Office of the
DEPUTY CHIEF
MANAGEMENT OFFICER

BEA Architecture Product Guide

March 15, 2012

Version History

Version	Publication Date	Author	Summary of Changes
1.0	August 15, 2006	BEA Development Team	Initial baseline BEA 4.0 release
1.1	March 2, 2007	BEA Development Team	Updated to reflect BEA 4.1 release
1.2	September 28, 2007	BEA Development Team	Updated to reflect BEA 5.0 release
1.3	March 14, 2008	BEA Development Team	Updated to reflect BEA 5.0 final release
1.4	August 22, 2008	BEA Development Team	Updated to reflect BEA 6.0 release
1.5	September 30, 2008	BEA Development Team	Updated to reflect BEA 6.0 methodology changes
1.6	March 13, 2009	BEA Development Team	Updated to reflect BEA 6.0 final release
1.7	June 5, 2009	BEA Development Team	Updated to prohibit child ICOM from being produced by more than one leaf level OV-5 Operational Activity. Updated the BIP Template. Modified wording Section 5.3.3.2.
1.8	June 30, 2009	BEA Development Team	Updated Section 5.2.2.1.3 to allow mapping of an Operational Activity to more than one Business Capability.
1.9	July 21, 2009	BEA Development Team	Added End to End Business Flow Model Section and checklists.
1.10	December 15, 2009	BEA Development Team	Included in the OV-6a Operational Rules Model section the "APG Compliant" input checkbox.
1.11	December 22, 2009	BEA Development Team	Added ability to map one E2E L0 to another E2E L0 or to E2E L1 within another E2E L0.
1.12	February 16, 2010	BEA Development Team	Updated: Enterprise Standards and replaced Initiative name, Data Structures,

			SvcV-1, SvcV-5 Sections, Checklists, OV-6c 2, CV-6, deleted or renamed Data Synonyms where appropriate.
1.13	January 25, 2011	BEA Development Team	Updated for BEA 8.0.
1.14	Jul 18-Sep 1, 2011	BEA Development Team	OV-6a, SV-10a Sections and Checklists, OV-6c
1.15	March 15, 2012	BEA Development Team	Updated for BEA 9.0

Table of Contents

Index of Figures

Index of Tables

1 Introduction

The BEA Architecture Product Guide (APG) was initially developed for the internal use of the architects, analysts and modelers on the BEA Development Team as a guide to describe the methods, rules, and modeling conventions to be used for the development of the models that comprise the Business Enterprise Architecture (BEA). The BEA is the enterprise level architecture for the Department of Defense (DoD) Business Mission Area (BMA) in support of the warfighter.

Although the APG is directed mainly to architectural models included in the DoD Architectural Framework (DoDAF), it also provides guidance and model descriptions for non-DoDAF models that are part of the BEA to help in understanding these models and their relationships to other DoDAF models.

1.1 Purpose and Scope

The purpose of the APG is to provide modelers and analysts with a user guide that describes how to define, create, update, and interpret models within the BEA. For each BEA model and associated elements, the APG contains a section that provides a summary description of the purpose, structure, relationship to other BEA models, and model definitions. Each section also includes procedures and tasks required to develop each model that can be applied independent of the repository and development tool used. Finally, each section identifies the applicable modeling standards and conventions required for the IBM Rational System Architect (SA) tool, which has been used for BEA development since the first BEA deliverable.

The APG is intended for an audience that understands DoDAF and has SA training and/or experience. This document does not address guidelines for the Systems Evolution Description (SV-8), which pertains to the Enterprise Transition Plan (ETP) and is external to SA.

Section 2 of this document reviews key concepts from DoDAF that are incorporated into the BEA. Sections 3 through 14 **set forth** the specific modeling methods and conventions for developing or maintaining each model built for the BEA and describe how individual models are integrated and linked across views. The ordering for these sections follows the development order for the BEA as documented in the BEA Development Methodology (BDM). The Appendices provide templates, checklists and reference tools to complement the guidelines set forth in this document.

The primary information sources used to develop, revise, and update this document are: DoDAF v2.0, Integrated Definition for Function Modeling (IDEF0), Institute of Electrical and Electronics Engineers (IEEE), Integrated Definition for Data Modeling (IDEF1X), Object Management Group's (OMG) Semantics of Business Vocabulary and Business Rules (SBVR) v1.0, and Business Process Modeling Notation v2.0 (BPMN).

1.2 Relationship to Other Documents

This document is intended to provide BEA practitioners the guidance and knowledge needed to maintain and extend the BEA. It provides supporting details on how to model relevant architecture models in accordance to the BEA Development Methodology (BDM) and compiles best practices that have been tried and tested across the BEA lifecycle. The BEA Configuration Management Document (CMD) provides guidance on specific documentation, approvals and sequencing of BEA development tasks. Changes to the BEA models are subject to Configuration Management (CM) procedures set forth by the CMD, including stakeholder reviews.

2 Key Concepts

The BEA is the enterprise architecture for the DoD BMA. The BEA defines the Department's business transformation priorities, the Business Capabilities required to support those priorities and the combinations of systems and initiatives that enable these capabilities. This section describes key concepts required to understand the BEA guidelines for developing the set of integrated DoDAF models relevant to the BEA.

2.1 DoD Architecture Framework Architecture Conventions

The BEA is an integrated architecture, as defined by DoDAF. An integrated architecture has designated, common points of reference linking the models. An architecture is deemed integrated when development of its models and constituent architecture objects enables the architecture objects defined in one model to be identical (have the same name, definition and properties) when the same object is referenced in another model.

DoD adopted DoDAF V2.0 on May 28, 2009, which specified architectures shall comply with DoDAF Version 2.0 in their next major release. As a result of this mandate; BEA 7.0 migrated from DoDAF V1.5 based architecture to DoDAF V2.0.

2.1.1 DoDAF V2.0 Details

DoDAF V2.0 serves as the overarching framework enabling the development of architectures within DoD. DoDAF focuses on architectural data and information required by key DoD decision-makers. Managers at all levels 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.

Shifting from Model-Centric to Data-Centric Focus, prior versions of DoDAF and earlier C4ISR versions of the Architecture Framework have emphasized reusable and interoperable data organized into 'models' (e.g., graphical representations or documents). DoDAF V2.0 shifts from model-centric to data-centric focus, placing its emphasis on utilizing architectural data to support analysis and decision making, and greatly expands the types of graphical representations that can be used to support decision-making activities. With appropriate architectural data, it is possible to support innovative and flexible presentation of the architectural data in a meaningful, useful, and understandable manner through the "Fit-for-Purpose Views" described in DoDAF Volumes 1 and 2.

DoDAF V2.0 focuses on architectural data, rather than on developing individual models as described in previous versions. In general, data can be collected, organized, and stored by a wide range of architecture tools developed by commercial sources.

2.1.2 Differences between DoDAF V1.5 and DoDAF V2.0

Under DoDAF V1.5 there were four architecture views: the All View (AV), the Operational View (OV), Systems View (SV) and Technical Standards View (StdV). They were replaced with 8 Viewpoints under DoDAF V2.0 Where DoDAF V1.5 was model based with model specific data according to the Core Architecture Data Model (CADM), DoDAF V2.0 is data centric relying on DoDAF Metamodel (DM2) based on 12 meta-data groups.

The DoDAF V2.0 viewpoints included in the BEA are as follows: All Viewpoint (AV), Capability Viewpoint (CV), Data and Information Viewpoint (DIV), Operational Viewpoint (OV), Services Viewpoint (SvcV), Standard Viewpoint (StdV), Systems Viewpoint (SV). Each DoDAF V2.0 viewpoint is composed of a set of architecture models.

All DoDAF V2.0 models are based on consistent architecture data. Architecture data is categorized into twelve DM2 Data Groups. DM2 specifies the conceptual relationship between architecture objects leaving implementation details of how to capture store and relate these concepts up to each architecture tool vendor. All DoDAF V2.0 models are interchangeable via the Physical Exchange Specification (PES), which specifies the precise file exchange format for each DoDAF model. It is anticipated that these tool vendors will adopt the DM2 PES for the exchange of architectural data.

2.1.3 DoDAF V2.0 implementation in BEA

While the BEA architecture data is defined according to the DM2 concepts, associations, attributes, and is capable of transfer in accordance with the PES; the production of the PES will be deferred until this capability is provided as a native feature of the architecture tools supporting BEA. As architecture tool vendors upgrade their models to adhere to DoDAF V2.0, the tool's enforcement of DM2 and their ability to produce and consume PES will permit architecture content to be exchanged between separate DoDAF architectures independent of specific vendors' models. As of the release of BEA 7.0, the vendors providing tools supporting BEA have not yet released DoDAF V2.0 upgrades. Therefore, additional changes are anticipated in future releases when the BEA content is migrated into a DoDAF V2.0 compliant tool.

Figure 2-1, DoDAF V2.0 Viewpoints contains the viewpoints, as defined by DoDAF.

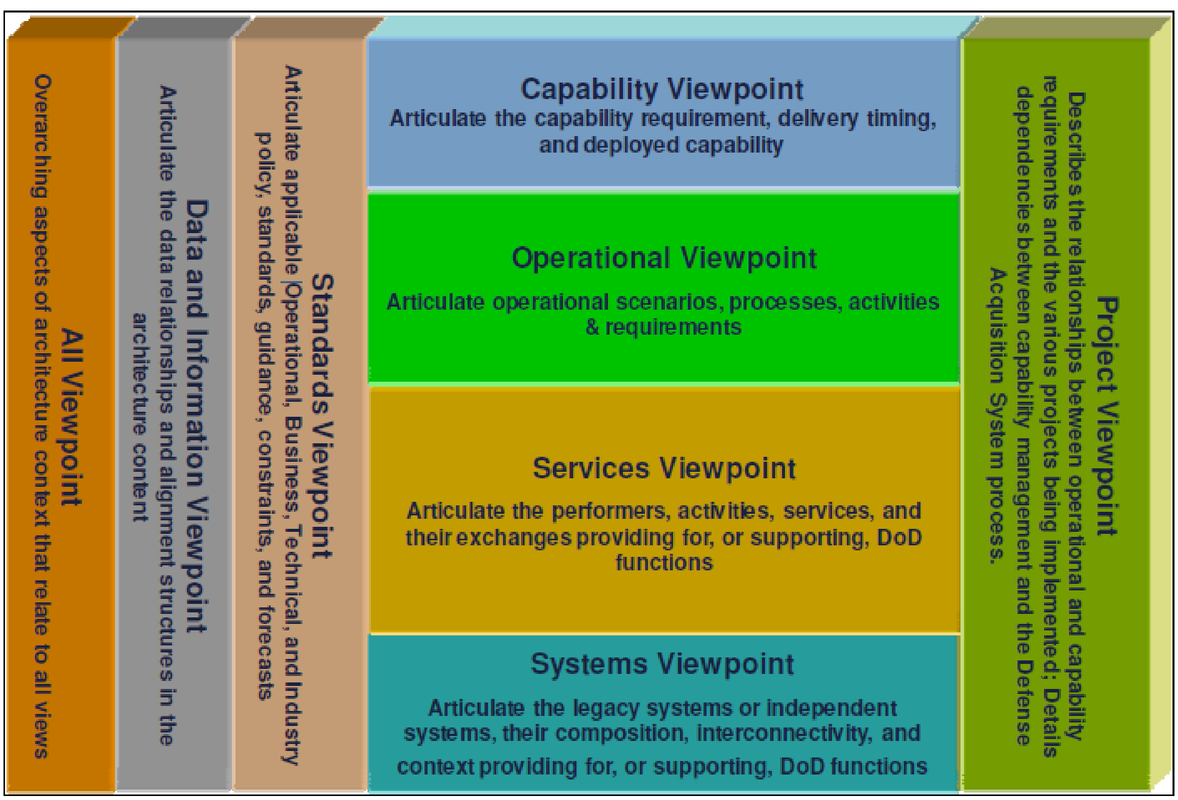


Figure 2-1, DoDAF V2.0 Viewpoints

To achieve consistency and ensure integration, DoDAF requires that the architecture provide consistent data among the models supporting each viewpoint. The subsections that follow provide a more detailed description of each of the viewpoints and its application to the BEA.

2.1.4 All Viewpoint

The All Viewpoint (AV) models provide information pertinent to the entire Architectural Description, such as the scope and context of the Architectural Description. The BEA AV-1 contains the Overview and Summary Document and the BEA AV-2 consists of an integrated dictionary.

2.1.5 Capabilities Viewpoint

The Capability Viewpoint (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. The BEA CV-1 captures the JCA taxonomy and lexicon along with their alignment to existing BEA operational activities per DoDAF V2.0 DM2.

2.1.6 Data and Information Viewpoint

The Data and Information Viewpoint (DIV) captures the business information requirements and structural business process rules for the Architectural Description.

The BEA DIV Model comprises a single integrated data model. It is used to capture the conceptual, logical and physical data and information requirements of the Business mission area of DoD. While most of the BEA DIV is presently composed of DIV-2 Logical Data Models future developments will reserve DIV-2 and DIV-3 level of detail for Enterprise Standards as required to capture the structural assertions to achieve the BMA data interoperability objectives.

2.1.7 Operational Viewpoint

The Operational Viewpoint (OV) captures the organizations, tasks, or activities performed, and information that must be exchanged between them to accomplish DoD missions.

The BEA OV is a representation of the operational activities and related operational information required to accomplish DoD Core Business Missions (CBMs). The OV contains graphical and textual models that describe the operational concept associated with accomplishing the core business missions, identifies the Operational Nodes, and assigned Activities to those nodes, and identifies the Information Exchanges (IEs) between Operational Nodes. The OV also defines the nature of the IEs, including the types of information being exchanged.

2.1.8 Services Viewpoint

The Services Viewpoint (SvcV) captures system, service, and interconnection functionality providing for, or supporting, operational activities.

The BEA provides a set of graphical and textual models that describe services, including their functions and interfaces. These models associate systems and service capabilities to the OV. Typically, a Service performs a single snippet of business functionality.

2.1.9 Standards Viewpoint

The Standards Viewpoint (StdV) is the minimal set of rules governing the arrangement, interaction, and interdependence of system parts or elements.

The BEA StdV is a set of tabular models that lists the set of standards that govern the arrangement, interaction and interdependence of system parts. The StdV ensures that a system adhering to a set of standards can provide the required capabilities to satisfy a specified set of operational requirements. The StdV provides the technical systems implementation guidelines for basic engineering specifications and common building blocks. The StdV may include a collection of the technical standards implementation conventions, standards options and rules. Furthermore, the StdV organizes criteria into profiles that govern systems and system elements for use across the BMA.

2.1.10 Systems Viewpoint

Systems Viewpoint (SV) captures the information on supporting automated systems, interconnectivity, and other systems functionality in support of operating activities.

The BEA provides a set of graphical and textual models that describe systems/services, including their functions and interfaces. These models associate systems and service capabilities to the OV. These system capabilities support the Operational Activities and facilitate the exchange of information among Operational Nodes. . Typically, a System performs a wide range of business functionality.

3 AV-1 Overview and Summary Information

3.1 Summary Description

This section describes the AV-1 architecture model, its relationship to other BEA models, the development method, and the guidelines used for its development.

3.1.1 Model Purpose

The AV-1 Overview and Summary Information document provides executive-level information in a consistent form to identify the Purpose and Viewpoint, Context and Scope of each BEA release. AV-1 serves two additional purposes. In the initial phases of architecture development, it serves as a planning guide. Upon completion of the BEA, AV-1 provides summary Findings and Recommendations concerning the architecture.

3.1.2 Model Structure

The AV-1 model is a text document without diagrams or matrixes and follows the standard DoDAF v2.0 structure for AV-1 models.

3.1.3 Relationship to Other BEA Models

The AV-1 is the overall summary document for the BEA and sets the context for all the other BEA models. In particular:

- Any AV-1 related term with specialized meaning must be listed within the AV-2 and any acronyms introduced on the AV-1 must be listed within the AV-2 Acronym list.
- The AV-1 documents are the authoritative source for terms that are also used in the ETP.

3.1.4 AV-1 Model Definitions

The AV-1 does not have model specific terms. The AV-1 templates contain the standard descriptions used in the AV-1.

3.2 Developing the AV-1 Model

This section describes the approach to develop the AV-1 model.

The DoD Business Mission Area has focused the scope of its architectural development efforts around selected functional areas or Core Business Missions (CBMs). CBMs have been identified as the highest priority transformation initiatives at the DoD Enterprise level and serve as the current focus of the BEA development effort. Two levels of the AV-1 are developed for each release of the BEA, the stakeholder level AV-1 and the BEA level AV-1.

3.2.1 Pre-Development Tasks

An AV-1 document is developed for each stakeholder to serve as a planning guide during the initial phases of architecture development. The initial AV-1 Overview and Summary document describes the scope of the work planned. The AV-1 is the basis for a BEA Improvement Proposal (BIP) that identifies specific gaps or business capability improvements to be addressed in a future release of the BEA. The specifics of each body of work planned are detailed in the BIP found in Appendix E.

The initial BEA level AV-1 Overview and Summary document is also developed during the initial planning phases of each deliverable according to the standard DoDAF V2.0 structure.

3.2.2 Development Tasks

During development of the specific BEA models, the BEA AV-1, stakeholder AV-1s and the approved BIPs are used as reference documents to guide development efforts and ensure that the actual development work stays within the scope planned.

3.2.3 Post-Development Tasks

The BEA AV-1 and stakeholder AV-1s are also used as a reporting mechanism for identifying Findings and Recommendations (F&R) during the final phases of the deliverable cycle after all development and integration work has been completed. The F&Rs section of the stakeholder AV-1 comprises two parts:

- Part 1 is a review of the work identified in the BIPs for completeness and specific mappings of architecture achievements to stakeholder objectives.
- Part 2 includes other F&Rs for additional future development efforts. New F&Rs are developed, previously documented F&Rs are updated, and a disposition of previous F&Rs is reported.

The initial stakeholder AV-1 Overview and Summary and the F&R (Parts 1 and 2) encompass the final stakeholder AV-1 document.

Also at the end of the development cycle the BEA AV-1 is finalized by updating the initial Overview and Summary as needed and by reviewing comments, findings and recommendations from all BEA stakeholders, and combining them into the BEA level F&Rs that are added to the final BEA AV-1 document.

Guidance on developing the stakeholder AV-1 is found within the *CBM AV-1 Template* provided in Appendix D and the *BIP Template* provided in Appendix E. The BIP is the basis for creating and approving BEA content changes and scheduling bodies of work into BEA Releases via Change Requests at the beginning of the development cycle. The *BEA Development Methodology* (BDM) describes the Change Request process and the development cycle.

4 AV-2 – Integrated Dictionary

4.1 Summary Description

This section describes the AV-2 Integrated Dictionary, its relationship to other BEA models, the development method and the model guidelines to be followed.

4.1.1 Model Purpose

The AV-2 is the Integrated Dictionary for the BEA and contains all descriptions and terms that are used in the other BEA models. The AV-2 provides a central repository for a given architecture's data and metadata which enables the set of architecture models to stand alone, allowing them to be understood with minimal reference to outside resources. The AV-2 is an accompanying reference to other models. The key to long-term interoperability resides in the accuracy and clarity of these definitions, and its value lies in unambiguous definitions.

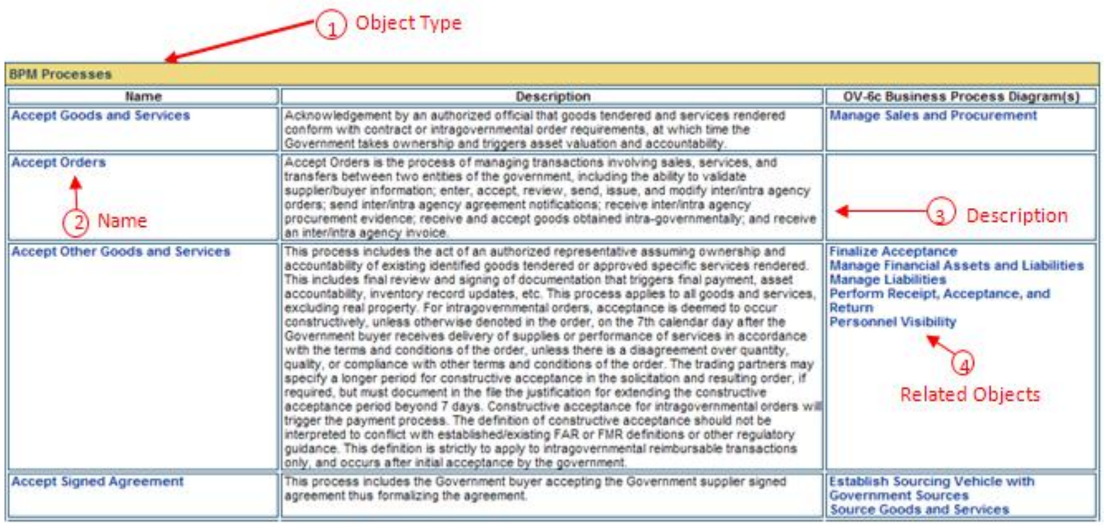
4.1.2 Model Structure

The AV-2 is a textual model in table format and consists of a glossary, a repository of architecture data, their taxonomies, and their metadata (i.e., data about architecture data), including metadata for tailored models, associated with the architecture models developed.

Each description name is classified and located in the AV-2 by:

1. Object Type: The major classification scheme in the AV-2.
2. Name: The specific name of an instance of the Object Type.
3. Description: The full description (definition) of the listed Name.
4. Related Object: Terms pertaining to certain Object Types in the AV-2 are also defined as related to other objects to assist the user in interpreting the Term based on the context of the relevant relationships in the BEA.

An example from the AV-2 for Business Capabilities is provided in Figure 4-1, AV-2 Integrated Dictionary, to illustrate a Business Capability that is related to multiple Operational Activities.



BPM Processes		
Name	**Description**	**OV-6c Business Process Diagram(s)**
Accept Goods and Services	Acknowledgement by an authorized official that goods tendered and services rendered conform with contract or intragovernmental order requirements, at which time the Government takes ownership and triggers asset valuation and accountability.	Manage Sales and Procurement
Accept Orders	Accept Orders is the process of managing transactions involving sales, services, and transfers between two entities of the government, including the ability to validate supplier/buyer information; enter, accept, review, send, issue, and modify inter/intra agency orders; send inter/intra agency agreement notifications; receive inter/intra agency procurement evidence; receive and accept goods obtained intra-governmentally; and receive an inter/intra agency invoice.	
Accept Other Goods and Services	This process includes the act of an authorized representative assuming ownership and accountability of existing identified goods tendered or approved specific services rendered. This includes final review and signing of documentation that triggers final payment, asset accountability, inventory record updates, etc. This process applies to all goods and services, excluding real property. For intragovernmental orders, acceptance is deemed to occur constructively, unless otherwise denoted in the order, on the 7th calendar day after the Government buyer receives delivery of supplies or performance of services in accordance with the terms and conditions of the order, unless there is a disagreement over quantity, quality, or compliance with other terms and conditions of the order. The trading partners may specify a longer period for constructive acceptance in the solicitation and resulting order, if required, but must document in the file the justification for extending the constructive acceptance period beyond 7 days. Constructive acceptance for intragovernmental orders will trigger the payment process. The definition of constructive acceptance should not be interpreted to conflict with established/existing FAR or FMR definitions or other regulatory guidance. This definition is strictly to apply to intragovernmental reimbursable transactions only, and occurs after initial acceptance by the government.	Finalize Acceptance Manage Financial Assets and Liabilities Manage Liabilities Perform Receipt, Acceptance, and Return Personnel Visibility
Accept Signed Agreement	This process includes the Government buyer accepting the Government supplier signed agreement thus formalizing the agreement.	Establish Sourcing Vehicle with Government Sources Source Goods and Services

Figure 4-1, AV-2 Integrated Dictionary

As a consequence of the BEA internal integration, there are cases where the same description name may be listed separately in the AV-2 under multiple object types, but with different descriptions due to its usage and context in the BEA. There are also limited cases where the same description is used for different names in different object types due to the development process for that description name.

4.1.3 Relationship to Other BEA Models

The AV-2 relates to all the other BEA models. In particular:

- All objects from selected deliverable object types (e.g. Operational Activity, Entity or Business Rule) from other BEA models must be listed and defined in the AV-2.
- Any terms with specialized meaning must be listed in the AV-2; these specifically include, but are not limited to the descriptions of all deliverable architectural object types.
- **Exception:** Terms from LRP and Defense Financial Management Information Retrieval (DFMIR)/Federal Financial Management Information Act (FFMIA) Guidance Statements are specifically excluded from the requirement to be captured in the AV-2.
- Any acronyms used within the architecture must be included in the AV-2 Acronym Definitions report, subject to the scope defined in the next section.

4.1.4 AV-2 Model Definitions

The AV-2 does not have model specific terms.

4.2 Developing the AV-2 Model

4.2.1 Pre-Development Tasks

There are no pre-development tasks related to the AV-2 Model.

4.2.2 Development Tasks

The AV-2 is generated automatically during development of the architecture models in the SA encyclopedia. The two exceptions are Acronyms and Term definitions pertaining to architecture models, which are entered manually in the Terms and Acronyms definitions in SA in order to appear in the AV-2.

Whenever an Acronym or a Term is used in a changed or new DoDAF object description, the existing Acronym List or Term List should be examined to see whether the Acronym or Term has been previously defined. If the Acronym or Term exists, then the description of the Acronym or Term should be reviewed to determine whether it is the correct description within the context use of the acronym or term.

The scope of the AV-2 Acronym Definitions report and Terms Definitions report cover all the BEA models that are in the SA encyclopedia and the AV-1 Overview and Summary Information document. AV-2 Acronyms and Terms that also appear on the BEA Summary, the ETP, BDM, APG, the Congressional Report[1], and overall summary documents and Web pages must be applied the same description set forth in the AV-2.

Acronyms introduced in Laws, Regulations and Policies (LRP) descriptions, Business Guidance statements, Configuration Management (CM) documents and the BEA Development Process documents are not required to be listed in the AV-2. The AV-2 Acronym Definitions report contains four types of character groupings:

1. **Abbreviation:** A shortening of a word or phrase.
2. **Acronym:** An abbreviation that forms a word.
3. **Initialism:** An abbreviation that uses the first letter of each word in a phrase. Also known as mnemonic.
4. **Short name:** A term that is a shortened version of a longer name. This grouping is reserved for the names of systems. Details of a particular short name may be found in Defense Information Technology Portfolio Repository (DITPR).

For the purpose of generating the AV-2 Acronym Definition report, the first three character groupings (abbreviations, acronyms and initialisms) are all considered "Acronyms". Although additional information about an acronym may be found within the BEA, the acronym is only defined in the AV-2 and not within the description of a particular BEA model. New or changed BEA model descriptions that include acronyms must follow this rule. It is anticipated that once Business Guidance items are converted to Business or System Rules, acronym use in all BEA model descriptions will be changed to conform to this rule.

Exception: Acronyms may be defined in Business Guidance items, LRP, ETP, BEA Summary, Congressional Report, and overall summary documents and Web pages. In particular:

- The BEA Summary and overall summary documents may also include an Acronym list that covers the scope of the document. For these documents, an acronym is described at first use

[1] Formerly known as the March Congressional Report

in the document and entered into the acronym list within the document. Subsequent uses of the acronym do not re-describe the acronym.

- Web pages may include a description with each use of the acronym to provide a quicker understanding by a more casual user or reader.

Conversely, a short name that is included in the AV-2 Acronym Definitions report can also be described within the description of a BEA model because a short name typically pertains to a system or application name that is used only within SV model descriptions. Additional information about a component system may be found outside of the BEA in DITPR.

In certain cases, the same acronyms or short name may have multiple descriptions. To avoid confusion, the term is listed only once in the AV-2 Acronym Definition report, with multiple entries in the description field. The BEA user must determine the correct description within the context of the use of the acronym. For example, the acronym “FMS” corresponds to both Financial Management System and Foreign Military Sales.

The AV-2 Term Definitions report contains words and phrases that are recognized by BEA users as having a specialized meaning within the BEA or DoD business community, which differs in scope of use from the typical dictionary definition.

To be included in the AV-2, a term must have a precisely limited meaning in a particular context and it should qualify as being:

- **Basic**: Not derived or computed from other terms.
- **Atomic**: Not divisible, collective or composite.
- **Knowable**: Not events or actions.
- **Unambiguous**: Not subjective.

Generally, each word in a Term is capitalized. There may be exceptions where common, generally accepted use of a Term does not follow this convention. Spelling and capitalization of a Term should be identical in the SA object description and the AV- 2 Term Definition list.

A term must be described in the AV-2 Term Definitions report within the context of its use. To avoid confusion when a Term has different uses with distinct definitions, the term is listed only once in the AV-2 Term Definitions report with multiple entries in the description field. The BEA user must determine the correct description within the context of the use of the term. In the example shown below, the term “Transition” may be applied to both the non-system context and the system context, but the term “Transition Type” has a distinct definition for each.

Transition	The act of carrying out business transformation - moving from one state to another.

Transition Type	Non-system - Policy Change - Process/Organizational Improvement - Outsourcing System - Modification or re-use of an existing system - Modification or use of an existing procurement - New Initiative

As a rule, a term is described only in the AV-2 Terms Definitions list and not in the description of a BEA model. New or changed BEA model descriptions that include terms must follow this rule. It is anticipated that over time, term use in all BEA model descriptions will be changed to conform to this rule.

Exception: Terms listed in the AV-2 may also be defined in the Business Guidance items, LRP, the ETP, BEA Summary, Congressional Report, and overall summary documents and Web pages. However, the term must be applied the definition set forth in the AV-2.

The Stakeholder that develops a BEA model also drafts the initial description of an Acronym or Term used in conjunction with the BEA model. The BEA Development Team reviews the description and may propose refinements, subject to approval by the team that developed it, before entering it into the AV-2. The Stakeholder may also provide key source information and other descriptive information, where deemed helpful.

With the exception of acronyms and terms manually entered into the AV-2, all AV-2 definitions are automatically generated from the BEA models in the SA encyclopedia. Any changes to these definitions must be made to the BEA models that introduce that specific term, as well as the definitions regenerated for the AV-2.

4.2.3 Post–Development Tasks

As part of the integration review for each model, all AV-2 descriptions are confirmed, both auto-generated and manually input names and descriptions. The object names and descriptions published in the AV-2 must be reviewed for spelling and grammar errors. Use MS Word to spell check the names of terms and acronyms sections of the AV-2.

- Add all of the valid terms and acronyms to the Word dictionary.
- Spell and grammar check the entire AV-2 names and definitions.
- Validate and confirm errors along with proposed solutions with stakeholders.

4.3 Modeling the AV-2 Model Using SA

4.3.1 AV-2 Modeling Conventions

Refer to each model section for modeling conventions of the AV-2 objects.

4.3.2 Modeling AV-2 Objects

Refer to each model section for modeling of the specific AV-2 objects.

4.3.3 AV-2 Best Practices

This section discusses best practices, including lessons learned from previous architecture development efforts and common modeling pitfalls to avoid.

4.3.3.1 AV-2 Lessons Learned

- Ensure that the AV-2 is generated after all of the BEA models are stabilized.

- Perform regular and early communication with other architects to assess impact of proposed changes in other models on the AV-2.
- Review BEA descriptions to ensure all shared Definitions, Acronyms and Terms have the same descriptions in related documents, such as the BEA AV-1, BEA Summary, BDM, ETP, and the Congressional Report.
- Ensure that all exception reports are reviewed and resolved.

4.3.3.2 AV-2 Common Pitfalls

- Failure to ensure that all relevant BEA Acronyms and Terms are included in the AV-2.
- Failure to coordinate AV-2 Acronym and Term Lists with the BEA Summary, ETP, BDM, APG, the Congressional Report, and overall summary documents and Web pages.

5 OV-5b – Operational Activity Model

5.1 Summary Description

This section describes the OV-5b Operational Activity Model, its relationship to other BEA models, the development method, and the modeling guidelines used which are to be followed.

5.1.1 Model Purpose

The OV-5b Operational Activity Model is the cornerstone architecture model for the BEA. The OV-5b is used to describe the operations that are normally conducted in the course of achieving a mission or a business capability.

The OV-5b describes what DoD does in the BMA, but does not address how the activities are performed or the sequence of those activities. Other Enterprise Architecture models are built from and aligned with the Enterprise OV-5b Operational Activity Model. For the BEA, there shall be a single Enterprise-level OV-5b Operational Activity Model to represent the DoD Business Mission.

The BEA OV-5b Operational Activity Model defines a set of Operational Activities, their relationships, and information requirements needed to perform major DoD business operations. The model incorporates industry and government leading practices, examines doctrine and policy implications, and defines operational requirements.

The OV-5b Operational Activity Model can be used to:

- Support alignment of services, systems and solutions to the prioritized strategic capabilities of the Department.
- Distinguish CBM-specific Activities.
- Define or flag issues, opportunities, or information dependencies among Activities that need to be scrutinized further.
- Identify data pertinent to the Enterprise.
- Identify the context or framework of how certain aspects of the business will be performed.

5.1.2 Model Structure

The OV-5b model is depicted as a set of diagrams. It comprises a single BEA OV-5a Operational Activity Decomposition Tree, or simply Decomposition Tree, and a series of OV-5b Operational Activity Model diagrams, as described in the following sections.

5.1.2.1 OV-5a Operational Activity Decomposition Tree

The first diagram constructed during development of the architecture is the OV-5a Operational Activity Decomposition Tree, a functional, hierarchical decomposition of Enterprise-level Operational Activities for the DoD BMA. The activities on the Decomposition Tree are created and defined to meet the following objectives:

- Adherence to DoD business needs.

- Compliance with DoD Enterprise Standards.
- Alignment with the direction of the "To Be" world.
- Alignment with the Federal Enterprise Architecture (FEA) Business Reference Model (BRM), where possible.

The Decomposition Tree is decomposed to a level that identifies the lowest-level activities that are performed at the Enterprise-level to support associated Business Capabilities. While multiple CBMs may be identified as stakeholders for an Operational Activity, meaning the activity is relevant to them, a primary CBM is responsible for defining its basic content. Further decomposition below the Enterprise level is the responsibility of the Components with a goal toward identifying Federation touchpoints to the Enterprise level.

Figure 5-1, OV-5a Operational Activity Decomposition Tree Example, illustrates the proper form for an Operational Activity Decomposition Tree diagram. (It should not be considered as the current Decomposition Tree.) The figure depicts the complete model and a snippet that is more legible.

There will be a single Operational Activity Decomposition Tree diagram for each BEA release. The top box of the Decomposition Tree diagram shall be a single Operational Activity that is shown on the Context Diagram (A-0) of the Activity Model. All remaining Activities that appear in the OV-5b Operational Activity Model shall then be arranged according to their proper Parent-Child relationships beneath the A-0 activity.

The white Operational Activities represent activities that have been integrated into the BEA, meaning they are on an OV-5b Operational Activity diagram with the Inputs, Controls, Outputs and Mechanisms (ICOMs) identified. The blue Operational Activities that appear only on the OV-5a Operational Activity Decomposition Tree represent activities which will be used for future development and/or portfolio management to categorize systems.

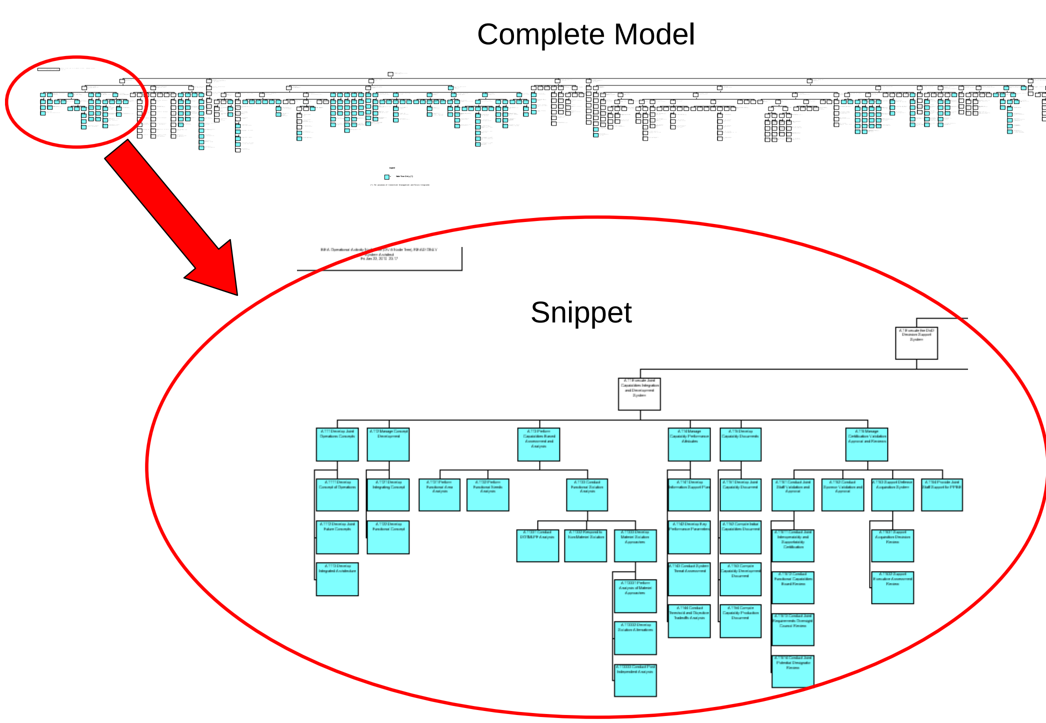


Figure 5-1, OV-5a Operational Activity Decomposition Tree Example

Based on the approved Decomposition Tree, the OV-5b models will be created or updated for non-leaf activities designated as "Integrated". The ICOMs for these activities are identified during workshops. All Operational Activities definitions must be distinct and expressed at their level of decomposition. The Operational Activity definitions will be updated during future work, such as during workshops, when ICOMs are added to the models and the activity definitions are updated to reflect them.

5.1.2.2 OV-5b Operational Activity Model

The OV-5b Operational Activity Model represents the Operational Activities and their information dependencies between Activities within the DoD BMA and external to the DoD BMA.

Associated with each Operational Activity, at any level of decomposition, is a description of the Activity, required information Inputs and Outputs of the Activity, Controls or Data Enterprise Standards that direct or constrain the Activity, and Mechanisms that identify CBM(s) and the System(s) that perform the activity.

Figure 5-2, Example of OV-5b Model for Manage Accession, shows an Activity model for Manage Accession supporting the Human Resources Management CBM. (It should not be considered as the current BEA model.)

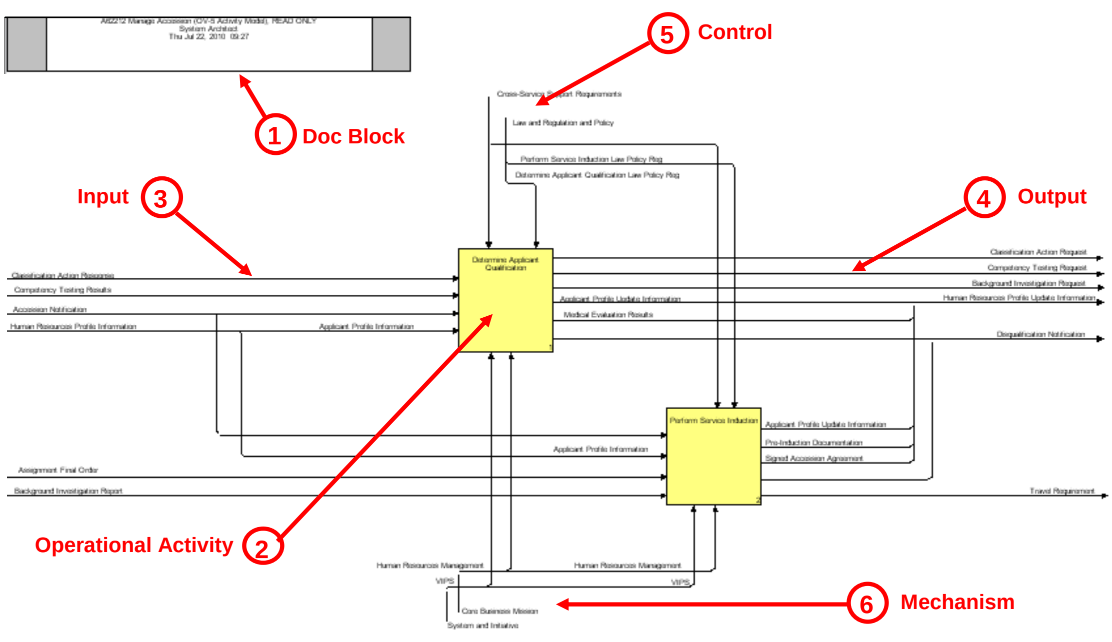


Figure 5-2, Example of OV-5b Model for Manage Accession

As indicated in Figure 5-2, the objects or main features used to represent the OV-5b Operational Activity Model are:

- **Doc (title) block (1),** located in the upper left corner of the diagram. The title block contains the diagram name and type in the format "A62212 Manage Accession (OV-5b Operational Activity Model)", as well as the last modification date. The double gray bars indicate that the diagram is a decomposition of a parent activity.
- Operational Activities (2), **shown as** the yellow rectangular shapes in the diagram. Two Operational Activities are shown in the example: "Determine Applicant Qualification" and "Perform Service Induction" which support Manage Accession. Operational Activities create or transform Outputs based on the Controls and Inputs
- ICOMs **(3, 4, 5, 6)**, shown as arrows on the diagram, represent the Inputs, Controls, Outputs and Mechanisms for the Operational Activities.
    - (3) **Inputs** – Information that is transformed or consumed by the Operational Activity;
    - (4) **Outputs** – Information that is produced by the Operational Activity;
    - (5) **Controls** – Identifies what Laws, Regulations and Policies and Data Enterprise Standards constrain the Operational Activity;

      (6) **Mechanisms** – The means used by an *Operational Activity* to transform *input* to *output*; examples are a *Service, Organization, System or Person Type (*Same as the DoDAF Performer).

    - The BEA OV-5b Operational Activity Models are developed in accordance with the specifications contained in this document, which are derived from DoDAF and IDEF0 modeling techniques. Exceptions to these standards that are required to develop this model shall be accommodated with appropriate approved authority and shall be incorporated within this document after each deliverable as the cornerstone model. The Enterprise OV-5b Operational Activity Model will be used as a guide to restructure and update other OV and SV models as development continues in future releases of the BEA. This will ensure a complete, integrated and consistent architecture.

5.1.3 Relationship to Other BEA Models

As illustrated in Figure 5-3, Relationships between OV-5b and Other BEA Models, the OV-5b Operational Activity Model is related to other BEA models as follows:

AV-1	The scope of the development effort for each CBM for a development cycle, as disclosed in the AV-1, will determine if the OV-5b is affected in the release.
AV-2	All OV-5b terms with specific meaning must be defined in the AV-2 Term definitions report. These terms must include, at a minimum, all the following object types: • ICOM Arrow Definitions • Operational Activity Definitions All acronyms used in the OV-5b descriptions must be listed and spelled out in the AV-2

	Acronym Definitions report.
OV-2	The Operational Resource Flow Description captures the resource Flows exchanged between operational activities. Operational Nodes in the OV-2 represent logical groupings of leaf-level[2] Operational Activities from the OV-5b. The Activities are assigned to the Operational Nodes in the OV-2, and related leaf-level Inputs, and Outputs from the OV-5b are then translated to IEs that depict the required information flow represented on the OV-2 as Need Lines between Operational Nodes. The Operational Nodes are CBM(s) and are represented as Mechanisms on the OV-5b Operational Activity Models.
OV-3	Each leaf-level Input and Output ICOM Arrow on the OV-5b model connecting Operational Activities in Operation Nodes is represented as one IE in the OV-3.
OV-6a	**Note:** For BEA 6.0, a decision was made that Business Rules shall only be linked to the OV-6c Process Steps, not to the OV-5b.
OV-6c	Process Steps in the OV-6c Event Trace Description are derived from and linked to Operational Activities in the OV-5b. Data Objects in the OV-6c are related one to one to OV-3 IEs, which are mapped to the Inputs and Outputs of Operational Activities in the OV-5b Operational Activity Model.
CV-2 **CV-6**	OV-5 Operational Activities are linked to the capabilities they support.
DIV-1 **DIV-2** **DIV-3**	Entities or Attributes within Entities in the DIV Model are derived from or directly linked to Inputs and Outputs on the OV-5b via the IEs in the OV-3.
SvcV-1	Enterprise Systems that provide system services used to perform activities are Mechanisms on the OV-5b
SvcV-5	OV-5 Operational Activities are linked to the Services in the SvcV-5.
SV-1	Enterprise Systems that perform system functions used to perform activities are Mechanisms on the OV-5b
SV-5	Operational Activities from the OV-5b are mapped to System Functions in the SV-5. The BEA development team has created a modified SV-5 that shows Operational Activities mapped from Business Capabilities and to System Functions with the identified Enterprise System, if available, in the intersection.

[2] Leaf-level Operational Activities are the lowest-level white Operational Activities.

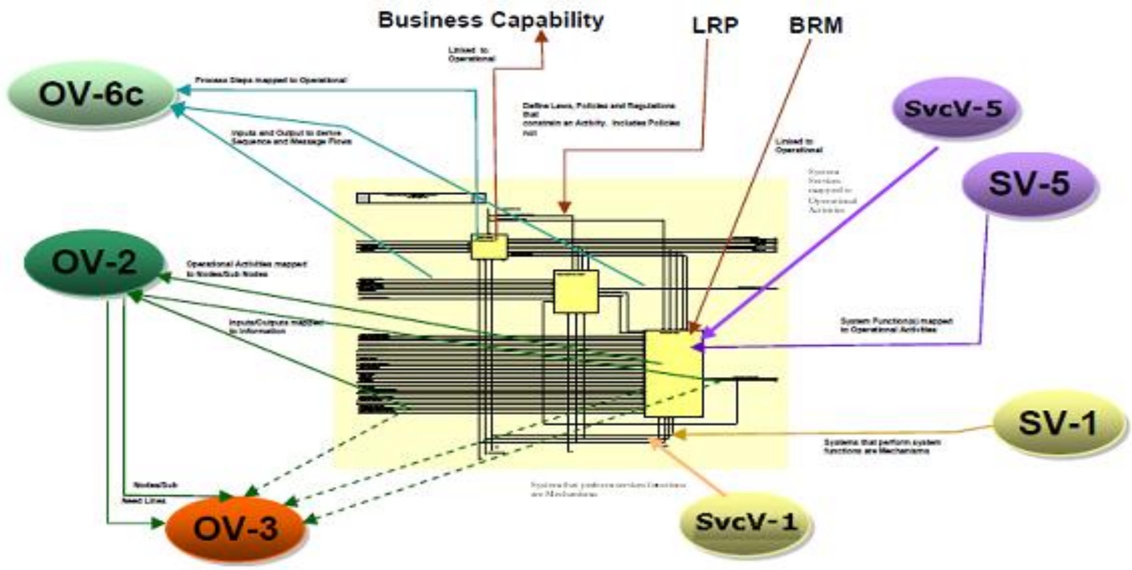


Figure 5-3, Relationships between OV-5b and Other BEA Models

5.1.4 OV-5 Definitions

5.1.4.1 OV-5a Operational Activity Decomposition Tree Definitions

The following are definitions of the key elements contained in the OV-5a Operational Activity Decomposition Tree:

- **Parent Activity:** An Operational Activity that is decomposed into two to nine Operational Activities, or Child Activities. The definition of the Parent Activity is the sum of the child Operational Activities and serves to set the scope of its decomposition.
- **Child Activity:** An Operational Activity that is a decomposition of a parent Operational Activity. It represents a functional aspect of its Operational Activity.
- **Hierarchy Chart Connectors:** These lines connect a Parent Activity to its Child Activities and show the relationships between activities.

5.1.4.2 OV-5b Operational Activity Model Definitions

The following are definitions of the key elements contained in OV-5b Operational Activity Models:

- **Operational Activity:** An action performed in conducting the business of an enterprise. This is a general term that does not imply a placement in a hierarchy or a timing sequence (for example, 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).
- **ICOM Arrows:** Represent the **I**nputs, **C**ontrols, **O**utputs and **M**echanisms that define information relationships in an Activity Model. In particular:
  - Inputs: Information received from another Operational Activity, either internal or external to the model, which is needed for the given Operational Activity to be carried out.
  - Controls: Information that guides or constrains the way an activity is performed.

- In the BEA, there are two types of Controls: External and Internal.
  - External Controls are decomposed from the Laws, Regulations and Policies parent Control and the definition of an External Control is generated based upon mappings of LRP to process steps as maintained in the LRP Repository. The External Control Information Assurance is not decomposed and is tunneled into the architecture[3]. External Data Enterprise Standards that represent Controls are those that are not created by a BEA Activity.
  - Internal Controls are Enterprise Standards that are created as Outputs from other Operational Activities within the BEA OV-5b Operational Activity Model. Internal Controls, while Outputs from other BEA Operational Activities, are not depicted as IEs in the OV-2 or OV-3 models.
    - Enterprise Standards, both External and Internal, can be one of the following:
    - **Data Enterprise Standards**
    - Standards that pertain to the exchange or domain values of attributes or data elements which are within the purview of the BEA.
    - **Policy Enterprise Standards**
    - Standards that pertain to the Laws, Regulations Policies or other guidance that constrains the development of the BEA.
    - **Process Improvement Enterprise Standards**
    - Standards that pertain to the efforts that are related to improving the processes within the scope of the BEA.

An Enterprise Standard can be a parent of a set of Enterprise Standards. This allows an Enterprise Standard to be composed of a set of smaller Enterprise Standards that increase the granularity of information captured in the BEA and reported from the BEA.

- **Outputs:** Information that has been transformed or created by the Operational Activity and sent to another internal Operational Activity or to an external activity (outside the scope of the BEA BMA model/viewpoint). An Output produced by a leaf level Operational Activity must be unique to that Operational Activity. That is, more than one leaf level Operational Activity cannot produce the same Output.
- **Mechanisms:** Action items that perform activities such as service performers, systems, personnel types and organizations. BEA identifies which CBM, System, or Service, performs the Operational Activity as well as any Enterprise Standard that will become a System or Service in the future. BEA Mechanisms are CBMs, Services, Systems or Enterprise Standards, as defined by the Stakeholders.
  - **Existing Systems Enterprise Standards (Milestone B and beyond):** Shown as Mechanisms on the OV-5b and as systems on the SV models.

[3] A tunneled arrow is an arrow (with special notation "()") that does not follow the normal requirement that each arrow on a diagram must correspond to arrows on related parent and child diagrams also called balancing the Parent-Child diagram.

- **Future Systems Enterprise Standards (pre-Milestone B):** Shown as Mechanisms on the OV-5b and not shown as systems on the SV models.

5.2 Developing OV-5 Models

This section describes the approach to develop, extend and maintain the OV-5b Operational Activity Model. The OV-5b is developed in SA as a diagram. In accordance with DoDAF, the BEA requires a single, integrated OV-5b Operational Activity Model that:

- Covers the scope of the DoD BMA.
- Incorporates results from Subject-Matter Expert (SME) attended workshops.
- Aligns Operational Activities with the FEA BRM

Such a single, integrated, Enterprise Activity Model provides the context for linking and grouping supporting Operational Activities within the DoD BMA, and provides a starting point for the development of more detailed activity models built by the CBMs.

The Enterprise-level OV-5b Operational Activity Model defines the "To Be" activities and business information requirements to optimize DoD business operations in support of the warfighter. Specifically, the Enterprise OV-5b will be an integrated architecture model used to identify business information, Systems/Enterprise Standards, constraints and activities as a basis for the rest of the enterprise architecture.

Using a spiral development approach through facilitated workshops, Business Analysts, Modelers and Architects provide functional and technical subject-matter expertise to perform tasks in every step of development of the OV-5b model.

As described above, the OV-5a Operational Activity Decomposition Tree is the initial model to be constructed; it is a functional hierarchical decomposition of DoD Enterprise Activities that bounds what DoD does across the BMA. These Enterprise Activities are reconciled against DoD business needs and with any existing OV-5b Operational Activity Models. FEA BRM functions also are analyzed during construction of the Decomposition Tree and any necessary adjustments are made to the alignment or the content of the Decomposition Tree. During this work, gaps and additional work are identified in the Enterprise Operational Activities. The Gap Analysis is performed in conjunction with the Planned Capability Improvements identified in the CBM AV-1 and the related BIP created for each proposed body of work to improve the BEA. These activities form the basis for selection of Change Requests (CR) for each architecture model. Identified gaps are addressed during a deliverable or noted for future work. Identified gaps in the FEA BRM are noted and passed to BEA development team.

Once the Decomposition Tree is defined, stabilized and approved, Enterprise-level OV-5b Operational Activity Models are created for each integrated Operational Activity of the Decomposition Tree. ICOMs are added to the models during workshops. The models show the information dependencies between Operational Activities as ICOMs. ICOMs are based on approved sources, to include existing Operational Activity Models and other industry reference materials or Government-Furnished Information (GFI).

5.2.1 Pre-Development Tasks

The tasks that must be completed prior to OV-5 development and/or maintenance are:

- Review the stakeholder's AV-1 and BIP to understand the impact of the planned body of work on the OV-5b Operational Activity Model.
- Identify models that could potentially be affected by OV-5 changes and begin coordination within the BEA Development Team. A thorough analysis of the existing SA USRPROP.*txt* file should be performed prior to the start of any extension or maintenance activities. The SA tool uses this file to add, delete and modify properties associated with architecture models and objects, such as Operational Activities and ICOMs. The USRPROP.*txt* file also allows the specification of linkages to other architecture definitions, such as FEA BRM Sub-functions as well as the OV-6c. This analysis provides a list of current tool configuration modifications contained in the USRPROP.txt file. The list will be assessed to:
- Identify modifications no longer needed to support the BEA.
- Identify modifications that will be needed to support extensions and/or modifications to the BEA.
- As necessary, changes to the USRPROP.txt file are proposed and approved through a configuration management process. The BEA Development Team implements these changes.

5.2.2 Development Tasks

The development and maintenance of the OV-5b Operational Activity model is accomplished in facilitated workshops that include Government Subject Matter Expert (SME) participation to address content and validate results. The following subsections describe the approach used to develop the Enterprise OV-5b Operational Activity Model for the BEA. Each subsection sets forth the specific tasks that must be accomplished in each stage of the development phase. Although most of these steps are sequential, it is common to start some steps before a previous step is completed.

5.2.2.1 Creating/Modifying the OV-5 Model

This section describes the approach to develop the OV-5b Operational Activity Model.

5.2.2.1.1 Update Operational Activity Decomposition Tree

Reviewing and updating the Decomposition Tree is the first modeling step when implementing any BEA Improvement Proposal. The Node Tree shows in a hierarchical fashion the proper grouping and decomposition of activities. It provides a navigation map for the OV-5b Operational Activity Models, which are based on the white Operational Activities on the Decomposition Tree.

The Decomposition Tree is decomposed to a level that identifies the lowest-level activities that are performed at the Enterprise level to support associated Business Capabilities.

The tasks that must be completed prior to the OV-5b Operational Activity Model development and/or maintenance are:

1. The leaf-level Operational Activities within each branch of the Decomposition Tree are linked to Business Capabilities and to Sub-functions of the FEA BRM. These linkages are maintained in the BEA, as described in Section 6. The Decomposition Tree is updated in workshops as a cooperative effort between Government SMEs and appropriate BEA technical and functional representatives.

2. During development of the Decomposition Tree, stakeholders shall be identified on the appropriate tab in the Operational Activity definition. Process Steps and System Functions will be mapped to the correct Operational Activity later in the process.

3. Use the procedure below to maintain the Decomposition Tree:

   - Create and define Operational Activities during CBM Workshops that cover the scope areas as defined in the BIPs, based on approved sources. Approved sources include existing approved OV-5b Operational Activity Models; CBM SMEs; and FEA BRM Lines of Business and Sub-function definitions.
   - Arrange activities in the Decomposition Tree and normalize the arrangement by eliminating overlapping and redundant activities. The top level of the Decomposition Tree shall have one Operational Activity, which sets the scope of the BEA in general and the OV-5 architecture model in particular. Subsequent Operational Activities shall be decomposed and arranged, where appropriate, into two to nine Operational Activities.
   - Align the Operational Activities with the Lines of Business and/or Sub-functions in the current FEA BRM that should be reflected in the OV-5. Gaps in the FEA BRM should be noted as feedback to the FEA BRM community; gaps in the BEA itself should be noted as areas for possible future BEA improvement.
   - Review artifacts from prior approved versions of the BEA relevant to current or new OV-5 development. Create or refine existing definitions of Operational Activities based on this review.
   - Validate the resulting Operational Activity definitions.

5.2.2.1.2 Generate OV-5b Activity Model Diagrams

The SA tool can automatically generate OV-5b Operational Activity Model diagrams from an OV-5a Operational Activity Decomposition Tree. However, this feature was only available for the initial setup of the OV-5 architecture model as part of BEA 1.0. Any subsequent changes to the OV-5 models must be incorporated manually, since SA cannot automatically generate updates to Activity Model diagrams from changes made in the OV-5a Operational Activity Decomposition Tree. Each time the utility is run it generates a completely new set of activity model diagrams, replacing only the diagrams modified by the BEA Development Team.

Therefore, attention to detail is required when updating the OV-5 to ensure that the OV-5a Operational Activity Decomposition Tree and all affected models are updated to the level of decomposition approved for each new or modified Operational Activity to avoid misalignment among the OV-5b Operational Activity Models or between the OV-5b Operational Activity Model and other architectural models.

The following diagrams comprise the OV-5b Operational Activity Model:

- The A-0 diagram, which is the top-level Operational Activity Model diagram created when developing the OV-5b Operational Activity Model. The A-0 is called the Context Diagram and is composed of a single activity box with a name that encompasses the entire scope of the Enterprise being described. The A-0 uses ICOM Arrows entering and leaving this box to represent interfaces of the Enterprise to its external environment, and those ICOMs are then carried down to subsequent child diagrams. A text box shall be added detailing the Purpose and Viewpoint of the Model.

- The A0 diagram, which is a child diagram of A-0, and decomposes the single upper-level context activity into two to nine child activities. The A0 diagram is recognized as the primary diagram of an OV-5 model, clearly showing the high-level DoD activities.

  Note: While IDEF0 conventions suggest that a given Activity Model diagram shown comprise no less than three or more than six activities, BEA requirements have necessitated this variation in the degree of decomposition for a Parent Activity.

- The A1 set of diagrams, which are children of the A0 and decompose a single A0 Operational Activities into two to nine sub-activities. The OV-5b Operational Activity Models and the Operational Activities contained within them must be decomposed to a level that supports the CBM, as necessary to answer questions (required outcomes). If the level of decomposition does not clearly show how it supports a particular CBM capability, then the Operational Activity must be decomposed to expose the capability. The ICOMs attached to the activities must be at the level of detail reflecting the level of Operational Activity to which it is attached.

Each of the Enterprise-level OV-5 models shall have viewpoint and purpose statements. These statements shall be placed in a text box on the A-0 diagram.

Each diagram in the OV-5b Operational Activity Model shall have a diagram description. The description shall describe the activities displayed on the diagram, the main themes characterizing their interactions, as depicted by ICOMs. Specific ICOMs should not be named, but the general type(s) of information contained within the ICOMs or the LRP or Systems represented by them should be included in the diagram description.

ICOMs

The ICOMs are the means by which information and relationships between Operational Activities are represented on the OV-5b Operational Activity Model diagram(s). Many other architecture models use or refer to these ICOMs therefore, it is imperative that ICOM names be representative of the business information they represent.

To identify and define ICOMs for the BEA OV-5b Operational Activity Model:

- Create the Inputs and Outputs on the appropriate OV-5 diagrams.

  Note: For Inputs and Outputs that originate from or go to Entities that are external to the BEA, develop generic external activities such as "Process Treasury Information", and populate the checkbox identifying the activity as external in SA. This activity is then assigned as a source or destination to the appropriate ICOM and appear in the OV-3.
- Define Inputs and Outputs. The definitions for said Inputs and Outputs should include a consumable-oriented noun.
- Identify External Controls. These originate outside the BEA and will be decomposed from the *Laws, Regulations and Policy* parent Control.
  - External Control definitions related to LRP are generated by a macro through the mapping of LRP to Process Steps in the LRP Repository, which are then mapped to

Operational Activities. The macro creates the ICOM definition with a naming convention of {*Operational Activity Name*} *Law Regulation Policy*; if a leaf-level Operational Activity is not mapped to an OV-6c Process Step then that Operational Activity will have the Parent Control "Law and Regulation and Policy". The OV-5 Architect will then add the Control to the diagram going into the appropriate activity.

  - External Initiatives are decomposed from the LRP Control. For the definition of these external initiatives, the definition should include control-oriented nouns that indicate elements of control and not performance or consumption.

- Identify Internal Controls. These Controls, generated by an activity within the BEA OV-5b Operational Activity Model, will follow the same naming conventions as Inputs and Outputs, not the external LRP Control.
- In conjunction with SMEs, identify and link Mechanisms to Operational Activities in the OV-5b in accordance with which CBM and the System or Enterprise Standard, if possible, that would perform the assigned Operational Activity. As defined in section 5.1.4, Mechanisms are the Systems or CBMs that actually perform some part of the Operational Activity and not just send or receive information to/from the activity.
- Create IEs from Input and Output ICOMs associated to leaf-level Operational Activities, but not Internal Controls. The ICOM Arrow shall be linked to the created IE in the SA tool.
- General rules for adding ICOMs to models:
  - Each Operational Activity will have one or more Inputs or Controls.
  - Each Operational Activity will have one or more Mechanisms.
  - Each Operational Activity will have one or more Outputs.

ICOM Bundling

To Bundle ICOMs for the BEA OV-5b Operational Activity Model:

- ICOM bundles will be created based upon the source and destination of ICOMs or to increase the functional understanding of the model. No changes will be made to the leaf-level models except to add parent ICOMs with appropriate ICOM Arrow forks and joins to connect the leaf-level ICOMs to their parent. Parent ICOMs are not shown if they were decomposed at a higher intermediate level of the activity model. Assign child ICOM to the parent ICOM.
- Identify A0 Level ICOMs that come from the same external node or go to the same external node that may be bundled. The A0 Level parent ICOM shall be decomposed in one or more intermediate level parent ICOMs, determined by the destination activities on each intermediate level of the Operational Activity model. For all other ICOMs (used by a CBM or between CBMs), a single ICOM will be created to bundle multiple ICOMs with the same source and destination Operational Activity.

5.2.2.1.3 OV-5 Model Coordination with the Stakeholders

The business analysts will closely review any changes made to any other BEA models and assess the impact of such changes on the OV-5. Models of particular concern, because of their close alignment with the OV-5 are the OV-6c and the SV/SvcV models System Functionality Descriptions (SV-5) and Operational Activity to Services Traceability Matrix (SvcV-5). Changes to the OV-6c Process Steps may require definition modifications in Operational Activities or remapping of Process Steps. Changes to System Functions may require corresponding changes to Operational Activities or definitions since Systems are Mechanisms and are represented in the SV-1, SV-5 and SvcV-5, while changes to the Data Objects or System Resource Flows (SRFs) may require changes to ICOMs. The OV-5s are linked to other models and those linkages are listed in Section 5.1.3, Figure 5-3 Relationship to OV-5b and Other BEA Models.

Changes to the FEA BRM could also result in changes to the OV-5. The business analysts will do periodic reviews of the FEA BRM to determine what changes have been implemented. Changes to Lines of Business and Sub-functions in the FEA BRM are assessed for the need to make corresponding changes to Operational Activities in the OV-5. At the very least, each such change to the FEA BRM requires changes to definitions of Lines of Business or Sub-functions maintained in SA and to existing linkages between these definitions and Operational Activities.

To ensure proper integration across models, a thorough analysis must be conducted with all models and definitions that impact the OV-5. Mappings from or to OV-5 Operational Activities must be validated.

The following subsections describe specific linkages that must be established for the BEA, and how those linkages are implemented using the SA tool.

Linking Operational Activities to Business Capabilities

Each CBM will identify a set of Business Capabilities. A Business Capability is the ability to execute a specific course of action. It can be a single business enabler or a combination of business enablers (e.g., business processes, policies, people, tools, or systems information) that assist an organization in delivering value to its customer.

Stakeholders review the definition of a Business Capability along with definitions of the Operational Activities to determine which Operational Activities support the Business Capability. One or more leaf level Operational Activities on an OV-5 Model can be mapped to the same Business Capability. Each leaf level Operational Activity can be mapped to one or more Business Capabilities.

Link OV-6c Process Steps to Activities

This task uses a SA utility to map BPM Processes to Operational Activities. These linkages will be identified during OV-6c workshops with the appropriate SMEs in attendance. Once the correct mappings are identified, the tab on the Operational Activity definition will be filled in.

Link FEA BRM Sub-functions to Operational Activities

This task uses the SA utility to map FEA BRM Sub-functions to leaf-level Operational Activities, as follows:

- Assign new FEA BRM Sub-functions to Operational Activities, in collaboration with appropriate SMEs.
- Assign new Operational Activities to existing FEA BRM Sub-functions.
- Reassign Operational Activities to deleted FEA BRM Sub-functions.

5.2.2.2 Model Clean-up

This activity is used to provide analysis/review support to the final delivery of OV-5 models and modeling objects. The models are assessed via numerous tools and checklists to uncover discrepancies.

Operational Activity Decomposition Tree (OV-5a) / Operational Activity Model (OV-5b) is validated against the following:

- Stakeholder AV-1.
- BIP.
- CR Content Summary.
- Model Checklists.
- Relevant BEA Reporting Service (BRS) reports.
- Encyclopedia Compare Reports.
- Threads Tool Report.

5.2.3 Post-Development Tasks

These tasks are done after the work has been approved by the stakeholders:

- Incorporate additional updates to the OV-5 based upon subsequent work sessions.
- Incorporate quality control and architecture verification changes into the BEA.
- Incorporate additional stakeholder updates based on their review.
- Incorporate additional Chief Information Officer (CIO) updates based on their review.

5.3 Modeling the OV-5b Model Using SA

5.3.1 OV-5b Modeling Conventions

The following modeling conventions shall be used to create an efficient and effective OV-5b:

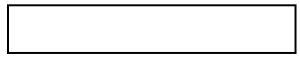
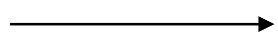
5.3.1.1 Use of Color, Size and Lines in a Diagram

Table 5-1, Modeling Guidelines for OV-5a Operational Activity Decomposition Tree

Element	Symbol	*Format*
Doc Block	Text Box:	*Position: Upper Left Corner* *Border: Solid Black* *Fill: None* *Text:* *Color: Black* *Font: Arial* *Size: Default*
Operational Activities	Rectangle:	*Border: Solid Black* *Fill:* *Operational Activity on Node Tree Only: Light Blue* *Operational Activity on OV-5b Models: White* *Text:* *Color: Black* *Font: Arial* *Size: Default*

Table 5-2, Modeling Guidelines for Operational Activity Model Diagrams

Element	Symbol	Format
Doc Block	Text Box:	Position: Upper Left Corner of Diagram Border: Solid Black Fill: None Text: Color: Black Font: Arial Size: Default
Operational Activity	Rectangular textbox:	Border: Solid Black Fill: Color: Yellow Text:

Element	Symbol	Format
		Color: Black Font: Arial Size: Default
ICOMS	Straight Arrow Connectors: 	Color: Solid Black Width: SA Default Additional restrictions: ICOM Arrows should not intersect.

5.3.1.2 Diagram Conventions

OV-5a Operational Activity Decomposition Tree

- There is only one Enterprise-level OV-5a Operational Activity Decomposition Tree diagram for the BEA in the SA encyclopedia.
- The OV-5a Operational Activity Decomposition Tree Diagram shall include a diagram description that shall be stored in the Description attribute under Diagrams Properties.
- All modeling objects shall have no truncated entries on the diagram.
- If a parent Operational Activity is decomposed on the diagram, it shall be decomposed to at least two, but no more than nine, child Operational Activities.
- The Operational Activity box label shall use title case (first letter of each word capitalized, other letters lowercase) should be non-plural (exception approved by BEA development team, and can use only the special character "-", "," and " ' ". Any acronyms used in the Operational Activity name must be from or added to the approved acronym list that is part of the BEA AV-2.
- The Operational Activity box label shall fall within the Operational Activity box border when printed.
- The Operational Activity box label shall not contain truncation indicators (dots) indicating that text is not visible.
- Operational Activity box labels and definitions shall be identical to those used in the OV-5b Operational Activity Model.
- The Operational Activity box numbers shall be sequential. Each Operational Activity node number will be assigned based on the position of the box within the model and will be generated automatically by SA. The Operational Activity numbers shall be prefaced by the capital letter "A" and will be shown at the beginning of the Operational Activity box label.

- Operational Activities must have a definition that is clear, concise and uses active voice. The definition must cover the type of information coming in (not specific Inputs), what the activity does with that information and identify the Controls that impact the Systems (if identified) that perform that activity, and what information it produces. It should also discuss major and minor flows of the activity as well what triggers the performance of the activity.
- Use the "Stakeholders" tab on the Operational Activity definition in SA to assign CBMs that have an interest in the Activity.
- The top box of the diagram shall be centered (as permitted by the tool) on the diagram.
- A Doc Block representing header information for the diagram (including the diagram name and date last updated) shall be placed in the upper left-hand corner of every diagram with no white space above or to the left of the Block. This Doc Block shall contain the title of the diagram and other pertinent information as automatically provided by the SA tool. No graphic comment shall be included. The Doc Block shall be enlarged so there are no truncation indicators (dots) indicating that text is not visible. The Doc Block shall be a box with no fill color and a black border.

Operational Activity Model

The following guidelines shall apply to the OV-5b Operational Activity Model diagrams:

- All modeling objects shall have no truncated entries on the diagram.
- A Doc Block representing header information for the diagram (including the diagram name and date last updated) shall be placed in the upper left-hand corner of every diagram with no white space above or to the left of the Block. This Doc Block shall contain the title of the diagram and other pertinent information as automatically provided by the SA tool. No graphic comment shall be included. The Doc Block shall be enlarged so there are no truncation indicators (dots) indicating that text is not visible. The Doc Block shall be a box with no fill color and a black border. For each diagram below the A-0, the Doc Block shall have gray shaded areas on the left and right side indicating a parent relationship to the activity diagram above it.
- The Operational Activity Model shall have a top-level A-0 Context Diagram consisting of a single Activity Box, labeled A0, with associated boundary ICOM Arrows representing appropriate interfaces with Activities outside the model.
- Each OV-5b diagram must be associated with an OV-5a Operational Activity on the OV-5a Operational Activity Decomposition Tree.
- In the OV-5b Operational Activity Model diagrams at the A0, A1, and A11 levels (and subsequent levels as necessary), Operational Activity boxes shall be arranged in a stair-step fashion from the upper left corner down to the bottom right corner of the page. The top of any subsequent Operational Activity must be below the top (not the bottom) of the previous Operational Activity.
- For each Diagram in the OV-5b, a text description shall be written to provide a clear, understandable narrative of what the Diagram portrays. The narrative shall describe the Operational Activities and their information interactions in general, both internal to the diagram and external. The diagram description should also attempt to discuss the main themes

of the diagram by following the critical ICOMs and their relationships to activities as shown in the diagram interactions, as well as the minor themes by following other ICOM interactions.

- During workshops, Operational Activity definitions shall be refined to reflect ICOMs. It must address the information received, what action is performed on that information, what regulations constrain the Operational Activity and what Outputs are produced by the Operational Activity.

5.3.1.3 Object Naming Conventions

- Operational Activities shall be named as verb-noun objects. They should represent succinct expressions of what the Operational Activity does, suitable to the level of Operational Activity decomposition. The Operational Activity Names must be unique and use only approved acronyms, as contained in the BEA AV-2. For new acronyms, the acronyms must be added to the SA Acronym Definition according to the AV-2 checklist.
- The only special characters allowed are "-", "," and " ' ".
- Use Title Case; the first letter of each word in an object name shall be uppercase; other letters should be lowercase. Incidental words, such as prepositions within the object name ("with," "at," "in," "and" or "the"), shall be all lowercase.
- Object names shall use the singular form (no plurals) with exceptions approved by the Chief Architect.
- Object names shall be spelled correctly and shall not use future tense.

5.3.2 Modeling OV-5 Objects

The following subsections provide guidelines for the individual elements or components that comprise the Operational Activity Model Diagrams.

5.3.2.1 Operational Activity

- All Operational Activities must be defined. Definitions should reflect the information transformation, creation and consumption actions performed by the Operational Activity. Each definition must be clear, concise, use active voice, and comprise complete, grammatically correct sentence.

Example of a good Operational Activity definition:

> **Manage Entitlement**: This activity includes calculating the amount to be paid as a result of a commercial vendor having provided materiel or services to the Department of Defense. The activity verifies funding availability for payment along with conducting a three-way match between the contract, the receiving report, and the certified invoice. The activity also applies any outstanding debt, such as a credit, for monies owed to a federal, commercial, and DoD entity against open invoices. The activity calculates interest and discounts in accordance with the contract and the Prompt Payment Act and generates a Certified Business Partner Pay file with the applicable banking information, which is sent to a disbursing activity.

- The Operational Activity label shall begin with a RETURN so that the label does not touch the upper border of the Operational Activity Box (required for SA text formatting).
- The Operational Activity box label must fall within the Activity box border when printed.
- The Operational Activity box border shall be a solid black line.
- The Operational Activity box numbers must be sequential. The Operational Activity Box numbers shall be positioned in the lower right corner of the Operational Activity box.
- Operational Activities are mapped to one or more CBMs that have a stake in that Operational Activity.
- Leaf-level Operational Activities will be associated with a BRM Sub-function(s) where appropriate.
- All Operational Activity modeling decompositions must follow the "2 to 9 Activity box" rule with the exception of the top-level A-0.

5.3.2.2 ICOMs

- All ICOMs shall be defined. All ICOM definitions shall be consistent with the level of decomposition of the Activity. The definition of the Parent ICOM shall include the list of child ICOMs.
- An ICOM Arrow cannot connect to the same Operational Activity more than once.
- Definitions shall be complete enough to support linkage of ICOMs to Entities and Attributes in the DIV-1/ DIV-2 through IEs. If the ICOM supports IEs at the Attribute level then its definition needs to be narrow enough to enable the IE to contain a finite set of Attributes supporting specific Data Enterprise Standards. Each Input and Output shall have a one-for-one relationship with an IE unless there are additional IEs, Data Objects and System Resource Flows (SRFs) associated to the ICOM.

Examples of good ICOM definitions:

> (Input) **Returned Payment Notice**: This is a notification that a previously issued payment has been returned and the reason(s) why. The payment was returned for at least one of the following reasons: invalid account number, invalid routing transcript number, account closed, or rejected by payee.

> (Output) **Wire Transfer Information**: Information that is provided to the U.S. Treasury for the wire transfer transaction. Information could include payee's name, bank account and routing number, amount of transfer and transaction date.

> (Mechanism) **DTS**: The Defense Travel System (DTS) transforms what is currently a paper-

> based, labor-intensive travel process into a fully automated and web-based system that will support official travel. When fully implemented, DTS will be the designated single standard system for temporary duty travel requirements for all DoD personnel.

- For the definition of an internal control, the definition should use control-oriented nouns that indicate elements of control and not performance or consumption.

Example of a good Internal Control definition:

> **ESOH Control Requirement**: This is the required set of operational controls implemented by a mission activity to comply with environment, safety, and occupational health legal, regulatory, policy and performance requirements. The controls take the form of terms and conditions established by agreement between stakeholders such as the owner or operator of the mission activity; supporting environment, safety, and occupational health organizations; permitting agencies; and the public. A documented terms and conditions agreement might be a: process authorization, permit, license, exemption, explosives safety site plan and management decision or operating exception. Environment, safety, and occupational health controls influence the doctrine, organization, training, materiel, leadership and education, personnel and facilities (DOTMLPF) of mission activities. They may also define standard operating procedures, specify safety requirements, detail experience levels and training requirements, prescribe monitoring and reporting requirements, or specify other required activities.

- ICOM names shall be consistent with their assigned Operational Activity box name and definition and names shall be unique within the model.
- ICOMs shall be linked to each CBM that has a stake in or uses that ICOM.
- Internal ICOMs (on the same diagram) shall have two 90-degree curves rather than be a straight line between two Operational Activity Boxes. The internal ICOM labels shall be placed on the horizontal line where it is the most legible.
- The vertical line segments for multiple child ICOMs must align exactly and have a clean connection line to the parent ICOM Arrow.
- If the label of an Input or Output ICOM is too long for the line or interferes with branch points, it shall be wrapped to two lines only.
- An Output of an Operational Activity cannot go into that same Operational Activity as an Input without changing the ICOM name and definition.
- Boundary ICOMs come from (Input, Control) or go to (Output) one boundary location.
- An ICOM Arrow on a diagram that is connected to multiple Operational Activities shall be drawn from a common source ICOM or to a common destination ICOM. The ICOM name shall be displayed once.

5.3.2.3 ICOM Bundling

ICOM bundling refers to creating ICOMs of higher levels of abstraction as parents of a number of more detailed child ICOMs. Grouping similar ICOMs together, either by type or by their relationship with a producing and consuming Operational Activity creates the bundles. Bundling is used to reduce the number of ICOMs on the A-0, A0, or A1 Activity Model diagrams and to keep ICOM detail consistent with that of the Activities at a given diagram level. Bundles represent the more detailed ICOMs shown on the lower-level diagrams, and are derived from other information sources relevant to the information dependency between Activity Boxes on a given OV-5b Operational Activity Model diagram.

The bundling process is done bottom-up. The rule is to form these higher-level ICOMs when multiple ICOMs can be combined into a higher-level abstraction while adhering to the "type of" or "part of" rule. If multiple lower-level ICOMs can be combined into an abstraction for the higher-level diagram, they shall be bundled into a higher-level ICOM, depicted on the lower-level diagram and connected to the appropriate Operational Activity on the higher level diagram. The following modeling conventions apply to ICOMs:

Inputs

- Input ICOM labels shall be left justified above the ICOM Arrow, closest to the boundary.
- Input boundary ICOMs must originate as a horizontal line from the left diagram boundary.
- Child Input ICOM labels should be placed above the horizontal line where most legible, preferably close to the using Operational Activity.
- When a child ICOM is drawn, or decomposed from a parent boundary ICOM, that ICOM and all siblings shall be attached at the same spot on the parent ICOM and connect to their appropriate Operational Activity. The ICOMs shall align vertically with each other.

Outputs

- Output ICOM labels are right justified above the ICOM Arrow and closest to the boundary.
- Output boundary ICOMs must terminate as a horizontal line at the right side of the diagram far enough away from the activity boxes that the labels can be legible. The Output ICOMs must align vertically with each other.
- Child Output ICOM labels should be placed above the horizontal line where most legible, preferably close to the producing Operational Activity.
- When a Child ICOM is drawn and attached to a parent boundary ICOM, said ICOM and all its siblings shall attach to the same spot on the parent ICOM and connect from the appropriate Operational Activity. The ICOMs shall align vertically with each other.

