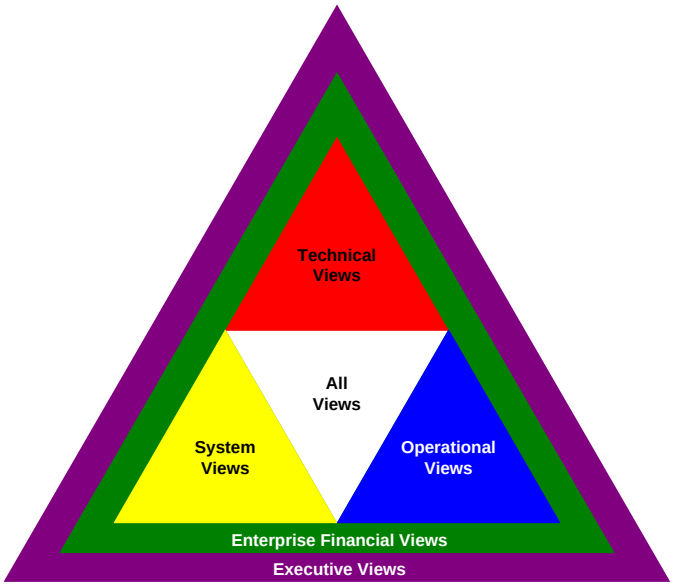
# Federal Aviation Administration

# National Airspace System Enterprise Architecture (NAS EA) Framework




## Version 3.0

**January 4, 2010**

# Version Control

| Version | Date | Author | Change Description |
|---|---|---|---|
| 1.0 | September 30, 2006 | SETA-II | Initial |
| 2.0 | September 30, 2007 | SETA-II | Introduces NAS EA Meta Model. Redefines set of hierarchical elements. Restructures product guides to emphasize NAS EA Meta Model and data-centricity. Restructures AV-1 Template. |
| 2.1 | September 30, 2008 | SETA-II | Reflects NAS EA Meta Model updates: removal of relationships between Operational Activity and Systems Function Input/Output and Systems Function to Operational Activity Input/Output. Adds new "Actor" architecture element, attributes, relationships, and rules. Adds new "source" and "sink" Systems Function Input/Output attributes and relationships. Reformulates *Rules* and updates *Sample Graphical Models* for each product development guide. Clarifies AV-1 Template treatment of alternative solutions and usage of Architecture Impact Assessment (AIA). |
| 3.0 | January 4, 2010 | SETA-II | Combines NAS EA Framework Volumes I, II, and III v2.1 into one comprehensive document. Establishes new Framework concepts for how the NAS EA fits into the rest of the ATO and the Agency. Presents a more flexible and less prescriptive set of guidelines for architecture development and analysis. |

Recommended by:

Jesse Wijntjes
ATO NAS Chief Architect

1-6-10
Date

Approved by:

James Williams
Director, Systems Engineering & Safety
Chairman, Enterprise Architecture Board

1-6-10
Date

# Table of Contents

**1 INTRODUCTION ..... 1**

1.1 Background ..... 1
1.2 Purpose ..... 1
1.3 Document Structure ..... 1

**2 FORM AND STRUCTURE ..... 3**

2.1 Levels of Enterprise Architecture ..... 3
2.2 NAS EA Conceptual Data Model ..... 5
2.3 Architecture Products ..... 7
2.4 Enterprise-Level Requirements ..... 16
2.5 Project Level Baselines and Planning ..... 16
2.6 JPDO-NextGen — External Stakeholders ..... 17

**3 ENTERPRISE-LEVEL PROCESSES AND PRACTICES ..... 19**

3.1 Service Roadmap Development ..... 19
3.2 Infrastructure Roadmap Development ..... 19
3.3 Sub-Capability Roadmap Development ..... 22
3.4 Funding Profile Development ..... 22
3.5 Enterprise-Level Architecture Development ..... 22

**4 PROJECT-LEVEL PROCESSES AND PRACTICES ..... 25**

4.1 Project-Level Architecture Development ..... 25
4.2 Project-Level Architecture Analysis ..... 29

**A APPENDIX: TEMPLATES ..... A-1**

A.1 AV-1 Overview and Summary Information ..... A-2
A.2 AV-2 Integrated Dictionary ..... A-15
A.3 OV-5 Operational Activity Model ..... A-16
A.4 OV-6c Operational Event-Trace Description ..... A-18
A.5 SV-1 Systems Interface Description ..... A-19
A.6 SV-2 Systems Communications Description ..... A-20
A.7 SV-4 Systems/Services Functionality Description ..... A-21
A.8 SV-6 Systems Data Exchange Matrix ..... A-23
A.9 SV-7 Systems Performance Parameters Matrix ..... A-24
A.10 SV-10c Systems Event –Trace Description ..... A-25
A.11 SV-11 Physical Schema ..... A-26
A.12 Infrastructure Roadmaps ..... A-27
A.13 Sub-Capability Roadmaps ..... A-30
A.14 Funding Profile ..... A-31

# 1 INTRODUCTION

## 1.1 Background

Since the release of the National Airspace System Enterprise Architecture (NAS EA) Framework v1.0 and subsequent versions, the general environment for building architectures has changed. Enterprise architecture acceptance in the Federal Aviation Administration (FAA) is gaining momentum. Commitment toward net-centric operations continues to mature. Greater visibility into the enterprise architecture is underway. Consistency and understandability of the enterprise is as important as ever.

The updated NAS EA Framework v3.0 streamlines the current documentation into one comprehensive volume. Version 3.0:

- Clarifies the bounds of the NAS EA and its environment.
- Incorporates net-centric structures as recognized in the DoDAF in response to the FAA's migration toward net-centric operations.
- Eliminates several complex and prescriptive notions from the previous volumes, balancing a greater flexibility within a structured environment.
- Provides an improved vehicle to present consistent architectural representation to a broader, more diverse community.

## 1.2 Purpose

The NAS EA Framework reflects core aspects of architecture development and use, aligning closely to the Department of Defense Architecture Framework (DoDAF). Viewed as a tailored interpretation and extension of DoDAF, this document establishes a structured and repeatable method for building architectures for the FAA's Air Traffic Organization (ATO). It defines and describes the products and processes that apply to architecture development at any level (enterprise, service unit, project) for the continued transformation and evolution of information technology comprising the National Airspace System.

## 1.3 Document Structure

This document is organized as follows:

Section 1, *Introduction*, provides background and purpose of the NAS EA Framework.

Section 2, *Form and Structure*, provides an overview of the NAS EA Framework, its structure, and the influences that shape and are shaped by the architectures developed with the Framework.

Section 3, *Enterprise-Level Processes and Practices*, describes key processes and approaches relevant to enterprise-level architecture construction.

Section 4, *Project-Level Processes and Practices*, describes key processes and approaches relevant to project-level architecture construction and analysis primarily in support of systems acquisition.

Appendix A, *Templates*, presents a set of reference templates for developing architecture products, roadmaps, and financials.

[This page intentionally left blank.]

