodology. The physical data schema model specifies how the logical data model will be instantiated. The most predominant are the relational database management systems and object repository products. In addition, this model may employ other technology mechanisms, such as messages or flat files. The essential elements of a physical data schema model (in the case of a relational database) are: tables, records and keys. In an object-oriented data model, all data elements are expressed as objects; whether they are classes, instances, attributes, relationships, or events.

The appropriate way to develop a physical data model depends on the product chosen to instantiate the logical data model (e.g., a relational database management system [RDBMS]). A physical data schema model seems best described using an entity-relationship diagramming technique. For Object-Oriented data modeling, a physical data schema seems best described using by Class and/or Object diagrams. For other implementation technologies, such as message orientation, a reference to a message format standard might be more appropriate.

3.1.4 Operational Viewpoint

DoDAF-described Models in the Operational Viewpoint describe the tasks and activities, operational elements, and resource flow exchanges required to conduct operations. A pure operational model is materiel independent. However, operations and their relationships may be influenced by new technologies, such as collaboration technology, where process improvements are in practice before policy can reflect the new procedures. There may be some cases, as well, in which it is necessary to document the way activities are performed, given the restrictions of current systems, to examine ways in which new systems could facilitate streamlining the activities. In such cases, operational models may have materiel constraints and requirements that need to be addressed. For this reason, it may be necessary to include some high-level system architectural data to augment information onto the operational models.

Use of Operational Viewpoint DoDAF-described Models should improve the quality of requirements definitions by:

- Explicitly tying user requirements to strategic-level capability needs, enabling early agreement to be reached on the capability boundary.
- Providing a validated reference model of the business/operations against which the completeness of a requirements definition can be assessed (visualization aids validation).
- Explicitly linking functional requirements to a validated model of the business or operations activities.
- Capturing information-related requirements (not just Information Exchange Requirements [IERs]) in a coherent manner and in a way that really reflects the user collaboration needs.
- Providing a basis for test scenarios linked to user requirements.
- Capturing the activities for Process Engineering or Process Re-engineering.

Names of the models and their descriptions (in **Table 3.1.4-1**) are provided below.

Table 3.1.4-1: Operational Model Descriptions

Model	Description
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.
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.

Table 3.1.4-1: Operational Model Descriptions

Model	Description
OV-6b: State Transition Description	One of three models used to describe operational activity (activity). It identifies business process (activity) responses to events (usually, very short activities).
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.

Mappings of the Operational Viewpoint DoDAF-described Models to the DM2 Concepts, Associations, and Attributes are in **Table B-1 DM2 Concepts, Associations, and Attributes Mapping to DoDAF-described Models**. The DM2 Concepts, Associations, and Attributes are described in the DoDAF Meta-model Data Dictionary.

3.1.4.1 Uses of Operational Viewpoint DoDAF-described Models. The OV DoDAF-described Models may be used to describe a requirement for a "To-Be" architecture in logical terms, or as a simplified description of the key behavioral and information aspects of an "As-Is" architecture. The OV DoDAF-described Models re-use the capabilities defined in the Capability Viewpoint and put them in the context of an operation or scenario. The OV DoDAF-described Models can be used in a number of ways, including the development of user requirements, capturing future concepts, and supporting operational planning processes.

One important way that architectural modeling supports the definition of requirements is in terms of boundary definition. Boundary definition is a process that often requires a significant degree of stakeholder engagement; the described models provided by DoDAF provide ideal support for this interactive process. The DoDAF provides support to the concept of functional scope and organizational span. When performing analysis of requirements relative to a particular capability or capabilities, it is important to know the specific functionality intended to be delivered by the capability. It is also important to know the limits of that functionality, to be able to determine necessary interfaces to other capabilities and organizations. The use of OV DoDAF-described Models (e.g., Operational Resource Flow Description and Operational Activity Model) supports identification of the boundaries of capabilities, thus rendering the functional scope and organizational span.

Definition of user-level interoperability requirements is another use for which there is applicability of the Operational Viewpoint DoDAF-described Models. Within the Operational Viewpoint DoDAF-described Models, DoDAF supports interoperability analysis in a number of ways.

Operational models can be used to help answering questions such as:

- What are the lines of business supported by this enterprise?
- What activities are in place to support the different lines of business?
- What is the functional scope of the capability or capabilities for which I am responsible? This can be answered by a combination of information from the activity model and CV DoDAF-described Models.
- What is the organizational span of influence of this capability or capabilities?
- What information must be passed between capabilities?

- What strategic drivers require that certain data are passed or tracked? This can be answered by a combination of data within the logical data model, information exchanges, activities, and CV DoDAF-described Models.
- What activities are being supported or automated by a capability or capabilities?
- What role does organization X play within a capability or capabilities?
- What are the functional requirements driving a particular capability?
- What rules are applied within a capability, and how are they applied?

3.1.4.2 Model Descriptions. The OV DoDAF-described Models are described below. In addition, examples of these models can be viewed online in the public DoDAF Journal.

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

Detailed Description:

Each Operational Viewpoint relates to a specific point within the Enterprise's timeline. The OV-1 describes a mission, class of mission, or scenario. The purpose of OV-1 is to provide a quick, high-level description of what the architecture is supposed to do, and how it is supposed to do it. An OV-1 can be used to orient and focus detailed discussions. Its main utility is as a facilitator of human communication, and it is intended for presentation to high-level decision-makers. An OV-1 identifies the mission/scope covered in the Architectural Description. OV-1 conveys, in simple terms, what the Architectural Description is about and an idea of the players and operations involved.

The content of an OV-1 depends on the scope and intent of the Architectural Description, but in general it describes the business activities or missions, high-level operations, organizations, and geographical distribution of assets. The model frames the operational concept (what happens, who does what, in what order, to accomplish what goal) and highlight interactions to the environment and other external systems. However, the content is at an executive summary-level as other models allow for more detailed definition of interactions and sequencing.

It may highlight the key Operational concepts and interesting or unique aspects of the concepts of operations. It provides a description of the interactions between the Architectural Description and its environment, and between the Architectural Description and external systems. A textual description accompanying the graphic is crucial. Graphics alone are not sufficient for capturing the necessary architectural data.

The OV-1 consists of a graphical executive summary for a given Architectural Description with accompanying text.

During the course of developing an Architectural Description, several versions of an OV-1 may be produced. An initial version may be produced to focus the effort and illustrate its scope. After other models within the Architectural Description's scope have been developed and verified, another version of the OV-1 may be produced to reflect adjustments to the scope and other Architectural Description details that may have been identified as a result of the architecture development. After the Architectural Description has been used for its intended purpose and the appropriate analysis has been completed, yet another version may be produced to summarize these findings to present them to high-level decision-makers. In other cases, OV-1 is the last model to be developed, as it conveys summary information about the whole Architectural Description for a given scenario.

The OV-1 is useful in establishing the context for a suite of related operational models. This context may be in terms of phase, a time period, a mission and/or a location. In particular, this provides a container for the spatial-temporally constrained performance parameters (measures).

To describe this, the operational performance measures for desert warfare in Phase 1 may be different to those in Phase 2. The measures for jungle warfare in Phase 2 may be different to those for desert warfare in Phase 2.

The context may also explicitly involve a Mission. When the subject of the Architectural Description is a business capability rather than a battlespace capability, then some adjustment is needed in the use of terminology. However, the idea of having a high-level (Business) operational concept still makes sense and the graphical representation in OV-1 adds value to the more structured models that may be created.

OV-1 is the most general of the architectural models and the most flexible in format. However, an OV-1 usually consists of one or more graphics (or possibly a video-clip), as needed, as well as explanatory text.

3.1.4.2.2 OV-2: Operational Resource Flow 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 materiel in addition to information. A specific application of the OV-2 is to describe a logical pattern of resource (information, funding, personnel, or materiel) 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.

Detailed Description:

The OV-2 depicts Operational Needlines that indicate a need to exchange resources. New to DoDAF V2.0, the OV-2 show flows of funding, personnel and materiel in addition to information. The OV-2 may also show the location of Operational facilities or locations, and may optionally be annotated to show flows of information, funding, people or materiel between Operational Activities. The Operational Activities shown in an OV-2 may be internal to the architecture, or may be external activities that communicate with those internal activities.

Use of OV-2 is intended to be logical. It is to describe *who* or *what*, not *how*. This model provides a focus for the operational requirements which may reflect any capability requirements that have been articulated but within the range of operational settings that are being used for operational architecture. In an “As-Is” architecture, an OV-2 may be used as an abstract (i.e., simplified) representation of the Resource Flows taking place in the Enterprise. An OV-2 can be a powerful way of expressing the differences between an “As-Is” Architectural Description and a proposed “To-Be” Architectural Description to non-technical stakeholders, as it simply shows how Resource Flows (or does not flow). The aim of the OV-2 is to record the operational characteristics for the community of anticipated users relevant to the Architectural Description and their collaboration needs, as expressed in Needlines and Resource Flows.

A specific application of the OV-2 is to describe a logical pattern of resource (information, funding, personnel, or materiel) flows. The purpose of an OV-2 model is to describe a logical pattern of Resource 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 OV-2 is intended to track the need for Resource Flows between specific Operational Activities and Locations that play a key role in the Architectural Description. OV-2 does not depict the physical connectivity between the Activities and Locations. The logical pattern established in an OV-2 model may act as the backbone onto which architectural elements may be overlaid – e.g., a SV-1 Systems Interface Description model can show which systems are providing the necessary capability.

The main features of this model are the Operational Resource Flows, and the location (or type of location/environment) where the resources need to be or are deployed, and the Needlines that indicate a need to exchange or share resources. An OV-2 indicates the key players and the interactions necessary to conduct the corresponding operational activities of OV-5a Operational Activity Decomposition Tree or OV-5b Operational Activity Model.

A Needline documents the required or actual exchanges of resources. A Needline is a conduit for one or more resource exchanges – i.e., it represents a logical bundle of Resource Flows. The Needline does not indicate how the transfer is implemented. For example, if information (or funding, personnel, or materiel) is produced at location A, routed through location B, and is used at location C, then location B would not be shown on the OV-2 – the Needline would go from Location A to Location C. The OV-2 is not a communications link or communications network diagram but a high-level definition of the logical requirement for resource exchange.

A OV-2 can also define a need to exchange items between Operational Activities and locations and external resources (i.e., Operational Activities, Locations, or Organizations that are not strictly within the scope of the subject Architectural Description but which interface to it either as important sources of items required within the Architectural Description or important destinations for items provided within the Architectural Description).

The OV-2 is intended to track the need to exchange items between key Operational Activities and Locations within the Architectural Description. The OV-2 does not depict the physical connectivity between the Operational Activities and Locations. The Needlines established in an OV-2 can be realized by resources and their interactions in a SV-1 Systems Interface Description model or SvcV-1 Services Context Description model. There may not be a one-to-one correspondence between an operational activity and a location in OV-2 and a resource in SV-1 Systems Interface Description model or SvcV-1 Services Context Description model. For example, an Operational Activity and location may be realized by two systems, where one provides backup for the other, or it may be that the functionality of an Operational Activity has to be split between two locations for practical reasons.

Needlines can be represented by arrows (indicating the direction of flow) and are annotated with a diagram-unique identifier and a phrase that is descriptive of the principal type of exchange – it may be convenient to present these phrases (or numerical labels) in a key to the diagram to prevent cluttering. It is important to note that the arrows (with identifiers) on the diagram represent Needlines only. This means that each arrow indicates only that there is a need for the transfer of some resource between the two connected Activities or locations. A Needline can be uni-directional. Because Needline identifiers are often needed to provide a trace reference for Resource Flow requirements (see OV-3 Operational Resource Flow Matrix), a combined approach, with numerical and text labels, can be used.

There may be several Needlines (in the same direction) from one resource to another. This may occur because some Needlines are only relevant to certain scenarios, missions or mission phases. In this case, when producing the OV-2 for the specific case, a subset of all of the Needlines should be displayed. There can be a one-to-many relationship from Needlines to Resource Flow (e.g., a single Needline in OV-2 represents multiple individual Resource Flows). The mapping of the Resource Flows to the Needlines of OV-2 occurs in the Operational Resource Flow Matrix (OV-3). For example, OV-2 may list Situation Report as a descriptive name for a Needline between two Operational resources. In this case, the Needline represents a number of resource flow (information in this case) exchanges, consisting of various types of reports (information elements), and their attributes (such as periodicity and timeliness) that are associated with the Situation Report Needline. The identity of the individual elements and their attributes are documented in OV-3 Operational Resource Flow Matrix model.

For complex Architectural Descriptions, OV-2 may consist of multiple graphics. There are several different ways to decompose OV-2. One method involves using multiple levels of abstraction and decomposing the Resource Flows. Another method involves restricting the Resource Flows and Needlines on any given graphic to those associated with a subset of operational activities. Finally it is possible to organize OV-2 in terms of scenarios, missions or mission phases. All of these methods are valid and can be used together.

Flows of Funding, Personnel and Material:

In addition to Needlines, Resource Flow Connectors can be used to overlay contextual information about how the Operational Activities and Locations interact via physical flows. This information helps to provide context for the business roles. Examples of Resource Flow Connector usage would be:

- Representing a logistics capability may have an interaction which involves supplying (physically delivering) personnel.
- Representing an air-to-air refueling capability may have an interaction with airborne platform capabilities which involves transfer of fuel.
- Representing a sensor capability may have an interaction with a target through a flow of physical energy that is sensed; this is not an information flow.

This is achieved by overlaying the Resource Flow Connectors on the diagram using a notation that is clearly distinct from Needlines (which only represent the requirement to flow resources).

Operational Activities:

The operational activities (from the OV-5b Operational Activity Model) performed may be listed on the graphic, if space permits. OV-2 and the OV-5b Operational Activity Model are complementary descriptions. OV-2 focuses on the Operational Resource Flows, with the activities being a secondary adornment. The OV-5b, on the other hand, places first-order attention on operational activities and only second-order attention on Resource Flows, which can be shown as annotations or swim lanes on the activities. In developing an Architectural Description, OV-2 and OV-5b Operational Activity Model are often the starting points and these may be developed iteratively.

3.1.4.2.3 OV-3: Operational Resource Flow Matrix. 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.

Detailed Description:

The OV-3 identifies the resource transfers that are necessary to support operations to achieve a specific operational task. This model is initially constructed from the information contained in the OV-2 Operational Resource Flow Description model. But the OV-3 provides a more detailed definition of the Resource Flows for operations within a community of anticipated users.

The Operational Resource Flow Matrix details Resource Flow exchanges by identifying which Operational Activity and locations exchange what resources, with whom, why the resource is necessary, and the key attributes of the associated resources. The OV-3 identifies resource elements and relevant attributes of the Resource Flows and associates the exchange to the producing and consuming Operational Activities and locations and to the Needline that the Resource Flow satisfies. OV-3 is one of a suite of operational models that address the resource content of the operational architecture (the others being OV-2 Operational Resource Flow Description, OV-5b Operational Activity Model, and DIV-2 Logical Data Model). Needlines are logical requirements-based collaboration relationships between Operational Activities and locations (as shown in OV-2 Operational Resource Flow Description). A Needline can be uni-directional.

A resource element (see DIV-2 Logical Data Model) is a formalized representation of Resource Flows subject to an operational process. Resource elements may mediate activity flows and dependencies (see OV-5b Operational Activity Model). Hence they may also be carried by Needlines that express collaboration relationships. The same resource element may be used in one or more Resource Flows.

The emphasis in this model is on the logical and operational characteristics of the Resource Flows being exchanged, with focus on the Resource Flows crossing the capability boundary. It is important to note that OV-3 is not intended to be an exhaustive listing of all the details contained in every Resource Flow of every Operational Activity and location associated with the Architectural Description in question. Rather, this model is intended to capture the most important aspects of selected Resource Flows.

The aspects of the Resource Flow that are crucial to the operational mission will be tracked as attributes in OV-3. For example, if the subject Architectural Description concerns tactical battlefield targeting, then the timeliness of the enemy target information is a significant attribute of the Resource Flow. To support the needs of security architecture, Resource Flows should also address criticality and classification. There is an important caveat on use of OV-3 for security architectures. In that context, it is important to identify every possible and required exchange.

There is not always a one-to-one mapping of OV-3 Resource Flows to OV-2 Operational Resource Flow Description Needlines; rather, many individual Resource Flows may be associated with one Needline.

The OV-3 information can be presented in tabular form. DoDAF V2.0 does not prescribe the column headings in an OV-3 Matrix.

3.1.4.2.4 OV-4: Organizational Relationships Chart. The OV-4 shows organizational structures and interactions. The organizations shown may be civil or military. 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 particular 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.

The intended usage of the actual OV-4 includes:

- Identify architecture stakeholders.
- Identify process owners.
- Illustrate current or future organization structures.

Detailed Description:

The OV-4 addresses the organizational aspects of an Architectural Description. A typical OV-4 illustrates the command structure or relationships (as opposed to relationships with respect to a business process flow) among human roles, organizations, or organization types that are the key players in the business represented by the architecture. An actual OV-4 shows real organizations and the relationships between them.

The more commonly used types of organizational relationship will be defined, in time, in the DoDAF Meta-model. DoDAF defines fundamental relationships between Organizational Resources; including structure (whole-part) and interaction. The interaction relationship covers most types of organizational relationship. An OV-4 clarifies the various relationships that can exist between organizations and sub-organizations within the Architectural Description and between internal and external organizations. Where there is a need for other types of organizational relationships, these should be recorded and defined in the AV-2 Integrated Dictionary as extensions to the DM2.

Organizational relationships are important to depict in an architecture model, because they can illustrate fundamental human roles (e.g., who or what type of skill is needed to conduct operational activities) as well as management relationships (e.g., command structure or relationship to other key players). Also, organizational relationships are drivers for some of the collaboration requirements that are viewed using Needlines.

Note that individual people are not viewed in DoDAF, but specific billets or Person Types may be detailed in an actual OV-4.

In both the typical and specific cases, it is possible to overlay resource interaction relationships which denote relationships between organizational elements that are not strictly hierarchical (e.g., a customer-supplier relationship).

The organizations that are modeled using OV-4 may also appear in other models, for example in the SV-1 Systems Interface Description as organizational constituents of a capability or a resource and PV-1 Project Portfolio Relationships where organizations own projects. In a SV-1 Systems Interface Description, for instance, the organizational resources defined in a typical OV-4 may be part of a capability or resources. Also, actual organizations may form elements of a fielded capability which realizes the requirements at the system-level (again, this may be depicted on a SV-1 Systems Interface Description).

A OV-4 may show types of organizations and the typical structure of those organizations. The OV-4 may alternatively show actual, specific organizations (e.g., the DoD) at some point in time. Alternatively, an OV-4 may be a hybrid diagram showing typical and actual organization structures.

3.1.4.2.5 OV-5a: Operational Activity Decomposition Tree and OV-5b: Operational Activity Model. 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 interactions (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.

The OV-5b describes the operational, business, and defense portion of the intelligence community 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.

Detailed Description:

The OV-5s and OV-2 Operational Resource Flow Description model are, to a degree, complements of each other. The OV-5s focuses on the operational activities whereas OV-2 Operational Resource Flow Description model focuses on the operational activities in relation to locations. Due to the relationship between locations and operational activities, these types of models should normally be developed together. An OV-5a or OV-5b describes the operational activites (or tasks) that are normally conducted in the course of achieving a mission or a business goal. The OV-5b also describes Input/Output flows between activities, and to/from activities that are outside the scope of the Architectural Description. The OV-5a and OV-5b are equally suited to describing non-military activities and are expected to be used extensively for business modeling.

The activities described in an OV-5a or OV-5b are standard Operational Activities which are mapped to corresponding capabilities in the CV-6 Capability to Operational Activities Mapping. Standard Operational Activities are those defined in doctrine, but which are not tailored to a specific system, i.e., they are generic enough to be used without closing off a range of possible solutions.

Possible Construction Methods: DoDAF does not endorse a specific activity modeling methodology. The OV-5b can be constructed using Integration Definition for Function Modeling (IDEF0) or Class Diagrams.

There are two basic ways to depict Activity Models:

- The Activity Decomposition Tree shows activities depicted in a tree structure and is typically used to provide a navigation aid.
- The Activity Model shows activities connected by Resource Flows; it supports development of an OV-3 Operational Resource Flow Matrix.

The OV-5a helps provide an overall picture of the activities involved and a quick reference for navigating the OV-5b.

3.1.4.2.6 Introduction to OV-6a, OV-6b and OV-6c. OV Models discussed in previous sections model the static structure of the Architectural elements and their relationships. Many of the critical characteristics of an architecture are only discovered when the dynamic behavior of these elements is modeled to incorporate sequencing and timing aspects.

The dynamic behavior referred to here concerns the timing and sequencing of events that capture operational behavior of a business process or mission thread. Thus, this behavior is related to the activities of OV-5b. Behavior modeling and documentation is essential to a successful Architectural Description, because it describes how the architecture behaves and that is crucial in many situations. Knowledge of the Operational Activities and Resource Flow exchanges is important; but knowing when, for example, a response should be expected after sending message X to Activity Y at Location A can also be essential to achieving successful operations.

Several modeling techniques may be used to refine and extend the Architectural Description's OV to adequately describe the dynamic behavior and timing performance characteristics of an architecture. The OV-6 DoDAF-described Models includes three such models. They are:

- Operational Rules Model (OV-6a).
- Operational State Transition Description (OV-6b).
- Operational Event-Trace Description (OV-6c).

OV-6 DoDAF-described Models portray some of the same architectural data elements, but each also portrays some unique architectural data elements. OV-6b and OV-6c may be used separately or together, as necessary, to describe critical timing and sequencing behavior in the OV. Both types of models are used by a wide variety of business process methodologies as well as Object-Oriented methodologies. OV-6b and OV-6c describe Operational Activity or business process responses to sequences of events. Events may also be referred to as inputs, transactions, or triggers. Events can be internally or externally generated and can include such things as the receipt of a message, a timer going off, or conditional tests being satisfied. When an event occurs, the action to be taken may be subject to a rule or set of rules (conditions) as described in OV-6a.

3.1.4.2.6.1 OV-6a: Operational Rules Model. 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.

Detailed Description:

The OV-6a specifies operational or business rules that are constraints on the way business is done in the enterprise. While other OV Models (e.g., OV-1 High Level Operational Concept Graphic, OV-2 Operational Resource Flow Description, and OV-5b Operational Activity Model) describe the structure and operation of a business, for the most part they do not describe the constraints and rules under which it operates.

At the mission-level, OV-6a may be based on business rules contained in doctrine, guidance, rules of engagement, etc. At lower levels, OV-6a describes the rules under which the architecture behave under specified conditions. Such rules can be expressed in a textual form, for example, If (these conditions) exist, and (this event) occurs, then (perform these actions). These rules are contrasted with the business or doctrinal standards themselves, which provide authoritative references and provenance for the rules (see StdV-1 Standards Profile). Operational Rules are statements that constrain some aspect of the mission or the architecture. Rules may be expressed in natural language (English) in one of two forms:

- Imperative – a statement of what shall be under all conditions, e.g., “Battle Damage Assessment (BDA) shall only be carried out under fair weather conditions.”
- Conditional Imperative – a statement of what shall be, in the event of another condition being met. If battle damage assessment shows incomplete strike, then a re-strike shall be carried out.