Controls

- Controls shall be bundled into one of the following high-level Controls, which will appear on the A-0 in descending stair-step order from left to right:
    1. Laws, Regulations and Policies
    2. Information Assurance
    3. Data Enterprise Standards

- Control ICOM labels shall be positioned to the right and at the top of the Control ICOM Arrow. Control ICOMs originate as a vertical line above the first Activity Box (top right) on an OV-5b Operational Activity Model Diagram. (An Output that becomes a Control shall be shown starting as a horizontal line coming from the left boundary and then turning down with the arrow terminating at the top of an Activity box.).
- A maximum of 12 Controls are allowed per Operational Activity box.
- Controls are drawn as a stair-step with the tallest Control on the left and the shortest on the right side of the Operational Activity box.

Mechanisms

- Mechanisms shall be assigned from the bottom up and will only be attached to the Operational Activity if they perform all or part of the activity being performed in the activity and not just support the activity by sending information. Mechanisms will be assigned to the leaf or lowest, level Operational Activities first, and will then be balanced upward into parent diagrams. The Mechanism will attach to the parent boundary Mechanism either on the leaf-level diagram or the parent diagram.
- The following high-level Mechanisms are the parents for all lower-level Mechanisms in the OV-5b, and are shown on the A-0 diagram:
    1. Core Business Mission
    2. System and Enterprise Standard
- Each Operational Activity is associated with at least one Mechanism.
- A maximum of 12 Mechanisms are allowed per Operational Activity box.
- Mechanisms shall be arranged in descending stair-step order with the tallest Mechanism on the left. (This is an exception from IDEF0, which calls for Mechanisms to be drawn from the right to left).
- Mechanism ICOM labels shall be positioned to the right and at the bottom of the Mechanism ICOM Arrow. If the Mechanism is decomposed from a parent Mechanism and there are 90-degree turns in the ICOM Arrow, the Mechanism label shall be along the horizontal line closest to the arrowhead.
- Mechanisms shall originate as a vertical line below the first Operational Activity box on the OV-5b Operational Activity Model Diagram. When child Mechanisms are decomposed from the parent, there will be two 90-degree turns in the Mechanism to attach the Mechanism to the appropriate Operational Activity box.

ICOM Balancing

- All ICOMs in the OV-5b Operational Activity Model diagrams should be balanced (that is, if you have an Input to an Operational Activity at a parent-level diagram, then that same Input will appear as a boundary Input on the child diagram). If that Input is to be decomposed, then the child Input(s) will be pulled out at the lower-level model.

- If an Operational Activity Box has a child diagram, each arrow connected to the parent box shall appear on the child diagram.

Information Exchange

- Every leaf-level Operational Activity Input and Output ICOM has at least one associated IE.
- All IEs shall have the same name, definition and CBM tag as the corresponding ICOM unless there are multiple IEs supporting Data Objects and SRFs.
- Each IE must be linked in the SA tool to the corresponding ICOM.
- Mechanism and Control ICOMs shall not be mapped to an IE.

5.3.3 OV-5 Best Practices

This section discusses best practices, including lessons learned from previous architecture development efforts and common modeling pitfalls to avoid.

5.3.3.1 OV-5 Lessons Learned

The following lessons learned have been and serve as the basis for the OV-5 Checklist on Appendix B:

- OV-5a Operational Activity Decomposition Tree must be stabilized before OV-5b model modeling can begin.
- All internal Operational Activities must be tagged with a CBM.
- All internal leaf-level Operational Activities must be associated with a BRM Sub-function.
- All External Operational Activities must be specified and tagged as being external.
- All leaf-level Operational Activities must be specified as such.
- All leaf-level Input and Output ICOMs must be defined and their sources or destinations must be explicitly specified.
- Parent-child ICOM linkages on models must be clear and consistent.
- OV-5 models should be balanced across associated Operational Activity diagrams.
- Standard color coding of diagrams during the workshop is useful for participants to identify where content was added, changed or deleted. Standard color coding should be in line with that used in the BEA Compare reports.
- All exception reports must be reviewed and resolved.

5.3.3.2 OV-5 Common Pitfalls

The following are common mistakes made to the OV-5 models in SA that must be avoided. Checking for these mistakes is included in the OV-5 Checklist on Appendix B.

- ICOM Arrows cross each other unnecessarily.
- ICOM Arrows not touching Operational Activity Boxes.
- Ineffective use of diagram space.
    - Activity boxes too large or too small.

    - ICOM connections unclear.
    - Diagram overly dense or too spread out.
- Inappropriate color coding of diagram objects.
- Incorrect bundling of ICOMs on diagram.
- Activities that do not match the level of decomposition of their ICOMs.
- Truncated text on ICOMs.
- Operational Activity diagrams description not properly defined.
- Operational Activity and ICOM definitions do not conform to guidance.
- Incorrect use of acronyms.
- An Output produced by a leaf level Operational Activity is also produced by another leaf Operational Activity

6 OV-6a – Operational Rules Model

6.1 Summary Description

This section describes the OV-6a Operational Rules Model, its relationship to other BEA Models, the development method, and the modeling guidelines to be followed. Model Purpose

The OV-6a specifies operational or business rules that are constraints on an enterprise, a mission, operation, business, or architecture. For example, while other OV models (e.g., OV-1, OV-2, and OV-5) describe the structure of a business—what the business can do—for the most part, they do not describe what the business ***must*** do, or what it ***cannot*** do. [DoDAF V2.0]

The OV-6a Operational Rules Model is the set of operational rules that constrain an Enterprise, mission, operation, business, or architecture. For the BEA, the operational rules are called "Business Rules." Business Rules are required in the BEA to fulfill the DoD business mission; describing what the business must do and cannot do. Business Rules define or constrain some aspect of the business.

The BEA is evolving to a Semantic BEA where the BEA is described in an ontology using a common language and modeling notation. World Wide Web Consortium open standards such as Web Ontology Language (OWL), Resource Description Framework (RDF) and Rule Interchange Format (RIF) are being used to describe the BEA. The RIF standard allows compound business rules to be expressed.

The BEA distinguishes between a Business Policy/Guidance and Business Rules. The distinctions are based on the following definitions of a Business Policy/Guidance and Business Rules as derived from *Object Management Group's Semantics of Business Vocabulary and Business Rules v1.0*.

- Business Policy/Guidance is different from Business Rules in that, Business Policy/Guidance is:
  - Less structured
  - Less discrete
  - Less carefully expressed in terms of a standard vocabulary
  - Not directly enforceable
- Business Rules are different from Business Policy/Guidance in that, a Business Rule:
  - Uses structured language and notation
  - Is unambiguous
  - Can be compound or atomic
  - Uses a standard vocabulary
  - Is directly enforceable

The BEA OV-6a, Operational Rules include a concept of "Business Rule Category" to distinguish between two categories of Business Rules; Operative Business Rules and Structural Business Rules. Operative Business Rules are those that constrain the dynamic aspects of an enterprise. Structural Business Rules are those that constrain the static aspects of an enterprise. The Operative Category of a Business Rule has a Type of either Derivation or Action, both of which can potentially be violated or ignored. The Structural Category of a Business Rule has a Type of Relational or Definitional, both of which are constrains that must be imposed. Examples are presented in the following sections.

There is a difference between the definition of a Business Rule and notation in which the Business Rule is written. This is similar to the difference between the definition of a Process Model and BPMN. Business Rule notation is a technique used to write Business Rules in a consistent way such that the Business Rule is unambiguous, uses a standard vocabulary, and is directly enforceable. While the APG specifies *RuleSpeak*™ as the Business Rule notation, there are earlier written Business Rules that do not conform to *RuleSpeak*™ notation.

A business statement that does not follow *Object Management Group's Semantics of Business Vocabulary and Business Rules v1.0* conventions or *RuleSpeak*™ notation can be entered as not being "APG Compliant". This allows the BEA to capture a business requirement that may be rewritten as an APG compliant business rule in subsequent BEA releases.

The development of Business Rules is an art rather than a science. It requires close collaboration between the Stakeholder SMEs and the BEA Development Team to create Business Rules that provide clear unambiguous guidance. There are few fixed guidelines that can be applied. One of the sources used as a guideline is the methodology developed by Ronald G. Ross and Gladys S.W. Lam, internationally acclaimed experts of Business Rule techniques and methodology. This methodology is documented in Ross's book *Business Rule Concepts: Getting to the Point of Knowledge* (2nd ed. Business Rule Solutions, LLC, 2005).

6.1.1 Model Structure

The OV-6a is a textual model in the form of definitions. The OV-6a is manually created. Refer to Section 6.1.3 for the required fields for the Business Rules to make the definition complete. Each Stakeholder can have an unlimited number of Business Rules within the BEA SA encyclopedia.

6.1.2 Relationship to Other BEA Models

OV-6a is related to other BEA models as follows:

AV-1	The scope of the development effort for each CBM for a development cycle, as disclosed in the AV-1, will determine if the OV-6a models are affected in the release.
AV-2	All OV-6a terms with specific meaning must be included in the AV-2. These terms must include, as a minimum, all object types included in the deliverable. All acronyms used in OV-6a descriptions must be listed and spelled out in the AV-2 Acronyms Definitions list.
OV-6c	Operative Action Business Rules are only mapped to the leaf level BPM Processes in the OV-6c Event Trace Diagram. Operative Derivation Business Rules are mapped to, and help to define, either BPM Processes, and/or Data Objects in the OV-6c Event Trace Diagram. Structural Business Rules are mapped to, and help to define, Data Objects in the OV-6c Event Trace Diagram.

DIV-1 **DIV-2** **DIV-3**	Structural Business Rules constrain the structure and validity of Entities and Data Elements captured in the DIV Model. The structure of the Entities, Attributes, and Relationships must be consistent with the Business Rules. **Note:** Currently, in the BEA, Business Rules do not map directly to the Entity Relationships. Instead, Structural Definitional Business Rules are directly mapped to specific Attributes or their underlying Data Elements, Data Structures supporting Enterprise Standards. In this case, the Enterprise Standard must be associated to the data element or data structure supporting the Attributes used for BEA compliance.

6.1.3 OV-6a Model Definitions

6.1.3.1 OV-6a Mapping Decision Tree

The current version of the BEA does not include DoDAF SV-10a System Rules. The primary difference between the OV-6a and the SV-10a is that the OV-6a Business Rules denote "logical" constraints while the SV-10a System Rules denote "physical" constraints; thus, the BEA models to which they are mapped are different.

Figure 6-1, OV-6a BEA Model Mapping Decision Tree, is a decision tree that guides the mapping of OV-6a Business Rules to the appropriate BEA model. For completeness and consistency, the decision tree includes the logic for both OV-6a and SV-10a mappings in anticipation for future inclusion of the SV-10a System Rules into the BEA. A complete decision tree helps in deciding the correct placement of the Business Rule.

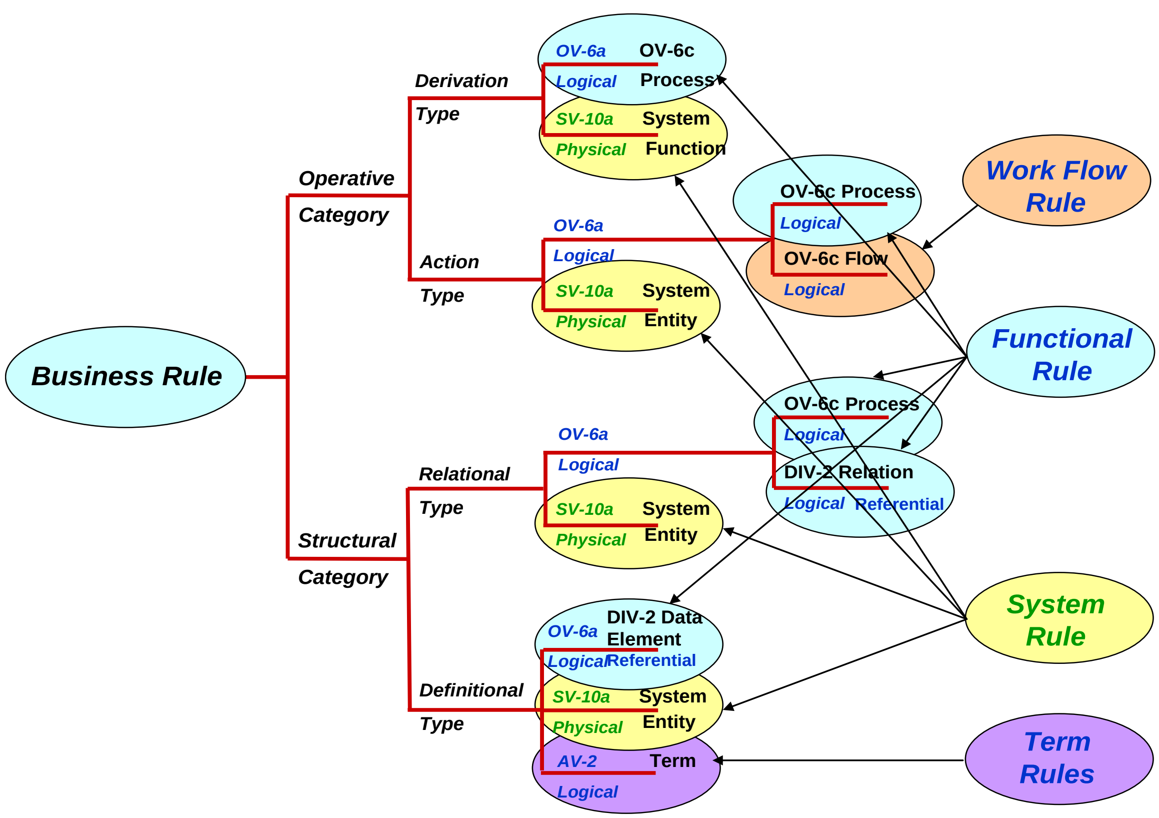


Figure 6-1, OV-6a BEA Model Mapping Decision Tree

The OV-6a mappings are indicated by the color blue to differentiate them from the SV-10a mappings in the color yellow. There are two groups of OV-6a mappings, Business Rules and Workflow Rules. Business Rules are constraints imposed by the business activities, while Workflow Rules are constraints that sequence business processes. Term Rules are the definitions of the terms used by the rules. The above decision tree is based on the Business Rule Mantra: "***Rules are built on Facts. Facts are built on Terms.***"

6.1.3.2 OV-6a Field Definitions

This section defines concepts and terms often used when discussing Business Rules. DoDAF terms are not repeated. Refer to DoDAF, Volume 2, Section 4.6, for the OV-6a Operational Model terms.

Figure 6-2, Data Fields and Structure of a Business Rule, shows the structure of a Business Rule and lists the eleven (11) associated data fields. As described in further detail below, five of the data fields are text entry, two of are single list selections, and the remaining four are multiple value list selections.

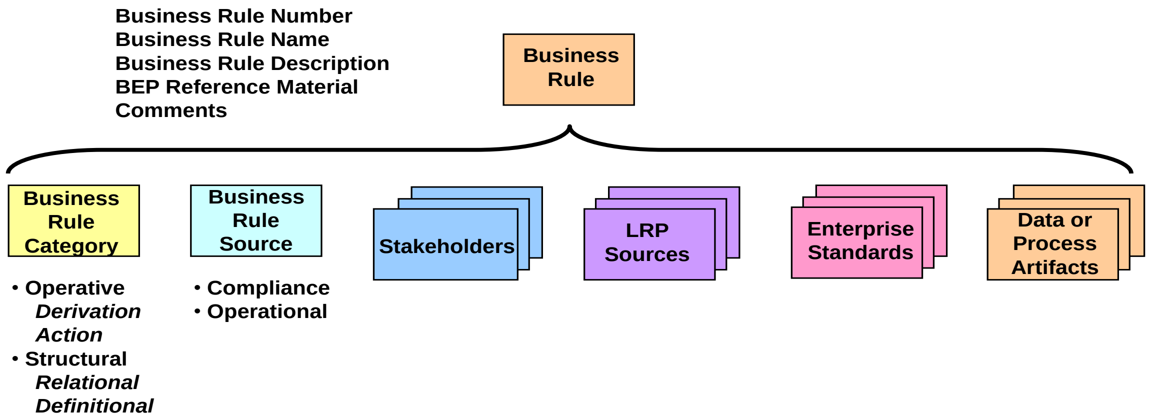


Figure 6-2, Data Fields and Structure of a Business Rule

5. Business Rule Number: A mandatory text data field for the unique number manually assigned to a Business Rule for identification purposes. The following are examples of correct Business Rule Numbers:

   - 50048
   - 50049

The Business Rule Number should be sequentially assigned within the BEA Release number as shown above. That is, the BEA Release is "5.0" and the sequential numbers are "48" and "49". This assignment technique allows all of the new Business Rules for a BEA Release to be easily identified.

6. **Business Rule Name**: A text entry field for the unique name assigned to a Business Rule for identification purposes. The Business Rule Name is limited to 80 characters and consists of concatenated terms separated by underscores (_). A term can be either alphabetic or numeric. The Business Rule Name must begin with an alphabetic term. Initial uppercase is used for each alphanumeric term, unless it is an acronym. The Business Rule Name is used to map to other architecture artifacts such as a BPM Process. An example of a Business Rule Name is:

"Payment_Request_Approval_1"

7. APG Compliant: A checkbox entry that denotes whether or not the statement contained in the Description field is compliant to the Object Management Group's Semantics of Business Vocabulary and Business Rules v1.0 conventions and to the RuleSpeak™ notation. If this field is entered, then the Description field is a business rule that is "APG Compliant". If the field is not entered, then the Description field is not an "APG Compliant" business rule and must be rewritten to be in compliance in the next BEA Release. All of the following data fields must be appropriately entered regardless of the setting of this checkbox entry. Note: all of the Business Rules prior to BEA Release 8.0 are denoted as "APG Compliant".

8. **Description**: A text entry field for the Business Rule set forth as a statement of constraint or permission with respect to a process and/or data. It may be stated at any level but should be stated at an appropriate level of detail consistent with the process level. Examples of Business Rule Descriptions are given in a following Business Rule Category Section.

9. **CBM Reference Material:** An optional text entry field in which to store relevant Business Rule source information. This is the recommended field to store Derivation sources. This field usually contains hyper-links to the referenced source material.

10. **Comments:** An optional text entry field for use of Stakeholders. For example, the stakeholder may decide to list related Operational Activities here

11. **Business Rule Category**: A single list selection field for defined Business Rule categories to assist in mapping the Business Rule to the OV-6c Event Trace Diagram and the DIV Models. Listed categories include:

    - **Operative Derivation**: A Business Rule that reflects computational or derived aspects of the mission or business and specifies constraints of a BPM Process or Sequence Flow in the OV-6c Event Trace Diagram. It must have a mapping to a BPM Process or Flow. The following is an example of a correct Operative Derivation Business Rule:

– **Operative Derivation Business Rule Description:** The ***Fee Amount*** must be calculated by multiplying the ***Fee Rate*** by the number of ***Days Late***.

Note: The underscored terms in the above description must be added as Terms in the AV-2.

- Operative **Action:** A Business Rule that reflects dynamic aspects of the mission or business and specifies constraints of a BPM Process or Sequence

Flow in the OV-6c Event Trace Diagram. It must have a mapping to a BPM Process or Flow. The following is an example of a correct Operative Action Business Rule:

−	**Operative Action Business Rule Description:**
−	A ***Library Card*** must be used to check out a ***Book***.

Note: The underscored terms in the above description must be added as Terms in the AV-2.

- **Structural Relational:** A Business Rule that reflects static aspects of the mission or business terms and facts. Structural Assertion Business Rules may result from important structural assertions graphically represented in the DIV models due to BEA Compliance Assessment requirements. It must have a mapping to the BPM Process which enforces the rule. This rule could have a mapping to the DIV models. The following is an example of a correct Structural Relation Business Rule:

−	**Structural Relational Business Rule Description:**
−	(BPM Process) A Contract must have one or more Line Items.

Note: The underscored item in the above description must be defined as Terms in the AV-2.

−	**Structural Relational Business Rule Description:** (DIV Relation)
−	A ***CONTRACT*** must have one or more ***LINE-ITEMS***.

Note: The underscored item in the above description must be defined as Entities with a Relationship in a DIV Model.

- **Structural Definitional:** A Business Rule that reflects static aspects of the mission or business terms and facts. Structural Definitional Business Rules may result from important structural definitions graphically represented in a DIV models due to BEA Compliance Assessment requirements. It must have a mapping to a Data Element. The following is an example of a correct Structural Definitional Business Rule:

−	**Structural Definitional Business Rule Description:** (Data Element)
−	A ***Zip_Code*** must have exactly nine numeric characters.

Note: The underscored item in the above description must be defined as a Data Element.

−	**Structural Definitional Business Rule Description:** (AV-2 Term)

–	A ***Contract*** is an agreement between two or more parties whereby
–	each party promises to do, or not to do, something.

Note: The underscored item in the above description must be defined as a Term in the AV-2.

12. **Business Rule Source:** A single value list selection field that indicates the source of the Business Rule.

- **LRP:** The origin of this type of a Business Rule is a Law, Regulation, and/or Policy applicable to a particular business process and/or system.

Note: When the Business Rule Source is "LRP", the Business Rule must be mapped using a valid LRP Source Identifier.

- **Reference Material:** The origin of this type of a Business Rule is the result of business transformational efforts and the need to constrain, or give permission to a System Function or System Entity. This type is not directly traceable to compliance requirements.

Note: When the Business Rule Source is "CBM Reference Material", the "CBM Reference Material" data field must be populated with the appropriate reference citation.

13. **Stakeholder:** A multiple value list selection field to identify the owner of the Business Rule. The owner may be an individual CBM designation.

14. **Enterprise Standards:** A multiple value list selection field that designates the Enterprise Standards that the Business Rule enforces or identifies the enterprise system(s) that the Business Rule influences. Business Rules mapped directly to specific DIV objects must have the enterprise system or Enterprise Standard they support identified here.

The following is an example of an Enterprise Standard:

SFIS: Standard Financial Information Structure

15. **LRP Sources**: An optional multiple valued data field that contains a link to the specific Law, Regulation or Policy in the LRP Repository. The LRP Sources are selected from a list of available LRP Sources.

16. **Data or Process Artifacts:** A multiple value list of valid SA data or process artifacts to which Business Rules are related.

6.1.3.3 OV-6a Input Field Capture

This section describes capturing the Business Rules field values for inserts or changes to the System Architect encyclopedia.

6.1.3.3.1 Input Document

The primary input document is a spreadsheet that has the following four tabs:

1. Main Load
2. LRP Sources
3. CBM Stakeholders
4. BPM Processes

The Main Load tab contains all of the above single entry fields as columns. That is, the fields for which there is only one value; such as the Business Rule Definition.

The other three tabs contain the fields for which multiple occurrences are possible; for example, a Business Rule can be mapped to more than one BPM Process. The map between the Business Rule and the other BEA artifact is the Business Rule Name. The following is an example of entries in the BPM Process tab.

Table 6-1, Example of a BPM Process Tab

BPM Process Name	Business Rule Name
Manage Financial Management Policy	Audit_Trails_9
Manage Financial Reporting	Deferred _Maintenance_And_Cleanup_Costs_1
Manage Financial Management Policy	Deferred _Maintenance_And_Cleanup_Costs_1

The LRP Sources and the CBM Stakeholders tabs have the same construct as the above. This input document is also used as a "turn around" document for reviews between the BEA Development Team and the stakeholder.

6.1.3.3.2 System Architect Update Process

There are two basic SA update processes, based on the volume of the update. A large update volume requires assistance from the A&IM Build Team using a spreadsheet provided by the BEA Development Team as input to an automated procedure. A small update volume is manually entered into SA by the BEA Development Team.

6.1.3.3.3 Mapped Field Validation

The mapped field names must be correct, otherwise errors will be generated and the Business Rule will not be complete.

The SA Build Team automated load procedure automatically inserts artifacts as new objects when a match is not found. For example, an incorrect BPM Process name in the load sheet will cause a new BPM Process to be added with that name and may result in incorrect mappings. This is an error that must be manually corrected. Consequently, care must be taken to ensure that the mapped names are valid and already exist in System Architect.

The same applies for LRP Sources; however, this is a benefit because no manual effort is required to update the BEA LRP Source list. The LRP Sources should be verified before the load to ensure that they are valid.

6.2 Developing the OV-6a Model

This section describes the approach used by the BEA Development Team to develop the OV-6a Operational Rules model. The BEA Development Team works with stakeholders and functional SMEs to produce Business Rules that support the business transformation. This process includes development, maintenance and retirement of Business Rules.

6.2.1 Pre-Development Tasks

In general, each stakeholder has their own process to identify Business Rule concepts for inclusion into the OV-6a Operational Rules model. The Stakeholder may request the assistance of the BEA Development Team during pre-development work, such as: providing general guidance on how to develop content; answering questions regarding form and structure; or generating and analyzing System Architect reports from the previous BEA release. This process for Business Rule creation does not require detailed analysis of the architecture. The Stakeholder provides the BEA Development Team their identified Business Rule concepts mapped to an architectural object for pre-analysis.

6.2.2 Development Tasks

6.2.2.1 Refine Business Rules Concept

The following steps refine the Business Rule concept.

- Develop the concept into a form ready for functional review.
- Refine the Business Rule to meet the project's BEA/DoDAF standards.
- Confirm that the proposed Business Rule concept does not duplicate an existing rule.
- Analyze the concept for accuracy and potential conflict with existing Business Rules.
- Determine the Business Rule Category.
- Refine the Business Rule concept by applying the *RuleSpeak*™ guidelines to ensure the proposed language of the Business Rule meets the BEA standards.
- Determine the Business Rule Category Type and Business Rule Source Type.
- Develop the Business Rule Name.
- Assign the OV-6a Operational Rules Model unique Business Rule Numbers.
- Forward the proposed Business Rules to the BEA Development Team for technical review.

6.2.2.2 OV-6a Operational Rules Model Coordination with Stakeholders

Each Business Rule is passed back to the Stakeholders for a functional review after the technical review by the BEA Development Team. The Stakeholders verify that the proposed Business Rule conveys the same idea as the original Business Rule concept. If the proposed Business Rule passes the functional review, the Stakeholders return it to the BEA Development Team for pre-load verification. If the proposed Business Rule does not pass functional review, the Stakeholders return it to the BEA Development Team for further refinement. In that case, the BEA Development Team uses Stakeholder comments to refine the Business

Rule concept. The Stakeholders and BEA Development Team work closely to achieve a consensus on a functionally and technically solid Business Rule.

6.2.2.3 OV-6a Operational Rules Load Preparation

After the validity of the Business Rules has been agreed, the Business Rules are ready for the load into the BEA. The BEA Development Team creates the load sheet, which is an MS Excel spreadsheet that lists the Business Rules and all associated fields discussed in 6.1.4 Definitions. This load sheet is submitted to all of the Stakeholders for approval through the designated BEA Workshops. After the load sheet has been approved, the BEA Development Team submits the load sheet and a copy of the approved CR to the SA Build Team. The BEA Development Team can manually input the approved Business Rules into System Architect. This is normally done when the Business Rule volume is less than twenty Business Rules.

If a Business Rule does not successfully pass the Stakeholders final approval stage, the BEA Development Team works with the Stakeholders towards final approval.

After the Business Rule is loaded into SA, the BEA Development Team verifies that the load sheet is correctly represented in the BEA. This step must wait for the SA Build Team to perform an Encyclopedia update so that the BEA Development Team can conduct the Post-Build Verification on the latest Encyclopedia build.

6.2.2.4 Creating/Modifying the OV-6a Models

To ensure that the Business Rules remain valid, the BEA Development Team follows a maintenance and retirement process. The Stakeholders notify the BEA Development Team that there is a change in the BEA affecting existing Business Rules. The BEA Development Team identifies the Business Rules and makes any necessary adjustments in System Architect. Next, the BEA Development Team analyzes the architecture for potential changes to the existing processes and other linkages in System Architect. Finally, the BEA Development Team validates the above Process for continued support of the business transformation. The following discusses the maintenance process in detail:

6.2.2.4.1 Identify Business Rules

Either the Stakeholders or the BEA Development Team may initiate an analysis to determine if changes are needed. A number of circumstances may trigger a change to some aspect of the Business Rule and/or its artifacts. Below are typical (but not an inclusive list of) triggers:

- Business Objective change
- Process Step change
- IE / Data Object / SRF change
- Requirement change
- LRP Source change

Note: If a LRP Source Identifier or description already mapped to a Business Rule changes in the LRP Repository, the LRP team will notify the BEA Development Team and/or the Stakeholders. The BEA Development Team will manage the change in System Architect.

6.2.2.4.2 Analyze Potential Changes

Once the change is identified, the BEA Development Team conducts analysis to determine any potential impacts on the BEA. The analysis includes impact to other architectural models, existing Business Rules and/or requirements.

6.2.2.4.3 Validate Business Rule Changes

The BEA Development Team performs a technical review. After completing technical review, the Development Team works with the appropriate Stakeholders for functional approval.

6.2.2.4.4 Update System Architect

Upon approval, the BEA Development Team makes the changes in SA or creates a load sheet to present to the Build Team for automated loading.

6.2.3 Post-Development Tasks

As with the Business Rule maintenance process, a number of Events may trigger the retirement process, such as:

- A Business Objective change or elimination
- BPM Process modification or elimination
- LRP Source change or elimination
- The Stakeholders notify the BEA Development Team that a Business Rule needs to be retired. The BEA Development Team, working closely with the Stakeholders, documents the reason for retirement, and then the Business Rule is retired from the SA encyclopedia. Finally, the appropriate validation steps are executed to ensure that the Business Rule was actually retired.

Note: The Stakeholders review each Business Rule identified for retirement, verifying whether it is appropriate to retire the Business Rule for all Processes. If a Business Rule, identified for retirement, still has a valid mapping to another Process Step, the BEA Development Team retains the Business Rule, removing only the link to the retired Process Step.

The following sections discuss the retirement process in more detail:

6.2.3.1 Identify Obsolete Business Rule

The Stakeholders are responsible for identifying a Business Rule that needs to be retired. In addition, if an architecture object deletion or change is the trigger, the Stakeholders identify whether each instance of the Business Rule must be deleted or just the mapping between the Business Rule and the architecture object being changed.

6.2.3.2 Retire Business Rule

The BEA Development Team removes the identified Business Rule from SA. The BEA Development Team also identifies to the LRP team a retired Business Rule that has a LRP Identifier associated with it. If the Business Rule is still valid, and possesses valid mappings to other architecture objects, the BEA Development Team removes only the prescribed mapping.

6.2.3.3 Validate Business Rule Retirement

The BEA Development Team SME conducts the appropriate quality assurance checks. These checks validate the appropriate retirement of a Business Rule and prevent the creation of orphan Processes and Data Elements.

6.3 Modeling the OV-6a Model Using SA

6.3.1 OV-6a Modeling Conventions

Guidelines that assist in the identification and definition of Business Rules are:

- Each Business Rule must have a unique Business Rule Name. The Business Rule Name format is described in Subsection 6.1.4 of this document.
- Each Business Rule must be assigned a unique Business Rule Number by which it can be identified.
- Each Business Rule must have a unique Description. The Description is the actual Business Rule.
- Each Business Rule must have a Stakeholder. The Stakeholder may be an individual CBM designation.
- Each Business Rule must be classified into only one of two Business Rule Categories: Operative or Structural.
- Each Business Rule of the Operative Category must be typed as either Derivation or Action.
- Each Business Rule of the Structural Category must be typed as either Relational or Definitional.
- Each Business Rule must be associated with only one of two Business Rule Sources: "LRP" or "CBM Reference Material".
- Each Business Rule with a Source of "LRP" must be associated with one or more LRP authoritative sources. The LRP Source Identifier is the unique identifier from the LRP Repository.
- Each Business Rule includes an optional Comment field for use of the CBM Stakeholder(s).
- Each Business Rule includes an optional CBM Reference Material field. This field is used to store information the CBM considers relevant for the Business Rule. For instance, the CBM may include the title of a Requirement Document Regulation and/or policy with a Hypertext Markup Language (HTML) link to the original document. The CBM may also include a list of related Operational Activities.
- Each Business Rule may be optionally associated with an integration Stakeholder. The Stakeholder designation must be from the approved Stakeholder acronym list.
- Each Structural Business Rule can have a mapping to an Attribute the underlying Data Structure/Data Element, a Relation in the DIV Models or a BPM Process. However, each Structural Business Rule is currently only mapped to an Attribute or its underlying Data Structure/Data Element.

- Every Operative Business Rule must have a mapping to at least one BPM Process and/or decision Flow in the OV-6c Event Trace Diagram.
- Each Structural Business Rule mapped to any DIV object must be associated with the Enterprise System or Enterprise Standard it supports in Enterprise Standards field.
- Each LRP Source Identifier must be associated with a unique Identifier from the LRP Repository.

6.3.2 Modeling OV-6a Objects

- A clear understanding of the OV-6c Event Trace Diagram and DIV models must come before OV-6a development.
- CBM OV-6a content that affects other CBMs should be socialized before the workshop.
- Prior to the workshop, the fields associated with a Business Rule, like Source Type and OV-6c Event Trace Diagram mapping, should be socialized with the CBM Coordinator and Team Lead to ensure these mappings are captured, in addition to the content of the Business Rule.
- Ensure acronyms are spelled out in AV-2.
- A Business Rule that has a source of a "LRP" must have a LRP Source Identifier associated with the Business Rule.
- A Business Rule must be mapped to an object in the OV-6c Event Trace Diagram or a DIV Model.
- Ensure all exception reports have been reviewed and resolved.

6.3.3 OV-6a Best Practices

This section discusses best practices, including lessons learned from previous architecture development efforts and common modeling pitfalls to avoid.

6.3.3.1 OV-6a Lessons Learned

- A clear understanding of the OV-6c Event Trace Diagram and DIV Models must come before OV-6a development.
- CBM OV-6a content that affects other CBMs should be socialized before the workshop.
- Prior to the workshop, the fields associated with a Business Rule, like Source Type and OV-6c Event Trace Diagram mapping, should be socialized with the CBM Coordinator and Team Lead to ensure these mappings are captured, in addition to the content of the Business Rule.
- Ensure acronyms are spelled out in AV-2.
- A Business Rule that has a source of a "LRP" must have a LRP Source Identifier associated with the Business Rule.
- A Business Rule must be mapped to an object in the OV-6c Event Trace Diagram or a DIV Model.
- Ensure all exception reports have been reviewed and resolved.

- Validate the existence of SA mapped names using spread sheet macros such as VLOOKUP. This helps reduce errors in the load process.

6.3.3.2 OV-6a Common Pitfalls

- Ambiguity.
- Embedding term definitions in Business Rules (Use AV-2 to define terms).
- Over-explaining or adding unrelated information.
- Subject is neither quantified nor singular (e.g., by using the word "each").
- Conditional Business Rules not in the "If…,then…" format.
- The result of a computation is not subject of a Business Rule with the Business Rule Category of Derivation.
- The logical structure of the Business Rule makes does not make sense.
- Contradictory or redundant Business Rule description.

7 DIV – Data and Information Viewpoint

7.1 Summary Description

This section describes the Data and Information Viewpoint which consists of the DIV-1 Conceptual Data Model, DIV-2 Logical Data Model, and DIV-3 Physical Data Model and its relationship to other BEA models, the development method and the modeling guidelines to be followed.

7.1.1 Model Purpose

The DIV Models describe the structure of the BEA data in terms of data types as Entities and their characteristics as Attributes. It provides wide definitions of the Entities and their Attributes and captures BMA structural Business Rules governing the interrelationships between these Entities and their Attributes.

The BEA DIV Model comprises a single integrated data model. It is used to capture the conceptual, logical and physical data and information requirements of the DoD Business Mission Areas. While most of the BEA DIV is presently composed of DIV-2 Logical Data Models, DIV-1 Conceptual Data Models and DIV-3 Physical Data Models are also included. Although the DIV Model in the BEA is a single integrated data model, no consolidated diagram exists. Instead it is depicted as a set of DIV-2 Entities Relationship diagrams. This is due to a SA tool limitation where the BEA Data Model when generated resembles the appearance of a microchip. DIV-3 level of detail has been captured for many releases starting with the SFIS elements using the Data Domains and Domain Permitted Values and has evolved with later Enterprise Standards to include the Table and Column names for entities directly supporting IEs used for BEA Compliance.

The DIV:

- Enables effective management of data resources by providing a single set of consistent data definitions for use within the DoD BMA.
- Captures the Business Rules describing the structure of data needs within the DoD BMA.
- Serves as data reference architecture to support the sharing of data between DoD BMAs, across the DoD Components, Services and Agencies and organizations outside DoD.

7.1.2 Model Structure

A DIV-1 model consists of Entities and Attributes and may have Relationships. A DIV-2 model consists of Entities, Attributes and Relationships. A DIV-3 model consists of Tables, Columns and Constraints. A DIV-3 model is generated from a corresponding DIV-2 model. The details of these models are presented in following subsections.

Error! Reference source not found., is an illustrative example of a DIV-2 Logical Data Model used within the BEA. A DIV-1 Conceptual Data Model can have the same depiction. Each individual DIV Model represents a particular view integrated across the entire DoD BMAs.

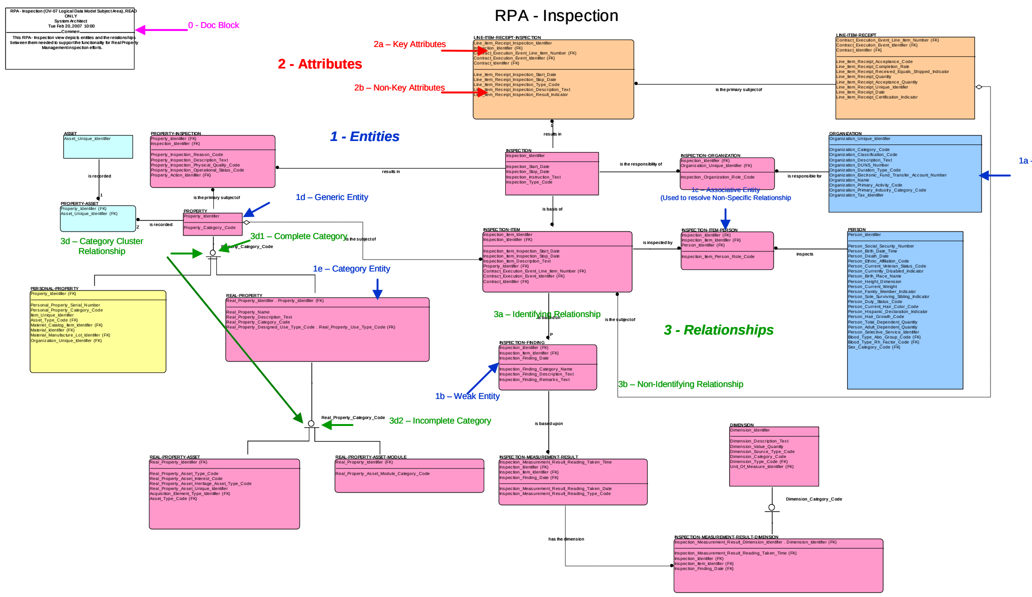


Figure 7-1, Example of a BEA DIV-2 Logical Data Model

Figure 7-2, Example of a BEA DIV-3 Physical Model is an illustrative example of a DIV-3 Physical Data Model that was generated from a DIV-2 Logical Data Model.

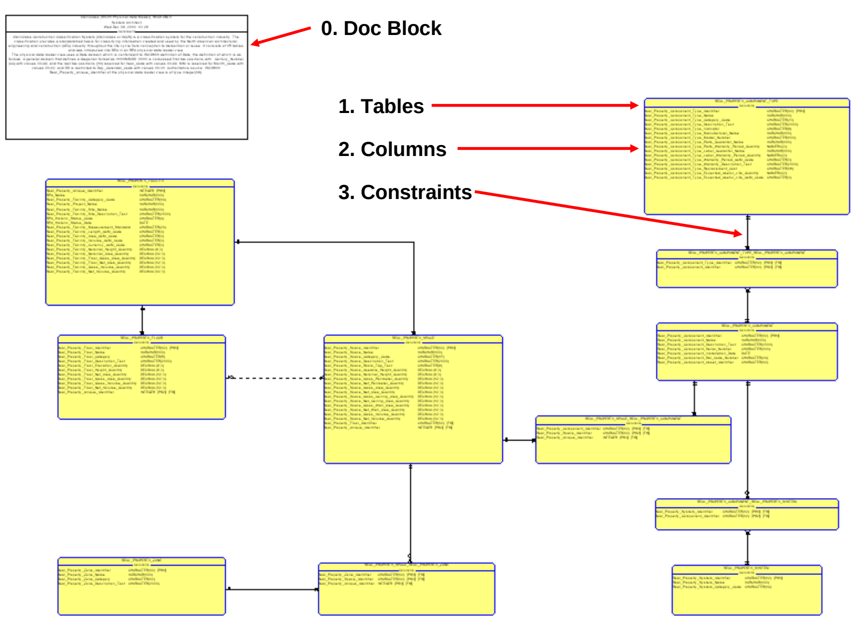


Figure 7-2, Example of a BEA DIV-3 Physical Data Model

7.1.3 Relationship to Other BEA Models

As illustrated in Figure 7-3; the DIV Models are related to other BEA models as follows:

<table>
<tr><td>AV-1</td><td>The scope of the development effort for each CBM for a development cycle, as disclosed in the AV-1, will determine if the DIV Model are affected in the release.</td></tr>
<tr><td>AV-2</td><td>All DIV-1, DIV-2, DIV-3 terms with specific meaning must be defined in the AV-2 Terms. These terms must include, as a minimum, all object types included in the deliverable.

The DIV-2 deliverable objects that must be listed and defined in the AV-2 are:
Attribute Definitions
Data Element Definitions
Data Domain Definitions
Domain Permitted Value Definitions. When the number of permitted values is few, typically three or four, the values are denoted in the AV-2. When the number is great, a link to an authoritative source is provided.
Entity Definitions
All acronyms used in DIV-2 names and descriptions must be listed and spelled out in the AV-2.</td></tr>
<tr><td>OV-3</td><td>One or more DIV Model Entities link to IEs in the OV-3, describing the IEs in terms of the Entities that comprise it. Each Entity in the DIV Models must either link to one or more IEs, or capture the structural assertions If the IE is sufficiently narrow and it supports a Data Enterprise Standard, then it may be populated with Attributes within Entities.</td></tr>
<tr><td>OV-5</td><td>Entities in the DIV-2 support the Inputs and Outputs on the OV-5 via the IEs in the OV-3.</td></tr>
<tr><td>OV-6a</td><td>Business Rules in the OV-6a may constrain the structure and validity of elements of the DIV-2. The structure of the Entities, Attributes and Entity Relationships must be consistent with the Business Rules.

Note: Business Rules in the BEA do not link directly to the DIV-2 Entities, Attributes and Relationships. Instead, Business Rules may be linked directly to specific Data Elements supporting enterprise systems and Enterprise Standards.</td></tr>
</table>

OV-6c	OV-6c Data Objects (including Message Flows and Message Events) inherit their characteristics from the OV-3 IEs as outlined above.
StdV-1	Standards represented in the StdV-1 may be related the Data Domains indirectly related to the DIV Models via the Data Elements and their relationship to DIV Attributes.

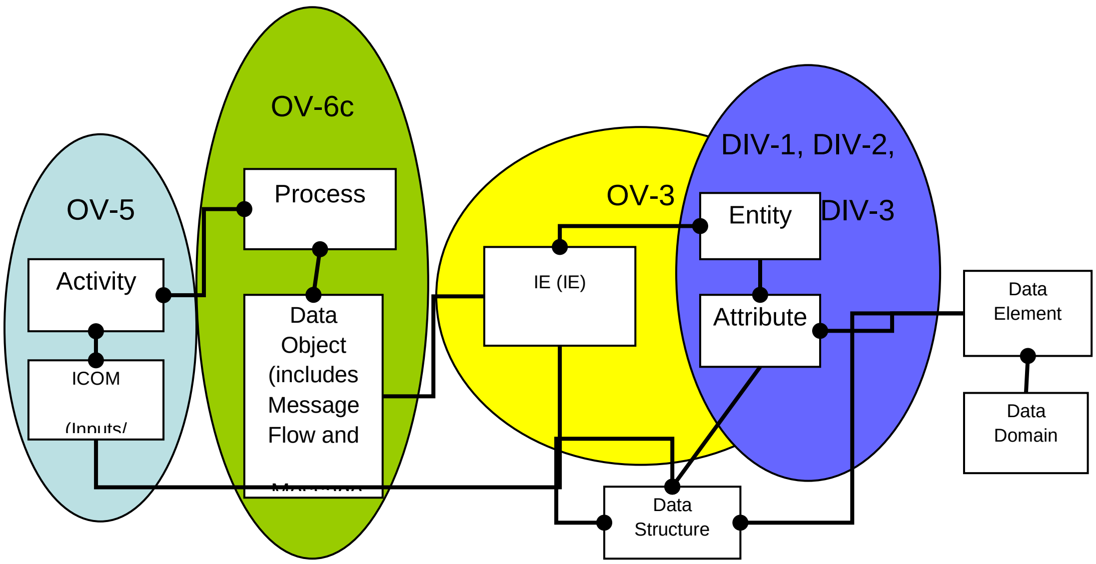


Figure 7-3, Relationship Between DIV Model and Other BEA Models

7.1.4 DIV Model Definitions

The following are definitions of the key elements contained in the DIV Models. The objects used to represent the DIV Model within BEA adhere to the IDEF1x Standard as implemented by the IBM Rational System Architect tool. The following numbered objects refer to **Error! Reference source not found.**. The main features of this diagram are as follows:

- **Doc Block (0)**, also known as a Title Block, contains the diagram name, last modification date and a brief description of the contents of the diagram. It is located in the upper left corner of the diagram.
- **Entities (1),** each refers to a unique person, place, or thing about which the DoD BMA CBMs desire to maintain information. In the context of an Entity, a thing may be either physical or conceptual (an event, a deed, an idea, a notion, a point, etc.). This information captured is on the characteristics of an Entity and/or on relationships between Entities.
- Each Entity represents a set of things having common characteristics and/or relationships to other Entities. The characteristics of each Entity are represented as having a common set of Attributes. The types of Entities are as follows:

- **Independent Entity (1a)**, also known as an Originating, Parent or Generic Entity, is an Entity that does not rely on another Entity for identification.
- **Dependent Entity (1b)**, also known as a Weak or Child Entity, is an Entity that relies on another Entity for identification.
- **Associative Entity (1c),** also known as an Intersection Entity, resolves a Non-Specific (many-to-many) Relationship between Entities.
- **Generic Entity (1d),** also known as a Supertype Entity, it is an abstraction representing the common characteristics in a set of Attributes shared by two or more Category (Subtype) Entities.
- **Category Entity (1e),** also known as a Subtype Entity, represents additional characteristics that differentiate it from the other Category Entities of the same Generic (Supertype) Entity.

- **Attributes (2)**: Characteristics that either identify or describe Entities. Attributes that identify Entities are key Attributes. Attributes that describe an Entity are non-key Attributes. Attributes are associated to one and only one Entity and represent the normalized view of Data Elements within DIV-2. DIV- 1 Entities may have Attributes as required to aid in the refinement of concepts.
  - **Key Attributes (2a):** Attributes that are used to identify Entities as well as describe Entities. Key Attributes uniquely identify an instance of an Entity.
  - **Non-Key Attributes (2b):** Attributes that are used to describe Entities.
- **Relationships (3):** In IDEF1X an Entity Relationship is simply an association or connection between two Entities. A Relationship instance is the meaningful association or connection between two Entity instances. For each Entity instance at one end, the Relationship shows the minimum and maximum number of instances possible for the Entity at the other end. Optionality describes the minimum and Cardinality describes the maximum. The types of Relationships are as follows:
  - **Identifying Relationship (3a):** A Relationship between two Entities in which the Dependent Entity is partially identified through its association with another Entity.
  - **Non-Identifying Relationship (3b):** A Relationship between two Entities in which the Attributes carried to the receiving Entity are used to describe the receiving Entity.
  - **Non-Specific Relationship (3c):** A many-to-many Relationship between two Entities.
  - **Note**: Non-Specific Relationships are only allowed on a DIV-1 and are not allowed in the BEA DIV-2. Therefore all Non-Specific Relationships must be resolved with an associative Entity (1c) before it may be published on a DIV-2 diagram.
  - **Category Cluster Relationship (3d):** (Supertype – Subtype Relationships) is used to express a set of one or more mutually exclusive categorization Relationships for the same Generic Entity.
    - **Complete Category (3d1):** Denotes that all possible categories are represented in the BEA Logical Data Model.
    - **Incomplete Category (3d2):** Denotes that all possible categories are not represented in the BEA Logical Data Model.

- **Data Element**: A Data Element is the smallest unit of stored data, which means that it cannot be broken down further, or that it makes no sense to break it down further. The data element, however, can inherit properties from a data domain. Data Elements are unique across the BMA and are associated with one or more Attributes within the BEA Entities.
- **Domain**: A data domain refers to all the unique values which a data element may contain.
- **Data Structure**: A group of data elements which are in some identifiable relationship with one another.

7.2 Developing the DIV Model

The BEA Development Team works with stakeholders to identify the BEA data requirements. The team then captures these data requirements and structural Business Rules within the DIV Models. The team participates in collaborative working sessions with stakeholders to support the CBM needs and ensures proper integration with other BEA models.

The following is a list of required background material each modeler must understand to ensure a proper DIV Model development:

- OV-2 Data and Information Resource Flows are indirectly represented in the DIV model.
- The OV-5b Inputs and Outputs inherit their DIV characteristics from the OV-3 IEs.
- The OV-6c Data Objects inherit their DIV characteristics from the OV-3 IE.
- The SV-6 SRFs inherit their DIV characteristics from the OV-3 IEs.
- An IE contains a finite set of characteristics from the DIV Model.
- Different sets of data characteristics require the creation of unique IEs.
- Each IE relates to one or more Entities in the DIV Model.
- Attribute level OV-3 IEs are the publication vehicle of Enterprise Standards to achieve enterprise interoperability via BEA compliance. The contents of these IEs may represent a partial implementation in support of an emerging Enterprise Standard.
- DIV-1 Conceptual Data Model is used for concept refinement and stakeholder comprehension of the Data and Information requirements. The Entities and Attributes depicted in these models are refined into a DIV-2 Logical Data Model where Entities and their Attributes are restructured according the rules of normalization. This is done to capture all of the Data and Information requirements prior to producing physical transactions and database schema documented in the DIV-3 Physical Data Model. This process is reversible starting from either the conceptual or physical as long as the integrity of the model is maintained as it is forward and reverse engineered.
- In addition to presenting data and information for concept refinement, BEA depicts IEs used for BEA compliance having a published XML transaction as a DIV-1 Conceptual Data Model. This prevents the migration of key attributes from distorting the data and information requirements which would occur in a DIV-2 using the Entity Relationship Diagram (currently employed in BEA).

Future releases of BEA may use Class and/or Object diagrams to capture the IE requirements when a published XML transaction exists.

- Attribute level OV-3 IEs may contain Data Structures related to their Attributes.
  - A Data Structure is a group of related data elements.
  - If stakeholder provided Data Elements adhere to the following constraints - are at the atomic level, conform to BEA standards, have agreement among all stakeholders on the name and definition - then the Stakeholder provided Data Element is directly incorporated into the BEA or an existing Data Element is updated to reflect the new standard. The Data Element is modeled as one or more Attribute(s) of one or more DIV Entities.
- Stakeholder contributed Data Models are incorporated into the BEA DIV Model to support the level of decomposition of the supporting OV-5.
  - All concepts gleaned from these models must be normalized into Entities and their supporting Attributes across the entire BEA DIV Model.
  - No information may be represented within the DIV Model that is outside of OV-5 activities, either directly supporting Inputs and Outputs (represented in the OV-3 IEs) or to document required structural assertion as of Enterprise Standards.
- All Data Elements in the BEA are at the atomic level, as known and must be represented as one or more Attributes within DIV Model Entities.
- Data Elements directly related to one or more Data Enterprise Standards or enterprise interoperability standards may have a Data Domain to capture the physical characteristics, authoritative source and may contain Domain Permitted Values along with their descriptions.
- Data Domains directly related to one or more Data Elements supporting a Data Enterprise Standard may have Domain Permitted Values only if the entire set of permitted values is included.

7.2.1 Pre-Development Tasks

Develop a thorough understanding of the scope, context, constraints, objectives and model deliverables of the BIP. The following items are the responsibility of the individual modelers assigned to each BIP:

1. Using the BEA HTML, determine existing content from the OV-2 to the OV-5b Operational Activity Model and identify the leaf-level activities supporting the BIP. (This represents the existing scope of the BIP within the BEA.)
   - Each leaf-level activity has supporting ICOMs. Review the Input and Output ICOMs and their associated IEs in the OV-3.
   - Each OV-3 IE is supported by either one or more Entities or a specific set of Attributes within the DIV Model.
   - In addition, Attributes of Entities directly supporting an OV-3 IE structured for BEA compliance may use Data Structures to capture compound Attributes. This enables BEA to resolve these Attributes down to the atomic level Data Element while capturing their relationship back to the published Enterprise Standard. This also preserves the BMA wide

list of Data Elements by excluding non-atomic level data without the need for Data Synonyms.

2. Review OV-6c Process Steps and their associated OV-6a Business Rules to determine additional constraints on the DIV Model.

3. Review OV-6c Data Objects, Message Flows and Message Events and their associated OV-6a Business Rules to determine additional constraints in the DIV Model.

4. Identify existing/proposed IEs in support of enterprise systems, services, and Enterprise Standards.

5. Identify existing unresolved or deferred BIPs, CRs or Tickets that fall within the scope of the BIP.

6. Identify existing/proposed solutions to the CRs in support of the BIP.

7. Identify existing/proposed OV-6a Structural Assertion and Derivation Business Rules that impact the content of the OV-3 and the DIV Model as are required to achieve the desired outcome of the BIP.

8. Identify existing/proposed IEs in support of SRF in the SV-1 / SV-6 that impact the content of the DIV Model.

9. Identify existing/proposed IEs in support of OV-6c Data Objects that impact the content of the DIV Model.

10. Identify existing/proposed IE definitions that impact the content of the DIV Model.

11. Work within the SMEs to identify proposed revisions to IEs supporting specific Enterprise Standards.

12. Identify existing/proposed operational nodes and need lines required to support the data and information requirements between federation partners.

13. Ensure that only IEs sufficiently narrow in scope (with the correct business context to impact the target audience) are populated at the Attribute level and they directly support Enterprise Standards and their enterprise interoperability objectives.

14. Determine all Entities required to support the BIP. Capture the following kinds of architectural information, Entities: directly supporting the IE when there is an Enterprise Standard transaction set, Entities to capture the structural assertions required to be imposed on the producing systems and services prior to producing the contents for the Enterprise Standard transaction, their derivation Business Rules, and referential integrity constraints necessary for the aggregation of raw data required to produce accurate and consistent data and information.

Note: For emerging Data Enterprise Standards and their enterprise interoperability objectives, break the IE requirements down into atomic level data elements, compare their physical characteristics and underlying

Data Domains, separate them into repeating groups, identify candidate keys and structure them into a set of one or more Entities that precisely capture the contents of the transaction set according to the rules of normalization. Compare the IE and the preliminary contents identified against existing transaction sets in the Meta Data Registry, Government and International consensus standards and determine the authoritative source for each Data Element and propose a standard transaction set to the governance authority.

15. Model an integrated straw-man representation of the BIP DIV that includes all items identified above.

16. Identify any individual Data Elements required for Data Enterprise Standards.

17. Review work within the BEA Development Team to ensure that new Entities, Attributes, and Relationships properly integrate into the BEA DIV model and the other viewpoint models within the BEA.

18. Identify requirements for required HTML and meta-model CRs to ensure the successful implementation of a BIP.

7.2.2 Development Tasks

The development and maintenance of the DIV Model is accomplished offline with individual working sessions with Government SME and impacted stakeholders. Proposed architecture enhancements are pre-briefed with directly impacted stakeholders in closed working sessions. Once a proposed solution is finalized and is acceptable to directly impacted stakeholders; the proposed changes are vetted with all BEA stakeholders in facilitated workshops. These workshops need to include Government SME participation to address content and validate results. The following subsections describe the approach used to develop the DIV Model for the BEA. Each subsection sets forth the specific tasks that must be accomplished in each stage of the development phase. Although most of these steps are sequential, it is common to start some steps before a previous step is completed.

All of these tasks must be completed while developing the DIV Model and are all the responsibility of the individual modelers. Identify data related changes that impact the DIV Model views, Entities, Attributes, Relationships or Data Elements. Work with the CBM representatives on the development and refinement of the OV-2 Need Lines, OV-5b ICOMs, OV-3 IEs, OV-6a Business Rules, OV-6c Data Objects, SV-1 System Interfaces and SV-6 SRFs, and ensure their proper representation within the DIV-1 Conceptual Data Model and DIV-2 Logical Data Model.

7.2.2.1 Creating/Modifying the DIV Model

1. Select impacted diagrams or create new diagram(s) as required.

2. Model new/revised Entities, Attributes and Relationships to capture the SME's functional requirements and meet the objectives of the BIP.

    - Determine the level of abstraction required to successfully meet the objectives of the BIP in the context of each IE.

- Treat conceptual requirements with Entity level IEs modeled at the highest level of abstraction available to capture the high-level or fundamental referential integrity constraints.
- Use existing high-level Entities such as PERSON, LOCATION and DOCUMENT to cover notional requirements.
- For emerging Enterprise Standards and their enterprise interoperability objectives capture the exact contents of the transaction as approved by the governance body as a set of one or more Entities in the BEA. Populate the Data Domains and create new Data Elements as required to capture the precise physical requirements of each Attribute.
- Model additional Entities, Attributes and Relationships required to accurately aggregate data in the IEs from BEA federation partners.
- Apply Derivation and Structural Assertion Business Rules to both sets of Entities directly and indirectly supporting each IE.
- Test the resulting model with SMEs to uncover requirement gaps and design flaws.
- Adjust the model and make refinements as required.

3 To ensure that there is no duplication of Entities, their supporting Attributes underlying Data Elements (Data Domains and Domain Permitted Values), and Relationships across the single BEA DIV Model (DIV-1, DIV-2, DIV-3), the modeler verifies that Entities:

- Do not duplicate existing Entities within the BEA DIV Model
- Either notionally support the Entity level IEs (serves as a place holder for further concept refinements and interactions with the Data Enterprise Standards and their enterprise interoperability objectives)

OR

- Directly support published content of the Data Enterprise Standards and their enterprise interoperability objectives in Attribute level IEs

OR

- Indirectly support the IEs required Derivation and Structural Assertion Business Rules required to produce and constrain the contents associated with Attribute level IEs

4 Work with SMEs to ensure that only IEs sufficiently narrow in scope are populated at the Attribute level and that these IEs directly support Data Enterprise Standards and their enterprise interoperability objectives.

5 Review Data Domains when identifying new Data Elements (A data element must have a single Data Domain).

6 Model new/revised Data Elements as Attributes of Entities that directly support the published content of the IEs.

7 Model new/revised Data Elements as Attributes of Entities that indirectly support the published content of the IEs (as required to capture Data Derivation and Structural Assertion Business

Rules, used to populate the contents of IEs and constrain the data prior to populating each Enterprise Standard transaction set. Modify assignment of Entities or Entity/Attributes pairs to IEs based on the addition and deletion of Entities and Attributes.

8 Only assign Attribute level information to IEs if they directly support the Data Enterprise Standards or enterprise interoperability standards.

9 Finalize the individual Data Elements by populating the Data Domains supporting Data Enterprise Standards or enterprise interoperability standards required for Systems Certification.

10 Assign Data Enterprise Standards to the Data Elements, Data Structures.

11 Review work with SMEs, BEA Development Team/Technical Lead, Independent Verification and validation (IV&V) and Architecture Verification team to ensure that data models are properly integrated within both DIV Model and the BEA.

12 Integrate the approved work models into the BEA.

13 Validate the depiction of the content within the DIV Model with the functional SMEs.

7.2.2.1.1 Diagram/Model Coordination with Stakeholders

All of these tasks are completed in the development phase for the DIV Model. These tasks are all the responsibility of the individual DIV Data Modelers.

- Integrate the approved work models into the BEA.
- Incorporate additional change modifications that impact the Stakeholders' data representation caused by subsequent work sessions.
- Identify subject areas and fundamental Entities for incorporation into the model from the Stakeholder-approved content.
- Remove all Data Synonyms and Attributes from IE not intended for BEA Compliance.
- Ensure that all IEs have DIV Model Entities or Attributes within Entities that cover the definitions of the leaf-level Input and Output ICOMs for all activities that fall within the scope of the developmental effort. Coordinate with other BEA Development Team members assigned to other BIPs to ensure consistent refinement across all products. Have the refinements validated by the SMEs.
- Ensure that all IE proposed to contain Attributes are sufficiently narrow as to enable their population by a finite set of characteristics and that the resulting set is applied in the proper business context. Work within other teams to make any required adjustments to the IEs as needed.
- Review work with SMEs, BEA Development Team/Technical Lead, IV&V and Architecture Verification team to ensure that data models are properly integrated within both DIV Model and the BEA.

7.2.2.1.2 Diagram/Model Clean-up

- Ensure that the Enterprise Standard and Stakeholder assignments agree with their representation on diagrams.
- Remove invalid and duplicate access paths that cause the display of AK1 designations in the primary key portion of Entities.
- Ensure that all Relationship lines on all DIV Model Diagrams display properly and are not hidden.
- Ensure that the associated tags of all Relationship lines are positioned properly on the diagram.
- Ensure that, at 21% zoom, all Attribute names are displayed on a single line within the Entity.
- Ensure that all Relationship lines are straight, not broken, and that all Relationship lines avoid crossing other Relationship lines whenever possible.
- Ensure that all diagram descriptions, including the text for the diagram Doc Blocks, diagram notes, diagram names, object names and object descriptions are spell checked and none are truncated.
- Ensure that all Entities are properly colored.
- Ensure that required Acronyms appear in the AV-2 and that they are defined as Terms in the AV-2.
- Ensure that words in the "Terms" list of SA are correctly and consistently represented in all object names and descriptions. Ensure that other definitions do not redefine the Terms.
- Ensure that all table names exactly match their physical names.
- Ensure that all the primary index and access path names exactly match the Entity name followed by "_PK" suffix.
- Ensure that the IDEF1X categorization names match the discriminator Attribute names with the removal of their class word and the replacement of the "_" between terms with spaces.
- Ensure that all column names exactly match their physical name.
- Address all items in the DIV Model Checklist (Diagrams, Definitions and Integration).
- Ensure that the CR packet includes all necessary items.
- Remove DIV Model objects from the encyclopedia that are not shown on or associated with DIV Model Diagrams

7.2.3 Post-Development Tasks

1. Incorporate additional updates to the DIV Model based upon approved Tickets.
2. Document known deficiencies to be resolved within the next release.

7.3 Modeling the DIV Model Using SA

7.3.1 DIV Modeling Conventions

The following modeling conventions shall be used to create the DIV Model. The BEA DIV-1 and the DIV-2 Logical Data Model development is in accordance with the standardized modeling techniques delineated in IEEE Std 1320.2-1998 as implemented within the confines of the SA tool.

7.3.1.1 Use of Color in DIV-2 Diagrams

Each of the CBM areas has a specific color scheme to be used on the diagrams for the DIV-2s. The colors can be found in the Basic Color Set within SA on the lower row of the color palette. The colors are applied to Entities within DIV-2 Diagrams as per Figure 7-4..

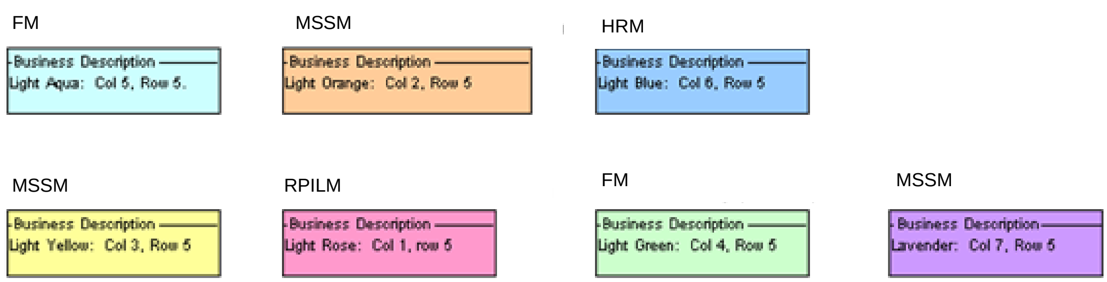


Figure 7-4, DIV-2 CBM Color Set

7.3.1.2 Modeling Guidelines

The Doc Block representing header information for the diagram (including the diagram name and date last updated) is placed on the diagram.

- A Doc Block is placed in the upper left-hand corner of every diagram as close to the corner as the printer margins will permit.
- The dimensions of the Doc Block are adjusted so truncation indicators (dots) are not displayed and all text is visible.
- A black border and no fill color are selected for the Doc Block.
- Borders are not needed for the DIV Model diagrams.
- Logical structures within a diagram are grouped to minimize crossing Relationship lines and to make the diagram more readable and understandable. Absent a specific reason to do otherwise, each view in the DIV Model shall display for each included Entity:
  - The name of the Entity
  - Each primary key Attribute within the Entity
  - Each non-primary key Attribute within the Entity
  - Each Relationship connected to or from the Entity

- The Title of the diagram shall include the following:
    - Be centered on the top of the diagram and in title case (determine center placement by printing single page printable diagrams on 8.5"x11" paper, and folding sheet in half side to side).
    - Not be underlined or bolded.
    - Be in Arial font with appropriate font size so the title is in proportion to all other diagrams when single page printable diagrams are printed on 8.5"x11" paper.
    - Be an exact match of the Diagram Name.
- Include for each included Entity Relationship:
    - The Relationship line and the nature of the Relationship (identified or non-identified).
    - The label (name) for at least one direction of the Relationship.
    - The Cardinality and Optionality of each end of the Relationship.
- When doing so adds value to understanding the particular aspect of the model being presented in the view:
    - One or more of the display characteristics listed above may be left out.
    - An Entity business description may be included.
- Each Entity shall have a black outline.
- Relationship lines shall be in black.

7.3.2 Modeling DIV Model Objects

The following guidelines are used to create or modify the DIV Model.

7.3.2.1 Entities

Each Entity must refer to a unique person, place, thing, or concept within the Enterprise about which the Enterprise desires to and can keep information.

7.3.2.1.1 Entity Names

Each Entity name must:

- Refer to the class of information, not the occurrence of the class.
- Be a singular noun or noun phrase.
- Include only uppercase alphabetic characters (A-Z) with the terms separated with dashes and no special characters (for example, BILLING-STATEMENT).

Or

Title Case with the terms separated with spaces (for example, Confirmed Collection TOP Header).

- Only contain abbreviations in the acronyms list in the AV-2.
- Exactly match the definition from an Enterprise Standard if: the Entity directly supports an IE used for BEA compliance, the Entity represents a specific portion of a mandated XML

transaction, and the Extensible Markup Language (XML) transaction contains a logical name of the object.

- Not contain articles (a, an, the) or prepositions (at, by, for, from, in, of, to).

Exception: The article or preposition is commonly used in Business and clearly aids in identifying the concept behind the Entity.

7.3.2.1.2 Entity Definitions

Each Entity must have a definition. Each Entity definition must:

- Describe the Entity in ordinary business language.
- Define what the Entity is, not how, where, or when the Entity is used, or who uses it.
- The definition should not merely restate or rephrase the name, or just provide a list of the Attributes or meta-Attributes within the Entity.
- Be precise and unambiguous. The exact meaning and interpretation of the defined concept should be apparent from the definition. A definition must be clear enough to allow only one possible interpretation. (Examples may be included to clarify the meaning.) Describe a term in such a way that it has only one meaning within its definition.
- Avoid using any word that appears in the Entity name. Instead, paraphrase or use synonyms whenever possible.
- Not be defined in terms of one Entity that is also defined in the terms of another Entity. (That is, no circular definition.)
- Describe one instance, not a group of instances. For example, begin the definition with "A" or "An."
- Be stated in terms of the thing of interest to the business, not in terms of the information captured about the thing of interest.
- Exactly match the definition from an Enterprise Standard if: the Entity directly supports an IE used for BEA compliance, the Entity represents a specific portion of a mandated XML transaction, and the XML transaction contains a definition of the object.

7.3.2.1.3 Reference Entities

A Reference Entity is one that contains a codified list of standard values as its primary key Attribute (for example, a U.S. State Code table for Virginia and Vermont). To avoid unnecessary visual clutter, a reference table will not be used except:

- When it is necessary to show additional information in the table other than name and description, or
- To provide clarity when the codified list is used in more than one place in the model, or
- Directly supports a Data Enterprise Standard and their enterprise interoperability objectives.

7.3.2.1.4 Entity Primary Keys

An Entity Primary key must:

- Be a natural key (not artificial) whenever possible – a natural key is one composed of Attributes that are natural characteristics of instances of the Entity.
- Be minimal – a key is said to be minimal if the removal of any Attribute would make the key not unique.
- Not have any component that is null.
- Be included on every Entity.
- Be absent of a compelling business reason; no Attribute chosen as a Primary Key should end in a class word other than Code, Date, Identifier, Name, Number, Time or (sometimes) Indicator.
- It must be recognized that common identifiers like Employee ID are surrogate keys. Other cases where a surrogate key may be used include:
  - Cases where there is no possible natural key—for example, a collection of items that creates a group of arbitrary size, but there may be two or more of the same item.
  - Cases where the Entity is an abstract concept—for example, geographic location.
  - Cases where one or more Attributes of a potential natural key could be null.
  - Cases where it directly supports a Data Enterprise Standard or enterprise interoperability standard.

7.3.2.2 Attributes

Each Attribute must describe a characteristic of its Entity. The use of a compound Attribute is not permitted unless the entity matches the definition from an Enterprise Standard, the Entity directly supports an IE used for BEA compliance, the Attribute uses a Data Structure instead of a Data Element and the Data Structure contains the definition of the object. BEA recommends that Compound Attributes are resolved down to the atomic Data Element Level using additional Data Structures as required. Each Attribute must:

- Represent a distinct piece of business information.
- Must be associated with at least one Entity.
- Be functionally dependent on the primary key (DIV-2 only).

7.3.2.2.1 Attribute Names

An Attribute Name must be a business term used to recognize the Attribute.

Each Attribute Name must:

- Be a singular noun or noun phrase; that is, no plurals.
- Use only acronyms or abbreviations as defined in the BEA AV-2.
- Be unique and associated with only one Attribute description.
- Not use names of organizations, computer or information systems, directives, forms, or reports.
- Not use the possessive form of a word (that is, a word that denotes ownership).

- Have all words in each term separated by underscores, no other special characters or spaces, and each word starting with an uppercase letter. For example: Billing_Statement_Identifier.

or

Have all terms in title case with the words in a term separated by spaces, no other special characters. For example: Account Classification Reference Number

- Match exactly to its corresponding Data Element name, with the following exceptions:

  1. Attributes of Entities that directly support an IE used for BEA compliance. In this instance the Entity represents a specific portion of a mandated XML transaction and the Data Element is used to unify multiple attributes in separate transactions sets or is in support of derivation Business Rules.

  or

  2. The Attribute is a Role-based Attribute used to resolve recursive relationships and the Attribute is compatible with the Data Element's Data Domain of the Data Element. This is a SA requirement.

It is recommended that each Attribute adopt the following:

- End in a Class Word represented in Table 7-1, BEA Accepted Class Words

7.3.2.2.2 Attribute Class Words

Table 7-1, BEA Accepted Class Words, illustrates class words and their definitions. A Class Word, which describes the category to which the Attribute belongs (for example, date, identifier, or quantity), must be added to the end of each Attribute's name. Abbreviations for class words may be used when appropriate.

Table 7-1, BEA Accepted Class Words

Class Word	BEA Definition [4]
Amount	A monetary value.
Angle	The rotational measurement between two lines/planes diverging from a common point/line.
Area	The two-dimensional measurement of a surface expressed in unit squares.
Code	A combination of one or more numbers, letters, or special characters substituted for a specific meaning.
Coordinate	One of a set of values that identifies the location of a point.
Date	A particular day of a calendar year.
Dimension	A one-dimensional measured linear distance.
Flag	A binary condition of two mutually exclusive options in a code set same as Indicator.

[4] Department of Defense Data Dictionary System

Class Word	BEA Definition [4]
Identifier	A combination of one or more numbers, letters, or special characters that designates a specific object or Entity occurrence, but has no readily definable meaning.
Image	The two-dimensional optical counterpart of an object produced by an optical device (as a lens or mirror) or an electronic device.
Indicator	A binary condition of two mutually exclusive options in a code set.
Mass	The measure of inertia of a body.
Name	A designation of an object expressed in a word or phrase.
Number	A series of symbols, letters, or numbers used to represent a reference or identification. This is basically the same as an Identifier. It is used when number is the natural, expected, or commonly used terminating word (for example, Social_Security_Number or Disbursing_Voucher_Number).
Quantity	A non-monetary numeric value. This Class Word should not be used if another more restrictive Class Word is more appropriate (for example, Rate, Volume, Weight, or Dimension).
Rate	A quantitative expression that represents the numeric relationship between two measurable units.
Temperature	The measure of heat in an object.
Text	An unformatted character string generally in the form of words, numbers, blanks and special characters. Formatting codes can be embedded in the character string.
Time	A chronological point within a day.
Volume	A measurement of space occupied by a three-dimensional figure.
Weight	The force with which an object is attracted toward the earth and/or other celestial body by gravitation.
Year	A particular calendar year.

7.3.2.2.3 Attribute Definitions

Each Attribute must have a definition. Each Attribute definition must:

- Be concise, brief and comprehensive.
- Be precise and unambiguous. The exact meaning and interpretation of the defined concept should be apparent from the definition. A definition should be clear enough to allow only one possible interpretation. (Examples may be included to clarify the meaning.)
- Avoid using any word that appears in the Attribute name. Instead, paraphrase or use synonyms whenever possible.
- Describe a singular instance, not a group of instances; thus, the definition begins with "A" or "An."
- Explain the Attribute in terms of one value, not several values (singular form).
- Not be defined in terms of one Attribute that is also defined in the terms of another Attribute. (No circular definitions.)
- Start with what the data is, not how, where, or when the Attribute is used, or who uses the data. Subsequent parts of the definition can optionally contain the business reason that the Attribute is important to the organization.

- Use ordinary business language. Where it helps communicate the nature of the Attribute, list a few typical values.
- Use a noun phrase for the first sentence that states the essence of the Attribute. Standard English grammar, including the use of subject and verb, is appropriate for the rest of the definition.
- Exactly match the definition from an Enterprise Standard if: the Entity directly supports an IE used for BEA compliance, the Entity represents a specific portion of a mandated XML transaction, and the XML transaction contains a definition of the object.

7.3.2.3 Data Element

In the BEA, a Data Element is the smallest unit of stored data, which means it cannot be broken down further, or that it makes no sense to break it down further. The Data Element, however, can inherit properties from a Data Domain. Data Elements are unique across the BMA and are associated with one or more Attributes within the BEA Entities.

- Data Elements must be linked to no more than one non-foreign key Attribute within each Entity.
- Each Data Element must represent a characteristic of a concept that is unique across the Enterprise.
- Only one Data Element may exist for a given data concept.
- A Data Element may be linked to a LRP Source.

7.3.2.3.1 Data Element Name

The name of each Data Element must:

- Have all terms separated by underscores, no other special characters or blanks spaces, and each term start with an uppercase letter. For example: Billing_Statement_Identifier.

Or

Have all terms in Title Case with the terms separated with spaces.

- Be unique across the enterprise (no synonyms are allowed).
- Consist of a singular noun or noun phrase.
- Contain characters A-Z (no special characters are permitted).
- Not contain abbreviations or acronyms, unless they appear in the approved acronym list in the AV-2.
- Represent the Business Term used.
- It is recommended that each Data Element adopt the following:
    - Always end with a class word represented in Table 7-1, BEA Accepted Class Words.
    - Use the following format: Prime Word (logical grouping/category), Class Qualifier (Optional), Class Word.

7.3.2.3.2 Data Element Definition

Each Data Element must have a definition. Each Definition must:

- Be concise, brief and comprehensive.
- Be precise and unambiguous. The exact meaning and interpretation of the defined concept should be apparent from the definition. A definition should be clear enough to allow only one possible interpretation.
- Describe a single instance, not a group of instances; thus, the definition begins with "A" or "An."
- Explain the Data Element in terms of one value, not several values (singular form).
- Not be defined in terms of another Data Element. (No circular definitions.)
- Start with what the data is, not how, where, or when the Data Element is used, or who uses the data. Subsequent parts of the definition can optionally contain the business reason that the Data Element is important.
- Use ordinary business language. Where it helps communicate the nature of the Data Element, list a few typical values.
- Use a noun phrase for the first sentence; it must state the essence of the Data Element. Standard English grammar, including the use of subject and verb, is appropriate for the rest of the definition.

7.3.2.4 Data Domain

In the BEA, the Data Domain represents a named and defined set of permitted values from which one or more Data Elements draw their values. A Data Domain is associated with Attributes through Data Elements. There are two kinds of Data Domains:

- **Specific Domain**: The precise set of possible values for a Data Element. Specific Domains may have Domain Permitted Values attached that reflect the entire set of values is available for publication.
- **General Domain**: A specified range of values a Data Element is permitted to have. In general, these domains are too large to be completely enumerated easily. For example: The general domain, *Date(8)* , is defined to contain any date possible, all using the same format (YYYYMMDD).

7.3.2.5 Domain Permitted Values

Domain Permitted Values are the entire set of the possible values with their definitions for a Specific Domain.

7.3.2.6 Entity Relationships

IDEF1X Entity Relationships model certain kinds of Business Rules (structural assertions). Those Business Rules describe the nature of a two-way association between potential instances of two Entities, one found at each end of the Relationship. For each Entity instance at one end, the Relationship shows the minimum and maximum number of instances possible for the Entity at the other end. Optionality describes the minimum and Cardinality describes the maximum. (The term Cardinality can also be used to describe both the minimum and the maximum, but this section of the guidelines uses separate terms as a way to distinguish the two concepts.)

If the Relationship is non-specific (many-to-many) in nature, an associative Entity must be used to resolve the Relationship on all DIV-2 views. If the Relationship is not many-to-many but is optional-to-optional, an associative Entity may be used to resolve the Relationship.

To determine the set of applicable Business Rules for the Relationships in the data model for each Relationship, there are several questions that a data modeler should ask.

- What is the Cardinality of the Relationship (e.g., "one-to-many" and "many-to-many")?
- Is the Relationship mandatory or optional in either or both directions?
- Is the Relationship identifying or non-identifying?