# 2 FORM AND STRUCTURE

The NAS EA Framework guides the evolution of information technology to meet the goals of the FAA and its stakeholders. Architectures built from it will assist in guiding the ATO's vision and strategic execution to attaining these goals. The NAS EA Framework, depicted in Figure 1, is a comprehensive structure for describing this vision and its attainment. It is comprised of an integrated set of architectural views and products representing various perspectives of time, breadth, and detail. Enterprise architecture "snapshots" (All-View (AV), Operational View (OV), Systems View (SV) , and Technical View (TV)) are interwoven with other planning views (Executive View (XV) and Enterprise Financial View (FV)) and other engineering and planning artifacts (Enterprise-Level Requirements, Project-Level Baselines, and Planning), influencing and influenced by each other and its external environment (Joint Planning and Development Office (JPDO)-Next Generation Air Traffic System (NextGen) – External Stakeholders).

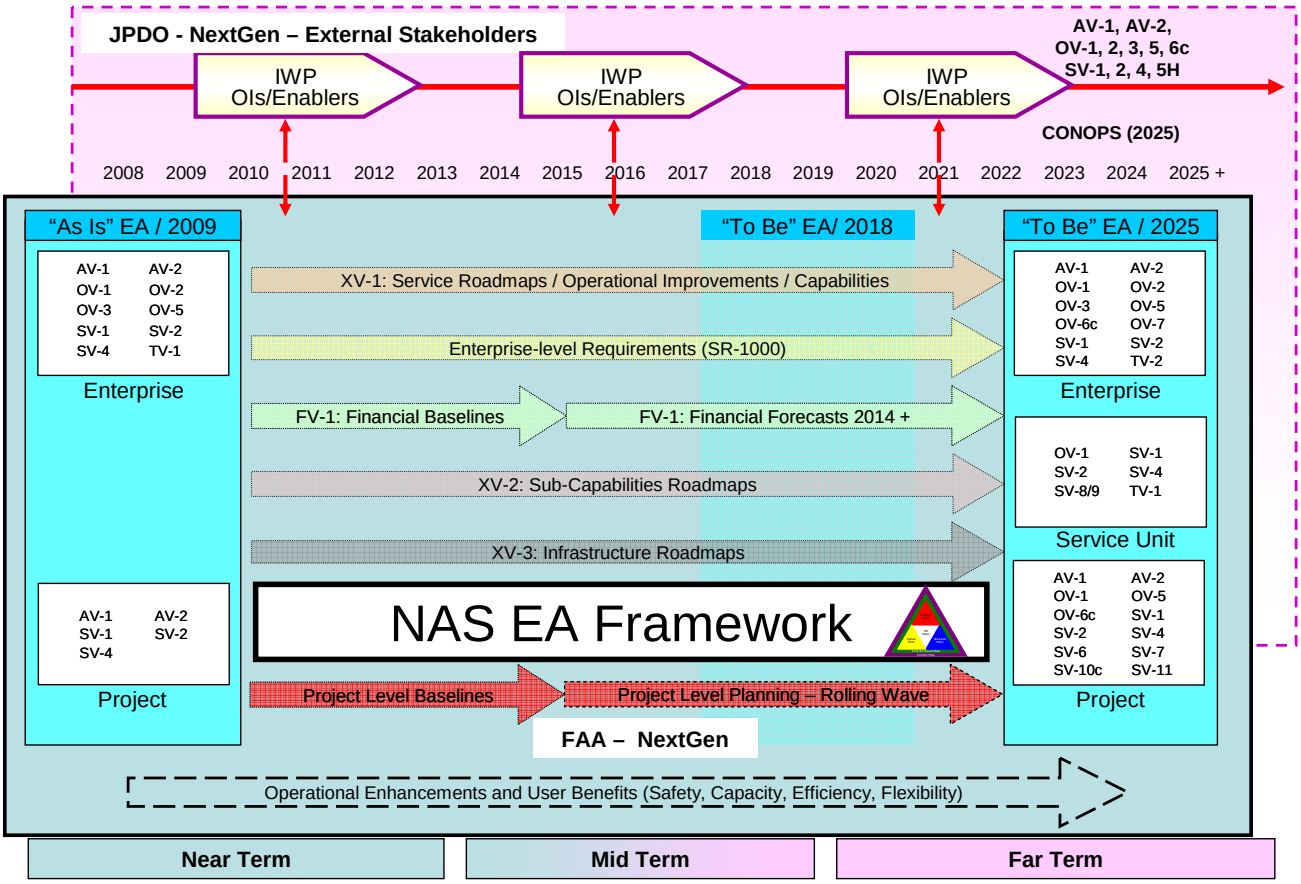



**Figure 1: National Airspace System Enterprise Architecture (NAS EA) Framework**

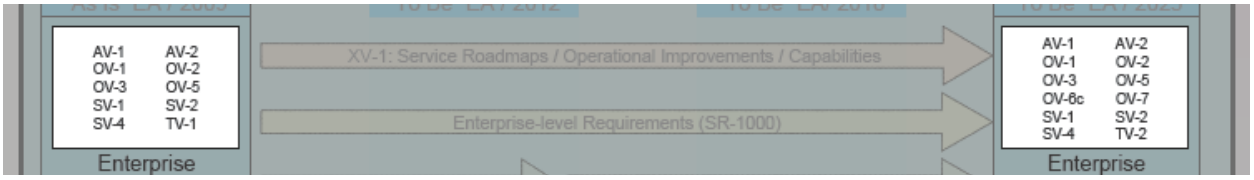
## 2.1 Levels of Enterprise Architecture

The enterprise architectures "snapshots" discussed above represent the NAS at particular points in time. Representative time frames appear in Figure 1: "As Is" 2009, "To Be" 2018 (mid-term), and "To Be" 2025 (far-term). Each representation consists of a set of integrated architectures at different degrees of breadth and abstraction. Each set contains an Enterprise-Level architecture and a varied number of related Service Unit and Project-Level architectures. Each architecture is described using the products identified within the white boxes according to level of perspective (Enterprise, Service Unit, or Project)

and type of timeframe ("As Is" or "To Be"). These product lists do not necessarily preclude developing other DoDAF-described products. Additional products may be selected and developed at the recommendation of the NAS Chief Architect for all architectures and in correspondence with Project-level architects for their architectures. Similarly, required products may be waived individually or en masse in favor of other products. Ultimately, product selection and development is based on the nature, purpose, and scope of individual architectures and the specific needs of architects, decision makers, and general stakeholder communities.

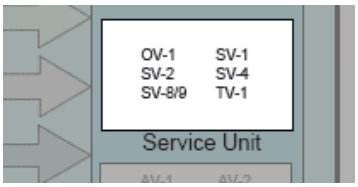
The following sections detail the extent of breadth and abstraction for each level of perspective.