As the model name implies, the rules captured in OV-6a are operational (i.e., mission-oriented) whereas resource-oriented rules are defined in the SV-10s or the SvcV-10s (OV-6 is the *what* to the SV-10’s or SvcV-10’s *how*). OV-6a rules can include such guidance as the conditions under which operational control passes from one entity to another or the conditions under which a human role is authorized to proceed with a specific activity.

A rule defined in textual form OV-6a may be applied to any Architectural element defined in an OV. A rule defined in a more structured way (i.e., for the purposes of sharing with other architects) should be defined in association with locations, operational activities or missions.

Rules defined in an OV-6a may optionally be presented in any other OV. For example, a rule “battle damage assessment shall be carried out under fair weather conditions” may be linked to the Conduct BDA activity in OV-5b. Any natural language rule presented (e.g., in a diagram note) should also be listed in OV-6a.

OV-6a rules may be associated with activities in OV-5a Operational Activity Decomposition Tree and OV-5b Operational Activity Model and can be useful to overlay the rules on an OV-5a Operational Activity Decomposition or OV-5b Operational Activity Model. OV-6a can also be used to extend the capture of business requirements by constraining the structure and validity of DIV-2 Logical Data Model elements.

Detailed rules can become quite complex, and the structuring of the rules themselves can often be challenging. DoDAF does not specify how OV-6a rules will be specified, other than in English.

From a modeling perspective, operational constraints may act upon Locations, Operational Activities, Missions, and Entities in Logical Data Models.

3.1.4.2.6.2 OV-6b: State Transition 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.

Detailed Description:

The OV-6b reflects the fact that the explicit sequencing of activities in response to external and internal events is not fully expressed in OV-5a Operational Activity Decomposition Tree or OV-5b Operational Activity Model. Alternatively, OV-6b can be used to reflect the explicit sequencing of actions internal to a single Operational Activity or the sequencing of operational activities. OV-6b is based on the statechart diagram. A state machine is defined as "a specification that describes all possible behaviors of some dynamic view element. Behavior is viewed as a traversal of a graph of state interconnected by one or more joined transition arcs that are triggered by the dispatching of a series of event instances. During this traversal, the state machine executes a series of actions associated with various elements of the state machine."

State chart diagrams can be unambiguously converted to structured textual rules that specify timing aspects of operational events and the responses to these events, with no loss of meaning. However, the graphical form of the state diagrams can often allow quick analysis of the completeness of the rule set, and detection of dead ends or missing conditions. These errors, if not detected early during the operational analysis phase, can often lead to serious behavioral errors in fielded systems or to expensive correction efforts.

States in an OV-6b may be nested. This enables quite complex models to be created to represent operational behavior.

3.1.4.2.6.3 OV-6c: Event-Trace 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 particular 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.

Detailed Description:

The OV-6c is valuable for moving to the next level of detail from the initial operational concepts. An OV-6c model helps define interactions and operational threads. The OV-6c can also help ensure that each participating Operational Activity and Location has the necessary information it needs at the right time to perform its assigned Operational Activity.

The OV-6c enables the tracing of actions in a scenario or critical sequence of events. OV-6c can be used by itself or in conjunction with OV-6b State Transition Description to describe the dynamic behavior of business activities or a mission/operational thread. An operational thread is defined as a set of operational activities, with sequence and timing attributes of the activities, and includes the resources needed to accomplish the activities. A particular operational thread may be used to depict a military or business capability. In this manner, a capability is defined in terms of the attributes required to accomplish a given mission objective by modeling the set of activities and their attributes. The sequence of activities forms the basis for defining and understanding the many factors that impact on the overall capability.

The information content of messages in an OV-6c may be related with the Resource Flows in the OV-3 Operational Resource Flow Matrix and OV-5b Operational Activity Model and information entities in the DIV-2 Logical Data Model.

Possible Construction Methods: DoDAF does not endorse a specific event-trace modeling methodology. An OV-6c may be developed using any modeling notation (e.g., BPMN) that supports the layout of timing and sequence of activities along with the Resource Flow exchanges that occur between Operational Activities/Locations for a given scenario. Different scenarios can be depicted by separate diagrams.

3.1.5 Project Viewpoint

The DoDAF-described Models within the Project Viewpoint describe how programs, projects, portfolios, or initiatives deliver capabilities, the organizations contributing to them, and dependencies between them. Previous versions of DoDAF took a traditional model of architecture in which descriptions of programs and projects were considered outside scope. To compensate for this, various DoDAF models represented the evolution of systems, technologies and standards (e.g., Systems and Services Evolution Description, Systems Technology Forecast, and Technical Standards Forecast).

The integration of Project Models (organizational and project-oriented) with the more traditional architecture models is a characteristic aspect of DoDAF V2.0-based enterprise Architectural Descriptions. These models expand the usability of the DoDAF by including information about programs, projects, portfolios, or initiatives and relating that information to capabilities and other programs, projects, portfolios, or initiatives thus expanding DoDAF's support to the portfolio management (PfM) process. Different levels of cost data can be captured in the architecture, based on the Process-owners requirements. An example is a Work Breakdown Structure, depicted as a Gantt chart.

Names of the models and their descriptions (in **Table 3.1.5-1**) are provided below.

Table 3.1.5-1: Project Model Descriptions

Models	Descriptions
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 programs and projects to capabilities to show how the specific projects and program elements help to achieve a capability.

Mappings of the Project Viewpoint DoDAF-described Models to the DM2 Concepts, Associations, and Attributes are in **Table B-1 DM2 Concepts, Associations, and Attributes Mapping to DoDAF-described Models**. The DM2 Concepts, Associations, and Attributes are described in the DoDAF Meta-model Data Dictionary.

3.1.5.1 Uses of Project Viewpoint DoDAF-described Models. As stated above, the Project Viewpoint DoDAF-described Models contain information that improves DoDAF's support to the portfolio management process. It is important to be able to look across portfolios (i.e., groups of investments) to ensure that all possible alternatives for a particular decision have been exhausted to make the most informed decision possible in support of the Department. Relating project information to the responsible organizations, as well as to other projects, forms a valuable architecture construct that supports PfM.

Incorporation of these models also makes the DoDAF a value-added framework to support the PPBE process. These models are especially applicable to the Programming Phase of the PPBE process. It is within this phase that the Program Objective Memorandum (POM) is developed. The POM seeks to construct a balanced set of programs that respond to the guidance and priorities of the Joint Programming Guidance within fiscal constraints. When completed, the POM provides a fairly detailed and comprehensive description of the proposed programs, which can include a time-phased allocation of resources (personnel, funding, materiel, and information) by program projected into the future. The information captured within the Project models (e.g., project relationships, timelines, capabilities) can be used within the PPBE process to develop the POM. Using these models enables decision-makers to perform well-informed planning and complements the use of the Capability Models.

The Project Models can be used to answer questions such as:

- What capabilities are delivered as part of this project?
- Are there other projects that either affect or are affected by this project? To what portfolios do the projects or projects belong?

- What are the important milestones relative to this project? When can I expect capabilities to be rendered by this project to be in place?

3.1.5.2 Model Descriptions. The Project Viewpoint DoDAF-described Models are described below. In addition, examples of these models can be viewed online in the public DoDAF Journal.

3.1.5.2.1 PV-1: Project Portfolio Relationships. 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 organizational 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.

Detailed Description:

The PV-1 describes how acquisition projects are grouped in organizational terms as a coherent portfolio of acquisition programs or projects, or initiatives related to several portfolios. The PV-1 provides a way of describing the organizational relationships between multiple acquisition projects or portfolios, each of which are responsible for delivering individual systems or capabilities. By definition, this model covers acquisition portfolios or programs consisting of multiple projects and is generally not for an individual project. In essence, PV-1 is an organizational breakdown consisting of actual organizations (see OV-4 Organizational Relationships Chart model). The model is strongly linked with the CV-4 Capability Dependencies model which shows capability groupings and dependencies.

The PV-1 is hierarchical in nature. Higher-level groupings of projects (the organizations that own these projects) form acquisition programs or initiatives.

The intent of a PV-1 is to show:

- All of the acquisition projects delivering services, systems, or SoS within the acquisition programs under consideration.
- Cross-cutting initiatives to be tracked across portfolios.
- Other services, systems, and SoS which may have a bearing on the architecture.
- How the services or systems will be best integrated into an acquisition program.
- The nesting of acquisition programs to form a hierarchy.

A PV-1 is specific to a particular point in the project lifecycle. This may change through time, i.e., the projects may change as new services, systems and capabilities are introduced into the acquisition program. Hence, it is possible that an acquisition program could have more than one PV-1, each showing how the acquisition projects are arranged for relevant periods of time. This is achieved by tying the PV-1 model to a capability phase in the CV-3 Capability Dependencies model.

3.1.5.2.2 PV-2: Project Timelines. 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 DoDD 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.

Detailed Description:

The PV-2 provides an overview of a program or portfolio of individual projects, or initiatives, based on a timeline. Portfolios, Programs, Projects, and Initiatives may be broken into work streams to show the dependencies at a lower-level. For capability-based procurement, these work streams might conveniently be equated with JCA. Sometimes, however, it is more appropriate to consider these acquisition projects in their own right.

Where appropriate, the PV-2 may also summarize, for each of the projects illustrated, the level of maturity achieved across the DoDD 5000.1 Defense Acquisition System policies at each stage of the DAS lifecycle, and the interdependencies between the project stages.

The PV-2 is intended primarily to support the acquisition and fielding processes including the management of dependencies between projects and the integration of DoDD 5000.1 Defense Acquisition System policies to achieve a successfully integrated capability. However, the PV-2 is not limited to the acquisition and fielding processes. The information provided by the Model can be used to determine the impact of either planned or unplanned programmatic changes, and highlight opportunities for optimization across the delivery program. The inclusion of the DoDD 5000.1 Defense Acquisition System policy information allows areas of concern that are outside the immediate scope being considered. Areas of concern identified across the DoDD 5000.1 Defense Acquisition System policies, e.g., a shortfall in training resource, can be coordinated across a program or group of projects, each of which require additional activity to be initiated for successfully delivery according to the project/program schedule.

Although a PV-2 may be compiled for a single system project, with supporting work streams, the model becomes particularly useful when considering the dependencies between the multiple projects (or increments within them) that contribute to an acquisition program. Such an acquisition program may be an oversight organization or any other useful grouping of projects that have strong dependencies or contribute towards a common goal (see CV-1 Vision model). Typical use of PV-2 is to represent an individual system development for use in the CV-3 Capability Phasing, while an Integrated Product Team (IPT) may be delivering several systems simultaneously. While PV-2 is expected to support acquisition management for a program consisting of a portfolio of acquisition projects, it may sometimes be convenient to use a PV-2 timeline model for other purposes, e.g., to show temporal relationships between transformation initiatives at the strategic-level or for technology roadmapping.

A PV-2 graphically displays the key milestones and interdependencies between the multiple projects that constitute a program, portfolio, or initiative. Use of PV-2 should support the management of capability delivery and be aligned with the CV-3 Capability Phasing model, if one exists. One presentational format for a PV-2 can be a Gantt chart that displays the entire lifecycle of each project, together with dependencies between them.

Optionally, the Gantt chart may be enhanced to show the level of maturity for each of the DOTMLPF factors associated with that project at each key milestone. The colored icon can be a segmented circular pie chart, a regular polyhedron or any appropriate graphic, providing that the graphic is explained and covers all DAS requirements.

3.1.5.2.3 PV-3: Project to Capability Mapping. 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 particular 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.

Detailed Description:

The PV-3 describes the mapping between capabilities and the programs, projects, portfolios, or initiatives that would support the capabilities. This model may be used to indicate that a project does or does not fulfill the requirements for a capability for a particular phase.

This model is analogous to the SV-5a Operational Activity to System Function Traceability Matrix, but provides the interface between Capability and Project Models rather than Operational to System Models.

In principle, there could be a different PV-3 table created for each development phase of the program, project, portfolio, or initiative development, or perhaps for different phasing scenarios. In most cases, a single table can be constructed because the programs, projects, portfolios, or initiatives that are most likely relevant to this model can be relatively high-level. If capabilities associated are generic (see CV-1 Vision model), then they should have a well understood relationship with a set of programs, projects, portfolios, or initiatives and this relationship is unlikely to change over time.

The PV-3 can have a tabular presentation. The rows can be the Capabilities and the columns can be the programs, projects, portfolios, or initiatives. An X can indicate where the capability is supported by the programs, projects, portfolios, or initiatives whereas a blank can indicate that it does not. Alternatively, a date or phase can indicate when programs, projects, portfolios, or initiatives will support capabilities by the date or phase indicated.

3.1.6 Services Viewpoint

The DoDAF-described Models within the Services Viewpoint describes services and their interconnections providing or supporting, DoD functions. DoD functions include both warfighting and business functions. The Service Models associate service resources to the operational and capability requirements. These resources support the operational activities and facilitate the exchange of information. The relationship between architectural data elements across the Services Viewpoint to the Operational Viewpoint and Capability Viewpoint can be exemplified as services are procured and fielded to support the operations and capabilities of organizations. The structural and behavioral models in the OVs and SvcVs allow architects and stakeholders to quickly ascertain which functions are carried out by humans and which by Services for each alternative specification and so carry out trade analysis based on risk, cost, reliability, etc.

Services are not limited to internal system functions and can include Human Computer Interface (HCI) and Graphical User Interface (GUI) functions or functions that consume or produce service data to or from service functions. The external service data providers and consumers can be used to represent the human that interacts with the service.

Names of the models and their descriptions (in **Table 3.1.6-1**) are provided below.

Table 3.1.6-1: Service Model Descriptions

Models	Descriptions
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 Operational Activity to Services Traceability Matrix	A mapping of services (activities) back to operational activities (activities).
SvcV-6 Services Resource Flow Matrix	It provides details of service Resource Flow elements being exchanged between services and the attributes of

Table 3.1.6-1: Service Model Descriptions

Models	Descriptions
	that exchange.
SvcV-7 Services Measures Matrix	The measures (metrics) of Services Model elements for the appropriate timeframe(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 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.

Mappings of the Services Viewpoint DoDAF-described Models to the DM2 Concepts, Associations, and Attributes are in **Table B-1 DM2 Concepts, Associations, and Attributes Mapping to DoDAF-described Models**. The DM2 Concepts, Associations, and Attributes are described in the DoDAF Meta-model Data Dictionary.

3.1.6.1 Uses of Services Viewpoint DoDAF-described Models. Within the development process, the service models describe the design for service-based solutions to support operational requirements from the development processes (JCIDS) and Defense Acquisition System or capability development within the JCAs.

Some of the Services Viewpoint DoDAF-described Models are discussed with examples in the DoDAF Product Development Questionnaire Analysis Report.doc. This document can be viewed online in the public DoDAF Journal.

3.1.6.2 Model Descriptions. The Services Viewpoint DoDAF-described Models are described below. In addition, examples of these models can be viewed online in the public DoDAF Journal.

3.1.6.2.1 SvcV-1: Services Interface Description. The SvcV-1 addresses the composition and interaction of Services. For DoDAF V2.0, SvcV-1 incorporates human elements as types of Performers - Organizations and Personnel Types. Resources are defined in Section 2.2.1.

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

It is important for the architect to recognize that the SvcV-1 focuses on the Resource Flow and the providing service. This differs from a SV-1 System Interface Description which focuses on the System-to-System point-to-point interface, for which the Source System and Target System have an agreed upon interface. For the SvcV-1, the focus on the provider and the data provided is a Net-Centric Data Strategy tenet appropriate for a publish/subscribe pattern. This pattern is not the only type of service that can be captured in the SvcV-1.

Sub-services may be identified in SvcV-1 to any level (i.e., depth) of decomposition the architect sees fit. The SvcV-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 locations depicted in an OV-2 Operational Resource Flow Description may well be the logical representation of the resource that is shown in SvcV-1.

The intended usage of the SvcV-1 includes:

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

The SvcV-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.

Detailed Description:

A SvcV-1 can be used simply to depict services and sub-services and identify the Resource Flows between them. The real benefit of a SvcV-1 is its ability to describe the human aspects of an architecture and how they interact with Services. In addition, DoDAF has the concept of Capability and Performers (see Capability Meta-model group in Section 2) which is used to depict Services, assets and people into a configuration, which can meet a specific capability. A primary purpose of a SvcV-1 model is to show resource structure, i.e., identify the primary sub-services, performer and activities (functions) and their interactions. SvcV-1 contributes to user understanding of the structural characteristics of the solution.

The physical resources contributing to a capability are either an organizational resource or a physical asset, i.e., a service cannot contribute alone (it must be hosted on a physical asset used by an organizational resource of both). Organizational aspects can now be shown on SvcV-1 (e.g., who uses a service). Resource structures may be identified in SvcV-1 to any level (i.e., depth) of decomposition the architect sees fit. DoDAF does not specifically use terms like sub-service and component as these terms often denote a position relative to a structural hierarchy. Any service may combine hardware and software or these can be treated as separate (sub) services. DoDAF V2.0 includes human factors (as Personnel Types and a type of Performer). Should an architect wish to describe a service which has human elements, then groupings of Services, Personnel Types and Performers should be used to wrap the human and service elements together.

A SvcV-1 can optionally be annotated with Operational Activities and Locations originally specified in OV-2 Operational Resource Flow Description. In this way, traceability can be established from the logical OV structure to the physical Service Model structure.

If a single SvcV-1 is not possible, the resource of interest should be decomposed into multiple SvcV-1 models.

Functions (Activities):

Some Resources can carry out service functions (activities) as described in SvcV-4 Services Functionality Description models and these functions can optionally be overlaid on a SvcV-1. In a sense SvcV-1 and SvcV-4 Services Functionality Description provide complementary representations (structure and function). Either could be viewed first, but usually an iterative approach is used to model these together gradually building up the level of detail in the service description. Note that the same type (class) of resource may be used in different contexts in a given SvcV-1. For this reason, the tracing of functions to resources is specified in context of their usage (see DM2 for details).

Resource Flows in SvcV-1:

In addition to depicting Services (Performers) and their structure, SvcV-1 addresses Service Resource Flows. A Service Resource Flow, as depicted in SvcV-1, is an indicator that resources pass between one service and the other. In the case of Services, this can be expanded into further detail in SvcV-2 Services Resource Flow Description model. A Service 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 SvcV-1 depicts all Resource Flows between resources that are of interest. Note that Resource Flows between resources may be further specified in detail in the SvcV-2 Services Resource Flow Description model and the SvcV-6 Services Resource Flow Matrix.

Interactions are only possible between services and systems. Service Resource Flows provide a specification for how the Resource Flow exchanges specified in OV-2 Operational Resource Flow Description Needlines are realized with services. A single Needline shown in the OV-2 Operational Resource Flow Description may translate into multiple Service Resource Flows. The actual implementation of Service Resource Flows may take more than one form (e.g., multiple physical links). Details of the physical pathways or network patterns that implement the interfaces are documented in SvcV-2 Services Resource Flow Description. Resource Flows are summarized in a SvcV-3a System-Service Matrix or SvcV-3b Service-Service Matrix and detailed definitions and attributes specific to each Service Resource Flows may be described in SvcV-6 Services Resource Flow Matrix.

The functions performed by the resources are specified in a SvcV-4 Service Functionality Description, but may optionally be overlaid on the Resources in a SvcV-1.

3.1.6.2.2 SvcV-2: Services Resource Flow Description. A SvcV-2 specifies the Resource Flows between Services and may also list the protocol stacks used in connections.

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

The intended usage of the SvcV-2 includes:

- Resource Flow specification.

Detailed Description:

For a network data service, a SvcV-2 comprises Services, their ports, and the Service Resource Flows between those ports. The SvcV-2 may also be used to describe non-IT type services such as Search and Rescue. The architect may choose to create a diagram for each Service Resource Flow and the producing Service, each Service Resource Flow and consuming Service, or to show all the Service Resource Flows on one diagram, if this is possible.

Each SvcV-2 model can show:

- Which ports are connected.
- The producing Services that the ports belong to.
- The Services that the Service Resource Flows are consumed by.
- The definition of the Service Resource Flow in terms of the physical/logical connectivity and any protocols that are used in the connection.

Note that networks are represented as Services. The architect may choose to show other Services being components of the network, i.e., if they are part of the network infrastructure.

Any protocol referred to in a SvcV-2 diagram needs be defined in the StdV-1 Standards Profile.

3.1.6.2.3 SvcV-3a: Systems-Services Matrix. A SvcV-3a enables a quick overview of all the system-to-service resource interactions specified in one or more SvcV-1 Services Context Description models. The SvcV-3a provides a tabular summary of the system and services interactions specified in the SvcV-1 Services Context Description for the Architectural Description. This model can be useful in support existing systems that are transitioning to provide services. The matrix format supports a rapid assessment of potential commonalities and redundancies (or, if fault-tolerance is desired, the lack of redundancies).

The SvcV-3a can be organized in a number of ways to emphasize the association of system-to-service interactions in context with the architecture's purpose.

The intended usage of the SvcV-3a includes:

- Summarizing system and service resource interactions.
- Interface management.
- Comparing interoperability characteristics of solution options.

Detailed Description:

The SvcV-1 concentrates on Service resources and their interactions, and these are summarized in a SvcV-3a or SvcV-3b. The SvcV-3a DoDAF-described Model can be a useful tool for managing the evolution of solutions and infrastructures, the insertion of new technologies and functionality, and the redistribution of Systems and Services and activities in context with evolving operational requirements.

Depending upon the purpose of the architecture, there could be several SvcV-3a DoDAF-described Models. The suite of SvcV-3a models can be organized in a number of ways (e.g., by domain, by operational mission phase, by solution option) to emphasize the association of groups of resource pairs in context with the Architectural Description's purpose.

The SvcV-3a is generally presented as a matrix, where the System and Services resources are listed in the rows and columns of the matrix, and each cell indicates an interaction between Systems and Services if one exists. Many types of interaction information can be presented in the cells of a SvcV-3a. The resource interactions can be represented using different symbols and/or color coding that depicts different interaction characteristics, for example:

- Status (e.g., existing, planned, potential, de-activated).
- Key interfaces.
- Category (e.g., command and control, intelligence, personnel, logistics).
- Classification-level (e.g., Restricted, Confidential, Secret, Top Secret).
- Communication means (e.g., Rim Loop Interface, Scalable Loop Interface).

DoDAF does not specify the symbols to be used. If symbols are used, a key for the symbols is needed.

3.1.6.2.4 SvcV-3b: Services-Services Matrix. A SvcV-3b enables a quick overview of all the services resource interactions specified in one or more SvcV-1 Services Context Description models. The SvcV-3b provides a tabular summary of the services interactions specified in the SvcV-1 Services Context Description 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). In addition, this model is useful in support of net-centric (service-oriented) implementation of services as an input to the SvcV-10a Services Rules Model, SvcV-10b Services State Transition Description, and SvcV-10c Services Event-Trace Description, implemented as orchestrations of services.