The general approach for managing Relationships can be summarized as follows:

- Each many-to-many Relationship (often referred to as a non-specific Relationship) is resolved by replacing it with an associative Entity. The key for the associative Entity consists of the Attributes that are the primary key for both Entities in the Relationship.
- Each optional-to-optional Relationship that carries data is resolved by replacing it with an associative Entity to carry the data.
- Each DIV Model shall include for each included Entity Relationship.
  - The Relationship line and the nature of the Relationship (identified or non-identified).
  - The Cardinality and Optionality of each end of the Relationship.

7.3.2.6.1 Relationship Label

On DIV-2 Diagrams each Relationship Label must:

- Display the label (name) from the source Entity to the target Entity direction of the Relationship.
- Be a meaningful verb phrase that is assigned to each Relationship line (e.g., "is related to" is not adequate, as the Relationship line obviously infers this Relationship).
- Match the Relationship from the source Entity to the target Entity.
- Be independent of target end's Optionality and Cardinality.
- Exist for each identifying and non-identifying Relationships.
- Be placed on each diagram:
  - As close to the Relationship line and the originating Entity as possible; or
  - Intersecting the Relationship line.
- Be specific, concise and comprehensive.

7.3.2.6.2 Relationship Definitions

Relationship definitions are not required, but may be added for better understanding of the Relationship.

7.3.2.7 Supertypes and Subtypes

A Supertype is an Entity whose instances have Attributes that are common to one or more Entity Subtypes. A Subtype is an Entity that inherits common Attributes or Relationships from an Entity and contains at least one other Attribute or Relationship that distinguishes it from other Subtypes of the same Supertype.

7.3.2.7.1 Subtype and Supertype Definitions

Each Supertype Entity must:

- Be related to at least one Subtype Entity.
- Connect to each of its subtype Entities through an IDEF1X Category cluster circle.

Each Subtype Entity must:

- Have the same Primary Key as its related Supertype.
- Must have either additional Relationships and/or Attributes from the Supertype.
- Be mutually exclusive of the others in the same IDEF1X category cluster.
- Be related to exactly one Supertype Entity.

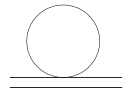
7.3.2.7.2 Subtype and Supertype Naming Convention

- Each IDEF1X Categorization in the Logical Data Model (LDM) shall have a name.
- Each IDEF1X Categorization name consists of a title cased singular noun or noun phrase.
- Each IDEF1X Categorization name may use hyphens between words when using hyphens in proper English construction but no other special characters.
- Each IDEF1X Categorization name uses normal business language.

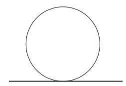
7.3.2.7.3 IDEF1X Categorization Definitions

Each IDEF1X Categorization definition must:

- Be concise, brief and comprehensive.
- Define the scheme that distinguishes among the related subtype Entities. It will define what the scheme is, not how, where, or when the scheme is used or who uses it.
- Indicate the relevant named subtype Entity. If a value is known, that exact value should be used to indicate the correct subtype. Otherwise, the meaning of the value must be stated clearly enough to unambiguously indicate the correct subtype.
- Must have its first sentence constructed as a noun phrase, and subsequent sentences should have normal subjects and verbs.
- Avoid using terms that appear in the IDEF1X Categorization name.
- Use the Complete Category cluster circle (double bar under a circle) if all possible categories have been identified and assigned to subtype Entities in the relevant project model.



- Use the Incomplete Category cluster circle (single bar under a circle) if fewer than all possible categories have been identified and assigned to subtype Entities in the relevant project model.



- Be assigned a discriminator from among the Attributes in the related Supertype Entity. The discriminator must have determinable values, each mapping to a maximum of one subtype Entity related to the categorization.

7.3.2.8 Data Structures

A Data Structure is a set of Data Elements or other Data Structures. A Data Structure is a SA technique to refer to a reusable set of Data Elements by a business or logical name rather than itemizing each Data Element every time the set is referenced. Data Structures permit SA to relate compound to Atomic level Data Elements. A Data Structure can be linked to Business Rules, Enterprise Standards and LRP Sources. All Data Structures must be directly or indirectly related to a compound Attribute in the DIV Model.

7.3.3 DIV Model Best Practices

This section discusses best practices, including lessons learned from previous architecture development efforts and common modeling pitfalls to avoid.

7.3.3.1 DIV Model Lessons Learned

- Monitor changes in the OV-2 resource flows (Need lines) for potential impact within the DIV Model.
- Changes in the OV-5 development (especially Activity and ICOM changes) need to be closely monitored to catch potential impacts on the Entities within the DIV Model.
- Monitor the changes of SV-6 SRFs for a potential impact on the DIV Model models.
- Work with the SMEs to determine the rules that govern the content of the DIV Model and explain the proposed solutions to these same SMEs to validate the model content.
- Comparison results must be reviewed and validated with functional SMEs prior to the completion of the DIV Model workshops.
- Standard color coding of diagrams during the workshop is useful for participants to identify where content was added, changed or deleted. Standard color coding should be in line with that used in the BEA Compare reports.
- All exception reports must be reviewed and resolved.

7.3.3.2 DIV Model Common Pitfalls

- Failure to follow the rules of normalization.
- Using a single Data Element to represent more than one Data Domain.

- Failure to create a set of Entities directly supporting an IE used for BEA Compliance.
- Failure to map Entities and Attributes to all IEs within scope prior to the last workshop.
- Failure to delete Data Synonyms from IEs used for BEA compliance.
- Attempting to map Attributes to ill-conceived or broadly defined IEs.
- Adding Attributes to IEs that do not directly support Data Enterprise Standards and their enterprise interoperability objectives.
- Failure to capture the published contents of IEs directly supporting Data Enterprise Standards.
- Failure to capture the essential Business Rules used to derive and constrain Attribute level contents of IEs.
- Failure to justify child and subtype Entities. Each child Entity must have one Attribute that differentiates it from its parent Entity. Each subtype Entity must have at least one non-key Attribute or at least one Relationship that differentiates the subtype from the supertype.
- Inappropriate color coding of Entities. Entity color must match one of the stakeholder (may require placement of the Entity on additional diagrams within the DIV Model).
- Use of acronyms not appearing in the acronym list and/or using an acronym without checking the Term definition.
- Invalid placement and formatting of DIV Model objects.
    - First, set display to 21% zoom to verify that:
        1. Truncation indicators (dots) are not displayed and all text is visible
        2. All Attributes appear on a single line
        3. All Attributes appear within Entity boxes
        4. All Relationship and categorization labels are properly placed
    - Second, repeat the first step from the single page printable diagrams on a color printer, checking plots at 8.5"x11" and 11"x17". Correct any additional diagram errors uncovered.

7.3 DIV-3 Physical Data Model

7.3.1 Creating the DIV-3

A DIV-3 Physical Data Model (PDM) is generated from a DIV-2 Entity Relationship Diagram (ERD). The ERD is generated from a DIV-2 LDM. The ERD is an intermediate product that is required to generate a PDM from a LDM. Therefore, the DIV-3 generation main steps are:

1. Create the LDM
2. Generate the ERD
3. Generate the PDM

A target DBMS must be selected to generate the PDM from the ERD. The ERD entities are transformed into PDM tables and ERD attributes are transformed into PDM rows of appropriate DBMS data types as derived from the attribute properties.

7.3.2 LDM, ERD and PDM Examples

The following three figures are examples of the LDM, ERD and PDM. The purpose of the three figures is to illustrate the similar structure of the diagrams. That is the same entity in a LDM and ERD, along with the resultant table in the PDM are in the same relative position in the diagrams. This permits a quicker understanding of the diagrams by architects, analysts and Stakeholders. The positioning of the objects in each diagram is a manual effort.

The entities bounded by the red rectangle are enlarged in subsequent figures to allow the contents to be reviewed.

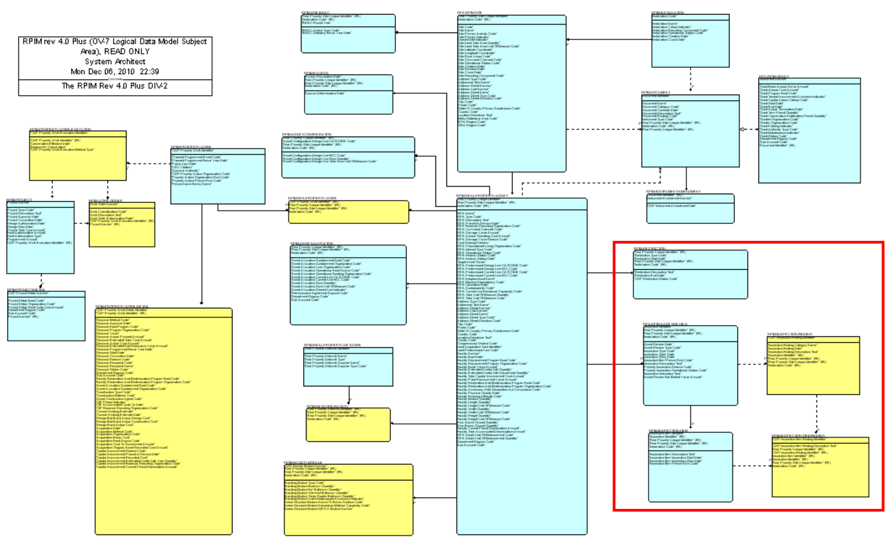


Figure 7-5, DIV-2 Logical Data Model

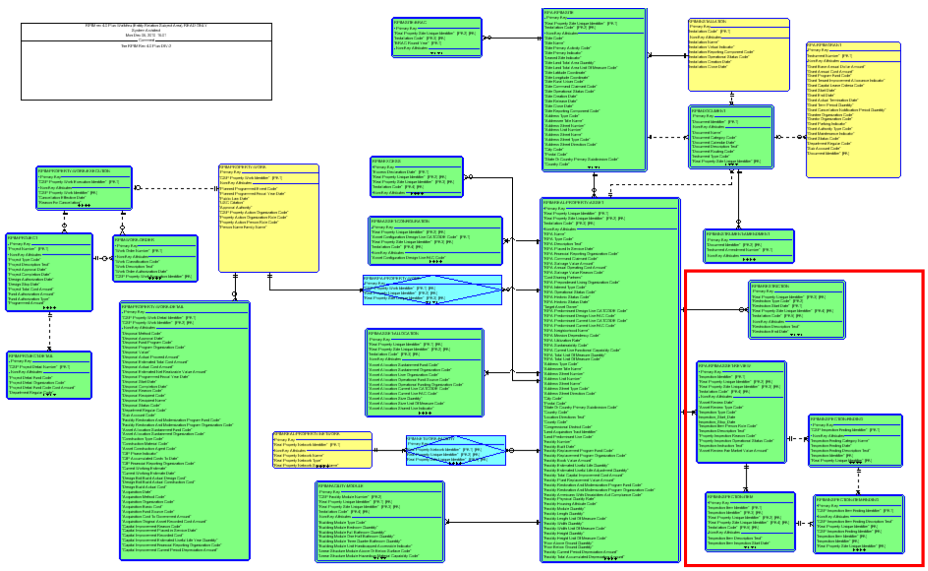


Figure 7-6, Entity Relationship Diagram

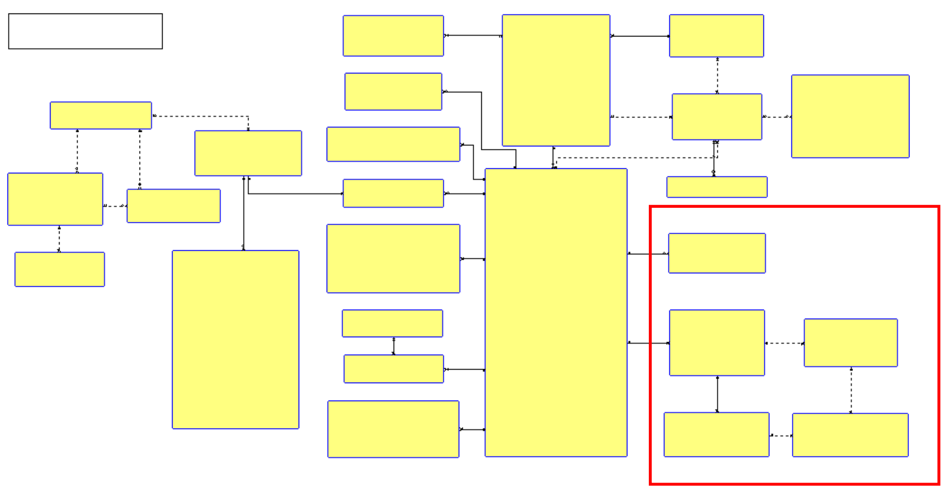


Figure 7-7, Physical Data Model

7.3.3 LDM, ERD and PDM Details

The following three figures are snippets of the same five data objects represented in the previous LDM, ERD and PDM figures. While the basic structure is similar, the details change.

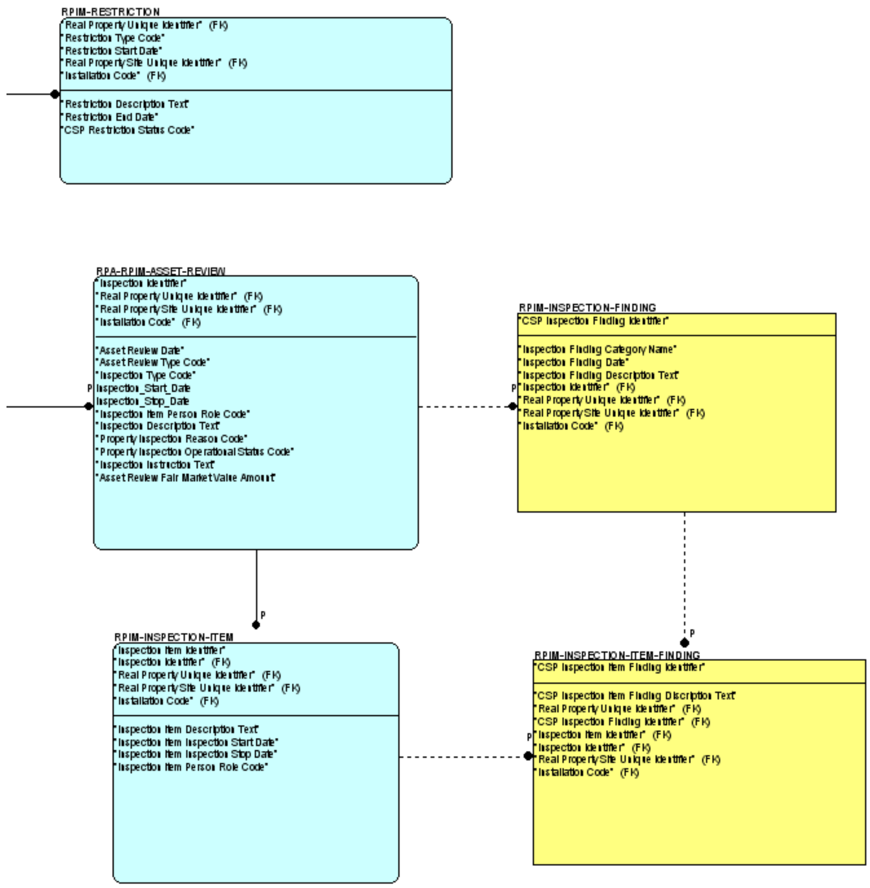


Figure 7-8, Logical Data Model Snippet

Relationships in the LDM are denoted using the IDEF1X conventions. Identifying relationships are denoted by solid lines. Optionality and cardinality are denoted by symbols at the ends of the relationships.

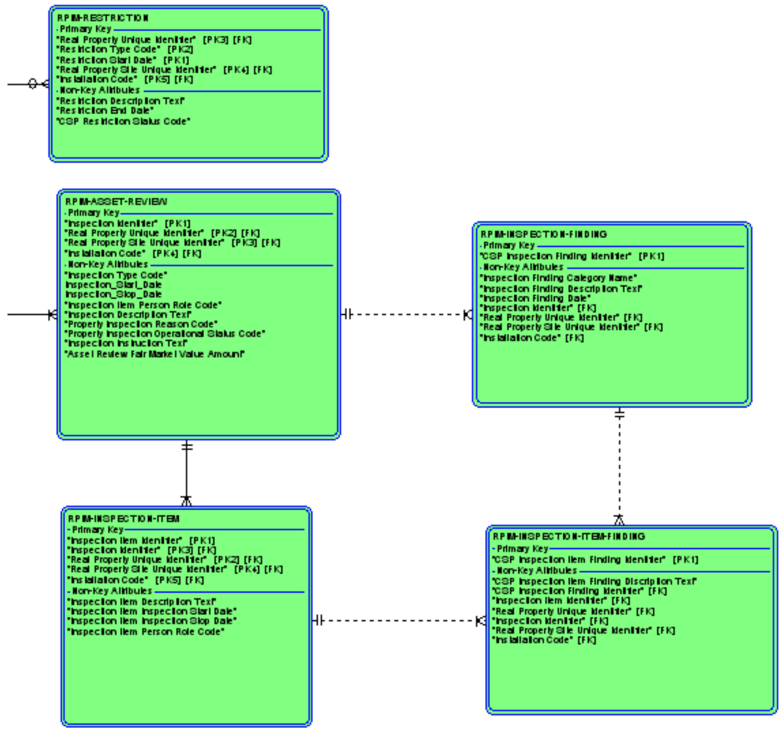


Figure 7-9, Entity Relationship Diagram Snippet

Relationships in the ERD are denoted using the "crows foot" convention. Identifying relationships are denoted by solid lines. Optionality is denoted by the "o" at the end of a relationship. Cardinality is denoted by the "crows foot" or the absence thereof.

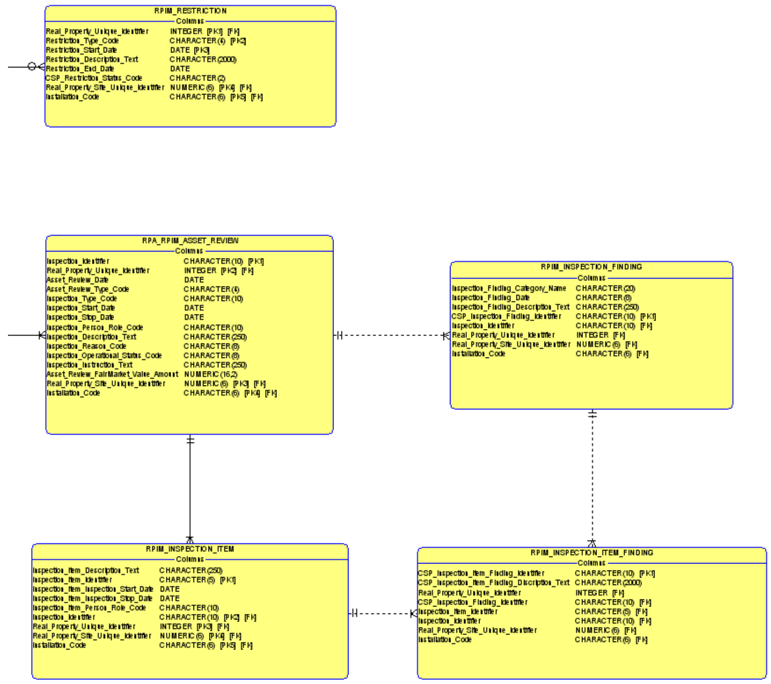


Figure 7-10, Physical Data Model Snippet

Relationships in the ERD are denoted using the “crows foot” convention. Identifying relationships are denoted by solid lines. Optionality is denoted by the “o” at the end of a relationship. Cardinality is denoted by the “crows foot” or the absence thereof. The PDM tables include the DBMS datatypes for the attribute domains and properties.

7.3.4 Generating the DIV-3

7.3.4.1 Pre-Generation Tasks

The LDM must be reviewed for technical correctness and for constructs that could cause implementation or operational problems.

Non-specific relationships must be resolved in the LDM. Non-specific relationships are:

1. Many to many – This relationship cannot be directly implemented.

2. Optional one on the independent entity – This relationship results in a null valued foreign key which means that the DBMS cannot enforce referential integrity. This can cause reporting anomalies. Also, business data is lost because there is no place to capture who, when or why the relationship was made.

Each involuted/fishhook relationship should be examined. This is a special case of modeling one instance of a hierarchy. In one instance of a hierarchy change history is lost. Typically, history is maintained, and more than one instance of hierarchy exists; for example, an organization chart.

Each of the above is resolved by adding another entity, attributes and relationship in the LDM.

Redundant parallel relationships must be examined and each redundant relationship must be removed. Parallel relationships occur when there is more than one path between the same two entities. This means that queries that follow each of the two paths through the database can return two different results.

Each entity with no business attributes should be reviewed. Typically, at least a date is required to provide business data.

Each entity with more than 10 business attributes should be reviewed. Typically, there are ten or less business attributes in an entity; however, there are exceptions. A large number of attributes may indicate that there are attributes dependent on other attributes in the entity.

Each attribute that is a code and is not a foreign key should be reviewed as to where do the code values exist. Perhaps a code table may be required.

Each foreign key should be reviewed to ensure that the entire identifier was migrated from the independent entity to the dependent entity. Partial identifier migration should be avoided to ensure reporting consistency.

7.3.4.2 Generation Tasks

The LDM is transformed into an ERD. The ERD entities and relationships must be arranged to the same positions in the diagram as the entities and relationships in the LDM.

Then, the ERD is transformed into the PDM. The target DBMS must be specified for the generation of the PDM. The PDM tables and relationships must be arranged in the position in the diagram as the entities and relationships in the ERD.

The transformations are automatically accomplished by SA generation processes.

7.3.4.3 Post Generation Tasks

The transformed PDM tables and relationships must be arranged in the same positions in the diagram as the entities and relationships in the LDM.

The PDM must be reviewed to ensure that all of the LDM structural business rules have been correctly represented in the PDM. The review should include the following:

1. Migration of identifiers as foreign keys
2. Assigned DBMS data types to rows
3. Optionality transformation
4. Cardinality transformation

Each change to the PDM must first be made in the LDM and then the transformation process from LDM to ERD to PDM must be re-executed.

8 OV-6c – Event Trace Description

8.1 Summary Description

This section describes the OV-6c Event Trace Description, its relationship to other BEA models, the development method, and the modeling guidelines to be followed. The standard DoDAF Event Trace model provides a time-ordered examination of the Resource Flows as a result of a particular scenario. However, the BEA uses a modified OV-6c Event Trace model that conforms to the Business Process Modeling Notation 2.0 (BPMN 2.0) Primitives and Patterns (Primitives) standards and conventions. The reason for this is that standard business processes are being defined across DOD and BPMN 2.0 Primitives is an accepted industry standard published by the Object Management Group. The BPMN 2.0 Primitives specification, used for the BEA OV-6c, is available at www.BPMN.org.

8.1.1 Model Purpose

The OV-6c provides a sequence-ordered examination of Business Process Steps to achieve a Business Capability. The Model represents the "To Be" Operational View of a Business Process, displaying a series of business steps that are executed sequentially, or in parallel, in response to business events, to produce a specific business result.

Secondary purposes include:

- Aligning LRP requirements and Business Rules with specific Business Processes.
- Providing a basis for capital planning and assessing the value of potential investment.
- Setting the foundation for controlled, systematic transformation.
- Establishing a basis for measuring the progress toward achieving transformation objectives.
- Establishing key criteria for testing and evaluating transformation solutions.
- A Business Process should produce a measurable improvement to a product or service. If the effect of a process cannot be measured, then it would be impossible to measure its effectiveness and would also be difficult to control.
- Business Processes should be as autonomous as possible. Tightly linked activities are less flexible and harder to change.
- Business processes should add value to a product or service. If they are not doing so, then the reason for their existence should be questioned.
- Linking Transaction Types to specific Business Processes.

8.1.2 Model Structure

The OV-6c model is depicted as a set of diagrams. It comprises OV-6c models that are developed using the BPMN 2.0 Primitives. BPMN 2.0 Primitives is a standard notation used across industry and Government to document Business Processes and is promoted by the Object Management Group. This standard has been developed specifically to model collaboration across organizations and support the implementation of a service-oriented architecture (SOA). The primary goal of BPMN 2.0 Primitives is to provide a standard notation that is readily understandable by all business stakeholders; the business analysts who create and refine the process; the technical staff responsible for designing and developing the software and infrastructure to support the process; and the business managers who implement, manage and monitor it. Consequently BPMN 2.0 Primitives is intended to serve as common language to bridge the communication gap that frequently occurs between business process design and implementation.

The OV-6c model are developed using primitive patterns as presented in Enterprise Architecture based in Design Primitives and Patterns - Guidelines for the Design of Business Process Models (DoDAF OV-6c) using BPMN 2.0 Primitives that was printed on February 11, 2009.

8.1.3 Relationship to Other BEA Models

Integrated architectures provide a structured and organized approach for defining capabilities and understanding the underlying relationships and requirements for achieving those capabilities. The full spectrum of the business can be effectively modeled and related in the OV-6c, so that detailed analyses and decisions can be supported by describing the sequence of business activities, tying them to Operational Nodes (representing functional areas, organizations or human roles), relating them to supporting systems or System Functions, and specifying the actions, events and related guard conditions or Business Rules that constrain those activities.

Figure 8-1, Relationship Between OV-6c and Other BEA Models depicts the linking rules and relationships among the following elements in other BEA models and the OV-6c elements. The OV-6c relates to other BEA models as follows:

AV-1	The scope of the development effort for each CBM for a development cycle, as disclosed in the AV-1, will determine if the OV-6c is affected in the release.
AV-2	All OV-6c terms with a specific meaning must be included in the AV-2 Term Definitions report. These terms must include, as a minimum, the following object types: • Events • Process Steps • Gateways • Data Objects All acronyms used in the OV-6c descriptions must be listed and spelled in the AV-2 Acronyms Definitions List.
OV-2	Pools represent OV-2 Nodes and Lanes represent OV-2 Subnodes
OV-3	IEs and subsets of an IE in the OV-3 are represented as Data Objects. Each Data Object may be linked to one and only one IE or subset of an IE.

OV-5	Process Steps in the OV-6c are derived from and linked to leaf-level Operational Activities in the OV-5b Operational Activity Model. Each Process Step must be related to an OV-5 Activity.
OV-6a	Each Process Step may be linked to one or more OV-6a Business Rules. Action Assertion Business Rules from the OV-6a help to define and are linked to Process Steps and Gateways in the OV-6c.
E2E	A Process Step may be mapped to an E2E Business Flow object.
DFMIR/FFMIA	A Process Step may be mapped to a DFMIR/FFMIA Guidance statement.

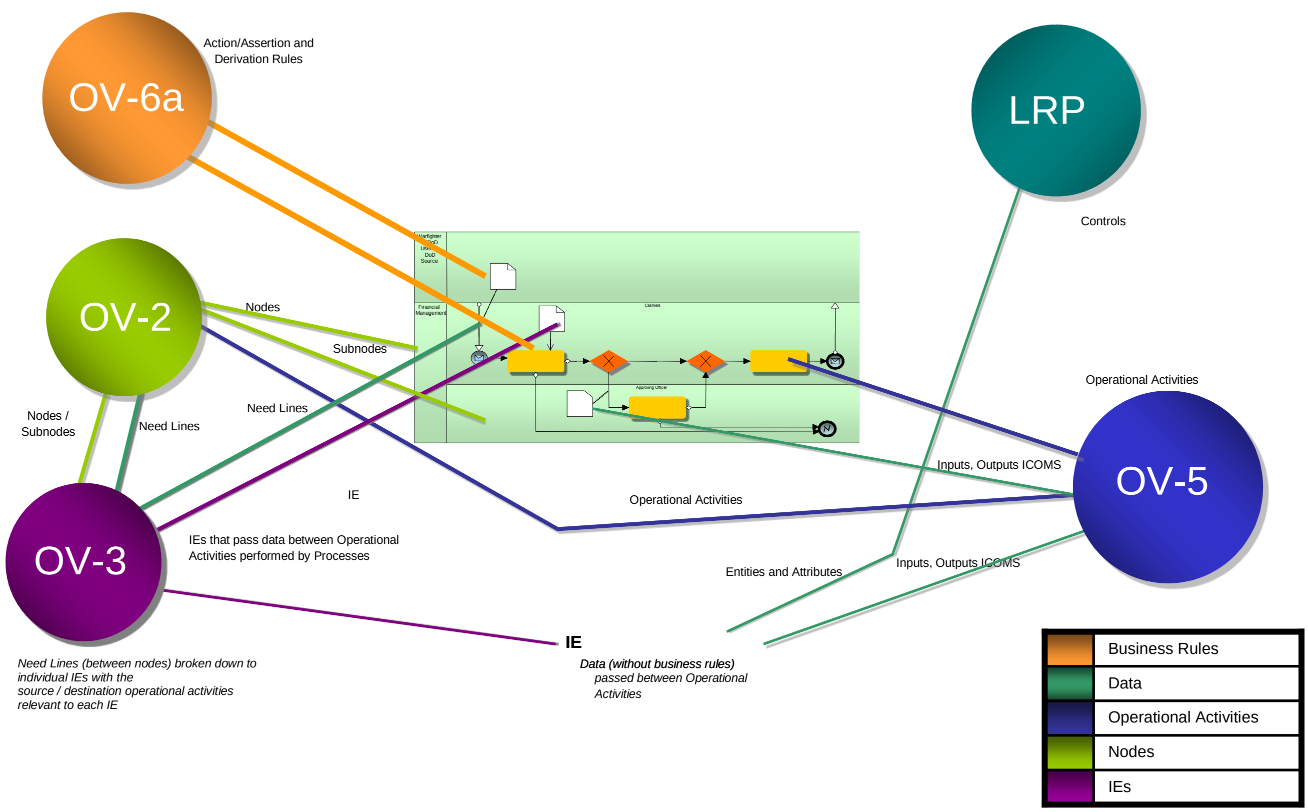


Figure 8-1, Relationship Between OV-6c and Other BEA Models

8.1.4 OV-6c Model Definitions

The OV-6c model consists of a set of BEA BPMN 2.0 Primitives Diagrams, which may drill down in detail. The BEA repository contains descriptions, attributes and linkages to objects in other BEA models described in this document. The OV-6c Diagram objects, as shown in Figure 8-2, Objects of an OV-6c Diagram, are characterized into four major groups: Flow Objects, Connection Objects, Participants and Artifacts.

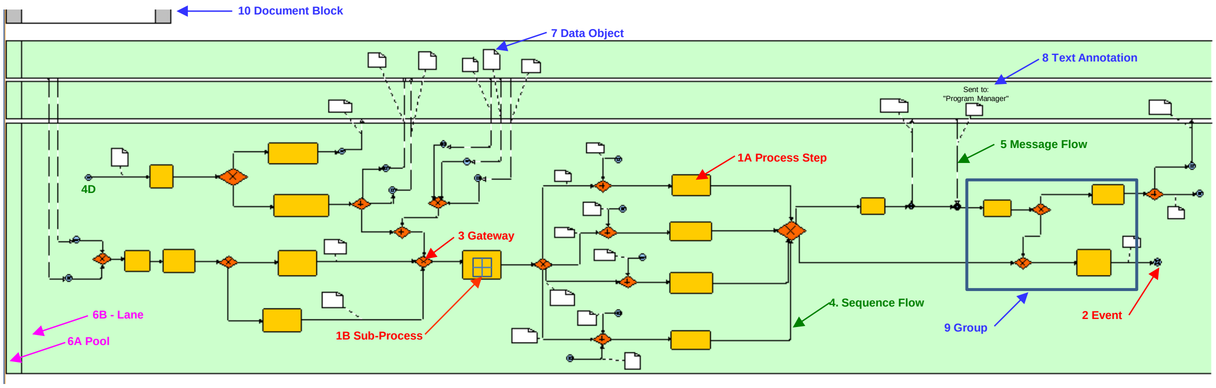


Figure 8-2, Objects of an OV-6c Diagram

Flow Objects actually perform the work and produce the models, synchronize the Process Steps, and direct the process flow. Red numbers identify these on Figure 8-2, Objects of an OV-6c Diagram. The Flow Objects are:

- **Process Steps** perform the work and produce the model. Process Steps, are also called *tasks* **(1A)**. Process Steps that are further decomposed into subtasks are called *Sub-Processes* **(1B)**. Sub-Processes are identified with a "+" sign at the bottom center of the Process Step symbol. Sub-Process detail is included in a separate Diagram.

The BPMN specification identifies Process Step attributes that address the association between Process Steps and Data Objects Input Data Objects, Output Data Objects, and IO Rules.[5]

- o **An Input Data Object** is simply a group of BPMN artifacts, which define the data requirements for input to the Process Step. [6] (Note that a Process Step may have more than one Data Object.) An Input Data Object has an attribute, *Required For Start*, which, when set to *True*, denotes that the Data Object is required for the Process Step to start.[7]
- o **An Output Data Object** is simply a group of BPMN artifacts, , which define the data requirements for output from the Process Step.[8] (Note that a Process Step may have more than one Output Data Object.) An Output Data Object has an attribute, *Produce At*

[5] BPMN v1.1, Appendix B.6.1, Common Activity Attributes

[6] BPMN v1.1, Appendix B.11.10, InputSet

[7] BPMN v1.1, Appendix B.11.1, ArtifactInput

[8] BPMN v1.1, Appendix B.11.13, OutputSet

Completion, which, when set to *True*, denotes that the Data Object is produced when the Process Step has been completed.[9]

- **IO Rules** specify the required relationship between one Input Data Object and one Output Data Object. That is, if the process activity is started with a specified input, that Process Step shall complete with the specified output.

- **Events** act like traffic signals and hold up the process or allow it to proceed in response to things that happen, called *triggers*. A *Start Event* starts the process in response to a trigger (in this case receipt of one of many allowable messages, shown by the envelope stereotype). An *End Event* **(2)** signifies the completion of the process. There are three types of Events, based on when they affect the flow, *Start, Intermediate* and *End:*
  - **Start Event** – A Start Event indicates where a particular process will start. In terms of Sequence Flow, the Start Event starts the flow of the process, and thus, will not have any incoming Sequence Flow. A Trigger is a mechanism that signals the start of a Business Process. A Start Event shall have a Trigger, indicating how the process starts: Message, Timer, Rule, Link, or Multiple. The Start Event shares the same basic shape of the Intermediate Event and End Event, a circle, but is drawn with a single thin line.
  - **Intermediate Event** - Intermediate Events occur between a Start Event and an End Event. This is an Event that occurs after a process has been started; it will affect the flow of the process but will not start or (directly) terminate the process. The Intermediate Event shares the same basic shape of the Start Event and End Event, a circle, but is drawn with two thin lines.
  - **End Event** - The End Event indicates where a process will end. In terms of Sequence Flow, the End Event is the end of a Task or an output that concludes the process, and thus, will not have any outgoing Sequence Flow. An End Event can have a specific Result that will appear as a marker within the center of the End Event shape. End Event Results are Message, Error, Compensation, Link and Multiple.

If there is more than one starting or ending Event for a process, combine them into a single Multiple (start or end) Event. Link all Initiating Events or Ending Events to the definition of the Multiple Event. Create a graphic comment on the Event that starts with "Initiating Events:" and list each initiating Event in quotes on a new line. If there is more than one Sequence Flow from the Start Event, use "Initiating Events" followed by a list of Initiating Events for that process. Follow this by a blank line and repeat for each process.

- **Gateways** control the divergence and convergence of a flow. Thus, it determines decisions, as well as the forking, and merging paths. The following types of Gateways may be used, depending in the conditions:
  -  **Parallel Gateway** (Parallel Split/Merge) - Parallel decision Gateways take only one (of many possible) outgoing paths regardless of the input Sequence Flow. Parallel Merge Gateways take the single output Sequence Flow when any (of many possible) input Sequence Flow occurs.

[9] BPMN v1.1, Appendix B.11.2, ArtifactOutput

-  **Event-Based Exclusive Gateway** - The Business Process may need to make a decision based on an Event, such as the receipt of a message or the passage of time. Use an *Event-Based* Gateway. The Event triggers determine the course of action. The *Event-Based* decision Gateway must be used when the passage of time is not data-dependent.
-  **Inclusive Gateway (**Inclusive OR Split/Merge) - Inclusive Gateways take more than one Sequence Flow when more than one decision condition evaluates as True. Think of each condition independently activating its own Sequence Flow. The major difference between Inclusive and Exclusive Gateways is that the Exclusive Gateway only takes one Sequence Flow when more than one condition evaluates as true, while the inclusive Gateway takes all the Sequence Flows.
- **Data based Exclusive Gateway** (Exclusive OR Split/Merge) – Indicates a complex or compound set of conditions that determine the Sequence Flow of the Process, commonly used to simplify and replace a set of linked Gateways. Process Architects provide complex expressions that determine the merging/splitting behavior of the Gateway. Back-to-back Gateways should be avoided unless their use clarifies the process flow.

Connection Objects are represented by arrows that show the flow via sequence or by synchronizing messages between different organizations. These are identified on Figure 8-2, Objects of an OV-6c Diagram, with green numbers. The two types of Connection Objects are:

- **Sequence Flows**, shown by solid arrows, indicate the direction of the Process, from one Process Step to the next. Displaying the name of a Sequence Flows **(4)** is optional. Some Sequence Flows have an *initiating Event* **(4D)** that triggers the Sequence Flow, such as the receipt of a message.

  13.3 A Process Step has one input Sequence Flow and one output Sequence Flow. An exception is when a Process Step also contains a throwing Event such as a timer or an exception2.

  13.4 Gateways are used to control multiple input Sequence Flows and direct multiple output Sequence Flows.

- **Message Flows**, shown by dotted arrows, denote messages between Participants (represented by separate *Pools*) that synchronize their separate internal processes. Message Flows are graphical representations of IEs or subsets of IEs on the OV-6c.

Participants and their roles are represented, respectively, by Pools and Lanes within Pools.

- **Pools (6A)** are represented by an open rectangle with a *Participant's* name on the left. The Pool contains the processes performed by a Participant. It also acts as a graphical container for portioning a set of Process Steps from other Pools. A Pool may be further divided into **Lanes (6B)**, if it is necessary to show *Roles* within the Pool.

  A Pool represents a Participant in a workflow collaboration. A Pool is a container for partitioning a Process from other Pools/Participants. A Pool is not required to contain a Process. That is, it can be thought of as a "black box". However, such a Pool must have a Message Flow.

A Participant is a business entity such as a company division, customer or business role that controls or is responsible for a business process or workflow. When Pools are modeled a specific Participant is associated with only one Pool. The name of the Pool must be name of the Participant and be a valid DoDAF Performer.

The diagram may contain *Artifacts*; notations that do not affect the process flow, but provide clarity to the reader. With the exception of the Document Block, Artifacts should be used only when necessary. Artifacts on the OV-6a diagram include:

- **Data Objects**, represented by a folded paper icon, reflect data that is consumed or produced by a Process Step. Data Objects are the mechanism to show what data is consumed or produced by Process Steps. Data Objects are graphical representations of IEs on the OV-6c.
- **Text Annotations** are comments provided on the diagram for clarification purposes. The OV-6c analyst frequently uses this artifact to show where Messages, Sequence Flows, or Data Objects come from or go to.
- **Groups**, shown as a box enclosing multiple diagram objects, indicate a grouping of diagram objects that does not affect the Sequence Flow. It can be used for documentation or analysis purposes. Groups may also be used to identify the objects related to a distributed transaction that is shown across Pools.
- **Document Block** is a text box located in the upper left corner of the diagram containing the diagram name and the last update date. For more details see Section 8.3.1.
- **Association,** represented by a dotted line with an arrowhead, indicates data, text, and other Artifacts associated with a Flow Object. Associations are used to show the inputs and outputs of Process Steps.

8.2 Developing the OV-6c

A top down modeling approach is used; at each level of decomposition, more detailed information is added. The OV-6c BPMN 2.0 Primitives diagrams depict end-to-end Business Processes representing how a Business Capability is achieved and its interaction with other Business Capabilities. DoD strategic direction for business transformation has evolved, requiring that future Business Processes be aligned with their respective Business Capabilities. SMEs and architects enhance and extend the current models as gaps and or new capabilities are identified.

Model development or extension of a current OV-6c BPMN 2.0 Primitives is accomplished in facilitated workshops to address model content and provide preliminary validation of the results. The remainder of this subsection describes in detail the approach to develop the OV-6c. Each subsection represents a step in the approach, and the specific tasks that must be accomplished to complete a given step. Although most of these steps are sequential, some may be started before the previous step is completed. The appropriate standards/guidelines that direct task accomplishment are contained in subsection 8.3.

8.2.1 Pre-Development Tasks

Analyze BEA Improvement Proposal

If there is an existing OV-6c for the Capability related to the BIP ensure that Operational Activities identified in the Decomposition Tree as supporting the BIP are represented as Process Steps in the OV-6c and are sequenced correctly.

Review fully attributed OV-5 and OV-2

Ensure all Activities are represented as Process Steps in the OV-6c, verify Input and Output ICOMS, verify OV-3 IE's for Data Objects linkages and usages, and verify Nodes in the OV-2 to provide Pool and Lane structure.

8.2.2 Development Tasks

The primary source for changes to the OV-6c is the BEA Stakeholder working group. Each team will conduct workshops with appropriate SMEs and business analysts from the BEA Stakeholder community. During the workshops, business analysts capture proposed changes to models and/or object descriptions. The business analysts conduct detailed analysis of approved changes and raise integration issues for resolution.

After revising all available materials and assessing the requirements, the OV-6c architect may develop new/revised OV-6c objects, based on the architectural standards in this document, in cooperation with the BEA Stakeholder analyst. The objects are driven by Stakeholder requirements in accordance with the configuration management procedures. This may involve updating existing symbols/definitions or creating new ones.

8.2.2.1 Creating/Modifying the OV-6c Models

Create a new diagram if the diagram is a decomposition of a process or Sub-Process. Modify existing diagrams if the modifications are a result of a workshop review.

BPMN 2.0 Primitives Diagrams

- Create or modify an existing diagram. If the new diagram is a decomposition of an existing Process Step, the new diagram shall be created as a child to the existing Sub-Process. The new diagram should inherit the same name as the parent Sub-Process. For each newly created diagram, the BEA Stakeholder representatives should develop a summary of the process model in the diagram "properties" dialog box.
- When creating a new diagram, ensure SME provides proper name and description. If the diagram already exists and content is added or changed, update name and description as required.
- Make sure a Document Block is in the upper left hand corner of the diagram and it contains all pertinent information including a Diagram Title and Diagram Type.

Participants

- Create new roles (Lanes) or modify existing ones per CBM guidance:
    - Individual Stakeholder teams may design role-based Lanes within its CBM Pool.
    - Each process has one or more CBM Stakeholders assigned.

Events

- Create/Revise Events (start, intermediate, and end). Each Event shall have clear and concise name and a well-formulated description that identifies and describes the trigger for start and intermediate Events and the result for End Events. Every diagram must have at least one start and one End Event. The following are the primitive events used in the BEA.
    - Start Event: A start event indicates the first node of a process.
    - Message Start Event: A message start event indicates that the process will start once a particular message has been received.
    - Signal Start Event: A signal start event indicates that the process will start once a broadcast message has been observed.
    - Timer Start Event: A timer start event indicates that the process will start at a specific time (or after a specific delay).
    - Conditional Start Event: A conditional start event indicates that the process will start when a set of rules (conditions) evaluates to true.
    - Intermediate Message Catching Event: An intermediate message catching event indicates that the execution of the process will halt until a specific message has been received.
    - Intermediate Timer Catching Event: An intermediate timer catching event indicates that the execution of the process will halt until a specific message has been received.
    - Intermediate Signal Catching Event: An intermediate signal catching event indicates that the execution of the process will halt until a broadcast signal has been observed.
    - Intermediate Conditional Catching Event: An intermediate conditional catching event indicates that the execution of the process will halt until a specific set of rules evaluates to true.
    - Intermediate Message Throwing Event: An intermediate message throwing event indicates that the process will send a message to a specific recipient at the point specified.
    - Intermediate Signal Throwing Event: An intermediate signal throwing event indicates that the process will send a broadcast signal at the point specified.
    - End Event: An end event indicates the last node of a process.
    - End Message Event: An end message event sends (throws) a message at the end of the process.
    - End Signal Event: An end signal event broadcasts (throws) a signal at the end of the process.

Process Steps

- Create/Revise Process Steps, Sub-Processes, and tasks. All processes shall have a clear and unambiguous description which describes in detail how the following participate within the Process Step:
    - **Inputs**: what is consumed or used as reference.
    - Pertinent **Business Rules**.
    - Value-added action; what is performed, what decisions are to be made.
    - **Outputs:** what is created or altered.

- Place Process Steps from left to right and top to bottom when possible.

Data Objects

- Data Objects may be associated to Sequence Flows, Message Flows, or as inputs or outputs to Process Steps. SA allows multiple Data Objects to be associated to a Sequence Flow or Message Flow in the Flow Objects definition.

Note: Data Objects are graphical representations of IEs or subsets of IEs on the OV-6c.

- Not every Message or Sequence Flow needs to have a Data Object associated with it. Add Data Objects only when it adds clarity to the process.

Sequence Flows

Sequence flows are BPMN Connecting Objects that relate BPMN Flow Objects (Events, Process Activities, or Gateways) that are in the same Pool.

- Sequence Flows are not required to have names unless the name adds clarity.
- If a Sequence Flow has a name, then the Sequence Flow must have a Description. Place Sequence Flow names above the flow line and close to the arrowhead, whenever possible.

Note: A sequence flow name uniquely identifies a sequence flow SA artifact. It may be referenced (by a symbol) multiple times either in one diagram or in other diagrams, but its attributes are the same for all instances.

Message Flows

Message Flows are BPMN Connecting Objects that relate BPMN Participants that are not in the same Pool. The Participants involved in a Message Flow are two Pools or optionally two Process Activities that are not in the same Pool or a combination of both – a Pool and a Process Step in a different Pool.

- Message Flows cannot originate from a start Event
- A message Flow to an external (remote) Pool and must be to the Pool and not to a Process Step in the external Pool.
- Message Flows may only be drawn between Pools or Processes in the Pools and not to Data Objects
- Place Message Flow names above the flow line and close to the arrowhead, whenever possible
- Message Flow properties include:
    - Source and target Pools or Processes
    - Definition of the message passed

Note: Message Flows are graphical representations of IEs or subsets of IEs on the OV-6c. A message flow name uniquely identifies a message flow SA artifact. A message flow may be referenced (by a symbol) multiple times either on the same diagram or different diagrams, but its attributes remain the same for all instances.

Gateways

- Create Gateways to represent a merge or split of the process flow. The following are the primitive gateways used in the BEA.
    - Parallel Gateway (Parallel Split/Merge): A parallel gateway splits one process thread into multiple concurrent threads or merges multiple concurrent threads into one thread via a synchronized join (i.e. the outgoing sequence flow will only be activated once all incoming sequence flows have been activated).
    - Data-based Exclusive Gateway (Exclusive OR Split/Merge):A data-based exclusive gateway when used as a split routes the sequence flow from one incoming flow to exactly one of multiple outgoing flows.
    - Inclusive Gateway (Inclusive OR Split/Merge): An inclusive gateway when used as a split activates one or more branches based on branching conditions. When used as a merge, it awaits all active incoming branches to complete.
    - Event-based Exclusive Gateway: An event-based exclusive gateway is always followed by catching events or receiving tasks. Sequence flow is routed to the subsequent event/task which happens first.
- When the results of a Gateway split are merged, the Gateway type used to create the split must be used to merge the results.
- Diagram layout should highlight logical structure by using standard patterns to show parallel or alternate paths and iteration.
    - The condition of the Gateway should be expressed in the form of a question, as specific and succinct as possible. The question and answers must set forth the context of the gateway, and the set of answers must be global.
    - Business Rules should be identified, whenever appropriate, to address the logic of the Gateway.
    - The Gateway should be identified as a Data-based Gateway or as an Event–based Gateway.
    - Data-based Gateways use the values of process data to determine which path should be taken.
    - The Event-Based Gateway uses the basic idea that the Decision represents a branching point in the process where the alternatives are based on Events that occur at that point in the process, rather than the evaluation of expressions using process data. A specific Event, usually the receipt of a message or expiration of a timer, determines which of the paths will be taken.
    - Whenever possible, use complex Gateways to avoid using back-to-back Gateways.
- All diagram objects must have a clear and concise description except un-named Sequence Flows, all Message Flows, and all Event-Based Gateways.

8.2.2.2 Main Pool and Remote Pool Process Messaging Modeling Conventions

Processes in the Main pool can send and receive messages from Processes in Remote Pools where the Remote Pool is not the primary object being modeled. Processes in the Remote Pool should be shown when constraints or dependencies are being documented.

There are three variations to messaging modeling between Pools.

The first variation is shown below where the Processes in the Remote Pool are not connected with each other. This convention is used when the Processes in the Main Pool are not dependent on the workflow in the Remote Pool.

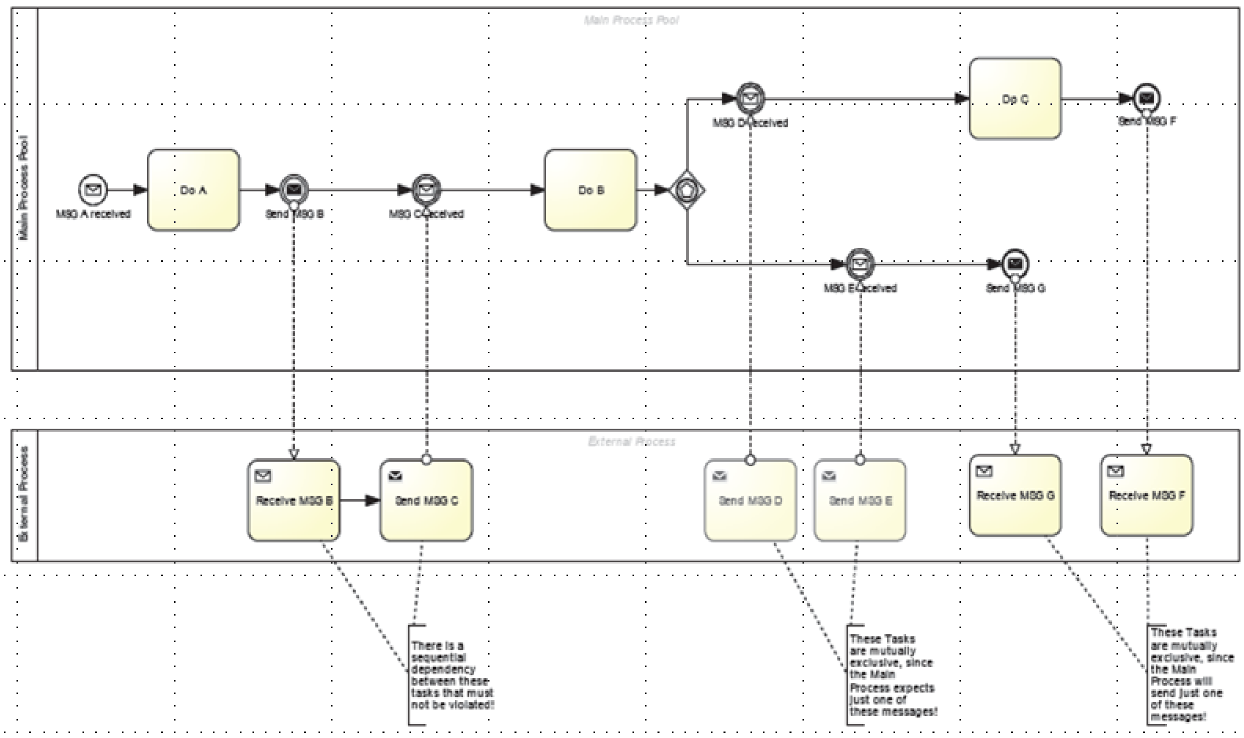


Figure 8-3, Main Pool Not Dependent of Remote Pool Workflow

The second variation is shown below where the Processes in the Remote Pool are connected with each other. This convention is used when the Processes in the Main Pool are dependent on the workflow in the Remote Pool.

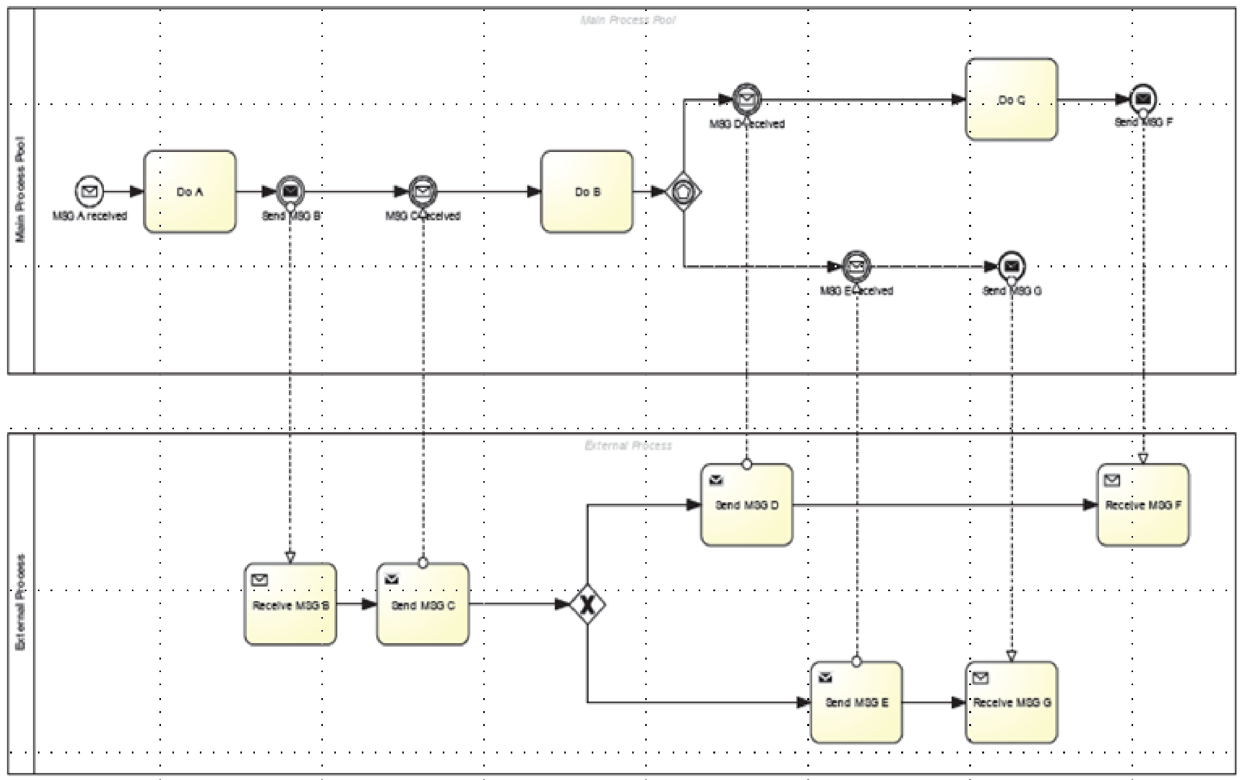


Figure 8-4, Main Pool Dependent of Remote Pool Workflow

The third variation is shown below where the Remote Pool does not contain any Processes and the Message Flows are to and from Processes in the Main Pool to and from the Remote Pool container.

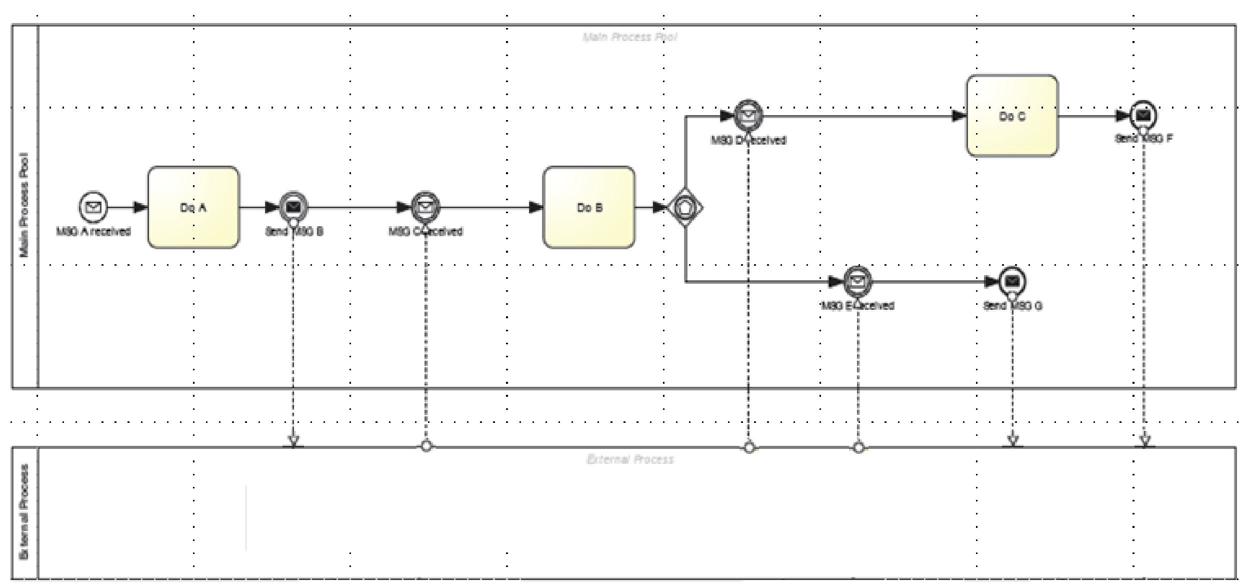


Figure 8-5, Main Pool Dependent of Remote Pool Processes

8.2.2.3 BPMN 2.0 Primitives and BEA Conformance Rules

This section itemizes the BPMN 2.0 Primitives and BEA diagram rules to which the OV-6c models must conform. The BPMN 2.0 Primitives conformance rules are from the OMG BPMN 2.0 Primitives Specification. The BEA Conformance Rules are specific to the BEA.

8.2.2.3.1 BPMN 2.0 Primitives Conformance Rules

- All flow objects other than Start Events, Boundary Events, and Compensating Activities must have an incoming Sequence Flow, if the Process level includes any Start or End Events.
- All flow objects other than End Events and Compensating Activities must have an outgoing Sequence Flow, if the Process level includes any Start or End Events.
- A Start Event cannot have an incoming Sequence Flow.
- A Start Event cannot have an outgoing Message Flow.
- A Start Event with incoming Message Flow must have a Message trigger.
- A Start Event cannot have an Error trigger.
- A Start Event in a SubProcess must have a 'None' trigger.
- A Boundary Event must have an outgoing Sequence Flow.
- A Boundary Event trigger must be only a Message, Timer, Signal, Error, Escalation, Conditional, Cancel, or Compensation.
- A Boundary Event cannot have incoming Sequence Flow.
- An Error Boundary Event on a SubProcess requires a matching Error throw Event.
- An Error Boundary Event cannot be non-interrupting.
- An Escalation Boundary Event on a SubProcess requires a matching Escalation throw Event.
- An Intermediate Event with incoming Message Flow must be catching type with Message trigger.
- An Intermediate Event with outgoing Message Flow must be throwing type with Message trigger.
- A throwing Intermediate Event result must be only a Message, Signal, Escalation, Link, or Compensation.
- A catching Intermediate Event trigger must be only a Message, Signal, Timer, Link, or Conditional.

- A throwing Link Event cannot have outgoing Sequence Flow.
- A catching Link Event cannot have incoming Sequence Flow.
- An End Event cannot have outgoing Sequence Flow.
- An End Event cannot have incoming Message Flow.
- An End Event with outgoing Message Flow must have Message result.
- A Gateway cannot have incoming Message Flow.
- A Gateway cannot have outgoing Message Flow.
- A Split Gateway must have more than one gate.
- An Event Gateway must have either a catching Intermediate Event or Receive task on each gate.
- A Sequence Flow cannot cross a Pool or SubProcess boundary.
- Sequence Flow out of a Parallel Gateway cannot be conditional.
- A Message Flow cannot connect nodes in the same Pool.
- A Message Flow only comes from a Message End or Intermediate Event; Send, User, or Service task; SubProcess; or black box Pool.
- A Message Flow only goes to a Message Start or Intermediate Event; Receive, User, or Service task; SubProcess; or black box Pool.

8.2.2.3.2 BEA Conformance Rules

- Conditional Sequence Flows are not permitted.
- A Task or Activity must have only one incoming Sequence Flow.
- A Task or Activity must have only one outgoing Sequence Flow.
- A matched pair of split and merge Gateways should be of the same type.
- The label of a child-level page must match the name of the SubProcess.
- Tasks and Activities must be labeled.
- A Send Task or Activity must have an outgoing Message Flow.
- A Receive Task or Activity must have an incoming Message Flow.

- A Message Start Event must have an incoming Message Flow.
- A Message Start Event must be labeled '[message name] *Receive*'.
- A Signal Start Event must be labeled to indicate the Signal name.
- A Conditional Start Event must be labeled to indicate the condition.
- A Start Event in a top-level Process must be labeled. If a top-level Process contains more than one Start Event, all must be labeled to identify the alternative start conditions.
- A Boundary Event must be labeled.
- An Error Boundary Event on a SubProcess must be labeled to match the throwing Error Event.
- An Escalation Boundary Event on a SubProcess must be labeled to match the throwing Escalation Event.
- A throwing Intermediate Event must be labeled.
- A catching Intermediate Event must be labeled.
- A catching Message Event must have an incoming Message Flow.
- A throwing Message Event must have an outgoing Message Flow.
- Two End Events in a Process level must not have the same name. If they mean the same end state, combine them; otherwise give them different names.
- An End Event must be labeled with the name of the end state.
- An Exclusive, Inclusive, or Event Gateway must have at all gates labeled.
- If a SubProcess is followed by a Gateway, at least one End Event of the SubProcess must be labeled to match the Gateway label.
- A Message Flow must be labeled with the name of the Message.
- A Message Flow from a collapsed SubProcess must be replicated in the child-level diagram.

A Message Flow to a collapsed SubProcess must be replicated in the child level diagram.

8.2.3 Post-Development Tasks

These tasks are performed after changes to the OV-6c have been approved by the BEA Stakeholders to ensure integration of the architecture:

- Use Microsoft Word to check the spelling and grammar of all objects in the diagram by exporting all object names, descriptions and graphic comments onto a Word document.
- Verify that all objects are connected via Sequence Flows, Message Flows or associations, as appropriate.
- Remove orphan objects, which do not appear on any diagram or are not connected to or contain any other object.
- Verify that only relevant Pools are included in the diagram.
- Verify that all Process Steps are mapped to an OV-5 Operational Activity.
- Incorporate additional updates to the OV-6c based upon subsequent BEA Stakeholders working group sessions.
- Incorporate peer reviews, quality control reviews, IV&V reviews and architecture verification changes into the BEA.

8.3 Modeling the OV-6c Using SA

8.3.1 OV-6c Modeling Conventions

This section is a brief overview of the OV-6c and the notation used in System Architect selected to model DoD Business Processes.

BPMN 2.0 Primitives was selected to depict Business Processes in the BEA. BPMN 2.0 Primitives is a standard notation and is utilized across industry and the government to document their Business Processes. The Business Process Management Initiative developed BPMN 2.0 Primitives.

An OV-6c diagram consists of a set of graphical elements. These elements enable the development of diagrams that have a familiar look to most business analysts (for example, a flowchart diagram). *The diagram elements are chosen to be clearly distinguishable from each other and to utilize shapes that are familiar to most process architects.* For example, Process Steps are represented by rounded rectangles and Gateways by diamonds. It should be emphasized that one of the drivers for choosing BPMN 2.0 Primitives is that it facilitates development of simple business process models, while handling the complexity inherent to real Business Processes.

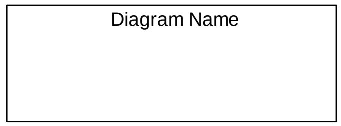
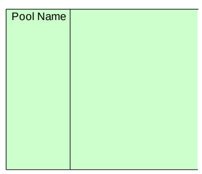
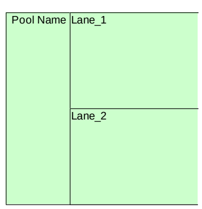
BPMN 2.0 Primitives provides a graphical notation for expressing Business Processes, as reflected in DoDAF. The objective is to support process management for both technical and business users by providing a notation that is familiar and understandable to business users, yet able to represent complex process semantics.

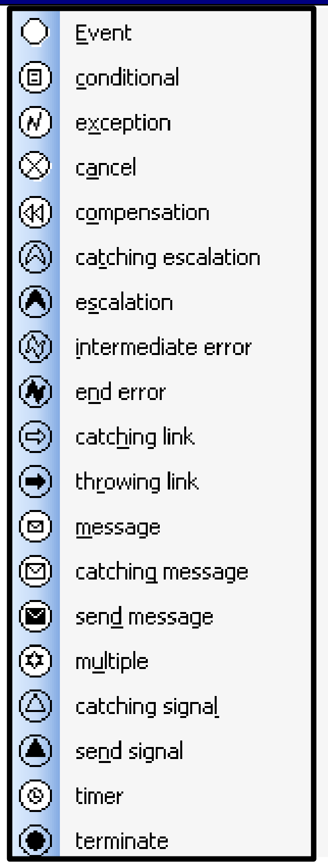
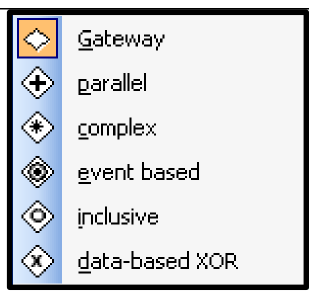
The OV-6c provides the capability of understanding internal business procedures in a graphical notation and gives organizations the ability to communicate these procedures in a standard format.

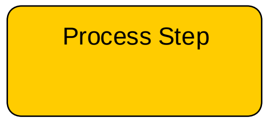
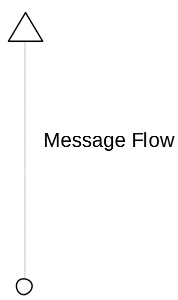
The OV-6c should not represent Data Flow modeling. However, the OV-6c may depict the flow of data (messages) and the association of Data Objects to Activities in a manner consistent with the OV-5 and OV-3.

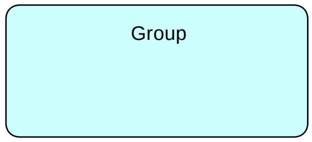
While the general flow of objects within diagrams will be from left to right and top to bottom, due to programmatic and space constraints, objects may be connected to other objects in a manner that is logical and readable, taking into account the constraints listed in Section 8.2.2.

Table 8-1, Modeling Guidelines for OV-6c

Element	Symbol	Format
Document Block		Position: Upper Left Corner Border: Solid Black Fill: None Text: Color: Black Font: Arial Size: Default
Pool		Border: Solid Black Fill: Light Green Text: Color: Black Font: Arial Size: Default
Lane		Border: Solid Black Fill: Light Green Text: Color: Black Font: Arial Size: Default

Element	Symbol	Format
Event Primitives		Border: Solid Black Fill: Light Brown Text: Color: Black Font: Arial Size: Default
Gateway Primitives		Border: Solid Black Fill: Light Brown Text: Color: Black Font: Arial Size: Default

Element	Symbol	Format
Process Step		Border: Solid Black Fill: Gold Text: Color: Black Font: Arial Size: Variable
Sequence Flow		Line: Solid Black Text: Color: Black Font: Arial Size: Default
Message Flow		Line: Dashed Black Text: Color: Black Font: Arial Size: Pen 7
Data Object		Border: Solid Black Fill: White Text: Color: Black Font: Arial Size: 10

Element	Symbol	Format
Group		Border: Solid Black Fill: Contrasting light pastel Text: Color: Black Font: Arial Size: Default

8.3.2 Modeling OV-6c Objects

8.3.2.1 Diagram Conventions

All Diagrams must be clearly named and defined:

- Diagram Names shall contain at least one (1) verb and one (1) noun, unless the diagram is the Enterprise Process Model or one of the CBM threads.
- Avoid using generic terms such as “Manage”, “Perform”, “Execute”; instead use:
  - For “Manage”, use “Handle”, “Collaborate”, “Sustain”, “Maintain”, “Monitor”, etc.
  - For “Perform”, use “Implement”, “Develop”, “Produce”, “Distribute”, “Publish”, etc.
  - For “Execute”, use “Initiate”, “Finalize”, etc.
- Diagrams that depict a Sub-Process should be named for the Sub-Process. However, it is recognized that in the case of Reusable Sub-Processes this is not always possible.
- Each OV-6c Diagram shall include a description to provide a clear understandable narrative of what the Diagram portrays. This information should be included in the Diagram Properties.
- The Diagram description must be clear, concise and unambiguous. The description shall include, as a minimum, a summary of the main Process Thread, a reference to the Events and their relationship to other diagrams, a reference to the Gateways and the decisions made, and a summary of the major Business Rules that impact the diagram.

Participants, Data Objects, and Process Steps must have labels containing name and/or other attributes placed inside the shape. Events, Gateways Sequence Flows and Message Flows labels should be placed above the shape as much as possible. However, labels may be placed below or to the right or left of the object to enhance readability of the Diagram.

While extensible, OV-6c diagrams still have the basic look and feel for any viewer to easily understand a diagram created by any process architect. Thus, the footprint of the basic flow elements (Events, Process Steps and Gateways) should not be altered.

8.3.2.2 Object Naming Conventions

Objects in the OV-6c diagrams shall have a concise and intuitive name according to the following standards:

- All OV-6c object names shall be title-case. Nouns must be singular, unless the plural form is required to correctly describe the object. Use only approved acronyms.
  - Process Steps must be clearly named and defined. The Process Step name shall contain at least one verb in the present tense and one (1) noun. For example, "Analyze Record."
  - Events shall be clearly defined and labeled. Event names shall consist of at least one noun and one verb or adjective, for example, "Record Analyzed", "Booking Successful." Event names shall be as specific as possible, avoiding generic names such as "End", "Stop", or "Start." Do NOT use verb-noun names for Events; for example, "Send Notification" is not a proper name for an Event.
  - Data Objects shall be clearly named and defined. The name must have at least one noun that accurately describes the Data Object. A Transition State may be used as necessary to identify changes in Data Objects content or State.
  - Decision Gateways must be clearly named and defined with a combination of nouns and verbs conveying a question or query, ending in a question mark. The question and answers must include the context, as the answer must be global in nature.

Example: The question "Adjustment required?" with the answers "Adjustment not required" and "Adjustment required" are not acceptable because they may also refer to other unrelated adjustments elsewhere in the architecture. A more specific question incorporating context would be "Adjustment to cost model required?" with the answers being "Adjustment to cost model required" and "Adjustment to cost model not required" would be better.

  - Gateway Control Types should be displayed consistently.
  - Participants (Pools) and Roles (Lanes) names shall be composed of nouns, and adjectives, where appropriate, and must be clearly defined.
  - Groups may be used to cluster related objects. A name shall be assigned to the Group, defined using appropriate nouns and verbs.
  - All Object names shall be less than 80 characters long.
- The following special characters shall not be used in object names:
  - "*"
  - "(" or ")"
  - "-"
  - "/"
  - "&"
  - "?" (except in Gateways)
- Use initial uppercase for all object names. Incidental words, such as prepositions within the object name ("with", "at", "in", "and", "no", "not", "a", "an", "to", or "the"), shall be all lowercase.

- Object names shall be spelled correctly and shall not use future tense.
- Refer to the AV-2 for the approved list of acronyms and abbreviations.

8.3.3 OV-6c Best Practices

This section discusses best practices, including lessons learned from previous architecture development efforts and common modeling pitfalls to avoid.

8.3.3.1 OV-6c Lessons Learned

- When decomposing a Process Step into a Sub-Process, ensure that all of the objects in the decomposition diagram support the purpose of the diagram.
- Account for multiple Data Objects associated with a Sequence Flow or Message Flow as they show up incorrectly as orphans on the current BART reports.
- Account for initiating Events that only show up on Message Flows or Sequence Flows; they show up incorrectly as orphans on the current BART reports.
- Delete a Pool on a diagram when all objects are removed from it and no Message Flows touch it.
- Remove the object from the encyclopedia after the symbol has been removed from the diagram, after first determining that the object has not been referenced in any other diagram.
- Verify that comparison reports are run using the most up to date correct baseline BEA modeling tool encyclopedia.
- Use Microsoft Word to check description spelling and grammar.
- Track version dates of the baseline and approved models.
- Ensure correct Gateway stereotypes are used.
- Ensure correct Event stereotypes are used.
- Standard color coding of diagrams during the workshop is useful for participants to identify where content was added, changed or deleted. Standard color coding should be in line with that used in the BEA Compare reports.
- Ensure that all exception reports have been reviewed and resolved.

8.3.3.2 OV-6c Common Pitfalls

- Events are duplicated on parent and child diagrams
- Sequence Flow and Message Flow lines cross each other unnecessarily
- Object labels are hidden
- All condition expressions out of a Gateway are not identified
- Child diagrams are not associated with their parent Process Step
- Object descriptions are not comprehensive.
- Object naming conventions are not followed.
- Process Steps are not associated with Operational Activities.

- Sequence Flows between pools
- Message Flows between objects in a pool
- Message Flows to Process Steps in external (remote) Pools
- Conditional Sequence Flows
- Multiple input Sequence Flows
- Multiple output Sequence Flows
- Missing Start Event
- Merge Gateway type incorrectly matched with Split Gateway type
- Missing alterative flow from a Split Gateway type
- Start Event used as an Intermediate Event

9 OV-2 – Operational Resource Flow Description

9.1 Summary Description

This section describes the OV-2 architecture model, its relationship to other BEA models, the development method, and the modeling guidelines to be used.

9.1.1 Model Purpose

The OV-2 Operational Resource Flow Description is the graphical depiction of Operational Nodes and the related Need Lines for the BEA. OV-2 diagrams contain Operational Nodes, which are logical collections of Operational Activities extracted from the OV-5b Operational Activity Model, and Need Lines, which represent the collection of all required IEs between these Nodes.

The BEA OV-2 Operational Nodes are internal or external to the DoD BMA. For the current BEA OV-2, the internal Operational Nodes correspond to the DoD CBM. External Nodes are Nodes not strictly within the scope of the DoD CBMs (e.g., Industry, Customer/Vendor, and the U.S. Congress), but with which the internal nodes must exchange information to accomplish their functions.

The BEA OV-2 models typically avoid representing Operational Nodes as real physical facilities and focus on virtual or logical nodes that can be based on operational roles or missions. Where appropriate, system or physical nodes that constitute the location of an Operational Node may augment the description of an Operational Node in corresponding System View architecture models.

9.1.2 Model Structure

The OV-2 model is depicted as diagrams. It comprises Of Operational Nodes, IEs and Need Lines; as described in the following sections.

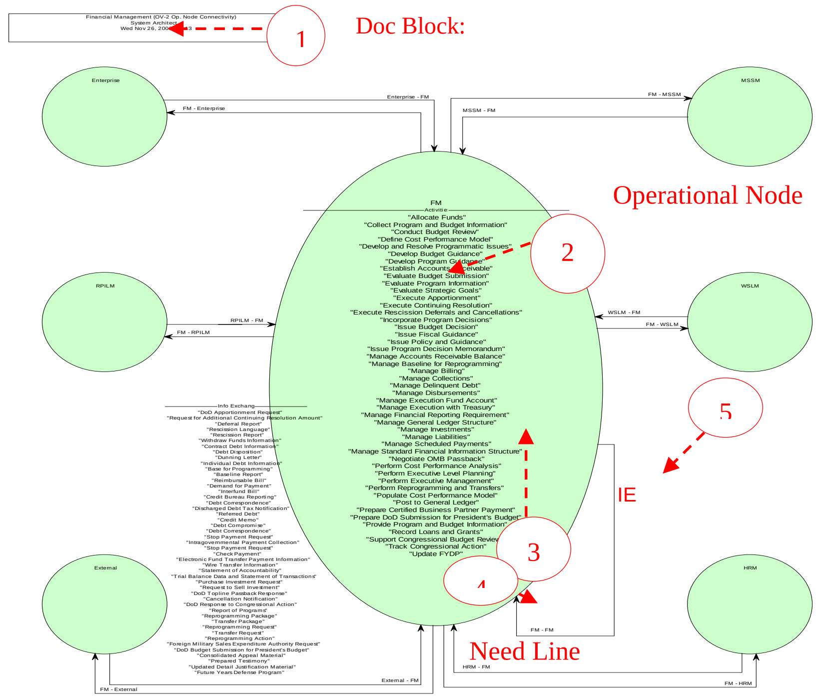


Figure 9-1, OV-2 Model Structure Diagram for Financial Management CBM

The main features of this diagram include:

- **Doc (title) block (1)** is a text box located in the upper left corner of the diagram containing the diagram name, and last modification date.
- **Example:** 'OV-2 Financial Management (OV-2 Op Node Connectivity) Mon Feb 06, 2006 15:07'
- **Operational Nodes (2)** are represented by the oval shapes in the diagram. In each OV-2 Diagram, the primary CBM Operational Node is centered on the diagram to identify it as the focus. Figure 9-1 shows the Financial Management (FM) CBM Operational Node exchanging information with the other CBMs (WSLM, HRM, RPLIM, MSSM, and Enterprise) as well as the External Operational Node. On each OV-2 diagram, only the primary node displays the list of related Operational Activities in the central Operational Node. As shown, the FM Operational Node displays the list of needed Operational Activities from the OV-5 model to associate with DoD Financial Management needs.
- ***Need Lines* (3, 4)** in Figure 9-1 symbolize the grouping of information to be exchanged between Operational Nodes. For example, the FM Operational Node for this diagram is exchanging information with itself through Need Line *FM-FM* (3) or with other Operational Nodes such as External through Need Lines FM-External or External-FM (4). Need Lines are labeled by an abbreviation that indicates the sending Operational Node to the Receiving Operational Node.

- **IEs (5)** are assigned to a Need Line, from ICOMs in the OV-5b models, and used to depict information being exchanged between OV-5b Activities assigned to Operational Nodes. An IE can be assigned to more than one Need Line, need Lines can have multiple IEs assigned to them and there is a many-to-many relationship between Need Lines and IEs.

Note: IEs are not shown graphically on the OV-2 diagram to avoid cluttering the model with too much information.

9.1.3 Relationship to Other BEA Models

As illustrated in Figure 9-2, Relationships Between OV-2 and Other BEA Models, the OV-2 is related to other BEA models as follows:

AV-1	The scope of the development effort for each CBM for a development cycle, as disclosed in the AV-1, will determine if the OV-2 is affected in the release.
AV-2	All OV-2 terms with specific meaning must be included in the AV-2 T4erm Definitions list. These terms must include, as a minimum, the following object types: • Need Line Definitions • Operational Node Definitions All acronyms used in OV-2 descriptions must be listed and spelled out in the AV-2 Acronym Definitions report.
OV-3	A Need Line in an OV-2 includes one or more IEs from the Operational IE Matrix (OV-3). The OV-3 provides the detailed attributes that define each IE.
OV-5	Operational Nodes in the OV-2 represent logical groupings of Operational Activities from the OV-5b Operational Activity Model. Once the OV-5b is stabilized, the Activities are assigned to the Operational Nodes in the OV-2, and related Inputs and Outputs from the OV-5b are then translated to IEs that depict the required information flow represented on the OV-2 as Need Lines between Operational Nodes.
SV-1	A System Node in a Systems Interface Description (SV-1) is linked to an Operational Node in the OV-2, indicating that the systems contained in that System Node are required to support the activities performed at the Operational Node.
SvcV-1	The Services Interface Description (SvcV-1) links together the operational and services architecture models by depicting key resource flows specified in the OV-2
StdV-1	Although all BEA models are related through integration of the architecture, no direct relationship exists between the OV-2 and the StdV-1.