### *2.1.1 Enterprise-Level*



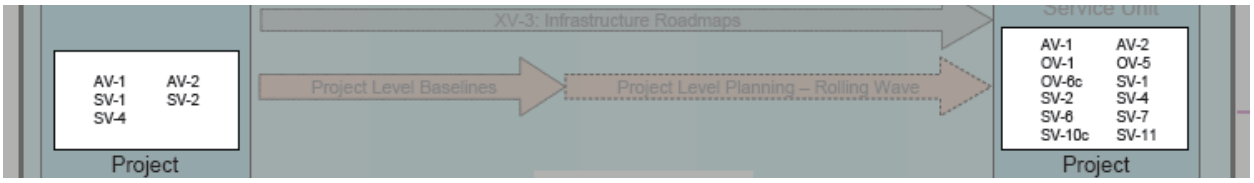

The Enterprise-Level covers the entire operational environment within the scope and influence of the FAA ATO. It has a high-level context, including the general aspects of air traffic control, traffic flow management, flight planning, airspace design and management, and infrastructure management. The context also includes, as integral external elements, pilots and aircraft, airline operations, airports, weather information providers, other federal agencies, etc. The architectural elements comprising the Enterprise-Level are equivalent in abstraction to systems or system types representing full-scale acquisition. Enterprise-Level systems, in particular, serve in their own right for establishing the context for Project-Level architectures.

### *2.1.2 Service Unit-Level*




The Service Unit-Level covers line-of-business architecture development. This level bridges the Enterprise-Level and the Project-Level, containing elements of both. A sub-set of Enterprise-Level architecture elements is identified to represent a particular line of business e.g., ATO En Route Services, Technical Operations, etc. Accordingly, relevant Project-Level architectures are integrated together, forming a comprehensive system of systems perspective across a particular service unit.

### *2.1.3 Project-Level*




Project-Level architectures typically are developed as a basis for individual system acquisition within the context of Enterprise-Level architecture elements. Enterprise-Level systems, for example, are decomposed into Project-Level system components, functions to sub-functions, etc.. These lower-level components are modeled using prescribed architecture products, at minimum, to represent an integrated Project-Level architecture description.

## 2.2 NAS EA Conceptual Data Model

Like DoDAF, the NAS EA structure primarily depends on a data model. Figure 2 depicts the NAS EA data structure as a Conceptual Data Model (CDM). Based largely on DoDAF, the CDM establishes 16 entity types and the general relationships between them. These entities and relationships, along with their required attributes as expressed in DoDAF, form the basis behind each set of architecture products/views. Note that populating the CDM and developing graphical architecture products are mutually dependent. They are a reflection of each other and are used interchangeably in various capacities to represent architectures to their developers, users, and general audiences.

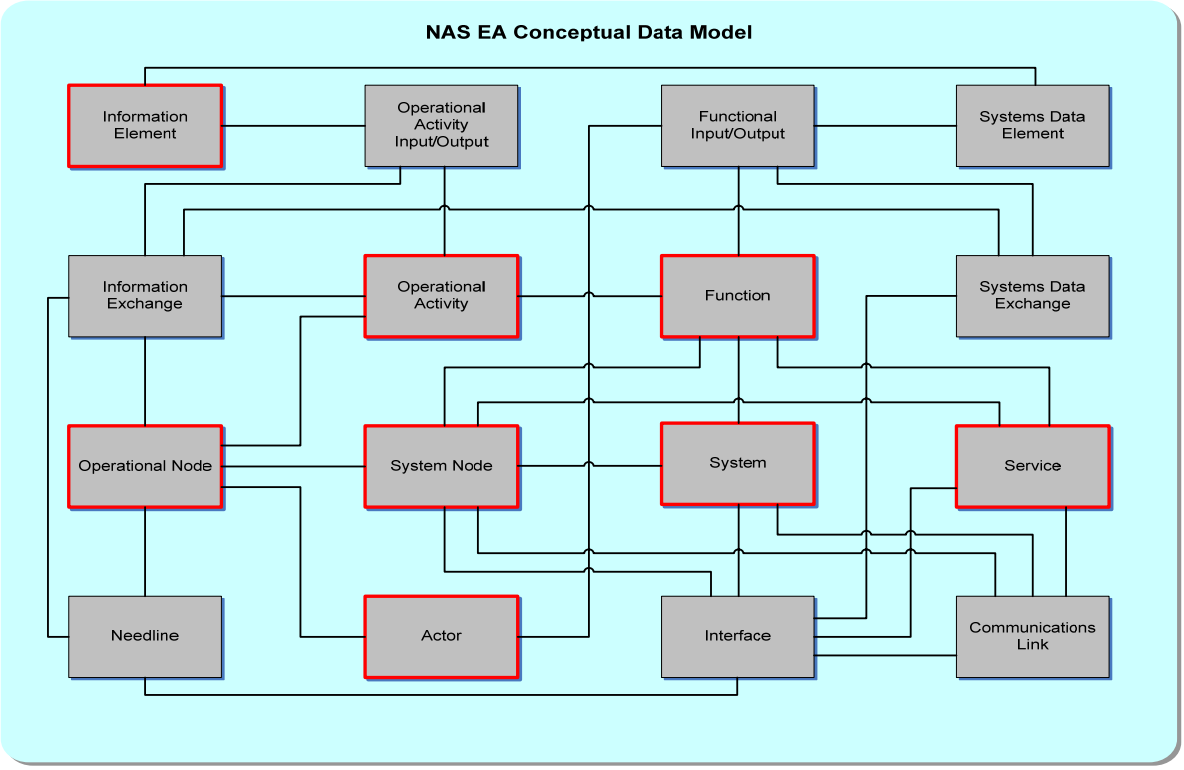



**Figure 2: NAS EA Conceptual Data Model**

The architecture entities highlighted with red-bordered boxes in Figure 2 represent the "building blocks" for architecture development, providing the foundation for architectural integration. These entities are populated with elements arranged in either hierarchical or taxonomical form and are governed by a set of fundamental properties attributable to each. Table 1 establishes these properties and the correspondence to each hierarchy/taxonomy.

**Table 1: Properties of Hierarchy/Taxonomy Elements**

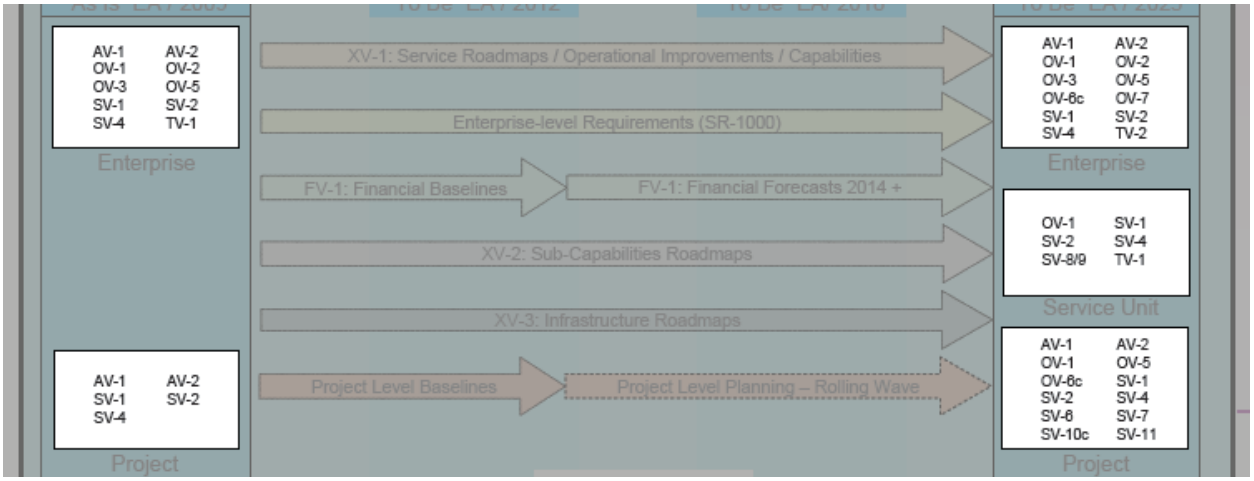
| | Hierarchy | | Taxonomy | | | | | |
|---|---|---|---|---|---|---|---|---|
| Property | Func | Op Act | Actor | Info Elem | Op Node | Svc | Sys | Sys Node |
| Non-System / Non-Solution Orientation | x | x | | x | | | | |
| Nonspecific to Environment | x | x | | x | | | | |
| Distinguishable from Other Architecture Elements | x | x | x | x | x | x | x | x |
| Non-Redundant | x | x | x | x | x | x | x | x |
| Conducive for Re-Use | x | x | x | x | x | x | x | x |