The SvcV-3b can be organized in a number of ways to emphasize the association of service pairs in context with the architecture's purpose. One type of organization is a Service Hierarchy or Taxonomy of Services.

The intended usage of the SvcV-3b includes:

- Summarizing service resource interactions.
- Interface management.
- Comparing interoperability characteristics of solution options.

It is important to note that one usage of the Service-Service Matrix (SvcV-3b) can support a net-centric (service-oriented) implementation in describing the interactions between producing services and consuming services.

Detailed Description:

The SvcV-1 concentrates on Service resources and their interactions, and these are summarized in a SvcV-3a or SvcV-3b. The SvcV-3b can be a useful tool for managing the evolution of solutions and infrastructures, the insertion of new technologies and functionality, and the redistribution of Services and activities in context with evolving operational requirements.

Depending upon the purpose of the architecture, there could be several SvcV-3b DoDAF-described Models. The suite of SvcV-3b DoDAF-described Models can be organized in a number of ways (e.g., by domain, by operational mission phase, by solution option) to emphasize the association of groups of resource pairs in context with the Architectural Description purpose.

The SvcV-3b is generally presented as a matrix, where the Services resources are listed in the rows and columns of the matrix, and each cell indicates an interaction between Services if one exists. There are many types of information that can be presented in the cells of a SvcV-3b. The resource interactions can be represented using different symbols and/or color coding that depicts different interaction characteristics, for example:

- Status (e.g., existing, planned, potential, de-activated).
- Key interfaces.
- Category (e.g., command and control, intelligence, personnel, logistics).
- Classification-level (e.g., Restricted, Confidential, Secret, Top Secret).
- Communication means (e.g., Rim Loop Interface, Scalable Loop Interface).

DoDAF does not specify the symbols to be used. If symbols are used, a key for the symbols is needed.

3.1.6.2.5 SvcV-4: Services Functionality Description. The SvcV-4 DoDAF-described Model addresses human and service functionality.

The primary purpose of SvcV-4 is to:

- Develop a clear description of the necessary data flows that are input (consumed) by and output (produced) by each resource.
- Ensure that the service 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 Services Functionality Description provides detailed information regarding the:

- Allocation of service functions to resources.
- Flow of resources between service functions.

The SvcV-4 is the Services Viewpoint counterpart to the OV-5b Operational Activity Model of the Operational Viewpoint.

The intended usage of the SvcV-4 includes:

- Description of task workflow.
- Identification of functional service requirements.
- Functional decomposition of Services.
- Relate human and service functions.

It is important to note that one usage of the SvcV-4 can support a net-centric (service-oriented) implementation in describing the producing services and consuming services. The Services Functionality Description information can support the registration of services in net-centric (service-oriented) implementation.

Detailed Description:

The SvcV-4 is used to specify the service functionality of resources in the architecture. The SvcV-4 is the behavioral counterpart to the SvcV-1 Services Context Description (in the same way that OV-5b Operational Activity Model is the behavioral counterpart to OV-2 Operational Resource Flow Description).

The scope of this model may be capability wide, without regard to which resources perform which service functions, or it may be resource-specific. Variations may focus on intra- or inter-resource data flows, or may simply allocate service functions to resources.

There are two basic ways to depict a SvcV-4:

- The Taxonomic Service Functional Hierarchy shows a decomposition of service functions depicted in a tree structure and is typically used where tasks are concurrent but dependent, such as a production line, for example.
- The Data Flow Diagram shows service functions connected by data flow arrows and data stores.

Within an Architectural Description, the SvcV-4 document service functions, the Resource Flows between those service functions, the internal system data repositories or service data stores, and the external sources and sinks for the service data flows, but not external to the Architectural Description's scope. They may also show how users behave in relation to those services.

3.1.6.2.6 SvcV-5: Operational Activity to Services Traceability Matrix. The SvcV-5 addresses the linkage between service functions described in SvcV-4 and Operational Activities specified in OV-5a Operational Activity Decomposition Tree or OV-5b Operational Activity Model. The SvcV-5 depicts the mapping of service functions (and, optionally, the capabilities and performers that provide them) to operational activities and thus identifies the transformation of an operational need into a purposeful action performed by a service solution.

During requirements definition, the SvcV-5 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 SvcV-5 includes:

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

Detailed Description:

An SvcV-5 is a specification of the relationships between the set of operational activities applicable to an Architectural Description and the set of service functions applicable to that Architectural Description. The relationship between operational activities and service functions can also be expected to be many-to-many (i.e., one activity may be supported by multiple functions, and one function may support multiple activities). The service functions shown in the SvcV-5 may be those associated with capabilities and performers. More focused SvcV-5 models might be used to specifically trace system functions to operational activities if desired.

DoDAF uses the term Operational Activity in the OVs and the term Service Function in the SVs to refer to essentially the same kind of thing—both activities and service functions are tasks that are performed, accept inputs, and develop outputs. The distinction between an Operational Activity and a Service Function is a question of *what* and *how*. The Operational Activity is a specification of what is to be done, regardless of the mechanism used. A Service Function specifies how a resource carries it out. For this reason, the SvcV-5 is a significant model, as it ties together the logical specification in the OV-5a Operational Activity Decomposition Tree or OV-5b Operational Activity Model with the physical specification of the SvcV-4 Services Functionality Description. Service Functions can be carried out by Resources.

Care should be taken when publishing a SvcV-5 with status information. Any presentation should clearly state the date of publication, so that users can see when status information is old.

The SvcV-5 may be further annotated with Services, Capabilities, Performers executing Activities, and capabilities and performers that conduct the functions.

The SvcV-5 is generally presented as a matrix of the relationship between service functions and activities. The SvcV-5 can show requirements traceability with Operational Activities on one axis of a matrix, the System Functions on the other axis, and with an X, date, or phase in the intersecting cells, where appropriate.

An alternate version of the tabular SvcV-5 can allow the implementation status of each function to be shown. In this variant model, each service function-to-operational activity mapping is described by a traffic light symbol that may indicate the status of the service support. DoDAF V2.0 does not prescribe a presentation technique. These symbols are usually colored circles with the following possible representations:

- Red may indicate that the functionality is planned but not developed.
- Yellow may indicate that partial functionality has been provided (or full functionality provided but system has not been fielded).
- Green may indicate that full functionality has been provided to the field.
- A blank cell may indicate that there is no service support planned for an Operational Activity, or that a relationship does not exist between the Operational Activity and the Service Function.

3.1.6.2.7 SvcV-6: Services Resource Flow Matrix. The SvcV-6 specifies the characteristics of the Service Resource Flows exchanged between Services. The focus is on resource crossing the service boundary. The SvcV-6 focuses on the specific aspects of the Service Resource Flow and the Service Resource Flow content in a tabular format.

In addition, this model is useful in support of net-centric (service-oriented) implementation of services. According to the Net-Centric Data Strategy, a net-centric implementation needs to focus in on the data in the Service Resource Flow, as well as the services that produce or consume the data of the Service Resource Flow. In a net-centric implementation, not all the consumers are known and this model emphasizes the focus on the producer and Service Resource Flow.

The intended usage of the SvcV-6 includes:

- Detailed definition of Resource Flows.

Detailed Description:

The SvcV-6 specifies the characteristics of Service Resource Flow exchanges between Services. The SvcV- is the physical equivalent of the logical OV-3 Operational Resource Flow Matrix and provides detailed information on the service connections which implement the Resource Flow exchanges specified in OV-3 Operational Resource Flow Matrix. Resource flow exchange solutions, whether automated or not, e.g., such as verbal orders, are also captured.

Service Resource Flow exchanges express the relationship across the three basic architectural data elements of a SvcV (Services, service functions, and Service Resource Flows) and focus on the specific aspects of the Service Resource Flow and the service resource content. These aspects of the service Resource Flow exchange can be crucial to the operational mission and are critical to understanding the potential for overhead and constraints introduced by the physical aspects of the implementation such as security policy and communications and logistics limitations.

The focus of SvcV-6 is on how the Service Resource Flow exchange is affected, in service-specific details covering periodicity, timeliness, throughput, size, information assurance, and security characteristics of the resource exchange. In addition, for Service Resource Flow of data, their format and media type, accuracy, units of measurement, applicable system data standards, and any DIV-3 Physical Data Models are also described or referenced in the matrix.

Modeling discipline is needed to ensure that the architecture models are coherent. Each Service Resource Flow exchange listed in the SvcV-6 table should be traceable to at least one Operational Resource Flow exchanged listed in the corresponding OV-3 Operational Resource Flow Matrix and these in turn trace to OV-2 Operational Resource Flow Description.

It should be noted that each resource exchanged may relate to a known service function (from SvcV-4) that produces or consumes it. However, there need not be a one-to-one correlation between data elements listed in the SvcV-6 matrix and the Resource Flows (inputs and outputs) that are produced or consumed in a related SvcV-4 because the SvcV-4 is more a logical solution, whereas the SvcV-6 is a more physical solution. In addition, Resource flows between known service functions performed by the same Services may not be shown in the SvcV-6 matrix. The SvcV-6 is about showing flows across service boundaries or a service boundary. If the Resource Flow is information, it may need to be reflected in the Data and Information Models.

The SvcV-7 Services Measures Matrix builds on the SvcV-6 and should be developed at the same time.

DoDAF does not prescribe the column headings in a SvcV-6 Matrix. Identifiers of the operational Resource Flow exchanges (OV-3) that are implemented by the Service Resource Flow Exchanges may be included in the table. All elements carried by the Resource Flow exchanges may be shown.

3.1.6.2.8 SvcV-7: Services Measures Matrix. The SvcV-7 depicts the measures (metrics) of resources (See Section 2.2.1 for the definition of resources). The Services Measures Matrix expands on the information presented in a SvcV-1 Services Context Description by depicting the characteristics of the resources in the SvcV-1 Services Context Description.

In addition, this model is useful in support of net-centric (service-oriented) implementation of services. Service measures for Service Level Agreements for each service and may include number of service consumers, service usage by consumers, and the minimum, average and maximum response times, allowed down time, etc. Measures of interest for a Chief Information Office or Program manager may include measures that assess service reuse, process efficiency, and business agility.

The intended usage of the SvcV-7 includes:

- Definition of performance characteristics and measures (metrics).
- Identification of non-functional requirements.

Detailed Description:

The SvcV-7 specifies qualitative and quantitative measures (metrics) of resources. It specifies all of the measures. The measures are selected by the end user community and described by the architect.

Performance parameters include all performance characteristics for which requirements can be developed and specifications defined. The complete set of performance parameters may not be known at the early stages of Architectural Description, so it is to be expected that this model is updated throughout the specification, design, development, testing, and possibly even its deployment and operations lifecycle phases. The performance characteristics are captured in the Measures Meta-model group described in Section 2.

One of the primary purposes of SvcV-7 is to communicate which measures are considered most crucial for the successful achievement of the mission goals assigned. These particular measures can often be the deciding factors in acquisition and deployment decisions, and figure strongly in services analysis and simulations done to support the acquisition decision processes and system design refinement and be input or may impact decisions about Service Level Agreement content. Measures of Effectiveness (MOEs) and Measures of Performers (MOPs) are measures that can be captured and presented in the Services Measures Matrix model.

SvcV-7 is typically a table, listing user defined measures (metrics) with a time period association. It is sometimes useful to analyze evolution by comparing measures (metrics) for current and future resources. For this reason, a hybrid SvcV-7 Model which spans architectures across multiple phases may be useful.

3.1.6.2.9 SvcV-8: Services Evolution Description. The SvcV-8presents a whole lifecycle view of resources (services), describing how it changes over time. It shows the structure of several resources mapped against a timeline.

In addition, this model is useful in support of net-centric (service-oriented) implementation of services. This model can present a timeline of services evolve or are replaced over time, including services that are internal and external to the scope of the architecture.

The intended usage of the SvcV-8 includes:

- Development of incremental acquisition strategy.
- Planning technology insertion.

Detailed Description:

The SvcV-8, when linked together with other evolution Models such as CV-2 Capability Taxonomy, CV-3 Capability Phasing and StdV-2 Standards Forecast, provides a rich definition of how the Enterprise and its capabilities are expected to evolve over time. In this manner, the model can be used to support an architecture evolution project plan or transition plan.

A SvcV-8 can describe historical (legacy), current, and future capabilities against a timeline. The model shows the structure of each resource, using similar modeling elements as those used in SvcV-1. Interactions which take place within the resource may also be shown.

The changes depicted in the SvcV-8 DoDAF-described Model are derived from the project milestones that are shown in a PV-2 Project Timelines model. When the PV-2 Project Timelines model is used for capability acquisition projects, there is likely to be a close relationship between these two models.

3.1.6.2.10 SvcV-9: Services Technology and Skills Forecast. The SvcV-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, and expected improvements or trends. New technologies and skills are tied to specific time periods, which can correlate against the time periods used in SvcV-8 Services Evolution Description model milestones and linked to Capability Phases.

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

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

In addition to providing an inventory of trends, capabilities and services, the SvcV-9 also includes an assessment of the potential impact of these items on the architecture. Given the future-oriented nature of this model, forecasts are typically made in short, mid and long-term timeframes, such as 6, 12 and 18-month intervals.

In addition, this model is useful in support of net-centric (service-oriented) implementation of services. As technologies change, like incorporation of Representational State Transfer (REST) services in the Web Services Description Language, this model can present a timeline of technologies related services over time.

The intended usage of the SvcV-9 includes:

- Forecasting technology readiness against time.
- HR Trends Analysis.
- Recruitment Planning.
- Planning technology insertion.
- Input to options analysis.

The SvcV-9 can be presented in a table, timeline, or a Herringbone diagram.

Detailed Description:

A SvcV-9 summarizes predictions about trends in technology and personnel. Architects may produce separate SvcV-9 products for technology and human resources. The specific time periods selected (and the trends being tracked) can be coordinated with architecture transition plans (which the SvcV-8 Services Evolution Description can support). That is, insertion of new capabilities and upgrading or re-training of existing resources may depend on or be driven by the availability of new technology and associated skills. The forecast includes potential impacts on current architectures and thus influences the development of transition and target architectures. The forecast is focused on technology and human resource areas that are related to the purpose for which a given architecture is being described and identifies issues affecting that architecture.

If standards are an integral part of the technologies important to the evolution of a given architecture, then it may be convenient to combine SvcV-9 with the StdV-2 Standards Forecast into a composite Fit-for-Purpose View.

The SvcV-9 is constructed as part of a given Architectural Description and in accordance with the its purpose. Typically, this involves starting with one or more overarching reference models or standards profiles to which the architecture is subject to using. Using these reference models or standards profiles, the architect selects the service areas and services relevant to the architecture. The SvcV-9 forecasts relate to the StdV-1Standards Profile in that a timed forecast may contribute to the decision to retire or phase out the use of a certain standard in connection with a resource. Similarly, the SvcV-9 forecasts relate to the StdV-2 Standards Forecasts in that a certain standard may be adopted depending on a certain technology or skill becoming available (e.g., the availability of Java Script may influence the decision to adopt a new HTML standard).

Alternatively, the SvcV-9 may relate forecasts to Service Model elements (e.g., Services) where applicable. The list of resources potentially impacted by the forecasts can also be summarized as additional information in SvcV-9.

3.1.6.2.11 Introduction to SvcV-10a, SvcV-10b and SvcV-10c. Many of the critical characteristics of an architecture are only discovered when an architecture's dynamic behaviors are defined and described. These dynamic behaviors concern the timing and sequencing of events that capture resource performance characteristics (i.e., a performer executing the service functions described in SvcV-4 Services Functionality Description).

Behavioral modeling and documentation are key to a successful Architectural Description, because it is understanding how the architecture behaves that is crucial in many situations. Although knowledge of the functions and interfaces is also crucial, knowing whether, for example, a response should be expected after sending message X to Service Y can be crucial to successful overall operations.

The SvcV-10 models are useful in support of net-centric (service-oriented) implementation of services as orchestrations of services. The SvcV-3 Services-Services Matrix can provide input for the SvcV-10 models. Three types of models may be used to adequately describe the dynamic behavior and performance characteristics of Service elements. These three models are:

- Services Rules Model (SvcV-10a).
- Services State Transition Description (SvcV-10b).
- Services Event-Trace Description (SvcV-10c).

SvcV-10b and SvcV-10c may be used separately or together, as necessary, to describe critical timing and sequencing behavior in the Service Model. Both types of diagrams are used by a wide variety of different Services methodologies.

Both SvcV-10b and SvcV-10c describe functional responses to sequences of events. Events may also be referred to as inputs, transactions, or triggers. When an event occurs, the action to be taken may be subject to a rule or set of rules as described in SvcV-10a.

3.1.6.2.11.1 SvcV-10a Services Rules Model. The SvcV-10a is to specify functional and non-functional constraints on the implementation aspects of the architecture (i.e., the structural and behavioral elements of the Services Model).

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

The intended usage of the SvcV-10a includes:

- Definition of implementation logic.
- Identification of resource constraints.

Detailed Description:

The SvcV-10a describes the rules that control, constrain or otherwise guide the implementation aspects of the architecture. Service Rules are statements that define or constrain some aspect of the business, and may be applied to:

- Performers.
- Resource Flows.
- Service Functions.
- System Ports.
- Data Elements.

In contrast to the OV-6a Operational Rules Model, the SvcV-10a focuses physical and data constraints rather than business rules.

Constraints can be categorized as follows:

- Structural Assertions – non-functional constraints governing some physical aspect of the architecture.
- Action Assertions – functional constraints governing the behavior of resources, their interactions and Resource Flow exchanges.
- Derivations – these involve algorithms used to compute facts.

Where a Service Rule is based on some standard, then that standard should be listed in the StdV-1 Standards Profile.

Some Service Rules can be added as annotations to other models. The SvcV-10a then should provide a listing of the complete set of rules with a reference to any models that they affect.

3.1.6.2.11.2 SvcV-10b Services State Transition Description. The SvcV-10b is a graphical method of describing a resource (or 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 SvcV-4 Services Functionality Description. SvcV-10b can be used to describe the explicit sequencing of the service functions. Alternatively, SvcV-10b can be used to reflect explicit sequencing of the actions internal to a single service function, or the sequencing of service functions with respect to a specific resource.

The intended usage of the SvcV-10b includes:

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

Detailed Description:

The SvcV-10b relates events to resource states and describes the transition from one state to another.

The SvcV-10b is based on the statechart diagram. A state machine is defined as "a specification that describes all possible behaviors of some dynamic view element. Behavior is viewed as a traversal of a graph of specific states interconnected by one or more joined transition arcs that are triggered by the dispatching of series of event instances. During this traversal, the state machine executes a series of actions associated with various elements of the state machine." Statechart diagrams can be unambiguously converted to structured textual rules that specify timing aspects of events and the responses to these events, with no loss of meaning. However, the graphical form of the state diagrams can often allow quick analysis of the completeness of the rule set, and detection of dead ends or missing conditions. These errors, if not detected early during the solution analysis phase, can often lead to serious behavioral errors in fielded capabilities and to expensive correction efforts.

The SvcV-10b models state transitions from a resource perspective, with a focus on how the resource responds to stimuli (e.g., triggers and events). As in the OV-6b Operational State Transition Description, these responses may differ depending upon the rule set or conditions that apply, as well as the resource's state at the time the stimuli is received. A change of state is called a transition. Each transition specifies the response based on a specific event and the current state. Actions may be associated with a given state or with the transition between states. A state and its associated actions specify the response of a resource or service function, to events. When an event occurs, the next state may vary depending on the current state (and its associated action), the event, and the rule set or guard conditions.

The SvcV-10b can be used to describe the detailed sequencing of service functions described in SvcV-4 Services Functionality Description. However, the relationship between the actions included in SvcV-10b and the functions in SvcV-4 depends on the purposes of the Architectural Description and the level of abstraction used in the models. The explicit sequencing of functions in response to external and internal events is not fully expressed in SvcV-4 Services Functionality Description. SvcV-10b can be used to reflect explicit sequencing of the functions, the sequencing of actions internal to a single function, or the sequencing of functions with respect to a specific resource.

States in a SvcV-10b model may be nested. This enables quite complex models to be created to represent Services behavior. Depending upon the architecture project's needs, the SvcV-10b may be used separately or in conjunction with the SvcV-10c Services Event-Trace Description.

3.1.6.2.11.3 SvcV-10c Services Event-Trace Description. The SvcV-10c provides a time-ordered examination of the interactions between services functional resources. Each event-trace diagram should have an accompanying description that defines the particular scenario or situation.

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

The intended usage of the SvcV-10c includes:

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

Detailed Description:

The SvcV-10c specifies the sequence in which Resource Flow elements are exchanged in context of a resource or Service Port. Services Event-Trace Descriptions are sometimes called sequence diagrams, event scenarios or timing diagrams. The components of a SvcV-10c include functional resources or service ports, owning performer, as well as the port which is the subject for the lifeline.

Specific points in time can be identified. The Resource Flow from one resource/port to another can be labeled with events and their timing. The Service Event-Trace Description provides a time-ordered examination of the Resource Flow elements exchanged between participating resources (external and internal) or service ports. Each Event-Trace diagram should have an accompanying description that defines the particular scenario or situation.

The SvcV-10c is typically used in conjunction with the SvcV-10b Services State Transition Description to describe the dynamic behavior of resources. The data content of messages that connect Resource Flows in a SvcV-10c model may be related, in modeling terms, with Resource Flows (interactions, in SvcV-1 Services Context Description, SvcV-3a Systems-Services Matrix, and SvcV-3b Services-Services Matrix), Resource Flows (data, in SvcV-4 Services Functionality Description and SvcV-6 Services Resource Flow Matrix) and entities (in DIV-3 Physical Data Model) modeled in other models.

3.1.7 Standards Viewpoint

The DoDAF-described Models within the Standards Viewpoint is the set of rules governing the arrangement, interaction, and interdependence of parts or elements of the Architectural Description. These sets of rules can be captured at the enterprise level and applied to each solution, while each solution's architectural description depicts only those rules pertinent to architecture described. Its purpose is to ensure that a solution satisfies a specified set of operational or capability requirements. The Standards Models capture the doctrinal, operational, business, technical, or industry implementation guidelines upon which engineering specifications are based, common building blocks are established, and solutions are developed. It includes a collection of the doctrinal, operational, business, technical, or industry standards, implementation conventions, standards options, rules, and criteria that can be organized into profiles that govern solution elements for a given architecture. Current DoD guidance requires the Technical Standards portions of models be produced from DISR to determine the minimum set of standards and guidelines for the acquisition of all DoD systems that produce, use, or exchange information.

Names of the models and their descriptions (in **Table 3.1.7-1**) are provided below.

Table 3.1.7-1: Standard Model Descriptions

Models	Descriptions
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.

3.1.7.1 Uses of Standards Viewpoint DoDAF-described Models. The Standards Viewpoint can articulate the applicable policy, standards, guidance, constraints, and forecasts required by JCIDS, DAS, System Engineering, PPBE, Operations, other process owners, and decision-makers.