DFMIR/ FFMIA Guidance Model	Although all BEA models are related through integration of the architecture, no direct relationship exists between the OV-2 and the DFMIR/FFMIA Guidance Model.

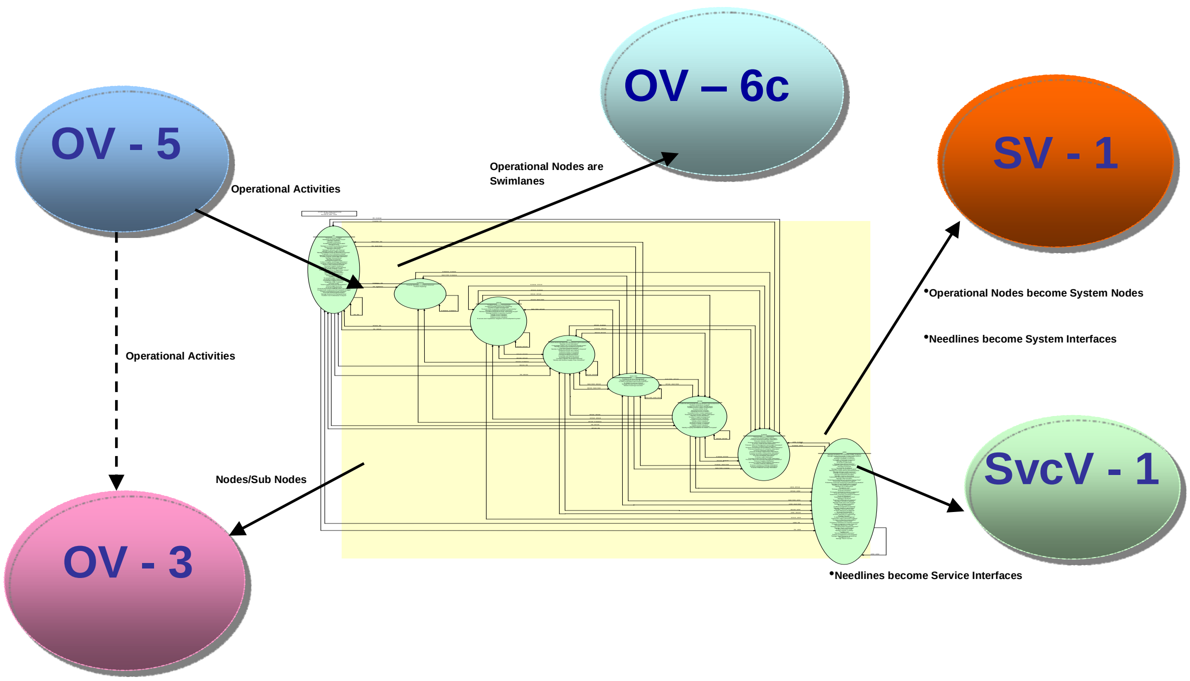


Figure 9-2, Relationships Between OV-2 and Other BEA Models

9.1.4 OV-2 Model Definitions

The following are definitions of the key elements contained in the OV-2 Operational Node:

- **Operational Node**: An Operational Node is an element of the Operational Architecture that represents a logical grouping of Operational Activities.
- **Need Line**: A Need Line documents the requirement to exchange information between Operational Nodes. The Need Lines are directional but do not indicate how the information transfer is to be implemented or sequenced. Need Lines are mapped to IEs, corresponding to Inputs and Outputs in the OV-5b Model Operational Activity diagram.
- **Operational Activity**: An action performed in conducting the business of an enterprise. This is a general term that does not imply a placement in a hierarchy or a timing sequence. For example, 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.

- **IE**: A communication between two Operational Nodes. A corresponding leaf-level Activity Input and Output ICOM is associated to the IE with the same name, definition, and CBM Stakeholders.

9.2 Developing the OV-2 Models

This section describes the approach to develop, extend and maintain the OV-2. The OV-2 is developed in SA as a diagram.

9.2.1 Pre-Development Tasks

The tasks that must be completed prior to OV-2 development and/or maintenance are:

- Identify, for each OV-2 CBM Focus Area diagram, the SMEs required for content contribution, validation and communication of Operational Nodes, Need Lines, IEs and hierarchy to primary stakeholders.
- Define specification/format for all object descriptions (Internal Operational Nodes, External Operational Nodes, Need Lines and IEs).
- Using approved sources and SME support, select Operational Nodes covering the full scope of the DoD BMA. Currently, the internal Operational Nodes will be limited to the Core Business Missions. For each focus diagram, a single External Operational Node will represent all external node interactions.
- Validate leaf-level Operational Activities to each Operational Node. Determine and map appropriate logical Operational Activities to each identified Internal and External Operational Node, based on Stakeholder requirements.
- Convert leaf-level ICOMs to IEs. From all Inputs, Controls, Outputs and Mechanisms associated with the leaf-level Operational Activities assigned to each Operational Node, identify the Inputs and Outputs (except for those Outputs that become Controls) and convert these to IEs. IEs are defined using the same name and definitions as their corresponding Input or Output ICOMs.
- Validate the external Operational Activity to be assigned to the correct leaf-level ICOM.
- Validate the parent ICOM to the child ICOM.
- Define and analyze Need Lines and IEs. Review and analyze from–to Operational Activity relationships for each IE. This is a starting point for determining the direction and exchange of information between Operational Nodes. Assign the Need Lines as groupings of IEs that share the same Source and Destination Operational Nodes on an OV-2 diagram, based on source and destination activity assignments and associated ICOMs on associated OV-5b diagrams. For each Need Line, enter the associated IEs.
- Validate OV-5b Linkages. Insure all of the diagrams are linked all the way up to the OV-5b Context level (A-0).

9.2.2 Development Tasks

When a well-defined OV-5 architecture model is completed and stabilized, the list of Operational Nodes is identified, and all the leaf-level Operational Activities are assigned to an Operational Node, the OV-2 Operational Resource Flow Description is automatically generated using the OV-2/3 auto generation tool for SA.

9.2.2.1 Creating/Modifying the OV-2

The following are the procedural steps to generate the OV-2 Diagram:

- Run the OV-2/3 generation tool
- Click on the OV-2 icon, then the "OV-2 Wizard – [Step 1]" screen will be displayed
- Retrieve the development Encyclopedia UDL for the "UDL File" tab
- Click the "Next>" command, then the "OV-2 Wizard – [Step 2]" screen will be displayed
- To delete the existing OV-2 Diagrams and Need Lines, then a new set of OV-2 Diagrams and Need Lines will be generated; select (or check) all five optional items as follows:
    - Generate 'All' Diagram
    - Generate Parent Diagrams
    - Generate Children Diagram
    - Generate Recursive Need Lines (Need Lines that connect an Operational Node to itself)
    - Delete OV-2 Diagrams and Need Line Definitions Before Starting (Clean Run)
- Click the "Next>" command, then the "OV-02 Results" screen will be displayed
- Click the "Start" Command
- Manually adjust the Need Lines in the OV-2 Diagrams
- Add the description for each OV-2 Diagram
- Add the description for each Need Line
- To update the OV-2 Diagrams and Need Lines, select (or check) four optional items as follows:
    - Generate 'All' Diagram
    - Generate Parent Diagrams
    - Generate Children Diagram
    - Generate Recursive Need Lines (Need Lines that connect an Operational Nope to itself)
- Click the "Next>" command, then the "OV-2 Results" screen will be displayed
- Click the "Start" command

9.2.2.2 Diagram Model Coordination with Stakeholders

- Perform impact analysis where a change in other BEA models may affect the OV-2. If a change is made to OV-2 models notify the owner of the related models.
- Map all OV-2 Nodes to OV-5 leaf-level Operational Activities.

- Verify each linkage with the appropriate SA report: OV-2 to Other Models.

9.2.2.3 Diagram/Model Cleanup

The process for completing the OV-2 models is described below. Some of these tasks can be performed concurrently and several of them actually can begin only when the other models are completed. The main tasks for this process are:

- Review, refine and modify Operational Node diagram objects. Perform an internal peer review to validate the OV-2 Operational Resource Flow Description against the most current APG Modeling Guidelines. Adjudicate and incorporate peer review recommendations and obtain approval for the OV-2 and OV-3 models.

9.2.3 Post-Development Tasks

These tasks are performed after changes to the OV-2 by the Stakeholders to ensure integration of the architecture:

- Conduct a model review to ensure that the OV-2 models adhere to the APG *modeling guidelines*, *have clean BART reports* and comply with the *OV-2 Model Checklists* (See Appendix B : B-3: Appendix B : B-3:).
- Incorporate quality control and Architecture Verification changes into the BEA OV-2 model.
- Incorporate recommendations from the peer review and obtain final approval for the OV-2 diagrams.

9.3 Modeling OV-2 Models Using SA

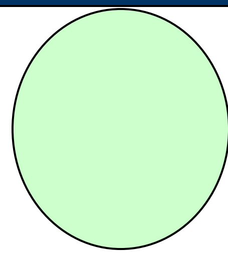
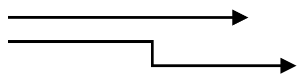
9.3.1 Modeling Conventions

The following modeling conventions shall be used to create the OV-2:

9.3.1.1 Use of Color, Size and Lines in a Diagram

Table 9-1, Modeling Guidelines for the OV-2 Diagrams

Element	Symbol	*Format*
Doc Block	Text Box: 	*Position: Upper left corner of the Diagram* *Border: Solid Black* *Fill: None* *Text:* *Color: Black* *Font: Arial*
Operational Nodes	Oval textbox:	*Border: Solid Black* *Fill:* *Color: Light Green*

Element	Symbol	*Format*
		*Text:* *Color: Black* *Font: Arial* *Size: 14 pt.* *Label: Operational Node name*
Need Lines	*Straight Arrow Connectors* 	*Color: Black* *Labels:* *Font: Arial* *Color: Black* *Size: 14 pt.*

9.3.1.2 Diagram Conventions

- There shall be at least one OV-2 diagram for each CBM, plus an integrated Enterprise-level OV-2 representing the sum of the CBM OV-2s.
- A single Need Line will be used to represent the interactions of all IEs that have a common source and destination pair of Operational Nodes.
- The OV-2 diagram should not have a border.
- Each OV-2 Diagram shall include a text description to provide a clear understandable narrative of what the Diagram portrays.
- All modeling objects should be clearly labeled and the dimensions of the label will be adjusted, if needed, to avoid truncated text.
- The Doc Block is placed in the upper left-hand corner of every diagram, with no white space above or to the left of the Doc Block.
  - The dimensions of the Doc Block should be adjusted so that the entire heading is displayed on a single line. No truncation indicators should appear on the Doc Block.
  - The Doc Block should not have a graphical comment. To confirm this in SA, right-click on Doc Block and choose Display Mode. Uncheck Graphical Comment.

9.3.1.3 Object Naming Conventions

- Valid Operational Node names are the acronyms for the five CBMs, plus two additional Nodes: “Enterprise and “External.”
- Sub-Operational Nodes names shall start with the acronym for the parent CBM, followed by the name.

- Need Line names shall consist of the sending Operational Node name or its approved abbreviation, a dash and the receiving Operational Node name or its approved abbreviation.
- IEs shall be named the same as the leaf-level ICOM. They should represent every OV-5b Input and Output linked to a leaf-level Operational Activity as an IE.

9.3.2 Modeling OV-2 Objects

9.3.2.1 Operational Nodes:

- The dimensions of the Operational Nodes may be adjusted to ensure the enclosed text is visible and avoid truncation indicators (dots).
- All Operational Nodes that must interact with a given CBM will be depicted in the corresponding OV-2 Diagram.
- The appropriate Operational Activities which are performed at that Node will be assigned to it, based on the OV-5 models.
- Operational Node labels contain the Operational Node name. Below the node name is the word "Activities," accompanied by a solid black horizontal line, and below the "Activities" label is a list of the associated Operational Activity names, in quotes. (Only display the associated Operational Activities for the central Operational Node on each diagram.)
- Each Operational Node must be associated with at least one Operational Activity.
- Each leaf-level Operational Activity is assigned to at least one Operational Node.
- Each Operational Node must be referenced by at least one Need Line.
- Using SA, fill in the "Stakeholders" tab on the definitions of each internal Operational Node with the appropriate name(s) of CBM elements that have an interest in that Node. Internal Operational Nodes are related only to the CBM for which they are named.
- Each Operational Node must be associated with a Type, either "Abstract" or "Physical."

9.3.2.2 Need Lines

- Arrows indicating the direction of information flow represent Need Lines.
- Need Lines shall be a grouping of OV-3 IEs, sharing common source and destination Operational Nodes. In SA, use the "Info. Exchanges" tab on the definition of the Need Line to link the appropriate IEs to the Need Line. Every Need Line must have at least one IE assigned.
- Every effort should be made to ensure that Need Line arrows do not intersect.
- Need Lines shall use the default SA pen width and be black in color.
- Do not display the associated IEs under the Need Lines on the OV-2 Diagram.
- A Need Line Name can exist on only two OV-2 diagrams unless it is linked to an External Operational Node or if a sub-node exists.

9.3.3 OV-2 Best Practices

This section discusses best practices, including lessons learned from previous architecture development efforts and common modeling pitfalls to avoid.

9.3.3.1 OV-2 Lessons Learned

The following lessons learned have been and serve as the basis for the OV-2 on Appendix B:

- OV-5 Models must be stabilized before OV-2 modeling can begin.
- All Operational Activities in the OV-5 model must be tagged to CBM(s).
- All leaf-level Operational Activities must be assigned to at least one Operational Node.
- All External Activities in the OV-5 models must be specified and tagged.
- All leaf-level Input and Output ICOMs must be defined and their sources or destinations must be explicitly specified.
- If the leaf-level ICOM is associated with an external activity, the external activity must be explicitly specified.
- If the ICOM is associated with a parent ICOM, the parent ICOM must be explicitly specified.
- Providing draft OV-2 and OV-3 models during OV-5 and OV-6c development will assist workshop participants to identify the impact of changes made in those models.
- All exception reports must be reviewed and resolved.

9.3.3.2 OV-2 Common Pitfalls

The following are common mistakes in the use of SA that could affect the development lessons learned have been and also serve as the basis for the OV-2 Checklist on Appendix B:

- Intersecting Need Lines.
- Ineffective use of diagram space:
  - Nodes too large or too small
  - Need Line connections unclear
  - Diagram overly dense or too spread out
- Inappropriate color coding of diagram objects.
- Not displaying Operational Activities associated with focused CBM Node.
- Inappropriately displaying IEs associated with Need Lines.
- Truncated text on Need Lines.
- Acronyms not spelled out in definitions.

10 OV-3 – Operational Resource Flow Matrix

10.1 Summary Description

This section describes the OV-3 Operational Resource Flow Matrix' its relationship to other BEA models, the development method, and the modeling guidelines to be followed.

10.1.1 Model Purpose

The OV-3 Operational Resource Flow Matrix is a tabular depiction of the information to be exchanged by identifying who will exchange the information, what information is to be exchanged, and with whom it will be exchanged in the BEA. It shows how IEs must occur and expresses why the information is necessary to be exchanged. The OV-3 matrix is developed to represent the CBM Operational Activities. All OV-3 information is associated to an IE.

10.1.2 Model Structure

The OV-3 model is depicted in a matrix. It comprises Need Line, IE, CBM(s), Source Node, Source Activity(ies), Destination Node, Destination Activity(ies), DIV Data Entity(ies), DIV Data Attributes and IE Description. The matrix contains several tabs that include the IE Matrix for each Operational Node (as shown in Table 10-2, Example of IE Tab), the we have the IE Tab that contains the information related to each IE (as shown in Table 10-2, Example of IE Tab) such as Description, DIV Data and if the IE is Available for Compliance.

Table 10-1, Example of an Operational Resource Flow Matrix

	(1)	(2)	(3)	(4)	(5)	(6)
	A	B	C	D	E	F
1	Need Line	Information Exchange	Source Node	Source Activity(ies)	Destination Node	Destination Activity(ies)
2	Enterprise - Enterprise	Information Assurance Technology Self Assessment	Enterprise	Provide Information Assurance Services	Enterprise	Perform Reporting
3	Enterprise - External	Certified Financial Statement Information	Enterprise	Perform Reporting	External	Process Treasury Information
4	Enterprise - External	Information Assurance Technology Self Assessment Report	Enterprise	Perform Reporting	External	Process Office of Management and Budget Information
5	Enterprise - External	Management Report of Cash Accounting	Enterprise	Perform Reporting	External	Process Treasury Information
6	Enterprise - External	Performance and Accountability Report	Enterprise	Perform Reporting	External	Process Office of Management and Budget Information
7	Enterprise - FM	Performance and Accountability Report	Enterprise	Perform Reporting	FM	Evaluate Program Information
8	External - FM	Accounts Payable Adjustments	External	Process Vendor or Customer Information	FM	Manage Liabilities
9	External - FM	Accounts Receivable Request	External	Process Vendor or Customer Information	FM	Establish Accounts Receivable Record Loans and Grants
10	External - FM	Additional Reconciliation Information	External	Process Vendor or Customer Information	FM	Manage Execution with Treasury
11	External - FM	Appropriation Act	External	Process Congressional Information	FM	Execute Apportionment Execute Rescission Deferrals and Cancellations Manage Baseline for Reprogramming
12	External - FM	Approved Apportionment	External	Process Office of Management and Budget Information	FM	Execute Apportionment
13	External - FM	Audit Comments	External	Process Government Accountability Office Information	FM	Manage Financial Reporting Requirement

All / Enterprise / External / FM / FM Detail / HRM / MSSM / RPILM / WSLM / IE / IA

Table 10-2, Example of IE Tab

	(1)	(2)	(3)	(4)	(5)	(6)
	A	B	C	D	E	F
1	Information Exchange	Available for BEA Complian	BEP(s)	OV-7 Data Entities	OV-7 Data Attributes	Information Exchange Description
2	Acceptance Evidence	X	AV CSE FV MV PV RPA	ELECTRONIC-ADDRESS MATERIEL-CATALOG-ITEM-COMPONENT PERSON-NAME-USAGE	Acceptance_Date [ACCEPTANCE] Acceptance_Point_Code [ACCEPTANCE] Acceptance_Quantity [ACCEPTANCE] Accounting_Classification_Structure_Assigned_Number [DEMAND-LINE-ITEM] Acquisition_Element_Price_Component_Amount [ACQUISITION-ELEMENT-PRICE-COMPONENT] Acquisition_Element_Type_Identifier [MATERIEL-ELEMENT-TYPE] Acquisition_Element_Type_Name [ACQUISITION-ELEMENT-TYPE] Activity_Identifier [ACTIVITY] Address_APO_FPO_Number [ADDRESS] Address_Post_Office_Box_Number [ADDRESS] Address_Street_Direction_Code [ADDRESS] Address_Street_Name [ADDRESS] Address_Street_Number [ADDRESS] Address_Street_Type_Code [ADDRESS] Advance_Flag_Code [TREASURY-APPROPRIATION-FUND-SYMBOL] Agency_Accounting_Identifier_Code [ACCOUNTING-AGENCY] Agency_Accounting_Identifier_Code [ACCOUNTING-CLASSIFICATION-STRUCTURE] Apportionment_Category_Code [APPORTIONMENT-CATEGORY] Asset_Type_Code [SFIS-ASSET-TYPE] Asset_Unique_Identifier [ASSET] Authority_Type_Code [AUTHORITY-TYPE] Availability_Time_Indicator [FUNDING-AUTHORIZATION-ALLOCATION-ALLOTMENT] Availability_Type_Code [TAFS-ACCOUNTING-CLASSIFICATION-STRUCTURE] BEA_Category_Indicator_Code [BEA-CATEGORY]	Documentation of the act of assuming ownership of legal title and accountability of goods tendered and services rendered, excluding Real Property Placed in Service Notification. This serves as proof that the deliverables received or services rendered have been provided fulfilling specified terms and conditions against the previously agreed upon obligating document, contract, and/or intragovernmental order between the supplier and the buyer.
	Acceptance Information		CSE		Activity_Identifier [ACTIVITY] Advance_Flag_Code [TREASURY-APPROPRIATION-FUND-SYMBOL] Agency_Accounting_Identifier_Code [ACCOUNTING-AGENCY] Apportionment_Category_Code [APPORTIONMENT-CATEGORY] Asset_Type_Code [SFIS-ASSET-TYPE] Asset_Unique_Identifier [ASSET] Authority_Type_Code [AUTHORITY-TYPE] Availability_Time_Indicator [FUNDING-AUTHORIZATION-ALLOCATION-ALLOTMENT] Availability_Type_Code [TAFS-ACCOUNTING-CLASSIFICATION-STRUCTURE] BEA_Category_Indicator_Code [BEA-CATEGORY] Borrowing_Source_Code [BORROWING-AUTHORITY] Budget_Activity_Identifier [BUDGET-ACTIVITY] Budget_Function_Code [BUDGET-FUNCTION] Budget_Function_Sub_Function_Code [BUDGET-FUNCTION-SUB-FUNCTION] Budget_Line_Item_Identifier [BUDGET-LINE-ITEM] Budget_Sub_Activity_Identifier [BUDGET-SUB-ACTIVITY]	Documentation that constructive acceptance has occurred or that ownership of legal title and accountability of goods and services has been transferred to the government buyer that allows the government supplier to record a receivable transaction.

All / Enterprise / External / FM / FM Detail / HRM / MSSM / RPILM / WSLM / **IE** / IA

10.1.3 Relationship to Other BEA Models

As illustrated in Figure 10-1, Relationships Between OV-3 and Other BEA Models, the OV-3 is related to other BEA models as follows:

AV-1	The scope of the development effort for each CBM for a development cycle, as disclosed in the AV-1, will determine if the OV-3 is affected in the release.
AV-2	All OV-3 terms with specific meaning must be included in the AV-2 Terms Definitions list. These terms must include, as a minimum, the following object types: • IE Definitions All Acronyms and Terms used in OV-3 descriptions must be listed and spelled out in the AV-2 Acronyms Definitions report.

OV-2	A Need Line in an OV-2 Diagram represents one or more IEs from the OV-3. The OV-3 provides the detailed attributes (for example, Source Node Identifier or Destination Activity) that define each IE. A Need Line must appear on at least one OV-2 diagram.
OV-5	Each Input and Output on the OV-5b connecting Operational Activities in different Operation Nodes is represented as one or more occurrences of an IE in the OV-3.
OV-6a	Although all BEA models are related through integration of the architecture, no direct relationship exists between the OV-3 and the OV-6a.
OV-6c	An IE represents a unique OV-6c Data Object. The OV-3 provides the detailed attributes (Entities / Attributes, Source Activity, and Destination Activity) that define the Data Object and its usage in the Business Process.
DIV-1 **DIV-2**	One or more Entities and/or Attributes in the Conceptual Data Model (DIV-1) or Logical Data Model (DIV-2) are linked to IEs in the OV-3, describing the IEs in terms of the Entities and/or Attributes that comprise it.
SV-6	One or more SRFs from the Systems IE Matrix (SV-6) are linked to each IE in the OV-3, showing which SRFs are required to support the IE. SRF attributes shown in the SV-6 are derived from similar attributes for related IEs in the OV-3.

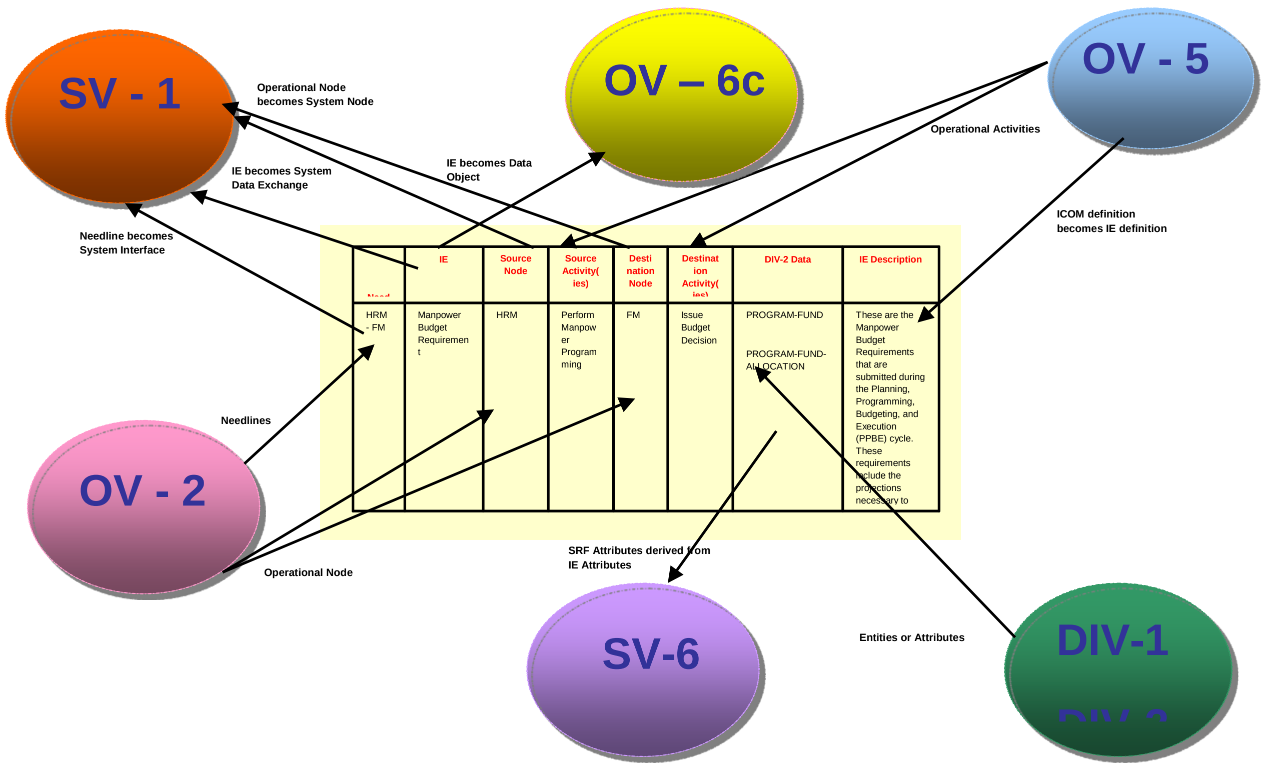


Figure 10-1, Relationships Between OV-3 and Other BEA Models

10.1.4 OV-3 Model Definitions

The following are definitions for these objects listed in each column of Table 10-1:

- **Need Line (1):** A Need Line documents the requirement to exchange information between Operational Nodes. The Need Lines are directional but do not indicate how the information transfer is to be implemented or sequenced.
- **IE (2):** The IEs between two distinct Operational Nodes.
- **Source Node (3):** Records the Operational Node that creates or otherwise provides information to the list of associated IEs.
- **Source Activity(ies) (4):** Records the Operational Activities associated with an Operational Node that create or otherwise provide information to the list of associated IEs.
- **Destination Node (5):** Records the Operational Node that requires, receives or utilizes information from the list of associated IEs.
- **Destination Activity(ies) (6):** Records the Operational Activities associated to Operational Nodes that require, receive, or utilize information from the list of associated IEs.

The following are definitions for the objects corresponding to each column of the IE Tab shown on Table 10-2:

- **IE (1):** The IEs between two distinct Operational Nodes.
- **Available for BEA Compliance (2):** Identifies if the IE is Available for Compliance.
- **CBM(s) (3):** The CBM Stakeholder associated with the IE based on the ICOM.
- **Data Entities (4):** One or more Data Entities from DIV Model that provide data to support an IE.
- **Data Attributes (5):** One or more Data Attributes from DIV Model Entities that provide data to support an IE.
- **IE Description (6):** A text description for the IE.

10.2 Developing the OV-3

This section describes the approach to develop, extend and maintain the OV-3. The OV-3 is developed in Excel as a table.

10.2.1 Pre-Development Tasks

The tasks that must be completed prior to OV-3 development and/or maintenance are:

- Identify all ICOMs associated with the leaf-level Operational Activities assigned to each Operational Node of interest to the CBM (Only Inputs, Outputs and assigned Controls are considered.)
- Define IEs using the same definitions as their corresponding ICOMs.
- Review and analyze the "from–to" Operational Activity relationship for each IE. Determine the direction and exchange of information between Operational Nodes.
- Identify DIV Model Entities and/or Attributes associated to IEs.
- Identify Information Assurance Characteristics and Attributes associated to IEs.

10.2.2 Development Tasks

Generation of the Operational Resource Flow Matrix is generated using the OV-2/3 auto generation tool for SA.

10.2.2.1 Creating/Modifying the OV-3

- Run the OV-2/3 generation tool
- Click on the OV-3 icon, then the "OV-3 Wizard – [Step 1]" screen will be displayed
- Retrieve the development Encyclopedia UDL for the "UDL File" tab
- Click the "Next>" command, then the "OV-3 Wizard – [Step 2]" screen will be displayed
- To generate the whole set of the OV-3 architecture models, select (or check) "OV-2 Diagrams(s)"
- To generate part of the OV-3 architecture models, select specific items
- Click the "Next>" command, then the "OV-2 Results" screen will be displayed
- Click the "Start" command, then the "OV-3 Wizard – [Step 3]" screen will be displayed

- Check the default option – Include DIV Model Entities (Referenced Data) in the Report
- Choose the file directory for the final Excel File location
- Click the "Next>" command. then the "OV-3 Generator Results" screen will be displayed
- Click the "Start" command

10.2.2.2 OV-3 Model Coordination with Stakeholders

- Perform impact analysis where a change in other models may affect the OV-3. If a change is made to OV-3 models, notify the owner of other models.
- Validate DIV Model Entities/Attributes to IEs.
- Validate the Information Assurance Characteristics and Attributes to IEs.
- Verify each linkage with the appropriate SA report.

10.2.2.3 Modeling Cleanup

- Review, refine and modify the Operational Resource Flow Matrix.
- Perform internal peer review to validate the OV-3.
- Incorporate peer review recommendation and obtain approval for the OV-3 model.

10.2.3 Post-Development Tasks

These tasks are performed after changes to the OV-3 by the Stakeholders to ensure integration of the architecture:

- Conduct a model review to ensure that the OV-3 models adhere to the APG, *have clean BART reports* and comply with the OV-3 Model Checklists.
- Incorporate quality control and Architecture Verification changes to BEA.
- Incorporate recommendation from the peer reviews and obtain final approval.

10.3 Modeling OV-3 Models Using SA

10.3.1 Modeling Conventions

The OV-3 is developed in tabular form, as illustrated Table 10-1, Example of an listing the object types on Section 10.1.4

10.3.2 Modeling OV-3 Objects

The following is a description of the standards and guidelines for the OV-3 objects:

10.3.2.1 OV-3 Primary Objects

- Need Line
  - The Need Line shall contain the name of the Need Line; the column cannot be blank.
  - Need Line names shall consist of the sending Operational Node name, a space, a dash, a space and the receiving Operational Node name.

  - A Need Line must be associated with at least one IE.
- IE
  - The IE column cannot be blank.
  - IE Names shall be title-case, use only approved acronyms and can use only the special character "-".
  - The IE column shall contain the name of the IE.
  - Represent every OV-5b Input and Output that is linked to a leaf-level Operational Activity as an IE.
  - The OV-5b leaf-level Inputs and Outputs are related one-to-one to the IE by a unique name.
- Source Node
  - The Source Node column cannot be blank.
  - The Source Node column shall contain the name of the Source Node for the IE from the OV-2.
  - The Source Node must contain the sending Operational Activity from the OV-5b that corresponds to the Output ICOM related to the IE.
- Source Activity(ies)
  - The Source Activity column cannot be blank.
  - The Source Activity name in the OV-3 shall be a valid leaf-level Operational Activity from the OV-5.
  - The Source Activity shall have as an Output the ICOM that corresponds to the IE.
- Destination Node
  - The Destination Node column cannot be blank.
  - The Destination Node column in the OV-3 shall contain the name of the Destination Node for the IE from the OV-2.
  - The Destination Node shall contain the receiving Operational Activity from the OV-5b that corresponds to the Input ICOM related to the IE.
- Destination Activity(ies)
  - The Destination Activity column cannot be blank.
  - The Destination Activity in the OV-3 shall be a valid leaf-level Operational Activity from the OV-5b.
  - The Destination Activity shall have as an Input the ICOM that corresponds to the IE.
- Referenced DIV-2 Entity(ies) / Attributes
  - The Referenced Data column cannot be blank.
  - The Referenced Data column shall contain one or more data Entities/Attributes from the DIV-2 that are linked to the IE.
  - Each IE in the OV-3 shall be linked to at least one data Entity/Attribute in the DIV-2.

- IE Description
  - The IE Description column cannot be blank.
  - The IE Description column shall contain the textual definition of the IE.

10.3.2.2 OV-3 Information Assurance Objects

- Information Assurance (IA) Characteristics and Attributes
  - The three IA characteristics are: Confidentiality, Integrity and Availability. Each characteristic may have one or more values and attributes associated with it, as described below
    - Confidentiality characteristic values are defined as Classified, Sensitive, or Public. For each characteristic value, there are one or more potential attributes. The full range of potential Confidentiality-related attributes is contained in the table below:

Classified	Sensitive	Public
Confidential Secret Top Secret	FOUO Unclassified Technical Data Proprietary Privacy UCNI Foreign DEA Sensitive	Approved for Release

    - Integrity characteristics values are defined as High, Medium, or Low. At each level of value, there are four potential attributes; the full range of potential Integrity-related attributes is contained in the table below:

Level of Assurance	Non-repudiation (Sender)	Non-repudiation (Receiver)	Authentication
Mandatory	Mandatory	Mandatory	Mandatory
Discretionary	Discretionary	Discretionary	Discretionary
Not required	Not required	Not required	Not required

    - Availability characteristic values are defined as High, Medium, or Low. At each level of value, there are two potential attributes; the full range of potential of Availability-related attributes is contained in the table below:

Timeliness	Level of Effort
Few or No Delays	Preemptive

Delays < 24 hrs	Specified
Delays > 24 hrs	Best Effort

- All IA characteristic and attribute values are assigned by the Stakeholders, and are used to determine the Mission Assurance Category and Confidentiality Level (MAC/CL) for each IE. These determinations indicate the MAC/CL of the system that would be expected to process, store, or transmit an IE with matching IA characteristics and attributes. DoD allows a total of nine potential MAC/CL combinations, as shown in the table below:

MAC I / Classified	MAC II / Classified	MAC III /Classified
MAC I / Sensitive	MAC II / Sensitive	MAC III / Sensitive
MAC I / Public	MAC II / Public	MAC III / Public

The combination of the Integrity and Availability characteristic values equated to the MAC level, while the Confidentiality characteristic is broken out separately. In this schema, MAC I equates to a requirement for HIGH Integrity and HIGH Availability, MAC II equates to HIGH Integrity and MEDIUM Availability, MAC III equates to LOW Integrity and LOW Availability.

For additional information concerning IA Characteristics, their values and their attributes refer to the "IA Methodology Implementation Guidance". The details of this methodology may be updated in ongoing workshops with stakeholders.

Table 10-3 reflects the IA Tab from the OV-3 Matrix that contains the Information Assurance characteristics of the IEs

Table 10-3, Example of Information Assurance Tab

IE	Confidentiality	Confidentiality Attribute 1	Confidentiality Attribute 2	Integrity	Level of Assurance	Non-repudiation (Sender)	Non-repudiation (Receiver)	Authentication	Availability	Timeliness	Level of Effort	MAC Level
Acceptance Evidence	Publicly Released	Approved for Release		Basic	Not Required	Not Required	Not Required	Not Required	Basic	> 24 Hours	Best Effort	MAC III/Public
Acceptance Information												
Apportionment												
Approved Apportionment												
Approved Payment Request	Sensitive	Unclassified Technical Data		Basic	Not Required	Not Required	Not Required	Not Required	Basic	> 24 Hours	Best Effort	MAC III/Sensitive
Award Order Requirement												
Awarded Agreement												
Awarded Contract	Sensitive	Unclassified Technical Data		Basic	Not Required	Not Required	Not Required	Not Required	Basic	> 24 Hours	Best Effort	MAC III/Sensitive

The information assurance (IA) characteristics and attributes associated with this IE (IE) were developed as part of a pilot project and are for reference only. They are provided to assist DoD Program Managers' with determining which of the DoD's information security controls baselines are appropriate to their information systems in light of the consequences that could result from the unauthorized disclosure, modification, or loss of availability of the IE or the information system. The IA characteristics and attributes associated with this IE are not intended to be used by Investment Review Boards (IRBs) or certification authorities (CAs) as a definitive checklist for controls assignments, or to dictate implementation solutions. The assignments provided in the architecture are for reference and represent typical IA attribution. Actual attributes may be higher or lower depending on business or system context.

10.3.3 OV-3 Best Practices

This section discusses best practices, including lessons learned from previous architecture development efforts and common modeling pitfalls to avoid.

10.3.3.1 OV-3 Lessons Learned

The following lessons learned have been and serve as the basis for the OV-3 on Appendix B:

- The OV-5 Operational Activity Model must be stabilized across all CBMs.
- All Operational Nodes must be identified.
- All leaf-level Operational Activities must be assigned to Operational Nodes.
- An IE must be assigned to each Input or Output ICOM associated with a leaf-level Operational Activity.
- All new or modified leaf-level Input or Output ICOMs (IEs) must have associated source and destination Operational Activities.
- All External Operational Activities must be identified and labeled as "Process…. Information."
- External Operational Activities must be linked to ICOMs via an ICOMs property sheet.
- All OV-5 Operational Activity diagrams must be balanced.
- The linkages from the Context Diagram to the leaf-level diagrams must be consistent and well defined.
- All exception reports must be reviewed and resolved.

10.3.3.2 OV-3 Common Pitfalls

The following are common mistakes in the use of Excel that could affect the development lessons learned have been and also serve as the basis for the OV-3 Checklist on Appendix B:

- ICOMs in the OV-5b Diagrams are not well defined or linked across different diagrams.
- ICOM Arrows are not touching Operational Activity boxes; the Auto generation tool will look for all the ICOMs touching an Operational Activity.
- IE descriptions contain special characters such as a carriage return code or acronyms not in the AV-2.
- DIV Model Data Columns blank or contain special characters or acronyms not defined in the AV-2.
- IA characteristic and attribute assignments are not assigned to leaf-level IEs.
- IE IA characteristic and attribute assignments are not in agreement with their analogous MAC/CL assignments.

11. SV-1 – Systems Interface Description

11.1 Summary Description

This section describes the System Interface Description, its relationship to other BEA models, the development method, and the modeling guidelines to be followed.

The SV-1 System Interface Description model depicts System Nodes, the systems resident at these nodes and System Interfaces needed to implement the automated IEs referenced by the OV-2 Operational Nodes and corresponding Need Lines.

11.1.1 Model Purpose

The System Interface Description is model to display the relationships of Systems to Systems Functions, Systems to System Nodes, and the interfaces between Systems. The SV-1 is a bridge between the Operational View and Systems View of the architecture. The SV-1 System Nodes are linked to the OV-2 Operational Nodes, the Systems Resource Flows on the system interfaces are linked OV-3 IEs, and Systems Interfaces to Need Lines in the OV-2, thus integrating the operational view business requirements and systems view that supports these requirements.

11.1.2 Model Structure

The SV-1 model is depicted as a set of diagrams It comprises System Nodes, System Entities, and System Interfaces, as described in the following sections. For the BEA, individual SV-1 diagrams are developed for each CBM and for the External Node. Each CBM has a System Node that represents a set of systems and functions performed by DoD for a specific line-of-business. The External Node represents non-DoD systems and federally mandated systems that interface with the CBM systems. In each SV-1 diagram, an oval in the center of the diagram represents the focus CBM or External System Node. All other peripheral System Nodes, CBM or External, that exchange information with the focus System Node are presented on the CBM-specific SV-1 diagram. The focus System Node includes the CBM identified Enterprise Systems and Family of Systems along with their supporting system interfaces. Each peripheral system node will only include systems that interface to the focus node. The System Interfaces on each model depict both internodal and intranodal exchanges of information with the focus node in support of Business Capabilities shown in the Operational Views.

Figure 11-1, SV-1 Model for Real Property and Installations Lifecycle Management CBM, is an example of an SV-1 Systems Interface Description model for the Real Property and Installations Lifecycle Management (CBM).

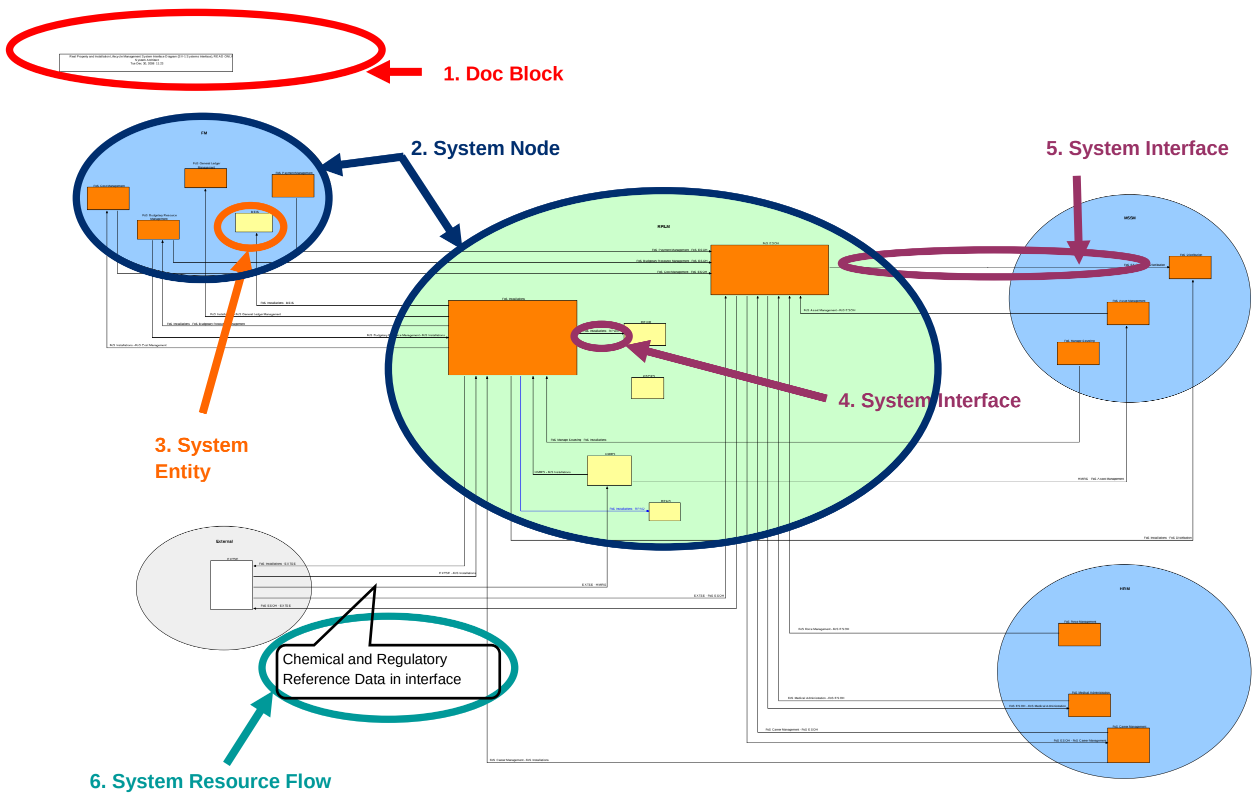


Figure 11-1, SV-1 Model for Real Property and Installations Lifecycle Management CBM

Individual SV-1 diagrams are developed for each CBM and the External node. The objects used to represent the SV-1 model are numbered as shown in Figure 11-1. The main features of this diagram are:

- **Doc (title) block (1)** is located in the upper left corner of the diagram. The title block contains the diagram name and is named after the focus CBM ("Real Property Lifecycle Management"), type "(SV-1 Systems Interface)" and last modification date.
- **System Nodes (2)** are the large oval shapes in the diagram that are named after the CBM. The SV-1 diagram name represents the focus System Node.
- **System Entities (3)** are the rectangles contained within each System Node. They represent the DoD CBM enterprise or Family of Systems, and external systems including federally mandated systems.
- **System Interface (4), (5)** are the directional lines between the System Entities. They represent intranodal communications between systems within a System Node (4) and internodal communications between systems across System Nodes (5).
- **System Resource Flows (6)** represent a collection of system Data Elements that System Functions produce or consume. In the BEA implementation of the SV-6, information assurance and performance characteristics of the exchange are not provided.

11.1.3 Relationship to Other BEA Models

As illustrated in Figure 11-2, Relationships Between SV-1 and Other BEA Models, the SV-1 is related to other BEA models as follows: This graphical model describes systems and interconnections providing for, or supporting, both DoD warfighting and business functions. The SV-1 associates systems resources to the business requirements in the OV models. These system resources are based on the Operational Activities and facilitate the automated portions of the IE among Operational Nodes.

AV-1	The scope of the development effort for each CBM for a development cycle, as disclosed in the AV-1, will determine if the SV-1 is affected by the release.
AV-2	All SV-1 terms with specific meaning must be defined in the AV-2 Terms Definition list. These terms must include, as a minimum, all object types included in the deliverable. All acronyms used in the SV-1 system interface descriptions must be listed and spelled out in the AV-2 Acronym Definitions report.
OV-2	In the BEA, System Nodes in the SV-1 are directly derived from the Operational Node of the OV-2. This is to clearly show that the systems contained in that System Node are required to support the Operational Activities performed at the corresponding Operational Node. Similarly, one or more System Interfaces in an SV-1 have a corresponding Need Line in an OV-2, thus showing the relationship between information flows and system data dependencies. Each OV-2 Need Line is comprised of IEs with their associated characteristics. As OV-2 Need Lines are comparable to SV-1 System Interfaces, IEs on those Need Lines are comparable to the Systems Resource Flows on the System Interface.
SV-5	The SV-5 links Business Capabilities and their Operational Activities to the System Functions and System Entities that supports them. These links are reviewed in SV-1 development to ensure the SV-1 interfaces are valid.
SV-6	The SV-6 describes the detailed characteristics of the Systems Resource Flows assigned to the System Interfaces on the SV-1.
StdV-1	The standards are mapped to the system entities on the SV-1. In future releases the standards may be mapped to the resource flows and interfaces.

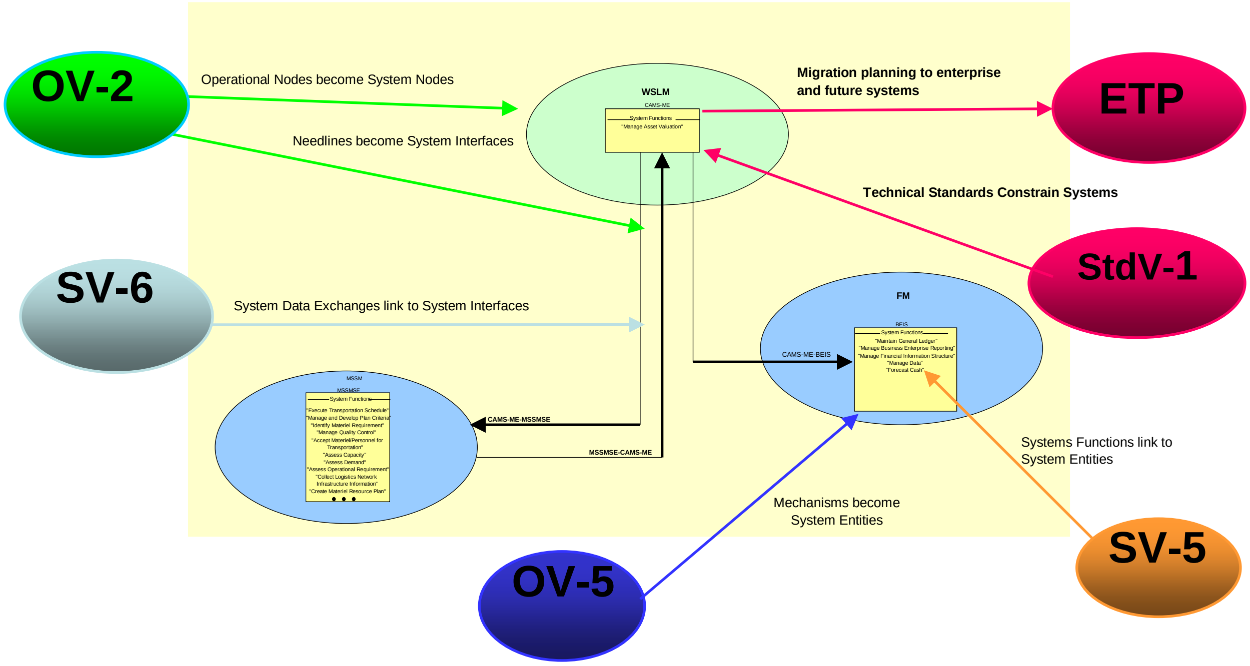


Figure 11-2, Relationships Between SV-1 and Other BEA Models

Note: Planned steps for either migrating the current suite of systems to a more efficient suite, or evolving a current system towards a future implementation are presented in the ETP. These future systems are identified as System Entities on the SV-1 and are detailed in the System Migration Summary Spreadsheet, of the ETP. Once an enterprise system is fully implemented it is removed from the ETP or the Congressional Report but will remain within the BEA architectural models until a replacement is identified.

11.1.4 SV-1 Model Definitions

The following are definitions of the key elements contained in the SV-1:

1. **System Node:** A BEA System Node represents one or more systems that work together within a Core Business Mission to support the automated portion of the business requirements described in the Operational View.

2. **System Entity:** System Entities represent DoD systems and their key external system connections. In the BEA, System Entities are identified as being enterprise systems, a Family of Systems (FoS) or external systems. An enterprise system is a system that has been identified as the standard across the DoD. The Family of Systems is a set or arrangement of independent systems that can be arranged or interconnected in various ways to provide different CBM capabilities. A FoS is used in the BEA to show connections between component and enterprise systems. An external system represents a system that is not in the BEA but that does interface with systems within the BEA.

3. **System Interface:** System Interfaces represent the resource flows between System Entities.

4. **System Function:** System Functions are the actions the system takes to transform data input into a data output in accordance to the Business Rules. It supports the automated portion of Operational Activities.

5. **Systems Resource Flows (SRF):** Systems Resource Flows represent a collection of system Data Elements that System Functions produce or consume.

11.2 Developing the SV-1 Models

This section describes the approach to develop, extend and maintain the SV-1. The SV-1 is developed in System Architect as a diagram.

The SV-1 model development begins concurrently with the development of the OV-5 and OV-2 models and continues after the completion of these models. The development is done in collaboration with the Stakeholders and, when necessary, DoD Program Managers who are responsible for the enterprise systems. Pre-development sessions and formal workshops are held with Stakeholders to identify and define system functionality represented by Enterprise Systems in the BEA.

The System View represents the business requirements in the Operational View models. The SV model analysis is conducted to ensure, for example, that OV business requirements represented as IEs become system resource flows placed on system interfaces.

During the pre-development period, the Stakeholders are provided with worksheets to collect information about their enterprise systems from the Program Managers and their community of interest. During development this information is analyzed, along with changes to Operational View models, to create the SV-1 models.

11.2.1 Pre-Development Tasks

The tasks that must be completed prior to SV-1 development and/or maintenance are:

1. The CBM representative is provided a copy of the existing SV-1 diagram and definitions. The CBM representative is responsible for identifying the CBM mission thread, defining information and data standards needed to implement the threads and defining the System Functions (automations) that systems provide in support of those threads.

2. The CBM representative determines the appropriate enterprise systems and, if applicable, contacts the responsible Program Manager for the following information about their enterprise systems:

    - Enterprise system name and description
    - System Function(s) performed
    - Interfacing systems and definition
    - System Interfaces and SRFs with definitions
    - SRFs and associated Data Elements
    - Applicable data standards needed for implementation

3. The Program Managers provide information on their respective Enterprise System to their CBM representative. The CBM representative reviews the information and provides it to the BEA Development Team for analysis and inclusion in the BEA.

4. As part of the pre-workshop activity the BEA Development Team works with the CBM representative to ensure that any new business requirements are represented in the system requirements and that any new system requirements are also reflected in the Operational View business requirements. A tight link between the SV-1 and the OV-5, OV-2 and OV-3 is maintained to ensure that the System View is integrated with the CBM business requirements.

11.2.2 Development Tasks

The development and maintenance of the System Interface Diagram is accomplished in facilitated workshops that include Government SME participation to address content and validate results. The following subsections describe the approach used to develop the SV-1 in the BEA. Each subsection sets forth the specific tasks that must be accomplished to in each stage of the development phase. Although most of these steps are sequential, it is common to start some steps before a previous step is completed.

11.2.2.1 SV-1 Analysis Tasks

Prior to any changes, an impact analysis is conducted to assess the impact of new or revised business requirements to the SV-1. The following impact analysis tasks are performed:

1. For creation or any changes to Systems Nodes:

    - Assess impact to OV-2 Operational Node.
    - Verify that the system entities with supporting system functions within the System Node supports the operational activities resident in the Operational Node.
    - Determine if the Operational Node definition needs to be refined.
    - Verify that the OV-2s support the CBM/Program Manager provided System Interfaces and SRFs.
    - Verify that the OV-3 IEs support proposed Systems Resource Flows assignments to system interfaces in the SV-1.
    - Assess impact to OV-5.
    - Assess impact to other CBM SV-1 diagrams.
    - Verify that other SV-1 diagrams support the required System Node and System Interfaces.

2. For creation or changes to System Entities:

    - Assess impact to OV-5 models.
    - Verify that enterprise systems appear as Mechanisms on corresponding Operational Activities in the OV-5b models.
    - Assess impact to System Nodes.
    - Assess impact to other SV-1s diagrams.

- Assess impact to System Functions.
- Assess impact to Enterprise Sub-Services..
- Assess impact to CBM Stakeholders.

3. For creation or changes to System Interfaces:

- Assess impact on OV-2 models.
- Assess impact on OV-5 models.
- Assess impact to System Interface name.
- Assess impact to SRFs.
- Assess impact to IE.
- Assess impact to Data Entities.
- Assess impact to CBM Stakeholders.

11.2.2.2 Creating / Modifying SV-1 Diagrams

This section describes the approach to develop the SV-1.

11.2.2.2.1 Review the SV-1 Model for Internal Consistency

Because the OV-5, OV-2 and OV-3 are so closely linked to the SV-1, the SV-1 is completed after the OV models are stabilized. The OV and SV models must be in accord. The following rules apply for the creation of a valid SV-1 diagram.

- All SV-1 interfaces must be supported by an OV-2 Need Line.
  - If the CBM has transactions between a FoS system and an Enterprise system, the interface will be shown on the SV-1 diagram, e.g. FoS ESOH – RPAD.
  - If the CBM has transactions between Enterprise systems the interface will be shown on the SV-1 diagram, e.g. WAWF – CAMS-ME.
  - If the CBM has transactions between FoSs, the interface will be shown on the SV-1 diagram, e.g. FoS Payment Management – FoS General Ledger Management.
- For each system interface the sending system entity must reside in the system node that corresponds to the sending operational activity's operational node.
- All SRFs must have a corresponding OV-3 IE.
- For each SRF the sending system entity must share a system function with the corresponding sending operational activity and the receiving system entity must share a system function with its corresponding receiving operational activity.
- Each System Node must be associated with a corresponding Operational Node.
- All SV-1 enterprise systems must be represented by a corresponding OV-5b ICOM mechanism.

11.2.2.2.2 Create the SV-1

The tasks the must be completed to produce an SV are:

1. Create a new diagram or open an existing diagram.

2. The following procedures are used for creating the various elements of the SV-1:

   - To create System Nodes:
     - Analyze OV-2 Operational Nodes
     - Create a System Node for each CBM
     - Define System Node
     - Map System Node to OV-2 Operational Node
   - To create System Entities:
     - Name System Entity
     - Define System Entity
     - Assign System Entity to System Node
     - Assign System Functions. Note: System functions do not display on the SV-1 diagrams, but are included in the HTML visualization of the system entities.
     - Designate as an Enterprise Service(s) (enterprise services only)
     - Assign CBM Stakeholders (enterprise systems only)
     - Ensure all enterprise systems are represented as mechanisms in the OV-5b and link to the ETP.
     - Ensure that system functions are only assigned to decomposed leaf level operational activities on OV-5b diagrams.
     - FoS is not required to be a mechanism on the OV-5b leaf level operational activities.
     - FoS represents federation touch points to the CBM mechanism on the OV-5b leaf level operational activities.
   - To create SRFs:
     - Name SRF after corresponding IE. Note: There is a many to one relation from SRFs to IEs. If only one SRF exists, the name will be the same as the IE. If multiple SRFs are linked to the IE different SRF names will be used.
     - If the IE name ends in "information" the last word of the SRF will be changed to "data", e.g. IE - Disbursing Information, SRF – Disbursing Data.
     - Define SRF by using the definitions of the corresponding IE or select manual and enter the new definition.
     - Assign CBM Stakeholder
     - Assign Data Elements
     - Link SRFs to IEs. Ensure that the SRF has a corresponding IE and the IE is resident of a Need Line that corresponds with the system nodes for the system interface that carries that SRF.
   - To create System Interfaces between systems:

- Name System Interface. The abbreviation of the source system entity and destination system entity is concatenated to form the system interface name.
- Define System Interface.
- Assign SRFs to System Interfaces. SRF's must be placed in system interfaces as described in "Link SRFs to IEs" above.

11.2.2.2.3 Update the SV-1

Following analysis of any changes to the OV models, existing SV-1 content shall be updated to reflect any impact of these changes. This may require creation or update of System Nodes, System Entities, SRFs or System Interfaces to the SV-1 model. For example, the addition of a new leaf-level output ICOM in the OV-5b will require the creation of a new System Resource Flow.

The tasks that must be completed to update the SV-1 are:

1. Changes to Operational Activity ICOMs:
    - If a new leaf-level input or output ICOM is added to an OV-5b, check to see if an IE was created. If there is an IE, identify the Need Line in the OV-2 where the IE is associated. Identify the System Interface that maps to the Need Line. Determine if there is an existing SRF that maps to the IE. If the SRF exists, map it to System Interface. If not, and the IE is to be automated, create a new SRF and map it to System Interface.
    - If a leaf-level ICOM is deleted from an OV-5b, identify the IE that maps to the ICOM. Identify the Need Line in the OV-2 where the IE is associated. Identify the System Interface that maps to the Need Line. Identify the SRF that maps to the IE. Delete the SRF from the System Interface.
    - If the leaf-level ICOM definition has been revised, identify the supporting SRF, review the definition and revise as necessary.
2. Changes to Operational Activity:
    - If a leaf level Operational Activity is added to an OV-5 diagram, check to determine if an existing System Function may support the activity. If there is a System Function, review definition and revise as necessary. Otherwise, create new System Function to support the Operational Activity if it is to be automated.
    - If a System Mechanism is added to a leaf-level operational activity on any OV-5b diagram, check to see if the system exists. If not, create a new System Entity and add System Functions associated to the Operational Activity.
3. Changes to Operational Node
    - Assess impact to System Node: Determine if System Node exists or one has to be created. If it exists, verify that the definition supports the Operational Node and revise as necessary. If the node does not exist, create node on each CBM specific SV-1 based on revisions to the OV-2 model.
4. Changes to Need Line on the Operational Node:

- If a Need Line is deleted, identify the System Interface that maps to the Need Line and delete.
- If a Need Line is added, determine if an existing System Interface maps to the Need Line. Create a new System Interface if there is not an existing System Interface, provided the Need Line represents an automated exchange.
- If an unexpected System Interface appears on the SV-1, an analysis of the OV models will be required to determine the corrective action that must be made to the OV or SV-1 models.

5. Changes to IE:

   - If an IE is added, identify the Need Line where the IE will be added. Determine if there is an existing SRF or if a new SRF needs to be created. Link the new SRF to the System Interface.
   - If an IE is deleted, identify the System Interface that maps to the Need Line where the IE is being deleted. Delete the SRF from the System Interface.

11.2.3 Post-Development Tasks

These tasks are performed after changes to the SV-1 by the Stakeholders to ensure integration of the architecture.

1. When a SV-1 diagram is updated, make updated diagrams available to the Stakeholders to review, identify corrections, and finalize acceptance of the model.
2. Verify all SV-1 acronyms are in AV-2.
3. Verify that all SV-1 Enterprise-level systems, Enterprise-wide systems and Component-level systems are in the Enterprise Transition Plan.
4. Review changes to the OV-2, OV-3 and OV-5 models and follow the SV-1 tasks that are mentioned in section 11.2.2.2.
5. Moreover, Quality Assurance checks are conducted to prepare for the formal quality assurance tasks conducted by IV&V group and Stakeholder review. The SV-1 Model Checklist is used to verify that the content is in accordance with the SV modeling guidelines. The major checks to ensure compliance include:
6. Spelling of all objects within the diagram is correct.
7. System Interfaces are connected to System Entities.
8. System Nodes are identified as Physical (within Abstract)
9. System Entities are within System Nodes.
10. System Nodes are associated with at least one System Entity.
11. Each System Node references at least one Operational Node.

12. Each System Entity has a definition in the System Entity dictionary.
13. Any System Entity that provides a Service must have the "Service Provider" flag in the system entity definition artifact set to "Yes".
14. Each enterprise System Entity is associated with at least one System Function. System functions do not display on the SV-1 diagrams, but are included in the HTML visualization of the system entities.
15. Each System Interface has a definition.
16. Each System Interface references at least one SRF.
17. Each SRF has a description of the data it represents that is based on the IE definition it supports.
18. Each SRF is linked to an IE.
19. At least one SRF is assigned to every System Interface.
20. Complete the SV-1 Checklist.

11.3 Modeling SV-1 Models Using SA

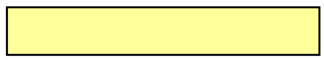
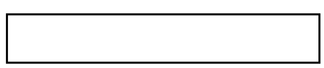
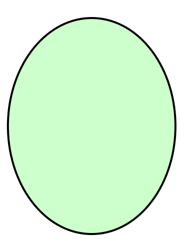
11.3.1 Modeling Conventions

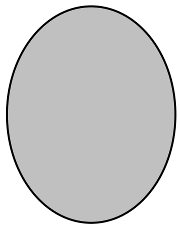
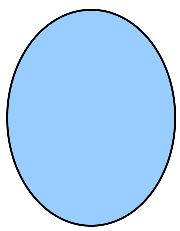
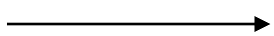
11.3.1.1 Use of Color, Size and Lines in Diagram

The following modeling conventions must be used to create the SV-1. The SV-1 diagrams use a standard color scheme, font and line size as follows:

Table 11-1, Modeling Guidelines for the System Interface Diagram

Element	Symbol	*Format*
Doc Block	Text Box:	*Position: Upper Left Corner* *Border: Solid Black* *Fill: None* *Text:* *Color: Black* *Font: Arial* *Size: Default*
System Entity	*Rectangle:*	*Border: Solid Black* *Fill: CBM Dependent* *Text:* *Color: Black* *Font: Arial 10, black*

Element	Symbol	*Format*
		Size: Default
Enterprise Level System Entity	*Rectangle:* 	*Border: Solid Black* *Fill:* Yellow boxes with a black border. The custom color settings are: Hue/Sat/Lum – 40/240/192 or Red/Green/Blue – 255/255/153 *Text:* *Color: Black* *Font: Arial 10, black* *Size: Default*
Family of Systems *System Entity*	*Rectangle:* 	*Border: Solid Black* *Fill:* Light orange boxes with a black border. The custom color settings are: Hue/Sat/Lum – 20/240/120 or Red/Green/Blue – 255/128/0 *Text:* *Color: Black* *Font: Arial 10, black* *Size: Default*
Non-DoD *System Entity*	*Rectangle:* 	*Border: Solid Black* *Fill:* White boxes with a black border. *Text:* *Color: Black* *Font: Arial 10, black* *Size: Default*
Central System Node	Elliptical*:* 	*Border: Solid Black* *Fill:* Light green fill and a black border. The custom color settings are: Hue/Sat/Lum – 140/240/192 or Red/Green/Blue – 204/255/204 *Text:* *Color: Black* *Font: N/A*

Element	Symbol	*Format*
		Size: N/A
External System Node	Elliptical*:* 	*Border: Solid Black* *Fill:* Light gray fill and a black border. The custom color settings are: Hue/Sat/Lum – 160/0/225 or Red/Green/Blue – 239/239/239 *Text:* *Color: Black* *Font:* N/A *Size:* N/A
GSA System Node	Elliptical*:* 	*Border: Solid Black* *Fill:* Light blue fill and a black border. The custom color settings are: Hue/Sat/Lum – 40/240/210 Red/Green/Blue – 255/255/191 *Text:* *Color: Black* *Font:* N/A *Size:* N/A
System Interface	*Arrow:* 	*Border: Solid Black* *Fill:* N/A *Text:* *Color: Black* *Font:* N/A *Size:* N/A

Additional guidance that applies to a system interface is provided below.

- System Interface labels will be placed, where possible, above the horizontal line and closest to either the arrowhead or 90 degree angle.
- System Interface line intersections are permissible, but should be minimized to the extent possible.

11.3.1.2 Diagram Conventions

Each SV-1 diagram shall have a Diagram Description contained within the Description block of the diagram properties that describes the purpose of the diagram, the CBM enterprise and federally mandated systems Information.

- A Doc Block representing header information for the diagram (including the diagram name and date last updated) is placed at the top center of every diagram. The Doc Block is enlarged so there are no truncation indicators (dots) indicating text is not visible. The Doc Block is a box with no fill color and has a black border.
- The SV-1 diagram shall not have a border.
- Each diagram is named after the CBM (for example, Weapon System Lifecycle Management).

11.3.1.3 Object Naming Conventions

Each SV-1 diagram uses standard object naming conventions as follows:

- The System Node name shall be the CBM acronym as used in the OV-2 to name the corresponding Operational Node, excluding the External Node and its subnodes.
- System Entity names are the official CBM acronyms, a Family of Systems of the CBM, or a federally mandated system acronym.
- The System Function form is a verb followed by a noun.
- System Entity names are used to create System Interface names. The naming convention for System Interfaces is "sending System Entity acronym" - "receiving System Entity acronym".
- Each System Interface name shall only use approved acronyms, non-plural and use no special characters except "-".
- The SRF names shall be provided by the CBM representatives. If they are not provided, the name of the IE that the SRF is linked to shall be used.
- An IE ending with "Information" will link to a SRF with the same name ending with "Data".

11.3.2 Modeling SV-1 Objects

The following guidelines are used to create or modify the SV-1.

- Modeling objects shall not have truncated entries on the diagram.
- All System Node labels shall be centered at the top of the System Node border and the label should not fall outside the boundary of the ellipse.
- Each System Node name shall be title-case, use only approved acronyms, non-plural and use no special characters except "-".
- All System Entity labels should be centered at the top of the System Entity box and the label should not fall outside the box boundary.
- All System Entities must be contained within their associated System Node elliptical boundary.
- Each System Entity name shall be upper-case or title-case, use only approved acronyms, non-plural and use no special characters except "-".

- Each System Node may contain a Family of Systems Entity to represent current and future CBM systems that have not been identified for the current release of the BEA architecture.
- The central node of each SV-1 diagram will contain the enterprise systems and any CBM FoS that have been identified.
- The SV-1 peripheral nodes will only include systems that interface to the central node.
- A text display “Service Provider” will be added to any system that provides a service.
- Each System Entity must have a Parent system assigned, e.g. CCR parent would be Federal IAE and Defense Acquisition Management Information Retrieval (DAMIR) parent would be DAMIR. For a System Entity with no Parent system, use the system entity name as the Parent system, e.g. BEIS parent would be BEIS.
- The system level of a System Entity must be set to “TOC”.
- System Interface lines are not permitted to traverse intermediate System Entities. To the maximum extent possible, System Interface lines shall not cross intermediate System Nodes.
- System Interface arrows shall be black with black filled arrowheads.
- Each end of a System Interface line must be connected to a System Entity.

11.3.3 SV-1 Best Practices

This section discusses best practices, including lessons learned from previous architecture development efforts and common modeling pitfalls to avoid.

11.3.3.1 SV-1 Lessons Learned

The following lessons learned have been and serve as the basis for the SV-1 Checklists in Appendix B.

- Ensure that the SV-1 analysis occurs concurrently with OV-5 development; ensure system mechanisms are properly assigned to Operational Activities that they automate.
- Regular and early communication with other BEA Development Teams is needed to assess impact of proposed changes in other models on the SV-1. The SV team will actively participate in the pre-analysis workshops to ensure that changes in the Operational View business requirements can be properly reflected in the System View.
- Standard color coding of diagrams during the workshop is useful for participants to identify where content was added, changed or deleted. Standard color coding should be in line with that used in the BEA Compare reports.
- All exception reports must be reviewed and resolved.

11.3.3.2 SV-1 Common Pitfalls

The following common mistakes in the use of SA that could affect the development lessons learned have been and also serve as the basis for the SV-1 Checklist on Appendix B:

- Acronyms in system entity descriptions not included in AV-2.
- Provide enough time in the original schedule to incorporate late changes in the OV-2, OV-3 or OV-5 models. Often adequate time to complete impact analysis or post development tasks to modify the SV-1 is not available.

- The SV-1 diagrams are not reviewed in the HTML/SVG rendition until after model stabilization, so flaws that do not show up in System Architect, such as superfluous line segments on System Interfaces, are exposed in the web-based version of the architecture. Superfluous line segments are eliminated on the System Interfaces by using the "reduce line segment" feature in System Architect.

12. SV-4 – Systems Functionality Description

12.1 Summary Description

The primary purpose of the SV-4 is to develop a clear description of the necessary system functionality and associated data flows that are input to (consumed) and output (produced) by each system, to ensure that functional connectivity is complete (that is, that a system's required inputs are all satisfied) and to ensure that the functional decomposition reaches an appropriate level of detail.

Figure 11-1, SV-4 Systems Functionality Description Diagram, is a Data Flow Diagram (DFD) that depicts the necessary interaction of System Functions, conceptual Data Stores, and external users and systems. SV-4 depictions represent the conceptual actions of a system in support of Operational Activities and related IEs by showing the existence of information system processing functions that operate upon linked SDEs.

SV-4 diagrams are derived from the BEA OV-5 Operational Activity models, as enhanced by the OV-6c, and related business process models.

12.1.1 Model Purpose

The SV-4 displays the functions performed by systems and the system resource flows among system functions.

The primary purposes of SV-4 are to:

- Develop a clear description of the necessary resource flows that are input (consumed) 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 Operational Viewpoint.

12.1.2 Model Structure

- There will be one or more System Functions shown on each SV-4 diagram.
- System Functions will be shown as green circles and should be arranged in 'stair step' fashion from upper left to lower right.
- Data Stores will be shown as "open" (top and bottom borders but no right or left borders) yellow rectangles.

- Externals (those representing a person or organization, for example, but not another system) will be shown as light Blue rectangles.
- External System Functions (i.e. those written within the BEA, but not assigned to the same System Entity) will be shown as light orange rectangles.
- The general flow of the SV-4 diagram should be from left to right on the diagram.
    - Objects that have outputs only should be placed on the left side of the diagram.
    - Objects that have inputs only should be placed on the right side of the diagram.

Objects with both inputs and outputs should be placed to balance the number of objects on the sides of the diagram.

12.1.3 Relationship to Other BEA Models

SV-4 is related to other BEA products as follows:

OV-5b	System Functions in the SV-4 represent required support for the business operations represented by the Operational Activities in the OV-5b. System Functions from the SV-4 are derived from and linked, through the SV-5, to the Operational Activities from the OV-5.
DIV-2	Data Attributes in the DIV-2 are linked to SDEs in the SV-4, describing the specific data elements that the SDEs represent and further defining those SDEs.
SV-1	The SV-4 defines System Functions that are executed by (and assigned to) System Entities defined in the SV-1. SDEs from the SV-4 that connect System Functions in different System Entities are assigned to System Interfaces on the SV-1.
SV-5b	System Functions in the SV-5b represent System Functions in the SV-4.
SV-6	SDEs in the SV-4 are described in detail in the SV-6. Each SDE is shown in the SV-6 with the detailed attributes that completely define its requirements.
StdV-1	Technical Standards from the StdV-1 apply to and further define the System Functions in the SV-4.
StdV-2	Timed Standard Forecasts in the Technical Standards Forecast (StdV-2) impact and may constrain System Functions in SV-4.

12.1.4 SV-4 Object Definitions

System Function definitions will focus on what the function does and how it does it (for example, how it processes inputs to produce outputs). For each definition, an introductory sentence that begins "This system function …" gives a general overview of system functionality and data flow. This is followed by a detailed description stating what caused the system function to be invoked, what it does, and what it produces. The following format will be used in the SA function description (defined as the "Purpose in the System Function definition in SA") text box:

- Each System Function and SDE must have its related CBMs identified.
- Each object definition must be grammatically and syntactically correct.

12.1.4.1 Externals

- Externals cannot directly send or receive data to/from Data Stores or to/from other Externals.
- Externals depicted must exchange data with the System Functions shown within the diagram.

12.1.4.2 System Data Exchanges

- On any diagram, if a data flow from a particular System Function triggers another System Function, the data flow should be shown as an SDE directly between the System Functions – rather than flow to and from a data store.
- If data output from a particular System Function triggers another System Function within the SV-4 diagram, the SDE should be shown directly between the System Functions and drawn black.
- Each Diagram will have at least one trigger input SDE identified in red, and one result SDE in blue.
- Each SDE that is not a trigger or primary result will be drawn in black.
- Each SDE acting as a trigger from an external entity should be drawn as a red arrow.
- Each SDE that has been identified as a primary result will be drawn as a blue arrow.
- Each SDE must have populated (non-blank) fields within the "IA Attributes" tab of the "edit" menu.
- Each SDE must have a source and a destination.
- Each SDE must have linked to it the CBMs that have a primary stake in or use that SDE.
- Diagrams will be drawn to show minimal crossings of SDE, where possible.
- An SDE will not be drawn from Data Store to Data Store or from External to External.
- An SDE may not enter and exit a System Function by the same name.
- SDEs are associated with child ICOMs from the OV-5a Operational Activity Diagram.

12.1.4.3 Data Stores

- Data Stores associated with a CBM will use the official three-letter CBM abbreviation as the Data Store name.
- Persistent data will be written to a Data Store.
- If data output from a particular System Function is needed by another System Function, but does not directly trigger that function, the SDE is shown as taking place through a Data Store.
- Data Stores shall be linked to one or more Data Elements on the DIV-2 Data Model.

12.1.4.4 Diagram Description

- Each diagram will have a description in the "diagram properties" description text box in SA. The description will consist of one or more sentences that summarize what the diagram does.

12.1.4.5 SV-4 Integration

- All SDEs that send or receive data to/from an External System Function must "match" (be of the same name) on the corresponding SV-4 diagram.
- Each External System Function found on a particular SV-4 diagram must exist as a System Function within the BEA. Additionally, the external should have the same name as the External System Function when referenced on another SV-4 diagram.
- Child System Functions will be mapped to System Entities.
- System Functions will be mapped, via the SV-5a, to Operational Activities in the OV-5.

12.2 Developing the SV-4

The SV-4 documents system functions and the system data flows between them. Such data flows may have definitions that have operational origins – that is, they may have been defined in operational (OV-5b or OV-6c) diagrams.

12.2.1 Pre-Development Tasks

This sub section will be completed in a subsequent release after a SV-4 has been developed.

12.2.2 Development Tasks

This sub section will be completed in a subsequent release after a SV-4 has been developed.

12.2.2.1 Creating/Modifying the SV-4 Models

This sub section will be completed in a subsequent release after a SV-4 has been developed.

12.2.3 Post-Development Tasks

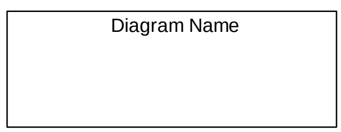
This sub section will be completed in a subsequent release after a SV-4 has been developed.

12.3 Modeling the SV-4 Using SA

12.3.1 SV-4 Modeling Conventions

- A Doc Block representing header information for the diagram (including the diagram name and date last updated) is placed in the top center of every diagram with no white space above or to the left of the Doc Block. The Doc Block will be enlarged so there are no truncation indicators (dots) indicating text is not visible and so that the Doc Block size is the entire width of the diagram. The Doc Block is a box with no fill color and a black border. There will not be a Comment line at the bottom of the Doc Block.
- Each SV-4 diagram will have a Diagram Description contained within the Description block of the diagram properties that describes how to step through the diagram from trigger to result.
- The SV-4 diagram should not have a border

Table 12-1, Modeling Guidelines for SV-4

Element	Symbol	Format
Document Block		Position: Upper Left Corner Border: Solid Black Fill: None Text: Color: Black Font: Arial Size: Default
External Interface Object	Rectangle This sub section will be completed in a subsequent release after a SV-4 has been developed	Border: Solid Black Fill: Blue Text: Color: Black Font: Arial Size: Default

Element	Symbol	Format
BEA Interface Object	Rectangle This sub section will be completed in a subsequent release after a SV-4 has been developed	Border: Solid Black Fill: Orange Text: Color: Black Font: Arial Size: Default
System Function	Circle This sub section will be completed in a subsequent release after a SV-4 has been developed	Border: Solid Black Fill: Green Text: Color: Black Font: Arial Size: Default
		Border: Solid Black Fill: Gold Text: Color: Black Font: Arial Size: Variable
Resource Flow – Common System Data Exchange	This sub section will be completed in a subsequent release after a SV-4 has been developed	Line: Solid Black Text: Color: Black Font: Arial Size: Default

Element	Symbol	Format
Resource Flow – Primary Result/Output System Data Exchange	This sub section will be completed in a subsequent release after a SV-4 has been developed	Line: Solid Blue Text: Color: Black Font: Arial Size: Default
Resource Flow – Trigger System Data Exchange	This sub section will be completed in a subsequent release after a SV-4 has been developed	Line: Solid Red Text: Color: Black Font: Arial Size: 10
Data Store	Double Parallel This sub section will be completed in a subsequent release after a SV-4 has been developed	Line: Solid Blue Fill: Yellow Text: Color: Black Font: Arial Size: Default

12.3.2 Modeling SV-4 Objects

12.3.2.1 Diagram Conventions

Diagram naming should be: name of the System Function System Function Diagram, such as: "Maintain General Ledger System Function Diagram."

12.3.2.2 Object Naming Conventions

- Each External will be named using a noun or noun phrase and will be named consistently.
- Each External representing an "external" system function will be named using the System Function name.
- Each object name will be spelled correctly.