A description of Table 1 properties follows:

- **Non-System / Non-Solution Orientation** – Elements are expressed so as to avoid connoting how it is implemented or executed. For example: A function "Display Data" may lead to the assumption that data must be made available in some visible form. "Present Data" not only allows for visual availability, but also the possibility of audible availability.
- **Nonspecific to Environment** – Elements are unconstrained by specific conditions within its environment. For example: Operational activities like "Control En Route Air Traffic" and "Control Terminal Air Traffic" could be generalized to "Control Air Traffic," allowing for simpler configuration management and control. A new operational activity need not be created and managed if a new environment becomes apparent (e.g., Commercial Space). Environmental constraints can be expressed through the scope and purpose attributable to specific models or relationships to particular operational nodes, actors, and so on.
- **Distinguishable from Other Elements** – Elements remain distinct from other elements. This may be difficult when, for example, an operational node and a system node have the same name. However, the assignment to its particular taxonomy and the context of the product(s) in which it appears should suffice to make the distinction readily apparent.
- **Non-Redundant** – Elements should be attributed to one and only one particular parent element or classification. For example: System "Airport Surveillance Radar – Model 11" is attributed to the class *Surveillance* and should not also be attributed to the class "W*eather"* elsewhere in the systems taxonomy.
- **Conducive for Re-Use** – This is the obverse to the "Nonspecific to Environment" property, allowing elements to be unconstrained and available for use in any number of relevant models.

The non-highlighted entities in Figure 2 represent the information and data used for transaction between the "building blocks" at varying degrees ranging from the general need of exchange (Needline or Interface) to the actual information and data exchanged (Information and System Data Elements).

## 2.3 Architecture Products




The architecture products of the NAS EA Framework are also based largely on DoDAF. The white boxes in Figure 1 list the products required for "As Is" and "To Be" architectures at each of the three levels. For example, an "As Is" Project-Level architecture is expected to be represented, at a minimum, by AV-2, SV-1, SV-2, and SV-4 products[1]. The product lists within each box does not preclude the development of other DoDAF products for any particular architecture. The architect may deem other products as necessary or desired.

Many architecture products (e.g., OV-5, OV-7, SV-4, et al) necessitate the use of some kind of standardized methodology or technique, such as the Unified Modeling Language (UML), Integrated Definition (IDEF), Entity-Relationship (ER) diagrams, etc. However, architectural integration and consistency is not necessarily achieved by mandating particular methodologies for development of particular products. Integration and consistency is achieved through a common set of architecture elements, attributed and related in accordance with the NAS EA Conceptual Data Model (Figure 2) above. These common elements are demonstrable using a multitude of standard methods. Therefore, the NAS EA Framework does not prescribe any particular methodology for product development, leaving such selection to the developing architect.

DoDAF is a tailorable Framework, evidenced by the selection and application of required products, as discussed above. Product form, in terms of relevant elements and attributes are tailorable as well. The following list details the choices, adaptations, and modifications made against DoDAF for multiple products for the NAS EA, in general.

- *OV-2:* Primarily an Enterprise-Level Product. Operational Nodes are defined as locations where actors/performers reside and conduct operational activities.
- *OV*-3: Primarily an Enterprise-Level Product. This product is modified to form a hybrid with SV-6 elements and attributes.
- *OV*-7: Strictly an Enterprise-Level Product. This product is used to form the logical basis for Project-Level SV-11 Physical Schemas.

---

[1] AV-1 is required for "As Is" Project-Level architectures built for a Service Life Extension Program (SLEP) or a Technical Refresh. Otherwise, AV-1 focuses on "To Be" architecture.

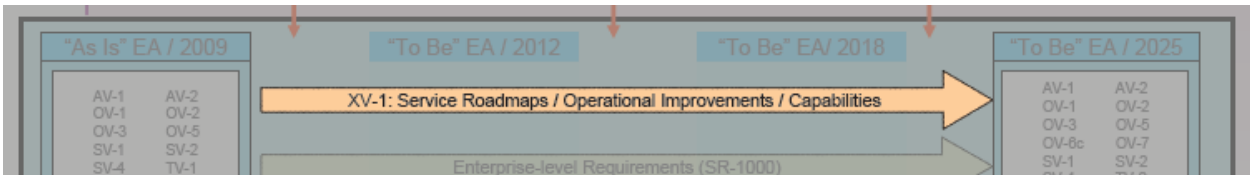
- *SV*-4: The Enterprise-Level SV-4 is represented as a "taxonomic functional hierarchy" rather than a set of data flow diagrams. A taxonomic functional hierarchy shows a decomposition of functions depicted in parent-child structure and is typically used where tasks are concurrent but dependent. This is particularly useful in capability-based procurement in which it is necessary to model the functions that are associated with particular capability. For Service Unit and Project-Level architectures, SV-4 is represented by Data Flow Diagrams.

Two additional views with subordinate products are added: Enterprise Financial Views (FV) and Executive Views (XV). The following sections describe them in detail.

### *2.3.1 Executive Views*

The Executive Views (XV) contain programmatic schedule information on the NAS including planning roadmaps and graphical schedules of initiatives. Three products are defined within the XV: Service Roadmap; Sub-Capabilities Roadmap and Infrastructure Roadmap.

#### *2.3.1.1 XV-1 Service Roadmap*




**Product Definition**. The Service Roadmap (XV-1) is the 15-year view of planned service improvement and sustainment initiatives. It consists of a structured graphical roadmap along with a set of assumptions and a list of key decisions.

**Product Purpose**. The XV-1 is a high-level view to guide, inform, and focus deliberations on NAS Capabilities. It shows the relationships among various capability elements, such as when a certain capability will replace another.

**Relationship to Other Views.** The XV-1 is closely linked with the Sub-Capability Roadmap (XV-2). The XV-2 provides the dependencies of XV-3 systems to XV-1 Capabilities via sub-capability linkages representing the planned functionalities of implementation programs.

**Product Detailed Description**. The XV-1 contains graphical and textual information that depicts how NAS Capabilities are expected to evolve over time, expressing anticipated benefits to be achieved through NAS evolution and implementation. Figure 3 depicts a sample Service Roadmap.