Mappings of the Standards Viewpoint DoDAF-described Models to the DM2 Concepts, Associations, and Attributes are in **Table B-1 DM2 Concepts, Associations, and Attributes Mapping to DoDAF-described Models**. The DM2 Concepts, Associations, and Attributes are described in the DoDAF Meta-model Data Dictionary.

3.1.7.2 Model Descriptions. The Standards Viewpoint DoDAF-described Models are described below. In addition, examples of these models can be viewed online in the public DoDAF Journal.

3.1.7.2.1 StdV-1: Standards Profile. 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 DoDAF V2.0 StdV-1 also documents the policies and standards that apply to the operational or business context. The DISR is an architecture resource for technical standards that can be used in the generation of the StdV-1 and StdV-2 Standards Forecast.

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.

Detailed Description:

The StdV-1 collates the various systems and services, standards, and rules that implement and constrain the choices that can be or were made in the design and implementation of an Architectural Description. It delineates the systems, services, Standards, and rules that apply. The technical standards govern what hardware and software may be implemented and on what system. The standards that are cited may be international such as ISO standards, national standards, or organizational specific standards.

With associated standards with other elements of the architecture, a distinction is made between applicability and conformance. If a standard is applicable to a given architecture, that architecture need not be fully conformant with the standard. The degree of conformance to a given standard may be judged based on a risk assessment at each approval point.

Note that an association between a Standard and an architectural element should not be interpreted as indicating that the element is fully compliant with that Standard. Further detail would be needeed to confirm the level of compliance.

Standards Profiles for a particular architecture must maintain full compatibility with the root standards they have been derived from. In addition, the StdV-1 model may state a particular method of implementation for a Standard, as compliance with a Standard does not ensure interoperability. The Standards cited are referenced as relationships to the systems, services, system functions, service functions, system data, service data, hardware/software items or communication protocols, where applicable, in:

- SV-1 Systems Interface Description.
- SV-2 Systems Resource Flow Description.
- SV-4 Systems Functionality Description.
- SV-6 Systems Resource Flow Matrix.
- SvcV-1 Services Context Description.
- SvcV-2 Services Resource Flow Description.
- SvcV-4 Services Functionality Description.
- SvcV-6 Services Resource Flow Matrix.
- DIV-2 Logical Data Model.
- DIV-3 Physical Data Model.

That is, each standard listed in the profile is associated with the elements that implement or use the standard.

The protocols referred to Resource Flow descriptions (see SV-2 Systems Resource Flow Description or SvcV-2 Services Resource Flow Description) are examples of Standards and these should also be included in the StdV-1 listing, irrespective of which models they appear in or are referred from.

3.1.7.2.2 StdV-2: Standards Forecast. 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-1Standards 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).

Detailed Description:

The Standards Forecast DoDAF-described Model contains expected changes in 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. One of the prime purposes of this model is to identify critical standards, their life expectancy, and the impact of these standards on the future development and maintainability of the Architectural Description and its constituent elements.

StdV-2 lists emerging or evolving standards relevant to the solutions covered by the Architectural Description. It contains predictions about the availability of emerging standards, and relates these predictions to the elements and the time periods that are listed 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.

The specific time periods selected (e.g., 6-month and 12-month intervals) and the standards being tracked are coordinated with architecture transition plans (which the SV-8 Systems Evolution Description and SvcV-8 Services Evolution Description can support). That is, insertion of new capabilities and upgrading of existing solutions may depend on, or be driven by, the availability of new standards and models incorporating those standards. The forecast specifies potential standards and thus impacts current architectures and influences the development of transition and objective (i.e., target) architectures. The forecast is tailored to focus on standards areas that are related to the purpose for which a given architecture is being described and should identify potential standards affecting that architecture. If interface standards are an integral part of the technologies important to the evolution of a given architecture, then it may be convenient to combine StdV-2 with SV-9 Systems Technology & Skills Forecast or SvcV-9 Services Technology & Skills Forecast into a composite Fit-for-

Purpose View. For other projects, it may be convenient to combine all the standards information into one composite Fit-for-Purpose View, combining StdV-2 with StdV-1 Standard Profile.

StdV-2 delineates the standards that potentially impact the relevant system and service elements (from SV-1 Systems Interface Description, SV-2 Systems Resource Flow Description, SV-4 Systems Functionality Description, SV-6 Systems Resource Flow Matrix, SvcV-1 Services Context Description, SvcV-2 Services Resource Flow Description, SvcV-4 Services Functionality Description, SV-6 Services Resource Flow Matrix, and DIV-2 Logical Data Model) and relates them to the time periods that are listed 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 system's evolution, specified in SV-8 Systems Evolution Description, or service's evolutions, specified in SvcV-8 Services Evolution Description, may be tied to a future standard listed in StdV-2. A timed technology and skills forecast from SV-9 Systems Technology & Skills Forecast or SvcV-9 Services Technology & Skills Forecast models is related to StdV-2 standards forecast in the following manner: a certain technology may be dependent on a StdV-2 standard (i.e., a standard listed in StdV-2 may not be adopted until a certain technology becomes available). This is how a prediction on the adoption of a future standard, may be related to standards listed in StdV-1 through the SV-9 Systems Technology & Skills Forecast or SvcV-9 Services Technology & Skills Forecast models.

3.1.8 Systems Viewpoint

The DoDAF-described Models within the Systems Viewpoint describes systems and interconnections providing for, or supporting, DoD functions. DoD functions include both warfighting and business functions. The Systems Models associate systems resources to the operational and capability requirements. These systems resources support the operational activities and facilitate the exchange of information. The Systems DoDAF-described Models are available for support of legacy systems. As architectures are updated, they should transition from Systems to Services and utilize the models within the Services Viewpoint.

Names of the models and their descriptions (in **Table 3.1.8-1**) are provided below.

Table 3.1.8-1: Systems Model Descriptions

Models	Descriptions
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 Operational Activity to Systems Function Traceability Matrix	A mapping of system functions (activities) back to operational activities (activities).

Table 3.1.8-1: Systems Model Descriptions

Models	Descriptions
SV-5b Operational Activity 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 Description	One of three models used to describe system functionality. It identifies system-specific refinements of critical sequences of events described in the Operational Viewpoint.

3.1.8.1 Uses of System Viewpoint DoDAF-described Models. Within the development process, the DoDAF-described Models describe the design for system-based solutions to support or enable requirements created by the operational development processes (JCIDS) and Defense Acquisition System.

Mappings of the Systems Viewpoint DoDAF-described Models, to the DM2 Concepts, Associations, and Attributes are in **Table B-1 DM2 Concepts, Associations, and Attributes Mapping to DoDAF-described Models**. The DM2 Concepts, Associations, and Attributes are described in the DoDAF Meta-model Data Dictionary.

3.1.8.2 Model Descriptions. The System Viewpoint DoDAF-described Models are described below. In addition, examples of models can be viewed online in the public DoDAF Journal.

3.1.8.2.1 SV-1: Systems Interface 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. Resources are defined in Section 2.2.1

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 locations 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.

Detailed Description:

A SV-1 can be used simply to depict Systems and sub-systems and identify the Resource Flows between them. The real benefit of a SV-1 is its ability to show the human aspects of an architecture, and how these interact with Systems. In addition, DoDAF has the concept of Capability and Performers (see Capability Meta-model group in Section 2) which is used to gather together systems, assets and people into a configuration, which can meet a specific capability. A primary purpose of a SV-1 DoDAF-described Model is to show resource structure, i.e., identify the primary sub-systems, performer and activities (functions) and their interactions. SV-1 contributes to user understanding of the structural characteristics of the capability.

The physical resources contributing to a capability are either an organizational resource or a physical asset, i.e., a system cannot contribute alone (it must be hosted on a physical asset used by an organizational resource of both). Organizational aspects can now be shown on SV-1 (e.g., who uses System). Resource structures may be identified in SV-1 to any level (i.e., depth) of decomposition the architect sees fit. DoDAF does not specifically use terms such as, sub-System and component as these terms often denote a position relative to a structural hierarchy. Any

System may combine hardware and software or these can be treated as separate (sub) Systems. DoDAF V2.0 includes human factors (as Personnel Types and a type of Performer). Should an architect wish to describe a System which has human elements, then Systems, Personnel Types and Performers should be used to wrap the human and system elements together.

A SV-1 can optionally be annotated with Operational Activities, Capabilities, and/or Locations originally specified in OV-2 Operational Resource Flow Description model. In this way, traceability can be established from the logical OV structure to the physical System Viewpoint structure.

If possible, a SV-1 shows Systems, Physical Assets and System interfaces for the entire Architectural Description on the same diagram. If a single SV-1 is not possible, the resource of interest should be decomposed into multiple SV-1 models.

Functions (Activities):

Some Resources can carry out System Functions (Activities) as described in SV-4 Systems Functionality Description model and these functions can optionally be overlaid on a SV-1. In a sense, the SV-1 and the SV-4 Systems Functionality Description model provide complementary representations (structure and function). Either could be modeled first, but usually an iterative approach is used to model these together gradually building up the level of detail in the System description. Note that the same type (class) of resource may be used in different contexts in a given SV-1. For this reason, the tracing of functions to resources is specified in context of their usage (see DM2 for details).

Resource Flows in SV-1:

In addition to depicting Systems (Performers) and their structure, the SV-1 addresses Resource Flows. A Resource Flow, as depicted in SV-1, is an indicator that resources pass between one System and the other. In the case of Systems, this can be expanded into further detail in SV-2 Systems Resource Flow Description.

Interactions are only possible between Systems and Services. System Resource Flows provide a specification for how the operational Resource Flows Exchanges specified in Needlines (in the OV-2 Operational Resource Flow Description model) are realized with Systems. A single Needline shown in the OV-2 Operational Resource Flow Description model may translate into multiple System Resource Flows.

The actual implementation of a System Resource Flow may take more than one form (e.g., multiple physical links). Details of the physical pathways or network patterns that implement the interfaces are documented in SV-2 Systems Resource Flow Description. System Resource Flows are summarized in a SV-3b Systems-Systems Matrix. The functions performed by the resources are specified in a SV-4 System Functionality Description, but may optionally be overlaid on the Resources in a SV-1.

An Operational Viewpoint (OV) suite may specify a set of requirements – either as a specific operational plan, or a scenario for procurement. As OV-2 Operational Resource Flow Description, OV-5a Operational Activity Decomposition Tree, and OV-5b Operational Activity Model specify the logical structure and behavior, SV-1 and SV-4 Systems Functionality

Description specify the physical structure and behavior (to the level of detail required by the architectural stakeholders). This separation of logical and physical presents an opportunity for carrying out architectural trade studies based on the architectural content in the DoDAF-described Models.

The structural and behavioral models in the OVs and SVs allow architects and stakeholders to quickly ascertain which functions are carried out by humans and which by Systems for each alternative specification and so carry out trade analysis based on risk, cost, reliability, etc.

3.1.8.2.2 SV-2: Systems Resource Flow 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.

Detailed Description:

A SV-2 comprises Systems, their ports, and the Resource Flows between those ports. The architect may choose to create a diagram for each Resource Flow for all Systems or to show all the Resource Flows on one diagram if possible.

Each SV-2 model can show:

- Which ports are connected?
- The Systems that the ports belong to.
- The definition of the System Resource Flow in terms of the physical/logical connectivity and any protocols that are used in the connection.

Note that networks are represented as Systems. The architect may choose to show other Systems being components of the network, i.e., if they are part of the network infrastructure.

Any protocol referred to in a SV-2 diagram needs to be defined in the StdV-1 Standards Profile.

3.1.8.2.3 SV-3: Systems-Systems Matrix. 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 a number of 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.

Detailed Description:

The SV-1 concentrates on System resources and their interactions, and these are summarized in a SV-3. The SV-3 can be a useful tool for managing the evolution of solutions and infrastructures, the insertion of new technologies and functionality, and the redistribution of systems and activities in context with evolving operational requirements.

Depending upon the purpose of the Architectural Description, there could be several SV-3s. The suite of SV-3 models can be organized in a number of ways (e.g., by domain, by operational mission phase, by solution option) to emphasize the association of groups of resource pairs in context with the Architectural Description purpose.

The SV-3 is generally presented as a matrix, where the Systems resources are listed in the rows and columns of the matrix, and each cell indicates an interaction between resources if one exists. Many types of interaction information can be presented in the cells of a SV-3. The resource interactions can be represented using different symbols and/or color coding that depicts different interaction characteristics, for example:

- Status (e.g., existing, planned, potential, de-activated).
- Key interfaces.
- Category (e.g., command and control, intelligence, personnel, logistics).
- Classification-level (e.g., Restricted, Confidential, Secret, Top Secret).
- Communication means (e.g., Rim Loop Interface, Scalable Loop Interface).

DoDAF does not specify the symbols to be used. If symbols are used, a key is needed.

3.1.8.2.4 SV-4: Systems Functionality 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.
- 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.

Detailed Description:

The SV-4 is used to specify the functionality of resources in the architecture (in this case, functional resources, systems, performer and capabilities). The SV-4 is the behavioral counterpart to the SV-1 Systems Interface Description (in the same way that OV-5b Operational Activity Model is the behavioral counterpart to OV-2 Operational Resource Flow Matrix).

The scope of this model may be capability wide, without regard to which resources perform which functions, or it may be resource-specific. Variations may focus on intra- or inter-resource data flows, or may simply allocate functions to resources.

There are two basic ways to depict SV-4:

- The Taxonomic Functional Hierarchy shows a decomposition of functions depicted in a tree structure and is typically used where tasks are concurrent but dependent, such as a production line, for example.
- The Data Flow Diagram shows functions connected by data flow arrows and data stores.

The Taxonomic Functional Hierarchy may be particularly useful in capability-based procurement where it is necessary to model the functions that are associated with particular capability (see SV-5).

Within an Architectural Description, the SV-4 documents system functions, the Resource Flows between those functions, the internal system data repositories or system data stores, and the external producers and consumers for the system data flows, but not those external to the Architectural Description scope. They may also show how users behave in relation to those systems.

The functions are likely to be related to Operational Activities captured in OV-5a. Although there is a correlation between the Operational Activity Model (OV-5b) and the functional hierarchy of SV-4, it need not be a one-to-one mapping, hence, the need for the Function to Operational Activity Traceability Matrix (SV-5), which provides that mapping.

Systems are not limited to internal system functions and can include HCI and GUI functions or functions that consume or produce system data. The external system data producers or consumers can be used to represent the human that interacts with the system. The System Resource Flows between the external system data source/sink (representing the human or system) and the HCI, GUI, or interface function can be used to represent human-system interactions, or system-system interfaces. Standards that apply to system functions, such as HCI and GUI standards, are also specified during development of this model (and recorded in StdV-1).

A graphical variant of the SV-4 Data Flow model may be used with swim lanes. A system swim lane may be associated with:

- A System.
- A grouping of Capabilities and System Functions (usually based on a Physical Asset).
- A Performer executing an Activity.

Swim lanes are presented either vertically or horizontally. A function can be placed in the swim lane associated with the System, Resources or Performer executing an Activity that it is allocated in the solution architecture. This provides a graphical means of presenting the interactions between Systems or Capabilities (shown through system connections on SV-1) in functional terms. This is a powerful technique for visualizing the differences between alternative solution options (which may have a common set of functions).

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

Detailed Description:

An SV-5a is a specification of the relationships between the set of operational activities applicable to an Architectural Description and the set of system functions applicable to that Architectural Description. The relationship between operational activities and system functions can also be expected to be many-to-many (i.e., one activity may be supported by multiple functions, and one function may support multiple activities). The system functions shown in the SV-5a may be those associated with capabilities and performers. More focused SV-5a models might be used to specifically trace system functions to operational activities if desired.

DoDAF uses the term Operational Activity in the OVs and the term System Function in the SVs to refer to essentially the same kind of thing; both activities and functions are tasks that are performed, accept inputs, and develop outputs. The distinction between an Operational Activity and a Function is a question of *what* and *how*. The Operational Activity is a specification of *what* is to be done, regardless of the mechanism used. A System Function is specifies *how* a resource carries it out. For this reason, SV-5a is a significant model, as it ties together the logical specification in the OV-5a with the physical specification of the SV-4 Systems Functionality Description. System Functions can be carried out by Functional Resources (systems, performers executing activities, and performers).

The SV-5a is generally presented as a matrix of the relationship between system functions and operational activities. The SV-5a can show requirements traceability with Operational Activities on one axis of a matrix, the System Functions on the other axis, and with an X, date, or phase in the intersecting cells, where appropriate.

An alternate version of the tabular SV-5a can allow the implementation status of each function to be shown. In this variant model, each system function-to-operational activity mapping is described by a traffic light symbol that may indicate the status of the system support. DoDAF V2.0 does not prescribe a presentation technique. These symbols are usually colored circles with the following possible representations:

- Red may indicate that the functionality is planned but not developed.
- Yellow may indicate that partial functionality has been provided (or full functionality provided but system has not been fielded).
- Green may indicate that full functionality has been provided to the field.
- A blank cell may indicate that there is no system support planned for an Operational Activity, or that a relationship does not exist between the Operational Activity and the System Function.

Care should be taken when publishing a SV-5a with status information. Any presentation should clearly state the date of publication, so that users can see when status information is old.

SV-5a may be further annotated with Systems, Capabilities, Performers executing Activities, and capabilities and performers that conduct the functions.

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

Detailed Description:

An SV-5b is a specification of the relationships between the set of operational activities applicable to an Architectural Description and the set of systems applicable to that Architectural Description. The relationship between operational activities and systems can also be expected to be many-to-many (i.e., one activity may be supported by multiple systems, and one system may support multiple activities). The system shown in the SV-5b may be those associated with resources. More focused SV-5b models might be used to specifically trace system to operational activities if desired.

The SV-5b is generally presented as a matrix of the relationship between systems and activities and can be a summary of the Operational Activity to System Function Traceability Matrix (SV-5a). The SV-5b can show requirements traceability with Operational Activities on one axis of a matrix, the System Functions on the other axis, and with an X, date, or phase in the intersecting cells, where appropriate.

An alternate version of the tabular SV-5b model can allow the implementation status of each system to be shown. In this variant model, each system-to-operational activity mapping is described by a traffic light symbol that may indicate the status of the system support. DoDAF V2.0 does not prescribe a presentation technique. These symbols are usually colored circles with the following possible representations:

- Red may indicate that the system is planned but not developed.
- Yellow may indicate that partial system functionality has been provided (or full functionality provided but system has not been fielded).
- Green may indicate that full system functionality has been provided to the field.
- A blank cell may indicate that there is no system support planned for an Operational Activity, or that a relationship does not exist between the Operational Activity and the System Function.

Care should be taken when publishing a SV-5b with status information. Any presentation should clearly state the date of publication, so that users can see when status information is old.

The SV-5b may be further annotated with Capabilities, Performers executing Activities, and capabilities and performers that conduct the functions. This can be used to identify which systems can support a particular capability. The architect may also wish to hide the systems in a SV-5b so that the table simply shows the mapping from performers executing activities, and capabilities and performers to Operational Activities.

3.1.8.2.7 SV-6: Systems Resource Flow Matrix. 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.

Detailed Description:

The SV-6 specifies the characteristics of Resource Flow exchanges between systems. The SV-6 is the physical equivalent of the logical OV-3 table and provides detailed information on the system connections which implement the Resource Flow exchanges specified in OV-3. Non-automated Resource Flow exchanges, such as verbal orders, are also captured.

System Resource Flow exchanges express the relationship across the three basic architectural data elements of a SV (systems, system functions, and system Resource Flows) and focus on the specific aspects of the System Resource Flow and the system resource content. These aspects of the System Resource Flow exchange can be crucial to the operational mission and are critical to understanding the potential for overhead and constraints introduced by the physical aspects of the implementation such as security policy and communications limitations.

The focus of SV-6 is on how the System Resource Flow exchange is affected, in system-specific details covering periodicity, timeliness, throughput, size, information assurance, and security characteristics of the resource exchange. In addition, the System Resource Flow elements, their format and media type, accuracy, units of measurement, and system data standard are also described in the matrix.

Modeling discipline is needed to ensure that the architecture models are coherent. Each system Resource Flow exchange listed in the SV-6 table should be traceable to at least one operational Resource Flow exchanged listed in the corresponding OV-3 Operational Resource Flow Matrix and these, in turn, trace to operation Resource Flows in the OV-2 Operational Resource Flow Description.

It should be noted that each data element exchanged may be related to the system function (from SV-4) that produces or consumes it. However, there need not be a one-to-one correlation between data elements listed in the SV-6 matrix and the data flows (inputs and outputs) that are produced or consumed in a related SV-4 Services Functionality Description. In addition, Data flows between system functions performed by the same systems may not be shown in the SV-6 matrix. SV-6 is about showing flows across system boundaries.

The SV-7 System Measures Matrix model builds on the SV-6 and should be developed at the same time.

DoDAF does not prescribe the column headings in a SV-6 Matrix. Identifiers of the operational Resource Flows from the OV-3 Operational Resource Flow Matrix that are implemented by the System Resource Flow Exchanges may be included in the table. All elements carried by the Resource Flow exchanges may be also shown.

3.1.8.2.8 SV-7: Systems Measures Matrix. The SV-7 depicts the measures (metrics) of resources (See Section 2.2.1 for the definition 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.

Detailed Description:

The SV-7 specifies qualitative and quantitative measures (metrics) of resources; it specifies all of the measures. The measures are selected by the end user community and described by the architect.

Performance parameters include all performance characteristics for which requirements can be developed and specifications defined. The complete set of performance parameters may not be known at the early stages of Architectural Description, so it is to be expected that this model is updated throughout the specification, design, development, testing, and possibly even its deployment and operations lifecycle phases. The performance characteristics are captured in the Measures Meta-model group described in Section 2.

One of the primary purposes of SV-7 is to communicate which measures are considered most crucial for the successful achievement of the mission goals assigned and how those performance parameters will be met. These particular measures can often be the deciding factors in acquisition and deployment decisions, and figures strongly in systems analysis and simulations done to support the acquisition decision processes and system design refinement. Measures of Effectiveness (MOEs) and Measures of Performers (MOPs) are measures that can be captured and presented in the Services Measures Matrix model.

The SV-7 DoDAF-described Model is typically a table listing user defined measures (metrics) with a time period association. It is sometimes useful to analyze evolution by comparing measures (metrics) for current and future resources. For this reason, a hybrid SV-7 model which spans architectures across multiple phases may be useful.

3.1.8.2.9 SV-8: Systems Evolution 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.

Detailed Description:

The SV-8, when linked together with other evolution Models, e.g., such as CV-3 Capability Phasing and StdV-2 Standards Forecast, provides a rich definition of how the Enterprise and its capabilities are expected to evolve over time. In this manner, the model can be used to support an architecture evolution project plan or transition plan.

A SV-8 can either describe historical (legacy), current, and future capabilities against a timeline. The model shows the structure of each resource, using similar modeling elements as those used in SV-1. Interactions which take place within the resource may also be shown.