- Each object name will be stated using the singular form (no plurals).
- Each object will have a concise and intuitive name. (Every definition should be self-contained and understandable in its own right, not requiring additional definitions in order to be understood.)
- Each System Function will be named using an active verb phrase (for example, Manage Requirement).
- Only approved abbreviations, those contained in the accepted acronym list in the BEA AV-2, will be used in object names.
- No special characters ("*," "/," and "&") will be used in object names.
- Object names will use Title Case; that is a combination of upper and lower case.
- Object names will not use future tense.
- SDEs and Data Stores will be named using noun phrases.

12.3.3 SV-4 Best Practices

This section discusses best practices, including lessons learned from previous architecture development efforts and common modeling pitfalls to avoid.

12.3.3.1 SV-4 Lessons Learned

This sub section will be completed in a subsequent release after a SV-4 has been developed.

12.3.3.2 SV-4 Common Pitfalls

This sub section will be completed in a subsequent release after a SV-4 has been developed.

13 SV-5a – Operational Activity to System Function Traceability Matrix

13.1 Summary Description

This section describes the Operational Activity to System Function Traceability Matrix, their relationship to other BEA models, the development method and the modeling guidelines to be followed.

13.1.1 Model Purpose

The Operational Activity to System Function Traceability Matrix depicts the relationships between the Operational Activities in the OV-5 Activity Models and the System Functions.

The DoDAF SV-5a, hereafter termed the SV-5, is used to meet a program requirement to link Business Capabilities, Operational Activities and System Functions. The Enterprise-level systems identified by each Stakeholder are shown on the SV-5 matrix, where the Enterprise-level system supports an Operational Activity/Business Capability, and is aligned with a specific System Function.

13.1.2 Model Structure

The SV-5 model is depicted as matrices. It comprises and relates System Functions to Operational Activities across the BEA Business Capabilities. For each matrix cross area or intersection the related Enterprise Systems are presented. There can be many Operational Activities related to a single Business Capability. The following four matrices are included in the BEA:

- SV-5a Operational Activity to System Function Traceability Matrix
- SV-5a System Function to Operational Activity Traceability Matrix
- SV-5 Traceability Matrix with Related Systems and Capabilities
- SV-5a System Function to Operational Activity Definition

The following figure is an example of the SV-5a Operational Activity to System Function Traceability Matrix:

SV-5a Operational Activity to Systems Function Traceability Matrix

Operational Activity	System Functions
Account for Time, Absence, and Labor	Collect Time, Absence, and Labor Data from External Source Maintain Time, Absence, and Labor Data via User Interface Process Time, Absence, and Labor Data Produce Time, Absence, and Labor Data Report Provide Time, Absence, and Labor Workflow Send Time, Absence, and Labor Data to External Destination
Account for Workforce	Produce Manpower Report Provide Manpower Workflow
Administer Assignment Action	Collect Assignment Action Data from External Source Maintain Assignment Action Data via User Interface Process Assignment Action Data Produce Assignment Action Report Provide Assignment Action Workflow Send Assignment Action Data to External Destination

Figure 13-1, Example of a SV-5a Operational Activity to Systems Function Traceability Matrix

The following figure is an example of the SV-5a System Function to Operational Activity Traceability Matrix:.

SV-5a Systems Function to Operational Activity Traceability Matrix

System Function	Operational Activities
Accept Materiel/Personnel for Transportation	Assemble and Marshal Forces Manage Inbound and Outbound Shipments Transport Materiel and Forces
Aggregate Spend Data	Manage Request and Sourcing Strategy Monitor Sourcing Execution
Assess Capacity	Conduct Logistics Business Planning
Assess Demand	Conduct Logistics Business Planning
Assess Operational Requirement	Conduct Logistics Business Planning
Calculate Supply Chain Entitlement	Calculate Supply Chain Entitlement
Chemical Management	Validate Product Hazard Data Validate Product Hazard Data Request

Figure 13-2, Example of a SV-5a Systems Function to Operational Activity Traceability Matrix

The following figure is an example of the SV-5 Traceability Matrix with Related Systems and Capabilities:

Business Capability	Conduct Program Management		Manage Acquisition Oversight Integration			Monitor Commercial Request for DoD Technology Export	
Operational Activity	Define Program	Develop Program	Conduct Acquisition Assessment	Conduct Periodic and Ad-hoc Reporting	Manage Capabilities Based Acquisition	Conduct Science and Technology	Calculate Supply Chain Entitlement
System Function							
Accept Materiel/Personnel for Transportation							
Aggregate Spend Data							
Assess Capacity							
Assess Demand							
Assess Operational Requirement							
Calculate Supply Chain Entitlement							DAI FoS Payment Management

Figure 13-3, Example of a SV-5 Traceability Matrix with Related Systems and Capabilities

The following figure is an example of the SV-5a System Function to Operational Activity Definition:

A	B	C	D	E
CBM	**Business Capability**	**Operational Activity**	**System Function**	**System Entity**
FM	Collect and Disburse	Manage Collections	Manage Collections	DAI
FM	Collect and Disburse	Manage Collections	Manage Collections	FoS Receivable Management
FM	Collect and Disburse	Manage Disbursements	Manage Disbursements	DAI
FM	Collect and Disburse	Manage Disbursements	Manage Disbursements	FoS Payment Management
WSLM	Conduct Program Management	Define Program	Manage Mission Support Requirements	FoS Defense Acquisition Management
WSLM	Conduct Program Management	Develop Program	Manage Mission Support Requirements	FoS Defense Acquisition Management
WSLM	Conduct Program Management	Develop Program	Perform Program Analysis	FoS Defense Acquisition Management
MSSM	Deliver Property and Forces	Assemble and Marshal Forces	Accept Materiel/Personnel for Transportation	FoS Distribution
MSSM	Deliver Property and Forces	Assemble and Marshal Forces	Package/Handle/ Transport Material/Personnel	FoS Distribution
MSSM	Deliver Property and Forces	Conduct Logistics Business Planning	Assess Capacity	FoS Logistics Planning
MSSM	Deliver Property and Forces	Conduct Logistics Business Planning	Assess Demand	FoS Logistics Planning
MSSM	Deliver Property and Forces	Conduct Logistics Business Planning	Assess Operational Requirement	FoS Logistics Planning

Figure 13-4, Example of an SV-5a System Function to Operational Activity Definition

13.1.3 Relationship to Other BEA Models

The SV-5 is related to other BEA models as follows.

AV-1	The scope of the development effort for each CBM for a development cycle, as disclosed in the AV-1, will determine if the SV-5 is affected by the release.
AV-2	All SV-5 terms with specific meaning must be defined in the AV-2 Terms Definition list. These terms must include, as a minimum, all object types included in the deliverable. All acronyms used in the SV-5 system function descriptions must be listed and spelled out in the AV-2 Acronym Definitions report.
OV-5	OV-5 Operational Activities are linked to the System Functions in the in the SV-5. In addition, Enterprise-level Systems are Mechanisms on the Operational Activities.
SV-1	The SV-1 System Entities and supporting System Functions match systems in the SV-5 matrix.

The Stakeholders and the BEA Development Team coordinate the components of the SV-5 to ensure integration with other BEA Models.

Business Capability	Manage Acquisition Oversight Integration			Monitor Commercial Request for DoD Technology Export
Operational Activity	Conduct Acquisition Assessment	Conduct Periodic and Ad-hoc Reporting	Manage Capabilities Based Acquisition	Conduct Science and Technology
System Function				
Perform Cross-Cutting Analysis and Reporting	DAMIR	DAMIR	DAMIR	
Perform Data Checks	DAMIR	DAMIR	DAMIR	
Perform Precedent Search				USXPORTS
Perform Program Analysis	FoS Defense Acquisition Management	FoS Defense Acquisition Management	FoS Defense Acquisition Management	
Perform Reporting				USXPORTS

Figure 13-5, Example of a SV-5 Traceability Matrix with Related Systems and Capabilities

13.1.4 SV-5 Model Definitions

The following are definitions of the key elements contained in the Operational Activity to System Function Traceability Matrix.

1. **Business Capability:** Each capability represents the ability to execute a specific course of action. It can be a single business enabler or a combination of business enablers (e.g. business processes, policies, people, tools, or systems information) that assist an organization in delivering value to its customer.

2. **System Entity:** System Entities represent DoD systems. In the BEA, these systems are identified as being enterprise systems, a Family of Systems or external system. An enterprise system is a system that has been identified as the standard across the DoD. The FoS is a set or arrangement of independent systems that can be arranged or interconnected in various ways to provide different capabilities. An external system represents any system not in the BEA but interfaces with systems within the BEA.

3. **Operational Activity:** An action performed in conducting the business of an enterprise. This is a general term that does not imply a placement in a hierarchy or a timing sequence (for example, 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).

4. **System Function:** System Functions are the actions the system takes to transform data input into a data output in accordance to the Business Rules. It supports the automated portion of Operational Activities.

13.2 Developing the SV-5 Models

This section describes the approach to develop, extend and maintain the SV-5. The SV-5 is developed in an Access Database with a web-based front-end using data exported directly from System Architect and is produced as spreadsheet.

A single enterprise matrix represents the SV-5 for all CBMs. The SV-5 provides an integrated architecture depiction of the relationships of Operational Activities to System Functions, and enterprise systems to both Operational Activities and System Functions. Through the mapping of Business Capabilities to Operational Activities, there is an indirect link between System Functions and Business Capabilities.

13.2.1 Pre-Development Tasks

The tasks that must be completed prior to SV-5 development and/or maintenance are:

1. Verify that BEA enterprise systems are included in the ETP.

2. Identify leaf-level Operational Activities that are on OV-5 diagrams.

3. Collect System Function information from CBM team leads for each leaf-level Operational Activity on an OV-5 diagram that may be automated.

4. Collect System Function information from CBM team leads for each enterprise system and FoS (optional for FoS).

5. Verify the mapping of System Functions to leaf-level Operational Activities with the CBM team leads.

6. Identify changes in the OV-5 and SV-1 that may impact the SV-5.

7. Identify System Functions that are performed, in whole or part, by an enterprise or FoS systems.

8. Verify mapping of System Functions to enterprise and FoS systems.

13.2.2 Development Tasks

The development and maintenance of the Operational Activity to System Function Traceability Matrix is accomplished in facilitated workshops that include Government SME participation to address content and validate results. The following subsections describe the approach used to develop the SV-5 for the BEA. Each subsection sets forth the specific tasks that must be accomplished to in each stage of the development phase. Although most of these steps are sequential, it is common to start some steps before a previous step is completed.

13.2.2.1 Creating/Modifying the SV-5 Models

This section describes the approach to develop the SV-5. To create the SV-5 matrix, the following tasks are performed:

1. Analyze definitions of Operational Activities and associated ICOMs.

2. Identify and create System Functions that will support leaf-level Operational Activities on OV-5 diagrams. System Functions must consume the inputs and produce the outputs of the Operational Activity in which they are linked.

3. Analyze System Function name and definition provided by CBM team leads to ensure they support the leaf-level Operational Activities on the OV-5 diagrams.

4. Analyze System Function name and definition provided by the CBM team leads and assign to enterprise systems and FoS to ensure these systems execute the functions assigned.

5. Verify enterprise systems as Mechanisms to Operational Activities.

6. Verify enterprise system shares at least one System Function with the Operational Activity where it appears as a mechanism.

7. Verify mapping of Operational Activities to Business Capabilities.

8. Link System Functions to Operational Activities

   - Use the SV-5 System Function to Operational Activity matrix browser to link a System Function to an Operational Activity with an "X".
   - To ensure accurate generation of this SV-5 report, detailed connections between the OV and SV models are necessary. For a system name to appear at the intersection of System Function and an Operational Activity instead of the "X":

- The Business Capability must be assigned to the Operational Activity of interest.
- The Operational Activity must show the CBM as a Stakeholder (stakeholder tab).
- The Operational Activity must have the system as a mechanism – otherwise displays the system entity as underlined. (FoS appears in the intersection as underlined.)
- The Operational Activity must be assigned to a system function AND the SE must be assigned to at least one of the system functions supporting the Operational activity.

9. The actual generation of the SV-5 matrix is automated. The general application resides at this web location:

   http://bta-beatools.btads.bta.mil/SV5Generator.asp ***(To be updated for BEA 9.0)***

To further explain step 8, The SV-5 report generator displays three types of system function/operational activity relationships with a system entity in the report legend. The relationships between these components are explained below.

System Entity (SE) for both Operational Activity (OA) and System Function (SF) − The SV-5 report shows the System Entity displayed at the SF/OA intersection in **Bold** when the System Entity appears as a mechanism ICOM on the subject OA in the SV-5 matrix intersection and the OA and SE have been assigned to the same system function present in the intersection.

System Entity for System Function Only -- The SV-5 report shows the System Entity displayed at the SF/OA intersection as underlined when the System Entity is not assigned as a mechanism; the System Entity is assigned to the system function in the intersection; AND the same system function is also assigned to the OA (through the SV-5 matrix). (The SE is not a mechanism on the OA. *This is where a FoS appears because it is not a mechanism in release 7.0*. FoS appears as underlined.)

System Entity for Operational Activity Only -- The SV-5 report shows the System Entity displayed at the SF/OA intersection in *italics* when the System Entity appears as a mechanism on the subject OV-5b Operational Activity AND the OA is assigned to the system function in the intersection AND the SE is **NOT** assigned to the SF at the intersection.

An ICOM mechanism within the context of the SV models means that the System Entity representing the ICOM does the work/creates the outputs of that OA. *If the SE does not share a system function with the OA, it is logically impossible for the SE to do the work it claims.* Remedies include:

A. Removing the SF from the OA (SV-5 matrix)
B. Adding the SF to the System Entity (SE dialogue box).
C. Removing the SE from the OA.

13.2.2.2.1 SV-5 Matrix Coordination with Stakeholders

The tasks that must be completed to coordinate with the Stakeholders are as follows.

1. Coordinate with Stakeholders

   - Print copy of SV-5 for Stakeholders to review proposed changes and confirm linkages.

2. Coordinate with the Enterprise Transition Plan team.
    - Compare enterprise systems in the ETP with enterprise systems in SV-5 for consistency.
    - Print copy of SV-5 for the ETP team to review proposed changes and provide comments.
3. Coordinate with BEA SV-1
    - Ensure that any System Function assigned to an enterprise system in the SV-5 is properly aligned to the corresponding System Entity on the SV-1 diagrams.
    - Ensure that System Functions are only assigned to leaf-level Operational Activities on the OV-5 diagrams.
    - Ensure that System Functions are not assigned to any OV-5 Node Tree only Operational Activities.
    - Ensure that any enterprise system that links a System Function to an Operational Activity is a Mechanism on that Operational Activity in the OV-5b diagram (with the exception of a Family of Systems).
    - If an enterprise system is not a Mechanism for an OV-5b Operational Activity, it should not link that Operational Activity to any System Function in the SV-5.
4. Coordinate with BEA AV-2
    - Ensure that all enterprise system acronyms are expanded correctly in the Long Name attribute of the System Entity and in the AV-2.

13.2.3 Post-Development Tasks

These tasks are performed after changes to the SV-5 by the Stakeholders to ensure integration of the architecture.

1. Ensure that Names of Operational Activities, System Functions, Business Capabilities and enterprise systems are current and accurate.
2. Add rationalization to the SV-5 to explain any anomalies.
3. Manually correct system entity assignments in the matrix when operational activities and systems support multiple capabilities that belong to different CBMs. The system entity should only reside under the capability that is owned by the system CBM.

13.3 Modeling the SV-5 Models Using SA

13.3.1 SV-5 Modeling Conventions

The following modeling conventions shall be used to create the SV-5.

The BEA creates a single SV-5 matrix that represents all CBMs and as well a single diagram for each CBM. A Microsoft Excel worksheet containing one ALL View and six CBM tabs represents the final SV-5 model. The SV-5 model includes Business Capabilities, and identified Enterprise Systems where a System

Function supports an Operational Activity. Please note that the ALL View SV-5 is very large and can only be legibly viewed in electronic format. A non-graphic representation of the SV-5 is also created in Excel to facilitate sorts and definition review of the SV-5 components and is titled, "SV-5 Systems Function to Operational Activity Matrix Definitions". This spreadsheet includes definitions for the following objects: CBM stakeholder, Business Capability, Operational Activity, System Function, and System Entity. The first worksheet tab displays the SV-5 in a vertical format and the remaining tabs are displayed in this order; Business Capability, Operational Activity, System Function and System Entity.

13.3.1.1 Modeling Use of Color, Size and Lines in the SV-5 Matrix

The following color, size and line conventions are used to create the SV-5.

- The title of the model is "SV-5b" and appears above the matrix in the spreadsheet header.
- The first row of the SV-5 matrix lists the Business Capabilities and the second row lists the corresponding Operational Activities. Thus, each column of the SV-5b matrix represents a Business Capability and Operational Activity intersection.
- The first column of the SV-5 lists the System Functions and so each row represents a specific System Function.
- The Operational Activities are sorted across the x axis by CBM so that enterprise systems that are relevant to a specific CBM are clustered near each other on the SV-5.
- The System Functions are sorted alphabetically along the Y axis of the matrix.
- All data cells on the ALL View tab are gray.
- In addition to the standard ALL View, the SV-5 matrix is also created for each CBM. In the CBM only diagram System Functions and Operational Activities are both sorted by CBM so that enterprise systems that are relevant to a specific CBM are clustered near one another.
- No color fill color is used on the individual CBM or All View diagrams other than the gray in the ALL View.

13.3.1.2 Matrix Diagram Conventions

The following matrix diagram conventions are used to create the SV-5.

- The names of Business Capabilities, Operational Activities and System Functions in the SV-5 should be consistent throughout the encyclopedia.
- All updates to the SV-5 matrix are implemented through System Architect's Matrix browser, SV-5 System Function to Operational Activity file.
- The SV-5 matrix only contains relationships between leaf-level Operational Activities, Business Capabilities, Enterprise-level systems, including FoS systems and System Functions.
- Every System Function mapped to an Operational Activity shall be reflected in the SV-5 properties tab of the Operational Activity.
- Each System Function must be mapped to at least one leaf level Operational Activity on an OV-5 diagram.
- The SV-5 should list all leaf-level Operational Activities from the OV-5 diagram that have supporting system functions.

13.3.1.3 Object Naming Conventions

The following guidelines are used to create or modify the SV-5.

- Enterprise system names are the official acronyms of the CBM Enterprise Systems and are expanded in the Long Name attribute of the System Entity and in the AV-2.
- The form of the System Function, Operational Activity and Business Capability is a verb followed by a noun.
- The first word and all the main words in System Function, Operational Activity and Business Capability names should have initial capitals, and all the joining words should be left in lower case.

13.3.2 Modeling SV-5 Objects

13.3.3 SV-5 Best Practices

This section discusses best practices, including lessons learned from previous architecture development efforts and common modeling pitfalls to avoid.

13.3.3.1 SV-5 Lessons Learned

The following lessons learned have been and serve as the basis for the SV-5 in Appendix B:

- Ensure the settings on the SV-5 generator are set properly prior to SV-5 generation.
- Be sure to reformat the Operational Activity row on the diagrams from horizontal to vertical.
- Generate the SV-5 AFTER the OV-5 is stabilized.

13.3.3.2 SV-5 Common Pitfalls

The following are common mistakes in the use of the SV-5 report generator that could affect the development lessons learned have been and also serve as the basis for the SV-5 model checklist in Appendix B:

- Making changes after the SV-5 Change Requests have been voted upon.
- Making any SV-5 changes by hand. These changes are frequently demanded by Stakeholders but are not supported by the data in the operational views and system views.

In general, ensure that all the system function, operational activity, system entity, stakeholder, and business capability links among the operational views and system view are correct.

14 SV-6 – Systems Resource Flow Matrix

14.1 Summary Description

This section describes the Systems Resource Flow Matrix (SV-6) architecture model and its relationship to other BEA models, the development method and the modeling guidelines to be followed.

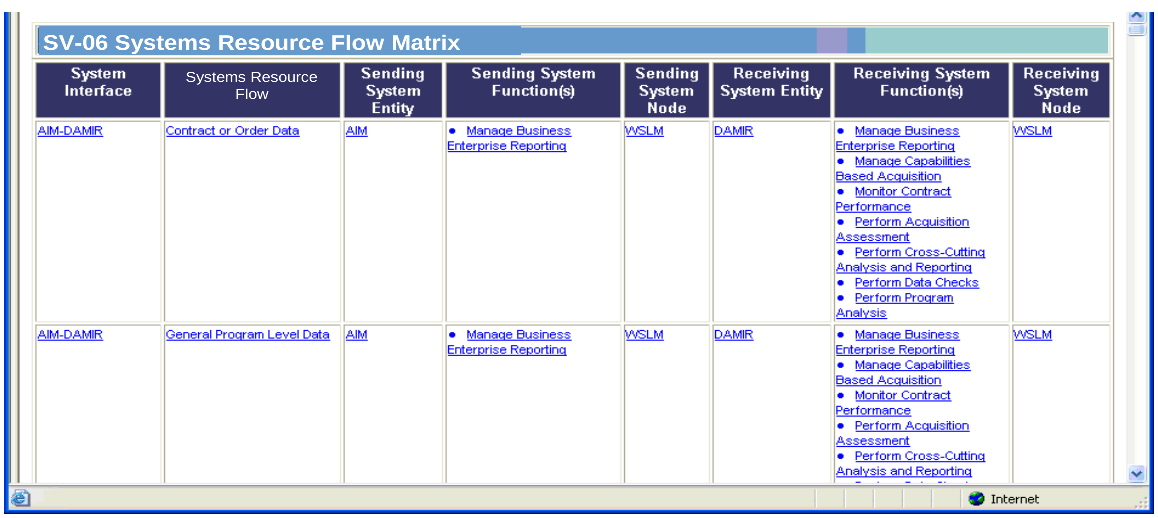
14.1.1 Model Purpose

The SV-6 Systems Resource Flow Matrix is the model which provides details about the Data Elements exchanged between systems and the characteristics of that resource flow for the BEA. The SV-6 is used to show the Systems Resource Flows on a system interface. The SV-6 relates to, and is derived from, the OV-3 and SV-1. The operational characteristics in the OV-3 IE matrix are used to develop the corresponding SRF attributes in the SV-6. Each SRF exchanged is related to the System Entity from the SV-1 that produces or consumes information.

The SV-6 discloses details about system interfaces for each CBM within their BMA. For each SRF, the source and destination System Entities, source and destination Systems Functions, and the sending and receiving System Nodes are provided. The BMA may use this information to ensure that the information on the SV-6 matches business requirements.

14.1.2 Model Structure

The SV-6 model is a matrix report. For the BEA it is generated through the SA reporting tool. It provides the information in tabular form for each SRF linked to System Interfaces in the SV-1. It comprises the System Interface, Systems Resource Flow, System Entity, System Function and System Node. Figure 14-1, Example of an SV-6 Systems Resource Flow Matrix represents a sample SV-6 Systems Resource Flow Matrix.


SV-06 Systems Resource Flow Matrix

System Interface	Systems Resource Flow	Sending System Entity	Sending System Function(s)	Sending System Node	Receiving System Entity	Receiving System Function(s)	Receiving System Node
AIM-DAMIR	Contract or Order Data	AIM	• Manage Business Enterprise Reporting	WSLM	DAMIR	• Manage Business Enterprise Reporting • Manage Capabilities Based Acquisition • Monitor Contract Performance • Perform Acquisition Assessment • Perform Cross-Cutting Analysis and Reporting • Perform Data Checks • Perform Program Analysis	WSLM
AIM-DAMIR	General Program Level Data	AIM	• Manage Business Enterprise Reporting	WSLM	DAMIR	• Manage Business Enterprise Reporting • Manage Capabilities Based Acquisition • Monitor Contract Performance • Perform Acquisition Assessment • Perform Cross-Cutting Analysis and Reporting	WSLM

14.1.3 Relationship to Other BEA Models

As illustrated in Figure 14-2, Relationships Between SV-6 and Other BEA Models, the SV-6 is related to other BEA models as follows:

AV-1	The scope of the development effort for each CBM for a development cycle, as disclosed in the AV-1, will determine if the SV-6 is affected by the release.
AV-2	All SV-6 terms with specific meaning must be defined in the AV-2 Terms Definition list. These terms must include, as a minimum, all object types included in the deliverable. There are no object types used in the SV-6 but rather the SV-6 uses the names of the elements listed in the section above. The SRF definitions must be listed and defined in the AV-2. All acronyms used in the SV-5 descriptions must be listed and spelled out in the AV-2 Acronym Definitions report.
OV-3	One or more SRFs described in the SV-6 are linked to each IE in the OV-3, showing which SRFs support an IE.
SV-1	The SV-1 provides the information needed to generate the SV-6 and is shown in Appendix B.
SV-5	The SV-5 provides the system function links need to generate the SV-6.
StdV-1	The standards are mapped to the system entities on the SV-1. In future releases the standards may be mapped to the resource flows and interfaces.

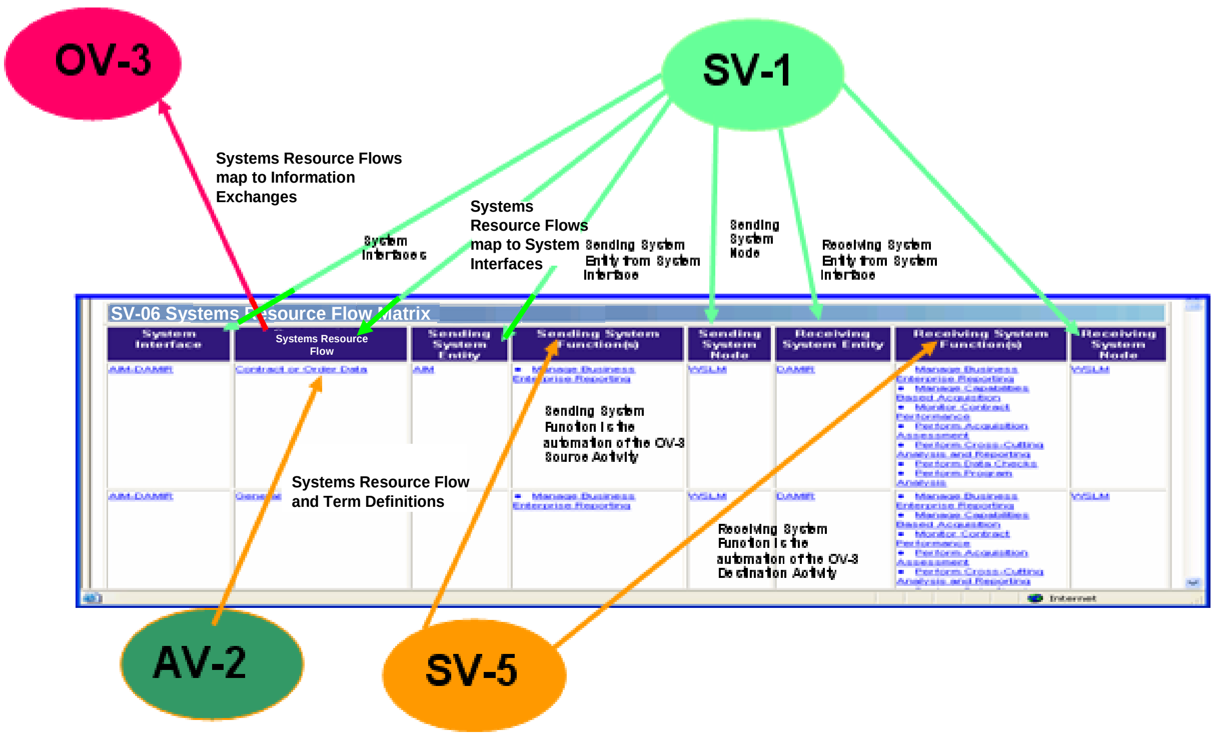


Figure 14-2, Relationships Between SV-6 and Other BEA Models

14.1.4 SV-6 Model Definitions

The following are definitions of the key elements contained in the Systems Resource Flow Matrix

1. **Sending/Receiving System Node:** A BEA System Node represents one or more systems that work together within a CBM to support the automated portion of the business requirements described in the Operational View.

2. **System Entity:** System Entities represent DoD systems. In the BEA, these systems are identified as being enterprise systems, a FoS or external system. An enterprise system is a system that has been identified as the standard across the DoD. The Systems FoS is a set or arrangement of independent systems that can be arranged or interconnected in various ways to provide different capabilities. An external system represents any system not in the BEA but interfaces with the BEA.

3. **System Interface:** System Interfaces represent the resource flow between System Entities.

4. **Sending/Receiving System Function:** System Functions are the actions the system takes to transform data input into a data output in accordance to the Business Rules. It supports the automated portion of Operational Activities.

5. **System Resource Flow:** System Resource Flows represent a collection of system Data Elements that System Functions produce or consume. In the BEA implementation of the SV-

6, information assurance and performance characteristics of the exchange are not currently provided.

14.2 Developing the SV-6

14.2.1 Pre-Development Tasks

This section describes the approach to develop, extend and maintain the SV-6. The SV-6 is developed in System Architect as a table by following these steps:

1. Review the CBM's AV-1 and BIP to understand the impact of the planned body of work on the SV-6.
2. Identify outstanding issues from the previous release.
3. The BEA Development Team meets with Stakeholders to review the issues and develop solutions.
4. After the OV-2, OV-3, OV-5 are stabilized additional issues are identified and reviewed.
5. The Stakeholder requiring a change pre-coordinates the solution with the affected Stakeholders.
6. These tasks three, four and five are repeated in development tasks.

14.2.2 Development Tasks

The development and maintenance of the Systems Resource Flow Matrix is accomplished in facilitated workshops that include Government SME participation to address content and validate results. The following subsections describe the approach used to develop the SV-6 in the BEA. Each subsection sets forth the specific tasks that must be accomplished in each stage of the development phase. Although most of these steps are sequential, it is common to start some steps before a previous step is completed.

Figure 14-1, shows all the components of the SV-6 report.

14.2.2.1 Creating/Modifying the SV-6 Model

This section describes the approach to develop the SV-6

14.2.2.1.1 SV-6 Development Approach

These tasks are completed to create the SV-6 in the BEA.

1. Validate the generated SV-6 against the SV-1 and the OV-3.
2. When the SV-6 is generated, provide copies of the generated matrix to the Stakeholders to review, identify corrections and for acceptance of the model.
3. Meet with Stakeholders to review comments.
4. Perform Impact Analysis to determine impact to OV and SV models.

5. Initiate Change Request process to address identified issues.

6. Upon completion of changes to impacted OV and SV models, regenerate the SV-6 matrix.

   - System Interface
   - Systems Resource Flow
   - Sending System Entity
   - Sending System Function(s)
   - Sending System Node
   - Receiving System Entity
   - Receiving System Function(s)
   - Receiving System Node

14.2.3 Post-Development Tasks

These tasks are performed after changes to the SV-6 by the Stakeholders to ensure integration of the architecture.

1. Compare information on generated SV-6 to SV-1:

   1.a. Validate that each System Node in the SV-1 is represented in the SV-6.

   1.b. Validate System Entities:

   - Ensure that every System Entity in the SV-6 is associated with the correct System Node.
   - Validate that the System Function to System Entity linkage on the SV-6 matches the content of the SV-1.

   1.c. Validate System Interfaces:

   - Validate that only inter-nodal System Interfaces are in the SV-6 matrix.
   - Validate the sending and receiving System Entities against the content of the SV-1.
   - Ensure that SRFs map to System Interfaces based on the content of the SV-1.

2. Compare information on generated SV-6 to OV-3:

   2.a. Validate source and destination System Nodes against source and destination Operational Nodes.

   2.b. Validate System Interfaces against Need Lines.

   2.c. Validate SRFs against IEs.

   2.d. Validate Data Elements against the IEs

14.3 Modeling the SV-6 Model Using SA

The SV-6 is an automated report that is generated from the SV-1 and OV-3.

14.3.1 SV-6 Modeling Conventions

There is no modeling of the SV-6 within SA. The content for the SV-6 is derived from other SV models

14.3.2 Modeling SV-6 Objects

There is no modeling of the SV-6 within SA. The content for the SV-6 is derived from other SV models

14.3.3 SV-6 Best Practices

This section discusses best practices, including lessons learned from previous architecture development efforts and common modeling pitfalls to avoid.

14.3.3.1 SV-6 Lessons Learned

The following lessons learned have been and serve as the basis for the SV-6 in Appendix B:

- Need regular and early communication with across BEA Development Team to assess impact of changes to the SV-1.
- Ensure that the OV-5 and OV-3 models are stabilized prior to SV-6 development.
- Ensure that the SV-6 is regenerated whenever there is any change to the SV-1 model. The models are directly linked so a change in the SV-1 will result in a change to the SV-6.
- Stakeholders should thoroughly review Systems Resource Flows and Attributes for each System Interface to ensure they are properly aligned to the Operational View models.
- Ensure that all exception reports have been reviewed and resolved.

14.3.3.2 SV-6 Common Pitfalls

The following example is a common mistake that could affect the development lessons learned have been and also serves as the basis for the SV-6 Checklist on Appendix B:

- As the SV-6 exists only as a large HTML report that is generated by the build team, flaws are only exposed in the Web version of the architecture, which may not be fully reviewed until after BEA model stabilization efforts.

15 SV-10a – System Rules Model

15.1 Summary Description

This section describes the SV-10a System Rules Model and its relationship to other BEA models, the model-development method, and the modeling guidelines used for development of the SV-10a. The SV-10a System Rules and the OV-6a Business Rules are the same in construct. The primary difference is that the SV-10a System Rules denote "physical" constraints while the OV-6a Business Rules denote "logical" constraints; thus, each rule is mapped to different BEA models.

The BEA is evolving to a Semantic BEA where the BEA is described in an ontology using a common language and modeling notation. World Wide Web Consortium open standards such as Web Ontology Language (OWL), Resource Description Framework (RDF) and Rule Interchange Format (RIF) are being used to describe the BEA. The RIF standard allows compound business rules to be expressed.

The BEA distinguishes between a Business Policy/Guidance and Business Rules. The distinctions are based on the following definitions of a Business Policy/Guidance and Business Rules as derived from Object Management Group's Semantics of Business Vocabulary and Business Rules v1.0 and DoDAF 2.0.

1. Business Policy/Guidance is different from Business Rules in that, Business Policy/Guidance is:
   - Less structured
   - Less discrete
   - Less carefully expressed in terms of a standard vocabulary
   - Not directly enforceable
2. Business Rules are different from Business Policy/Guidance in that, a Business Rule:
   - Uses structured language and notation
   - Is unambiguous
   - Uses a standard vocabulary
   - Is directly enforceable

The BEA contains OV-6a Operational Rules and SV-10a System Rules. Operational Rules are different from System Rules (specified in DoDAF 2.0 as SV-10a System Rules Model) in that, the Operational View "identifies what needs to be accomplished and who does it," while the System View "relates systems, services, and characteristics to operation needs." Using a data modeling analogy, Operational Rules correspond to the logical model and System Rules correspond to the physical model.

The BEA SV-10a, System Rules include a concept of "System Rule Category" to distinguish between two types of System Rules: Operative System Rules and Structural System Rules. Operative System Rules are those that constrain the dynamic aspects of an enterprise. The Operative Category of a System Rule has a Type of either Derivation or Action, both of which can potentially be violated or ignored. Structural System Rules are those that constrain the static aspects of an enterprise. The Structural Category of a System Rule has a Type of Relational or Definitional, both of which are constrains that must be imposed. Examples are presented in the following sections.

There is a difference between the definition of a System Rule and notation in which the System Rule is written. This is similar to the difference between the definition of a Process Model and BPMN. System Rule notation is a technique used to write System Rules in a consistent way such that the System Rule is atomic, unambiguous, uses a standard vocabulary, and is directly enforceable. The APG specifies *RuleSpeak*™ as the System Rule notation.

15.1.1 Model Purpose

The SV-10a System Rules Model is the set of rules that constrain the performance or implementation of systems and services. These are constraints on "how" the business functionality is implemented and "how" interoperability is achieved; consequently, the System Rules are mapped to System View models.

15.1.2 Model Structure

The SV-10a is created manually. It resides in the SA encyclopedia in the form of definitions (there are no diagrams for the SV-10a in the SA encyclopedia). Please refer to Section 6.1.4 for the required fields for the System Rules to make the definition complete. Each Stakeholder as well as each CBM can have an unlimited number of System Rules within the BEA SA encyclopedia.

15.1.3 Relationship to Other BEA Models

The SV-10a is related to other BEA models through the following:

AV-2	All SV-10a terms with specialized meaning must be defined in the AV-2 as Terms; these must include, as a minimum, all deliverable object types. All acronyms must be properly designated in the System Rule definitions and must be defined in the AV-2 as Acronyms.
SV-1/ ScvV-1	A SV-10 System Rule that constrains data structure or performance is mapped to the appropriate System Entity.
SV-4	A SV-10 System Rule that constrains a derivation is mapped to the appropriate System Function.

15.1.4 SV-10a - BEA Model Mapping

15.1.4.1 SV-10 Mapping Decision Tree

The primary difference between the SV-10a and the OV-6a is that the SV-10a System Rules denote "physical" constraints while the OV-6a Business Rules denote "logical" constraints; thus, the BEA models to which they are mapped are different. Figure 15-1 is a decision tree that guides the mapping of SV-10a System Rules to the appropriate BEA model. The decision tree includes the logic for both SV-10a and OV-6a mappings.

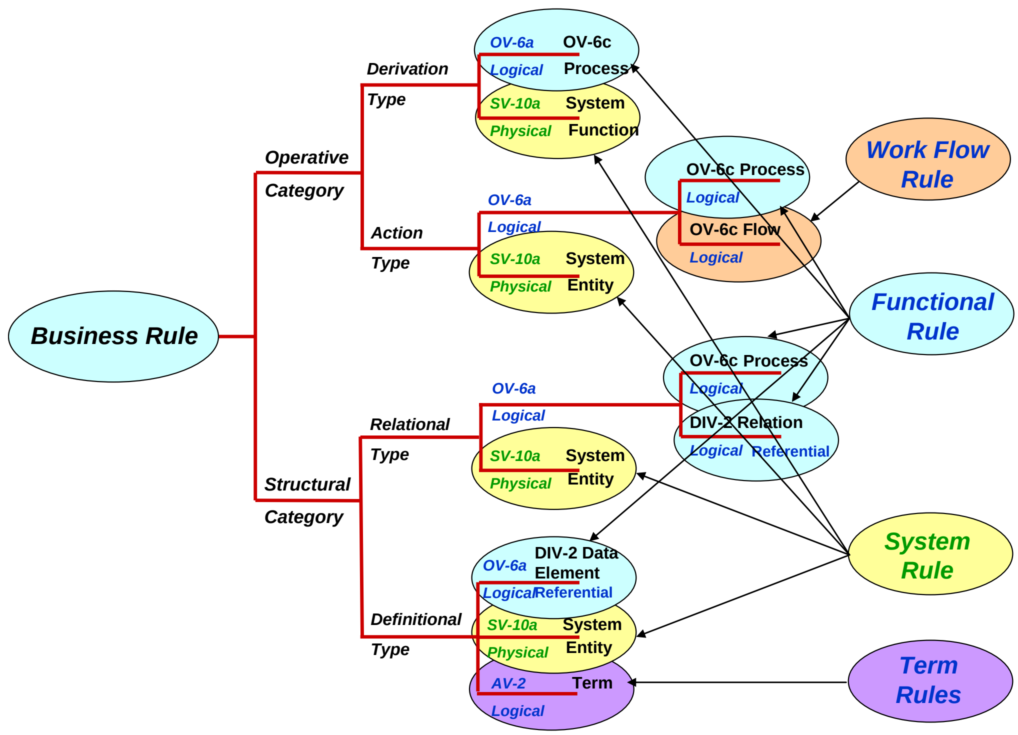


Figure 15-1, SV-10a - BEA Model Mapping Decision Tree

The SV-10a mappings are indicated by the color yellow to differentiate them from the OV-6a mappings in the color blue. There are two groups of OV-6a mappings, Business Rules and Workflow Rules. Business Rules are constraints imposed by the business activities, while Workflow Rules are constraints that sequence business processes. Rule Terms are the definitions of the terms used by the rules. The above decision tree is based on the Business Rule Mantra: "***Rules are built on Facts. Facts are built on Terms.***"

15.1.4.2 SV-10a Field Definitions

This section defines the fields that comprise a System Rule in the BEA. Included in this section are explanations and examples of correct field input. DoDAF terms are not repeated.

Figure 15-2 illustrates the eleven (11) data fields and structure of a System Rule. Five data fields are text entry. Two of the data fields are single list selections. Four of the data fields are multiple value list selections.

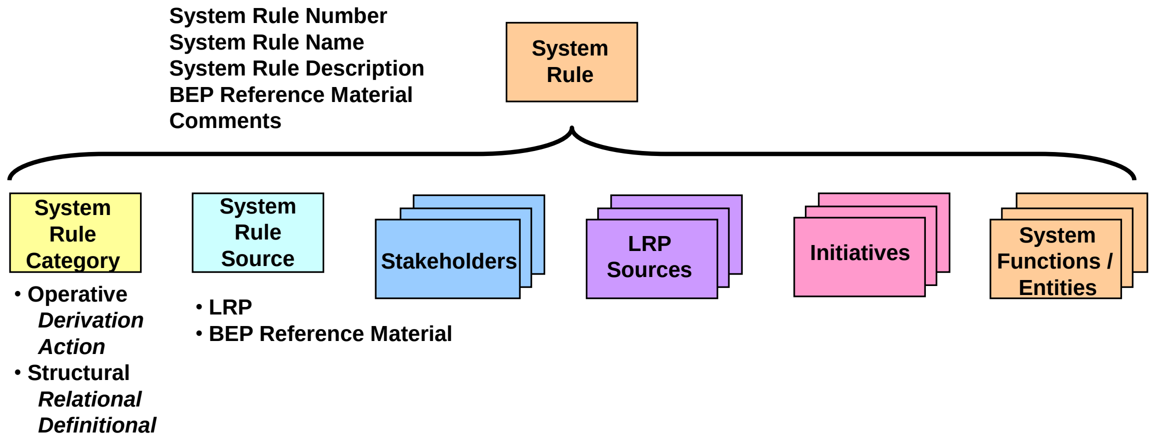


Figure 15-2, Data Fields and Structure of a System Rule

1. **System Rule Number:** A text entry field that is a unique number given to a System Rule for identification purposes. The unique number is manually assigned. The following are examples of correct System Rule Numbers:
   - 1000
   - 185444
2. **System Rule Name:** A text entry field that is a unique name given to a System Rule for identification purposes. The System Rule Name is limited to 80 characters and consists of concatenated terms separated by underscores. A term can be either alphabetic or numeric. The System Rule Name must begin with an alphabetic term. Each alphabetic term must start with an uppercase letter followed by lowercase letters or numbers unless the term is a capitalized acronym. The System Rule Name is used to map to other architecture artifacts such as a System Entity. An example of a System Rule Name is: "Payment_Request_Approval_1".
3. **Description:** A text entry field that is the System Rule. It is a statement of constraint or permission stated in relation to a System Entity or system Function. It may be stated at any level. Examples of Descriptions are given in a following System Rule Category section.
4. **Stakeholder Reference Material:** An optional text entry field in which to store relevant System Rule information. It is recommended to store Derivation sources here but it is not required. In addition, Stakeholder teams may use the Stakeholder Reference Material field for any other information they do not wish to maintain in the comment field. For example, a Stakeholder may choose to enter in this field a list of related Operational Activities.
5. **Comments:** An optional text entry field for use of the CBM team, Stakeholder Team, or Stakeholder(s).
6. **System Rule Category:** A single list selection field that is a defined category that assists in mapping the System Rule to either a System Function or a System Entity.

- **Operative Derivation:** A System Rule that reflects dynamic aspects of a system and constrains a sequence or calculation. Operative Derivation System Rules must be mapped to System Functions. These are constraints on systems that are or will be implemented. The following is an example of a correct Operative Derivation System Rule.
  - **Operative Derivation System Rule Description:** The ***Fee Amount*** must be calculated by multiplying the ***Fee Rate*** by the number of ***Days Late***.
  - **Note:** The underscored terms in the above description denote physical data names that must be used by the implementing System Function.
- **Operative Action:** A System Rule that reflects dynamic aspects of a system and constrains a performance or output functionality. Operative Action System Rules must be mapped to System Functions. These are constraints on systems that are or will be implemented. The following is an example of a correct Operative Action System Rule
  - **Operative Action System Rule Description:** The Invoice Retrieval Window must display the beginning of the query within two (2) seconds.
- **Structural Relational:** A System Rule that reflects static aspects of a system or service. Structural Relational System Rules may result from important structural assertions graphically represented in the OV-7 Logical Data model due to BEA Compliance Assessment requirements. Structural Relational System Rules must be mapped to System Entities or a Family of Systems. These are constraints on systems that are or will be implemented. The following is an example of a correct Structural Assertion System Rule
  - **Structural Relational System Rule Description:** A ***CONTRACT*** has one or more ***CONTRACT-LINE-ITEMs***.
  - **Note:** The underscored terms in the above description denote the actual physical data names that must be used by the implementing System Entity.
- **Structural Definitional:** A System Rule that reflects physical data definitions. Structural Definitional System Rules may result from data attributes graphically represented in the OV-7 Logical Data model due to BEA Compliance Assessment requirements.
  - **Structural Definitional System Rule Description**: A ***Zip_Code*** has exactly nine numeric characters.
  - **Note:** The underscored terms in the above description denote actual physical data name that must be used by the implementing System Entity and the actual physical domain.

7. **System Rule Source:** A single value list selection field that indicates the source of the System Rule.
   - **LRP:** The origin of this type of a System Rule is a Law, Regulation, and/or Policy applicable to a particular business process and/or system.
   - **Stakeholder Reference Material:** The origin of this type of a System Rule is the result of business transformational efforts and the need to constrain, or give permission to a System Function or System Entity. This type is not directly traceable to compliance requirements.

Note: When the System Rule Source is "Stakeholder Reference Material", the "Stakeholder Reference Material" data field must be populated with the appropriate reference citation.

8. **Stakeholder:** A multiple value list selection field that designates the owner of the System Rule. The owner may be an individual CBM designation or a Stakeholder designation.
9. **Initiatives:** A multiple value list selection field that relates System Rules to initiatives and systems. It identifies the enterprise initiative that the System Rule enforces or identifies the system that the System Rule influences. The following are the valid Initiatives:
   - SFIS: Standard Financial Information Structure
10. **LRP Sources:** An optional multiple valued data field that contains a link to the specific Law, Regulation or Policy the LRP Repository. The LRP Sources are selected from a list of available LRP Sources.
11. **System Functions / Entities:** A multiple value list of valid SA System Entities or System Functions that relates System Rules to appropriate System View models.

15.1.4.3 SV-10a Input Field Capture

This section describes capturing the System Rules field values for inserts or changes to SA.

15.1.4.3.1 Input Document

The primary input document or load sheet is a spread sheet that has the following four tabs:

1. Main Load
2. LRP Sources
3. Stakeholders
4. System Entities or System Functions

The Main Load tab contains all of the above single entry fields as columns. That is, the fields for which there is only one value; such as the System Rule Definition.

The other three tabs contain the fields for which multiple occurrences are possible; for example, a System Rule can be mapped to more than one System Entity or System Function. The map between the System Rule and the other BEA artifact is the System Rule Name. The following is an example of entries in the System Entity tab. Note: These are notional values and are not actual values.

System Entity Name	System Rule Name
WAWF	Zip_Code_Domain
WAWF	AUID_Domain

The LRP Sources and the Stakeholders tabs have the same construct as the above but with each replacing the "System Entity Name" column.

This input document is also used as a "turn around" document for reviews between the SV-10a Team and the rule originator.

15.1.4.3.2 System Architect Update Process

There are two basic SA Update processes: one for a large update volume and the other for a small update volume.

A large update volume requires assistance from the SA Build Team. The SA Build Team uses the spreadsheet as input to an automated procedure.

A small update volume is manually inputted into SA by the SV-10a Team members.

15.1.4.3.3 Mapped Field Validation

The mapped field names must be correct, otherwise errors will be generated and the System Rule will not be complete. The Stakeholder Sources are few (AV, CSE, FV, MV, PV, RPA) and pose little problem. However there are many LRP Sources and System Entities and System Functions.

The SA Build Team automated load procedure automatically inserts artifacts when a match is not found. For example, an incorrect System Entity name will cause a new System Entity name to be added with that name. This is an error that must be manually corrected. Consequently, care must be taken to assure that the mapped names are valid and already exist in SA. The same applies for LRP Sources; however, this is a benefit because no manual effort is required to update the BEA LRP Source list. The LRP Sources should be verified before the load to ensure that they are valid.

15.2 Developing the SV-10a Model

This section describes the approach used by the System Rules Analyst to develop the SV-10a System Rules model. The System Rules Analyst works with Stakeholder Team and functional Subject Matter Experts (SMEs) to produce System Rules that support the business transformation. The System Rule process includes development, maintenance, and retirement of System Rules.

15.2.1 Pre-Development Tasks

In general, there are a number of ways Stakeholder representatives choose to identify System Rule concepts for inclusion into the SV-10a System Rules model. Each Stakeholder has their own process for deciding System Rules content and may request the assistance of the System Rules analyst to provide pre-analysis work. Pre-analysis work may include suggesting general guidance on how to develop content, answering any form and structure questions, or generating and analyzing SA reports from the prior BEA release. This process for System Rule creation does not require detailed analysis of the architecture. The Stakeholder Team provides the System Rules Analyst with their identified System Rule concepts mapped to an architectural object.

15.2.2 Development Tasks

The BEA development schedule must allot time for SV-10a development tasks. This period of time is referred to as the SV-10a workshop, but these tasks are performed outside of a formal workshop setting.

15.2.2.1 Refine System Rule Concept

Once the Stakeholder Team identifies the System Rule concept, its architectural mapping, and the LRP Requirement Identifier(s), if applicable, through the external input process, the System Rules Analyst must develop the concept into a form ready for functional review and refine the System Rule to meet the project's BEA/DoDAF standards.

First, the System Rules Analyst determines whether the System Rule concept already exists in the BEA. If the proposed System Rule concept does not duplicate an existing System Rule, the System Rules Analyst further analyzes the concept for accuracy and potential conflict with existing System Rules.

The System Rules Analyst then determines the System Rule Category which determines the mapping.

The System Rules Analyst develops the System Rule Name and assigns the SV-10a System Rules Model unique System Rule Numbers.

Following, the System Rules Analyst refines the System Rule concept by applying the *RuleSpeak*™ guidelines to ensure the proposed language of the System Rule meets the BEA standards. The System Rules Analyst shall use Ron Ross' BRS *RuleSpeak*™ as its technical language standard for System Rules. This is the private sector standard for what constitutes a quality System Rule.

Generally, SV-10a System Rules will be stated in *RuleSpeak*™ format. But, if appropriate to achieve the goals of Stakeholder Team, an exception may be made. However, due to their nature, Operational Derivation (formulaic) and Operational Action (not derived from a documented requirement) System Rules must be in *RuleSpeak*™ format.

The Business Rules Analyst also determines the System Rule Category Type and System Rule Source Type. The Business Rules Analyst develops the System Rule Name and assigns the SV-10a System Rules Model unique System Rule Numbers

All of the above artifacts are placed in an MS Excel spread sheet referred to as the "Load Sheet" that is used to add the System Rules to the SV-10a System Rules Model. This "Load Sheet" is sent to the appropriate Stakeholder Team for a preliminary review.

In the last step, the System Rules Analyst forwards the proposed System Rules to the System Rule Team Lead for technical review. The System Rule Team Lead verifies the System Rules for technical compliance. The System Rules Team Lead returns any System Rules that do not pass the technical review to the System Rules Analyst for further refinement.

In extreme cases, a System Rule may be accepted with a technical violation as an exception, provided the System Rules Team Lead documents why the System Rule was accepted with the violation on the SV-10a Model Review Checklist.

15.2.2.2 SV-10a System Rules Model Coordination with Stakeholders

After the System Rules Analyst completes the technical review with the System Rules Team Lead, each System Rule is passed back to the Stakeholder Team for a functional review. The Stakeholder Team verifies that the proposed System Rule conveys the same idea as the original System Rule concept. When analysis is complete, the Stakeholder Team communicates the results of the functional review to the System Rules Analyst. If the proposed System Rule passes the functional review, the Stakeholder Team returns it to the System Rules Analyst for pre-load verification. If the proposed System Rule does not pass functional review, the Stakeholder Team returns it to the System Rules Analyst for further work. In that case, the System Rules Analyst uses the Stakeholder Team comments to refine the System Rule concept. Normally, the Stakeholder Team and System Rules Analyst work closely to achieve a consensus on a functionally and technically solid System Rule.

15.2.2.3 SV-10a System Rules Load Preparation

After the validity of the System Rules has been agreed, the System Rules are ready for the load into the BEA. The BEA Development Team creates the load sheet, which is an MS Excel spreadsheet that lists the System Rules and all associated fields discussed in previous subsection **15.1.4** Definitions. This load sheet is submitted to all of the Stakeholders for approval through the designated BEA Workshops. After the load sheet has been approved, the BEA Development Team submits the load sheet and a copy of the approved CR to the SA Build Team. The BEA Development Team can manually input the approved System Rules into System Architect. This is normally done when the System Rule volume is less than twenty System Rules.

If a System Rule does not successfully pass the Stakeholders final approval stage, the BEA Development Team works with the Stakeholders towards final approval.

After the System Rules are loaded into SA, the BEA Development Team verifies that the load sheet is correctly represented in the BEA. This step must wait for the SA Build Team to perform an Encyclopedia update so that the BEA Development Team can conduct the Post-Build Verification on the latest Encyclopedia build.

15.2.2.4 Creating/Modifying the SV-10a Models

To ensure that the System Rules remain valid, the BEA Development Team follows a maintenance and retirement process. The Stakeholders notify the BEA Development Team that there is a change in the BEA affecting existing System Rules. The BEA Development Team identifies the System Rules and makes any necessary adjustments in System Architect. Next, the BEA Development Team analyzes the architecture for potential changes to the existing processes and other linkages in System Architect. Finally, the BEA Development Team validates the above Process for continued support of the business transformation. The following sections discuss the maintenance process in detail:

15.2.2.4.1 Identify System Rules

Either the Stakeholders or the BEA Development Team may initiate an analysis to determine if changes are needed. A number of circumstances may trigger a change to some aspect of the System Rule and/or its artifacts. Below are typical (but not an inclusive list of) triggers:

- System Objective change
- Process Step change

- IE / Data Object / SRF change
- Requirement change
- LRP Source change

Note: If a LRP Source Identifier or description already mapped to a System Rule changes in the LRP Repository, the LRP team will notify the BEA Development Team and/or the Stakeholders. The BEA Development Team will manage the change in System Architect.

15.2.2.4.2 Analyze Potential Changes

Once the change is identified, the BEA Development Team conducts analysis to determine any potential impacts on the BEA. The analysis includes impact to other architectural models, existing System Rules and/or requirements.

15.2.2.4.3 Validate System Rule Changes

The BEA Development Team hands off the System Rule for technical review to the BEA Development Team. After completing technical review, the Development Team works with the appropriate Stakeholders for functional approval.

15.2.2.4.4 Update System Architect

Upon approval, the BEA Development Team makes the changes in SA or creates a load sheet to present to the Build Team for automated loading.

15.2.3 Post Development Tasks

As with the System Rule maintenance process, a number of Events may trigger the retirement process, such as:

- A System Objective change or elimination
- BPM Process modification or elimination
- LRP Source change or elimination
- The Stakeholders notify the BEA Development Team that a System Rule needs to be retired. The BEA Development Team, working closely with the Stakeholders, documents the reason for retirement, and then the System Rule is retired from the SA encyclopedia. Finally, the appropriate validation steps are executed to ensure that the System Rule was actually retired.

Note: The Stakeholders review each System Rule identified for retirement, verifying whether it is appropriate to retire the System Rule for all Processes. If a System Rule, identified for retirement, still has a valid mapping to another Process Step, the BEA Development Team retains the System Rule, removing only the link to the retired Process Step.

The following sections discuss the retirement process in more detail:

15.2.3.1 Identify Obsolete System Rule

The Stakeholders are responsible for identifying a System Rule that needs to be retired. In addition, if an architecture object deletion or change is the trigger, the Stakeholders identify whether each instance of the

System Rule must be deleted or just the mapping between the System Rule and the architecture object being changed.

15.2.3.2 Retire System Rule

The BEA Development Team removes the identified System Rule from SA. The BEA Development Team also identifies to the LRP team a retired System Rule that has a LRP Identifier associated with it. If the System Rule is still valid, and possesses valid mappings to other architecture objects, the BEA Development Team removes only the prescribed mapping.

15.2.3.3 Validate System Rule Retirement

The BEA Development Team SME conducts the appropriate quality assurance checks. These checks validate the appropriate retirement of a System Rule and prevent the creation of orphan Processes and Data Elements.

15.3 Modeling the SV-10a Model using SA

15.3.1 SV-10a Modeling Conventions

Guidelines that assist in the identification and definition of System Rules are:

- Each System Rule must have a unique System Rule Name. The System Rule Name format is described in Subsection 15.1.5 of this document.
- Each System Rule must be assigned a unique System Rule Number by which it can be identified.
- Each System Rule must have a unique Description. The Description is the actual System Rule. Subsection 15.1.5 of this document provides details on the construction of System Rules.
- Each System Rule must be classified into only one of two System Rule Categories: Operative or Structural.
- Each System Rule must be classified into only one of two System Rule Sources: "LRP" or "Stakeholder Reference Material".
- Each Business Rule with a Source of "LRP" must be associated with one or more LRP authoritative sources. The LRP Source Identifier is the unique identifier from the LRP Repository.
- Each System Rule must have a Stakeholder. The Stakeholder may be an individual Stakeholder designation or a CBM designation.
- Each System Rule includes an optional Comment field for use of the Stakeholder(s).
- Each System Rule includes an optional Stakeholder Reference Material field. This field is used to store information the Stakeholder considers relevant for the System Rule. If the System Rule Source is "LRP", then this field is populated with the source from which the System Rule is derived. For instance, the Stakeholder may include the title of a Requirement Document

Regulation and/or policy with a Hypertext Markup Language (HTML) link to the original document. The Stakeholder may also include a list of related Operational Activities.

- Each System Rule may be optionally associated with an integration Stakeholder. The Stakeholder designation must be from the approved Stakeholder acronym list.
- Every Structural System Rule must have a mapping to a System Entity.
- Every Operative System Rule must have a mapping to A System Function.
- Each LRP Source Identifier must be associated with a unique Identifier from the LRP Repository.

15.3.2 Modeling SV-10a Objects

- A clear understanding of the SV-4 Systems Functionality model and DIV models must come before SV-10a development.
- Stakeholder SV-10a content that affects other Stakeholders should be socialized before the workshop.
- Prior to the workshop, the fields associated with a Business Rule, like Source Type and OV-6c Event Trace Diagram mapping, should be socialized with the Stakeholder Coordinator and Team Lead to ensure these mappings are captured, in addition to the content of the System Rule.
- Ensure acronyms are spelled out in AV-2.
- A System Rule that has a source of a "LRP" must have a LRP Source Identifier associated with the Business Rule.
- A System Rule must be mapped to an object in the SV-4 Systems Functionality model or a DIV Model.
- Ensure All Exception reports have been reviewed and resolved.

15.3.3 SV-10a Best Practices

This section discusses lessons learned from previous SV-10a architecture development and mention best practices while developing the SV-10a architecture model.

15.3.3.1 SV-10a Lessons Learned

- A clear understanding of the System Functions and System Entities.
- Stakeholder SV-10a content that affects other Stakeholders should be socialized before the workshop.
- Prior to the workshop, the fields associated with a System Rule should be socialized with the Stakeholder Coordinator and Team Lead to ensure these mappings are captured, in addition to the content of the System Rule.
- Ensure that acronyms are entered in the AV-2.

- Ensure that terms are entered in the AV-2.
- A System Rule that that has a source of a "LRP" must have a LRP Source Identifier associated with the System Rule.
- A System Rule must be mapped to either a System Function or a System Entity.

15.3.3.2 SV-10a Best Practices

- Avoid ambiguity
- Avoid embedding term definitions in System Rules (Use AV-2 to define terms)
- Simplify as much as possible by not over-explaining or adding unrelated information
- Make sure subject is quantified and singular (e.g., by using the word "each")
- Use the "If - Then" statement construct for conditional System Rules
- For the System Rule Category of Operative - Derivation, make the result of a computation the subject of the System Rule
- Ensure that the logical structure of the System Rule makes sense
- Review all domain specific System Rules for contradictory or redundant statements.
- Validate the existence of SA mapped names using spread sheet macros such as VLOOKUP. This helps reduce errors in the load process.
- Assign new System Rule numbers by BEA Release number. For example, new System Rule Numbers for BEA Release 6.0 could start with "60001" and sequentially incremented by "1" for each subsequent new rule. This allows the new System Rules to be easily identified.

16 StdV-1 – Standards Profile

16.1 Summary Description

This section describes the Technical Standards Profile, its relationship to other BEA models, the development method, and the modeling guidelines to be followed.

Note: The StdV-1 is in the process of being restructured. For this deliverable the standards are only mapped to the Technical Services identified in the DoD IT Standards Repository (DISR).

16.1.1 Model Purpose

The purpose of the BEA StdV-1 is to describe the mandated Information Technology (IT) standards that a BEA-compliant system must implement, as needed, to provide interoperability and net-centric services across the DoD Enterprise. The fundamental requirement driving the content of the StdV-1 is the mandate for compliance with the DISR. In general, the Standards Profile (StdV-1) provides the minimal set of rules, standards, and protocols governing the arrangement, interaction, and interdependence of system parts or elements. In addition to a collection of the technical standards, the StdV may include implementation conventions, standards options, rules, and criteria organized into profile(s) that govern systems and system elements for a given Architecture. When the standards profile is tied to system elements through services, the StdV-1 serves as the bridge between the SV and StdV

The StdV-1 is a listing of standards that must be followed when implementing a given architecture. This will ensure the implementation will provide the system capabilities identified in the Systems Views (SV) that are required to meet the operational needs defined by the architecture's Operational Views and its specific concept of operations.

16.1.2 Model Structure

The StdV-1 is a tabular model. It comprises a Technology Service Area, Technical Services, Standards, and Standard Description as outlined in the following sections.

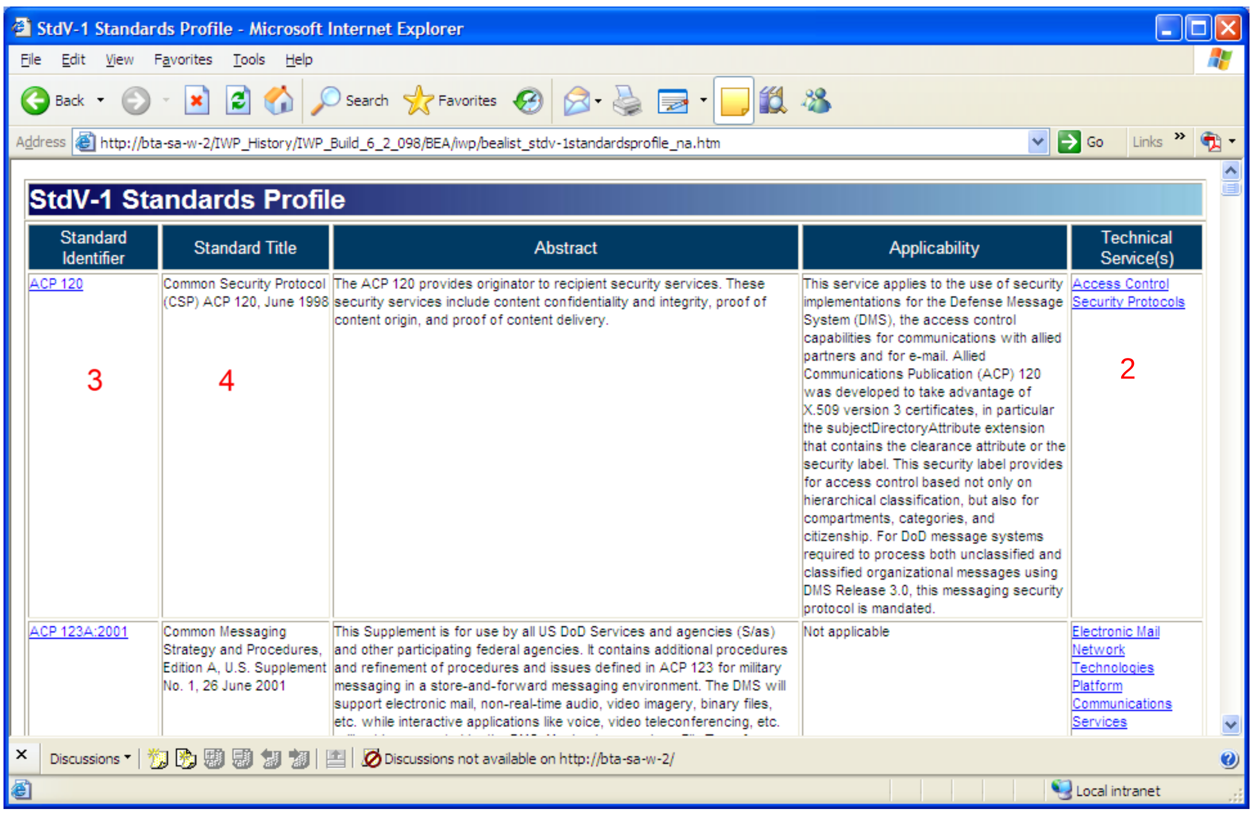
The StdV-1 is aligned with the SV-1, SV-5 and SV-6 Systems Views. In the architecture, the selected standards are related to the Systems in the SV-1. In support of the architecture implementer or system designer, each standard listed in the profile is associated with the SV elements that implement or use that standard.

The StdV-1 Standards Profile is in a table matrix format. The hierarchical structure of the StdV-1 consists of a three-tier set of categories: Technology Service Areas, which contain Technical Services that conform to Standards. Such a structure makes it easier for architecture implementers and system designers to locate the standards that apply. The structure adopted for the BEA StdV-1 is that defined by the Core Architecture Data Model for DoDAF-compliant architecture.

The categories used to represent the StdV-1 model are numbered as shown in Table 16-1, Example of the BEA StdV-1 Matrix.

Table 16-1, Example of the BEA StdV-1 Matrix

Technology Service Area	Technical Services	Standards	Standard Description
Component Framework 1	Access Control 2	FIPS Pub 140-2 3	Security Requirements for Cryptographic Modules May 25, 2001. 4
Component Framework	Information Assurance	SDN.801	Access Control Concept and Mechanisms, Revision C, May 12, 1999.
Service Access and Delivery	Collaboration	ISO/IEC 11171-2	Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s - Part 2 Video, 1993.


StdV-1 Standards Profile - Microsoft Internet Explorer

http://bta-sa-w-2/IWP_History/IWP_Build_6_2_098/BEA/iwp/bealist_stdv-1standardsprofile_na.htm

StdV-1 Standards Profile

Standard Identifier	Standard Title	Abstract	Applicability	Technical Service(s)
ACP 120 3	Common Security Protocol (CSP) ACP 120, June 1998 4	The ACP 120 provides originator to recipient security services. These security services include content confidentiality and integrity, proof of content origin, and proof of content delivery.	This service applies to the use of security implementations for the Defense Message System (DMS), the access control capabilities for communications with allied partners and for e-mail. Allied Communications Publication (ACP) 120 was developed to take advantage of X.509 version 3 certificates, in particular the subjectDirectoryAttribute extension that contains the clearance attribute or the security label. This security label provides for access control based not only on hierarchical classification, but also for compartments, categories, and citizenship. For DoD message systems required to process both unclassified and classified organizational messages using DMS Release 3.0, this messaging security protocol is mandated.	Access Control Security Protocols 2
ACP 123A:2001	Common Messaging Strategy and Procedures, Edition A, U.S. Supplement No. 1, 26 June 2001	This Supplement is for use by all US DoD Services and agencies (S/as) and other participating federal agencies. It contains additional procedures and refinement of procedures and issues defined in ACP 123 for military messaging in a store-and-forward messaging environment. The DMS will support electronic mail, non-real-time audio, video imagery, binary files, etc. while interactive applications like voice, video teleconferencing, etc.	Not applicable	Electronic Mail Network Technologies Platform Communications Services

Discussions not available on http://bta-sa-w-2/

Local intranet

Figure 16-1, Example of the BEA StdV-1 Standards Profile

Figure 16-2, Relationships between StdV-1 and Other BEA Models, the StdV-1 is related to other BEA models as follows:

<table>
<tr><td>AV-1</td><td>The scope of the development effort for each CBM for a development cycle, as disclosed in the AV-1, will determine if the StdV-1 is affected in the release.</td></tr>
<tr><td>AV-2</td><td>All StdV-1 terms with specific meaning must be defined in the AV-2 Terms. These terms must include, as a minimum, all object types included in the deliverable.

These StdV-1 deliverable objects must be listed and defined in the AV-2:
Standard Definitions
Technical Service Definitions
Technology Service Area Definitions
All acronyms used in the StdV-1 descriptions must be listed and spelled out in the AV-2 Acronyms.</td></tr>
<tr><td>OV-5</td><td>Technical Standards in the StdV-1 are selected based on operational requirements derived from the business operations defined by the OV-5. However, the StdV-1 dictates the function modeling notation used to develop the OV-5.</td></tr>
<tr><td>OV-6c</td><td>Although all BEA models are related through integration of the architecture, no direct relationship exists between the StdV-1 and the OV-6c. However, the StdV-1 dictates the business process modeling notation used to develop the OV-6c.</td></tr>
<tr><td>DIV-1

DIV-2</td><td>Although all BEA models are related through integration of the architecture, no direct relationship exists between the StdV-1 and the DIV-2. However, the StdV-1 dictates the data modeling notation used to develop the DIV-2 and specific Standards may be related to Data Domains related to the DIV Model.</td></tr>
<tr><td>SvcV-1</td><td>Through Technical Services in the StdV-1, standards govern what hardware and software may be implemented on what system.</td></tr>
<tr><td>SvcV-5</td><td>The System Entity in the SvcV-1 uses the services in the SvcV-5 to describe its service capabilities. This establishes the relationship to the StdV-1. In addition, the OV-5 activity represented in the SvcV-5 provides further linkage.</td></tr>
<tr><td>SV-1</td><td>Through Technical Services in the StdV-1, standards govern the arrangement, interaction, and interdependence of systems (parts and/or elements) represented as System Entities in the SV-1.</td></tr>
<tr><td>SV-5</td><td>The System Entity in the SV-1 uses System Functions in the SV-5 to describe its system capabilities. This establishes a relationship to the StdV-1. In addition, the OV-5 activity represented in the SV-5 provides further linkage.</td></tr>
<tr><td>SV-6</td><td>The System Entity in the SV-1 is represented as a sending or receiving system data (a system element) in the SV-6. This establishes a relationship to the StdV-1.</td></tr>
</table>

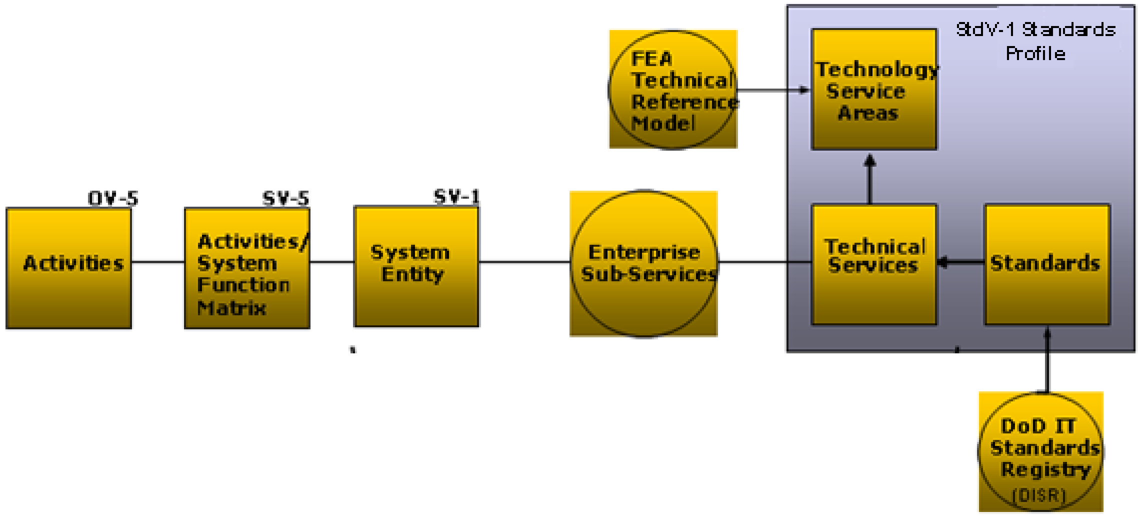


Figure 16-2, Relationships between StdV-1 and Other BEA Models

16.1.3 StdV-1 Model Definitions

The tabulated columns of this matrix, as illustrated in Table 16-1, Example of the BEA StdV-1 Matrix, represent the following:

Technology Service Area (1)

Technology Service Areas group similar Technical Services together for increased organization and comprehension. There may be one or more Technical Services in a Technology Service Area. The current StdV-1 takes its highest-level structure from the DoD Enterprise Architecture (EA) Technical Reference Model (TRM). It contains four Technology Service Areas, drawn from the Core Service Areas of the DoD EA TRM. This provides a high degree of traceability between the two documents and makes optimal use of the DoD EA TRM as the interface between the BEA and the FEA Consolidated Reference Model (CRM), to which DoD programs must map for Office of Management and Budget (OMB) Exhibit 300 purposes.

The current BEA Technology Service Areas are:

- **Component Framework**: The underlying foundation, technologies, standards and specifications by which system capabilities are built, exchanged and deployed across the BMA.
- **Service Access and Delivery:** The collection of standards and specifications that support external access, exchange and delivery of a system capability.
- **Service Interface and Integration:** The collection of technologies, standards and specifications that govern the interface with a system capability.
- **Service Platform and Infrastructure:** The collection of delivery and support platforms, infrastructure capabilities and hardware requirements to support the construction, maintenance and availability of a system capability.

Technical Services (2)

In the StdV-1 model a Technical Service, with its constituent standards, is a technical capability designed to support a Net Centricity. Technical Services are assigned to each Technology Service Area within the BEA StdV-1 to support the development of BEA-compliant systems. There may be one or more Technical Services in any given Technology Service Area.

Note: Enterprise Sub-Services - A fundamental component of the BEA framework is the Enterprise-wide infrastructure that will provide the foundation for all relevant business services. Whereas much of the framework development centers on the operational business aspects of the architecture, there are several areas that focus on those components that support the business processes, but are not directly related to the business requirements. They describe the intersection between the business processes and Technical Services, and define standard attributes to bring order to that point. The BEA refers to these components as Enterprise Sub-Services. Each identified Enterprise Sub-Services can be linked to the specific Technical Services that will implement the Enterprise Sub-Service. Enterprise Sub-services are directly related to System Entities that are represented on the StdV-1. There may be more than one Enterprise Sub-service that supports a System Entity. Enterprise sub-services support the linkage of a SV-StdV Bridge.

Standards (3)

This is the standard identifier, found in the DISR, determined and designated by the applicable Standards Development Organization (SDO). It is represented as the Standard name in the BEA. One or more Standards support a given Technical Service.

Standard Description (4)

This is the official title of the standard determined by the applicable SDO and is represented as the standard description in the BEA. The standard description represents an agreed upon means to implement all or part of a Technical Service in the BEA. The DISR is the origin of all Standards in the BEA StdV-1, and appropriate references to the DISR and to additional information about the Standards is provided for each Standard. As content of the DISR changes over time, the BEA StdV-1 updates will reflect the relevant changes in the Standards.

Since a fundamental requirement driving the content of the StdV-1 is the mandate for compliance with the DISR, all updates to DISR are analyzed for applicability to the business mission and those resulting updated Standards are included into the BEA Technical Standards Profile. Mandated Standards are essential for providing interoperability and net-centric services across the DoD Enterprise. These are current and established Standards that are required as the "must comply" Standards that implement the Technical Services without deviation. Mandated Standards usually are widely adopted and mature technologies. Compliance with the DISR is mandated for all new DoD information systems to support interoperability and net-centricity across the DoD Enterprise. To accommodate this requirement, the BEA StdV-1 Standards Profile is adopted from Standards in the latest version of the DISR.

16.2 Developing the StdV-1 Models

This section describes the approach to develop, extend and maintain the StdV-1. The StdV-1 is developed in MS Excel and HTML as a table.

Development of the StdV-1 starts with collecting information relevant to the standards contained in the BEA and ensuring the most up-to-date DISR release is being used. Technical Services are then developed that may be used by the specific enterprise systems listed in the Operational Activity to Systems Functionality Traceability Matrix (SV-5). Individual IT Standards that are required to implement each identified Technical

Service are from the DISR and associated with the Technical Service. Identified Technical Services that do not have Standards associated with them are not included in the final StdV-1. The StdV-1 identifies the source documents used for each Standard it identifies. The StdV-1 also includes any relevant IA Standards listed in the latest DISR baseline.

Throughout the development of the BEA StdV models, analysts and engineers make a number of decisions that affect the content of each new model release. These decisions occur periodically during the StdV development process, a process comprised of five high order procedures. The Engineering Decisions made during these procedures, and their impact upon the BEA Standards Profile models, is as follows:

16.2.1 Pre-Development Tasks

The tasks that must be completed prior to StdV-1 development and/or maintenance are:

1. Review the CBM's AV-1 and BIP to understand the impact of the planned body of work on the StdV-1 Standards Profile.

2. Identify models that could potentially be affected by changes to the StdV-1and begin coordination within the BEA Development Team.

3. Collect Information through SME interviews.

    a. Conduct periodic interviews with industry and DoD technology authorities. This decision is derived from the project plan; however, specific implementation is subject to the team consensus regarding areas of technology that should be addressed first. Therefore, areas such as security or web services may hold an apparently arbitrary advantage over technologies such as XML based upon the Engineering Decision of the StdV analyst.

    b. Participate in DISR Information Technology Standards Working Groups (ISWGs). This provides a forum for discussing the Standards with representatives from various DoD organizations who have the proper technical, functional and acquisition expertise from their organizations. The ISWGs are responsible for making recommendations for updating the DISR. The technology areas provide the primary body for identifying the lifecycle stage of each Standard and profile contained in the DISR. The ISWGs are responsible for making recommendations for updating the DISR.

4. Coordinate internal StdV-1 Development Cycle with DISR release schedule.

    a. Refer to DISR online to get specific instructions for activities associated with the DISR release schedule.

    b. The StdV-1 must be developed using the latest DISR updated baseline, which happens three times a year. This insures that all standards within the BEA are accurate and up-to-date.

16.2.2 Development Tasks

The development and maintenance of the Technical Standards Profile is accomplished in facilitated workshops that include Government SME participation to address content and validate results. The following subsections describe the approach used to develop the StdV-1 for the BEA. Each subsection sets forth the specific tasks that must be accomplished to in each stage of the development phase. Although most of these steps are sequential, it is common to start some steps before a previous step is completed.