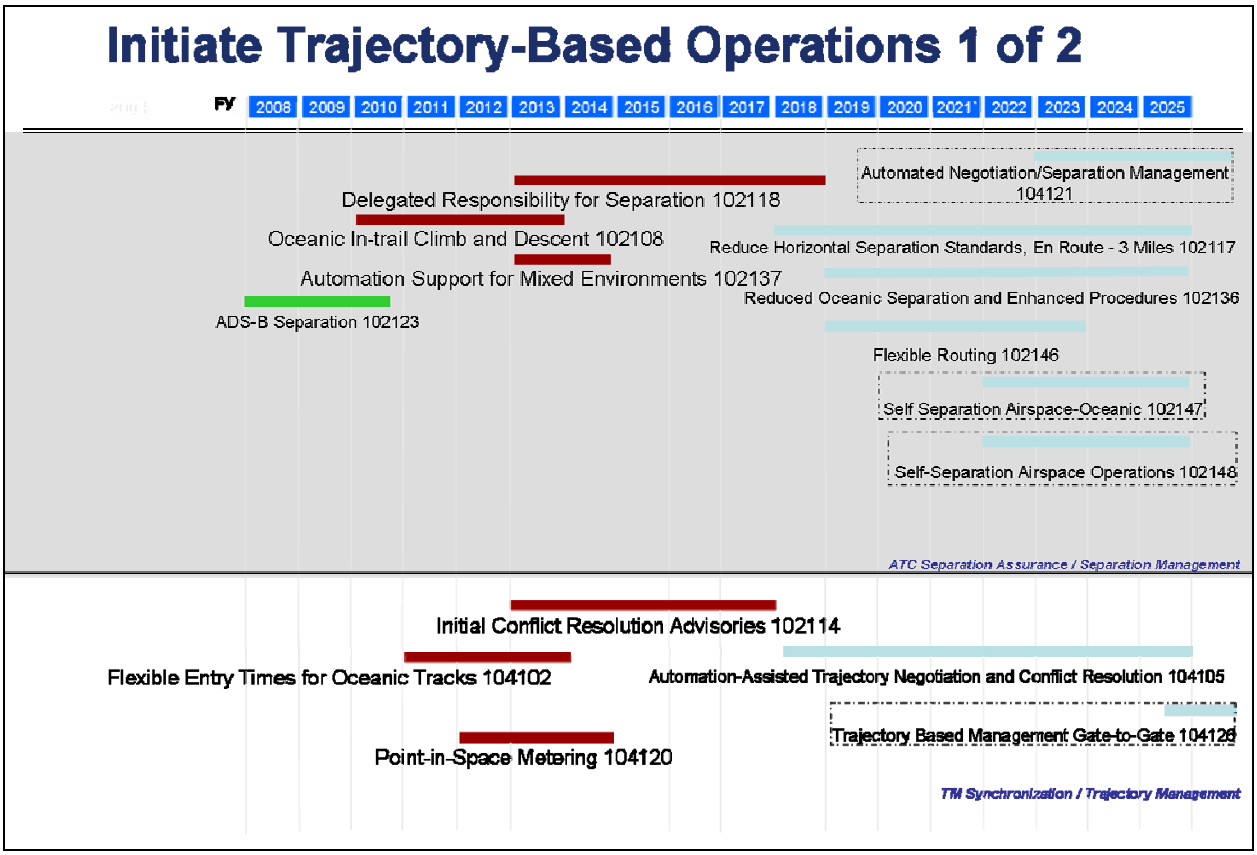



**Figure 3: Sample Service Roadmap for Initiate Trajectory-Based Operations**

The XV-1 consists of the following components:

**Service Roadmap Assumptions.** Service Roadmap Assumptions capture the basis for deciding to improve, enhance, or sustain any of the NAS service capabilities. Each assumption is based on the strategies, performance targets, and initiatives that must be accomplished to achieve ATO strategic goals and objectives. Each assumption contains two textual components: the header and the assumptions. Details regarding this component will appear in future versions of this document.

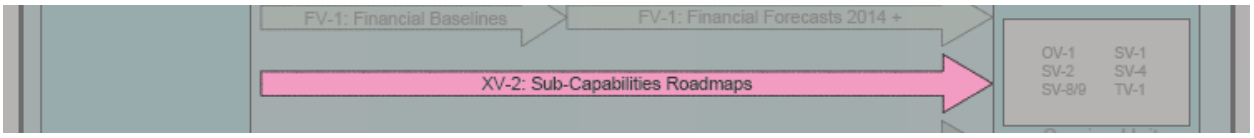
**Service Roadmap.** The Service Roadmap graphically represents the service elements over a period of time showing how they relate to each capability within each NAS service and the solution set for each initiative.

**Data Elements.** Table 2 lists the XV-1 data elements and their attributes.

**Table 2: Service Roadmap (XV-1) Data Elements**

| Data Element | Attributes | Explanation / Example Value |
|---|---|---|
| Roadmap | Title | Initiate Trajectory-Based Operations; Increase Arrivals/Departures at High Density Airports; Increase Flexibility in the Terminal Environment; Improve Collaborative Air Traffic Environment; Reduce Weather Impact; Increase Safety, Security, and Environmental Performance; Transform Facilities |
| | Timeline | Fifteen-year planning period (e.g., 2010, 2011, …, 2025) |
| | Version Number | Baseline version number of all roadmaps |
| | Date | Date of baseline for all roadmaps |
| Capability | Name | Title of Capability (aka: Operational Improvement) |
| | Identifier | Identifier of Capability |
| | Description | Summary explanation of Capability |
| | Planned Time Frame | Near-Term; Mid-Term; Far-Term |
| | Initial Operational Capability Date Range | Date range in calendar years where benefits of capability can initially be claimed |
| | Swim Lane | • ATC-Separation Assurance / Separation Management<br>• TM-Synchronization / Trajectory Management<br>• TM-Strategic Flow / Flow Contingency Management (Strategic Flow)<br>• Airspace Management / Capacity Management (Airspace)<br>• ATC-Advisory & Flight Planning, Emergency and Alerting, Infrastructure-Information Management / Flight and State Data Management |

#### *2.3.1.2 XV-2 Sub-Capability Roadmap*




**Product Definition.** The XV-2 provides the dependencies of XV-3 systems to XV-1 Capabilities via sub-capability linkages.

**Product Purpose.** The XV-2 is a high-level view to guide, inform, and focus deliberations on NAS functionalities, the systems that execute them, and delivery of expected benefits.

**Relationships to Other Views.** This product is closely linked with the Service Roadmap (XV-1) and the Infrastructure Roadmap (XV-3). The XV-1 depicts the relationships among various capability elements, such as when a certain capability will enhance another. The XV-3 depicts the programmatic and schedule relationships between the infrastructure elements.

**Product Detailed Description**. The XV-2 depicts graphical and textual information of the planned implementation and evolution of Sub-Capabilities in concert with their associated XV-1 Capabilities and XV-3 Systems. Sub-Capabilities represent functionalities planned for program implementation. These functional Sub-Capabilities form the linkages that complete the relationship between the XV-3 Systems that will execute them and the XV-1 Capabilities that express the expected benefits to which they contribute. Figure 4 depicts a sample Sub-Capability Roadmap.