The changes depicted in the SV-8 are derived from the project milestones that are shown in a PV-2 Project Timelines. When the PV-2 Project Timelines is used for capability acquisition projects, there is likely to be a close relationship between these two models.

3.1.8.2.10 SV-9: Systems Technology and Skills Forecast. 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.

In addition to providing an inventory of trends, capabilities and systems, the DoDAF-described Model SV-9 also includes an assessment of the potential impact of these items on the architecture. Given the future-oriented nature of this model, forecasts are typically made in short, mid and long-term timeframes, such as 6, 12 and 18-month intervals.

The intended usage of the SV-9 includes:

- Forecasting technology readiness against time.
- HR Trends Analysis.
- Recruitment Planning.
- Planning technology insertion.
- Input to options analysis.

The SV-9 can be presented in a table, timeline, or a Herringbone diagram.

Detailed Description:

A SV-9 summarizes predictions about trends in technology and personnel. Architects may produce separate SV-9 products for technology and human resources. The specific time periods selected (and the trends being tracked) are coordinated with architecture transition plans (which the SV-8 Systems Evolution Description model can support). That is, insertion of new capabilities and upgrading or re-training of existing resources may depend on or be driven by the availability of new technology and associated skills. The forecast includes potential impacts on current architectures and thus influences the development of transition and target architectures.

The forecast is focused on technology and human resource areas that are related to the purpose for which a given architecture is being described and identifies issues affecting that architecture.

If standards are an integral part of the technologies important to the evolution of a given architecture, then it may be convenient to combine SV-9 with the StdV-2 Standards Forecast in a composite Fit-for-Purpose View.

The SV-9 is constructed as part of a given Architectural Description and in accordance with the Architectural Description purpose. Typically, this involves starting with one or more overarching reference models or standards profiles to which the architecture must conform. Using these reference models or standards profiles, the architect selects the service areas and services relevant to the architecture. The SV-9 DoDAF-described Model forecasts relates to the Standards Profile (StdV-1) in that a timed forecast may contribute to the decision to retire or phase out the use of a certain standard in connection with a resource. Similarly, SV-9 forecasts relate to the Standards Forecasts (StdV-2) in that a certain standard may be adopted depending on a certain technology or skill becoming available (e.g., the availability of Java Script may influence the decision to adopt a new HTML standard).

Alternatively, the SV-9 may relate forecasts to SV elements (e.g., systems) where applicable. The list of resources potentially impacted by the forecasts can also be summarized as additional information in a SV-9.

3.1.8.2.11 Introduction to SV-10a, SV-10b and SV-10c. Many of the critical characteristics of an architecture are only discovered when an architecture's dynamic behaviors are defined and described. These dynamic behaviors concern the timing and sequencing of events that capture resource performance characteristics (i.e., a performer executing the system functions described in SV-4).

Behavioral modeling and documentation are key to a successful Architectural Description, because it describes how the architecture behaves which is crucial in many situations. Although knowledge of the functions and interfaces is also crucial, knowing whether, for example, a response should be expected after sending message X to System Function Y can be crucial to successful overall operations.

The SV-10 DoDAF-described Models are useful in support of net-centric (service-oriented) implementation of services as orchestrations of services. The SV-3 Systems-Systems Matrix can provide input for the SV-10 DoDAF-described Models. Three types of models may be used to adequately describe the dynamic behavior and performance characteristics of System elements. These three models are:

- Systems Rules Model (SV-10a).
- Systems State Transition Description (SV-10b).
- Systems Event-Trace Description (SV-10c).

SV-10b and SV-10c may be used separately or together, as necessary, to describe critical timing and sequencing behavior in the SV. Both types of diagrams are used by a wide variety of different systems methodologies.

Both SV-10b and SV-10c describe functional responses to sequences of events. Events may also be referred to as inputs, transactions, or triggers. When an event occurs, the action to be taken may be subject to a rule or set of rules as described in SV-10a.

3.1.8.2.11.1 SV-10a: Systems Rules Model. 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.

Detailed Description:

The Systems Rules Model DoDAF-described Model describes the rules that control, constrain or otherwise guide the implementation aspects of the architecture. System Rules are statements that define or constrain some aspect of the business, and may be applied to:

- Performers.
- Resource Flows.
- System Functions.
- System Ports.
- Data Elements.

In contrast to the OV-6a Operational Rules Model, SV-10a focuses on physical and data constraints rather than business rules.

Constraints can be categorized as follows:

- Structural Assertions – non-functional constraints governing some physical aspect of the architecture.
- Action Assertions – functional constraints governing the behavior of resources, their interactions and Resource Flow exchanges.
- Derivations – these involve algorithms used to compute facts.

Where a System Rule is based on some standard, then that standard should be listed in the StdV-1 Standards Profile.

Some System Rules can be added as annotations to other models. The SV-10a then should provide a listing of the complete set of rules with a reference to any models that they affect.

3.1.8.2.11.2 SV-10b: Systems State Transition 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.

Detailed Description:

The SV-10b relates events to resource states and describes the transition from one state to another. The SV-10b is based on the state chart diagram. A state machine is defined as "a specification that describes all possible behaviors of some dynamic view element. Behavior is modeled as a traversal of a graph of specific states interconnected by one or more joined transition arcs that are triggered by the dispatching of series of event instances. During this traversal, the state machine executes a series of actions associated with various elements of the state machine." State chart diagrams can be unambiguously converted to structured textual rules that specify timing aspects of events and the responses to these events, with no loss of meaning. However, the graphical form of the state diagrams can often allow quick analysis of the completeness of the rule set, and detection of dead ends or missing conditions. These errors, if not detected early during the solution analysis phase, can often lead to serious behavioral errors in fielded capabilities, or to expensive correction efforts.

The SV-10b models state transitions from a resource perspective, with a focus on how the resource responds to stimuli (e.g., triggers and events). As in the OV-6b Operational State Transition Description, these responses may differ depending upon the rule set or conditions that apply as well as the resource's state at the time the stimuli is received. A change of state is called a transition. Each transition specifies the response based on a specific event and the current state. Actions may be associated with a given state or with the transition between states. A state and its associated actions specify the response of a resource or function, to events. When an event occurs, the next state may vary depending on the current state (and its associated action), the event, and the rule set or guard conditions.

The SV-10b can be used to describe the detailed sequencing of functions described in SV-4 Systems Functionality Description. However, the relationship between the actions included in SV-10b and the functions in SV-4 Systems Functionality Description depends on the purposes of the architecture and the level of abstraction used in the models. The explicit sequencing of functions in response to external and internal events is not fully expressed in SV-4 Systems Functionality Description. SV-10b can be used to reflect explicit sequencing of the functions, the sequencing of actions internal to a single function, or the sequencing of functions with respect to a specific resource.

States in a SV-10b model may be nested. This enables quite complex models to be created to represent systems behavior. Depending upon the architecture project's needs, the SV-10b may be used separately or in conjunction with the SV-10c Systems Event-Trace Description.

3.1.8.2.11.3 SV-10c: Systems Event-Trace 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 particular 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.

Detailed Description:

The SV-10c specifies the sequence in which Resource Flow elements are exchanged in context of a resource or System Port. Systems Event-Trace Descriptions are sometimes called sequence diagrams, event scenarios or timing diagrams. The components of a SV-10c include functional resources or system ports, owning performer as well as the port which is the subject for the lifeline.

Specific points in time can be identified. The Resource Flow from one resource/port to another can be labeled with events and their timing. The System Event-Trace Description provides a time-ordered examination of the Resource Flow elements exchanged between participating resources (external and internal) or system ports. Each Event/Trace diagram should have an accompanying description that defines the particular scenario or situation.

The SV-10c is typically used in conjunction with the SV-10b Systems State Transition Description to describe the dynamic behavior of resources. The data content of messages that connect Resource Flows in a SV-10c may be related with Resource Flows (the interactions in the SV-1 Systems Interface Description and SV-3 Systems-Systems Matrix), Resource Flows (the data in the SV-4 Systems Functionality Description and SV-6 Systems Resource Flow Matrix) and entities (in DIV-3 Physical Data Model) modeled in other models.

3.1.9 Note on System Engineering

There is not a separate set of system engineering DoDAF-described Models or Fit-for-Purpose Views since the entire DM2 could be used for a "Fit-for-Purpose" presentations. System engineers and system engineering decision-makers can use the existing DoDAF-described Models and create their own Fit-for-Purpose Views. If an existing model does not meet the purpose, the architect can select the appropriate data to create a "composite" Fit-for-Purpose View. In **Table 3.1.9-1**, a non-inclusive initial traceability of SE concepts to the DoDAF Meta-model Data Groups is below and can be the starting point for the "Fit-for-Purpose" presentations.

Also, while not inclusive of all possible SE concepts, Table 3.1.9 is not a prescribed set of data. An example of a "Fit-for-Purpose" presentation is the System Engineering charts in chapter 4.0 of the Defense Acquisition Guide which can be rendered as Gantt or Pert Charts. Each organization and their decision-makers will need to determine their own architectural data needs. System engineering efforts could be tracked as projects and have an associated WBS and be reflected in a PV-1 and PV-2.

Table 3.1.9-1: System Engineering Concepts to DoDAF Meta-model Data Groups Mapping

System Engineering Concepts	DoDAF Meta-model Data Groups
Strategies, Scenarios, Threat, Objectives, Goals	Goals
Enterprise Priorities	Goals
Capabilities (UJTLs, Business Process Analysis [BPA] Standard processes, etc.)	Capability, Activity
Operational Performance Metrics (KPPs, etc.)	Measures
Processes/Activities	Performer, Activity
Need Lines (Connectivity)	Resource Flow
Information and Information Flow (Conceptual Data Design)	Resource Flow, Data and Information
Tactics, Techniques, and Procedures	Performer, Capability
Automation, Mechanization, Material Priorities	Goals
Strategies to Process Traceability	Goals, Performer, Activity
Operational Standards (Doctrinal, Procedural, Business Rules, etc. [Joint Chiefs of Staff {JCS} Pubs, etc.])	Rules
KPP to allocated performance Traceability	Measures, Performer
Technical Standards	Rules
Process to System Function/Service Traceability	Performer, Activity
Top-level Requirement Specifications (ICD, CDD, CPD, CRD)	Capability, Services, Goals, Rules, Measures, Location, Doctrine, Training/Skill/Education, Performer, Resource Flow, Data and Information
Non-Acquisition and Acquisition WBS	Project
Cost (Training, Man Power, etc.)	Project, Measures
System Concept of Operations	Goals, Performer
System Functions	Performer, Activity
System Constraints	Rules
System Interfaces	Performer, Resource Flow, Activity
System Behavior	Performer, Activity, Rules
Trade Studies (Automation/Mechanization, Technology, commercial off the shelf [COTS], government off the shelf [GOTS], SOA, etc.) Tradeoffs	Project, Performer, Location (as in URL locations)

3.2 DoDAF V1.5 Support

The architectures for DoDAF V1.0 and DoDAF V1.5 may continue to be used. When appropriate (usually indicated by policy or by the decision-maker), DoDAF V1.X architectures will need to update their architecture. When pre-DoDAF V2.0 architecture is compared with DoDAF V2.0 architecture, concept differences (such as Node) must be defined or explained for the newer architecture.

In regard to DoDAF V1.5 products, they have been transformed into parts of the DoDAF V2.0 models. In most cases, the DoDAF V2.0 Meta-model supports the DoDAF V1.5 data concepts, with one notable exception: Node. As explained in Section 1.5 of V2.0, Node is a complex, logical concept that is represented with more concrete concepts. **Table 3.2-1** indicates the mapping of DoDAF V1.5 products to DoDAF V2.0 models.

Table 3.2-1: Mapping of DoDAF V1.5 Products to DoDAF V2.0 Models

DoDAF V2.0 / DoDAF V1.5	Operational Viewpoint	Systems Viewpoint	Services Viewpoint	All Viewpoint	Standards Viewpoint	Data & Information Viewpoint
AV-1				AV-1		
AV-2				AV-2		
OV-1	OV-1					
OV-2	OV-2					
OV-3	OV-3					
OV-4	OV-4					
OV-5	OV-5a, OV-5b					
OV-6a	OV-6a					
OV-6b	OV-6b					
OV-6c	OV-6c					
OV-7						DIV-2
SV-1		SV-1	SvcV-1			
SV-2		SV-2	SvcV-2			
SV-3		SV-3	SvcV-3a, SvcV-3b			
SV-4a		SV-4				
SV-4b			SvcV-4			
SV-5a		SV-5a				
SV-5b		SV-5b				
SV-5c			SvcV-5			
SV-6		SV-6	SvcV-6			
SV-7		SV-7	SvcV-7			
SV-8		SV-8	SvcV-8			

DoDAF V2.0 / DoDAF V1.5	Operational Viewpoint	Systems Viewpoint	Services Viewpoint	All Viewpoint	Standards Viewpoint	Data & Information Viewpoint
SV-9		SV-9	SvcV-9			
SV-10a		SV-10a	SvcV-10a			
SV-10b		SV-10b	SvcV-10b			
SV-10c		SV-10c	SvcV-10c			
SV-11						DIV-3
TV-1					StdV-1	
TV-2					StdV-2	

3.3 DoDAF Meta-model Groups Support of Viewpoints and DoD Key Processes

The DoDAF V2.0 Meta-model Groups support the viewpoints and DoD Key Processes of JCIDS, DAS, PPBE, System Engineering, Operations, and Portfolio Management (IT and Capability). Table 3.3-1 indicates a non-inclusive mapping of DoDAF Meta-model Groups to the DoDAF Viewpoints and DoD Key Processes. The support for the Key Processes is for the information requirements that were presented at the workshops for the key processes and, as such, do not reflect all of the information requirements that a key process could need.

Table 3.3-1: DoDAF Meta-model Groups Mapping to Viewpoints and DoD Key Processes

	Viewpoints	DoD Key Proceses
Metamodel Data Groups	AV, CV, DIV, OV, PV, StdV, SvcV, SV	JCIDS, DAS, PPBE, System Engineering, Operations, Portfolio Management (IT and Capability)
Performer	CV, OV, PV, StdV, SvcV, SV	J, D, P, S, O, C
Activity	OV	J, O, C
Resource Flow	OV, SvcV, SV	J, S, O
Data and Information	AV, DIV	J, D, P, S, O, C
Capability	CV, PV, SV, SvcV	J, D, P, S, O, C
Services	CV, StdV, SV	P, S, C
Project	AV, CV, PV, SvcV, SV	D, P, S, C
Training / Skill / Education	OV, SV, SvcV, StdV	J, S, O
Goals	CV, PV	J, D, P, O, C
Rules	OV, StdV, SvcV, SV	J, D, S, O
Measures	SvcV, SV	J, D, S, O, C
Location	SvcV, SV	P, S, O

APPENDIX A
ACRONYMS

This is the integrated DoDAF V2.0 acronyms and their definitions. Some have more than one definition depending on their usage; they could have a specific meaning in Architectural Descriptions as well as generic English language usage.

The collection of acronyms and their definitions are presented for the first time since the development of DoDAF V 2.0 began. The acronyms list shown here is a first draft. Assistance is requested to ensure that correct acronyms and applicable definition have been assembled.

Acronym	Definition
AV	All Viewpoint
BDA	Battle Damage Assessment
BPA	Business Process Analysis
BPM	Business Process Model
BPMN	Business Process Modeling Notation
BRAC	Base Realignment and Closure
BRM	Business Reference Model
CADM	Core Architecture Data Model
CCDR	Contractor Cost Data Reporting
CDD	Capabilities Description Document
CDM	Conceptual Data Model
CFSR	Contract Funds Status Reports
CIEL	Common Information Exchange List
CJCS	Chairman of the Joint Chiefs of Staff
CM	Configuration Management
CMMI	Capability Maturity Model® Integration
COI	Community of Interests
COMSEC	Communication Security
CONOPS	Concepts of Operations
COTS	Commercial Off The Shelf
CPD	Capability Production Document
CPM	Capability Portfolio Management
CPR	Contract Performance Reports
CRD	Capstone Requirements Document
CV	Capability Viewpoint
DAI	Defense Acquisition Initiative
DAS	Defense Acquisition System
DBMS	Data Base Management System
DBSMC	Defense Business Systems Management Committee
DDMS	Department of Defense Discovery Metadata Specification
DISR	DoD Information Technology Standards and Profile Registry
DITPR	DoD Information Technology Portfolio Repository
DIV	Data and Information Viewpoint

Acronym	Definition
DM2	DoDAF Meta-model
DoDAF	Department of Defense Architecture Framework
DOTMLPF	Doctrine, Organization, Training, Material, Leadership and Education, Personnel, and Facilities
E-R	Entity-Relationship
EA	Enterprise Architecture
ERD	Entity Relationship Diagram
EVMS	Earned Value Management System
FBI	Federal Bureau of Investigation
FEA	Federal Enterprise Architecture
GEX	Global Exchange
GML	Geospatial Markup Language
GOTS	Government Off The Shelf
GUI	Graphical User Interface
HCI	Human Computer Interface
HR	Human Resources
IA	Information Assurance
IC-ISM	Intelligence Community – Intelligence Standard Markings
ICD	Initial Capabilities Document
IDEAS	International Defence Enterprise Architecture Specification
IDEF0	Integration Definition for Function Modeling
IDL	Interface Definition Language
INFOSEC	Information Security
IP	Internet Protocol
IPT	Integrated Product Team
IRB	Investment Review Board
ISO	International Standards Organization
ISP	Information Support Plan
IT	Information Technology
JC3IEDM	Joint Consultation, Command and Control Information Exchange Data Model
JCA	Joint Capability Areas
JCIDS	Joint Capabilities Integration and Development System
JCS	Joint Chiefs of Staff
JCSFL	Joint Common System Function List
KI	Key Interface
KIP	Key Interface Profile
KPP	Key Performance Parameter
MODAF	Ministry of Defence Architecture Framework
NAF	NATO Architecture Framework
NATO	North Atlantic Treaty Organization
NCID	Net-Centric Implementation Document
OASIS	Organization for the Advancement of Structured Information Standards
OMG	Object Management Group
OOAD	Object-Oriented Analysis & Design

Acronym	Definition
OV	Operational Viewpoint
PE	Program Element
PMBOK	Project Management Body of Knowledge
POI	Program of Instruction
POM	Program Objective Memorandum
PPBE	Planning, Programming, Budgeting, and Execution
PV	Project Viewpoint
RA	Reference Architecture
RDBMS	Relational Database Management System
REST	Representational State Transfer
RIPR	Real Property Inventory Requirements
SADT	Structured Analysis and Design Technique
SBVR	Semantics of Business Vocabulary and Business Rules
SE	Systems Engineering
SEP	Systems Engineering Plan
SLA	Service Level Agreement
SMART	Specific, Measurable, Attainable, Relevant, Timely
SOA	Service Oriented Architecture
SOO	Statement of Objectives
SOW	Statement of Work
SRDR	Software Resource Data Report
SRR	Systems Requirements Review
SV	Systems Viewpoint
SWOT	Strengths, Weaknesses, Opportunities and Threats
TEMPEST	Transient Electromagnetic Pulse Emanation Standard
TLE	Target Location Error
TTP	Tactics, Techniques, and Procedures
TV	Technical Standards View
TWG	Technical Working Group
UJTL	Universal Joint Task List
UML	Unified Modeling Language
UPDM	Unified Profile for DoDAF and MODAF
URL	Universal Resource Locator
URN	Universal Resource Name
U.S.	United States
WBS	Work Breakdown Structure
XML	eXtensible Markup Language
XSD	XML Schema Definition

This page left intentionally blank

APPENDIX B
MAPPINGS TO DM2 CONCEPT

A mapping of the DM2 Concepts (classes), Associations (relationships), and Attributes to DoDAF-described Models, is shown in **Table B-1**. In the DM2 Concept, Association, or Attribute column, the Black text is a concept or attribute, the Red text is an association, and the Green Text is the security attributes in the DM2.