16.2.2.1 Creating/Modifying the StdV-1 Models

This section describes the approach to develop the StdV-1.

1. Identify and Define Technical Services and Standards Data

    a. The widest possible array of authoritative sources is consulted for guidance and cross reference of Standards, either mandated or emerging, within DoD and its services. Ongoing business and technical analysis of mandated Standards in the DISR determines relevance to the BEA StdV-1. Standards that do not directly apply to business systems (such as Standards for routing protocols, backplane buses and weapons systems) in the business domains are outside the scope of the BEA StdV-1.

2. Organize Information into a Local Data Repository

    a. Use System Architect (and updates) as the Data Repository and schema for standards-related data to facilitate the integration of data between Operational View (OV), SV and StdV models.

    b. Use Microsoft Excel as the working repository of Technical Services-related data. The use of Excel decreases the requirement for additional operator training on the System Architect model. It increases the flexibility with which multiple analysts can interact with the data repository. Excel increases the ease with which draft versions of the repository are shared with other members of the BEA Development Team. It also alleviates the limitations placed upon users who need access to work with the data repository (licenses, learning curve and level of effort). Excel increases the team's ability to perform analytical reviews of the data repository using Excel's data analysis capabilities.

3. Conduct Data Analysis to Target BEA Requirements

    a. Tailor the collection of Technical Services to meet BEA requirements.
    The repository of Technical Services changes as needed by Engineering Decision to eliminate Technical Services that are outside the area of direct interest to BEA and include new Technical Services when appropriate. For example, to better align with the newly emerging DISR Standards organization schema, the Application, Collaboration, Discovery, Enterprise Service Management (ESM), Information Assurance and Security (IAS), Infrastructure Transport, Mediation, Messaging, Storage, and User Assistance, Logistics and Human Resources enterprise services may be treated as BEA Technical Services.

4. Produce the StdV-1

    a. Load the System Architect Data Repository immediately prior to model delivery. The latest available version of data available through Systems Architect is the version last delivered as a finished model.

    b. Customize the System Architect Standards Profile reports, and generate the StdV-1 and associated reports. The predefined Standards Profile reports were found to be inadequate to the needs of the BEA. Customized reports are created to show the StdV-1 and the linkage to the System View models.

16.2.3 Post-Development Tasks

These tasks are performed after changes to the StdV-1 by the Stakeholders to ensure integration of the architecture:

1. Hold Integration workshop.
2. Obtain feedback and make modifications.
3. Obtain approval from the Stakeholders.

16.3 Modeling the StdV-1 Models Using SA

16.3.1 StdV-1 Modeling Conventions

This section does not apply to the StdV-1.

16.3.2 Modeling StdV-1 Objects

This section does not apply to the StdV-1.

16.3.3 StdV-1 Best Practices

This section discusses best practices, including lessons learned during previous BEA development cycles pertaining to the development of the StdV-1 and common pitfalls to avoid while updating the model.

16.3.3.1 StdV-1 Lessons Learned

The following lessons learned have been and serve as the basis for the StdV-1 on Appendix B:

- The SV-1, SV-5 and SV-6 models must be stabilized.
- Technical Standards must be grouped into meaningful Enterprise Services to ensure that only the appropriate Standards are mapped to enterprise systems.
- Ensure that all exception reports have been reviewed and resolved.

16.3.3.2 StdV-1 Common Pitfalls

The following are common mistakes encountered in the development of the StdV-1 that have been identified and may also serve as the basis for the StdV-1 Checklist on Appendix B:

- Excluding acronyms in the AV-2.
- Grouping too many Standards into too few Enterprise Services.
- Not using the most recent DISR baseline release

17 DFMIR/FFMIA Guidance Model

17.1 Summary Description

This section describes the Defense Acquisition Management Information Retrieval and Federal Financial Management Improvement Act (DFMIR/FFMIA) Guidance Model architecture model, its relationship to other BEA models, the model-development method and the modeling guidelines used for development of the DFMIR/FFMIA Guidance Model.

FFMIA is a public law that requires each Federal agency to implement and maintain financial management systems that comply with applicable accounting standards and systems requirements. DFMIR are system and operational rules which substantially improve the timeliness, reliability, and accuracy of financial information across DoD and contribute towards the Department's objectives of achieving audit readiness and delivering interoperable financial management systems. DFMIR/FFMIA Guidance statements may or may not be compliance artifacts based on an indicator and type described in subsequent sub-sections.

17.1.1 Model Purpose

The BEA distinguishes between a Business Policy/Guidance and Business Rules.[10] The distinctions are based on the definitions of a Business Policy/Guidance and Business Rules as derived from Object Management Group's *Semantics of Business Vocabulary and Business Rules v1.0* and *DoDAF v2.0*. The primary differences between Business Policy/Guidance and Business Rules are that Business Policy/Guidance can be:

- Less structured
- Less discrete or not atomic
- Less carefully expressed in terms of a standard vocabulary
- Not directly enforceable.

The DFMIR/FFMIA Guidance Model is the set of statements that constrain an Enterprise, mission, operation, business, or architecture, but the statements do not conform to the BEA Business Rule standards. The BEA Operational Rules and System Rules follow the BEA standards and are atomic, unambiguous and written using a standard notation. DFMIR/FFMIA Guidance statements may not be atomic, may not be unambiguous, and may not be written in a standard notation. DFMIR/FFMIA Guidance statements are entered into the BEA to fulfill the DoD business mission of describing what the business can and cannot do.

In the future, DFMIR/FFMIA Guidance statements may be rewritten to comply with the BEA APG Business Rule standards and then migrated to become BEA OV-6a Operational Rules or SV-10a System Rules. Additionally, the rewriting of the DFMIR/FFMIA Guidance statements may update the AV-2 Acronyms and Terms.

[10] These are also distinct from Laws, Regulations and Policies (LRP).

17.1.2 Model Structure

The DFMIR/FFMIA Guidance model is depicted in a textual matrix format with no diagrams. It is manually created and an unlimited number of DFMIR/FFMIA Guidance statements can be stored within the BEA System Architect encyclopedia.

17.1.3 Relationship to Other BEA Models

DFMIR/FFMIA Guidance is related to other BEA models through the following:

AV-1	The scope of the development effort for each CBM for a development cycle, as disclosed in the AV-1, will determine if the StdV-1 is affected in the release.
AV-2	DFMIR/FFMIA Guidance terms with specific meaning and acronyms may be defined in the AV-2 Terms; however, it is not a requirement.
OV-6c	DFMIR/FFMIA Guidance statements must be mapped to at least one BPM Processes in the OV-6c Event Trace Diagrams.

17.1.4 DFMIR/FFMIA Guidance Model Definitions

17.1.4.1 DFMIR/FFMIA Field Definition

This section defines concepts and terms often used when discussing DFMIR/FFMIA Guidance statements. The DFMIR/FFMIA Guidance Model is not a DoDAF model. However, DFMIR/FFMIA Guidance statements can be used as specific compliance artifacts.

Figure 17-1, Data Fields and Structure of a DFMIR/FFMIA Guidance Statement, illustrates the nine data fields and structure of a DFMIR/FFMIA Guidance statement. Four data fields are text entry. Three of the data fields can have multiple data values.

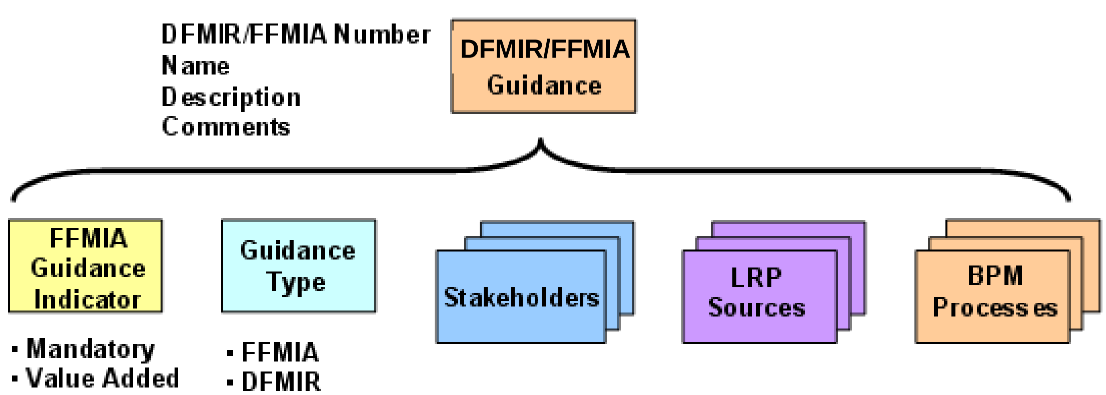


Figure 17-1, Data Fields and Structure of a DFMIR/FFMIA Guidance Statement

1. **DFMIR/FFMIA Number**: A mandatory data field that is a unique number given to a DFMIR/FFMIA Guidance statement for BEA identification purposes. The unique number is manually assigned. The following are examples of correct DFMIR/FFMIA Guidance Numbers:

   - 6001579
   - 6001580

The DFMIR/FFMIA Number should be assigned sequentially within the BEA Release number as shown above. That is, the BEA Release is “6.0” and the sequential numbers are “1579” and “1580”. This assignment technique allows all of the new DFMIR/FFMIA Guidance Statements for a BEA Release to be easily identified.

2. **Name**: A mandatory data field that is a unique name given to a DFMIR/FFMIA Guidance for user identification purposes. The DFMIR/FFMIA Guidance Name is limited to 80 characters and contains an underscore between each of its parts. Each part can be either alphabetic or numeric. The DFMIR/FFMIA Guidance Name must begin with an alphabetic part. Each alphabetic part must start with an uppercase letter followed by lowercase letters unless the term is a capitalized acronym. The DFMIR/FFMIA Guidance Name is used to map to other architecture artifacts such as a BPM Process. The following is an example of a correct DFMIR/FFMIA Guidance Names:

   - Budget_Sub_Activity_Identifier_Association_1

3. **Description**: A mandatory data field that contains the actual DFMIR/FFMIA Guidance statement. It is a statement of constraint or permission. Each DFMIR/FFMIA Guidance Description must be unique. In other words, there must not be any duplicate DFMIR/FFMIA Guidance Descriptions in a BEA release. The following is an example of a correct DFMIR/FFMIA Guidance Description:

   > An appropriation must be recognized as revenue
   > when it is used for goods and services received.

4. **FFMIA Guidance Indicator:** A mandatory data field that must be either “Mandatory” or “Value Added”. DFMIR/FFMIA Guidance statements that are denoted as “Mandatory” are compliance artifacts.

5. **Guidance Type:** A mandatory data field that must be either “FFMIA” or “DFMIR” depending on the source of the statement.

6. **Comments**: An optional data field for use by the Stakeholder(s).

7. **LRP Sources**: An optional multiple valued data field that contains a link to the specific Law, Regulation or Policy in the LRP Repository. The LRP Sources are selected from a list of authoritative LRP Sources.

8. **Stakeholders**: A mandatory data field that can have multiple values that assigns a DFMIR/FFMIA Guidance statement to one or more owners. The Stakeholders field must contain at least one CBM or CBM designation. The selections are made from a drop down list.

9. **BPM Processes**: A mandatory multiple valued data field that contains the name or names of BPM Processes. Each DFMIR/FFMIA Guidance statement must be linked to at least one BPM Process. The BPM Process artifacts are selected from the list of BPM Processes.

17.1.4.2 LRP Source Definition

This section discusses the update to the LRP Source Definition that is performed after the DFMIR/FFMIA Definitions have been loaded. The reason for this is that a new LRP Source Definition is automatically created during the DFMIR/FFMIA Definition load when that LRP Source does not already exist. The Name data field of the LRP Source Definition is the identifier of a LRP Source Definition. This value is used to subsequently load the corresponding LRP Source Definition data fields for a DFMIR/FFMIA Guidance Statement Definition.

The four text data fields required for LRP source definition are described below:

1. **Name**: A mandatory data field that is a valid LRP Source Identifier to the specific Law, Regulation or Policy in the LRP Repository. The value must be the same as the value loaded into the corresponding DFMIR/FFMIA Guidance Statement Definition.

2. **Description**: A mandatory data field that contains the description of the LRP item stored in the LRP Repository. The value of the data field is provided by the Stakeholder.

3. **Primary Reference**: A mandatory data field that is the cited source of the description in the LRP Repository. The value of the data field is provided by the Stakeholder.

4. **Link**: A mandatory data field that is the same as the value stored in the Name data field.

17.1.4.3 DFMIR/FFMIA Guidance Input Field Capture

This section describes capturing the DFMIR/FFMIA Guidance field values for inserts or changes to the System Architect encyclopedia.

17.1.4.3.1 Input Document

The primary input document is a spread sheet that has the following four tabs:

1. Main Load

2. LRP Sources

3. Stakeholders

4. BPM Processes.

The Main Load tab contains all of the above single entry fields as columns. That is, the data fields for which there is only one value; such as the DFMIR/FFMIA Guidance Description.

The other three tabs contain the fields for which multiple occurrences are possible; for example, a DFMIR/FFMIA Guidance statement can be mapped to more than one Process. The map between the DFMIR/FFMIA Guidance statement and the other BEA artifacts is the DFMIR/FFMIA Guidance Name. The following is an example of entries in the BPM Process tab.

BPM Process Name	DFMIR/FFMIA Guidance Name
Manage Financial Management Policy	Audit_Trails_9
Manage Financial Reporting	Deferred _Maintenance_And_Cleanup_Costs_1
Manage Financial Management Policy	Deferred _Maintenance_And_Cleanup_Costs_1

The LRP Sources and the Stakeholders tabs have the same construct as the above.

This input document is also used as a "turn around" document for reviews between the BEA Development Team and the Stakeholders.

17.1.4.3.2 System Architect Update Process

There are two basic System Architect update processes; one for a large update volume and the other for a small update volume.

A large update volume requires assistance from the System Architect Build Team. The System Architect Build Team uses the spread sheet as input to an automated procedure.

A small update volume is manually entered into System Architect by the BEA Development Team.

17.1.4.3.3 Mapped Field Validation

The mapped field names must be correct, otherwise errors will be generated and the DFMIR/FFMIA Guidance will not be complete. The System Architect Build Team automated load procedure automatically inserts artifacts when a match is not found. For example, an incorrect BPM Process name will cause a new BPM Process to be added with that name. This is an error that must be manually corrected. Consequently, care must be taken to assure that the mapped names are valid and already exist in System Architect.

17.2 Developing the DFMIR/FFMIA Guidance Model

This section describes the approach used by the BEA Development Team to develop the DFMIR/FFMIA Guidance model. The BEA Development Team works with Stakeholders' functional SMEs to produce DFMIR/FFMIA Guidance statements that support the business transformation. The DFMIR/FFMIA Guidance Process includes development, maintenance, and retirement of DFMIR/FFMIA Guidance statements.

17.2.1 Pre-Development Tasks

Pre-development work may include suggesting general guidance on how to develop content, answering any form and structure questions, or generating and analyzing System Architect reports from the prior BEA release. This process for DFMIR/FFMIA Guidance creation does not require detailed analysis of the architecture. The Stakeholders provide the BEA Development Team with their identified DFMIR/FFMIA Guidance statements mapped to BPM Processes for pre-analysis.

17.2.2 Development Tasks

The BEA development schedule must allot time for DFMIR/FFMIA Guidance development tasks. This period of time is referred to as the DFMIR/FFMIA Guidance workshop, but these tasks are performed outside of a formal workshop setting.

17.2.2.1 Refine DFMIR/FFMIA Guidance Statement

The Stakeholders identifies the DFMIR/FFMIA Guidance statements, its BPM Process mapping, and the LRP Source Identifier(s), if applicable, during the Pre-Development Task. The Stakeholders determine whether the DFMIR/FFMIA Guidance statement already exists in the BEA.

The DFMIR/FFMIA Guidance Indicator data value is determined by the Stakeholders. For DFMIR/FFMIA Guidance statements, each statement must have an indicator of "Mandatory" or "Value Added". The indicator is used while assessing DFMIR/FFMIA compliance.

17.2.2.2 DFMIR/FFMIA Guidance Model Coordination with Stakeholders

After the BEA Development Team completes the technical review, a DFMIR/FFMIA Guidance statement may be passed back to the Stakeholders for a functional review. The Stakeholders verify the proposed DFMIR/FFMIA Guidance statement. If the proposed DFMIR/FFMIA Guidance statement passes the functional review, the Stakeholders return it to the BEA Development Team for pre-load verification. The BEA Development Team cannot change the content of the DFMIR/FFMIA Guidance statement; however, spelling and minor format discrepancies can be denoted for change.

17.2.2.3 DFMIR/FFMIA Guidance Load Preparation

After the Stakeholders and the BEA Development Team agree upon the validity of the DFMIR/FFMIA Guidance statements, the DFMIR/FFMIA Guidance statements are ready for the BEA Development Team to load them into the BEA. The BEA Development Team uses the load sheet, which is an MS Excel spreadsheet that lists the DFMIR/FFMIA Guidance statements and all associated fields discussed in 15.1.4 Definitions. This load sheet is submitted to the Stakeholders for approval through the designated BEA development process. Once the Stakeholders grant final approval, the BEA Development Team submits the load sheet and a copy of the approved CR to the System Architect Build Team.

The BEA Development Team can manually input the approved DFMIR/FFMIA Guidance statements into System Architect. This approach is normally used when the DFMIR/FFMIA Guidance statement volume is less than 20.

Once the DFMIR/FFMIA Guidance statements are in System Architect, the BEA Development Team verifies that the load sheet is correctly represented in the BEA. This step must wait for the System Architect Build Team to perform an Encyclopedia update so that the BEA Development Team can conduct the Post-Build Verification on the latest Encyclopedia build. Typically, this verification is performed using the SA Report Generation to create a report that includes all of the loaded data fields. If the BEA Development Team discovers a discrepancy at this point, he or she works with the System Architect Build Team to correct it.

17.2.2.4 Creating/Modifying the DFMIR/FFMIA Guidance Model

To ensure that the DFMIR/FFMIA Guidance statements remain valid, the BEA Development Team follows a maintenance and retirement process. The Stakeholders notify the BEA Development Team that there is a change in the BEA affecting existing DFMIR/FFMIA Guidance statements. The BEA Development Team identifies the DFMIR/FFMIA Guidance statements and makes any necessary adjustments in System Architect. Next, the BEA Development Team analyzes the architecture for potential changes to the existing processes and other linkages in System Architect. Finally, the BEA Development Team validates the above process for continued support of the business transformation. The following discusses the maintenance process in detail:

17.2.2.4.1 Identify DFMIR/FFMIA Guidance Statements

Either the Stakeholders or BEA Development Team may initiate an analysis to determine if changes are needed. A number of circumstances may trigger a change to some aspect of the DFMIR/FFMIA Guidance statement and/or its artifacts. Below are typical (but not an inclusive list of) triggers:

- Business Objective change
- Process Step change
- Requirement change
- LRP Source change

Note: If a LRP unique Identifier or description already mapped to a DFMIR/FFMIA Guidance statement changes in the LRP Repository, the LRP team will notify the BEA Development Team and/or the Stakeholders. The BEA Development Team will manage the change in System Architect.

17.2.2.4.2 Analyze Potential Changes

Once the change is identified, the BEA Development Team conducts analysis to determine any potential impacts on the BEA. The analysis includes impact to other architectural models, existing DFMIR/FFMIA Guidance statements, Business Rules and/or requirements.

17.2.2.4.3 Validate DFMIR/FFMIA Guidance Changes

The BEA Development Team reviews the DFMIR/FFMIA Guidance statements. After completing technical review, the BEA Development Team works with the appropriate Stakeholder for functional approval.

17.2.2.4.4 Update System Architect

Upon approval, the BEA Development Team makes the changes in System Architect or creates a load sheet to present to the Build Team for automated loading.

17.2.3 Post-Development Tasks

As with the DFMIR/FFMIA Guidance maintenance process, a number of events may trigger the retirement process, such as:

- A Business Objective change or elimination

- BPM Process modification or elimination
- LRP Source change or elimination

The Stakeholders notify the BEA Development Team that a DFMIR/FFMIA Guidance statement needs to be retired. The BEA Development Team, working closely with the Stakeholders, documents the reason for retirement, and then the DFMIR/FFMIA Guidance statement is retired from the System Architect encyclopedia. Finally, the appropriate validation steps are executed to ensure that the DFMIR/FFMIA Guidance statement was actually retired.

Note: The Stakeholders review each DFMIR/FFMIA Guidance statement identified for retirement, verifying whether it is appropriate to retire the DFMIR/FFMIA Guidance statement for all Processes. If a DFMIR/FFMIA Guidance statement, identified for retirement, still has a valid mapping to another Process Step, the BEA Development Team retains the DFMIR/FFMIA Guidance statement, removing only the link to the retired Process Step.

The following sections discuss the retirement process in more detail:

17.2.3.1 Identify Obsolete DFMIR/FFMIA Guidance Statement

The Stakeholders are responsible for identifying a DFMIR/FFMIA Guidance statement that needs to be retired. In addition, if an architecture object deletion or change is the trigger, the Stakeholders identify whether each instance of the DFMIR/FFMIA Guidance statement must be deleted or just the mapping between the DFMIR/FFMIA Guidance statement and the architecture object being changed.

17.2.3.2 Retire DFMIR/FFMIA Guidance Statement

The BEA Development Team removes the identified DFMIR/FFMIA Guidance statement from System Architect. The BEA Development Team also identifies to the LRP team a retired DFMIR/FFMIA Guidance statement that has a LRP Identifier associated with it. If the DFMIR/FFMIA Guidance statement is still valid, and possesses valid mappings to other architecture objects, the BEA Development Team removes only the prescribed mapping.

17.2.3.3 Validate DFMIR/FFMIA Guidance Statement Retirement

The BEA Development Team conducts the appropriate quality assurance checks. These checks validate the appropriate retirement of a DFMIR/FFMIA Guidance statement and prevent the creation of orphan process and data artifacts.

17.3 Modeling DFMIR/FFMIA Guidance Using SA

17.3.1 DFMIR/FFMIA Guidance Modeling Conventions

Guidelines that assist in the identification and definition of DFMIR/FFMIA Guidance statement are:

- Each DFMIR/FFMIA Guidance statement must have a unique DFMIR/FFMIA Guidance Name. The DFMIR/FFMIA Guidance statement Name format is described in Subsection 15.1.4 of this document.
- Each DFMIR/FFMIA Guidance statement must be assigned a unique DFMIR/FFMIA Number by which it can be identified.

- Each DFMIR/FFMIA Guidance statement must have a unique description. The Description is the actual DFMIR/FFMIA Guidance statement. Subsection 15.1.4 of this document provides details on the construction of DFMIR/FFMIA Guidance statements.
- Each DFMIR/FFMIA Guidance statement must have at least one Stakeholder. The Stakeholder may be an individual CBM designation.
- Each DFMIR/FFMIA Guidance statement must have a FFMIA Guidance Indicator of either "Mandatory" or "Value Added".
- Each DFMIR/FFMIA Guidance statement must have a Guidance Type of "FFMIA" or "DFMIR".

17.3.2 DFMIR/FFMIA Guidance Best Practices

This section discusses best practices, including lessons learned from previous architecture development efforts and common modeling pitfalls to avoid.

17.3.2.1 DFMIR/FFMIA Guidance Model Lessons Learned

- A clear understanding of the OV-6c Event Trace Diagrams must come before DFMIR/FFMIA Guidance development and linkage from the DFMIR/FFMIA Guidance to one or more Business Processes.
- CBM DFMIR/FFMIA Guidance content that affects other CBMs should be socialized before the workshop.
- Avoid embedding term definitions in DFMIR/FFMIA Guidance. (Use AV-2 to define terms.)
- Ensure that all exception reports have been reviewed and resolved.
- Validate the existence of System Architect mapped Process names using spreadsheet macros such as VLOOKUP. This helps reduce errors in the load process.
- Try to avoid ambiguity.
- Simplify as much as possible by not over-explaining or adding unrelated information.
- Make sure subject is quantified and singular (e.g., by using the word "each").
- Separate compound statements ("and" or "or") into several individual DFMIR/FFMIA Guidance statements.
- Use the "If - Then" statement construct for conditional DFMIR/FFMIA Guidance statements.
- Ensure that the logical structure of the DFMIR/FFMIA Guidance makes sense.
- Review all domain specific DFMIR/FFMIA Guidance statements for contradictory or redundant statements.

17.3.2.2 DFMIR/FFMIA Guidance Model Common Pitfalls

- Ambiguity.
- Over-explaining or unrelated information.
- Contradictory or redundant statements.

18 End to End Model

18.1 Summary Description

This section describes the End to End (E2E) Models, their relationship to other BEA models, the development method and the modeling guidelines to be followed. The following 15 E2Es have been identified:

1. Acquire to Retire
2. Budget to Report
3. Concept to Product
4. Cost Management
5. Deployment to Retrograde/Redeployment
6. Environmental Liabilities
7. Hire to Retire
8. Market to Prospect
9. Order to Cash
10. Plan to Stock -- Inventory Management
11. Procure to Pay
12. Proposal to Reward
13. Prospect to Order
14. Service Request to Resolution
15. Service to Satisfaction

DoD Executive Management determines the implementation priority of the identified E2Es. The following E2Es Views are the currently identified priorities for BEA 8.0:

- Hire to Retire
- Procure to Pay

The identified priorities have been modeled utilizing an enhanced approach which better supports the BMA's transition from a function-centered approach to one that enables management of DoD business from an E2E perspective. This new approach focuses on the depiction of the activities within an E2E lifecycle in a hierarchy with associated process diagrams. As new E2E priorities are identified they will be modeled utilizing the new more integrated approach replacing the previous depiction of business flow models and mapped process steps.

18.1.1 Model Purpose

The E2E models enable the BEA to better support the BMA's move from a function-centered approach to one that looks at DoD business functions across the enterprise from an end-to-end process perspective. The E2E Business Flow Model provides a standardized alignment of BEA Activities and process diagrams to the business capabilities being implemented in the Commercial Off-the-Shelf (COTS)/Enterprise Resource Planning (ERP) programs being implemented in DoD. Because the E2Es span the CBMs, the models allow the BEA to better support the development of business capabilities from an enterprise perspective beyond just the COTS/ERP programs. This alignment also improves the alignment of BEA requirements to guide system implementations, investment management decisions, and portfolio management.

18.1.2 Model Structure

The E2E Model is depicted as a set of diagrams. There is one diagram for each E2E business lifecycle.

Each E2E model is a hierarchy of the E2E lifecycle as shown in the following figure:

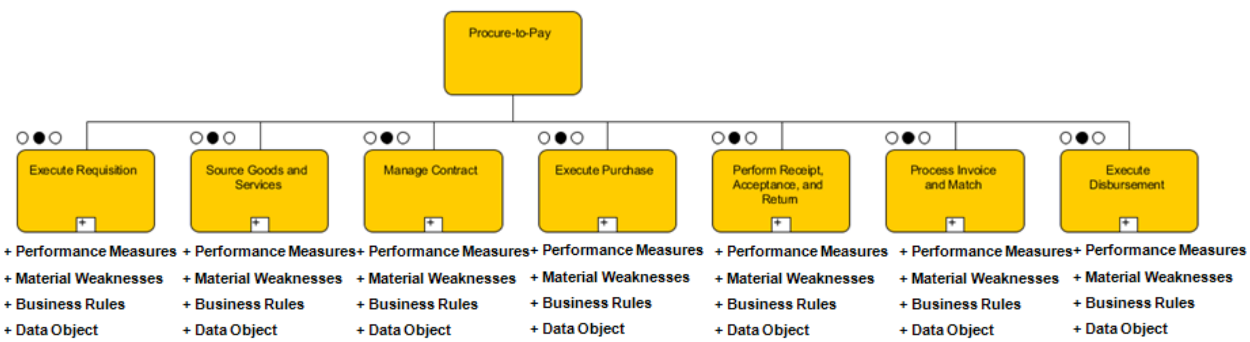


Figure 18-1, E2E Model Structure

The top level of the hierarchy is the Level 0 or name of the E2E lifecycle. The next level of the hierarchy depicts the top level of activities performed in the lifecycle of the E2E and is referred to as the Level 1. Some E2Es may require a further level of decomposition below the above depicted level 1 to expose the necessary level of detail. Each activity in the E2E lifecycle may then have an associated process diagram that outlines process details in business process modeling notation (BPMN). An activity that has an attached BPMN diagram is denoted by a plus (+) symbol. Furthermore, the following information may be associated with each of the E2E activities:

- Performance Measures: Performance Measurements are a means of assessing progress against stated goals and objectives in a way that is unbiased and quantifiable. They bring with it an emphasis on objectivity, fairness, consistency, and responsiveness. At the same time, they function as a reliable indicator of an organization’s long-term health. Their impact on an organization can be both immediate and far-reaching.

- Materiel Weaknesses: Internal controls put in place to prevent significant financial statement irregularities are considered to be ineffective.
- Business Rules: Statements that define or constrain some aspect of a business.
- Data Objects: Data Information which does not have any direct effect on the Sequence Flow or Message Flow but does show the data that may be passed, created, or consumed. Data Objects are a mechanism to show how data is required or produced by flow steps.

18.1.3 Relationship to Other BEA Models

Integrated architectures provide a structured and organized approach for defining capabilities and understanding the underlying relationships and requirements for achieving those capabilities. The E2E Business Model is integrated into the BEA as indicated in the following table.

AV-1	The scope of the development effort for each CBM for a development cycle, as disclosed in the AV-1, will determine if the E2E Business Flow Model is affected in the release.
AV-2	Each term used in the E2E Business Flow Model definitions with a specific meaning must be included in the AV-2 Term Definitions report. Each acronym used in the E2E Business Flow Model descriptions must be listed and spelled in the AV-2 Acronyms Definitions List.
OV-6a	Business Rules in the OV-6a may constrain the BPMN Process Steps attached to the E2E.
OV-6c	BPMN process diagrams are attached to the E2E Level 1 or 2 activities.

18.2 Developing the E2E Diagrams

A top down modeling approach is used; at each level of decomposition, more detailed information is added.

New diagram development or extension of a current E2E Diagram is accomplished in facilitated workshops to address model content and provide preliminary validation of the results. The remainder of this subsection describes in detail the approach to develop the E2E Diagrams. Each subsection represents a step in the approach, and the specific tasks that must be accomplished to complete a given step. Although most of these steps are sequential, some may be started before the previous step is completed. The appropriate standards/guidelines that direct task accomplishment of the process models attached to the E2E diagrams are contained in subsection 8.3.

18.2.1 Pre-Development Tasks

Analyze BEA Improvement Proposal

Analyze BIP to determine if there is an existing E2E that may be enhanced to address the improvement proposal or if a new E2E is required.

18.2.2 Development Tasks

The primary source for changes to an E2E Diagram is the BEA Stakeholder working group. Workshops are conducted with appropriate SMEs and business analysts from the BEA Stakeholder community. During the workshops, business analysts capture proposed changes to diagrams and/or object descriptions. The business analysts conduct detailed analysis of approved changes and raise integration issues for resolution.

After revising all available materials and assessing the requirements, the E2E Business Flow Diagram architect may develop new/revised E2E Business Diagram objects, based on the architectural standards in this document, in cooperation with the BEA Stakeholder analyst. The objects are driven by Stakeholder requirements in accordance with the configuration management procedures.

18.2.2.1 Creating/Modifying the E2E Diagrams

Create a new E2E Diagram if the requirement is a new COTS/ERP business capability. Modify existing E2E diagrams if the modifications are a result of a workshop review. Workshop reviews can result from changes to activities or process diagrams in the BEA.

For each E2E develop a set of activities at Level 1 that encompass the lifecycle. Where deemed necessary by the SMEs for each Level 1 activity develop a process diagram(s) in BPMN that describes the detail to accomplish the activity. If the level of abstraction in the Level 1 activities is too high to develop meaningful process diagrams then Level 2 activities should be developed and process diagrams associated to those activities.

When creating a new diagram, ensure SME provides proper name and description. If the diagram already exists and content is added or changed, then update names and descriptions as required.

The appropriate standards/guidelines that direct task accomplishment of the remaining E2E objects are contained in subsection 23.3 for Performance Measures, subsection 24.3 for Material Weaknesses, and subsection 6.3 for Business Rules.

An E2E Business Flow Diagrams is not meant to be a BPMN diagram and does not follow BPMN conventions.

18.2.3 Post-Development Tasks

These tasks are performed after changes to the E2E Diagram have been approved by the BEA Stakeholders to ensure integration of the architecture:

- Use Microsoft Word to check the spelling and grammar of all objects in the diagram by exporting all object names, descriptions and graphic comments onto a Word document.
- Incorporate additional updates to the E2E Diagram based upon subsequent BEA Stakeholders working group sessions.
- Incorporate peer reviews, quality control reviews, IV&V reviews and architecture verification changes into the BEA.

18.3 Modeling the E2E Diagram

18.3.1 Modeling E2E Objects

18.3.1.1 Diagram Conventions

All Diagrams must be clearly named and defined.

- Each L0 E2E name shall be the name of the E2E Diagram.
- Each E2E Diagram shall include a description to provide a clear understandable narrative of what the diagram portrays
- Each E2E Diagram description must be clear, concise and unambiguous. The description shall include, as a minimum, a summary of the business capability being portrayed.
- Each L1 E2E Activity can be linked to one or more process diagrams
- Each L2 E2E Activity can be linked to one or more process diagrams

18.3.1.2 Object Naming Conventions

Objects in the E2E diagrams shall have a concise and intuitive name and description according to the following standards:

- The word "to" in a Level 0 Activity name must be immediately preceded and immediately followed by a "-" that adjoins two Level 0 Activity words into a single term
- Each E2E Activity must have a description.
- Each E2E Activity word must begin with a capital letter.
- Each E2E Activity acronym must be included in the Acronym Definitions.
- Each new term in an E2E definition must be included in the Term Definitions.

18.3.2 E2E Diagram Best Practices

This section discusses best practices, including lessons learned from previous architecture development efforts and common modeling pitfalls to avoid.

18.3.2.1 E2E Diagram Lessons Learned

- Use MS Word to spell and grammar check descriptions.
- Ensure that Level 0 acronyms are defined in the AV-2.
- During the workshops, color code adds, changes and deletes to identify content changes for the Stakeholders.

18.3.2.2 E2E Diagram Common Pitfalls

- Having ambiguous descriptions for E2E Activities.
- Ineffective use of diagram space by making the object layout either too dense or too sparse.
- Using inappropriate colors on the diagram objects.

18.3.3 Previous E2E Business Flow Object Definitions

The adoption of the new approach in developing the E2E models necessitates that E2Es developed utilizing the previous approach get updated. This update will be performed as the priorities for the next E2Es to address are identified. As reference and to aid in the conversion of the E2Es the following two sections describe the objects and modeling conventions previously used in developing the E2Es.

The E2E Business Flow Model consists of a set of E2E Business Flow Diagrams, which begin with a COTS/ERP business capability which, in turn, contains sequenced business flow elements that are mapped to BEA BPM Process Steps. The following are the object names used in the E2E Business Flow Diagram:

- Level 0 E2E Business Flow – COTS/ERP Business Capability
- Level 1 E2E Business Flow – Sequenced Business Flow Step
- BPM Process Step – OV-6c Process
- BPM Child Process Step – OV-6c Process
- BPM Process Group – Grouping of ten or more BPM Processes

Figure 18-2 depicts an E2E Business Flow Diagram and the positioning of the objects.

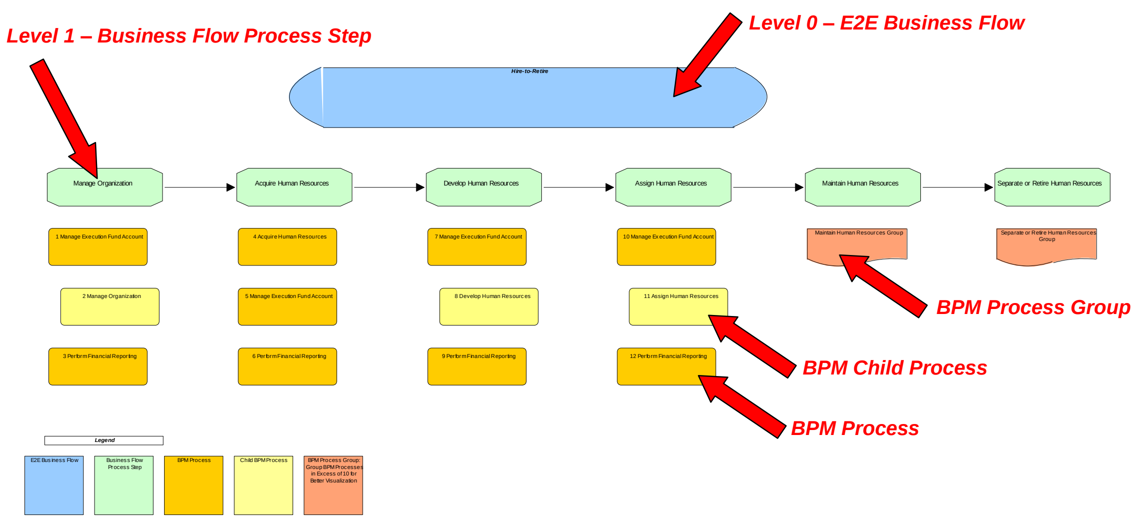


Figure 18-2, Objects of an Old E2E Business Flow Diagram

18.3.4 Previous E2E Business Flow Modeling Conventions

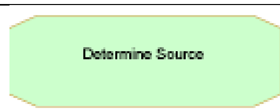
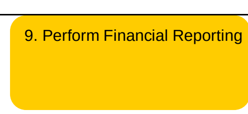
This section is a brief overview of the previous E2E Business Flow Diagram and the notation used to prepare the E2E Business Flow Diagram. This section is provided only as reference for E2Es developed in the past and does not reflect current modeling conventions.

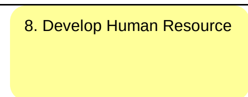
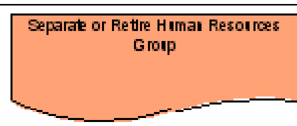
An E2E Business Flow Diagram consists of a set of four graphical elements. These elements enable the development of diagrams that have a familiar look to most business analysts (for example, a flowchart diagram). *The diagram elements are chosen to be clearly distinguishable from each other and to utilize shapes that are familiar to most process architects.*

An E2E Business Flow Diagram is not meant to be a BPMN diagram and does not follow BPMN conventions. An E2E Business Flow Diagram is only a means to map BEA BPM Process Steps to the business capabilities being implemented in the COTS/ERP programs.

While the general flow of objects within a diagram will be from left to right and top to bottom, due to programmatic and space constraints, objects may be connected to other objects in a manner that is logical and readable.

Table 18-1, Previous Modeling Guidelines for a E2E Business Flow Diagram

Element	Symbol	Format
L0 E2E Business Flow		Border: Solid Black Fill: Light Blue Text: Color: Black Font: Arial Size: Variable
L1 E2E Business Flow		Border: Solid Black Fill: Aqua Text: Color: Black Font: Arial Size: Variable
Sequence Flow		Line: Solid Black Text: Color: Black Font: Arial Size: Default
BPM Process Step		Border: Solid Black Fill: Gold

Element	Symbol	Format
		Text: Color: Black Font: Arial Size: Variable
BPM Child Process Step		Border: Solid Black Fill: Yellow Text: Color: Black Font: Arial Size: Variable
BPM Process Group		Border: Solid Black Fill: Light Orange Text: Color: Black Font: Arial Size: Variable

19 SvcV-1 – Services Context Description

19.1 Summary Description

This section describes the Services Context Description, its relationship to other BEA models, the development method, and the modeling guidelines to be followed.

The SvcV-1 Services Context Description model depicts Services, Data Sources, Consumer and Resource Flow needed to implement Services.

19.1.1 Model Purpose

The Services Context Description identifies the leveraged interface specifications utilized by systems to expose capability to the "net" with a separate distinction that identifies an instance of the specification as an interface between a provider and a consumer.

19.1.2 Model Structure

The SvcV-1 model is depicted as a diagram. It comprises Services Container, Data Sources Container, Service Consumer and Services Resource Flows, as described in the following sections. For the BEA, individual SvcV-1 diagrams are developed for each enterprise system implemented using Services. Figure 19-1, SvcV-1 Model for AT&L AV SOA Service, is an example of a SvcV-1 Services Context Description.

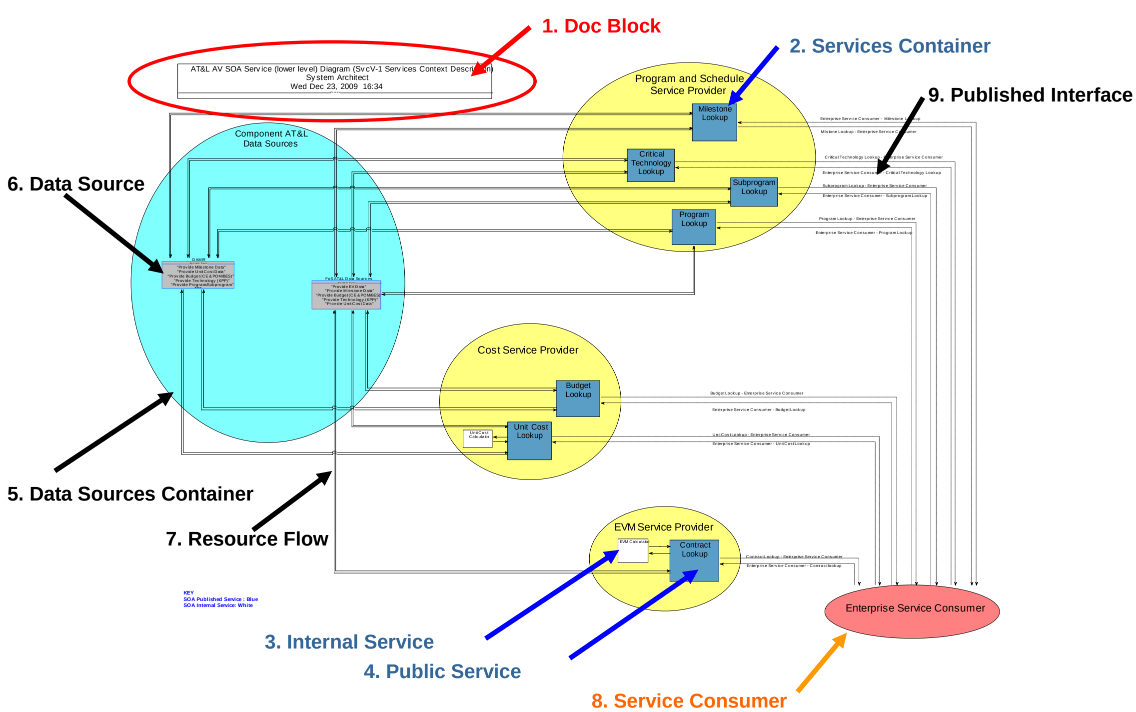


Figure 19-1, SvcV-1 Model for AT&L AV SOA Service

Individual SvcV-1 diagrams are developed for each application using Services. The objects used to represent the SvcV-1 model are numbered as shown in Figure 19-1. The main features of this diagram are:

- **Doc (title) block (1)** is located in the upper left corner of the diagram. The title block contains the diagram name and is named after the focus CBM ("Real Property Lifecycle Management"), type "(SvcV-1 Service Context Description)" and last modification date.
- **Services Container (2)** is the large yellow oval shape in the diagram. This is a container that contains the individual Services. The name of the Services oval represents the purpose of the Services. Each service container is composed of service functions.
- **Internal Service (3)** is a white rectangle contained within the Service Container. Each white rectangle represents a Service that is not exposed to the Public, but is used by another Service. Each internal service is composed of service functions.
- **Public Service (4)** is a blue rectangle contained within the Service Container. Each blue rectangle represents a Service that is exposed to the Public.
- **Data Sources Container (5)** is the large light blue oval shape in the diagram. This is a container that contains the data sources used by the Services. The name assigned to the Data Sources oval represents the purpose of the Services.
- **Data Source (6)** is a grey rectangle contained within the Data Sources Container. Each grey rectangle represents a data source accessed by a Service. The name above the grey rectangle is the name of a Data Source. The names within the grey rectangle is the names of the System Functions that directly accesses the Data Source
- **Resource Flow (7)** is a solid directed line from a Service to a Data Source and from a Data Source to a Service. The directed line from a Service to a Data Source represents the data request and the directed line Data Source to a Service represents the data response. The name assigned to a Service Resource Flow is the concatenation of the from-to-objects.
- **Service Consumer (8)** is the smaller orange oval that represents the user the Services. The name assigned to the Service Consumer is the name of the primary user.
- **Published Resource Flow (9)** is a dashed directed line from a Service Consumer to a Service and from a Service to a Service Consumer. The directed line from a Service Consumer to a Service represents a data request and the directed line Service to a Service Consumer represents the data response. The name assigned to a Public Resource Flow is the concatenation of the from-to-objects.

19.1.3 Relationship to Other BEA Models

As illustrated in Figure 19-2, Relationships Between SvcV-1 and Other BEA Models, the SvcV-1 is related to other BEA models as follows:

AV-1	The scope of the development effort for each CBM for a development cycle, as disclosed in the AV-1, will determine if the SvcV-1 is affected by the release.

AV-2	All SvcV-5 terms with specific meaning must be defined in the AV-2 Terms Definition list. These terms must include, as a minimum, all object types included in the deliverable. All acronyms used in the SvcV-5 Services descriptions must be listed and spelled out in the AV-2 Acronym Definitions report.
OV-2	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.
OV-5	Services are defined to support the Operational Activities.
SvcV-5	OV-5 Operational Activities are linked to the Services in the in the SvcV-5.

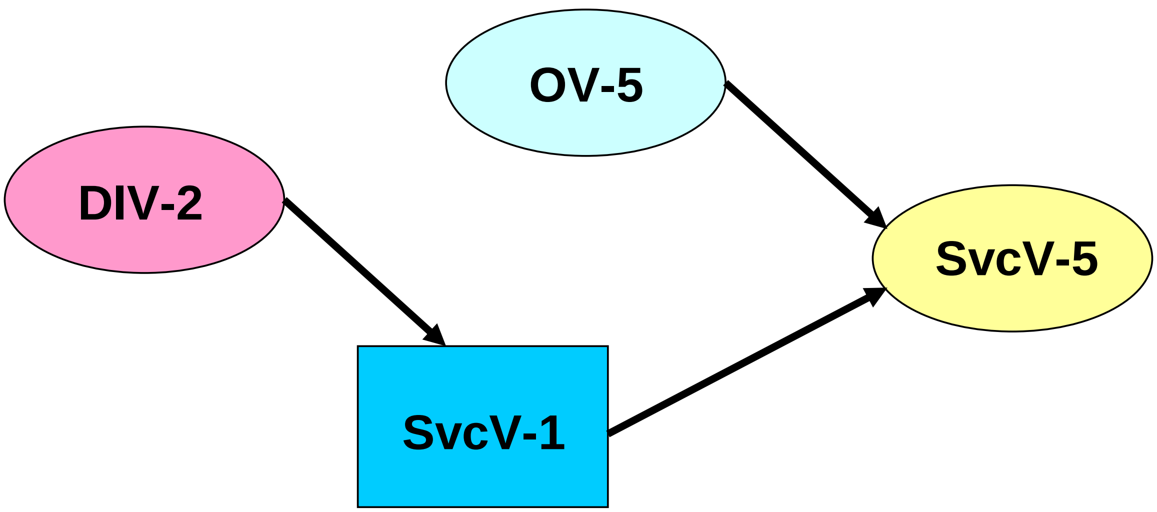


Figure 19-2, Relationships Between SvcV-1 and Other BEA Models

19.1.4 SvcV-1 Model Definitions

The following are definitions of the key elements contained in the SvcV-1:

1. **Services Container:** A graphical representation of a container that houses one or more individual Services.

2. **Service Consumer:** An application, component or other software module that requests a service from a separate application. The service consumer finds the service provider in a service registry, sends a service request and receives a response.

3. **Data Sources Container:** A graphical representation of a container that houses one or more individual Data Sources.

4. **Data Source:** A data repository or database that contains data accessed by System Functions.

5. **Published Service:** A Service that has a Published Interface in a Service Registry that is available to a Service Consumer.

6. **Internal Service:** A Service that is not published and is consumed by another Service.

7. **Resource Flow:** The interface between a Service and the Data Source.

8. **Service Resource Flow Element** is being exchanged between services and includes the attributes of that exchange.

9. **Published Interface:** A public interface to a Service that is available for general consumption by an unanticipated user, Consumer, by being available in a service registry.

19.2 Developing the SvcV-1 Models

This section describes the approach to develop, extend and maintain the SvcV-1. The SvcV-1 is developed in System Architect as a diagram.

The SvcV-1 model development begins with the development of the OV-5 model and continues after the completion of that model. The development is done in collaboration with the Stakeholders and, when necessary, DoD Program Managers who are responsible for the enterprise systems. Pre-development sessions and formal workshops are held with Stakeholders to identify and define system functionality represented by Enterprise Systems in the BEA.

The System Service View represents the business requirements in the Operational View models. The SvcV-1 model analysis is conducted to ensure, for example, that OV business requirements represented as IEs become system resource flows.

During the pre-development period, the Stakeholders are provided with worksheets to collect information about their enterprise systems from the Program Managers and their community of interest. During development this information is analyzed, along with changes to the OV-5 model to create the SvcV-1 models.

19.2.1 Pre-Development Tasks

The tasks that must be completed prior to SvcV-1 development and/or maintenance are:

1. The CBM representative is provided a copy of the proposed SvcV-1 diagram and definitions. The CBM representative is responsible for identifying the CBM mission thread, defining information and data standards needed to implement the Resource Flows and defining the Operational Activities provide in support of those Resource Flows.

2. The CBM representative determines the appropriate enterprise services and, if applicable, contacts the responsible Program Manager for the following information about their enterprise systems:

    - Enterprise service name and description
    - Operational Activities performed

- System Functions that support the services and implement the Operational Activities
- Resource Flow services and definition
- Service Resource flows with definitions
- Services and associated Data Elements
- Applicable data standards needed for implementation

3. The Program Managers provide information on their respective Enterprise services to their CBM representative. The CBM representative reviews the information and provides it to the BEA Development Team for analysis and inclusion in the BEA.

4. As part of the pre-workshop activity the BEA Development Team works with the CBM representative to ensure that any new business requirements are represented in the Operational View business requirements. A tight link between the SvcV-1 and the OV-5 is maintained to ensure that the System Service View is integrated with the CBM business requirements.

19.2.2 Development Tasks

The development and maintenance of the Services Context Description Diagram is accomplished in facilitated workshops that include Government SME participation to address content and validate results. The following subsections describe the approach used to develop the SvcV-1 in the BEA. Each subsection sets forth the specific tasks that must be accomplished to in each stage of the development phase. Although most of these steps are sequential, it is common to start some steps before a previous step is completed.

19.2.2.1 SvcV-1 Analysis Tasks

Prior to any changes, an impact analysis is conducted to assess the impact of new or revised business requirements to the SvcV-1. The following impact analysis tasks are performed:

1. For creation or any changes to Services Containers:
    - Assess impact to OV-5 Operational Activities.
    - Verify that the services and their related service functions are supported by the operational activities resident in the Operational Node Tree.
    - Determine if the Operational Node Tree definition needs to be refined.
    - Assess impact to OV-5.
    - Assess impact to other SvcV-1 diagrams.
    - Verify that other SvcV-1 diagrams support the required Services.
2. For creation or changes to Operational Activities:
    - Assess impact to OV-5 models.
    - Assess impact to other SvcV-1s diagrams..
    - Assess impact to CBM Stakeholders.

19.2.2.2 Creating / Modifying SvcV-1 Diagrams

This section describes the approach to develop the SvcV-1.

19.2.2.2.1 Review the SvcV-1 Model for Internal Consistency

Because the OV-5 is so closely linked to the SvcV-1, the SvcV-1 is completed after the OV-5 model is stabilized. The OV-5 and SvcV-1 models must be in accord. The following rules apply for the creation of a valid SvcV-1 diagram.

- Each Service must be associated with a corresponding Operational Node.
- All SvcV-1 enterprise systems must be represented by a corresponding OV-5b ICOM mechanism.

19.2.2.2.2 Create the SvcV-1

The tasks the must be completed to produce a SvcV-1 are:

1. Create a new diagram or open an existing diagram.
2. The following procedures are used for creating the various elements of the SvcV-1:
    - Create Service Consumer
        - Name Service Consumer
    - Create Services Container:
        - Create a Service object
        - Define Operational Activity
        - Map Service to OV-5 Operational Activity
        - Map Service Function to Service
        - Link Service to Service Consumer and vice versa
        - Name linked Resource Flows
        - Link Service Resource Flow Elements to the Service Resource Flow
    - Create Data Sources Container:
        - Name Data Source
        - Map Data Source to System Functions
        - Link Data Source to Service and vice versa
        - Name linked Resource Flows

19.2.2.2.3 Update the SvcV-1

Following analysis of any changes to the OV models, existing SvcV-1 content shall be updated to reflect any impact of these changes. This may require creation or update of System Services, System Resource Flow elements or System Resource Flows to the SvcV-1 model. For example, the addition of a new leaf-level output ICOM in the OV-5b will require the creation of a new system Resource Flow.

The tasks that must be completed to update the SvcV-1 are:

1. Changes to Operational Activity ICOMs:

   - If a new leaf-level input or output ICOM is added to an OV-5b, check to see if an IE was created. If there is an IE, identify the Need Line in the OV-2 where the IE is associated. Identify the System Resource Flow that maps to the Need Line. Determine if there is an existing System Resource Flow Element that maps to the IE. If the System Resource Flow Element exists, map it to System Interface. If not, and the IE is to be automated, create a new System Resource Flow Element and map it to System Interface.
   - If a leaf-level ICOM is deleted from an OV-5b, identify the IE that maps to the ICOM. Identify the Need Line in the OV-2 where the IE is associated. Identify the System Interface that maps to the Need Line. Identify the System Resource Flow Element that maps to the IE. Delete the System Resource Flow Element from the System Interface.
   - If the leaf-level ICOM definition has been revised, identify the supporting System Resource Flow Element, review the definition and revise as necessary.

2. Changes to Operational Activity:

   - If a leaf level Operational Activity is added to an OV-5b diagram, check to determine if an existing System Function may support the activity. If there is a System Function, review definition and revise as necessary. Otherwise, create new System Function to support the Operational Activity if it is to be automated.
   - If a System Mechanism is added to a leaf-level operational activity on any OV-5b diagram, check to see if the system exists. If not, create a new System and add services or system function associated to the Operational Activity.

3. Changes to Operational Node

   - Assess impact to System Node: Determine if System Node exists or one has to be created. If it exists, verify that the definition supports the Operational Node and revise as necessary. If the node does not exist, create node on each CBM specific SvcV-1 based on revisions to the OV-2 model.

4. Changes to Need Line on the Operational Node:

   - If a Need Line is deleted, identify the System Interface that maps to the Need Line and delete.
   - If a Need Line is added, determine if an existing System Interface maps to the Need Line. Create a new System Interface if there is not an existing System Interface, provided the Need Line represents an automated exchange.
   - If an unexpected System Resource Flow appears on the SvcV-1, an analysis of the OV models will be required to determine the corrective action that must be made to the OV or SvcV-1 models.

5. Changes to IE:

   - If an IE is added, identify the Need Line where the IE will be added. Determine if there is an existing System Resource Flow Element or if a new System Resource Flow Element needs to be created. Link the new System Resource Flow Element to the Service or published resource flows.

- If an IE is deleted, identify the System Interface that maps to the Need Line where the IE is being deleted. Delete the System Resource Flow Element from the System Interface.

19.2.3 Post-Development Tasks

These tasks are performed after changes to the SvcV-1 by the Stakeholders to ensure integration of the architecture.

1. When a SvcV-1 diagram is updated, make updated diagrams available to the Stakeholders to review, identify corrections, and finalize acceptance of the model.
2. Verify all SvcV-1 acronyms and terms are in the AV-2.
3. Verify that all SvcV-1 Enterprise-level Services are in the Enterprise Transition Plan.
4. Review changes to the OV-5 model and follow the SvcV-1 tasks that are mentioned in section 11.2.2.2.

Moreover, Quality Assurance checks are conducted to prepare for the formal quality assurance tasks conducted by IV&V group and Stakeholder review. The SvcV-1 Model Checklist is used to verify that the content is in accordance with the SV modeling guidelines. The major checks to ensure compliance include:

1. Spelling of all objects within the diagram is correct.
2. Service Resource Flows are connected to Services.
3. Services are within the Services Container.
4. Services are associated with a Service Consumer.
5. Each Service references at least one Operational Activity.
6. Each Service has at least one Service Function.
7. Each Service Resource Flow has a definition.
8. Each Service Resource Element has a definition.
9. Complete the SvcV-1 Checklist.

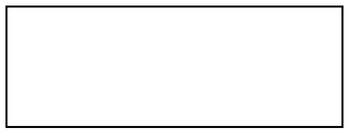
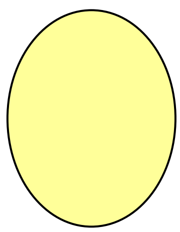
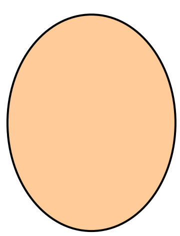
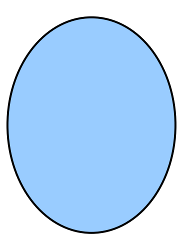
19.3 Modeling SvcV-1 Models Using SA

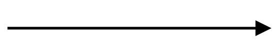
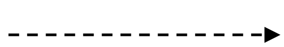
19.3.1 Modeling Conventions

19.3.1.1 Use of Color, Size and Lines in Diagram

The following modeling conventions must be used to create the SvcV-1. The SvcV-1 diagrams use a standard color scheme, font and line size as follows:

Table 19-1, Modeling Guidelines for the Service Context Description Diagram

Element	Symbol	*Format*
Doc Block	*Text Box*: 	*Position: Upper Left Corner* *Border: Solid Black* *Fill: None* *Text:* *Color: Black* *Font: Arial* *Size: Default*
Services Container	*Elliptical:* 	*Border: Solid Black* *Fill: Yellow* *Text:* *Color: Black* *Font: Arial 10, black* *Size: Default*
Service Consumer	*Elliptical:* 	*Border: Solid Black* *Fill:* *Light Orange* *Text:* *Color: Black* *Font: Arial 10, black* *Size: Default*
Data Sources Container	*Elliptical:* 	*Border: Solid Black* *Fill:* Light blue with a black border. *Text:* *Color: Black* *Font: Arial 10, black* *Size: Default*

Element	Symbol	*Format*
Internal Service	*Rectangle:* 	*Border: Solid Black* *Fill:* White Box with a black border. *Text:* *Color: Black* *Font: N/A* *Size: N/A*
Data Source	*Rectangle:* 	*Border: Solid Black* *Fill:* Light gray fill and a black border. *Text:* *Color: Black* *Font: N/A* *Size: N/A*
Resource Flow	*Arrow:* 	*Border: Solid Black* *Fill:* *N/A* *Text:* *Color: Black* *Font: N/A* *Size: N/A*
Public Interface	*Rectangle:* 	*Border: Solid Black* *Fill:* Blue Box with a black border. *Text:* *Color: Black* *Font: N/A* *Size: N/A*
Published Interface	*Arrow:* 	*Border: Dashed Black* *Fill:* *N/A* *Text:* *Color: Black* *Font: N/A* *Size: N/A*

Additional guidance that applies to a system resource flow is provided below.

- System Resource Flow labels will be placed, where possible, above the horizontal line and closest to either the arrowhead or 90 degree angle.
- System Resource Flow line intersections are permissible, but should be minimized to the extent possible.

19.3.1.2 Diagram Conventions

Each SvcV-1 diagram shall have a Diagram Description contained within the Description block of the diagram properties that describes the purpose of the diagram, the CBM enterprise and federally mandated systems Information.

- A Doc Block representing header information for the diagram (including the diagram name and date last updated) is placed at the top center of every diagram. The Doc Block is enlarged so there are no truncation indicators (dots) indicating text is not visible. The Doc Block is a box with no fill color and has a black border.
- The SvcV-1 diagram shall not have a border.

19.3.1.3 Object Naming Conventions

Each SvcV-1 diagram uses standard object naming conventions as follows:

- The Service Container name shall the name of the primary provider of the services.
- The Service name is the name of the registered service.
- The Service Consumer name is the name of the initial Consumer of the service.
- Each System Resource Flow name shall be a concatenation of the “from-to” object names.

19.3.2 Modeling SvcV-1 Objects

The following guidelines are used to create or modify the SvcV-1.

- A Service name should be an illustrative name that indicates what the Service performs; such as "Acquisition Program Budget Lookup" instead of "Budget Lookup".
- Modeling objects shall not have truncated entries on the diagram.
- All Container labels shall be centered at the top of the Container border and the label should not fall outside the boundary of the ellipse.
- Each Service name shall be title-case, use only approved acronyms, non-plural and use no special characters except “-”.
- All Service name labels shall be centered in the Service rectangle and the label should not fall outside the rectangle boundary.
- All Data Sources must be contained within their associated Data Sources Container elliptical boundary.
- Each Data Source name shall be upper-case or title-case, use only approved acronyms, non-plural and use no special characters except “-”.

- The SvcV-1 will only include External Services that interface to the Service Consumer.
- System Resource Flow lines are not permitted to traverse intermediate Services. To the maximum extent possible, System Resource Flow lines shall not cross intermediate Services.
- Service Resource arrows shall be black with black filled arrowheads.
- Each end of a System Resource Flow line must be connected to a Service.

19.3.3 SvcV-1 Best Practices

This section discusses best practices, including lessons learned from previous architecture development efforts and common modeling pitfalls to avoid.

19.3.3.1 SvcV-1 Lessons Learned

The following lessons learned have been and serve as the basis for the SvcV-1 checklist in Appendix B.

- Ensure that the SvcV-1 analysis occurs concurrently with OV-5 development; ensure system mechanisms are properly assigned to Operational Activities that they automate.
- Regular and early communication with other BEA Development Team s is needed to assess impact of proposed changes in other models on the SvcV-1. The SvcV team will actively participate in the pre-analysis workshops to ensure that changes in the Operational View business requirements can be properly reflected in the Services Context Description.
- Standard color coding of diagrams during the workshop is useful for participants to identify where content was added, changed or deleted. Standard color coding should be in line with that used in the BEA Compare reports.
- All exception reports must be reviewed and resolved.

19.3.3.2 SvcV-1 Common Pitfalls

The following common mistakes in the use of SA that could affect the development lessons learned have been and also serve as the basis for the SvcV-1 Checklist in Appendix B:

- Provide enough time in the original schedule to incorporate late changes in the OV-5 models. Often adequate time to complete impact analysis or post development tasks to modify the SvcV-1 is not available.
- The SvcV-1 diagrams are not reviewed in the HTML/Scalable Vector Graphic (SVG) rendition until after model stabilization, so flaws that do not show up in SA, such as superfluous line segments on SRFs, are exposed in the web-based version of the architecture. Superfluous line segments are eliminated on the SRFs by using the “reduce line segment” feature in SA.

20 SvcV-5 – Operational Activity to Service Traceability Matrix

20.1 Summary Description

This section describes the Operational Activity to Service Traceability Matrix, its relationship to other BEA models, the development method and the modeling guidelines to be followed.

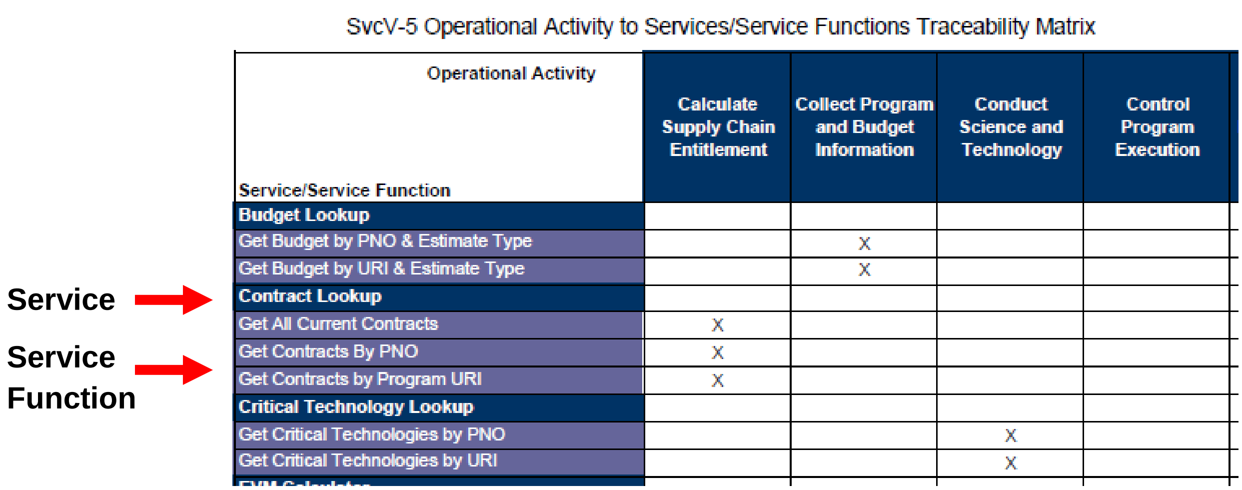
20.1.1 Model Purpose

The Operational Activity to Service Traceability Matrix depicts the relationships between the Operational Activities in the OV-5 Activity Models and the Services.

The DoDAF SvcV-5 is used to meet a program requirement to link Operational Activities and Services. The Enterprise-level Services identified by each Stakeholder are shown on the SvcV-5 matrix, where the Enterprise-level Service supports an Operational Activity.

20.1.2 Model Structure

The SvcV-5 model is depicted as a matrix. It comprises and relates Services to Operational Activities across the BEA. For each matrix cross area or intersection the related Enterprise Service is entered. There can be many Operational Activities related to a single Service. The matrix is illustrated in Figure 20-1, Example of a SvcV-5.



SvcV-5 Operational Activity to Services/Service Functions Traceability Matrix

Operational Activity / Service/Service Function	Calculate Supply Chain Entitlement	Collect Program and Budget Information	Conduct Science and Technology	Control Program Execution
Budget Lookup				
Get Budget by PNO & Estimate Type		X		
Get Budget by URI & Estimate Type		X		
Contract Lookup				
Get All Current Contracts	X			
Get Contracts By PNO	X			
Get Contracts by Program URI	X			
Critical Technology Lookup				
Get Critical Technologies by PNO			X	
Get Critical Technologies by URI			X	

Figure 20-1, Example of a SvcV-5

20.1.3 Relationship to Other BEA Models

As illustrated in Figure 20-2, Relationships Between SvcV-5 and Other BEA Models and described in the table below, the SvcV-5 is related to other BEA models as follows.

AV-1	The scope of the development effort for each CBM for a development cycle, as disclosed in the AV-1, will determine if the SvcV-5 is affected by the release.
AV-2	All SvcV-5 terms with specific meaning must be defined in the AV-2 Terms Definition list. These terms must include, as a minimum, all object types included in the deliverable. All acronyms used in the SvcV-5 Services descriptions must be listed and spelled out in the AV-2 Acronym Definitions report.
OV-5	OV-5 Operational Activities are linked to the Services in the in the SvcV-5. In addition, Enterprise-level Services are Mechanisms on the Operational Activities.
SvcV-1	The SvcV-1 Services and supporting Operational Activities are mapped the SvcV-5 matrix.

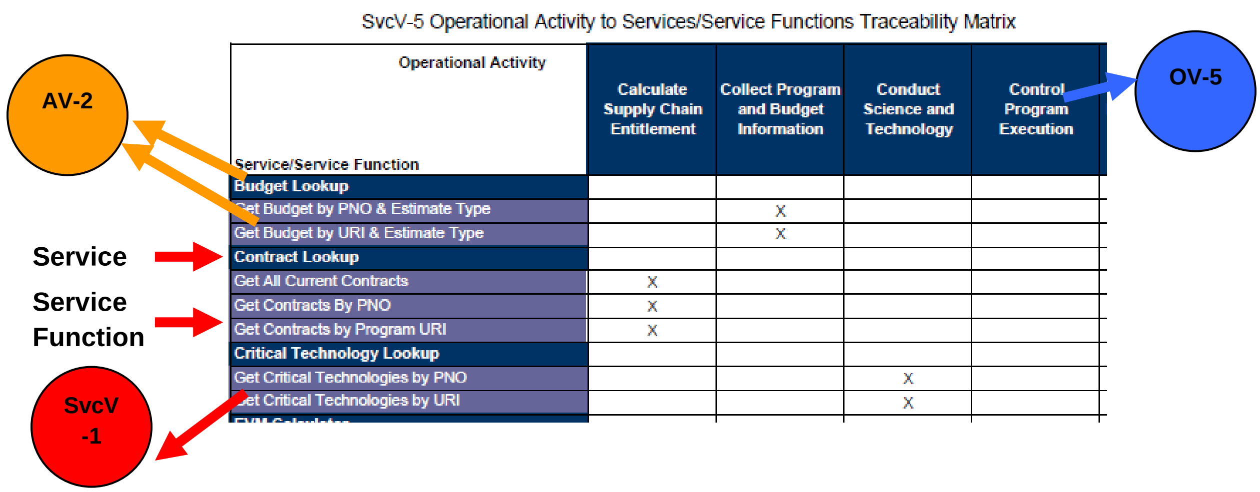


Figure 20-2, Relationships Between SvcV-5 and Other BEA Models

The Stakeholders and the BEA Development Team coordinate the components of the SvcV-5 to ensure integration with other BEA Models.

20.1.4 SvcV-5 Model Definitions

The following are definitions of the key elements contained in the Operational Activity to Service Traceability Matrix.

1. **Service:** Services represent DoD functionality in a SOA. In the BEA, these services are identified as being enterprise services that has been identified as the standard across the DoD.

2. Service Function: Processes performed by a system.

Note: Service Functions are system functions that are flagged as a service.

3. **Operational Activity:** An action performed in conducting the business of an enterprise. This is a general term that does not imply a placement in a hierarchy or a timing sequence (for example, 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).

20.2 Developing the SvcV-5 Models

This section describes the approach to develop, extend and maintain the SvcV-5. The SvcV-5 is developed in an Access Database with a web-based front-end using data exported directly from System Architect and is produced as spreadsheet.

A single enterprise matrix represents the SvcV-5 for all CBMs. The SvcV-5 provides an integrated architecture depiction of the relationships of Operational Activities to Systems. Through the mapping of Services to Operational Activities there is an indirect link between System Functions and Business Capabilities.

20.2.1 Pre-Development Tasks

The tasks that must be completed prior to SvcV-5 development and/or maintenance are:

1. Verify that BEA enterprise systems are included in the ETP.
2. Identify leaf-level Operational Activities that are on OV-5 diagrams.
3. Collect Service information from CBM team leads for each leaf-level Operational Activity on an OV-5 diagram that may be automated.
4. Verify the mapping of Services to leaf-level Operational Activities with the CBM team leads.
5. Verify the mapping of Service Function to Services with the CBM team leads.
6. Identify changes in the OV-5 and SvcV-1 that may impact the SvcV-5.

20.2.2 Development Tasks

The development and maintenance of the Operational Activity to Service Traceability Matrix is accomplished in facilitated workshops that include Government SME participation to address content and validate results. The following subsections describe the approach used to develop the SvcV-5 for the BEA. Each subsection sets forth the specific tasks that must be accomplished to in each stage of the development phase. Although most of these steps are sequential, it is common to start some steps before a previous step is completed.

20.2.2.1 Creating/Modifying the SvcV-5 Models

This section describes the approach to develop the SvcV-5. To create the SvcV-5 matrix, the following tasks are performed:

1. Analyze definitions of Operational Activities and associated ICOMs.

2. Identify and create Services that will support leaf-level Operational Activities on OV-5 diagrams. Services must consume the inputs and produce the outputs of the Operational Activity in which they are linked.

3. Analyze Service Function name and definition provided by CBM team leads to ensure they support the leaf-level Operational Activities on the OV-5 diagrams.

4. Verify enterprise systems as Mechanisms to Operational Activities.

5. Verify enterprise system shares at least one Service Function with the Operational Activity where it appears as a mechanism.

6. Verify mapping of Operational Activities to Business Capabilities.

7. Link Services to Operational Activities

    - Use the SvcV-5 Service to Operational Activity matrix browser to link a Service to an Operational Activity.
    - To ensure accurate generation of this SvcV-5 report, detailed connections between the OV and SV models are necessary.
        - The Operational Activity must show the CBM as a Stakeholder (stakeholder tab)
        - The Operational Activity must be assigned to a Service AND a Service Function assigned to the Operational Activity must be assigned to at least one of the services that are assigned to the Operational activity. In other words, the Service and OA must have at least one service function in common.

Finally, the actual generation of the SvcV-5 matrix is automated.

20.2.2.1.1 SvcV-5 Matrix Coordination with Stakeholders

The tasks that must be completed to coordinate with the Stakeholders are as follows.

1. Coordinate with Stakeholders

    - Print copy of SvcV-5 for Stakeholders to review proposed changes and confirm linkages.

2. Coordinate with BEA SvcV-1

    - Ensure that any System Function assigned to an enterprise system in the SvcV-5 is properly aligned to the corresponding Service on the SvcV-1 diagrams.
    - Ensure that Service (System) Functions are only assigned to leaf-level Operational Activities on the OV-5 diagrams.
    - Ensure that Service (System) Functions are not assigned to any OV-5a Operational Activity Decomposition Tree only Operational Activities.
    - Ensure that any enterprise system that links a System Function to an Operational Activity is a Mechanism on that Operational Activity in the OV-5b diagram (with the exception of a Family of Systems).

3. Coordinate with BEA AV-2

- Ensure that all enterprise system acronyms are expanded correctly in the Long Name attribute of the System Entity and in the AV-2.

20.2.3 Post-Development Tasks

These tasks are performed after changes to the SvcV-5 by the Stakeholders to ensure integration of the architecture.

1. Ensure that Names of Operational Activities, Service Functions and enterprise services are current and accurate.

2. Add rationalization to the SvcV-5 to explain any anomalies.

20.3 Modeling the SvcV-5 Models Using SA

22.1.1 SvcV-5 Modeling Conventions

The following modeling conventions shall be used to create the SvcV-5.

- The BEA creates a single SvcV-5 matrix that represents all CBMs and as well a single diagram for each CBM. A Microsoft Excel worksheet containing one ALL View and CBM tabs represents the final SvcV-5 model. The SvcV-5 model includes Business Capabilities, and identified Enterprise Systems where a System Function supports an Operational Activity. Please note that the ALL View SvcV-5 is very large and can only be legibly viewed in electronic format. A non-graphic representation of the SvcV-5 is also created in Excel to facilitate sorts and definition review of the SvcV-5 components and is titled, "SvcV-5 Service to Operational Activity Matrix Definitions". This spreadsheet includes definitions for the following objects: CBM stakeholder, Business Capability, Operational Activity, System Function, and System Entity. The first worksheet tab displays the SvcV-5 in a vertical format and the remaining tabs are displayed in this order; Business Capability, Operational Activity, System Function and System Entity.

20.3.1.1 Modeling Use of Color, Size and Lines in the SvcV-5 Matrix

The following color, size and line conventions are used to create the SvcV-5.

- The title of the model is “SvcV-5 Operational Activity to Services/Service Function Traceability Matrix” and appears above the matrix in the spreadsheet header.
- The first row of the SvcV-5 matrix lists the corresponding Operational Activities. Thus, each column of the SvcV-5 matrix represents an Operational Activity intersection.
- The first column of the SvcV-5 lists the Service and supporting Service Functions groupings. The first row of the group is the Service name. It will have a dark blue background. The following rows in the group are the Service Function(s). They are a lighter blue in color.
- The Operational Activities are sorted across the x axis by CBM so that enterprise services that are relevant to a specific CBM are clustered near each other on the SvcV-5. NOTE: Currently, only AV has services represented in the BEA.
- All data cells on the ALL View tab are white.

- In addition to the standard ALL View, the SvcV-5 matrix is also created for each CBM. In the CBM only diagram Services and Operational Activities are both sorted by CBM so that enterprise systems that are relevant to a specific CBM are clustered near one another.

20.3.1.2 Matrix Diagram Conventions

The following matrix diagram conventions are used to create the SvcV-5.

- The names of Operational Activities and Services in the SvcV-5 should be consistent throughout the encyclopedia.
- All updates to the SvcV-5 matrix are implemented through System Architect's Matrix browser, SvcV-5 System Function to Operational Activity file.
- The SvcV-5 matrix only contains relationships between leaf-level Operational Activities, Services and Service Functions.
- Every Service Function mapped to an Operational Activity shall be reflected in the SvcV-5 properties tab of the Operational Activity.
- Each Service Function must be mapped to at least one leaf level Operational Activity on an OV-5 diagram.
- The SvcV-5 should only list leaf-level Operational Activities from the OV-5 diagram that maps to a Service.

20.3.1.3 Object Naming Conventions

The following guidelines are used to create or modify the SvcV-5.

- The first word and all the main words in Service, Service Function and Operational Activity names should have initial capitals, and all the joining words should be left in lower case.

20.3.2 Modeling SvcV-5 Objects

20.3.3 SvcV-5 Best Practices

This section discusses best practices, including lessons learned from previous architecture development efforts and common modeling pitfalls to avoid.

20.3.3.1 SvcV-5 Lessons Learned

The following lessons learned have been and serve as the basis for the SvcV-5 in Appendix B:

- Generate the SvcV-5 AFTER the OV-5 is stabilized.

20.3.3.2 SvcV-5 Common Pitfalls

The following are common mistakes in the use of the SvcV-5 report generator that could affect the development lessons learned have been and also serve as the basis for the SvcV-5 model checklist in Appendix B:

- Making changes after the SvcV-5 Change Requests have been voted upon.

In general, ensure that all the services, service function, operational activity, system entity, stakeholder, and business capability links among the operational views and system view are correct.

21 CV-2 – Capabilities Taxonomy

21.1 Summary Description

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.

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 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. 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 addition to the capability nomenclature, appropriate quantitative attributes and measures for that specific capability or function may 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.

21.1.1 CV-2 Model Purpose

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.

21.1.2 CV-2 Model Structure

The CV-2 has no mandated DoDAF v2.0 structure although the architectural data must be able to support the representation of a structured/hierarchal list. The CV-2 structure in the BEA is a hierarchy diagram that is similar in construct to the OV-5a Operational Activity Decomposition Tree.