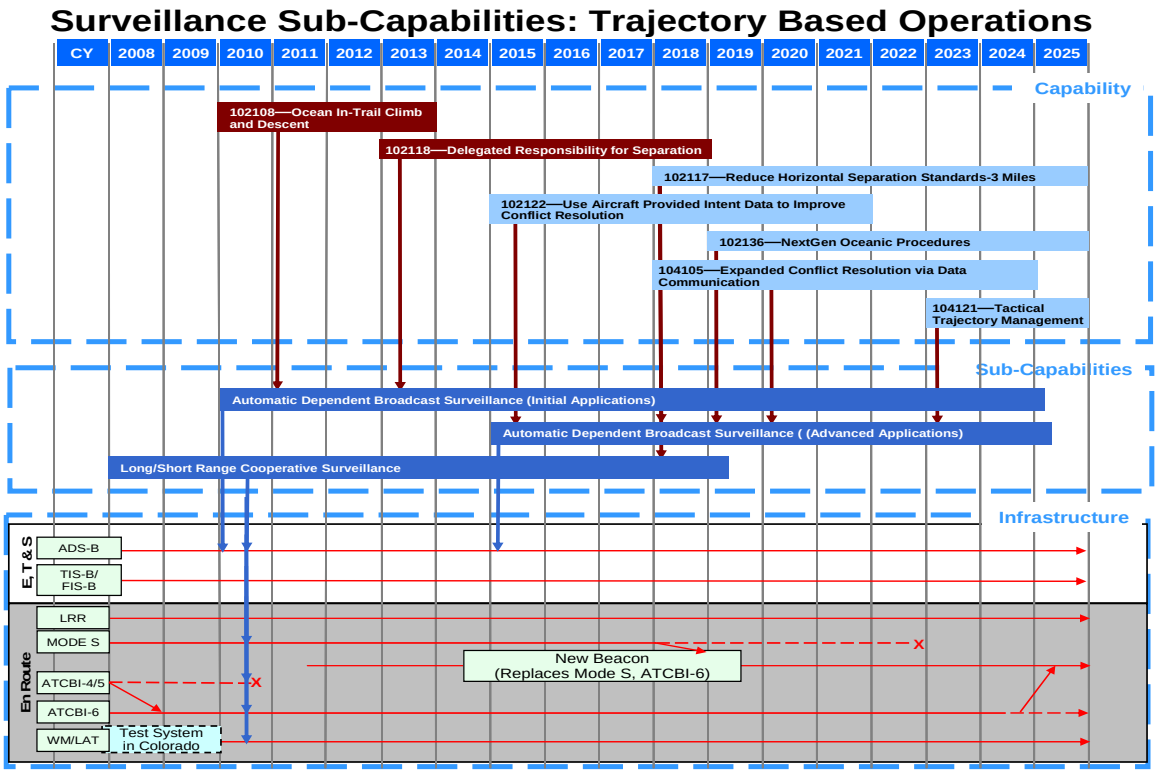


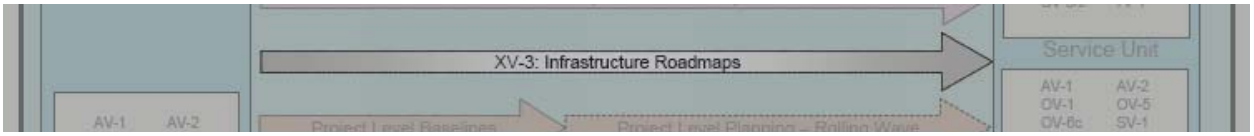

**Figure 4: Sample Roadmap for Surveillance Sub-Capabilities in Trajectory-Based Operations**

**Data Elements.** Table 3 lists the XV-2 data elements and their attributes.

**Table 3: Sub-Capability Roadmap (XV-2) Data Elements**

| Data Element | Attributes | Explanation / Example Value |
|---|---|---|
| Roadmap | Title | Initiate Trajectory-Based Operations; Increase Arrivals/Departures at High Density Airports; Increase Flexibility in the Terminal Environment; Improve Collaborative Air Traffic Environment; Reduce Weather Impact; Increase Safety, Security, and Environmental Performance; Transform Facilities |
| | Timeline | Fifteen-year planning period (e.g., 2010, 2011, …, 2025) |
| | Version Number | Baseline version number of all roadmaps |
| | Date | Date of baseline for all roadmaps |
| Sub-Capability | Name | Title of Sub-Capability |
| | Description | Summary explanation of Sub-Capability |
| | Date Range | Date range in calendar years where benefits of Sub-Capability can initially be claimed |
| | Capability Name | Name of associated Capability |
| | Infrastructure Roadmap Title | Name of Associated Infrastructure Roadmap |
| | System Name | Name of associated System |
| | Supporting Activity Name | Name of associated Supporting Activity |
| | Decision Point | Identifier of associated Decision Point |

#### *2.3.1.3 XV-3 Infrastructure Roadmap*




**Product Definition.** The Infrastructure Roadmap (XV-3) is the 15-year view of planned infrastructure improvement and sustainment initiatives. It contains programmatic and schedule information that defines the enabling infrastructure (i.e., people, systems, facilities, and support activities) for services delivery to the aviation community and other aviation services providers. It consists of a structured graphical roadmap along with a set of assumptions and a list of key decisions.

**Product Purpose.** The XV-3 is a high-level view to guide, inform, and focus deliberations on the NAS infrastructure. It shows the relationships among various infrastructure elements, such as when a certain system will replace another. This roadmap, combined with the Funding Profile (FV-1), facilitates analysis of the cost and schedule tradeoffs that exist in the budgeting and planning cycle.

**Relationship to Other Views.** The XV-3 is closely linked with the Funding Profile (FV-1). The XV-3 contains programmatic and schedule relationships between infrastructure elements, while the FV-1 contains the Facilities and Equipment (F&E) financial elements associated with those infrastructure elements. The infrastructure programmatic and schedule information in XV-3 flows to the System Views (SV-*).

**Detailed Description.** The XV-3 contains graphical and textual information depicting the infrastructure roadmap initiative's schedule. The XV-3, through convergence and modernization of enabling components, shows how to achieve the goal of optimizing the NAS. The XV-3, when combined with the other executive products, defines how the services and systems within the NAS are expected to evolve over time. This product can be used as an architecture evolution project plan or transition plan. Figure 5 depicts a sample Infrastructure Roadmap.