Table B-1: DM2 Concepts, Associations, and Attributes Mapping to DoDAF-described Models

DM2 Concept	AV-1	AV-2	OV-1	OV-2	OV-3	OV-4	OV-5a	OV-5b	OV-6a	OV-6b	OV-6c	SV-1	SV-2	SV-3	SV-4	SV-5a	SV-5b	SV-6	SV-7	SV-8	SV-9	SV-10a	SV-10b	SV-10c	SvcV-1	SvcV-2	SvcV-3a	SvcV-3b	SvcV-4	SvcV-5	SvcV-6	SvcV-7	SvcV-8	SvcV-9	SvcV-10a	SvcV-10b	SvcV-10c	StdV-1	StdV-2	PV-1	PV-2	PV-3	CV-1	CV-2	CV-3	CV-4	CV-5	CV-6	CV-7	DIV-1	DIV-2	DIV-3
Activity	n	o		n	n		n	n	o	n	n	n	n	n	n	n	o	n	n	o	o	o	n	n	n	n	n	n	n	o	n	n	o	o	o	n	n	n	n	n	n	n	n	n	n	n	n	n	n		o	o
activityChangesResource	o								o	o	o											o	o	o											o	o	o					n	n	n	n	n	n	o	o			
activityChangesResourceTypeInstanceOfMeasure	o								o	o	o											o	o	o											o	o	o					n	n	n	n	n	n	o	o			
activityPartOfCapability																	o												o													n	n	n	n	n	n	n	o			
activityPartOfCapabilityTypeInstanceOfMeasure																	o																									n	n	n	n	n	n	n	o			
activityPartOfProjectType																																								o	o				n		n					
activityPerformableUnderCondition	o			o	o			o	o	o	o				o	o	o	o	o	o	o	o	o	o				o	o		o	o	o	o	o	o	o	n	n	n	n	n	n	n	n	n	n	n	n			o
activityPerformableUnderConditionTypeInstanceOfMeasure								o	o	o	o				o		o					o	o	o											o	o	o				n	n	n	n	n	n	n	n	o			
activityPerformedByPerformer	o			o	o			o	o	o	o				o	o	o	o	o	o	o	o	o	o				o	o		o	o	o	o	o	o	o	n	n	n	o	o			o	o	o	o	n			o
activityPerformedByPerformerTypeInstanceOfMeasure	o				o			o	o	o	o				o		o		o	o	o	o	o	o					o			o	o	o	o	o	o			o	o	o			o	o	o		o			
activityPerformedByPerformerTypeInstanceOfRule								o	o	o	o				o			o				o	o	o				o			o				o	o	o	n	n								o	o				o
activityResourceOverlap				n	n			n	o	n	n	n	n	n	n			n				o	n	n	n	n	n	n			n				o	n	n			n	o	n			o	o	o	o			o	o
activityResourceOverlapTypeInstanceOfMeasure					o			o	o	o	o	o	o	o	o			o				o	o	o	o	o	o	o			o				o	o	o			o	o	o			o		o	o				
activityResourceOverlapTypeInstanceOfRule					o			o	o	o	o	o	o	o	o			o				o	o	o	o	o	o	o			o				o	o	o			o	o	o			o		o	o				o
activityTypeInstanceOfMeasureType	o				o			o	o	o	o	o	o	o	o			n	n	o	o	o	o	o	o	o					n	n	o	o	o	o	o			o	n	n	o	n	o	n	o					
activityWholeConsumingPartOfActivity				n	n		o	n	o	n	n	n	n	n	n			n				o	n	n	n	n	n	n			n				o	n	n			n	n	n			o	o	o	o				o
activityWholeProducingPartOfActivity				n	n		o	n	o	n	n	n	n	n	n			n				o	n	n	n	n	n	n			n				o	n	n			n	n	n			o	o	o	o				o
AdaptabilityMeasure		o			o			o	o	o	o	o	o	o	o			o	o	o	o	o	o	o	o	o	o	o			o	o	o	o	o	o	o					o	o	o	o	o	o	o	o			

DM2 Concept	AV-1	AV-2	OV-1	OV-2	OV-3	OV-4	OV-5a	OV-5b	OV-6a	OV-6b	OV-6c	SV-1	SV-2	SV-3	SV-4	SV-5a	SV-5b	SV-6	SV-7	SV-8	SV-9	SV-10a	SV-10b	SV-10c	SvcV-1	SvcV-2	SvcV-3a	SvcV-3b	SvcV-4	SvcV-5	SvcV-6	SvcV-7	SvcV-8	SvcV-9	SvcV-10a	SvcV-10b	SvcV-10c	StdV-1	StdV-2	PV-1	PV-2	PV-3	CV-1	CV-2	CV-3	CV-4	CV-5	CV-6	CV-7	DIV-1	DIV-2	DIV-3
Address	o											o	o					o				o	o		o	o					o				o	o																
Agreement	o	o						o	o	o	o																											o	o	o					o	o	o					
axesDescribedBy									o			o	o									o			o	o									o										o		o		o			
beforeAfter	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
beforeAfterPowertypeInstanceOfBeforeAfterType	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
beforeAfterType	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
Capability		o															o												o													n	n	n	n	n	n	n	n			
capabilityPerformerManifestation																	o																									o			o	o	o	o	n			
CircularArea									o			o	o									o			o	o									o										o		o		o			
circularAreaPowertypeInstanceOfCircularAreaType									o			o	o									o			o	o									o										o		o		o			
CircularAreaType									o			o	o									o			o	o									o										o		o		o			
Condition	o	o		o	o			o	o	o	o				o	o	o	o	o	o	o	o	o	o				o	o		o	o	o	o	o	o	o	n	n	n	n	n	n	n	n	n	n	n	n			o
conditionTypeInstanceOfMeasure		o			o			o	o	o	o				o		o	o	o	o	o	o	o	o					o		o	o	o	o	o	o	o	o	o		o	n	n	n	n	n	n	n	o			
Constraint	o	o							o	o												o	o												o	o		o	o		o				o	o					o	o
ConsumingPartOfActivity		o		n	n		o	n	o	n	n	n	n	n	n			n				o	n	n	n	n	n	n	o	o	n	o	o	o	o	n	n	o	o	n	n	n			o	o	o	o	o		o	o
coordinateCenterDescribedBy									o			o	o									o			o	o									o										o		o		o			
Country		o							o			o	o									o			o	o									o										o		o		o			
countryPowertypeInstanceOfCountryType									o			o	o									o			o	o									o										o		o		o			
CountryType									o			o	o									o			o	o									o										o		o		o			
couple	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
couplePowertypeInstanceOfCoupleType	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
coupleType	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f

DM2 Concept	AV-1	AV-2	OV-1	OV-2	OV-3	OV-4	OV-5a	OV-5b	OV-6a	OV-6b	OV-6c	SV-1	SV-2	SV-3	SV-4	SV-5a	SV-5b	SV-6	SV-7	SV-8	SV-9	SV-10a	SV-10b	SV-10c	SvcV-1	SvcV-2	SvcV-3a	SvcV-3b	SvcV-4	SvcV-5	SvcV-6	SvcV-7	SvcV-8	SvcV-9	SvcV-10a	SvcV-10b	SvcV-10c	StdV-1	StdV-2	PV-1	PV-2	PV-3	CV-1	CV-2	CV-3	CV-4	CV-5	CV-6	CV-7	DIV-1	DIV-2	DIV-3
Data		o													n			n		o		o	o	o				n			n		o		o	o	o	o	o						o	o	o		o		n	n
dataPowertypeInstanceOfDataType																																																			n	n
DataType																																																			n	n
describedBy	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
DesiredEffect	n								o											o	o	o											o	o	o					o	o	n	n	n	n	n	n	n	n			
desiredEffectDirectsActivity																																								o	o	o										
desiredEffectGuidesActivity																																								o	o	o	o	o	o	o	o	o	o			
desiredEffectIsRealizedByProjectType																																								o	o	o										
desiredEffectPartOfCapability																																										n	n	n	n	n	n	n	n			
desiredEffectTypeInstanceOfMeasure									o											o	o	o											o	o	o					o	o	n	n	n	n	n	n	n	n			
disjoint	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	o	f	o	f	f	f	f	f
DomainInformation		o						n		o																																o	o	o	o	o	o	o	o	n	o	o
EffectsMeasure	o	o			o			o	o	o	o	o	o	o	o			o	o	o	o	o	o	o	o	o	o	o			o	o	o	o	o	o	o			o	o	n	n	n	n	n	n	n	n			
EllipticalArea									o			o	o									o			o	o									o										o		o		o			
ellipticalAreaPowertypeInstanceOfEllipticalAreaType									o			o	o									o			o	o									o										o		o		o			
EllipticalAreaType									o			o	o									o			o	o									o										o		o		o			
endBoundary	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	n	f	n	f	f	f	f	f
endBoundaryPowertypeInstanceOfEndBoundaryType	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
endBoundaryType	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	n	f	n	f	f	f	f	f
endBoundaryTypeInstanceOfMeasure	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
endBoundaryTypeTypeInstanceOfMeasure	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
Facility	o	o							o			o	o									o			o	o									o										o		o		o			

DM2 Concept	AV-1	AV-2	OV-1	OV-2	OV-3	OV-4	OV-5a	OV-5b	OV-6a	OV-6b	OV-6c	SV-1	SV-2	SV-3	SV-4	SV-5a	SV-5b	SV-6	SV-7	SV-8	SV-9	SV-10a	SV-10b	SV-10c	SvcV-1	SvcV-2	SvcV-3a	SvcV-3b	SvcV-4	SvcV-5	SvcV-6	SvcV-7	SvcV-8	SvcV-9	SvcV-10a	SvcV-10b	SvcV-10c	StdV-1	StdV-2	PV-1	PV-2	PV-3	CV-1	CV-2	CV-3	CV-4	CV-5	CV-6	CV-7	DIV-1	DIV-2	DIV-3
facilityPartOfSite	o								o			o	o									o			o	o									o										o		o		o			
facilityPowertypeInstanceOfFacilityType	o	o							o			o	o									o			o	o									o										o		o		o			
FacilityType	o	o							o			o	o									o			o	o									o										o		o		o			
FunctionalStandard		o			o			o				o	o		o					o		o			o	o	o	o	o		o	o	o	o	o			o	o					o	o	o	o		o			
GeoFeature		o							o			o	o									o			o	o									o										o		o		o			
geoFeaturePowertypeInstanceOfGeoFeatureType		o							o			o	o									o			o	o									o										o		o		o			
GeoFeatureType		o							o			o	o									o			o	o									o										o		o		o			
GeoPoliticalExtent	o	o							o			o	o									o			o	o									o										o		o		o			
geoPoliticalExtentPowertypeInstanceOfGeoPoliticalExtentType	o	o							o			o	o									o			o	o									o										o		o		o			
GeoPoliticalExtentType	o	o							o			o	o									o			o	o									o										o		o		o			
GeoStationaryPoint									o			o	o									o			o	o									o										o		o		o			
geoStationaryPointPowertypeInstanceOfGeoStationaryPointType									o			o	o									o			o	o									o										o		o		o			
GeoStationaryPointType									o			o	o									o			o	o									o										o		o		o			
Guidance	o	o			n			o	n	o	o	o	n	o	o	o	o	n	o	o	o	n	o	o	o	n	o	o	o	o	n	o	o	o	n	o	o	n	n	o	o	o	o	o	n	o	n	o	o	o	o	o
Individual	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
IndividualActivity	n																																								n	o										
individualActivityPowertypeInstanceOfActivity	n																																								n	o										
IndividualDesiredEffect	n																																								o	o										
individualDesiredEffectPowertypeInstanceOfDesiredEffect	n																																								o	o										
IndividualPerformer	n	o		o	o	o		o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	n	o	o		o	o	o	o	o	o		o	o
individualPerformerPowertypeInstanc	n	o		o	o	o		o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	n	o	o		o	o	o	o	o	o		o	o

DM2 Concept	AV-1	AV-2	OV-1	OV-2	OV-3	OV-4	OV-5a	OV-5b	OV-6a	OV-6b	OV-6c	SV-1	SV-2	SV-3	SV-4	SV-5a	SV-5b	SV-6	SV-7	SV-8	SV-9	SV-10a	SV-10b	SV-10c	SvcV-1	SvcV-2	SvcV-3a	SvcV-3b	SvcV-4	SvcV-5	SvcV-6	SvcV-7	SvcV-8	SvcV-9	SvcV-10a	SvcV-10b	SvcV-10c	StdV-1	StdV-2	PV-1	PV-2	PV-3	CV-1	CV-2	CV-3	CV-4	CV-5	CV-6	CV-7	DIV-1	DIV-2	DIV-3
eOfPerformer																																																				
individualPowertypeInstanceOfIndividualType	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
IndividualResource	n	o		o	o	o		o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	n	o	o		o	o	o	o	o	o		o	o
individualResourcePowertypeInstanceOfResource	n	o		o	o	o		o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	n	o	o		o	o	o	o	o	o		o	o
IndividualType	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	n	f	n	f	f	f	f	f
Information	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
informationAssociation															o			o				o	o	o				o			o				o	o	o								o	o	o			o	n	n
informationPowertypeInstanceOfInformationType	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
InformationType	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
Installation		o							o			o	o									o			o	o									o										o		o		o			
installationPowertypeInstanceOfInstallationType		o							o			o	o									o			o	o									o										o		o		o			
InstallationType		o							o			o	o									o			o	o									o										o		o		o			
intersection	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	o	f	o	f	f	f	f	f
Line		o							o			o	o									o			o	o									o										o		o		o			
linePartOfPlanarSurface									o			o	o									o			o	o									o										o		o		o			
linePowertypeInstanceOfLineType		o							o			o	o									o			o	o									o										o		o		o			
LineType		o							o			o	o									o			o	o									o										o		o		o			
Location	o	o							o			o	o					o				o	o		o	o					o				o	o									o		o		o			
locationNamedByAddress	o											o	o					o				o	o		o	o					o				o	o																
locationPowertypeInstanceOfLocationType	o	o							o			o	o									o			o	o									o										o		o		o			
LocationType	o	o							o			o	o				o					o			o	o									o									o	o	o	o	o	o			

DM2 Concept	AV-1	AV-2	OV-1	OV-2	OV-3	OV-4	OV-5a	OV-5b	OV-6a	OV-6b	OV-6c	SV-1	SV-2	SV-3	SV-4	SV-5a	SV-5b	SV-6	SV-7	SV-8	SV-9	SV-10a	SV-10b	SV-10c	SvcV-1	SvcV-2	SvcV-3a	SvcV-3b	SvcV-4	SvcV-5	SvcV-6	SvcV-7	SvcV-8	SvcV-9	SvcV-10a	SvcV-10b	SvcV-10c	StdV-1	StdV-2	PV-1	PV-2	PV-3	CV-1	CV-2	CV-3	CV-4	CV-5	CV-6	CV-7	DIV-1	DIV-2	DIV-3
MaintainabilityMeasure	o	o			o			o	o	o	o	o	o	o	o			o	o	o	o	o	o	o	o	o	o	o			o	n	o	o	o	o	o			o	o	o	o	o	o	o	o	o	o			
materialPartOfSystem												o	o							o	o	o			o	o							o	o	o						o				o	o	o					
Materiel		o										o	o							o	o	o			o	o							o	o	o						o				o	o	o					
Measure	n	o		o	n			o	o	n	n	o	o	o	o	o	o	n	n	o	o	o	n	n	o	o	o	o	o		n	n	o	o	o	n	n	o	o	o	n	n	n	n	n	n	n	n	n	o	o	
measurePowertypeInstanceOfMeasureType	n	o			n			o	o	n	n	o	o	o	o			n	n	o	o	o	n	n	o	o	o	o			n	n	o	o	o	n	n			o	n	n	n	n	n	n	n	n	n			
MeasureType	n	o			n			o	o	n	n	o	o	o	o			n	n	o	o	o	n	n	o	o	o	o			n	n	o	o	o	n	n	o	o	o	n	n	n	n	n	n	n	n	n			
Name	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
namedBy	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
namePowertypeInstanceOfNamingScheme	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
NamingScheme	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
NeedsSatisfactionMeasure	o	o			n			o	o	o	o	o	o	o	o			n	o	o	o	o	o	o	o	o	o	o			n	o	o	o	o	o	o			o	o	o	o	o	o	o	o	o	o			
Organization	o	o		o	o	o		o	o	o	o	o													o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	n	o				o	o	o				o	o
OrganizationalMeasure	o	o			o			o	o	o	o	o													o	o	o	o			o	o	o	o	o	o	o			o	o	o	o	o	o	o	o	o	o			
organizationPowertypeInstanceOfOrganizationType		o		o	o	o		o	o	o	o	o													o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o				o	o	o				o	o
OrganizationType		o		o	o	o		o	o	o	o	o													o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o				o	o	o				o	o
overlap	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	n	f	n	f	f	f	f	f
overlapPowertypeInstanceOfOverlapType	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
overlapType	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
PerformanceMeasure	o	o			n			o	o	o	o	o	o	o	o			n	n	o	o	o	o	o	o	o	o	o			n	n	o	o	o	o	o			o	o	o	o	o	o	o	o	o	o			
Performer	n	o		n	n	n		o	o	o	o	n	n	n	o	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	o	o	o	o	o	o	n		o	o
performerPerformsAtLocationType																	o																											o	o	o	o	o	o			
PersonType	o	o			o			o	o	o	o	o	o	o	o			o	n	o	o	o	o	o	o	o	o	o			o	n	o	o	o	o	o	o	o	o	n	n		o	o	o	o	o	o			

DM2 Concept	AV-1	AV-2	OV-1	OV-2	OV-3	OV-4	OV-5a	OV-5b	OV-6a	OV-6b	OV-6c	SV-1	SV-2	SV-3	SV-4	SV-5a	SV-5b	SV-6	SV-7	SV-8	SV-9	SV-10a	SV-10b	SV-10c	SvcV-1	SvcV-2	SvcV-3a	SvcV-3b	SvcV-4	SvcV-5	SvcV-6	SvcV-7	SvcV-8	SvcV-9	SvcV-10a	SvcV-10b	SvcV-10c	StdV-1	StdV-2	PV-1	PV-2	PV-3	CV-1	CV-2	CV-3	CV-4	CV-5	CV-6	CV-7	DIV-1	DIV-2	DIV-3
personTypePartOfSystem	o											o	o												o	o												o	o		o			o	o	o	o	o	o			
PhysicalMeasure	n	o			n			o	o	n	n	o	o	o	o			n	o	o	o	o	n	n	o	o	o	o			n	o	o	o	o	n	n			o	n	o	o	o	n	n	n	o	o			
PlanarSurface									o			o	o									o			o	o									o										o		o		o			
planarSurfacePowertypeInstanceOfPlanarSurfaceType									o			o	o									o			o	o									o										o		o		o			
PlanarSurfaceType		o							o			o	o									o			o	o									o										o		o		o			
Point									o			o	o									o			o	o									o										o		o		o			
pointPartOfLine									o			o	o									o			o	o									o										o		o		o			
pointPartOfPlanarSurface									o			o	o									o			o	o									o										o		o		o			
PointPowertypeInstanceOfPointType									o			o	o									o			o	o									o										o		o		o			
PointType									o			o	o									o			o	o									o										o		o		o			
pointTypeInstanceOfMeasure									o			o	o									o			o	o									o										o		o		o			
PolygonArea									o			o	o									o			o	o									o										o		o		o			
polygonAreaPowertypeInstanceOfPolygonAreaType									o			o	o									o			o	o									o										o		o		o			
PolygonAreaType									o			o	o									o			o	o									o										o		o		o			
Port		o											n		o			o							n	n	n	n	n	n	n	n	n	n	n	n	n	o	o		o	o			o	o	o		o			o
portPartOfPerformer													n		o			o								n		o			o							o	o		o	o			o	o	o		o			o
PositionReferenceFrame									o			o	o									o			o	o									o										o		o		o			
Powertype	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
powertypeInstance	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
ProducingPartOfActivity		o		n	n		o	n	o	n	n	n	n	n	n			n				o	n	n	n	n	n	n	o	o	n	o	o	o	o	n	n	o	o	n	n	n			o	o	o	o	o		o	o
Project	n																																							n	n	n			o	o	o					
projectPowertypeInstanceOfProjectType	n																																							n	n	n			o	o	o					

DM2 Concept	AV-1	AV-2	OV-1	OV-2	OV-3	OV-4	OV-5a	OV-5b	OV-6a	OV-6b	OV-6c	SV-1	SV-2	SV-3	SV-4	SV-5a	SV-5b	SV-6	SV-7	SV-8	SV-9	SV-10a	SV-10b	SV-10c	SvcV-1	SvcV-2	SvcV-3a	SvcV-3b	SvcV-4	SvcV-5	SvcV-6	SvcV-7	SvcV-8	SvcV-9	SvcV-10a	SvcV-10b	SvcV-10c	StdV-1	StdV-2	PV-1	PV-2	PV-3	CV-1	CV-2	CV-3	CV-4	CV-5	CV-6	CV-7	DIV-1	DIV-2	DIV-3
ProjectType	n																																							n	n	n			n	o	n					
projectTypeTypeInstanceOfMeasure	o																																							o	o	o			o	o	o					
RealProperty	o	o							o			o	o									o			o	o									o										o		o		o			
realPropertyPowertypeInstanceOfRealPropertyType	o	o							o			o	o									o			o	o									o										o		o		o			
RealPropertyType	o	o							o			o	o									o			o	o									o										o		o		o			
RectangularArea									o			o	o									o			o	o									o										o		o		o			
rectangularAreaPowertypeInstanceOfRectangularAreaType									o			o	o									o			o	o									o										o		o		o			
RectangularAreaType									o			o	o									o			o	o									o										o		o		o			
RegionOfCountry									o			o	o									o			o	o									o										o		o		o			
regionOfCountryPartOfCountry									o			o	o									o			o	o									o										o		o		o			
regionOfCountryPowertypeInstanceOfRegionOfCountryType									o			o	o									o			o	o									o										o		o		o			
RegionOfCountryType									o			o	o									o			o	o									o										o		o		o			
RegionOfWorld									o			o	o									o			o	o									o										o		o		o			
Resource	n	o		n	n	n		n	o	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	o	n	n	n	n
resourceTypeInstanceOfMeasure	o			o	o			o	o	o	o	o	o	o	o	o		o	o	o	o	o	o	o	o	o	o	o			o	o	o	o	o	o	o			o	o	o			o	o	o	o	o	o	o	
Rule	o	o			n			o	o	o	o	o	n	o	o	o	o	n	o	o	o	o	o	o	o	n	o	o	o	o	n	o	o	o	o	o	o	n	n	o	o	o	o	o	n	o	n	o	o	o	o	o
ruleConstrainsActivity	o				n			o	o	o	o				o	o		o	o	o		o	o	o				o	o		o	o	o		o	o	o	o	o		o	o	o	o	o	o	o	o	o			
ruleConstraintOfActivityValidUnderCondition	o				o				o	o	o				o			o		o		o	o	o				o			o		o		o	o	o	o	o		o	o	o	o	o	o	o	o	o			
rulePartOfMeasureType	o	o			o			o	o	o	o	o	o	o	o			o	o	o	o	o	o	o	o	o	o	o			o	o	o	o	o	o	o			o	o	o	o	o	o	o	o	o	o			
SecurityAttributesGroup	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s
Service		o																							n	n	n	n	n	n	n	n	n	n	n	n	n	o	o		o	o	o	o	o	o	o	o	n			o
serviceChannel		o																							o	o	o	o	o	o	o	o	o	o	o	o	o	o	o		o				o	o	o		o			

DM2 Concept	AV-1	AV-2	OV-1	OV-2	OV-3	OV-4	OV-5a	OV-5b	OV-6a	OV-6b	OV-6c	SV-1	SV-2	SV-3	SV-4	SV-5a	SV-5b	SV-6	SV-7	SV-8	SV-9	SV-10a	SV-10b	SV-10c	SvcV-1	SvcV-2	SvcV-3a	SvcV-3b	SvcV-4	SvcV-5	SvcV-6	SvcV-7	SvcV-8	SvcV-9	SvcV-10a	SvcV-10b	SvcV-10c	StdV-1	StdV-2	PV-1	PV-2	PV-3	CV-1	CV-2	CV-3	CV-4	CV-5	CV-6	CV-7	DIV-1	DIV-2	DIV-3
ServiceDescription																									n	n	n	n	n	n	n	n	n	n	n	n	n	o	o		o		o	o	o	o	o		n			o
serviceEnablesAccessTo																									o	o	o	o	o	o	o	o	o	o	o	o	o						o	o	o	o	o		n			
ServiceLevel	o	o			n			o	o	o	o	o	o	o	o			n	n	o	o	o	o	o	o	o	o	o			n	n	o	o	o	o	o	o	o	o	n	o	o	n	o	n	o	n	n			
ServicePort		o																							n	n	n	n	n	n	n	n	n	n	n	n	n	o	o		o				o	o	o		o			o
Site	o								o			o	o									o			o	o									o										o		o		o			
sitePartOfInstallation									o			o	o									o			o	o									o										o		o		o			
sitePowertypeInstanceOfSiteType									o			o	o									o			o	o									o										o		o		o			
SiteType									o			o	o									o			o	o									o										o		o		o			
Skill		o			o			o	o	o	o	o	o	o	o			o	n	o	o	o	o	o	o	o	o	o			o	n	o	o	o	o	o	o	o	o	n	n		o	o	o	o	o	o			
skillPartOfPersonType					o			o	o	o	o	o	o	o	o			o	n	o	o	o	o	o	o	o	o	o			o	n	o	o	o	o	o	o	o	o	n	n		o	o	o	o	o	o			
skillPartOfPersonTypeTypeInstanceOfMeasure					o			o	o	o	o	o	o	o	o			o	n	o	o	o	o	o	o	o	o	o			o	n	o	o	o	o	o			o	n	n		o	o	o	o	o	o			
SolidVolume									o			o	o									o			o	o									o										o		o		o			
solidVolumePowertypeInstanceOfSolidVolumeType									o			o	o									o			o	o									o										o		o		o			
SolidVolumeType									o			o	o									o			o	o									o										o		o		o			
SpatialMeasure									o			o	o									o			o	o									o										o		o		o			
Standard		o			o			o				o	n		o		o	n		o	o	o			o	n	o	o	o	o	n	o	o	o	o			n	n					o	o	o	o		o	o	o	o
startBoundary	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	n	f	n	f	f	f	f	f
startBoundaryPowertypeInstanceOfStartBoundaryType	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
startBoundaryType	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	n	f	n	f	f	f	f	f
startBoundaryTypeInstanceOfMeasure	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
startBoundaryTypeTypeInstanceOfMeasure	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f