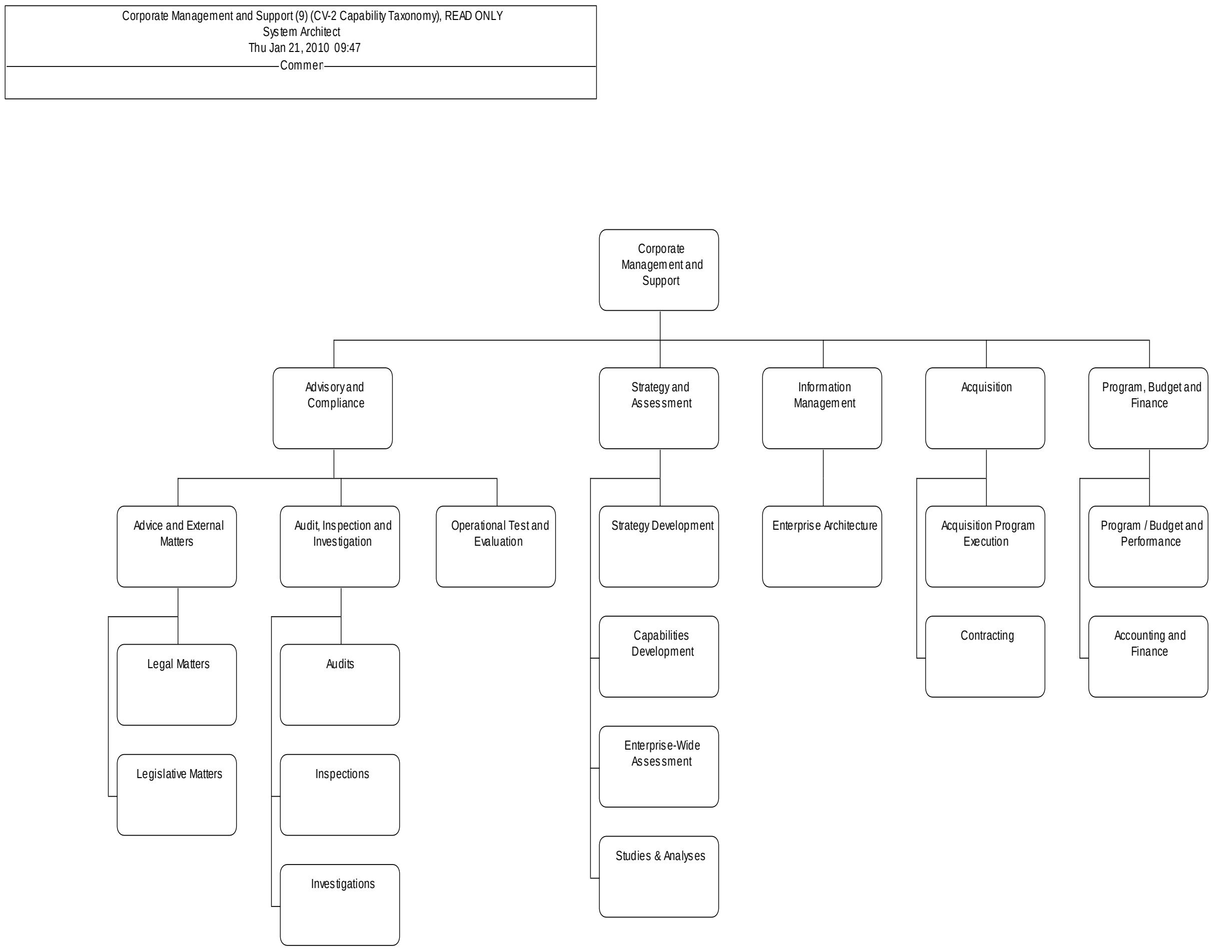


Figure 21-1, Example of a CV-2 Model

As indicated in Figure 21-1, Example of a CV-2 Model, the objects or main features used to represent the Capability Taxonomy are:

- **Doc (title) block (1),** located in the upper left corner of the diagram. The title block contains the diagram name and type in the format "A83 Manage General Ledger Transactions (OV-5b Operational Activity Model)", as well as the last modification date. The double gray bars indicate that the diagram is a decomposition of a parent activity.
- Capabilities **(2)**, **shown as** the yellow rectangular shapes in the diagram.

In addition to the hierarchy diagram, the capabilities with their definitions are listed.

21.1.3 Relationship to Other BEA Models

Integrated architectures provide a structured and organized approach for defining capabilities and understanding the underlying relationships and requirements for achieving those capabilities. The CV-2 is integrated into the BEA as indicated in the following table.

AV-2	All Each term used in the CV-2 definitions with a specific meaning must be included in the AV-2 Term Definitions report. Each acronym used in the CV-2 descriptions must be listed and spelled out in the AV-2 Acronyms Definitions List.
CV-6	The CV06 is the mapping of leaf level CV-2 Capabilities to leaf level OV-5 Operational Activities.
OV-5a	The OV-5 leaf level Operational Activities are mapped to leaf level CV-2 Capabilities.

21.1.4 CV-2 Definitions

The following are definitions of the key elements contained in the CV-2 Capability Taxonomy:

- **Parent Capability:** A Capability that is decomposed into two to nine Capabilities, or Child Capabilities. The definition of the Parent Capability is the sum of the Child Capabilities and serves to set the scope of its decomposition.
- **Child Capability:** A Capability that is a decomposition of a Parent Capability. It represents a functional aspect of its Parent Capability.
- **Hierarchy Chart Connectors:** These lines connect a Parent Capability to its Child Capabilities and show the relationships between Capabilities.

21.2 Developing CV-2 Models

This section describes the approach to develop, extend and to maintain the CV-2 Capabilities Taxonomy in the BEA.

The BEA defined the Department's Business Capabilities to support transformation priorities through the CBMs and the combination of systems and initiative that enable those capabilities. The Business Capabilities are used by the IRB Chairs for investment decision making and portfolio management.

However, DoD has adopted the JCAs as its capability management language and framework for warfighter needs. JCAs are collections of like DoD capabilities functionally grouped to support capability analysis, strategy development, investment decision making, capability portfolio management, and capabilities-based force development and operational planning for the warfighter. In addition, DoD recognized other categories such as Force Support, and Corporate Management and Support might be necessary to address the DoD's enterprise needs.

DoDAF 2 specifies the JCAs as a key component in the federation of DoD Architectures. By continuing to exclude the JCAs from the BEA it prohibits the use of the JCAs as a federation mechanism. In addition, Component, Services and Agencies architectures will remain at risk of not properly federating to the BEA.

In BEA 7.0 the BEA captured the JCA taxonomy and lexicon in the BEA along with their alignment to existing BEA operational activities per DoDAF 2 DM-2. Depicting the JCAs in the BEA enables a leveraging of the relationship between the two such that IRBs can more readily accomplish Portfolio Management and alignment of enterprise architectures across DoD components, services and agencies. In addition, this mapping depicts how the BEA supports the Warfighter's need.

The BEA CV-2 is a hierarchical representation of the JCAs in this release of the BEA. In future release, the Business Capabilities will be analyzed for inclusion into the CV-2.

21.2.1 Pre-Development Tasks

The CV-2 Capabilities Taxonomy is developed from the Joint Capability Area Management Systems (JCAMS). JCAMS is the authoritative Department of Defense database for the JCA taxonomy and lexicon. The latest version of the JCAs are downloaded from this site and reviewed for changes to the current CV-2.

In a future release, the Business Capabilities will be analyzed for inclusion into the CV-2. The CBMs must determine the taxonomy and lexicon of the Business Capabilities and work with the Joint Staff/J7 to add the Business Capabilities into the JCAs.

21.2.2 Development Tasks

21.2.2.1 Analysis Tasks

The current CV-2 is compared to the latest JCAs downloaded from the JCAMS site. All additions, changes and deletions are noted on a spread sheet that is used to create or modify the CV-2 Models.

The CBMs must determine the taxonomy and lexicon of the Business Capabilities for incorporation into the CV-2. This requires identification of the top functional tier and their functional decomposition. In addition, a decision will be required to create a new taxonomy for the Business Capabilities or to work with the Joint Staff/J7 to add the Business Capabilities into the JCAs.

21.2.2.2 Creating / Modifying CV-2 Diagrams

21.2.2.2.1 Review the CV-2 Model for Internal Consistency

The current CV-2 Diagrams are compared and contrasted to the latest JCAs to assess the impact of the changes.

Business Capabilities definitions and linkage to operational activities are validated against requirements submitted by the CBMs.

21.2.2.2.2 Create the CV-2

New high level JCAs may cause new CV-2 Models to be created. The new CV-2 Models are direct images of the new JCAs.

Business Capabilities are not included in the CV-2 models for this deliverable. Each new functional tier defined for Business Capabilities will require a CV-2 diagram to reflect the functional tier and it’s decomposition,

21.2.2.2.3 Update the CV-2

Changes and deletions to the CV-2 Models are made to conform to the JCAs.

Business Capabilities are not included in the CV-2 models for this deliverable.

21.2.3 Post-Development Tasks

The updated CV-2 Models are compared to the latest JCAs to ensure that the same hierarchical representation is presented. Each JCA and Business Capability must have a name and definition. All acronyms used in name or definition must be included in the Acronym list.

21.3 Modeling CV-2 Models Using SA

21.2.2 CV-2 Modeling Conventions

The following modeling conventions shall be used to create an efficient and effective CV-2:

21.2.2.1 Use of Color, Size and Lines in a Diagram

Table 21-1, Modeling Guidelines for CV-2 Capability Taxonomy

Element	Symbol	*Format*
Doc Block	Text Box:	*Position: Upper Left Corner* *Border: Solid Black* *Fill: None* *Text:* *Color: Black* *Font: Arial* *Size: Default*
Capability	Rectangle:	*Border: Solid Black* *Fill:* *White* *Text:* *Color: Black* *Font: Arial* *Size: Default*

21.2.2.2 Diagram Conventions

CV-2 Capability Taxonomy

- There is one CV-2 Capability Taxonomy diagram for each functional tier in the BEA SA encyclopedia.
- The CV-2 Capability Taxonomy diagram shall include a diagram description that shall be stored in the Description attribute under Diagram Properties.
- All modeling objects shall have no truncated entries on the diagram.

- If a parent Capability is decomposed on the diagram it should follow the "2 to 9" Capability box rule in that a parent shall be decomposed to at least two, but no more than nine, child Capabilities.
- The Capability box label shall use title case (first letter of each word capitalized, other letters lowercase) should be non-plural (exception approved by BEA development team, and can use only the special character "-". Any acronyms used in the Capability name must be from the approved acronym list that is part of the BEA AV-2. New Acronyms will be added to the Acronym list in the AV-2.
- The Capability box label shall fall within the Capability box border when printed.
- The Capability box label shall not contain truncation indicators (dots) signifying that there is additional text not visible.
- Capabilities must have a definition that is clear, concise and uses active voice.
- The top box of the diagram shall be centered (as permitted by the tool) on the diagram.
- A Doc Block representing header information for the diagram (including the diagram name and date last updated) shall be placed in the upper left-hand corner of every diagram with no white space above or to the left of the Block. No graphic comment shall be included. The Doc Block shall be enlarged so there are no truncation indicators (dots) signifying that there is additional text not visible. The Doc Block shall be a box with no fill color and a black border.
- The diagram is not bounded by a border.

21.2.2.3 Object Naming Conventions

- Capabilities shall be named as verb-noun objects. They should represent succinct expressions of what the Capability provides, suitable to the level of decomposition. The Capability names must be unique and use only approved acronyms, as contained in the BEA AV-2. For new acronyms, the acronyms must be noted and passed to the AV-2 model team lead for inclusion in the model.
- The only special characters allowed are "-" and " ' ".
- Use Title Case; the first letter of each word in an object name shall be uppercase; other letters should be lowercase. Incidental words, such as prepositions within the object name ("with," "at," "in," "and" or "the"), shall be all lowercase.
- Object names shall use the singular form (no plurals) with exceptions approved by the Chief Architect.
- Object names shall be spelled correctly and shall not use future tense.

21.2.3 Modeling CV-2 Objects

The following subsections provide guidelines for the individual elements or components that comprise the Capability Taxonomy.

- All Capabilities must be defined. Definitions should reflect the information transformation, creation and consumption actions performed by the Capability. Each definition must be clear, concise, use active voice, and comprise complete, grammatically correct sentence.

- The Capability label shall begin with a RETURN so that the label does not touch the upper border of the Capability Box (required for SA text formatting).
- The Capability box label must fall within the Capability box border when printed.
- The Capability box border shall be a solid black line.
- The Capability box numbers must be sequential. The Capability Box numbers shall be positioned in the lower right corner of the Capability box.
- Leaf-level Capabilities will be associated with a leaf level Operational Activity where appropriate.
- All Capability modeling decompositions must follow the "2 to 9 Capability box" rule with the exception of the top-level C-0. A parent shall be decomposed to at least two, but no more than nine, child Capabilities

21.2.4 CV-2 Best Practices

This section discusses best practices, including lessons learned from previous architecture development efforts and common modeling pitfalls to avoid.

21.2.4.1 CV-2 Lessons Learned

The following lessons learned have been and serve as the basis for the CV-2 Checklist on Appendix B:

- All leaf-level Capabilities must be specified as such.
- Standard color coding of diagrams during the workshop is useful for participants to identify where content was added, changed or deleted. Standard color coding should be in line with that used in the BEA Compare reports.
- All exception reports must be reviewed and resolved.

21.2.4.2 CV-2 Common Pitfalls

The following are common mistakes made to the CV-2 models in SA that must be avoided. Checking for these mistakes is included in the CV-2 Checklist on Appendix B.

- Ineffective use of diagram space.
    - Capability boxes too large or too small.
    - Diagram overly dense or too spread out.
- Inappropriate color coding of diagram objects.
- Capability diagram description not properly defined.
- Capability definitions do not conform to guidance.
- Incorrect use of acronyms.

22 CV-6 – Capabilities to Operational Activities Mapping

22.1 Summary Description

The CV-6 describes the mapping between the capabilities required and the activities that enable those capabilities. It is important to ensure that the operational activity matches the required capability. The CV-6 DoDAF-described Model provides a bridge between capability analyzed using CVs and operational activities analyzed using OVs. Specifically, it identifies how operational activities perform or fulfill, in whole or in part, various capability elements or requirements. 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.

22.1.1 Model Purpose

The intended usage of the CV-6 includes:

- Tracing capability requirements to operational activities.
- Capability audit.

22.1.2 Model Structure

A CV-6 shows which elements of capability may be utilized in support of specific operational activities by means of a mapping matrix. Until the Business Capabilities and JCAs form a common taxonomy and lexicon in the BEA, separate CV-6s will be created to reflect the relationship between the Capabilities and Operational Activities.

For Business Capabilities it comprises the Business Capability, Business Capability Definition, BEA CBM, BEA Activity Number, BEA Activity and BEA Activity Definition. For the JCAs, it comprises the JCA Number, Joint Capability Area (JCA), Joint Capability Area Definition, BEA CBM, BEA Activity Number, BEA Activity and BEA Activity Definition.

The following CV-6 reflects the mapping of the BEA Activities to the JCAs.

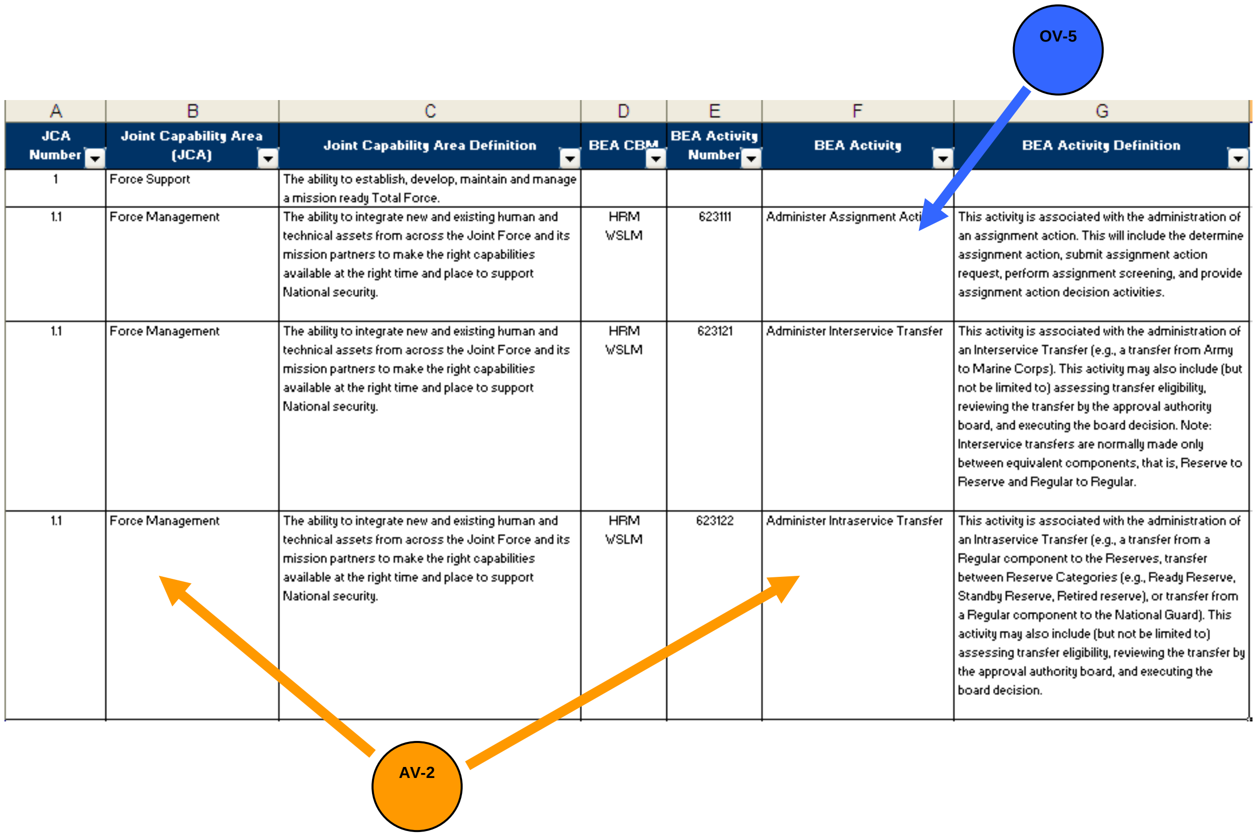
JCA Number	Joint Capability Area (JCA)	Joint Capability Area Definition	BEA CBM	BEA Activity Number	BEA Activity	BEA Activity Definition
1	Force Support	The ability to establish, develop, maintain and manage a mission ready Total Force.				
1.1	Force Management	The ability to integrate new and existing human and technical assets from across the Joint Force and its mission partners to make the right capabilities available at the right time and place to support National security.	HRM WSLM	623111	Administer Assignment Action	This activity is associated with the administration of an assignment action. This will include the determine assignment action, submit assignment action request, perform assignment screening, and provide assignment action decision activities.
1.1	Force Management	The ability to integrate new and existing human and technical assets from across the Joint Force and its mission partners to make the right capabilities available at the right time and place to support National security.	HRM WSLM	623121	Administer Interservice Transfer	This activity is associated with the administration of an Interservice Transfer (e.g., a transfer from Army to Marine Corps). This activity may also include (but not be limited to) assessing transfer eligibility, reviewing the transfer by the approval authority board, and executing the board decision. Note: Interservice transfers are normally made only between equivalent components, that is, Reserve to Reserve and Regular to Regular.
1.1	Force Management	The ability to integrate new and existing human and technical assets from across the Joint Force and its mission partners to make the right capabilities available at the right time and place to support National security.	HRM WSLM	623122	Administer Intraservice Transfer	This activity is associated with the administration of an Intraservice Transfer (e.g., a transfer from a Regular component to the Reserves, transfer between Reserve Categories (e.g., Ready Reserve, Standby Reserve, Retired reserve), or transfer from a Regular component to the National Guard). This activity may also include (but not be limited to) assessing transfer eligibility, reviewing the transfer by the approval authority board, and executing the board decision.

Figure 22-1, Example of a CV-6

22.1.3 Relationship to Other BEA Models

Integrated architectures provide a structured and organized approach for defining capabilities and understanding the underlying relationships and requirements for achieving those capabilities. The CV-6 is integrated into the BEA as indicated in the following table.

AV-2	All Each term used in the CV-6 definitions with a specific meaning must be included in the AV-2 Term Definitions report. Each acronym used in the CV-6 descriptions must be listed and spelled out in the AV-2 Acronyms Definitions List.
CV-2	The leaf level CV-2 Capabilities are mapped to leaf level OV-5 Operational Activities.
OV-5a	The leaf level OV-5 Operational Activities are mapped to leaf level CV-2 Capabilities.



A	B	C	D	E	F	G
JCA Number	Joint Capability Area (JCA)	Joint Capability Area Definition	BEA CBM	BEA Activity Number	BEA Activity	BEA Activity Definition
1	Force Support	The ability to establish, develop, maintain and manage a mission ready Total Force.				
1.1	Force Management	The ability to integrate new and existing human and technical assets from across the Joint Force and its mission partners to make the right capabilities available at the right time and place to support National security.	HRM WSLM	623111	Administer Assignment Acti	This activity is associated with the administration of an assignment action. This will include the determine assignment action, submit assignment action request, perform assignment screening, and provide assignment action decision activities.
1.1	Force Management	The ability to integrate new and existing human and technical assets from across the Joint Force and its mission partners to make the right capabilities available at the right time and place to support National security.	HRM WSLM	623121	Administer Interservice Transfer	This activity is associated with the administration of an Interservice Transfer (e.g., a transfer from Army to Marine Corps). This activity may also include (but not be limited to) assessing transfer eligibility, reviewing the transfer by the approval authority board, and executing the board decision. Note: Interservice transfers are normally made only between equivalent components, that is, Reserve to Reserve and Regular to Regular.
1.1	Force Management	The ability to integrate new and existing human and technical assets from across the Joint Force and its mission partners to make the right capabilities available at the right time and place to support National security.	HRM WSLM	623122	Administer Intraservice Transfer	This activity is associated with the administration of an Intraservice Transfer (e.g., a transfer from a Regular component to the Reserves, transfer between Reserve Categories (e.g., Ready Reserve, Standby Reserve, Retired reserve), or transfer from a Regular component to the National Guard). This activity may also include (but not be limited to) assessing transfer eligibility, reviewing the transfer by the approval authority board, and executing the board decision.

Figure 22-2, Relationships Between CV-6 and Other BEA Models

The Stakeholders and the BEA Development Team coordinate the components of the CV-6 to ensure integration with other BEA Models.

22.1.4 CV-6 Model Definitions

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

In principle, there could be a different CV-6 created for each top functional tier and their functional decomposition. 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. Currently, the BEA depicts nine CV-6s, one for each of the JCA functional tiers.

22.2 Developing the CV-6 Models

This section describes the approach to develop, extend and maintain the CV-6 Capabilities to Operational Activities in the BEA.

Currently, the BEA provides separate enterprise matrixes to represent the mappings of JCA to Operational Activities and Business Capability to Operational Activities.

22.2.1 Pre-Development Tasks

The updated CV-2 Capabilities Taxonomy and the latest OV-5a Operational Activity Decomposition Tree models are captured as work sheets in Excel. These work sheets include object names with definitions and color coding of new objects to the CV-2 and OV-5a. These work sheets are provided to the stakeholders for review and mapping of new objects for the CV-6.

The new objects are reviewed by stakeholders to determine proper mappings between capabilities (business and JCA) and operational activities for the CV-6. One or more leaf level Operational Activities on an OV-5 Diagram can be mapped to the same Capability. Each leaf level Operational Activity can be mapped to one or more Capabilities.

22.2.2 Development Tasks

The development and maintenance of the CV-6 Capabilities to Operational Activities Mapping is accomplished in facilitated workshops that include Government SME participation to address content and validate results. The following subsections describe the approach used to develop the CV-6 for the BEA. Each subsection sets forth the specific tasks that must be accomplished in each stage of the development phase.

22.2.2.1 Analysis Tasks

The stakeholders review the CV-2 and compare each capability to their OV-5a Operational Activities. First the name of a Capability is compared and contrasted to the name of an Operational Activity in an effort to reduce the number of possible mappings. Then, the descriptions of both the Capabilities and Operational Activities are analyzed to determine whether a mapping is appropriate. The CV-6 for the stakeholder is updated by the CBMs to reflect the current mappings of Capabilities to Operational Activities. When a CBM has completed analyzing all of their Operational Activities for mappings to the Capabilities, their updated CV-6 is returned to the BEA architects Creating / Modifying CV-6 Diagrams

22.2.2.1.1 Review the BEA CV-6 Model for Internal Consistency

The BEA architects review the returned stakeholder CV-6 spreadsheets and compare and contrast the changes to the current BEA CV-6. Any questions or possible irregularities are submitted to the appropriate stakeholder for resolution.

22.2.2.1.2 Create the CV-6

The BEA architects merge the individual stakeholder CV-6 spread sheets to create a consolidated BEA CV-6 view.

22.2.2.1.3 Update the CV-6

The BEA architects send the new BEA CV-6 spread sheets to the stakeholders for their review. Any corrections are returned to the BEA architects and the BEA CV-6 is appropriately updated.

22.2.3 Post-Development Tasks

The new BEA CV-6 is given a final review for correctness and completeness. The JCAs in the BEA are validated to be the latest version of the JCAs. The JCA names and the BEA OA names are reviewed for possible mismatches and gaps. As required, the definitions of the JCAs and the BEA OAs are reviewed to assure that proper mapping has occurred.

22.3 Modeling CV-6 Models Using SA

22.3.1 CV-6 Modeling Conventions

The following modeling conventions shall be used to create the CV-6.

The BEA creates separate CV-6 matrixes for JCA and Business Capability mappings. Each matrix represents all CBMs mappings within a Microsoft Excel spreadsheet. The JCA spreadsheet includes: JCA Number, Joint Capability Area (JCA), Joint Capability Area Definition, BEA CBM, BEA Activity Number, BEA Activity and BEA Activity Definition. For Business Capabilities it comprises the Business Capability, Business Capability Definition, BEA CBM, BEA Activity Number, BEA Activity and BEA Activity Definition. There are two versions of the CV-6 for JCA and Business Capabilities with the only difference being the order of the column headings and sort sequence. This permits users to easily determine system mappings based on their familiarity with the capabilities or operational activities within the BEA.

22.3.1.1 Use of color, Size and Lines in Diagram

The following color, size and line conventions are used to create the SV-5.

- The title of the model is "CV-6" with one of the following titles in the spreadsheet header: Joint Capability Area to Operational Activity Mapping, Operational Activity Mapping to Joint Capability, Area Business Capability to Operational Activity Mapping and Operational Activity Mapping to Business Capability.
- The first row of the CV-6 are the column headings with a dark blue fill ( ) with white letters.
- The font for all rows, except for column headings, is Arial, 8 point, black, no fill.
- The column headings and sort sequence for each CV-6 matrix follow:
    - For Joint Capability Area to Operational Activity Mapping spreadsheet the column headings are JCA Number, Joint Capability Area (JCA), Joint Capability Area Definition, BEA CBM, BEA Activity Number, BEA Activity and BEA Activity Definition. This is sorted by JCA then Operational Activity.
    - For Operational Activity Mapping to Joint Capability spreadsheet the column headings are BEA CBM, BEA Activity Number, BEA Activity and BEA Activity Definition, JCA Number, Joint Capability Area (JCA), Joint Capability Area Definition. This is sorted by Operational Activity then JCA.
    - For Business Capability to Operational Activity Mapping spreadsheet the column headings are BEA Capability, BEA Capability Definition, BEA CBM, BEA Activity Number, BEA Activity, and BEA Activity Definition. This is sorted by BEA Capability then Operational Activity.
    - For Operational Activity Mapping to Business Capability spreadsheet the column headings are BEA CBM, BEA Activity Number, BEA Activity, BEA Activity Definition, BEA Capability and BEA Capability Definition. This is sorted by BEA CBM then BEA Activity.

- Each remaining row of the CV-6 represents the mapping of an operational activity to a capability.

22.3.1.2 Matrix Diagram Conventions

The following matrix diagram conventions are used to create the CV-6.

- The names of Capabilities and Operational Activities in the CV-6 should be consistent throughout the encyclopedia.
- All updates to the CV-6 matrix are implemented through System Architect's mapping of Operational Activity to the Business Capability definition.
- The CV-6 matrix contains relationships between Operational Activities and Business Capabilities.
- Every Operational Activity mapped to a Business Capability shall be reflected in the Operational Activity properties tab of the Business Capability.
- Each Operational Activity must be mapped to at least one Business Capability on the CV-6 matrix.

22.3.1.3 Object Naming Conventions

The following guidelines are used to create or modify the CV-6.

- The form of the Operational Activity and Business Capability is a verb followed by a noun.
- The first word and all the main words in Operational Activity and Business Capability names should have initial capitals, and all the joining words should be left in lower case.

22.3.2 Modeling CV-6 Objects

CV-6 objects are not modeled.

22.3.3 CV-6 Best Practices

22.3.3.1 CV-6 Lessons Learned

The CV-2 Capabilities Taxonomy and the OV-5a Operational Activity Decomposition Tree models must be complete before the CV-6 mapping is attempted.

22.3.3.2 CV-6 Common Pitfalls

Using the names and descriptions of the Capabilities and Operational Activities to find candidates for mapping can be subjective.

23 Measures Functional Description

23.1 Summary Description

Measures are quantitative or qualitative ways to characterize and define performance. They provide a tool for organizations to manage progress towards achieving predetermined goals, defining key indicators of organizational performance by allowing performance toward strategic objectives to be monitored over time.

The BEA Measures Model is DoDAF compliant and is consistent with the DoDAF 2.0 DM2 structure.

23.1.1 Model Purpose

The primary purpose of Measures is to 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 can be specified throughout a system development cycle; from Planning to Operation. Some of the uses include:

- Planning –

  Adequacy Analysis. The purpose of an adequacy analysis is to compare Measures associated with a Capability to 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.

  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.

- 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 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.

23.1.2 Model Structure

Measures are captured in the BEA as an AV-2 Definition and then are linked or associated with appropriate BEA objects. The linkage is provided through specific tabs on the Measure Definition. A Measure must be linked to at least one other BEA architectural object.

A Measure must also be linked to a Measure Type and contain a description of the metric or measurement.

The following is a list of the specific tabs for the AV-2 Measure Definition:

- Parent Measure
- Measure Type
- Metric Frequency Type
- Metric Types
- LRP Sources
- Enterprise Standards
- Need Lines
- ICOM Arrows
- Information Exchanges
- Message Flows
- System Nodes
- System Interfaces
- System Data Exchanges

Each tab has a drop list of valid BEA objects from which one or more selections can be made.

23.1.3 Relationship to Other BEA Products

A Measure is related to other BEA products as follows:

LRP	Linked to appropriate LRP.
E2E	Linked to appropriate E2E L1 Business Flows.
Enterprise Standards	Linked to appropriate Enterprise Standards.
OV-2	Linked to appropriate Need Lines.
OV-5b	Linked to appropriate ICOM Arrows, Information Exchanges or System Data Exchanges
OV-6c	Linked to appropriate Message Flows and Data Objects
SV-1	Linked to appropriate System Nodes or System Interfaces

23.1.4 Measure Model Definitions

The following are the new BEA objects introduced by the incorporation of Measures.

- ***Parent Measure*** - A hierarchy of Measures can be developed though a parent-child hierarchical relationship. A Measure must belong to only one hierarchy.
- ***Measure Type -*** The specific category of the Measure such as Physical, Performance, Spatial, etc.
- ***Metric Frequency Type -*** The data or time interval of the Measure.
- ***Metric Types -*** The unit of the Measure such as meters, square feet, etc.
- ***Metric Driver –*** The specific performance measure type that quantifies measurements of a business process or activity used to measure an outcome or an objective. E.g. accuracy, cycle time reduction, timeliness, etc.

23.2 Developing the Measure Model

23.2.1 Pre-Development Tasks

Measures are inextricably tied to SMP Goals. The Stakeholders must determine the Measures to accomplish the Goals within the scope of the BIP. Upon capturing a Measure, it is imperative that it is actually measuring the progress towards a SMP Goal. In general, metrics would ideally be:

- Measurable: This refers to the general ability to realistically *measure* a metric. The measurement of the metric should be determined prior to adding it to the BEA Encyclopedia.
- Objective: While this is similar to "measurable", it is not quite the same. Objective means that all observers will agree on the observed quantity. For example, the number of tall people in a room is

measurable but only if observers agree on an "objective" measure of "tall". Prior to being entered into the BEA Encyclopedia architects should ensure all measures should be vetted for objectivity.

- Nonrelativistic: This simply means measures should not be moving targets i.e. they should have a clear reference point. Prior to being entered into the BEA Encyclopedia architects should ensure all measures have a clear definition and reference point.

- Meaningful: It is entirely possible to create measurable, objective, nonrelativistic, *understandable* measures that are, in effect, meaningless. Stated differently, meeting or exceeding a measure should yield success as defined by the goal or requirement it's measuring. Prior to being entered into the BEA Encyclopedia architects should ensure all measures are meaningful and thus yielding success for the organization or system of concern.

- Understandable: Creating unnecessarily complex measures impairs the ability to interpret its efficacy. Prior to being entered into the BEA Encyclopedia architects should ensure all measures are simple without compromising the aforementioned characteristics

Specifically, architects must ensure a valid Measure Type and description of the metric or measurement is available prior to adding it to the BEA Encyclopedia.

23.2.2 Development Tasks

Architects must ensure each measure is linked to at least one architectural object. In the event measures are only linked to one object, architects must also ensure that object is Measure Type and that it contains a description of the metric or measurement. For in-depth details on adding Measures to the BEA Encyclopedia see section 23.3 Modeling Measures Using SA.

23.2.3 Post-Development Tasks

Post-development actions are contingent upon usage information received from the stakeholders in addition to the overall purpose of creating a measures construct in the BEA.

23.3 Modeling Measures Using SA

23.3.1 Modeling Conventions

The Measures model consists of textual definitions and linkages to other BEA objects. The Measures input window is invoked from the Definitions list and the following is displayed.

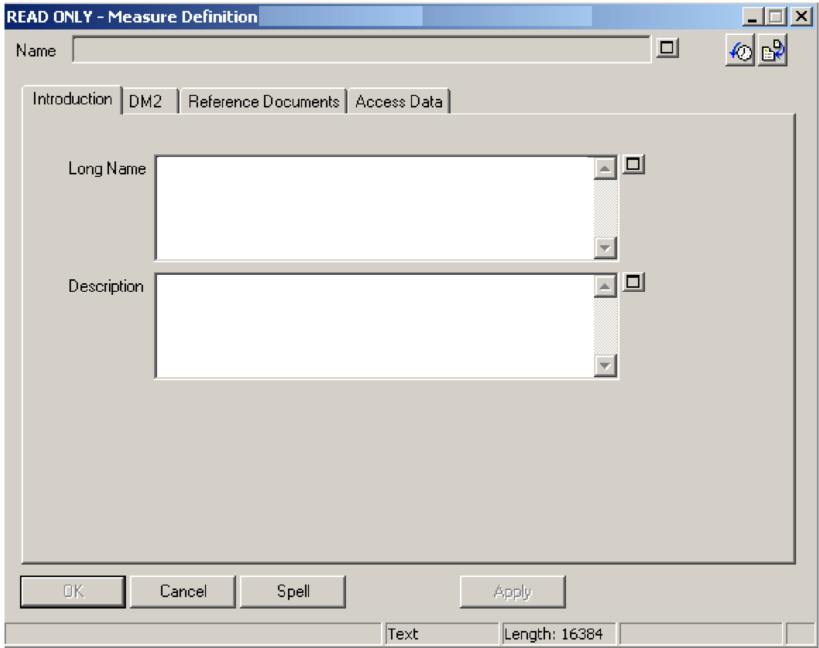


Figure 23-1, Measure Definition Input Window

The following three input data fields must be entered:

1. Name: Enter an alphanumeric short name that uniquely identifies the Measure. This name is used to identify the Measure and to link the Measure to other BEA objects.

2. Long Name: Enter an alphanumeric meaningful business name of the Measure. This name is used by the stakeholders.

3. Description: Enter an alphanumeric field that outlines the purpose of the Measure and how the Measure is to be implemented.

The DM2 tab on the Measures allows a Measure to be linked to appropriate DoDAF DM2 artifacts and conform to DoDAF DM2 Metamodel. The input window allows the display of thirteen input windows one at a time in the following sequence:

1. Parent Measure
2. Measure Type
3. Metric Frequency Type
4. Metric Types
5. LRP Sources
6. Enterprise Standards
7. Need Lines
8. ICOM Arrows
9. Information Exchanges
10. Message Flows

11. System Nodes
12. System Interfaces
13. System Data Exchanges

Windows 1 to 3 have multiple entry data fields and windows 4 to 13 have only a single entry data field. There is only one alphanumeric input data field. All other data fields are populated by selections from drop down lists.

The following figure shows the first window of the DM2 tab which allows a hierarchy of Measures to be developed.

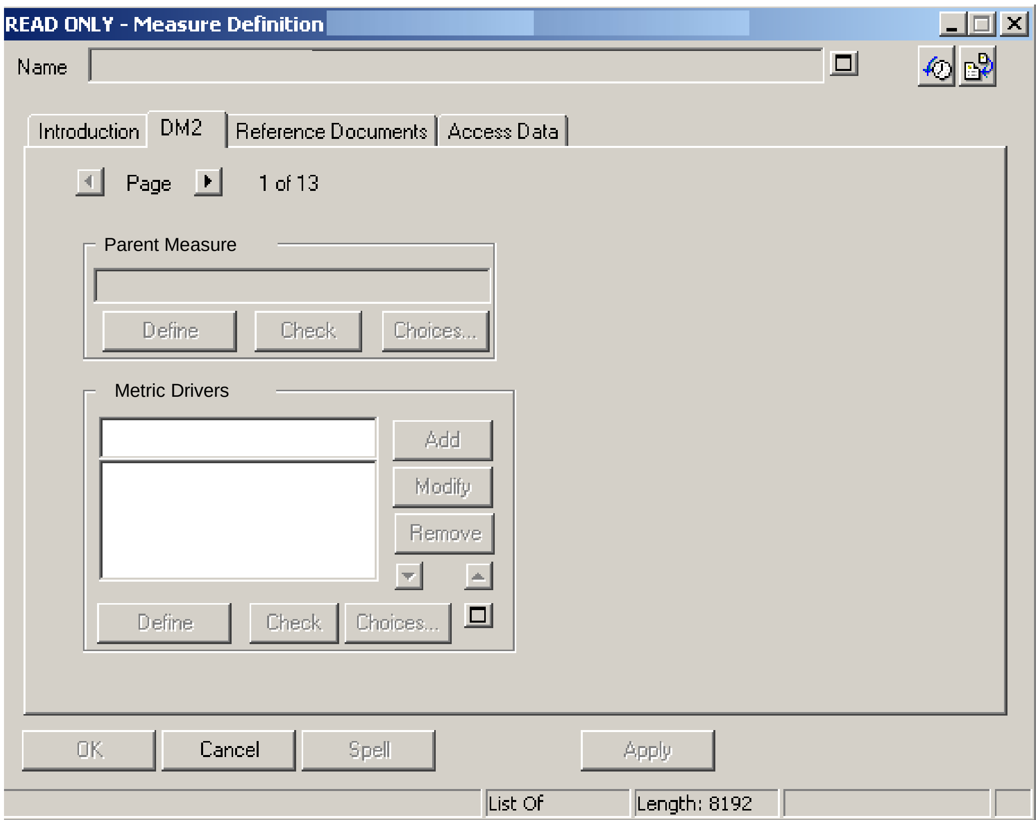


Figure 23-2, Measure DM2 Tab First Input Window

The following two data fields must entered:

1. Parent Measure: The parent of this child must be selected from a drop down list of BEA Measures.
2. Metric Drivers: One or more Metric Drivers must be selected from a drop down list of Metric Drivers.

The following figure shows the second window of the DM2 tab which allows the algorithm for the calculation of the Measure and the category of the Measure to be entered.

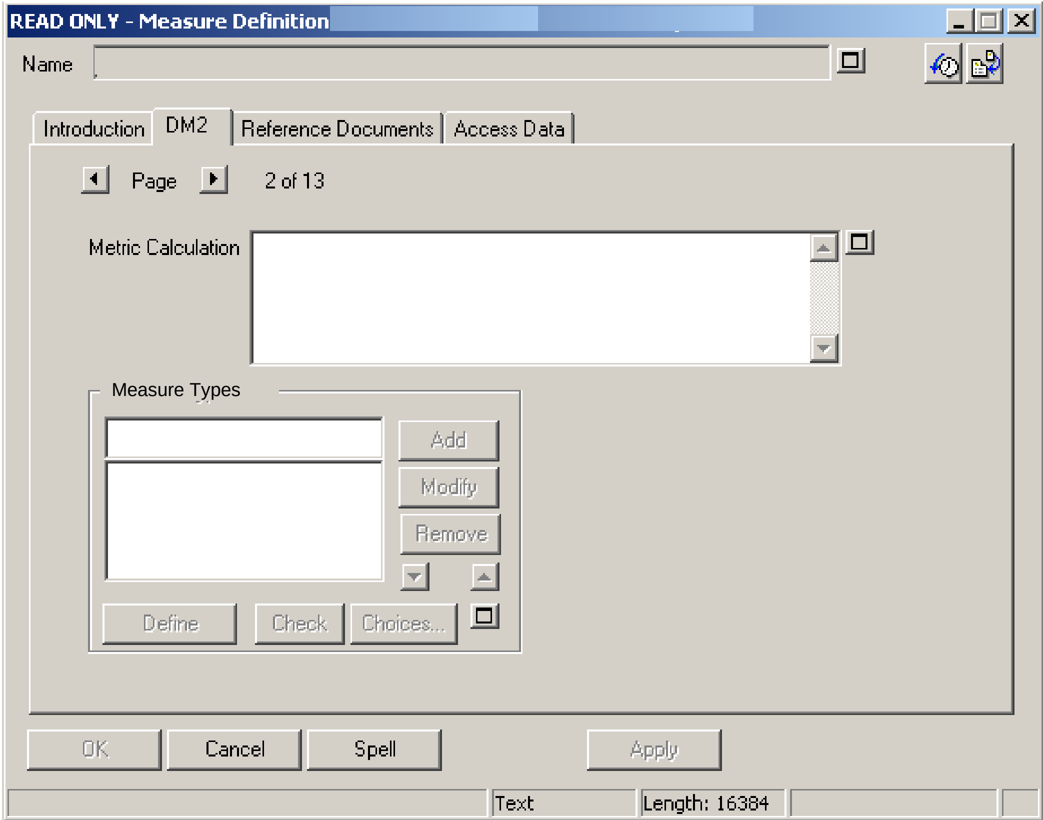


Figure 23-3, Measure DM2 Tab Second Input Window

The following two data fields must be entered:

1. Metric Calculation: Enter an alphanumeric field that defines the calculation that is to be performed
2. Measure Type: One Measure Type must be selected from a drop down list of Measure Types.

The following figure shows the format of the third to thirteenth windows of the Measure DM2 tab that allows links to other BEA objects to be established.

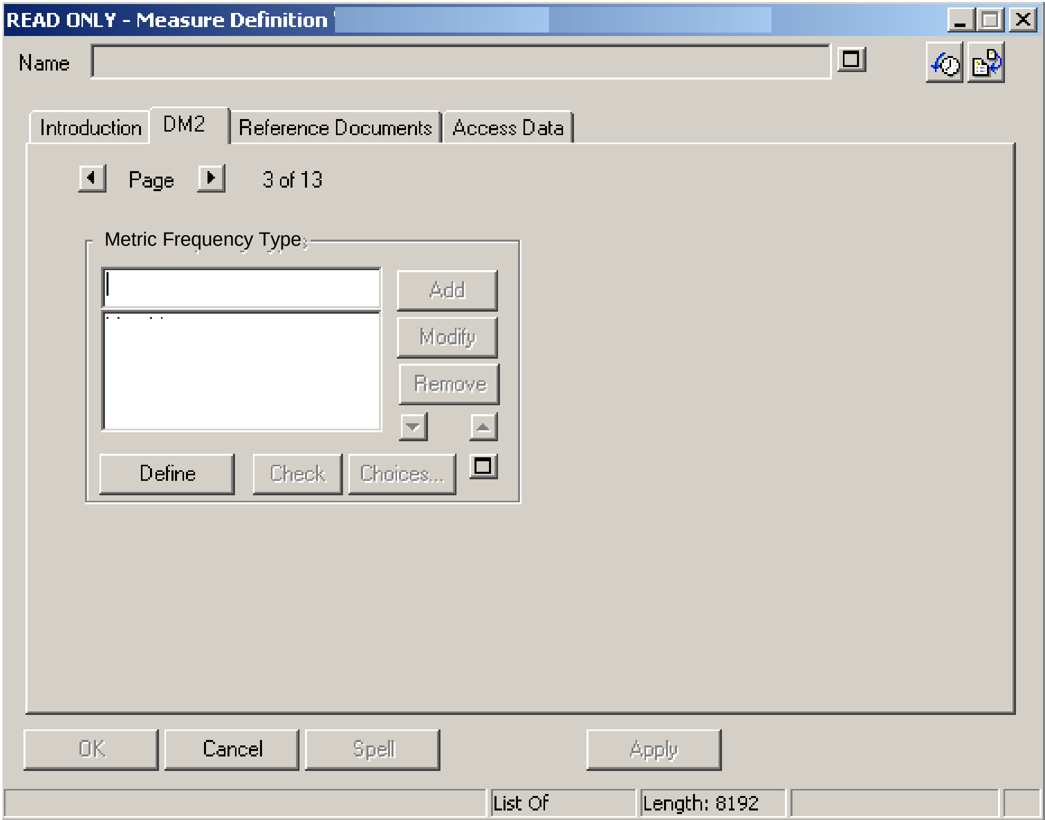


Figure 23-4, Measure DM2 Tab 3 to13 Window Format

Each input window consists of a single drop down list of the following appropriate BEA object:

- Metric Frequency Type
- Metric Types
- LRP Sources
- Enterprise Standards
- Need Lines
- ICOM Arrows
- Information Exchanges
- Message Flows
- System Nodes
- System Interfaces
- System Data Exchanges

23.3.2 Modeling Measures Objects

The Measures Model consists of descriptions of Measures and links to other appropriate BEA objects. Each Measure description must be written in complete and understandable English sentences. Each Measure must be linked to at least on other BEA object.

23.3.3 Measures Best Practices

23.3.3.1 Measures Lessons Learned

Because this is the first instance of Measures usage in the BEA, there is insufficient data available to provide pertinent lessons-learned guidance.

23.3.3.2 Measures Common Pitfalls

Because this is the first instance of Measures usage in the BEA, there is insufficient data available to provide pertinent common-pitfalls guidance.

24 Condition Functional Description

24.1 Summary Description

This section describes the Condition Functional Description, its relationship to other BEA models, the development method, and the modeling guidelines to be followed.

24.1.1 Model Purpose

The primary purpose of the Condition Model is to record each Condition and link the Condition to BEA objects. This allows the planning for resolution of a Condition to consider the impact across the Stakeholders of the BEA.

24.1.2 Model Structure

A Condition is captured in the BEA as a AV-2 Definition and then is linked or associated with other appropriate BEA objects. The linkage is provided through specific tabs on the Condition Definition. A Condition must be linked to at least one other BEA architectural object.

The following is a list of the specific tabs for the Condition AV-2 Definition.

- Parent Condition
- Measures
- Business Rules
- LRP Sources
- Operational Activities
- BPM Processes
- System Functions

Each tab has a drop list of valid BEA objects from which one or more selections can be made.

24.1.3 Relationship to Other BEA Products

A Condition can be related to the following other BEA products:

OV-6a Linked to appropriate Business Rules.

LRP Linked to appropriate LRP.

OV-5 Linked to appropriate Operational Activities.

OV-6c Linked to appropriate BPM Processes.

24.1.4 Condition Model Definitions

The following are the new BEA objects introduced by the incorporation of Conditions.

- ***Parent Condition***. A hierarchy of Conditions can be developed though a parent-child hierarchical relationship. A Condition must belong to only one hierarchy.

24.2 Developing the Condition Model

24.2.1 Pre-Development Tasks

A Condition is defined by the Stakeholder.

24.2.2 Development Tasks

The BEA Architect enters a Condition into SA.

24.2.3 Post-Development Tasks

The BEA Architect verifies that each Condition is linked to at least one appropriate BEA object.

24.3 Modeling Conditions Using SA

24.3.1 Modeling Conventions

The Material Weakness Condition is used as an example of capturing a Condition in the BEA. The example creates a hierarchy of Conditions.

The Material Weakness is a type of Condition and consists of textual definitions and linkages to other BEA objects. The Condition input window is invoked from the Definitions list and the following is displayed.

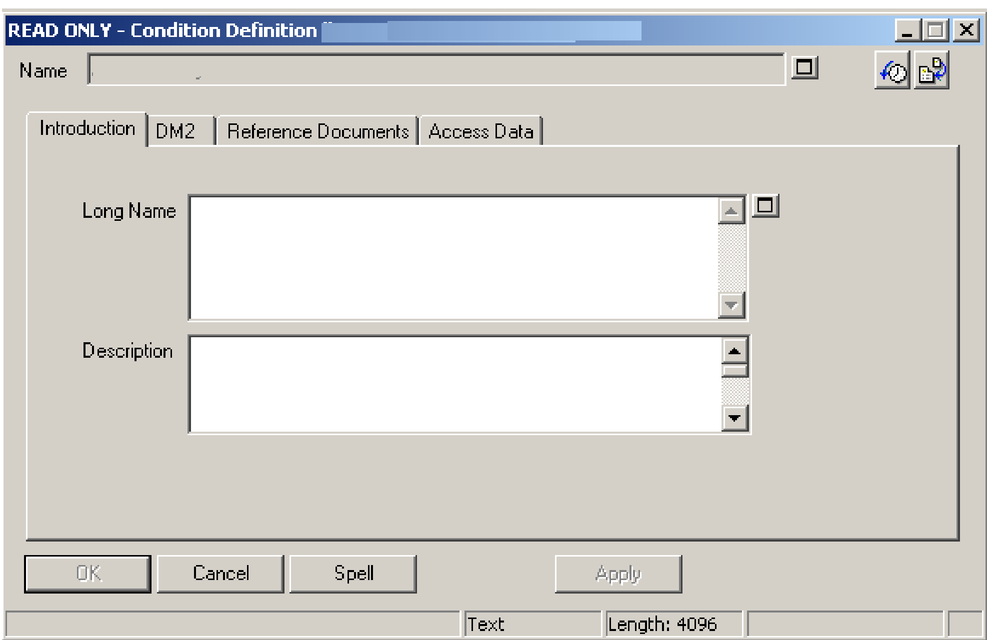


Figure 24-1, Condition Definition Input Window

The following three input data fields must be entered:

1. Name: Enter an alphanumeric short name that uniquely identifies the Condition. This name is used to identify the Condition and to link the Condition to other BEA objects.
2. Long Name: Enter an alphanumeric meaningful business name of the Condition. This name is used by the stakeholders.
3. Description: Enter an alphanumeric field that outlines the purpose of the Condition and the impact of the Condition.

The DM2 tab on the Condition allows a Condition to be linked to appropriate DoDAF DM2 artifacts and conforms to DoDAF DM2 Metamodel. The input window allows the display of seven input windows, one at a time, in the following sequence:

1. Parent Condition
2. Measures
3. Business Rules
4. LRP Sources
5. Operational Activities
6. BPM Processes
7. System Functions

Windows 1 has multiple entry data fields and Windows 2 to 7 have only a single entry data field. There is only one alphanumeric input data field. All other data fields are populated by selections from drop down lists.

The following figure shows the first Window of the DM2 tab which allows a hierarchy of Conditions to be developed.

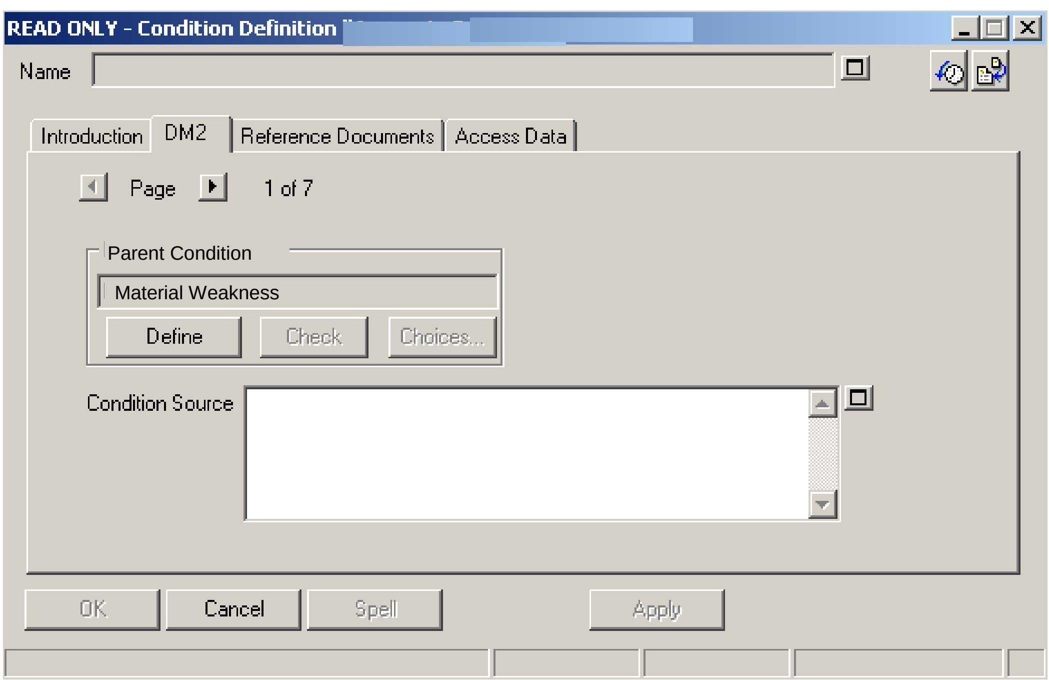


Figure 24-2, Condition DM2 Tab First Input Window

The following two data fields must be entered:

1. Material Weakness must be selected as the Parent Condition to denote a Material Weakness is being recorded.

2. The Condition Source must describe the Material Weakness condition and the source of the condition

The following figure shows the format of the second to seventh windows of the Condition DM2 tab that allows links to other BEA objects to be established.

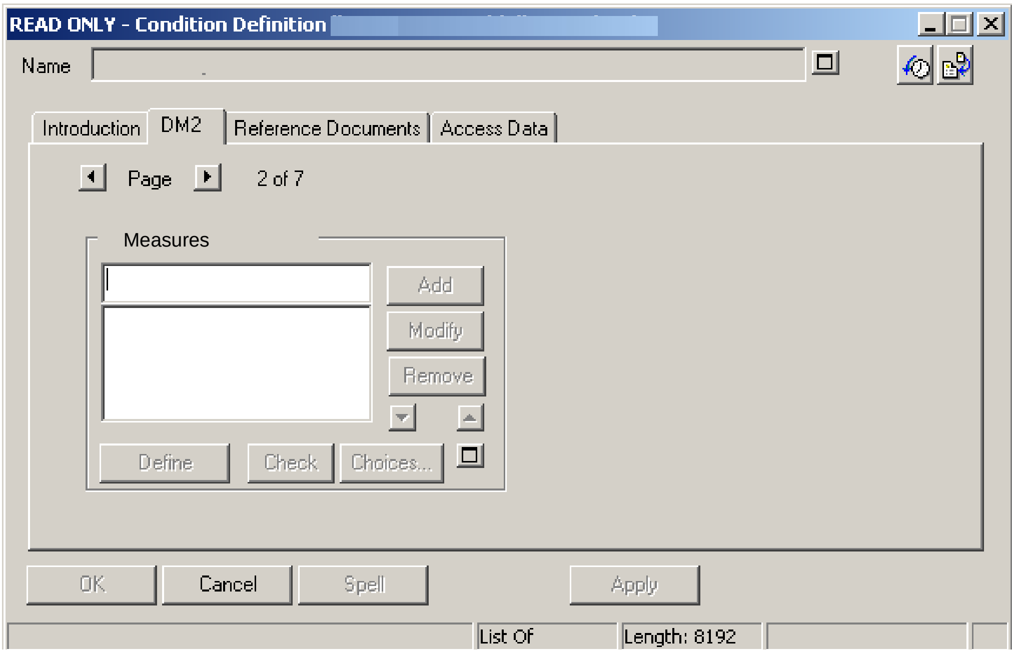


Figure 24-3, Condition DM2 Tab 2-7 Input Window

Each input window consists of a single drop down list of the following appropriate BEA object:

- Measures
- Business Rules
- LRP Sources
- Operational Activities
- BPM Processes
- System Functions

24.3.2 Condition Best Practices

24.3.2.1 Condition Lessons Learned

None have been identified at this time.

24.3.2.2 Material Weakness Common Pitfalls

None have been identified at this time.

Appendix A : References

1. *Introduction to BPMN*, Stephen A. White, IBM Corporation (date of publication unknown).

2. *Business Process Modeling Notation (BPMN)*, *Version 2.0 January 2008*, Stephen A. White, Business Process Management Initiative, *May 3, 2004*.

3. *Semantics of Business Vocabulary and Business Rules version 1.0, January 2008*, Object Management Group

4. *BRS RuleSpeak™, Version 1.0,* Ronald G. Ross and Gladys S.W. Lam

5. *Principles of the Business Rule Approach*, Ronald G. Ross

6. *Business Rule Concepts: Getting to the Point of Knowledge,* Ronald G. Ross

7. *BEA Development Methodology, March 11, 2011*

8. *DoD Architecture Framework (DoDAF), Version 2.0 May 18, 2009*

9. *Integrated Definition for Function Modeling* (IDEF0)

10. *Integrated Definition for Data Modeling* (IDEF1X)

11. *System Rule Concepts: Getting to the Point of Knowledge*, 2nd ed. System Rule Solutions, LLC, 2005

12. *Enterprise Architecture based in Design Primitives and Patterns - Guidelines for the Design of Business Process Models (DoDAF OV-6c) using BPMN, February 11, 2009*

Appendix B : Model Checklists

B-1: AV-1 Model Checklist

CR Number: ______________ **Date:** ______________

Approval Signatures:

Contractor Lead ________________________ **Government Lead** __________________________

			Reviewer Name:		
#	**Source APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	3.1.3	Manual Check	BEA Names and Descriptions should be same in Enterprise Transition Plan (ETP). BEA AV-1 is authoritative source for BEA Names, Description Goals and Objective.		
2	3.1.2	Manual Check	Lead and support Core Business Missions (CBM) should be correctly identified and in accordance with the ETP.		
3	3.1.2	Manual Check	Listing of "Models Developed" should be accurate and complete.		
4		Manual Check	All spelling is correct and there are no grammatical errors.		

B-2: AV-2 Checklists

AV-2 Acronym Definitions Checklist

CR Number: ______________ **Date:** ______________

Approval Signatures:

Contractor Lead ________________________ **Government Lead** __________________________

			Reviewer Name:		
#	Source APG	BRS Report	Checklist Item Description	Modeler	Reviewer
1	4.1.3	Manual Check	The acronym is described in Acronym List		
2	4.2.1.1	Manual Check	The acronym is not described in BEA model description with the exception of Business Guidance items.		
3	4.2.1.1	Manual Check	Short name is described in Acronym List.		
4		Manual Check	Plurals of acronyms are not described in Acronym List.		
5	3.1.3	Manual Check	If the acronym is used in the AV-1 Overview, then the acronym must be described at first use.		
6	3.1.3	Manual Check	If the acronym is used in the AV-1 Overview, then the acronym must be in the SA Acronym List.		
7	4.2.1.1	Manual Check	The acronym must be used within the ETP, BEA, or architecture guidance document.		

AV-2 Term Definitions Checklist

CR Number: ______________ **Date:** ______________

Approval Signatures:

Contractor Lead ________________________ **Government Lead** __________________________

-

			Reviewer Name:		
#	Source APG	BRS Report	Checklist Item Description	Modeler	Reviewer
1		Manual Check	The object names and descriptions published in the AV-2 do not contain any spelling or grammar errors.		
2	4.2.3.1	Manual Check	All AV-2 exception reports have been reviewed and resolved.		

B-3: OV-2 Model Checklists

OV-2 Diagrams Checklist

CR Number: _____________ **Date:** _____________

Approval Signatures:

Contractor Lead ______________________ **Government Lead** ________________________

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1		Manual Check	Verified intended content changes with Subject Matter Expert		
2	9.2 9.3.1.2	Manual Check	There is at least one OV-2 diagram for each internal Operational Node, which are CBMs.		
3	9.3.1.2	Manual Check	The OV-2 diagram does not have a border.		
4	9.3.1.2	Genrl-001	The OV-2 diagram has a Doc Block for header information that includes the title of the diagram on one line, its creation/modification date, but no graphic comments. The Doc Block is in the extreme upper left corner of the diagram with a black border, no fill color and no truncation indicators.		
5	9.3.1.1	Manual Check	Operational Nodes are depicted as light green ovals with black borders and black lettering (the SA default), with no truncation indicators.		
6	9.3.1.1	Manual Check	Operational Node names are in Arial 14, Normal and Black font.		
7	9.1.2	Manual Check	The Operational Node for the CBM represented by the diagram shall be shown at the center of the diagram. Only Operational Nodes that interface with the center Operational Node shall be shown on each diagram.		
8	9.1.2	Manual Check	Operational Activities are only displayed for the center Operational Node on the diagram.		
9	9.3.2.1	OV02-004	All existing Need Lines are used on at least one OV-2 Diagram.		

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
10	9.3.1.1	Manual Check	Need Lines are solid straight lines, containing 90-degree angles (where appropriate) to achieve readability.		
11	9.3.2.2	Manual Check	Need Lines use the default SA pen width and black font.		
12	9.3.2.2	Manual Check	Individual IEs are not displayed under the Need Lines on the OV-2 Diagram.		
13	9.3.2.2	Manual Check	Need Line arrows should not intersect, if at all possible.		
14	9.3.2.2	Manual Check	Need Line can exist on only two OV-2 diagrams unless is linked to an external Operational Node or if a sub-node exists.		
15	9.3.1.2	Genrl-002	The OV-2 Diagrams have a narrative description of the diagram, using the diagram properties comment box to explain the Operational Nodes and their relationships.		
16	9.3.2.1	OV02-001	All Operational Nodes must be referenced by at least one Need Line		
17	9.4.1	Manual Check	All exception reports have been reviewed and resolved.		

OV-2 Definitions Checklist

CR Number: ______________ **Date:** ______________

Approval Signatures:

Contractor Lead ________________________ **Government Lead** __________________________

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	9.3.1.3	OV02-007	Only the following valid Operational Node names are used: • Acronyms for the CBMs • Enterprise Node • External Node		
2	9.3.1.1	AV02-002 AV02-006 AV02-014	Operational Nodes shall be defined in accordance with the related CBM. Name checks shall include Operational Nodes with missing descriptions, short descriptions and truncated descriptions.		
3	9.3.2.1	OV02-007	Each Operational Node is identified by Type ("Abstract" or "Physical").		
4	9.3.1.3	OV02-007	Need Line names are in the following format: Sending CBM – Receiving CBM (for example "HRM – FM").		
5	9.4.1	Manual Check	All exception reports have been reviewed and resolved.		

OV-2 Integration Checklist

CR Number: ______________ **Date:** ______________

Approval Signatures:

Contractor Lead ________________________ **Government Lead** __________________________

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	9.3.2.1	OV02-002	Each Operational Node is mapped to one or more leaf-level Operational Activities.		
2	9.3.2.1	OV05-019	Each leaf-level Activity is assigned to at least one Operational Node.		
3	9.3.2.2	OV02-003	A Need Line includes one or more IEs.		
4	9.3.2.2	Manual Check	A single Need Line is used to represent the interactions of all IEs that have a common source and destination between a pair of Operational Nodes.		
5	9.3.2.1	OV02-005 OV02-006	Each Operational Node has only one CBM Stakeholder assigned, and one or more CBM Stakeholders assigned.		
6	9.4.1	Manual Check	All exception reports have been reviewed and resolved.		

B-4: OV-3 Model Checklists

OV-3 Matrix Checklist

CR Number: _____________ **Date:** _____________

Approval Signatures:

Contractor Lead ______________________ **Government Lead** ________________________

			Reviewer Name:		
#	**Source APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	10.3.1	Manual Check	All fields in each column are filled in.		
2	10.3.1	Manual Check	The “IE Description” column contains the IE definition.		
3	10.3.1	OV03-003	IE names must be in title case, use only approved acronyms and can use only the special character “-”.		
4	10.4.1	Manual Check	All exception reports have been reviewed and resolved.		

OV-3 Integration Checklist

CR Number: _____________ **Date:** _____________

Approval Signatures:

Contractor Lead ______________________ **Government Lead** ________________________

			Reviewer Name:		
#	**Source APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	10.1.2	OV03-001	Each IE is mapped to one or more Entities or Attributes in the DIV Model.		
2	10.2.1	OV03-002	Each IE is mapped to an ICOM Arrow.		

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
3	10.4.1	Manual Check	All exception reports have been reviewed and resolved.		

B-5: OV-5 Model Checklists

OV-5 Diagrams Checklist

CR Number: _____________ **Date:** _____________

Approval Signatures:

Contractor Lead ______________________ **Government Lead** ________________________

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
			OV-5a Operational Activity Decomposition Tree		
1	5.1.2.1 5.3.1.2	Manual Check OV05-017	The OV-5a Operational Activity Decomposition Tree Activity names match the OV-5b Operational Activity Model.		
2	5.3.1.2	Manual Check	The Activity boxes are numbered sequentially, relative to position and match corresponding activity numbering on the Activity Diagram.		
3	5.3.1.2	Manual Check	The top-level box of the Decomposition Tree is centered on the diagram as permitted by the tool.		
4	5.1.4.1 5.3.1.2	OV05-034	Parent Activities are decomposed to at least two, but not more than nine, child Activities.		
5	5.3.1.1	Manual Check	The OV-5a Operational Activity Decomposition Tree follows modeling conventions: Names are Normal and Black font. The Operational Activity box border shall be a solid black line. The integrated Operational Activities shall be white. The standalone Operational Activities that appear on the Decomposition Tree only shall be blue.		
6	5.2.3.1	Manual Check	Activities on the Decomposition Tree should be decomposed to a level low enough to support the Business Capability.		

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
7	5.3.1.3	Manual Check	Each Operational Activity is named as a Verb-Noun, using an active verb phrase (for example, Allocate Resource).		
8	5.1.2.1	Manual Check	Each leaf level Operational Activity should be aligned to the FEA BRM where possible.		
			OV-5b Operational Activity Models		
9	5.2.3.2	Manual Check	The A-0 Activity Diagram (Context Diagram) has a purpose and a viewpoint in the lower left corner of the diagram.		
			Operational Activities		
10	5.3.1.1 5.3.1.2 5.3.2	Manual Check	Activity names follow modeling conventions: The font must be normal and Black. The Activity name begins with a RETURN character. Activity name falls within the Activity box border with no truncation indicators The Activity Box has a solid black line border. All activities should be colored Yellow.		
11	5.3.1.2 5.3.2	Manual Check	Activity boxes are numbered sequentially in the lower right corner. The number inside the Activity box shall match the last digit of the Activity number sitting outside the box beginning with an "A." Activity boxes are stair-stepped vertically and numbered in descending order appropriately.		
12	5.2.3.2	OV05-014	Each Operational Activity has at least one (1) Input or Control and one (1) Output.		
13	5.3.2	OV05-033	Each Activity shall have at least one Mechanism and no more than 12 Mechanisms.		

			Reviewer Name:		
#	**Source APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
14	5.3.1.2	OV05-22	Diagram name (Operational Activity box label) shall use title case (first letter of each word capitalized, other letters lowercase), should be non-plural (exception approved by BEA development team and can use only the special character "-". Any acronyms used in the Operational Activity name must be from the approved acronym list that is part of the BEA AV-2.		
15	5.1.3	Manual Check	Each OV-5 term must be defined in the AV-2.		
16	5.1.3	Manual Check	Each acronym used in OV-5 descriptions must be defined in the AV-2.		
17	5.1.3	Manual Check	Leaf level Operational Activities must be assigned to Operational Nodes in the OV-2.		
18	5.1.3	Manual Check	Leaf level Operational Activities must be assigned to the Capabilities they support		
19	5.1.3	Manual Check	Operational Nodes are represented as mechanisms on the OV-5 Activity Model Diagrams.		
20	5.1.3	Manual Check	Process Steps in the OV-6c Event Trace Description are derived from and link to Operational Activities in the OV-5.		
21	5.1.3	Manual Check	Operational Activities from the OV-5 are mapped to System Functions in the SV-5.		
22	5.1.3	Manual Check	The BEA SV-5 that shows Operational Activities mapped from Business Capabilities and to System Functions with the identified Enterprise System, if available, in the intersection.		

			Reviewer Name:		
#	Source APG	BRS Report	Checklist Item Description	Modeler	Reviewer
			ICOMs		
22	5.2.3.2	Manual Check	ICOMs follow modeling conventions: Naming conventions for External Controls are "{Activity Name} Law Regulation And Policy" Initiatives are Internal Controls and are represented as Controls Mechanism are CBMs and/or Systems / Services Systems and/ or Services are only depicted as Mechanism if they perform the activity - they transform the input ICOMs into the output ICOMs Controls and Mechanisms are stair-stepped in a descending manner from left to right in relation to the positioning of the activity.		
23	5.4.2	OV05-004 OV05-018	All ICOMs are physically connected to a given Activity.		
24	5.4.2	Manual Check	ICOM Arrows have minimal crossings in respect to other ICOM Arrows on any given diagram.		
25	5.3.2	AV02-019 OV05-003 OV05-005	Input ICOMs cannot be represented as Outputs for the same activity. Output ICOMs cannot be represented as Inputs for the same activity.		
26		Manual Check	ICOMs are evenly spaced relative to the edge of an Activity box.		
27	5.3.2	Manual Check	Boundary Input ICOMs come into the diagram even with the first Activity it is attached to, Input ICOM names are left justified, Boundary Output ICOMs exit the diagram at even with the highest Activity it exits from, and names are right justified.		
28	5.3.2	Manual Check	Controls and Mechanisms are vertically aligned as per guidelines.		

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
29	5.3.2	Manual Check, OV05-035	ICOMs are balanced: Input/Output ICOMs on a parent activity are consistent with Inputs/Outputs on its child diagram and vice versa		
30	5.1.4.2	Manual Check	An Output produced by a leaf level Operational Activity is unique to that Operational Activity and is not produced by another Operational Activity which is constrained by the same controls and performed by the same mechanisms.		
31	5.1.4.2	Manual Check	Each leaf-level Input and Output ICOM Arrow on the OV-5 diagram connecting Operational Activities in Operation Nodes is represented as one or more IEs in the OV-3.		
32	5.1.2.1.2	Manual Check	Leaf level Inputs and Outputs must be translated to IEs that depict the information flow represented in the OV-2 as Need Lines between Operational Nodes.		
33	5.1.3	Manual Check	Data Objects in the OV-6c are related one to one to OV-3 IEs, which are mapped to the Inputs and Outputs of Operational Activities in the OV-5 Activity Model.		
34	5.3.2.2	Manual Check	Entities or Attributes within Entities in the DIV Model are either derived from or directly linked to Inputs and Outputs on the OV-5 via the IEs in the OV-3.		
35	5.1.4.2	Manual Check	Enterprise Systems and services that perform system functions are the Mechanisms on the OV-5 if Systems and/ or Services perform the activity - they transform the input ICOMs into the output ICOMs		
			General		
36	5.3.1.1	Manual Check	The OV-5a Operational Activity Decomposition Tree diagram and OV-5b Operational Activity Model diagrams do not have a border.		

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
37	5.3.1.1 5.3.1.2	Genrl-001	The OV-5a Operational Activity Decomposition Tree diagram and OV-5b Operational Activity Model diagrams have a Doc Block for header information that includes the title of the diagram on one line, its creation/modification date, but no graphic comments. The Doc Block is in the extreme upper left corner of the diagram with a black border, no fill color and no truncation indicators.		
38	5.3.1.2	Genrl-002	The OV-5a Operational Activity Decomposition Tree diagram and OV-5b Operational Activity Model diagrams have a narrative description of the diagram, using the diagram properties comment box to explain the Activities and their relationships.		
39	5.4.1	Manual Check	All exception reports have been reviewed and resolved.		

OV-5 Definitions Checklist

CR Number: _____________ **Date:** _____________

Approval Signatures:

Contractor Lead ______________________ **Government Lead** ________________________

			Reviewer Name:		
#	**Source APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	5.3.1.3	AV02-018	Object names are unique and in the proper tense.		
2	5.3.1.3	AV02-010 AV02-011 AV02-012 AV02-013 AV02-009	Object names are in title case, do not end in "s", and contain no special characters or invalid acronyms.		
3	5.3.2	AV02-005 AV02-006 AV02-002 AV02-014 Spell Check	Each Operational Activity and ICOM has a unique grammatically correct definition that has been spell checked.		

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
4	5.3.1.2	AV02-001 AV02-002 AV02-004 AV02-014 AV02-020	The Activity definition identifies what the Activity does, suitable to the level of decomposition and how information is transformed, created or consumed in the Activity.		
5	5.3.2	AV02-001 AV02-002 AV02-004 AV02-014 AV02-020	Each ICOM definition is consistent with the definition of the Activity that produces or consumes it and is consistently decomposed with the Activity. The definition of the Parent ICOM includes the list of child ICOMs.		
6	5.3.2	GENRL-003 GENRL-004	Each Activity and ICOM has one or more CBM Stakeholders assigned.		
7	5.4.1	Manual Check	All exception reports have been reviewed and resolved.		

OV-5 Integration Checklist

CR Number: ______________ **Date:** ______________

Approval Signatures:

Contractor Lead ________________________ **Government Lead** __________________________

			Reviewer Name:		
#	**Source APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	5.1.3.2 5.2.3.3	OV06c-005	Each Leaf-Level Activity is mapped to one or more Leaf-Level Processes.		
2	5.2.3.1 5.2.3.3	OV05-015	Each leaf-level Activity is mapped to the FEA BRM Sub-functions.		
3	5.2.3.2 5.3.2	OV03-002 OV05-020 OV05-024	Each leaf-level Activity Input and Output ICOM has a corresponding IE, with the same name, definition and linked to the same CBMs unless there are multiple IEs. Controls and Mechanisms are not mapped to IEs.		
4	5.1.3.2 5.3.1.2 5.3.1.3	AV02-009 AV02-010 AV02-011 AV02-012 AV02-013 OV05-008	Object names use only approved abbreviations and acronyms contained in the AV-2 and are free of symbol characters, (for example, /, $, @, &).		
5	5.4.1	Manual Check	All exception reports have been reviewed and resolved.		

B-6: OV-6a Model Checklists

OV-6a Definitions Checklist

CR Number: _____________ **Date:** _____________

Approval Signatures:

Contractor Lead ______________________ **Government Lead** ________________________

			Reviewer Name:		
#	**Source APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	6.3.2	Manual Check	Rule can be readily understood by any business or DoD party and is always interpreted the same.		
2	6.4.2	Manual Check	Rule is unambiguous.		
3	6.3.2	Manual Check	Rule is in declarative form – no reference to how, where, when, or who.		
4	6.1	Manual Check	No indication of how rule to be enforced (how).		
5	6.4.2	Manual Check	No indication of where to enforce the rule (where).		
6	6.4.2	Manual Check	No indication of when to enforce the rule (when).		
7	6.4.2	Manual Check	No indication of Events or Event sequence (when).		
8	6.4.2	Manual Check	No indication of who will enforce the rule (who).		
9	6.4.2	Manual Check	Rule is not procedural (use of “else” and “if”).		
10	6.4.2	Manual Check	Rule constrains (or alternatively permits).		
11	6.1.5	OV06a-009	Rule contains one of the key rule words such as “is,” “may,” “must,” “no,” “not,” “shall,” “should,” “will,” or “only if.”		
12	6.1.5	Manual Check	Words such as “can” are not used.		
13	6.1	Manual Check	Rule uses standard terminology such as the common language from the data model.		

			Reviewer Name:		
#	Source APG	BRS Report	Checklist Item Description	Modeler	Reviewer
14	6.4.2	Manual Check	Facts are explicitly expressed in the rule (no hidden facts or computations).		
15	6.1	Manual Check	Rule is written in *RuleSpeak*™ formal language.		
16	6.4.1	Manual Check	All exception reports have been reviewed and resolved.		
17	6.4.2	Manual Check	The rule does not have a plural subject.		
18	6..2	Manual Check	A time element is not the subject.		
19	6.4.2	Manual Check	Rule has explicit subject.		
20	6.4.2	Manual Check	Computations are the subject of the rule.		

OV-6a Integration Checklist

CR Number: _____________ **Date:** _____________

Approval Signatures:

Contractor Lead ______________________ **Government Lead** ________________________

			Reviewer Name:		
#	Source APG	BRS Report	Checklist Item Description	Modeler	Reviewer
1	6.2.2.3	Manual Check	All required fields filled in properly from load.		
2	6.1.4, 6.3.1.1	OV06a-006	Each Operative Derivation Business Rule is linked to a BPM Process.		
3	6.1.4, 6.3.1.1	OV06a-005	Each Operative Action Business Rule is linked to a leaf level Process Flow or leaf level BPM Process.		
4	6.1.4, 6.3.1.1	OV06a-007	Each Structural Relational Business Rule is linked to a DIV Model Relation or BPM Process.		
5	6.3.1.1	OV06a-	Each Structural Definitional Business Rule is linked to		

			Reviewer Name:		
#	Source APG	BRS Report	Checklist Item Description	Modeler	Reviewer
		013	a DIV Model Data Element or an AV-2 Term.		
6		OV06a-012	Each Business Rule must either have an Operative or Structural Category.		
7	6.3.1.1	OV06a-011	Each Business Rule that is an Operative Category must be either a Derivation Type or an Action Type.		
8	6.3.1.1	OV06a-014	Each Business Rule that is a Structural Category must be either a Relational Type or a Definitional Action Type.		
9	6.3.1.2	OV06a-015	All Business Rules must have unique rule number associated with it.		
10	6.4.1	Manual Check	All exception reports have been reviewed and resolved.		

B-7: OV-6c Model Checklists

OV-6c Diagrams Checklist

CR Number: _____________ **Date:** _____________

Approval Signatures:

Contractor Lead ______________________ **Government Lead** ________________________

			Reviewer Name:		
#	**Source APG**	**BART Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1		Manual Check	Verified intended content changes with CBM Subject Matter Expert		
2	8.3.1	OV06c-014	All Diagrams have at least 2 Events and 2 Processes.		
3	8.2.3	Genrl-001	All OV-6c Diagrams have a Doc Block for header information that includes the title of the diagram on one line, its creation/modification date, but no graphic comments. The Doc Block is in the extreme upper left corner of the diagram with a black border, no fill color and no truncation indicators.		
4	8.3.1.2	Manual Check	Diagrams are named in accordance with APG OV-6c guidelines.		
5	8.3.1.2	Manual Check	The Pool structure is in accordance with the APG OV-6c guidelines.		
6	8.3.1.3	AV02-011	OV-6c objects must comply with naming conventions and not use special characters except for Question Marks at the end of Gateway names.		
7	8.2.3	OV06c-004	Data Objects are either associated with a Process Step (as an Input and/or Output), Input or Output Set, or linked to a Sequence Flow or Message Flow and are in accordance with APG OV-6c guidelines.		
8	8.2.3	OV06c-006 OV06c-007	Message Flows are used between Pools and Sequence Flows are used within Pools and are in accordance with APG OV-6c guidelines.		