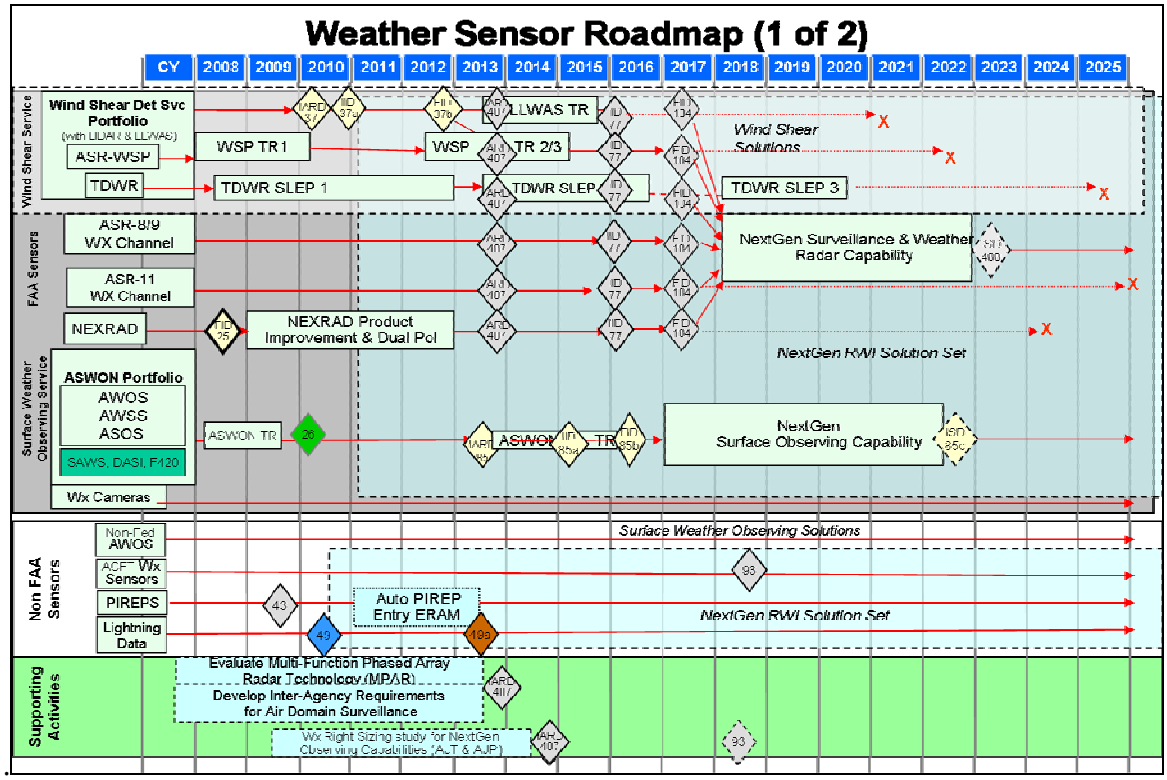



**Figure 5: Sample Infrastructure Roadmap for Weather**

The XV-3 consists of the following components:

**Infrastructure Roadmap Assumptions.** Infrastructure Roadmap Assumptions capture the basis for deciding to improve or sustain the NAS infrastructure. Each assumption is based on the strategies, performance targets, and initiatives that must be accomplished to achieve ATO strategic goals and objectives.

**Infrastructure Roadmap.** The Infrastructure Roadmap graphically represents the infrastructure elements over time and the relationships that exist between them.

**Infrastructure Roadmap Decisions.** The Infrastructure Roadmap Decisions component lists key decision points that document the FAA's approval of a particular improvement/sustainment initiative; identifies an investment decision that must precede implementation of an improvement initiative; or indicates that research and/or analysis must be conducted before an investment decision or solution implementation.

**Data Elements.** Table 4 lists the XV-3 data elements and their attributes.

**Table 4: Infrastructure Roadmap (XV-3) Data Elements**

| Data Element | Attributes | Explanation / Example Value |
|---|---|---|
| Roadmap | Title | Air-Ground, Aircraft, Airports, Airspace and Procedures, Automation, Communications, Enterprise Services, Facilities, Human Systems Integration, Navigation, Safety, Security, Surveillance, Weather |
| | Timeline | Fifteen-year planning period (e.g., 2010, 2011, …, 2025) |
| | Version Number | Baseline version number of all roadmaps |
| | Date | Date of baseline for all roadmaps |
| Assumption | Identifier | Identifier of the assumption |
| | Description | Textual description of the assumption |
| | Source Roadmap | The Roadmap in which the assumption is associated |
| | Related Decision Point | Decision Point Identifier from any Roadmap in which the assumption is associated |
| System | Name | Name of the system |
| | Swim Lane | Logical organization of related Systems |
| | Connector Line | Are used to depict predecessor/successor relationships between systems |
| | Date Range | Date range in calendar years of System lifecycle from In-Service Decision to End of Service |
| | Supporting Activity Name | Name of associated Supporting Activity |
| | Note | Notes may be attached to roadmaps to provide clarity |
| Supporting Activity | Name | Name of Supporting Activity: Prototype, Demonstration, RE&D Activity, Standards Development |
| | Date Range | Date range in calendar years of Supporting Activity effort |
| | Swim Lane | Logical organization of related Supporting Activities |
| | Technology Readiness Level | TRL Level 1, 2, 3, 4, 5, 6, 7, 8, or 9 |
| | Decision Point Identifier | Identifier of associated Decision Point |
| Decision Point | Identifier | Unique identifier for decision point |
| | Name | Title of decision point |
| | State | Condition of being for decision point: Active; Complete; Replaced; Deleted |
| | Planning/Placeholder DP | Yes/No indicator of decision point with a rough order of magnitude estimate of Target CY Date |
| | High Priority | Yes/No indicator of decision point with significant technical dependents or influences |
| | Description | Summary of key decision point characteristics and qualities |
| | Target CY Date | Calendar year date expected for decision to be made |
| | Target FY Date | Calculated Fiscal Year date corresponding to Target CY Date |
| | Actual Date | Date decision was made |
| | Related Assumption | Contains ID's of assumptions that impact decision |
| | Decision Type | • Concept and Requirements Definition Readiness Decision (CRDR)<br>• Investment Analysis Readiness Decision (IARD)<br>• Initial Investment Decision (IID)<br>• Final Investment Decision (FID)<br>• Baseline Change Decision (BCD)<br>• In-Service Decision (ISD)<br>• Executive Level<br>• Strategy<br>• Policy |
| | Impacts | Answers the question: What are the impacts if the decision is not made? |
| | Impacts NextGen | |
| | Required Activities | Key tasks to be performed prerequisite to decision being made |
| | Systems Affected | List of systems that are effected by this decision |
| | Legacy Systems Affected | List of legacy systems that are effected by this decision |
| | Affected Roadmap | Roadmaps that the decision appears on |
| | Approving Authority | Governing body or individual responsible to make decision: Enterprise Architecture Board; Executive Council; Joint Resource Council; NextGen Management Board; Service Unit Vice President |
| | Decision Activity Lead | FAA Organization responsible for the conduct of activities required for decision |
| | Supporting Organizations | List of FAA organizations having a role in the conduct of activities required for decision |
| | Status | The status of the activity with respect to the target date |
| | Status Indicator | An acknowledgement of the probability of decision occurring at or before Target CY Date: Green (Good); Yellow (At-Risk); Red (Breached) |
| | Primary Roadmap | Title of Roadmap that originates decision point |

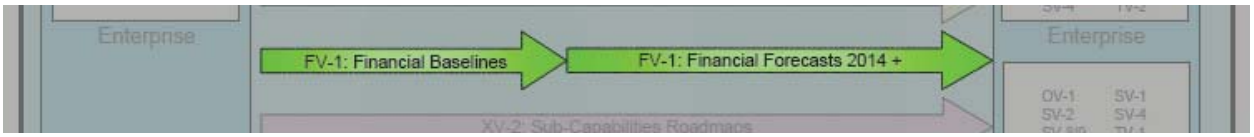
**Table 4: Infrastructure Roadmap (XV-3) Data Elements**

| Data Element | Attributes | Explanation / Example Value |
|---|---|---|
| | Related Roadmaps | List of Roadmaps influencing or influenced by decision point |
| | Related Decision Points | List of decision point identifiers and names influencing or influenced by subject decision point |
| | Supporting Activity | Name of associated Supporting Activity |
| | Decision Point Identifier | Identifier of associated Decision Point |

### *2.3.2 Enterprise Financial Views*

The Enterprise Financial Views contain baseline and forecast expenditures for funding initiatives identified in the FAA Capital Investment Plan and a mapping of Funding Initiatives to Infrastructure Roadmap Decision Points. The Enterprise Financial View defines one product: Funding Profile.