DM2 Concept	AV-1	AV-2	OV-1	OV-2	OV-3	OV-4	OV-5a	OV-5b	OV-6a	OV-6b	OV-6c	SV-1	SV-2	SV-3	SV-4	SV-5a	SV-5b	SV-6	SV-7	SV-8	SV-9	SV-10a	SV-10b	SV-10c	SvcV-1	SvcV-2	SvcV-3a	SvcV-3b	SvcV-4	SvcV-5	SvcV-6	SvcV-7	SvcV-8	SvcV-9	SvcV-10a	SvcV-10b	SvcV-10c	StdV-1	StdV-2	PV-1	PV-2	PV-3	CV-1	CV-2	CV-3	CV-4	CV-5	CV-6	CV-7	DIV-1	DIV-2	DIV-3
superSubType	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	o	n	f	f	f	f	f	f	f	f
Surface									o			o	o									o			o	o									o										o		o		o			
surfacePowertypeInstanceOfSurfaceType									o			o	o									o			o	o									o										o		o		o			
SurfaceType		o							o			o	o									o			o	o									o										o		o		o			
System	o	o										n	n	n	o		n	n	n	n	n	n	n	n	o	o	n	o			o	o	o	o	o	o	o	o	o	o	o	o		o	o	o	o	o	o			o
TechnicalStandard		o			o							o	n		o		o	n		o	o	o			o	n	o	o		o	n		o	o	o			n	n					o	o	o	o		o	o	o	o
temporalBoundary	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	n	f	n	f	f	f	f	f
temporalBoundaryPowertypeInstanceOfTemporalBoundaryType	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
temporalBoundaryType	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	n	f	n	f	f	f	f	f
TemporalMeasure	n				n			o	o	n	n	o	o	o	o			n	o	o	o	o	n	n	o	o	o	o			n	o	o	o	o	n	n			o	n	o	o	o	n	n	n	o	o			
temporalWholePart	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	n	f	n	f	f	f	f	f
temporalWholePartPowertypeInstanceOfTemporalWholePartType	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
temporalWholePartType	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	n	f	n	f	f	f	f	f
Thing	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
tuple	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
tuplePowertypeInstanceOfTupleType	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
TupleType	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
Type	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
typeInstance	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
union	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	o	f	o	f	f	f	f	f
Vision	o																			o	o												o	o					o	o	o	o	o									
visionIsRealizedByDesiredEffect	o																			o	o												o	o						o	o	o										
wholePart	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f

DM2 Concept	AV-1	AV-2	OV-1	OV-2	OV-3	OV-4	OV-5a	OV-5b	OV-6a	OV-6b	OV-6c	SV-1	SV-2	SV-3	SV-4	SV-5a	SV-5b	SV-6	SV-7	SV-8	SV-9	SV-10a	SV-10b	SV-10c	SvcV-1	SvcV-2	SvcV-3a	SvcV-3b	SvcV-4	SvcV-5	SvcV-6	SvcV-7	SvcV-8	SvcV-9	SvcV-10a	SvcV-10b	SvcV-10c	StdV-1	StdV-2	PV-1	PV-2	PV-3	CV-1	CV-2	CV-3	CV-4	CV-5	CV-6	CV-7	DIV-1	DIV-2	DIV-3
wholePartPowertypeInstanceOfWholePartType	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
wholePartType	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f	f
wholePartTypeInstanceOfMeasure					o			o	o	o	o	o	o	o	o			n	n	o	o	o	o	o	o	o	o	o			n	n	o	o	o	o	o			o	n	n	o	n	o	n	o	n	n			
Classification	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s
classificationPartOfSecurityAttributesGroup																																																				
ClassificationReason	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s
classificationReasonPartOfSecurityAttributesGroup																																																				
ClassificationType	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s
ClassifiedBy	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s
classifiedByPartOfSecurityAttributesGroup																																																				
DateOfExemptedSource	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s
dateOfExemptedSourcePartOfSecurityAttributesGroup																																																				
DeclassDate	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s
declassDatePartOfSecurityAttributesGroup																																																				
DeclassEvent	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s
declassEventPartOfSecurityAttributesGroup																																																				
DeclassException	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s
declassExceptionPartOfSecurityAttributesGroup																																																				
DeclassManualReview	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s

DM2 Concept	AV-1	AV-2	OV-1	OV-2	OV-3	OV-4	OV-5a	OV-5b	OV-6a	OV-6b	OV-6c	SV-1	SV-2	SV-3	SV-4	SV-5a	SV-5b	SV-6	SV-7	SV-8	SV-9	SV-10a	SV-10b	SV-10c	SvcV-1	SvcV-2	SvcV-3a	SvcV-3b	SvcV-4	SvcV-5	SvcV-6	SvcV-7	SvcV-8	SvcV-9	SvcV-10a	SvcV-10b	SvcV-10c	StdV-1	StdV-2	PV-1	PV-2	PV-3	CV-1	CV-2	CV-3	CV-4	CV-5	CV-6	CV-7	DIV-1	DIV-2	DIV-3
declassManualReviewPartOfSecurityAttributesGroup																																																				
DerivedFrom	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S
derivedFromPartOfSecurityAttributesGroup																																																				
DisseminationControls	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S
disseminationControlsPartOfSecurityAttributesGroup																																																				
FGIsourceOpen	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S
FGIsourceOpenPartOfSecurityAttributesGroup																																																				
FGIsourceProtected	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S
FGIsourceProtectedPartOfSecurityAttributesGroup																																																				
NonICmarkings	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S
nonICmarkingsPartOfSecurityAttributesGroup																																																				
OwnerProducer	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S
ownerProducerPartOfSecurityAttributesGroup																																																				
ReleasableTo	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S
releasableToPartOfSecurityAttributesGroup																																																				
SARIdentifier	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S
SARIdentifierPartOfSecurityAttributesGroup																																																				
SCIControls	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S
SCIControlsPartOfSecurityAttributesGroup																																																				

DM2 Concept	AV-1	AV-2	OV-1	OV-2	OV-3	OV-4	OV-5a	OV-5b	OV-6a	OV-6b	OV-6c	SV-1	SV-2	SV-3	SV-4	SV-5a	SV-5b	SV-6	SV-7	SV-8	SV-9	SV-10a	SV-10b	SV-10c	SvcV-1	SvcV-2	SvcV-3a	SvcV-3b	SvcV-4	SvcV-5	SvcV-6	SvcV-7	SvcV-8	SvcV-9	SvcV-10a	SvcV-10b	SvcV-10c	StdV-1	StdV-2	PV-1	PV-2	PV-3	CV-1	CV-2	CV-3	CV-4	CV-5	CV-6	CV-7	DIV-1	DIV-2	DIV-3
TypeOfExemptedSource	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s	s
typeOfExemptedSourcePartOfSecurityAttributesGroup																																																				
ArchitecturalDescription	n	o	n																																										n		n					
ArchitectureOverviewAndPurpose	n	o																																																		
IndividualPerson	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m
informationPedigree	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m	m

Legend:
N = necessary data in this DoDAF-described Model
O = optional data in this DoDAF-described Model
Blank = cannot be a part of this DoDAF-described Model
F = IDEAS foundation common to all DoDAF-described Models
S = Classification markings common to all DoDAF-described Models
M = Metadata common to all DoDAF-described Models

Light Green background indicates this concept pertains to Architecture Metadata.
Grey background with green text indicates this concept is a security classification markings concept.

Table B-2 indicates the alignment of the CADM independent entities (supertype or parent) to the DM2 data elements. The dependent entities (subtype entities or children) will map to the same DM2 data elements as their supertype entity or parent entity.

Table B-2: Mapping of CADM Independent Entities to DM2 Data Elements

CADM Entity Name	CADM Entity Definition	DM2 Mappings	Mapping Notes
ACTION	(325/1) (A) AN ACTIVITY.	Activity	
ACTION-VERB	(11373/1) (A) A FUNCTION TO BE PERFORMED.	Activity	
ACTIVITY-MODEL-INFORMATION-ELEMENT-ROLE	(4182/2) (A) THE ROLE ASSIGNED TO AN INFORMATION-ELEMENT FOR A PROCESS-ACTIVITY IN A SPECIFIC ACTIVITY-MODEL.	N/A -- model artifact	
ACTIVITY-MODEL-PROCESS-ACTIVITY	(4188/3) (A) THE ASSOCIATION OF AN ACTIVITY-MODEL WITH A PROCESS-ACTIVITY.	describedBy	
ACTIVITY-MODEL-THREAD	(20160/1) (A) A PATH IN AN ACTIVITY-MODEL CONSISTING OF SEQUENTIAL INFORMATION FLOWS FROM ONE PROCESS-ACTIVITY TO ANOTHER.	Activity, activityResourceOverlap, beforeAfter	
AGREEMENT	(332/1) (A) AN ARRANGEMENT BETWEEN PARTIES.	Agreement	
ANTENNA-TYPE	(6542/2) (A) THE CLASSIFICATION OF A DEVICE FOR THE COLLECTION OR RADIATION OF ELECTROMAGNETIC SIGNALS.	Materiel and powerType/superSubType	COI extension
ARCHITECTURE	(19524/1) (A) THE STRUCTURE OF COMPONENTS, THEIR RELATIONSHIPS, AND THE PRINCIPLES AND GUIDELINES GOVERNING THEIR DESIGN AND EVOLUTION OVER TIME.	ArchitectureInformation	
ARCHITECTURE-ORGANIZATION	(19546/1) (A) THE RELATION OF AN ARCHITECTURE TO A SPECIFIC ORGANIZATION.	informationPedigree	
BATTLEFIELD-FUNCTIONAL-AREA-PROPONENT	(19563/1) (A) A DISCRETE AREA OF RESPONSIBILITY READILY IDENTIFIABLE BY FUNCTION PERFORMED WHICH CONTRIBUTES DIRECTLY TO BATTLEFIELD MANAGEMENT.	activityPerformerOverlap	COI extension
BUSINESS-SUBFUNCTION	(22594/1) (A) THE LOWER-LEVEL SET OF FUNCTIONS PERFORMED BY THE FEDERAL GOVERNMENT FOR A SPECIFIC LINE-OF-BUSINESS.	Activity, powerType/superSubType, wholePart	
CAPABILITY	(333/1) (A) AN ABILITY TO ACHIEVE AN OBJECTIVE.	Measure	
CAPABILITY-CATEGORY	(22750/1) (A) THE CLASS OF A CAPABILITY.	MeasureType	
COMMUNICATION-CIRCUIT	(19575/1) (A) A PATH USED FOR TRANSMITTING DATA.	System, Activity, beforeAfter	COI extension
COMMUNICATION-CIRCUIT-TYPE	(19576/1) (A) A KIND OF PATH USED FOR TRANSMITTING DATA.	System	COI extension
COMMUNICATION-LINK-TYPE	(19579/1) (A) A GENERIC KIND OF COMMUNICATION-LINK.	System and powerType/superSubType	COI extension
COMMUNICATION-MEANS	(19580/1) (A) A PHYSICAL OR ELECTROMAGNETIC INSTANTIATION OF TELECOMMUNICATIONS.	System	COI extension
COMMUNICATION-MEDIUM	(19582/1) (A) A MODE OF DATA TRANSMISSION.	Systems and overlap parts	COI extension
COMMUNICATION-SPACE-USE-CLASS	(19585/1) (A) THE SPECIFICATION OF CATEGORIES OF UTILIZATION OF SPACE FOR TELECOMMUNICATION IN BUILDINGS AND OTHER FACILITIES.	Activity, Peformer, and performerTypeInstanceLocation	COI extension
COST-BASIS	(19590/1) (A) THE SPECIFICATION USED TO DETERMINE AN UNDERLYING EXPENSE.	MeasureType	
COUNTRY	(39/1) (A) A NATION OF THE WORLD.	Country	
DATA-ITEM-TYPE	(19595/1) (A) A KIND OF DATA-ITEM.	Data and powerType/superSubType	
DATA-REFERENCE	A SELECTION OF INSTANCES OF DATA THAT ARE FORMALLY CONTROLLED FOR DOD USE.	Data and Rule	Policy requirement
DECISION-MILESTONE	(20170/1) (A) A DECISION POINT THAT SEPARATES THE PHASES OF A DIRECTED, FUNDED EFFORT THAT IS DESIGNED TO PROVIDE A NEW OR IMPROVED MATERIAL CAPABILITY IN RESPONSE TO A VALIDATED NEED.	Activity	

CADM Entity Name	CADM Entity Definition	DM2 Mappings	Mapping Notes
DEFENSE-OCCUPATIONAL-SPECIALTY-CROSS-REFERENCE	(22526/1) (C) THE RELATIONSHIP OF THE DEPARTMENT OF DEFENSE OCCUPATIONAL CONVERSIONS TO SERVICE-SPECIFIC OCCUPATIONAL SPECIALTIES.	Skill	
DEPLOYMENT-LOCATION-TYPE	(19596/1) (A) THE CHARACTERIZATION OF A KIND OF GENERIC PLACE FOR DEPLOYED OPERATIONS.	Condition	
DISCOVERY-METADATA	(22757/1) (A) SPECIFICATION OF THE MEANING OF THE ATTRIBUTES OF ANY ENTITY THAT IS COMPRISED OF DATA.	powertype of Information	
DOCUMENT	(119/1) (A) RECORDED INFORMATION REGARDLESS OF PHYSICAL FORM.	Information	
EVENT	(49/1) (A) A SIGNIFICANT OCCURRENCE.	Activity	
EVENT-NODE-CROSS-LINK	(19978/1) (A) THE SPECIFICATION OF HOW A SPECIFIC EVENT FOR A SPECIFIC ORIGINATOR NODE TEMPORALLY RELATES TO ANOTHER TERMINATOR NODE SUBJECT TO A CONSTRAINT.	Activity, beforeAfter, temporalWholePart, overlap	
EVENT-TYPE	(12341/1) (A) A CATEGORY OF EVENT.	Activity and powerType/superSubType	
EXCHANGE-RELATIONSHIP-TYPE	(19608/1) (A) THE SPECIFICATION OF A CLASS OF PAIRING FOR INFORMATION EXCHANGE.	activityResourceOverlap and powerType/superSubType	
FACILITY	(334/1) (A) REAL PROPERTY, HAVING A SPECIFIED USE, THAT IS BUILT OR MAINTAINED BY PEOPLE.	Facility	
FACILITY-CLASS	(5742/1) (A) THE HIGHEST LEVEL OF REAL PROPERTY CLASSIFICATION BY THE DEPARTMENT OF DEFENSE.	Facility and powerType/superSubType	
FACILITY-IMPROVEMENT-ACTIVITY	(19541/1) (A) A PROCESS TO IMPROVE CAPABILITIES FOR A SPECIFIC FACILITY.	Project	COI extension
FACILITY-TYPE	(50/1) (A) A SPECIFIC KIND OF FACILITY.	Facility and powerType/superSubType	
FEDERAL-SERVICE-COMPONENT	(22751/1) (A) A SELF-CONTAINED BUSINESS PROCESS OR SERVICE WITH PREDETERMINED FUNCTIONALITY THAT MAY BE EXPOSED THROUGH A BUSINESS OR TECHNOLOGY INTERFACE.	Service	
FEDERAL-SERVICE-COMPONENT-TYPE	(22752/1) (A) A HIGH LEVEL CATEGORIZATION OF BUSINESS CAPABILITIES. Note: IT IS A BUILDING BLOCK OF THE FEDERAL ENTERPRISE ARCHITECTURE SERVICE COMPONENT REFERENCE MODEL, WHICH IS A COMPONENT-BASED FRAMEWORK THAT PROVIDES--INDEPENDENT OF BUSINESS FUNCTION--A LEVERAGEABLE FOUNDATION TO SUPPORT THE REUSE OF APPLICATIONS, APPLICATION CAPABILITIES, COMPONENTS, AND BUSINESS SERVICES.	Service and powerType/superSubType	
FEDERAL-SERVICE-DOMAIN	(22754/1) (A) A HIGH-LEVEL VIEW OF THE SERVICES AND CAPABILITIES THAT SUPPORT ENTERPRISE AND ORGANIZATIONAL PROCESSES AND APPLICATIONS.	Service and powerType/superSubType	
FEDERAL-SERVICE-TYPE	(22755/1) (A) A GROUP OF SIMILAR CAPABILITIES THAT SUPPORT A SINGLE FEDERAL-SERVICE-DOMAIN.	Service and powerType/superSubType	
FUNCTIONAL-AREA	(4198/2) (A) A MAJOR AREA OF RELATED ACTIVITY.	Activity and powerType/superSubType	
FUNCTIONAL-PROCESS-FUNCTION	(22044/1) (A) A GENERAL CLASS OF ACTIVITY IN A SPECIFIC FUNCTIONAL-AREA.	Activity and powerType/superSubType	
GUIDANCE	(336/4) (A) A STATEMENT OF DIRECTION RECEIVED FROM A HIGHER ECHELON.	Guidance	
HAND-RECEIPT	(21353/1) (A) THE SPECIFICATION OF TRANSFER OF PROPERTY RESPONSIBILITY.	Information and powerType/superSubType	Not required in DoDAF 2
ICON-CATALOG	(19625/1) (A) A DIRECTORY OF IMAGES DEPICTED IN GRAPHICAL PRESENTATION SOFTWARE.	Information and powerType/superSubType	Not required in DoDAF 2
ICON-DATA-CATEGORY	(22294/1) (A) A CLASSIFICATION OF ELEMENTS OF INFORMATION THAT APPLY TO ICONS WITHIN AN ICON-CATALOG.	Information and powerType/superSubType	Not required in DoDAF 2

CADM Entity Name	CADM Entity Definition	DM2 Mappings	Mapping Notes
IDENTIFICATION-FRIEND-FOE	(17031/1) (A) THE RECOGNIZED HOSTILITY CHARACTERIZATION OF A BATTLEFIELD OBJECT.	Performers whose dispositional Activities DesiredEffects dimishes ownforce DesiredEffect goals below a threshold	Not required in DoDAF 2
IMPLEMENTATION-TIME-FRAME	(19731/1) (A) THE SPECIFICATION OF A GENERAL CHRONOLOGICAL PERIOD FOR THE INSTANTIATION OF A CONCEPT, SYSTEM, OR CAPABILITY.	Project, an Activity within (Instantiation) and timePeriod of that Activity related to an activityResourceOverlap where the Resource is a System or Performer that manifests a Capability	
INFLATION-FACTOR	(19732/1) (A) ADJUSTMENTS TO COSTS THAT DEPEND ON FISCAL YEAR.	MeasureType	
INFORMATION-ASSET	(4246/3) (A) AN INFORMATION RESOURCE.	Information and, if needed, System and wholePart	
INFORMATION-ELEMENT	(4199/2) (A) A FORMALIZED REPRESENTATION OF DATA SUBJECT TO A FUNCTIONAL PROCESS.	Information, Performer, and Rule (In CADM, an Information Element is really an IDEF0 ICOM.)	
INFORMATION-TECHNOLOGY-REGISTRATION	(20501/1) (A) THE IDENTIFICATION OF A MISSION-CRITICAL/MISSION-ESSENTIAL INFORMATION TECHNOLOGY SYSTEM OR OTHER ASSET.	A type of Information (Registration) that describes a System and that possibly has been consumed by a registrar (type of Performer) after have been produced by a registrant, possibly in response to a Rule.	Not required in DoDAF 2
INFORMATION-TECHNOLOGY-STANDARD-CATEGORY	(20513/1) (A) A CLASSIFICATION OF INFORMATION-TECHNOLOGY-STANDARD.	Type of Standard	
INTERNAL-DATA-MODEL-TYPE	(9289/2) (A) A CLASSIFICATION OF AN INTERNAL-DATA-MODEL.	Type of Data	COI extension
INTERNET-ADDRESS	(19762/1) (A) THE SPECIFICATION OF A VALUE OR RANGE OF VALUES CONSTITUTING THE LABEL FOR A NODE ON THE INTERNET.	Type of Address	COI extension
LANGUAGE	(2228/1) (A) A MEANS OF COMMUNICATION BASED ON A FORMALIZED SYSTEM OF SOUNDS AND/OR SYMBOLS.	Type of Rule or Standard	COI extension
LINE-OF-BUSINESS	(22593/1) (A) THE TOP-LEVEL SET OF FUNCTIONS PERFORMED BY THE FEDERAL GOVERNMENT.	Activity and powerType/superSubType	
LOCATION	(343/2) (A) A SPECIFIC PLACE.	Location	
MATERIEL	(337/1) (A) AN OBJECT OF INTEREST THAT IS NON-HUMAN, MOBILE, AND PHYSICAL.	Materiel	
MATERIEL-TYPE	(787/1) (A) A CHARACTERIZATION OF A MATERIEL ASSET.	Materiel and powerType/superSubType	
MATERIEL-TYPE-PRODUCTION	(733/2) (A) A MATERIEL-ITEM THAT IS IDENTIFIED BY PRODUCER OR INDUSTRY MANUFACTURER.	Materiel, activityResourceOverlap, and activityPerformer	COI extension
MILITARY-PLATFORM	(22100/1) (A) AN OBJECT FROM WHICH OR THROUGH WHICH MILITARY TASKS CAN BE CONDUCTED.	Performer	
MILITARY-TELECOMMUNICATION-USE	(19773/1) (A) THE CHARACTERIZATION OF SPECIFIC USE-DEPENDENT BUT FACILITY-INDEPENDENT PARAMETERS FOR ESTIMATING THE COMMUNICATIONS, WIRING, AND EQUIPMENT REQUIRED BY MILITARY OCCUPANTS OF FACILITIES.	Performer and wholePart of Organization, Materiel, and System	COI extension
MILITARY-UNIT-LEVEL	(42/2) (A) A MILITARY-UNIT ACCORDING TO A STRATUM, ECHELON, OR POINT WITHIN THE MILITARY COMMAND HIERARCHY AT WHICH CONTROL OR AUTHORITY IS CONCENTRATED.	Measure, MeasureType, and a subtype of resourceTypeInstanceOfMeasure	
MISSION	(1/3) (A) THE TASK, TOGETHER WITH THE PURPOSE, THAT CLEARLY INDICATES THE ACTION TO BE TAKEN.	Activities and DesiredEffect	
MISSION-AREA	(2305/1) (A) THE GENERAL CLASS TO WHICH AN OPERATIONAL MISSION BELONGS.	Activities, DesiredEffect, and powerType/superSubType	
MODELING-AND-SIMULATION-JUSTIFICATION	(19776/1) (A) A STATEMENT PROVIDING RATIONALE TO JUSTIFY REQUIREMENTS FROM THE POINT OF VIEW OF MODELING AND SIMULATION.	description of DesiredEffects and Performer dispositions	Not required in DoDAF 2