			Reviewer Name:		
#	**Source APG**	**BART Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
9	8.3.1.2	General-002	The OV-6c Diagram has a narrative description of the diagram, using the diagram properties comment box to explain the Activities and their relationships.		
10	8.4.1	Manual Check	All exception reports have been reviewed and resolved.		
11	8.1.3	Manual Check	All OV-6c terms with a specific meaning must be included in the AV-2 Term Definitions report.		
12	8.1.3	Manual Check	All acronyms used in the OV-6c descriptions must be listed and spelled in the AV-2 Acronyms Definitions List.		
13	8.1.3	Manual Check	Check for impacts to E2E Business Flows.		

OV-6c Definitions Checklist

CR Number: ____________ **Date:** ____________

Approval Signatures:

Contractor Lead ______________________ **Government Lead** ________________________

			Reviewer Name:		
#	**Source APG**	**BART Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	8.3.1.3	Genrl-008 AV02-009 AV02-010 AV02-011 AV02-012 AV02-013 AV02-020	All Pools, Swimlanes, Process Steps, Data Objects, Gateways, Events and groups are named and defined and are in accordance with APG OV-6c guidelines. Names and definitions are optional for Sequence Flows and Message Flows		
2	8.3.1	AV02-013	Only approved acronyms have been used.		
3	8.2.3	OV06c-001	Events do not reference themselves.		
4	8.4.1	Manual Check	All exception reports have been reviewed and resolved.		

OV-6c Integration Checklist

CR Number: ______________ **Date:** ______________

Approval Signatures:

Contractor Lead ________________________ **Government Lead** __________________________

			Reviewer Name:		
#	**Source APG**	**BART Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	8.1.4	OV06c-005	Each leaf-level Process has been mapped to one or more leaf-level Activities in the OV-5.		
2	8.1.4	OV6a-005 OV6a-006	OV-6a non-SFIS Business Rules have been mapped to OV-6c Process Steps, Gateways, or Conditional Sequence Flows		
3	8.1.4	OV06c-015	All Data Objects must be mapped to one and only one IE related to an ICOM on the OV-5b.		
4	8.2.3	Genrl-003 Genrl-004	Each Process, Data Object, Event, or Gateway has one or more CBM Stakeholders assigned.		
5	8.2.5	AV02-004	All OV-6c objects must appear on a diagram. **Exception:** Data Objects associated to Process Steps via an Input Set or an Output Set.		
6	8.4.1	Manual Check	All exception reports have been reviewed and resolved.		
7	8.1.3	Manual Check	OV-6c Pools represent OV-2 Nodes and OV-6c Lanes represent OV-2 Subnodes		
8	8.1.3	Manual Check	IEs in the OV-3 are represented as Data Objects. Each Data Object may be linked to one and only one IE.		
9	8.1.3	Manual Check	Process Steps in the OV-6c are derived from and linked to leaf-level Operational Activities in the OV-5 Activity Model. Each of these process level activities must have at least one Process Step mapped to them.		

			Reviewer Name:		
#	**Source APG**	**BART Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
10	8.1.3	Manual Check	Each Process Step may be linked to one or more OV-6a Business Rules. Action Assertion Business Rules from the OV-6a help to define and are linked to Process Steps and Gateways in the OV-6c.		
11	15.1.4	Manual Check	Each Process Step may be linked to one or more DFMIA/FFMIA Guidance statements. A Process Step change may affect the linkage.		

B-8: DIV-1 Checklists

DIV-1 Diagrams Checklist

CR#: __________ **Date:** __________________

Approval Signatures:

Contractor Team Lead _________________ **Government Product Lead** _____________________

			Reviewer Name:		
#	**Source APG**	**BART Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	7.3.2.1.1 7.3.2.1.2	OV07-006, OV07-018	All Concepts names follow DIV section of the APG.		
2	7.3.2.2 7.3.2.2.1 7.3.2.2.2	OV07-001 OV07-002 OV07-004 OV07-011 OV07-016	All Attribute names follow DIV section of the APG.		
3	7.3.2.3 7.3.2.3.1 7.3.2.3.2	OV07-012 OV07-014	All Data Element names follow DIV section of the APG. (Never use a single Data Element to represent more than one Data Domain.)		
4	7.3.2.1.1	OV07-009	Each DIV-1 Concept must contain at least one Attribute.		
5	7.4.1	Manual Check	Ensure that all exception reports have been reviewed and resolved.		

DIV-1 Definitions Checklist

CR#: ________ Date: __________

Approval Signatures:

Contractor Team Lead _________________ Government Product Lead _____________________

			Reviewer Name:		
#	**Source APG**	**BART Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	7.1.4 7.2.2.3	AV02-002	All objects are defined (Diagrams, Concepts, Attributes and Data Elements).		
2	7.3.2.1.2	AV02-002	All Concepts definitions follow DIV section of the APG.		
3	7.3.2.2.3	OV07-008 AV02-002	All Attribute definitions follow DIV section of the APG.		
4	7.3.2.3.2	AV02-002	All Data Element definitions follow DIV section of the APG.		
6	7.1.4 7.2.2.3	Manual Check	Ensure that words on the “Terms” list are represented correctly and consistently in all object names and descriptions. Ensure that the Terms are not redefined within definitions.		
7	7.2.2.3	AV02-004 OV07-015	Remove DIV objects from the encyclopedia which are not on or referenced from any DIV Model Diagram		
8	7.4.1	Manual Check	Ensure that all exception reports are reviewed and resolved.		

DIV-1 Integration Checklist

CR#: ________ Date: __________

Approval Signatures:

Contractor Team Lead _________________ Government Product Lead _____________________

			Reviewer Name:		
#	**Source APG**	**BART Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	7.2.2.3 7.2.2.4 7.1.4	Manual Check	By-product changes resulting from the CR solution have been identified (Items 2 thru 7 below).		
2	7.2.2.2	OV07-013	All DIV-1 Concepts provided by the CBM are accounted for in the CBM's IEs in their OV-3 or within OV-6a derivation rules.		
3	7.2.2.1	Manual Check	Any Data Elements required by the CBM for system certification are identified.		
4	7.4.1	Manual Check	All exception reports have been reviewed and resolved.		

B-9: DIV-2 Model Checklists

DIV-2 Diagrams Checklist

CR Number: _____________ **Date:** _____________

Approval Signatures:

Contractor Lead ______________________ **Government Lead** ________________________

			Reviewer Name:		
#	**Source APG**	**BART Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	7.2.1 7.2.2.1 7.2.2.3	Manual Check	Verified intended content changes with Subject Matter Expert using the HTML pages and printable Diagrams from the latest BEA Development Build containing the BEA refinements. Ensure that all content is properly displayed, Ensure that all objects appearing on reports and diagrams are linkable Ensure that all objects display all possible relationships to other objects Ensure that the return link is provided on related objects Ensure that all object definitions for the same object type display consistently across BEA		
2	7.3.1.2	Manual Check	All diagrams are a Noun or Noun phrase or the diagram name must exactly match the Name and definition of the IE		
3	7.3.1 7.3.1.2	Manual Check	All diagram titles match the diagram name.		
4	7.3.1.1 7.3.1.2	Manual Check	Each Diagram has a Doc Block describing the diagrams contents in sufficient detail as to aid the viewer in understanding the diagram.		
5	7.1.3 7.3.1.2	Manual Check	Diagram definitions must describe the intended use of the particular view and level of maturity information may be placed in the Notes area.		
6	7.3.2.1.1 7.3.2.1.2	OV07-006, OV07-	All Entity names follow DIV section of the APG.		

			Reviewer Name:		
#	**Source APG**	**BART Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
		018			
7	7.3.2.2 7.3.2.2.1 7.3.2.2.2	OV07-001 OV07-002 OV07-004 OV07-011 OV07-016	All Attribute names follow DIV section of the APG.		
8	7.3.2.3 7.3.2.3.1 7.3.2.3.2	OV07-012 OV07-014	All Data Element names follow DIV section of the APG. (Never use a single Data Element to represent more than one Data Domain.)		
9	7.3.3.1.1 7.3.3.1.2	Manual Check	All Relationships between Entities follow DIV section of the APG		
10	7.3.2.1.4	OV07-018	Each DIV-2 Entity has a Primary Key.		
11	7.3.2.1.4	Manual Check	Each Primary Key uses the natural key when one is available.		
12	7.2 7.3 7.4	Manual Check	All diagramming techniques follow the DIV section of the APG.		
13	7.3.3.2.1	OV07-007	All Subtypes have the same primary key as the super-type (Role-based names are allowed but this practice is discouraged since by definition a subtype must have the same key as the super type. If stakeholders require a different key attributes for the subtype consider modeling as an independent entity along with an associative entity).		
14	7.3.3.2.1	Manual Check	All subtypes have one or more Attributes and/or one or more Relationships to differentiate them form the supertype and the other subtypes.		
15	7.4.2	OV07-009	All DIV-2 Child Entities have one or more Attributes that differentiate them from their parent Entity.		

			Reviewer Name:		
#	Source APG	BART Report	Checklist Item Description	Modeler	Reviewer
16	7.3.2.1.1	OV07-009	Each DIV-2 Entity must contain at least one Attribute.		
17	7.2.2.3	Manual Check	Ensure that CBM Stakeholders agree with their representation on diagrams.		
18	7.2.2.3	Manual Check	Remove invalid and duplicate access paths that cause the display of AK1 designations in the primary key portion of Entities.		
19	7.2.2.3	Manual Check	Ensure that all Relationship lines on all DIV-2 Diagrams display properly and are not hidden.		
20	7.2.2.3	Manual Check	Ensure that the associated tags of all Relationship lines are positioned properly on the diagram.		
21	7.2.2.3	Manual Check	Ensure that, at 21% zoom, all Attribute names are displayed on a single line within the Entity.		
22	7.2.2.3	Manual Check	Ensure that all Relationship lines are straight, not broken, and that all Relationship lines avoid crossing others whenever possible.		
23	7.3.1.2	Manual Check	Ensure that all diagram descriptions, diagram doc blocks, diagram notes, diagram names, object names and object descriptions are spell checked.		
24	7.3.1.1	Manual Check	Ensure that all Entities are colored properly.		
25	7.3	Manual Check	Ensure that all DIV Model printable diagrams display correctly.		
26	7.4.1	Manual Check	Ensure that all exception reports have been reviewed and resolved.		

DIV-2 Definitions Checklist

CR Number: ____________ **Date:** ____________

Approval Signatures:

Contractor Lead ______________________ **Government Lead** ________________________

			Reviewer Name:		
#	**Source APG**	**BART Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	7.1.4 7.2.2.3	AV02-002	All objects are defined (Diagrams, Entities, Attributes and Data Elements).		
2	7.3.2.1.2	AV02-002	All Entity definitions follow DIV section of the APG.		
3	7.3.2.2.3	OV07-008 AV02-002	All Attribute definitions follow DIV section of the APG.		
4	7.3.2.3.2	AV02-002	All Data Element definitions follow DIV section of the APG.		
5	7.3.3.2.2	OV07-010	All DIV-2 IDEF1X Categorizations have a Name and Discriminator		
6	7.1.4 7.2.2.3	Manual Check	Ensure that words on the "Terms" list are represented correctly and consistently in all object names and descriptions. Ensure that the Terms are not redefined within definitions.		
7	7.2.2.3	OV07-021	Ensure that all table names directly supporting IEs used for compliance have the correct physical name set.		
9	7.2.2.3	Manual Check	Ensure that the IDEF1X categorization names on DIV-2 diagrams match the discriminator Attribute names with the removal of their class word and the replacement of the "_" between terms with spaces.		
10	7.2.2.3	Manual Check	Ensure that all column names directly supporting IEs used for compliance have the correct physical name set.		
11	7.2.2.3	AV02-004	Remove DIV objects from the encyclopedia which are not on or referenced from any DIV Model		

			Reviewer Name:		
#	**Source** **APG**	**BART** **Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
		OV07-015	Diagram		
12	7.4.1	Manual Check	Ensure that all exception reports are reviewed and resolved.		

DIV-2 Integration Checklist

CR Number: _____________ **Date:** _____________

Approval Signatures:

Contractor Lead ______________________ **Government Lead** ________________________

			Reviewer Name:		
#	**Source APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	7.2.2.3 7.2.2.4 7.1.4	Manual Check	By-product changes resulting from the CR solution have been identified (Items 2 thru 7 below).		
2	7.2	OV05-008 OV03-001	Each OV-3 IE within the scope of a CR is related to one or more DIV Model Entities or Attributes within Entities.		
3	7.2.2.2	OV07-013	All DIV-2 Entities provided by the Stakeholder are either directly support the content of the IEs or represent the structural assertions that must be applied to the data at rest or support the OV-6a derivation rules.		
6	7.3.2.3	OV07-005	All Data Elements are directly or indirectly linked to one or more Attributes within one or more Entities within the DIV Model.		
7	7.2.2.1	Manual Check	Any Data Elements required by the stakeholder for system certification are identified.		
8	7.4.1	Manual Check	All exception reports have been reviewed and resolved.		

B-10: DIV-3 Model Checklists

DIV-3 Diagram Checklist

CR#: ________ Date: __________

Approval Signatures:

Contractor Team Lead ________________ Government Product Lead ____________________

			Reviewer Name:		
#	**Source APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	TBD	Manual Check	All diagram titles match the diagram name.		
2	TBD	Manual Check	Each Diagram has a Doc Block describing the diagrams contents in sufficient detail as to aid the viewer in understanding the diagram.		
3	TBD	Manual Check	Diagram definitions must describe the intended use of the particular view and level of maturity information may be placed in the Notes area.		
4	TBD	Manual Check	Each DIV-3 Table has at least one Primary Key		
5	TBD	Manual Check	Each Primary Key uses the physical key when one is available.		
6	TBD	Manual Check	All diagramming techniques follow the DIV section of the APG.		
7	TBD	Manual Check	Each DIV-3 Table must contain at least one Column		
8	TBD	Manual Check	Ensure that CBM Stakeholders agree with their representation on diagrams.		
9	TBD	Manual Check	Ensure that all Constraint lines on all DIV-3 Diagrams display properly and are not hidden.		
10	TBD	Manual Check	Ensure that, at 21% zoom, all Column names are displayed on a single line within the Table.		
11	TBD	Manual Check	Ensure that all Constraint lines are straight, not broken, and that all Constraint lines		

			Reviewer Name:		
#	**Source** **APG**	**BRS** **Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
			avoid crossing others whenever possible.		
12	TBD	Manual Check	Ensure that all diagram descriptions, diagram doc blocks, diagram notes, diagram names, object names and object descriptions are spell checked.		
13	TBD	Manual Check	Ensure that all DIV Model printable diagrams display correctly.		
14	TBD	Manual Check	Ensure that all exception reports have been reviewed and resolved.		

DIV-3 Definitions Checklist

CR#: _______ Date: __________

Approval Signatures:

Contractor Team Lead _________________ Government Product Lead _____________________

			Reviewer Name:		
#	**Source APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	TBD	Manual Check	All objects are defined (Diagrams, Tables, Columns and Constraints).		
2	TBD	Manual Check	All Tables have a definition.		
3	TBD	Manual Check	All Column Definitions follow DIV section of the APG.		
4	TBD	Manual Check	All Data Element definitions follow DIV section of the APG.		
5	TBD	Manual Check	Ensure that all table names exactly match their physical name.		
6	TBD	Manual Check	Ensure that all column names exactly match their physical name.		
7	TBD	AV02-004	Remove DIV objects from the encyclopedia which are not on or referenced from any DIV Model Diagram		
8	TBD	Manual Check	Ensure that all exception reports are reviewed and resolved.		

DIV-3 Integration Checklist

CR#: __________ Date: ___ ____

Approval Signatures:

Contractor Team Lead _________________ Government Product Lead _____________________

			Reviewer Name:		
#	**Source** **APG**	**BRS** **Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	TBD	Manual Check	By-product changes resulting from the CR solution have been identified (Items 2 thru 4 below).		
2	TBD	Manual Check	Each DIV-3 Table is related to a DIV-2 Entity.		
3		Manual Check	Each DIV-3 Column is related to a DIV-2 Attribute.		
4		Manual Check	Each DIV-3 Constraint is related to a DIV-2 Relationship		

B-11: SV-1 Model Checklists

SV-1 Diagrams Checklist

CR Number: _____________ **Date:** _____________

Approval Signatures:

Contractor Lead ______________________ **Government Lead** ________________________

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1		Manual Check	Verified intended content changes with Subject Matter Expert		
2	11.3.1.2	Manual Check	There is a SV-1 for each CBM that has identified Enterprise Systems for the BEA and an External SV-1 to reflect DoD use with federally mandated systems.		
3	.11.3.1.2	Manual Check	Each SV-1 diagram has a Diagram Description.		
4	11.3.1.2	Manual Check	Each SV-1 diagram has a Doc Block representing header information for the diagram (including the diagram name and date last updated) that is placed at the top left of every diagram.		
5	11.3.1.2	Manual Check	Doc Block has been enlarged so there are no truncation indicators (dots) indicating text is not visible.		
6	11.3.1.2	Manual Check	The Doc Block has a box with no fill and a black border.		
7	11.3.1.2	Manual Check	The SV-1 diagram name is the name of the focus CBM or External system node, light green, on the diagram.		
8	11.3.1.1	Manual Check	All System Node labels are Arial 14, Bold and Black font.		
9	11.3.1	Manual Check	All System Node labels are top center of the System Node border and the label does not fall outside the boundary of the ellipse.		
10	11.3.1.1	Manual Check	Internal System Nodes are elliptical with a light blue fill and black border.		

			Reviewer Name:		
#	Source APG	BRS Report	Checklist Item Description	Modeler	Reviewer
11	11.3.1.1	Manual Check	The central System Node on each diagram is elliptical with a light green fill and a black border.		
12	11.3.1.1	Manual Check	The external System Nodes is elliptical with a light gray fill and a black border.		
13	11.3.1.1	Manual Check	All System Entity labels are Arial 10, Bold and Black font.		
14	11.3.1	Manual Check	All System Entity labels are at top center of the System Entity box and the label should not fall outside the box boundary.		
15	11.3.1.1	Manual Check	Enterprise-level System Entities are yellow boxes with a black border.		
16	11.3.1.1	Manual Check	Family of System Entities are light orange boxes with a black border.		
17	11.3.1	Manual Check	Each System Node may contain a Family of System Entity.		
18	11.3.1.1	Manual Check	Non-DoD System Entities are white boxes with a black border.		
19	11.3.1	Manual Check	All System Entities are contained within their associated System Node elliptical boundary.		
20	11.2.2.1	SV01-004 SV01-003	Each internal System Entity lists related System Functions.		
21	11.3.1.3	Manual Check	The naming convention for System Interfaces is "sending System Entity abbreviation"- "receiving System Entity abbreviation."		
22	11.3.1.1	Manual Check	System Interface labels are placed, where possible, above the horizontal line where most visible and close to the arrowhead.		
23	11.3.1.	Manual Check	System Interface lines should not traverse intermediate System Entities.		
24	11.3.1.	Manual Check	To the maximum extent possible, System Interface lines should not cross intermediate System Nodes.		
25	11.3.1	Manual Check	System Interface arrows are black with black filled arrowheads.		

			Reviewer Name:		
#	Source APG	BRS Report	Checklist Item Description	Modeler	Reviewer
26	11.3.1	Manual Check	System Interfaces connect to a System Entity at both ends.		
27	11.3.1.1	Manual Check	System Interfaces are solid, straight, lines with 90-degree curves, when necessary.		
28	11.4.1	Manual Check	All exception reports have been reviewed and resolved.		

SV-1 Definitions Checklist

CR Number: _____________ **Date:** _____________

Approval Signatures:

Contractor Lead ______________________ **Government Lead** ________________________

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	11.3.1.2.	Manual Check	Each SV-1 has a diagram description that explains what it represents and a brief description of the Enterprise Systems.		
2	11.3.1.2.	Manual Check	Each System Node has a definition consistent with that of the related Operational Node.		
3	11.1.4	Manual Check	Each System Entity has a description of what the Enterprise System does.		
4	11.1.4	Manual Check	Each SRF has a description of the data it represents.		

SV-1 Integration Checklist

CR Number: _____________ **Date:** _____________

Approval Signatures:

Contractor Lead ______________________ **Government Lead** ________________________

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	11.2.2.1, 11.2.2..3	SV01-002	Operational Nodes (Physical Nodes) are assigned to the System Node from the OV-2, excluding External sub nodes.		
2	11.2.2.3	SV01-004 SV01-003	Each BEA Enterprise System must have at least one System Function assigned.		
3	11.2.1.2	Manual Check	System Entities have both CBM Stakeholders assigned.		
4	11.3.1	Manual Check	System Entities must have a Parent system assigned.		
5	11.2.1.2	Manual Check	System Interfaces between System Entities reflect Need Lines from the OV-2.		
6	11.2.2.1	SV01-001	Each System Interface must link to at least one SRF		
7	11.2.1.2	SV01-005	Each SRF must link to at least one IE from the OV-3.		
8	11.2.1.2	Manual Check	System Interfaces between Family of System Entities reflects Need Lines from the OV-2 and IEs from the OV-3.		

B-12: SV-5 Model Checklists

SV-5 Matrix Checklist

CR Number: ______________ **Date:** ______________

Approval Signatures:

Contractor Lead ________________________ **Government Lead** __________________________

			Reviewer Name:		
#	**Source APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	12.3.1.	Manual Check	There is one SV-5 matrix that represents all CBMs.		
2	12.3.1.	Manual Check	The SV-5 consists of Business Capabilities, Operational Activities, and System Functions and Enterprise-level Systems.		
3	12.3.1.	SV04-009	All systems functions must be referenced by at least one Operational Activity.		
4	12.2.2.3	Manual Check	All exception reports have been reviewed and resolved.		

SV-5 Definitions Checklist

CR Number: ______________ **Date:** ______________

Approval Signatures:

Contractor Lead ________________________ **Government Lead** __________________________

			Reviewer Name:		
#	**Source APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	12.1.4 12.3.1.2	Manual Check	Each Business Capability shall have a definition.		
2	12.1.4	Manual Check	Each System Function should describe the automation of the OV-5 leaf-level Operational Activity it supports.		
3	12.2.2.3	Manual Check	All exception reports have been reviewed and resolved.		

SV-5 Integration Checklist

CR Number: _____________ **Date:** _____________

Approval Signatures:

Contractor Lead ______________________ **Government Lead** ________________________

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	12.3.1.1	Manual Check	The SV-5 matrix only contains relationships between leaf-level Operational Activities, Business Capabilities, Enterprise-level systems and System Functions.		
2	12.3.1.1	Manual Check	Each System Function should map to at least one leaf-level Operational Activity with an Enterprise System/Enterprise Entity Name, FoS or by an "X".		
3	12.2.1.2	Manual Check	Each System Function must have at least one CBM Stakeholder.		
4	12.3.1.2	Manual Check	Each System Function must have at least one CBM Stakeholder.		
5	12.2.1.	Manual Check	The SV-5 must be consistent with the ETP		
6		Manual Check	System Functions shall be used to develop Enterprise Sub-Services		
7	12.2.2.3	Manual Check	All exception reports have been reviewed and resolved.		

B-13: SV-6 Model Checklists

SV-6 Matrix Checklist

CR Number: _____________ **Date:** _____________

Approval Signatures:

Contractor Lead _______________________ **Government Lead** _________________________

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	13.1.2	Manual Check	There is a System Interface abbreviation from the SV-1 diagram in the first column of the SV-6 matrix.		
2	13.1.2	Manual Check	Corresponding SRFs for each System Interface appear in the second column.		
3	13.1.3	Manual Check	System Interface Sending System Entity and System Function from the SV-1 diagram appear.		
4	13.1.3	Manual Check	The Sending System Node from the SV-1 appears.		
5	13.2.3	Manual Check	System Interface Receiving System Entity and System Function from the SV-1 appear.		
6	13.1.2	Manual Check	The Receiving System Node from the SV-1 appears.		
7	13.1.3	Manual Check	The Data Entity for the SRF appears.		
8	13.4.1	Manual Check	All exception reports have been reviewed and resolved.		

SV-6 Integration Checklist

CR Number: ____________ **Date:** ____________

Approval Signatures:

Contractor Lead _____________________ **Government Lead** _______________________

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	13.2.3	Manual Check	The SV-6 only shows System Interfaces between System Nodes.		
2		Manual Check	Each SRF has a corresponding DIV-2 Data Entity		
3	13.4.1	Manual Check	All exception reports have been reviewed and resolved.		

B-14: StdV-1 Model Checklists

StdV-1 Definitions Checklist

CR Number: ______________ **Date:** ______________

Approval Signatures:

Contractor Lead ________________________ **Government Lead** __________________________

			Reviewer Name:		
#	**Source APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	14.1.2	Manual Check	The Standards are mandated in the DoD Information Technology Standards Registry.		
2	14.1.2	Manual Check	Brief descriptions for the Information Technology Standards are provided.		
3	14.1.2	Manual Check	References to the Standard administration/publishing organization are provided.		
4	14.1.2	Manual Check	DISR reference to where the Standards details can be obtained is provided.		
5	14.3.1	Manual Check	All exception reports have been reviewed and resolved		

StdV-1 Integration Checklist

CR Number: ______________ **Date:** ______________

Approval Signatures:

Contractor Lead ________________________ **Government Lead** __________________________

			Reviewer Name:		
#	**Source APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	14.1.2	Manual Check	Each Technical Service shall map to a Technology Service Area, which represents a Core Service Area of the DoD EA TRM.		

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
2	14.1.2	Manual Check	Each Standard shall link to the Technical Service it supports.		
3	14.1.2	Manual Check	Each Technical Service shall link to Enterprise Sub-Services that it supports.		
4	14.3.1	Manual Check	All exception reports have been reviewed and resolved.		

B-15: DFMIR/FFMIA Guidance Model Checklists

DFMIR/FFMIA Guidance Model Checklist

CR Number: ____________ Date: ____________

Approval Signatures:

Contractor Lead ______________________ Government Lead ________________________

#	Source APG	BRS Report	Reviewer Name: Checklist Item Description	Modeler	Reviewer
1		Manual Check	Each FFMIA/DFMIR Guidance Name is correctly spelled.		
2		Manual Check	Each FFMIA/DFMIR s Guidance Description is correctly spelled.		
3	15.1.4 15.3.1	Manual Check	Each FFMIA/DFMIR Guidance must have unique rule number associated with it.		
4	15.1.4.1	Manual Check	Each FFMIA/DFMIR Guidance has a corresponding LRP Source Definition that has the same Name value as the LRP Source value in the FFMIA/DFMIR Guidance.		
5	15.1.4.1	Manual Check	Each corresponding LRP Source Definition has a Description value and is correctly spelled.		
6	15.1.4.1	Manual Check	Each corresponding LRP Source Definition has a Link value that is the same as the Name value of the LRP Source Definition.		
7	15.4.1	Manual Check	All exception reports have been reviewed and resolved.		

DFMIR/FFMIA Guidance Integration Checklist

CR Number: ____________ **Date:** ____________

Approval Signatures:

Contractor Lead ______________________ **Government Lead** ________________________

#	Source APG	BART Report	Reviewer Name: Checklist Item Description	Modeler	Reviewer
1	15.1.4	Manual Check	All required fields filled in properly from load.		
2	15.1.4 15.3.1	Manual Check	Each FFMIA/DFMIR Guidance statement must have unique rule number associated with it.		
3	15.1.4	Manual Check	Each FFMIA/DFMIR Guidance statement must be linked to at least one BPM Process.		
4	15.1.4	Manual Check	Each FFMIA/DFMIR Guidance statement must be linked to at least one LRP.		
5	15.1.4 15.3.1	Manual Check	Each FFMIA/DFMIR Guidance statement must be linked to at least one Stakeholder.		
6	15.4.1	Manual Check	All exception reports have been resolved.		

B-16: End to End Business Flow Checklists

End to End Business Flow Definitions Checklist

CR Number: _____________ **Date:** _____________

Approval Signatures:

Contractor Lead ______________________ **Government Lead** ________________________

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	18.3.1	Manual Check	The word "to" in a Level 0 Activity name must be immediately preceded and immediately followed by a "-" that adjoins two Level 0 Activity words into a single term		
2	18.3.1	Manual Check	Each E2E Activity must have a description.		
3	18.3.1	Manual Check	Each E2E Activity word must begin with a capital letter.		
4	18.3.1	Manual Check	Each E2E Activity acronym must be included in the Acronym Definitions.		
5	18.3.1	Manual Check	Each new term in an E2Edefinition must be included in the Term Definitions.		

End to End Business Flow Integration Checklist

CR Number: ____________ **Date:** ____________

Approval Signatures:

Contractor Lead ______________________ **Government Lead** ________________________

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	18.1.3	Manual Check	BPMN process diagrams are attached to the appropriate E2E Level 1 or 2 activities.		

B-17: SvcV-1 Services Context Description

SvcV-1 Services Context Description

CR Number: _____________ **Date:** _____________

Approval Signatures:

Contractor Lead ______________________ **Government Lead** ________________________

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Review er**
1	17.2.1	Manual Check	Verify intended content changes with Subject Matter Expert		
2	17.3.1.1	Manual Check	Each SvcV-1 diagram has a Diagram Description.		
3	17.3.1.1	Manual Check	Each SvcV-1 diagram has a Doc Block representing header information for the diagram (including the diagram name and date last updated) that is placed at the top left of every diagram.		
4	17.3.1.1	Manual Check	Doc Block has been enlarged so there are no truncation indicators (dots) indicating text is not visible.		
5	17.3.1.2	Manual Check	The Doc Block has a box with no fill and a black border.		
6	17.3.1.1	Manual Check	All Service Container labels are Arial 14, Bold and Black font.		
7	17.3.1.1	Manual Check	All Service Container labels are top center of the Service Container border and the label does not fall outside the boundary of the ellipse.		
8	17.3.1.1	Manual Check	All Service labels are Arial 10, Bold and Black font.		
9	17.3.1.1	Manual Check	All Service labels are at top center of the System Entity box and the label should not fall outside the box boundary.		
10	17.3.1.1	Manual Check	All Services are contained within their associated Service Container elliptical boundary.		

			Reviewer Name:		
#	Source APG	BRS Report	Checklist Item Description	Modeler	Reviewer
11	17.3.1.1	Manual Check	The naming convention for Resource Flows is "sending object abbreviation"- "receiving object abbreviation."		
12	17.3.1.1	Manual Check	Resource Flows labels are placed, where possible, above the horizontal line where most visible and close to the arrowhead.		
13	17.3.1.1	Manual Check	Resource Flow lines should not traverse intermediate System Entities.		
14	17.3.1.1	Manual Check	To the maximum extent possible, Resource Flow lines should not cross intermediate Service objects.		
15	17.3.1.1	Manual Check	Resource Flow arrows are black with black filled arrowheads.		
16	17.3.1.1	Manual Check	Resource Flows connect to a Service objects at both ends.		
17	17.3.1.1	Manual Check	Resource Flows are solid, straight, lines with 90-degree curves, when necessary.		
18		Manual Check	All exception reports have been reviewed and resolved.		

SvcV-1 Definitions Checklist

CR Number: ______________ **Date:** ______________

Approval Signatures:

Contractor Lead ________________________ **Government Lead** __________________________

			Reviewer Name:		
#	Source APG	BRS Report	Checklist Item Description	Modeler	Reviewer
1	17.2.2.2.2	Manual Check	Each SvcV-1 has a diagram description that explains what it represents and a brief description of the Service application.		
3	17.2.2.2.2	Manual Check	Each Service has a description of what the Service does.		

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
4	17.2.3	Manual Check	Each Resource Flow has a description of the data it represents.		

SvcV-1 Integration Checklist

CR Number: ______________ **Date:** ______________

Approval Signatures:

Contractor Lead ________________________ **Government Lead** __________________________

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	17.1.4	Manual Check	Each Service must have at least one System Function assigned.		
2	17.1.3	Manual Check	Resource Flows between Service objects reflect Need Lines from the OV-2.		
3	17.1.3	Manual Check	Each Resource Flow must link to at least one IE from the OV-3.		

B-18: SvcV-5 Operational Activity to Service Traceability

Svc-5 Operational Activity to Service Traceability

CR Number: _____________ **Date:** _____________

Approval Signatures:

Contractor Lead _______________________ **Government Lead** _________________________

			Reviewer Name:		
#	Source APG	BRS Report	Checklist Item Description	Modeler	Reviewer
1	18.2.1	Manual Check	Each Service must be associated with at least one leaf level Operational Activity.		
2	18.2.1	Manual Check	Each Service in the SvcV-1 is captured in the SvcV-5		

SvcV-5 Integration Checklist

CR Number: _____________ **Date:** _____________

Approval Signatures:

Contractor Lead _______________________ **Government Lead** _________________________

			Reviewer Name:		
#	Source APG	BRS Report	Checklist Item Description	Modeler	Reviewer
1	18.2.1	Manual Check	The SvcV-5 matrix only contains relationships between leaf-level Operational Activities and Services.		
3	18.2.1	Manual Check	Each Service must have at least one CBM Stakeholder.		
4	18.2.1	Manual Check	The SvcV-5 must be consistent with the ETP.		
5	18.2.1	Manual Check	Services shall link to System Functions.		

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
6		Manual Check	All exception reports have been reviewed and resolved.		

B-19: CV-2 Model Checklists

CV-2 Diagrams Checklist

CR Number: ______________ **Date:** ______________

Approval Signatures:

Contractor Lead ________________________ **Government Lead** __________________________

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	19.3.1.2	Manual Check	The Capability boxes are numbered sequentially relative to position.		
2	19.3.1.2	Manual Check	The top-level box of the diagram is centered on the diagram as permitted by the tool.		
3	19.3.1.2	Manual Check	Parent Capabilities are decomposed to at least two, but not more than nine, child capabilities.		
4	19.3.1.1	Manual Check	The CV-2 Capability Taxonomy follows modeling conventions: Names are Normal and Black font. The Capabilities box border shall be a solid black line. The Capabilities shall be white.		
5	19.3.1.1	Manual Check	Capability names follow modeling conventions: The font must be normal and Black. The Capability name begins with a RETURN character. Capability name falls within the Capability box border with no truncation indicators The Capability Box has a solid black line border. All capabilities should be colored Yellow.		
6	19.3.1.2	Manual Check	Capability boxes are numbered sequentially in the lower right corner. The number inside the Activity box shall match the last digit of the Activity number sitting outside the box beginning with a "C." Capability boxes are stair-stepped vertically and numbered in descending order appropriately.		

			Reviewer Name:		
#	**Source** **APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
7	19.3.1.2	Manual Check	Diagram name (Capability box label) shall use title case (first letter of each word capitalized, other letters lowercase), should be non-plural (exception approved by The BEA development team and can use only the special character "-". Any acronyms used in the Capability name must be from the approved acronym list that is part of the BEA AV-2.		
8	19.1.3	Manual Check	Each Capability term must be defined in the AV-2.		
9	19.1.3	Manual Check	Each acronym used in Capability descriptions must be defined in the AV-2.		
10	19.1.3	Manual Check	Operational Activities from the OV-5 are mapped to Capabilities in the CV-6.		
			General		
11	19.3.1.2	Manual Check	The CV-2 Capability Taxonomy diagram does not have a border.		
12	19.3.1.2	Manual Check	The CV-2 Capability Taxonomy diagram has a Doc Block for header information that includes the title of the diagram on one line, its creation/modification date, but no graphic comments. The Doc Block is in the extreme upper left corner of the diagram with a black border, no fill color and no truncation indicators.		
13	19.3.1.2	Manual Check	CV-2 Capability Taxonomy diagram has a narrative description of the diagram, using the diagram properties comment box to explain the Activities and their relationships.		
14		Manual Check	All exception reports have been reviewed and resolved.		

B-20: SV-10a Model Checklist

CR#: __________ **Date:** __________ **Approval Signatures:**
Contractor Team Lead __________________ **Government Model Lead** ___________________

SV-10a Model Checklist

			Reviewer Name:		
#	**Source APG**	**BRS Report**	**Checklist Item Description**	**Modeler**	**Reviewer**
1	14.4.2	Manual	Rule can be readily understood by any business or DoD party and is always interpreted the same.		
2	14.4.2	Manual	Rule is unambiguous.		
3	14.2.2	Manual	Rule is in declarative form – no reference to how, where, when, or who.		
4	14.2.2	Manual	No indication of how rule to be enforced (how).		
5	14.2.2	Manual	No indication of where to enforce the rule (where).		
6	14.2.2	Manual	No indication of when to enforce the rule (when).		
7	14.2.2	Manual	No indication of Events or Event sequence (when).		
8	14.2.2	Manual	No indication of who will enforce the rule (who).		
9	14.2.2	Manual	Rule is not procedural (use of “else” and “if”).		
10	14.2.2	Manual	Rule constrains (or alternatively permits).		
11	14.2.2	SV10a-009	Rule contains one of the key rule words such as “is,” “may,” “must,” “no,” “not,” “shall,” “should,” “will,” or “only if.”		
12	14.2.2	Manual	Words such as “can” are not used.		
13	14.2.2	Manual	Rule uses standard terminology such as the common language from the data model.		
14	14.2.2	Manual	Facts are explicitly expressed in the rule (no hidden facts or computations).		
15	14.2.2	Manual	Rule is written in *RuleSpeak*™ formal language.		
16	14.4.2	Manual	The rule does not have a plural subject.		
18	14.2.2	Manual	A time element is not the subject.		
19	14.2.2	Manual	Rule has explicit subject.		
20	14.4.2	Manual	Computations are the subject of the rule.		

			Reviewer Name:		
#	Source APG	BRS Report	Checklist Item Description	Modeler	Reviewer

SV-10a Integration Checklist

			Reviewer Name:		
#	Source APG	BRS Report	Checklist Item Description	Modeler	Reviewer
1	14.3.1	N/A	All required fields filled in properly from load.		
2	14.3.1	SV10a-006	Each Operative Derivation System Rule is linked to a System Function.		
3	14.3.1	SV10a-005	Each Operative Action System Rule is linked to a System Entity.		
4	14.3.1	SV10a-007	Each Structural Relational System Rule is linked to a System Entity.		
5	14.3.1	SV10a-013	Each Structural Definitional System Rule is linked to a System Entity.		
6	14.3.1	SV10a-012	Each System Rules must have either an Operative Category or a Structural Category		
7	14.3.1	SV10a-011	Each System Rule that is an Operative Category must be either a Derivation Type or an Action Type.		
8	14.3.1	SV10a-014	Each System Rule that is a Structural Category must be either a Relational or a Definitional Type.		
9	14.3.1	SV10a-015	All System Rules must have unique rule number associated with it.		

Appendix C : Glossary

Table C-1, Glossary, contains a list of terms and associated descriptions used in this document.

Table C-1, Glossary

Term	Description
A-0 Diagram	The special case of a one-box IDEF0 A-0, containing the top-level function being modeled and its Inputs, Controls, Outputs, and Mechanisms, along with statements of model purpose and viewpoint.
A0 Diagram	An IDEF0 diagram contains the first tier sub-activities under the A-0 diagram, their ICOM relationships, their Inputs, Controls, Outputs, and Mechanisms, along with a diagram description.
Abstract Processes	See Public Processes. [OV-6c]
Acronym	The initials of a standard phrase used in the BEA or ETP.
Activity Box	Represented by an enclosed rectangular box within which an operational function is performed in conducting the business of the enterprise.
Activity [OV-6c]	Business Process Modeling Notation uses the term "Activity" to mean work that can be performed within a Business Process. An activity can be atomic or non-atomic (compound). The types of activities that are a part of an OV-6c Event Trace Diagram are Process Step, Sub-Process and Task. OV-6c uses the term "Process Step" to avoid confusion with the term "Activity" used in OV-5 to mean "Operational Activity."
Action Assertion Business Rule	These rules concern some dynamic aspects of the business and specify constraints on the results that actions produce. There are three types of action assertions: − Condition: This is a guard or the "if" portion of an "if-then" statement. If the condition is true, it may signal the need to enforce or test additional action assertions. − Integrity Constraint: These must always be true (for example, a declarative statement). − Authorization: This restricts certain actions to certain human roles or users.
AND-Split	See "Fork (AND-Split)". [OV-6c]
AND-Join	See "Join (AND-Join)". [OV-6c]
Arrow	A directed line, composed of one or more arrow segments, that models an open channel or conduit conveying data or objects from source (no arrowhead) to use (with arrowhead). There are four arrow classes: Input Arrow, Output Arrow, Control Arrow and Mechanism Arrow (includes Call Arrow).
Arrow Label	A noun or noun phrase associated with an IDEF0 arrow or arrow segment, specifying its meaning.
Arrow Segment	A line segment that originates or terminates at a box side, a branch (fork or join), or a boundary (unconnected end).

Term	Description
Attribute	An Attribute is a property or characteristic that is common to some or all of the instances of an Entity. Attributes that identify Entities are key Attributes. Attributes that describe an Entity are non-key Attributes. Attributes are associated to one and only one Entity and represent the normalized view of Data Elements within DIV-2 Entities.
Artifact	A graphical object that shows additional information about a process that is not directly related to the Sequence Flow or Message Flow. There are three artifacts: Data Objects, Groups and Annotations. [OV-6c]
Association	An Association is used to link information with Flow Objects. An Association may or may not have direction. [OV-6c]
Availability	Timely, reliable access to data and information services for authorized users.
Boundary Arrow	An arrow with one end (source or use) not connected to any box on a diagram. Contrast with Internal Arrow.
Boundary Arrow Box	A rectangle, containing a name and number, used to represent a function.
Box Name	The verb or verb phrase placed inside an IDEF0 box to describe the modeled function.
Box Number	The number (0 to 9) placed inside the lower right corner of an IDEF0 box to uniquely identify the box on a diagram.
BPM Event	See “Event”.
BPM Process	See “Process”.
Branch	A junction (fork or join) of two or more arrow segments.
Branching Point	Branching points are Gateways within a Business Process where the flow of control can take one or more alternative paths. Synonymous with *Decision*. [OV-6c]
Bundling/Unbundling	The combining of arrow meanings into a composite meaning (bundling), or the separation of arrow meanings (unbundling), expressed by arrow join and fork syntax.
Business Capability	This is the ability to execute a specific course of action. It can be a single business enabler or a combination of business enablers (e.g., business processes, policies, people, tools and systems information) that assist an organization in delivering value to its customer.
Business Process	A Business Process is a set of activities that are performed within an organization or across organizations. A Business Process, as shown in an Event Trace Diagram, may contain more than one separate process. [OV-6c]
Business Rule	Listed in the OV-6a, a Business Rule is “a rule that is under business jurisdiction”, SBVR OMG. It is a constraint on an enterprise, a mission, operation, business, or architecture. A Business Rule describes what the business must or cannot do. A Business Rule is an atomic piece of business logic, specified declaratively, whose intent is to control, guide, or enhance behavior.
Call Arrow	A type of Mechanism Arrow that enables the sharing of detail between models (linking them together) or within a model.

Term	Description
Categorization Relationship	A relationship in which instances of both Entities represent the same real or abstract thing. One Entity, Supertype, represents the complete set of things the other Subtype represents a sub-type or sub-classification of those things. The Subtype may have one or more characteristics or a Relationship with instances of another Entity not shared by all Supertype instances. Each instance of the Subtype is simultaneously an instance of the Supertype.
Child Box	A box on a Child diagram.
Child Diagram	The diagram that details a Parent box.
Class Word	A Class Word is a word selected from a specified list that is used in an Attribute name to establish the general structure and domain of that Attribute.
Collaborative Process	A collaboration process depicts the interactions between two or more business Entities. This is shown as two or more processes communicating with each other. In OV-6c, collaborative processes also include processes from the same Business Entity, but commonly assigned to a different higher-level process. [OV-6c]
Collapsed Sub-Process	A collapsed Sub-Process is a graphical representation of a Process Step in which the details of the Sub-Process are not visible in the diagram. This is indicated by a "+" stereotype. [OV-6c]
Conditional Flow	A Sequence Flow that has a condition expression evaluated at run time to determine whether the flow will be used. [OV-6c]
Confidentiality	Assurance that information is not disclosed to unauthorized individuals, processes or devices.
Connecting Objects	Connecting objects connect Flow Objects together. There are three connecting objects: Sequence Flow, Message Flow and Association. [OV-6c]
Context	The immediate environment in which a function (or set of functions on a diagram) operates.
Context Diagram	A diagram that presents the context of a model, whose node number is A-n (n greater than or equal to zero). The one-box A-0 is a required A-0; that with node numbers A-1, A-2, are optional A-0s.
Control Arrow	The class of arrows that express IDEF0 Control, that is, conditions required to produce correct Output. Data or objects modeled as Controls may be transformed by the function, creating Output. Control arrows are associated with the topside of an IDEF0 box.
Core Business Mission (CBM) Stakeholder	The CBM Stakeholder for an OV-6c process is one or more of the Core Business Missions that is responsible for executing that process.
Criticality	The criticality assessment of the information being exchanged in relationship to the mission being performed.
Data-Based Decision	A Gateway in which the Decision represents a branching point where Alternatives are based on conditional expressions based on data contained within the outgoing Sequence Flow. [OV-6c]

Term	Description
Data Element	A Data Element is the smallest unit of stored data, which means that it cannot be broken down further, or that it makes no sense to break it down further. The Data Element, however, can inherit properties from a Data Domain. Data Elements are unique across the BMA and are associated with Attributes within the BEA.
Data Model	A graphical and textual representation of analysis that identifies the data needed by an organization to achieve its mission, functions, goals, objectives, and strategies and to manage and rate the organization. A data model identifies the Entities, domains (Attributes) and Relationships (or associations) with other data.
Data Object	Additional information on an OV-6c, which does not have any direct effect on the Sequence Flow or Message Flow but does show the data that may be passed, created, or consumed by the BPM Process. Data Objects are a mechanism to show how data is required or produced by Process Steps. A Data Object is considered an artifact because it does not have a direct effect on the Sequence or Message Flow of the process. [OV-6c]
Decision	Synonymous with "Branching Point." [OV-6c]
Decomposition	The partitioning of a modeled function into its component functions.
Default Flow	Sequence Flow, for Data-Based Exclusive Decisions or Inclusive Decisions, which shall be used only if all the other outgoing Conditional Flows are not true at run time. [OV-6c]
Derivation Business Rule	These rules concern algorithms used to compute a derivable fact from other terms, facts, derivations, or action assertions.
Description	Text description of mission or role being performed by the Node.
Diagram	A single unit of an IDEF0 Model that presents the details of a box.
Diagram Node Number	That part of a diagram's node references that corresponds to its Parent box node number.
End Event	An Event that indicates where the process concludes. [OV-6c]
Enterprise Sub-Services	Used in the SV-StdV Bridge, it describes the intersection between enterprise systems and Technical Services, and defines Standard attributes to bring order to that point.
Entity	An Entity is the representation of a set of real or abstract things (people, objects, places, events, ideas, combination of things, etc.) that are recognized as the same type because they share the same characteristics and can participate in the same Relationships.
Event	An Event is something that happens during the course of a Business Process. These Events affect the flow of the process and usually have a cause (trigger) or an impact (result). Events are represented as circles with open centers to allow internal markers to differentiate different triggers or results. There are three types of Events, based on when they affect the flow: Start, Intermediate and End. [OV-6c]
Event-Based Decision	The Decision represents a branching point where Alternatives are based on an Event that occurs at a particular point in the process. [OV-6c]
Event Name	Name of the Event that triggers the IE.

Term	Description
Exception Flow	Sequence Flow occurring outside the Normal Flow of the process and is based upon an Intermediate Event that occurs during the performance of the process. [OV-6c]
Exclusive Gateway (XOR)	An Exclusive Gateway restricts the flow such that only one of a set of alternatives may be chosen during runtime. There are two types of Exclusive Gateways: Data-based and Event-based. Also see "Inclusive Decision". [OV-6c]
Expanded Sub-Process	An expanded Sub-Process is a graphical representation of a Sub-Process in which the boundary of the Sub-Process icon is expanded and the details (a process) are visible within its boundary. [OV-6c]
Flow Object	Flow Objects are the main graphical elements to define the behavior of a Business Process. The three Flow Objects are Events, Process Steps and Gateways. [OV-6c]
Fork	The junction where an IDEF0 arrow segment (going from source to use) divides into two or more arrow segments. May denote unbundling of meaning.
Fork (AND-Split) [OV-6c]	Dividing a path into two or more parallel paths, where Process Steps can be performed concurrently, rather than sequentially. [OV-6c]
Function	An activity, process, or transformation (modeled by an IDEF0 box) identified by a verb or verb phrase that describes what must be accomplished.
Gateway	Used on an OV-6c, this Flow Object controls the divergence and convergence of multiple Sequence Flows. [OV-6c]
Grouped Attribute	Grouped Attributes bring together several Attributes in a particular order to form a group. The classic example of a Grouped Attribute is Person Name that brings together First Name, Middle Name and Last Name.
ICOM	The acronym of Input, Control, Output, Mechanism. A code that associates the Boundary Arrows of a Child diagram with the arrows of its Parent box; also used for reference purposes.
IDEF0 Model	A graphic description of a system or subject that is developed for a specific purpose and from a selected viewpoint. A set of one or more IDEF0 diagrams that depict the functions of a system or subject area with graphics, text and glossary

Term	Description
ICOM Arrow	Used on an OV-5, it represents the Input, Control, Output, or Mechanism that defines information relationships in an Activity Model. − Input: Information received from another Operational Activity, either internal or external to the model, which is needed for the given Operational Activity to be carried out. − Control: Information that affects the way an activity is performed or that constrains that activity. Primary sources are policies, regulations and laws. CBM Enterprise Standards are also reflected as Controls to emphasize the impact on a specific activity of those business transformation concepts. In the BEA, there are two types of Controls: External and Internal. External Controls are decomposed from the LRP Parent. Internal Controls are Enterprise Standards that are created as Outputs from other Operational Activities within the BEA OV-5 Activity Model. − Output: Information that has been transformed or created by the Operational Activity and is sent to another internal Operational Activity or to an external activity (one outside the scope of the model/viewpoint). − Mechanism: Resource used to perform the activity. Mechanisms will be CBMs and those Systems or Enterprise Standards defined by the CBM Executives.
Identified Relationship	A specific connection relationship in which every Attribute in the Primary Key of the Parent is contained in the Primary Key of the Child Entity.
Inclusive Decision	A branching point (Gateway) where Alternatives are based on conditional expressions contained within the Sequence Flow. Since each path is independent, all combinations of the paths may be taken. Also see "Exclusive Gateway". [OV-6c]
Integrity	Quality of an IS reflecting the logical correctness and reliability of the operating system; the logical completeness of the hardware and software implementing the protection mechanisms; and the consistency of the data structures and occurrence of the stored data. Note that, in a formal security mode, integrity is interpreted more narrowly to mean protection against unauthorized modification or destruction of information.
Intermediate Event	An Event that occurs between a Start Event and an End Event. It affects the flow of the process, but will not start or (directly) terminate the process. [OV-6c]
Information Exchange	Listed in the OV-3, shows the Information Exchanges between two Operational Nodes. A corresponding leaf-level Activity Input or Output ICOM is associated to the IE with the same name, definition and CBM Stakeholder.
Information Exchange Identifier	Identifier for the IE – usually based on the relevant Need Line identifier; should be unique for the architecture.
Input Arrow	The class of arrows that expresses IDEF0 Input that is the data or objects that are transformed by the function into Output. Input arrows are associated with the left side of an IDEF0 box.
Interface	A shared boundary across which data or objects are passed; the connection between two or more model components for the purpose of passing data or objects from one to the other.

Term	Description
Internal Arrow	An Input, Control, or Output arrow connected at both ends (source and use) to a box on a diagram. Contrast with Boundary Arrow.
Interoperability Level Required	Level of information systems interoperability, or other interoperability measure, required.
Join	The junction at which an IDEF0 arrow segment (going from source to use) merges with one or more other arrow segments to form a single arrow segment. May denote bundling of arrow segment meanings.
Join (AND-Join) [OV-6c]	A Gateway that combines two or more parallel paths into one path. Synonymous with "AND-Join" and "synchronization". [OV-6c].
Lane	A Lane is a sub-partition within a Pool and extends the entire length of the Pool. Lanes are used to organize and categorize Process Steps within a Pool (representing a single Business Entity). [OV-6c]
Leaf-level	Refers to the lowest level of detail described for a given Operational Activity, system, or process model. It represents the lowest level of decomposition of higher-level models needed to represent objects and relationships of interest to the topic under study.
Level Identifier	If using hierarchical decomposition of Nodes: identifier that corresponds to the Node's place in the Node hierarchy.
Logical Data Model	The DIV-2 data model that provides the structure for organizing the data as well as the metadata need for an understanding of the data. The DIV-2 can serve as a guide for the acquisition and evaluation of systems by assisting portfolio managers in quantitatively assessing the contents of their portfolios in the evaluation of how well the alternative solutions meet the data needs of the BMA.
Mandatory Non-Identified Relationship	A non-identified Relationship in which an instance of the Child Entity must be related to an instance of the Parent Entity.
Mechanism Arrow	The class of arrows that express IDEF0 Mechanism, that is, the means used to perform a function; includes the special case of call arrow. Mechanism Arrows are associated with the bottom side of an IDEF0 box.
Merging (OR-Join)	Merging exclusively combines two or more paths into one path. A Merge Gateway represents merging. [OV-6c]
Message Flow	A Message Flow shows the flow of messages between two Entities that are prepared to send and receive them. Two separate Pools in the Diagram will represent the two business Entities. [OV-6c]
Model Note	A textual comment that is part of an IDEF0 diagram used to record a fact not otherwise depicted.
Name	Name or label of Node box on diagram.

Term	Description
Need Line	Shown on an OV-2, it documents the requirement to exchange information between Operational Nodes. Arrows on Need Lines indicate the direction of the information flow. Each arrow only indicates that there is a need for some kind of information transfer between the two connected nodes, not how the information transfer is implemented.
Need Line Identifier	Identifier for the Need Line that carries the exchange.
Node	A box from which Child boxes originate; a Parent box.
Node Index	A listing, often indented, showing nodes in an IDEF0 Model in outline order. Same meaning and content as Node Tree.
Node Reference	A code assigned to a diagram to identify it and to specify its position in the model hierarchy; composed of the model name (abbreviated) and the diagram node number, with optional extensions.
Node Tree	The graphical representation of the Parent-Child relationships between the Nodes of an IDEF0 Model, in the form of a graphical tree.
Non-Identified Relationship	A specific connection Relationship in which at least one of the Attributes contained in the Primary Key of the Parent does not participate in the Primary Key on the Child Entity.
Non-Specific Relationship	Non-specific Relationship: A Relationship in which an instance of either Entity can be related to a number of instances of the other.
Normal Flow	Normal Sequence Flow refers to the flow that originates from a Start Event and continues through Process Steps via alternative and parallel paths until it ends at an End Event. [OV-6c]
Normal Form	Normal form is the condition of an Entity relative to satisfaction of a set of normalization theory constraints on its attribution. A specific normal form is achieved by successive reduction of an Entity from its existing condition to some more desirable form.
Normalization	The process of refining and regrouping Attributes in Entities according to the normal forms.
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. [OV-5]
Operational Activity Name	Name of the Operational Activity (at the originating Node of the Need Line) that produces the IE.
Operational Node	Shown in an OV-2, it describes what type of mission or role will be performed within an organizational unit. It is a job performed within an organizational unit.
Operational Node Name	Name of the Operational Node that produces the IE.
Optional Non-Identified Relationship	A non-identified Relationship in which an instance of the Child Entity can exist without being related to an instance of the Parent Entity.

Term	Description
OR-Split	Synonymous with "Branching Point." [OV-6c]
Organizational Unit	An Organizational Unit is a business organized in terms of roles also known as Operational Nodes. Each Organizational Unit includes a list of roles performed within that Organizational Unit only. In the BEA, CBMs represent Organizational Units.
Output Arrow	The class of arrows that express IDEF0 Output; that is, the data or objects produced by a function. Output arrows are associated with the right side of an IDEF0 box.
Parent Box	A box that is detailed by a Child diagram.
Parent Diagram	A diagram that contains a Parent box.
Participant	A Participant is a single business Entity or a business role, which controls or is responsible for a Business Process. A Pool represents a Participant in the process. [OV-6c]
Periodicity	How often the IE occurs; may be an average or a worst-case estimate and may include conditions (for example, wartime or peacetime).
Persistent Data	Data that has been saved and remains available even when it is not being used.
Pool	A Pool represents a Participant – a single Business Entity – in a process. It also acts as a Swimlane and a graphical container for partitioning a set of Process Steps from other Pools. [OV-6c]
Primary Key Attribute	Represented by one or more textual names in the upper portion of the Entity box. Primary Key Attributes contain characteristics that uniquely define a single instance of an Entity.
Private Business Process	Private Business Processes are those internal to a specific organization and are the types of processes that have been generally called workflow or BPM Processes. [OV-6c]
Process	Used on an OV-6c, this denotes a set of activities performed within a business organization, where an activity (not to be confused with the OV-5 usage for 'Operational Activity) is a generic term for work that a business organization performs. A BPM Process is depicted as a graph of Flow Objects, which are a set of other Process Steps and the controls that sequence them. [OV-6c]
Process Break	A location in a process that shows where an expected delay will occur within a process. An Intermediate Event is used to show the actual behavior. [OV-6c]
Process Step	Work that can be performed within a Business Process. A Process Step can be atomic or non-atomic (compound). The types of Process Steps that are a part of an OV-6c Event Trace Diagram are: Process Step, Sub-Process and Task. The term "Process Step" is a synonym for the Business Process Modeling Notation term "Activity." OV-6c uses the term "Process Step" to avoid confusion with the OV-5 term "Activity," representing an "Operational Activity." [OV-6c]
Protection Duration	How long the information must be safeguarded.
Protection Suspense Calendar Date	The calendar date on which the designated level of safeguarding discontinues.

Term	Description
Protection Type Name	The name for the type of protection.
Public Processes	Public processes represent the interaction between a private Business Process and another process or participant. Only those activities that are used to communicate outside the private Business Process, plus the appropriate flow control mechanisms, are included in the public process. All other internal activities of the private Business Process are not shown in the public process. Synonymous with "Abstract Process". [OV-6c]
Purpose	A brief statement of the reason for a model's existence.
Receiving Operational Activity	The identity of the Operational Activity consuming the information.
Receiving Operational Node Name	Name/identifier of the Operational Node that consumes the information.
Relationship	A Relationship is an association between two Entities or between instances of the same Entity.
Relationship Cardinality	Relationship Cardinality is the number of Entity instances that can be associated with each other in a Relationship.
Relationship Cardinality Constraint	A Relationship Cardinality Constraint is a limit on the number of Entity. Instances that can be associated with each other in a Relationship.
Relationship Name	A Relationship Name or label is "a verb or verb phrase, which reflects the meaning of the Relationship expressed between the two Entities shown in the diagram on which, the name appears."
Semantics	The meaning of the syntactic components of a language.
Sending Operational Activity Name	The identity of the Operational Activity producing the information.
Sending Operational Node Name	Name/identifier of Operational Node that produces the information.
Sequence Flow	Arrows that show the order that Process Steps will be performed in a process.
Standard	An agreed upon means that establishes uniform engineering and technical requirements to implement all or part of a Technical Service.
Start Event	An Event that indicates where a particular process will start. A process must have one or more Start Events. [OV-6c]
Stereotype	A graphical icon that indicates the type of Flow Object. [OV-6c]

Term	Description
Structural Assertion Business Rule	These rules concern mission or business CBM terms and facts that are usually captured by the Entities and Relationships of Entity-Relationship models. They reflect static aspects of Business Rules that may also be captured in the DIV-2. − Terms: Entities. − Facts: Association between two or more terms (for example, relationship).
Sub-Process	A compound Process Step that is included within a process. It is compound in that it can be broken down into a finer level of detail (a process) through a set of Sub-Processes. A Sub-Process may be shown graphically as a collapsed or expanded Sub-Process. [OV-6c]
Swimlanes	Swimlanes group the primary modeling elements by organization or other criteria. There are two Swimlane objects, Pools and Lanes. [OV-6c]
Synchronization	See "Join (AND-Join)". [OV-6c]
Syntax	Structural components or features of a language and the rules that define relationships among them.
Systems Resource Flow	Listed in the SV-6, it represents resource flows between System Functions and may include additional information assurance or performance attributes to characterize the exchange. The data in the SRF is represented using data Entities and/or Data Elements within the DoDAF DIV-2 architecture model.
System Entity	Shown on a SV-1, it represents computer systems, family of systems or systems of systems. A System Entity resides within a System Node and may contain one or more System Functions.
System Function	Used in the SV models, this set of organized actions produces a defined automated output when given specific data inputs. Within the context of DoDAF, a System Function transforms the data in an IE as constrained by operational and structural Business Rules.
System Interface	Shown on an SV-1, it represents the resource flow between System Entities.
System Node	Shown on an SV-1, it represents the system capabilities that are required to support the business practices that are described in the Operational View.
Task	An atomic Process Step that is included within a process. A Task is used when the work in the Process Step is not broken down to a finer level of process model detail. [OV-6c]
Technical Service	Listed in the StdV-1 with its constituent Standards, represents a technical capability designed to support an Enterprise Sub-Service.

Term	Description
Technology Service Area	Shown in the StdV-1, it groups similar Technical Services together for increased organization and comprehension. There may be one or more Technical Services in a Technology Service Area. The current StdV-1 takes its highest-level structure from the DoD Enterprise Architecture Technical Reference Model (EA TRM). It contains four Technology Service Areas, drawn from the Core Service Areas of the DoD EA TRM.
Term	Used in the BEA or ETP, this is a word or group of words designating a selected concept.
Text	An overall textual (non-graphical) comment about an IDEF0 graphic diagram.
Text Annotation	Text Annotations are mechanisms (Artifacts), attached with an Association, for a process architect to provide additional information for the reader of the Process Diagram. [OV-6c]
Timeliness	Required maximum allowable time of exchange from Node to Node (in seconds).
Title	A verb or verb phrase that describes the overall function presented on an IDEF0 diagram; the title of a Child diagram corresponds to its Parent box name
Transaction Type	Descriptive field that identifies the type of the IE.
Triggering Event	Brief textual description of the Event(s) that trigger the IE.
Tunneled Arrow	An arrow (with special notation) that does not follow the normal requirement that each arrow on a diagram must correspond to arrows on related Parent and Child diagrams.
Uncontrolled Flow	Sequence Flow that is not affected by any conditions or does not pass through a Gateway. [OV-6c]
View	A subset of Entities, Attributes and Relationships that has meaning from a specific perspective. For example, a view can show only that portion of the data model that is relevant to a specific domain or to a macro Process Step.
Viewpoint	A brief statement of the perspective of the model.
XOR	See “Exclusive Gateway”. [OV-6c]

Appendix D : CBM AV-1 Template

<CBM Name (Acronym>
Overview and Summary Information (AV-1)
Version 9.0, March 15, 2012

(Description of the purpose of this document) The AV-1 is an executive-level summary of the DoD <*CBM Name*> Core Business Mission (<*Acronym*> CBM). Initially, the AV-1 is used to focus the <*Acronym*> CBM development effort and document its scope. The final version will include findings and recommendations from the effort.	
Architecture Project Identification	
CBM Name	*<Core Business Mission Name>*
CBM Description	*<The CBM Description. This is the authoritative source for the CBM Description in the ETP narrative and for the BEA in SA.>*
Architect	DoD Office of the Deputy Chief Management Office (ODCMO)
Developed By	*<Lead Core Business Mission >* *<Supporting Organizations>*
Assumptions **and Constraints**	The <*CBM Name*> CBM: • Will make maximum reuse of existing BEA models with changes only made when necessary. • Will address only DoD enterprise-level business and strategic plans, goals, objectives, and strategies, which are the primary drivers for the BEA. *<Additional list of CBM specific assumptions and constraints placed on the architecture>*
Approval Authority	The Deputy Secretary of Defense, acting through the Defense Business Systems Management Committee (DBSMC).
Date Completed	Architecture specified content freeze date, and final release date.
LOE and Development Costs	Level of effort and projected and actual costs to develop the CBM Models may be requested from the Office of the Deputy Chief Management Officer (ODCMO).

Business Outcome	*<This section should be from a business perspective and identify how DoD's Business Mission Area will be improved as a result of the BEA development work. Avoid architecture style language. This is the business case for doing the work. >*
Scope: Architecture View and Models Identification	
Models Developed	*<List of architecture models to be developed or updated for this deliverable.>*
CBM Capabilities	*<List the BEA Business Capabilities for which the CBM is the primary sponsor.>*
Scope	*<Summary of BEA Improvements Proposals (BIPs) approved for development for the CBM in current deliverable.>*
Time Frames Addressed	The BEA is the "To Be" architecture for transformation efforts at DoD. The current BEA "To Be" end state has intermediate time frames for implementation addressed in the Enterprise Transition Plan (ETP).
Organizations Involved	*<Identify the CBMs or other stakeholders involved in the BEA development effort>*

Purpose and Viewpoint	
Purpose **(Problems, Needs, Gaps)**	*<A brief description that explains the need for the CBM content in the architecture, including significant issues, problems, needs, or gaps that this priority is intended to address or target for resolution.>*
Questions to be Answered	*<CBM specific questions that serve as discovery tools to identify required data, activities and rules aligned with the BEA Purpose and Intended Uses.>*
Architecture Viewpoint	*<The viewpoint from which the architecture is being developed.>* *Sample from HRM:* *"HRM-CBM will be developed from a personnel management perspective focusing on using strategic plans, key DoD enterprise-level processes and information. The enterprise-level deals with business capabilities that are DoD wide as established by statute, policy, or longstanding practice and includes the systems and Enterprise Standards that support those capabilities."*
Context	
Mission	*<A functional statement from the Core Business Mission Area as it pertains to the CBM. Align with the ETP.>*
SMP Goals	*<List related SMP Goals. >*
SMP Key Initiatives	*<List related SMP Key Initiatives. >*

SMP Measures	*<List related SMP Measures under Key Initiatives as they related to the BEA.>*
Rules, Conventions, and Criteria	**Rules:** The <*CBM Name*> CBM adheres to the DoD Architecture Framework (DoDAF) Version 2.0. **Conventions:** The conventions and methodology to be followed are documented in the BEA Development Methodology and the Architecture Product Guide. **Criteria:** ODCMO establishes detailed evaluation criteria for the delivery. **Information Assurance Posture:** The <*CBM Name*> CBM information confidentiality, integrity, and availability must be protected to the extent required by applicable DoD policy.
BEA Tasking / Linkages to Other Architectures	**Tasking --** The 2005 National Defense Authorization Act (NDAA) requires architectures to assess and maintain investments throughout the DoD BMA. **Linkages to Other Architectures** – BEA is linked to the Federal Enterprise Architecture (FEA) Business Reference Model through the DoD EA Reference Models and federated with Component and program architectures through tiered accountability. *<Additional list of architectures to which the CBM content in the BEA may be linked.>*
Tools and File Formats to be Used	
IBM Rational System Architect v 11.47, Microsoft SQL Server, Word, Access, and Excel.	

Note: *Text in italics to be added by CBMs.*

Appendix E : BEA Improvement Proposal Template

BEA Improvement Proposal V 1.4

Section 1 BIP Administrative Information

This section contains the tracking data for the BIP. This provides information as to the origin of the BIP and insight into setting the priority of the BIP for scheduling.

Field Name	
BIP Name	
BIP Identifier	
BEA Release	
Business Owner Name	
IRB	| ***FM*** | | ***HRM*** | | ***RPILM*** | | ***WSLM/MSSM***
Assignee Name	
Submitter Name	
Submission Date	
Priority	| ***High*** | | ***Medium*** | | ***Low***
Status	***Informal Draft*** / ***Working Draft*** / ***Formal Draft*** / ***Submitted*** / ***Approved*** / ***Updated*** / ***Unapproved***
Status Date	
Approver Name	
Signature	
Approval Date	

Reference Material

Section 2 BIP Description

This section describes the intent of the BIP and the level of work effort to satisfy the BIP. This provides insight into scheduling the BIP for implementation.

BIP Improvement Description

	Problem or Need	*Business Outcome*	*Completion Criteria*	*LOE*
1				
2				
3				
4				
5				
6				

Total Estimated Level of Work Effort (in Days) _____

Has a proposed solution been developed within your community?

If yes, can the solution be provided to A&IM for analysis?

Guiding Principle Addressed: Optional for CBMs – Entered by BEA

	Strategic Alignment		*Streamline*

	Standardize		***Stovepipe Elimination***
	Simplify		***Systems and Services***

The definitions of the Guiding Principles can be found at: `http://www.dod.mil/`

Section 3 BIP Scope

This section denotes the scope of the BIP to provide an indication of the work effort. There are four parts each of which contains check boxes. Enter an "X" in front of the affected items.

Affected Business Capability or New

	Collect and Disburse		*Manage Military Health Services*
	Conduct Program Management		*Manage Organization*
	Deliver Property and Forces		*Manage Payment*
	Develop Personnel		*Manage Personnel Sustainment*
	Dispose or Return Property and Materiel		*Manage Quality of Life and Morale, Welfare and Recreation*
	Environmental Liabilities Identification and Validation		*Manage Receipt and Acceptance*
	Financial Reporting		*Manage Recruiting and Accessions*
	Forecast, Plan, Program, Budget and Funds Distribution and Control		*Manage Request*
	Hazardous Material Process Controls and Information Management		*Manage Retirement and Separation*
	Manage Acquisition Oversight Integration		*Manage Sourcing*
	Manage Assignment and Transfer		*Manage Travel*
	Manage Benefits		*Managerial Accounting*
	Manage Compensation and Reimbursement		*Monitor Commercial Request for DoD Technology Export*
	Manage Financial Assets and Liabilities		*Perform Asset Accountability*
	Manage General ledger		*Perform Build and Make and Maintenance and Sustainment*
	Manage Human Resource Information		*Real Property Acceptance*
	Manage Human Resource Information Security		*Real Property Inventory*

Affected Enterprise System, Enterprise Standard or New (DBSAE and ETP)

	ASAS - Acquisition Spend Analysis Service		*GEX - Global Exchange Service*
	BEIS - Business Enterprise Information Services		*HMIRS - Hazardous Materials Information Resource System*
	CAMS-ME - Capital Asset Management System - Military Equipment		*HMPC&IMR - Hazardous Materials Process Controls & Information Management Requirements*
	CCR - Central Contractor Registration		*IGT - Intragovernmental Transactions*

	CPARS - Contractor Performance Assessment Reporting System		*IUID - Item Unique Identifier*
	DAI - Defense Agencies Initiative		*KBCRS - Knowledge Based Corporate Reporting System*
	DAMIR - Défense Acquisition Management Information Retrie val		*MILS - Military Standards*
	DCPDS - Defense Civilian Personnel Data System		*ORCA - Online Representations and Certifications Application*
	DIMHRS - Defense Integrated Military Human Resources System		*PPIRS - Past Performance Information Retrieval System*
	DoD EMALL -DoD Electronic Mall		*RFID - Radio Frequency Identification*
	DTS -Defense Travel System		*RPAD - Real Property Asset Database*
	EDA - Electronic Document Access		*RPAR - Real Property Acceptance Requirements*
	EFD -Enterprise Funds Distribution		*RPCIPR - Real Property Construction In Progress Requirements*
	EL - Environmental Liabilities		*RPIR - Real Property Inventory Requirements*
	EPLS - Excluded Parties List System		*RPUIR - Real Property Unique Identifier*
	eSRS - Electronic Subcontracting Reporting System		*SFIS - Standard Financial Information Structure*
	FedBizOpps – Federal Business Opportunities		*SPOT - Synchronized Pre-deployment and Operational Tracker*
	FedReg - Federal Agency Registration		*SPS - Standard Procurement System*
	FedTeDs -Federal Technical Data Solution		*WAWF - Wide Area Workflow*
	FPDS-NG - Federal Procurement Data System-Next Generation		*WDOL - Wage Determinations Online*
	FYDP - Future Years Defense Program		

Affected BEA Models and Other

	AV-1 – Overview and Summary Information		***SV-5 – Operational Activity/ Systems Functions Matrix***
	AV-2 – Integrated Dictionary		***SV-6 – Systems Resource Flow Matrix***
	OV-2 – Operational Resource Flow Description		***StdV-1 – Standards Profile***
	OV-3 – Operational Resource Flow Matrix		***DFMIR – Defense Financial Management Improvement***
	OV-5 – Operational Activity Models		***Visualization***
	OV-6a – Business Rule Definitions		***Compliance Guidance***
	OV-6c – Event Trace Diagrams		***ACART* - Architecture Compliance and Requirements Traceability**
	DIV –Data and Information Viewpoint (Includes DIV-1 Conceptual, DIV-2 Logical Data Models)		***LRP – Laws, Regulations and Policies***

	SV-1 – Systems Interface Diagrams		*ETP – Enterprise Transition Plan*

Section 4 BIP Impacts

This provides insight into the collaboration work effort across the CBMs and within DoD and the BEA.

Stakeholder Impacted

	AV – Acquisition Visibility		***FV – Financial Visibility***		***PV – Personnel Visibility***
	CSE – Common Supplier Engagement		***MV – Materiel Visibility***		***RPA – Real Property Accountability***
	ENT - Enterprise				

End to End Impacted

	Acquire to Retire		***Procure to Pay***		***Proposal to Reward***
	Budget to Report		***Environmental Liabilities***		***Market to Prospect***
	Hire to Retire		***Cost Management***		***Prospect to Order***
	Order to Cash		***Concept to Product***		***Deployment to Retrograde/Redeployment***
	Plan to Fulfill		***Service Request to Resolution***		***Service to Satisfaction***

Section 5 BIP Risks

This section denotes any *Risks* that may be encountered by the implementation of the BIP or encountered in by not implementing the BIP. This provides insight into the priority of the BIP.

Risk	***/ None***
1.	
2.	
3.	
4.	

Section 6 BIP Dependencies

This section denotes any *Dependencies* on other BIPs or work that must be completed before this BIP can be implemented, and any other BIPs or work that are dependent on the implementation of this BIP. This provides insight into the implementation sequence of BIPs.

Dependency	***/None***
1.	
2.	
3.	
4.	

Section 7 BIP Version History

This section reports the revisions to the BIP. The revisions are automatically captured as changes are entered into the CM tool.

Version	***Date***	***Description of Revision***	***Approver***

Appendix F: Acronym List

Acronym	Definition
A-0	Context Diagram (DoDAF OV-5)
APG	Architecture Model Guide
AT&L	Acquisition, Technology and Logistics
AV	All Viewpoint (DoDAF) Acquisition Visibility Architecture Verification
AV-1	Overview and Summary
AV-2	Integrated Dictionary
BDM	BEA Development Methodology
BEA	Business Enterprise Architecture
BIP	BEA Improvement Proposal
BMA	Business Mission Area
BPMN	Business Process Modeling Notation
BRM	Business Reference Model
BRS	BEA Reporting Service
C4ISR	Command, Control, Communications, Computers, Intelligence, Surveillance and Reconnaissance
CADM	Core Architecture Data Model
CBM	Core Business Mission
CIO	Chief Information Officer
CL	Confidentiality Level
CM	Configuration Management
COTS	Commercial Off-the-Shelf
CR	Change Request

Acronym	Definition
CRM	Consolidated Reference Model
CSE	Common Supplier Engagement
CV	Capability Viewpoint
CV-1	Capability Taxonomy
CV-6	Capability to Operational Activity Mapping
DAMIR	Défense Acquisition Management Information Retrie val
DFMIR	Defense Financial Management Improvement Rules
DISR	DoD IT Standards Registry
DITPR	Defense Information Technology Portfolio Repository
DIV	Data and Information Viewpoint
DIV-1	Conceptual Data Model
DIV-2	Logical Data Model
DIV-3	Physical Data Model
DM2	DoDAF Meta-model
DoD	Department of Defense
DoDAF	Department of Defense Architecture Framework
DOTMLPF	Doctrine, Organization, Training, Materiel, Leadership and Education, Personnel and Facilities
DTS	Defense Travel System
E2E	End-to-End Business Flows
EA	Enterprise Architecture
ERP	Enterprise Resource Planning
ESM	Enterprise Service Management
ETD	Event Trace Diagram

Acronym	Definition
ETP	Enterprise Transition Plan
F&R	Findings and Recommendations
FEA	Federal Enterprise Architecture
FFMIA	Federal Financial Management Improvement Act
FM	Financial Management
FMS	Financial Management System Foreign Military Sales
FoS	Family of Systems
FV	Financial Visibility
GFI	Government Furnished Information
HRM	Human Resources Management
HTML	Hypertext Markup Language
IA	Information Assurance
IAS	Information Assurance and Security
ICOM	Input, Control, Output, Mechanism
IDEF0	Integrated Definition for Function Modeling
IDEF1X	Integrated Definition for Data Modeling
IE	Information Exchange
ISWG	Information Technology Standards Working Group
IT	Information Technology
IV&V	Independent Verification and Validation
JCA	Joint Capability Area
JCAMS	Joint Capability Area Management System
LRP	Laws, Regulations and Policies

Acronym	Definition
MAC	Mission Assurance Category
MSSM	Materiel Supply and Service Management
MV	Materiel Visibility
OA	Operational Activity
OMB	Office of Management and Budget
OMG	Object Management Group
OV	Operational Viewpoint
OV-2	Operational Resource Flow Description
OV-3	Operational Resource Flow Matrix
OV-5	Common to both OV-5a and OV-5b
OV-5a	Operational Activity Decomposition Tree
OV-5b	Operational Activity Model
OV-6a	Operational Rules Model
OV-6c	Event Trace Description (Referred to as a Business Process Model in BEA)
OV-7	Logical Data Model (Replaced with DIV-1 and DIV-2)
OWL	Web Ontology Language
PV	Project Viewpoint (DoDAF) Personnel Visibility
RDF	Resource Description Framework
RIF	Rule Interchange Format
RPA	Real Property Accountability
RPILM	Real Property and Installations Lifecycle Management
SA	System Architect (IBM Rational System Architect)
SBVR	Semantics of Business Vocabulary and Business Rules

Acronym	Definition
SDO	Standards Develop Organization
SE	System Entity
SF	System Function
SFIS	Standard Financial Information Structure
SME	Subject Matter Expert
SOA	Service Oriented Architecture
SRF	System Resource Flow
StdV	Standards Viewpoint
StdV-1	Standards Profile
SV	Systems Viewpoint
SV-1	Systems Interface Description
SV-5	Operational Activity to System Function Traceability Matrix
SV-6	Systems Resource Flow Matrix
SV-8	Systems Evolution Description
SV-10a	Systems Rules Model
SvcV	Services Viewpoint
SvcV-1	Services Context Description
SvcV-5	Operational Activity to Service Traceability Matrix
SVG	Scalable Vector Graphic
TRM	Technical Reference Model
WSLM	Weapon System Lifecycle Management
XML	Extensible Markup Language

(This page intentionally left blank.)