CADM Entity Name	CADM Entity Definition	DM2 Mappings	Mapping Notes
NETWORK	(10972/1) (A) THE SPECIFICATION FOR THE JOINING OF TWO OR MORE NODES FOR A SPECIFIC PURPOSE.	Systems and overlaps	
NETWORK-CONTROLLER-TYPE	(20591/2) (A) THE KIND OF FUNCTIONAL PROPONENT WHO EXERCISES AUTHORITY OVER A NETWORK.	Person Type or Organization Type	
NETWORK-ECHELON	(22486/1) (A) THE NORMAL OPERATIONAL LEVEL SUPPORTED BY A NETWORK.	System, Organization Type, and overlap	
NETWORK-TYPE	(11570/1) (A) A SPECIFIC KIND OF NETWORK.	System (made up of Systems and overlaps) and powerType/superSubType	
NODE	(956/1) (A) A ZERO DIMENSIONAL TOPOLOGICAL PRIMITIVE THAT DEFINES TOPOLOGICAL RELATIONSHIPS.	EffectObject	
NODE-ASSOCIATION	(19796/1) (A) AN ASSOCIATION OF ONE SPECIFIC NODE TO ANOTHER NODE.	could be wholePart, superSubType, overlap, or beforeAfter	
NODE-LINK-ASSOCIATION	(20498/1) (A) THE ASSOCIATION OF ONE NODE-LINK WITH ANOTHER NODE-LINK.	usually wholeParts or overlaps	
NODE-SYSTEM	(19840/1) (A) THE ASSOCIATION OF A SPECIFIC NODE WITH A SPECIFIC SYSTEM.	System and overlaps with other types of Nodes	
NODE-SYSTEM-ASSET-OWNERSHIP	(20009/1) (A) THE POSSESSION, IN WHOLE OR PART, OF THE OBJECTS OF VALUE ASSOCIATED TO A SPECIFIC NODE-SYSTEM.	Organization, Resources, Rule, and activityResourceOverlap	
NODE-SYSTEM-COST-MANAGEMENT	(20011/1) (A) THE AMOUNTS ASSOCIATED WITH VARIOUS ASPECTS OF THE MANAGEMENT OF A NODE-SYSTEM.	System, resourceTypeInstanceOfMeasure, and possibly Location	
OCCUPATION	(2009/1) (A) A FIELD OF WORK.	Person Type	
OPERATIONAL-CONDITION	(19589/1) (A) A VARIABLE OF THE OPERATIONAL ENVIRONMENT OR SITUATION IN WHICH A UNIT, SYSTEM, OR INDIVIDUAL IS EXPECTED TO OPERATE THAT MAY AFFECT PERFORMANCE.	Condition	
OPERATIONAL-DEPLOYMENT-MISSION-TYPE	(19848/1) (A) THE KIND OF HIGH-LEVEL TASKING FOR DEPLOYED OPERATIONS.	Activity and powerType/superSubType	
OPERATIONAL-DEPLOYMENT-PHASE	(19849/1) (A) A STAGE OF THE OPERATIONAL ACTIVITIES CONDUCTED FOR DEPLOYED OPERATIONS.	Activities, temporalWholePart, and beforeAfter	
OPERATIONAL-FACILITY-ECHELON	(19853/1) (A) A SUBDIVISION OF A HEADQUARTERS (OR) A SEPARATE LEVEL OF COMMAND AS IT APPLIES TO AN OPERATIONAL-FACILITY.	Measure associated with Organization	
OPERATIONAL-FACILITY-PROPONENT	(19854/2) (A) THE AGENT RESPONSIBLE FOR REQUIREMENTS DEVELOPMENT OF OPERATIONAL FACILITIES.	Organization, Facility, Rule, and activityResourceOverlap	
OPERATIONAL-MISSION-THREAD	(19857/1) (A) AN IDENTIFIED INFORMATION EXCHANGE SEQUENTIAL PROCEDURE TO SUPPORT TASK EXECUTION BY INFORMATION SYSTEMS AND ORGANIZATION-TYPES.	Activities, temporalWholePart, overlaps, and beforeAfter and their System and Organization Type Performers	
OPERATIONAL-ROLE	(22459/1) (A) THE SPECIFICATION OF A SET OF ABILITIES REQUIRED FOR PERFORMING ASSIGNED ACTIVITIES AND ACHIEVING AN OBJECTIVE.	Activities, DesiredEffect, and activityTypeInstanceOfMeasure	
OPERATIONAL-SCENARIO	(19860/1) (A) A CONCEPT AND SCRIPT FOR POSSIBLE EVENTS AND ACTIONS FOR MILITARY OPERATIONS.	Activities, Performers, beforeAfter, temporalWholePart, overlap (in an possible or future time)	
ORGANIZATION	(345/1) (A) AN ADMINISTRATIVE STRUCTURE WITH A MISSION.	Organization	
ORGANIZATION-ASSOCIATION	(1077/1) (A) AN ASSOCIATION OF AN ORGANIZATION WITH ANOTHER ORGANIZATION.	could be wholePart, superSubType, overlap, or beforeAfter	
ORGANIZATION-TYPE	(892/2) (A) A CLASS OF ORGANIZATIONS.	Organization Type	
ORGANIZATION-TYPE-ASSOCIATION	(9211/1) (A) THE ASSOCIATION OF AN ORGANIZATION-TYPE WITH ANOTHER ORGANIZATION-TYPE.	could be wholePart, superSubType, overlap, or beforeAfter	

CADM Entity Name	CADM Entity Definition	DM2 Mappings	Mapping Notes
PERIOD	(1321/1) (A) INTERVAL OF TIME.	temporalMeasure	
PERSON-TYPE	(897/2) (A) A CLASS OF PERSONS.	PersonType	
POINT-OF-CONTACT	(19867/1) (A) A REFERENCE TO A POSITION, PLACE, OFFICE, OR INDIVIDUAL ROLE IDENTIFIED AS A PRIMARY SOURCE FOR OBTAINING INFORMATION.	Person	
POINT-OF-CONTACT-TYPE	(22039/1) (A) A KIND OF POINT-OF-CONTACT.	PersonType	
POSITION	(2112/1) (A) A SET OF ESTABLISHED DUTIES.	PersonType, Activities, and activityPerformerOverlap	
PROCESS-ACTIVITY	(4204/3) (A) THE REPRESENTATION OF A MEANS BY WHICH A PROCESS ACTS ON SOME INPUT TO PRODUCE A SPECIFIC OUTPUT.	Activity	
PROCESS-ACTIVITY-FUNCTIONAL-PROCESS	(22043/1) (A) THE MEANS BY WHICH TO CARRY OUT A HIGH-LEVEL FUNCTION.	Activity	
PROCESS-STATE-VERTEX	(20025/1) (A) THE ABSTRACTION OF AN OBSERVABLE MODE OF BEHAVIOR.	Activity	
RECORD-TRACKING	(19871/1) (A) INFORMATION REGARDING A SPECIFIC RECORD IN A TABLE OF DATA.	N/A -- modeling artifact	Not required in DoDAF 2
REGIONAL-COST-FACTOR	(19544/1) (A) THE EXPECTED EXPENSE MODIFICATION FOR A GEOGRAPHIC AREA THAT ACCOUNTS FOR SPECIFIC LOCAL COSTS IN RELATION TO A NATIONAL AVERAGE.	MeasureType	
RELATION-TYPE	(6515/2) (A) AN ASSOCIATION BETWEEN OBJECTS THAT DEFINES AN INFORMATION ASSET.	dataAssociation	
ROOM-TYPE	(5605/1) (A) A KIND OF A ROOM.	Facility and powerType/superSubType	COI extension
RULE-MODEL-OPERATIONAL-RULE	(20032/1) (A) AN ASSOCIATION OF A SPECIFIC RULE-MODEL WITH A SPECIFIC OPERATIONAL-RULE.	ArchitectureDescription, describedBy, and Rules	
SATELLITE	(14361/1) (A) A MAN-MADE BODY WHICH REVOLVES AROUND AN ASTROMETRIC-ELEMENT AND WHICH HAS A MOTION PRIMARILY DETERMINED BY THE FORCE OF ATTRACTION OF THAT ASTROMETRIC-ELEMENT.	Type of Materiel	COI extension
SECURITY-ACCESS-COMPARTMENT	(16224/2) (A) THE SPECIFICATION OF AN EXCLUSION DOMAIN FOR INFORMATION RELEASED ON A FORMALLY RESTRICTED BASIS (E.G., TO PROTECT SOURCES OR POTENTIAL USE).	IC-ISM	
SECURITY-CLASSIFICATION	(940/2) (A) THE LEVEL ASSIGNED TO NATIONAL SECURITY INFORMATION AND MATERIAL THAT DENOTES THE DEGREE OF DAMAGE THAT ITS UNAUTHORIZED DISCLOSURE WOULD CAUSE TO NATIONAL DEFENSE OR FOREIGN RELATIONS OF THE UNITED STATES AND THE DEGREE OF PROTECTION REQUIRED.	IC-ISM	
SKILL	(2226/1) (A) AN ABILITY.	Skill	
SOFTWARE-LICENSE	(1856/1) (A) THE STIPULATION(S) (AND LEGAL TERMS) BY WHICH THE SOFTWARE MAY BE USED.	Type of Agreement	
SYSTEM	(326/1) (A) AN ORGANIZED ASSEMBLY OF INTERACTIVE COMPONENTS AND PROCEDURES FORMING A UNIT.	System	
SYSTEM-ASSOCIATION	(12546/1) (A) AN ASSOCIATION BETWEEN A SYSTEM AND ANOTHER SYSTEM.	could be wholePart, superSubType, overlap, or beforeAfter	
SYSTEM-STATUS	(19891/1) (A) THE SPECIFICATION OF THE CONDITION OF A SYSTEM AT A SPECIFIC POINT IN TIME.	generally typeInstances	
SYSTEM-STATUS-DEPENDENCY	(19892/1) (A) THE MANNER IN WHICH ONE SYSTEM-STATUS DEPENDS ON ANOTHER SYSTEM-STATUS.	The overlaps, beforeAfters, and temporalWholeParts of the objects for which systemTypeInstanceOf applies	

CADM Entity Name	CADM Entity Definition	DM2 Mappings	Mapping Notes
SYSTEM-STATUS-TYPE	(22098/1) (A) THE SPECIFICATION OF A KIND OF DEVELOPMENT OR TRANSITION OF ONE OR MORE SYSTEMS.	The powerType/superSubType of the objects for which systemTypeInstanceOf applies	
SYSTEM-TYPE	(9083/2) (A) A SPECIFIC KIND OF SYSTEM.	System and powerType/superSubType	
TASK	(290/2) (A) A DIRECTED ACTIVITY.	Activity	
TECHNICAL-INTERFACE	(21694/1) (A) A GENERIC CONNECTION BETWEEN TWO ELEMENTS THAT IMPLEMENT INFORMATION TECHNOLOGY IN WHICH INFORMATION IS CAPABLE OF BEING TRANSMITTED FROM THE SOURCE ELEMENT TO THE DESTINATION ELEMENT.	activityResourceOverlap and the Performers the perform the consuming and producing of the information	
TECHNICAL-INTERFACE-TYPE	(19761/1) (A) A KIND OF GENERIC CONNECTION BETWEEN ELEMENTS THAT IMPLEMENT INFORMATION TECHNOLOGY.	a powerType/superSubType on the TECHNICAL-INTERFACE	
TECHNICAL-SERVICE	(19676/1) (A) A DISTINCT PART OF THE SPECIALIZED FUNCTIONALITY THAT IS PROVIDED A SYSTEM ELEMENT ON ONE SIDE OF AN INTERFACE TO A SYSTEM ELEMENT ON THE OTHER SIDE OF AN INTERFACE.	activityResourceOverlap and the Performers the perform the consuming and providing service	
TECHNICAL-SERVICE-AREA	(19677/2) (A) A FIELD OF SPECIALIZED FUNCTIONALITY, USUALLY SPECIFIED BY A REFERENCE-MODEL TO DEFINE INTERFACES.	a powerType/superSubType on the TECHNICAL-SERVICE-AREA	
TECHNICAL-STANDARD-FORECAST-ELEMENT	(20043/2) (A) A SECTION OF A SPECIFIC TECHNOLOGY-STANDARD-FORECAST, WHICH CITES A TECHNICAL-SERVICE, TIME FRAME, OR INFORMATION-TECHNOLOGY-STANDARD.	Standard with future date and pedigree of the forecaster	
TECHNOLOGY	(8936/1) (A) THE APPLICATION OF SCIENCE TO MEET ONE OR MORE OBJECTIVES.	Technology (TBD)	
TECHNOLOGY-FORECAST	(20078/1) (A) A DETAILED DESCRIPTION OF EMERGING TECHNOLOGIES.	Technology with future date and pedigree of the forecaster	
TELEPHONE-ADDRESS	(1938/1) (A) AN ELECTRONIC ADDRESS THAT SUPPORTS COMMUNICATION VIA TELEPHONIC MEDIA.	Type of Address	COI extension
TRANSITION-PROCESS	(20082/1) (A) THE DESCRIPTION OF A METHOD FOR RELATING A "SOURCE" PROCESS-STATE-VERTEX TO A "TARGET" PROCESS-STATE-VERTEX.	Activities, wholeParts, and beforeAfters, with some possibly in the future	
UML-MODEL-ELEMENT	(22684/1) (D) A BASIC ARTIFACT OF THE UNIFIED MODELING LANGUAGE. Comment: USED TO CONSTUCT DIAGRAMS FOR EACH TYPE OF UML-MODEL..	N/A -- modeling artifact	
UNIFORMED-SERVICE-ORGANIZATION-COMPONENT-TYPE	(2726/2) (A) A SPECIFIC KIND OF SUBDIVISION OF A UNIFORMED-SERVICE-ORGANIZATION.	Type of OrganizationType	
UNIT-OF-MEASURE	(2482/2) (A) THE INCREMENT BY WHICH MATTER IS MEASURED.	MeasureType	

This page left intentionally blank

APPENDIX C
HOW DOES DODAF REPRESENT SECURITY?

Capabilities are subject to a variety of threats to the integrity, availability, and confidentiality of their operation. These threats range from failures of equipment, attempts to gain unauthorized access to their services and data, to sabotage of their functions. Security engineering is concerned with identifying the potential threats to a capability, and then, using a risk management approach, devising a set of measures which reduce the known and potential vulnerabilities to an acceptable level. In general, the measures that can be applied fall into the following categories:

- Physical – measures such as guards, guard dogs, fences, locks, sensors, including Closed Circuit Television, strong rooms, armor, weapons systems, etc.
- Procedural – the specification of procedures, including vetting (which tests that personnel have a sufficient level of integrity and trust to be given responsibility to access and use a capability's services and data) that will reduce the likelihood of vulnerabilities being exploited.
- Communication Security (COMSEC) – using encryption and other techniques to ensure that data transmission is available at sufficient bandwidth, that the traffic pattern and content of data in transit are indecipherable to a third party who might intercept the data, and that its integrity is protected.
- Transient Electromagnetic Pulse Emanation Standard (TEMPEST) – measures to ensure that the electromagnetic transmissions from equipment can't be intercepted to derive information about the equipment's operation and the data it processes.
- Information Security (INFOSEC) – ensuring the integrity, availability and confidentiality of data and IT-based services.

In general, the measures employed to protect a capability will have undesirable impacts on all of the capability's lines of development, and in particular on it's deploy ability, usability and procurement and maintenance costs. It is therefore desirable to minimize the strength of the measures to be employed in a fashion commensurate with the value of the assets being protected. This requires a risk-managed approach based on the assessment of the likely threats posed to the asset. A risk assessment approach considers the following characteristics:

- Environment – The level of hostility of the environment the asset is being deployed to.
- Asset Value – this is denoted by a protective marking which indicates the impact of the loss or disclosure of the asset would have on the effective operation of the government and its departments of state.
- Criticality – an assessment of the criticality of the asset to enabling the government to undertake its activities.
- Personnel Clearance – a measure of the degree of trust that the government is willing to put in the personnel that will have (direct or indirect) access to the asset.

The aim of this guidance for representing security considerations is to enable sufficient information to be recorded for interested parties (accreditors, security advisors, users, system managers) to understand the potential security exposure of capabilities so that security can be managed effectively throughout the life of a capability.

The Table C-1 below shows the DoDAF scheme for assigning security characteristics and protective measures to elements of DoDAF. There is not a specific security viewpoint in DoDAF; security information can be shown on models using annotations and call–outs. The DoDAF Meta-Model contains the concepts, associations, and attributes for capturing and representing security characteristics in a consistent way between models. Table B-1, DM2 Concepts, Associations, and Attributes Mapping to DoDAF-described Models indicates the security elements within the DM2.

Table C-1: DoDAF Viewpoints and Concept Mapped to Security Characteristics and Protective Measures

Viewpoint	Concept	Security Characteristics	Protective Measures	Notes
Capability	Capability requirement	Security Marking Criticality Environment User Security Profile		The security characteristics of capability requirements provide the security envelope for the capability for a particular timeframe.
Operational	Location	User Security Profile Environment		The User Security Profile is the lowest clearance of the users within a location, facility, or organization. The environment identifies the most hostile conditions for the location, facility, or organization.
	Operational Activity	Security Marking Criticality		The security marking identifies the highest security marking of information that will be processed by a Operational Activity and the Criticality measures the impact on government operations with the disruption of the operational activity.
	Resource Flow	Security Marking		The security marking identifies the highest security marking that will be exchanged in a Resource Flow.
	Organization	User Security Profile Environment		The minimum clearances of

Table C-1: DoDAF Viewpoints and Concept Mapped to Security Characteristics and Protective Measures

Viewpoint	Concept	Security Characteristics	Protective Measures	Notes
				members of the organization, post, base, fort.
System	Capability Taxonomy	Security Marking Criticality Environment User Security Profile		The security characteristics of a capability taxonomy are to be derived from the constituent systems.
	System	Security Marking Criticality Environment User Security Profile	Physical TEMPEST COMSEC	The environment of a system is derived from the Physical Asset to which is deployed. The User Security Profile is derived from the Organization which uses the system, its Criticality and Security Marking from its Functions.
	Physical Asset	Environment	Physical TEMPEST	The environment identifies the worst environment to which the Physical Asset will be deployed.
	Function	Security Marking Criticality	INFOSEC Procedural	The Security Marking identifies the maximum security marking of the data the Function will process and the criticality represents the degree of harm to government operations if disrupted.
	System Resource Flow	Security Marking	COMSEC	The Security Marking represents the maximum security marking of the Resource Flow.
	Performer and Function	User Security Profile	Procedural	The User Security Profile is the lowest clearance of the user performing the function. This should be derived from Organizations who perform the Function, if the information

Table C-1: DoDAF Viewpoints and Concept Mapped to Security Characteristics and Protective Measures

Viewpoint	Concept	Security Characteristics	Protective Measures	Notes
				exists.
Service	Capability Taxonomy	Security Marking Criticality Environment User Security Profile		The security characteristics of a capability taxonomy are to be derived from the constituent services.
	Service	Security Marking Criticality Environment User Security Profile	Physical TEMPEST COMSEC	The environment of a service is derived from the Physical Asset to which is deployed. The User Security Profile is derived from the Organization which uses the service, its Criticality and Security Marking from its Functions.
	Physical Asset	Environment	Physical TEMPEST	The environment identifies the worst environment to which the Physical Asset will be deployed.
	Function	Security Marking Criticality	INFOSEC Procedural	The Security Marking identifies the maximum security marking of the data the Function will process and the criticality represents the degree of harm to government operations if disrupted.
	System Resource Flow	Security Marking	COMSEC	The Security Marking represents the maximum security marking of the Resource Flow.
	Performer and Function	User Security Profile	Procedural	The User Security Profile is the lowest clearance of the user performing the function. This should be derived from

Table C-1: DoDAF Viewpoints and Concept Mapped to Security Characteristics and Protective Measures

Viewpoint	Concept	Security Characteristics	Protective Measures	Notes
				Organizations who perform the Function, if the information exists.
Standards	Performer	Security Marking	INFOSEC Procedural	The Security Marking identifies the security standard for the data the Function will process and the criticality represents the degree of harm to government operations if there is unauthorized access.

This page left intentionally blank

APPENDIX D
REFERENCES

Business Rule Concepts Ross, R. (2005), 2nd Ed., Business Rules Forum. 135pp

Chairman of the Joint Chiefs of Staff (CJCS) Manual 3500.04E, Universal Joint Task Manual, 25 August 2008. The current version can be found at: http://www.dtic.mil/doctrine/training/cjcsm3500_04e.pdf

Department of Defense Directive (DoDD) 5000.1, The Defense Acquisition System, 12 May 2003 (certified current as of November 20, 2007). A current copy of the directive can be found at: http://www.dtic.mil/whs/directives/corres/pdf/500001p.pdf

Department of Defense Instruction (DoDI) 5000.2, Operation of the Defense Acquisition System, December 8, 2008. A current copy of the directive can be found at: http://www.dtic.mil/whs/directives/corres/pdf/500002p.pdf

Dictionary of Military and Associated Terms, Joint Publication 1-02, As Amended Through 17 October, 2008. A current copy of this document can be found at: http://www.dtic.mil/doctrine/jel/new_pubs/jp1_02.pdf

Ministry of Defense Architecture Framework (MODAF). Ministry of Defense of the United Kingdom. The latest electronic version of the MODAF can be found at the MODAF Website: http://www.modaf.org.uk/. An offline (PDF) version can be accessed here for downloading: http://www.modaf.org.uk/file_download/39/20081001-MODAF_Handbook_V1_2_003_Adobe6-U.pdf

NATO Architecture Framework (NAF) for C3 Systems (2006). The North Atlantic Treaty Organization. The latest version of the NAF, and other associated documents can be found at the NATO Website http://194.7.80.153/website/book.asp?menuid=15&vs=0&page=volume2%2Fch02s02.html (Requires User Registration)

A Guide to the Project Management Body of Knowledge (PMBOK® Guide) - Third Edition, Project Management Institute. 2004 The PMBOK Guide can be purchased at http://www.pmi.org/Marketplace/Pages/ProductDetail.aspx?GMProduct=00100035801 (Requires a Purchase)

Work Breakdown Structure Handbook (MIL-HDBK-881A), 30 July 2005. The curren version can be found at: http://www.acq.osd.mil/pm/currentpolicy/wbs/MIL_HDBK-881A/MILHDBK881A/WebHelp3/MILHDBK881A.htm

Zachman, John. Zachman Framework. © Zachman International. The Zachman Framework can be found at the Zachman International Website: http://zachmaninternational.com/index.php/the-zachman-framework/26-articles/13-the-zachman-framework-a-concise-definition