#### *2.3.2.1 Funding Profile (FV-1)*




**Product Definition.** The Funding Profile (FV-1) is a 15-year view of the Facilities and Equipment (F&E) funding for approved and forecasted infrastructure programs, projects, and segments. It consists of the F&E funding schedules that defines the enabling infrastructure (i.e., people, systems, facilities, and support activities) for services delivery to the aviation community and other aviation services providers.

**Product Purpose.** The FV-1 is a high-level view to guide, inform, and focus Executive Council deliberations on the NAS infrastructure. The FV-1 depicts the 15-year F&E funding schedule for the ATO to achieve its goals. This product, combined with the Infrastructure Roadmap (XV-3), facilitates analysis of the cost and schedule tradeoffs that exist in the budgeting and planning cycle.

**Relationship to Other Views.** This product is closely linked with the Infrastructure Roadmap (XV-3), which depicts the programmatic and schedule relationships between the infrastructure elements.

**Product Detailed Description.** An FV-1 is created from the ATO Enterprise perspective, and contains funding information at the Capital Investment Plan (CIP) program level. From the Integrated Product Team (IPT) perspective, it contains funding information at the project cost level. At each level of detail, it consists of a spreadsheet detailing programs, projects, and NAS segments.

The FV-1 consists of the following components:

**Profile Summary.** The Profile Summary provides a condensed overview of F&E dollar allocations relative to a particular Infrastructure Roadmap by the following expenditure type: Funded, NextGen, Forecast, and Delta.

**Profile Chart.** The Profile Chart depicts the Profile in graphical form.

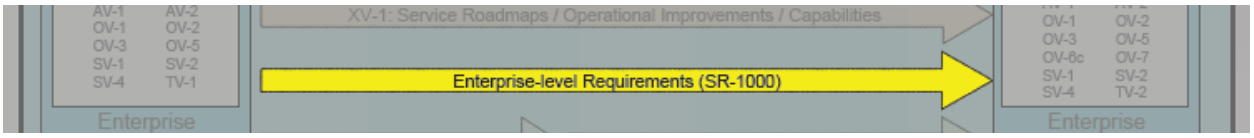
**Profile Spreadsheets.** The Profile Spreadsheets contain detailed dollar allocations for each Project identified in the CIP. It also provides a mapping of each Project to one or more Decision Point(s) defined in its associated Infrastructure Roadmap. There is one Profile Spreadsheet for each expenditure type.

**Data Elements.** Table 5 lists the FV-1 data elements and their attributes.

**Table 5: Funding Profile (FV-1) Data Elements**

| Data Element | Attributes | Explanation / Example Value |
|---|---|---|
| Profile | Title | Automation, Communications, Navigation, Surveillance, Weather, Facilities, etc. |
| | Timeline | Fifteen-year planning period (e.g., 2010, 2011, …, 2025) |
| | Version Number | Baseline version number of all roadmaps |
| | Date | Date of baseline for all roadmaps |
| Funding Summary | Type | Baseline, NextGen, Forecast, Delta |
| | Amount | Dollars allocated or forecast for each funding type per year |
| Project | Name | Title of Project from capital Investment Plan |
| | CIP Number | Identifier of Project from Capital Investment Plan, e.g., N03.1-00 |
| | Decision Point | Infrastructure Roadmap Decision Point(s) associated with Project |
| | Amount | Dollars allocated or forecast for each Project per year |
| | Type | Funded, NextGen, Forecast, Delta |
| Assumption | Identifier | Identifier of the assumption |
| | Description | Textual description of the assumption |
| | Source Initiative | The Funding Initiative in which the assumption is associated |

## 2.4 Enterprise-Level Requirements

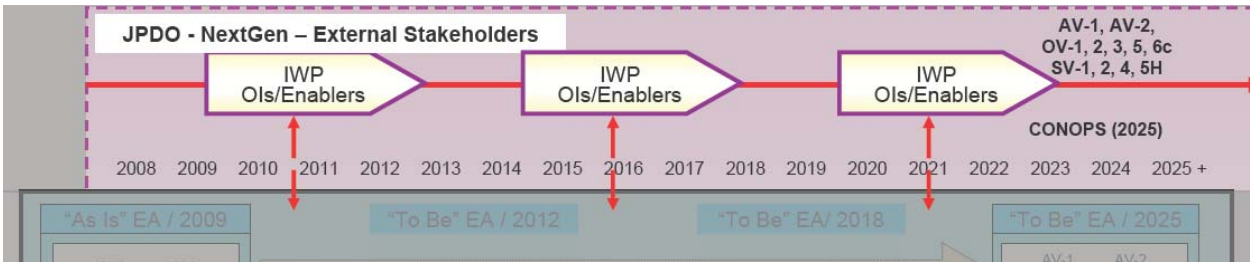



Enterprise-Level Requirements are developed within the constraints described by each Enterprise-Level architecture. The Enterprise-Level structure of these requirements mimics the architecture's functional structure. Within this structure, individual requirements are managed based upon functional content of the architectures. The current functional structure and content for each Enterprise-Level Architecture are available in the NAS Enterprise Architecture Portal, http://nasea.faa.gov.

## 2.5 Project Level Baselines and Planning

ATO program offices are responsible for managing the development and evolution of individual NAS systems within their domains. Program offices use the architecture to frame and organize how they intend to transform and implement their systems according to FAA goals.

## 2.6 JPDO-NextGen — External Stakeholders




The Joint Planning and Development Office (JPDO) is the central organization that coordinates the specialized efforts of several federal government stakeholders (Departments of Transportation, Defense, Homeland Security, and Commerce, and, the FAA, National Aeronautics and Space Administration, and the White House Office of Science and Technology Policy) in a public/private partnership to bring the Next Generation Air Transportation System (NextGen) to fruition by the year 2025. The architectural scope of this effort encompasses a broader "curb-to-curb" representation of aviation than that of the NAS' "gate-to-gate" environment, expanding into airport operations and support, FAA and non-FAA weather operations, transportation security and screening, etc. The NAS Enterprise Architecture serves as the foundation for the broader JPDO Enterprise Architecture.

[This page intentionally left blank.]

# 3 ENTERPRISE-LEVEL PROCESSES AND PRACTICES

## 3.1 Service Roadmap Development

The Integration and Implementation office within ATO's Systems Engineering and Safety division executes the Service Roadmap Development process. The output of this process is the Service Roadmaps, constructed according to section 2.3.1.1 XV-1 Service Roadmap above in concert with the NextGen Implementation Plan. Input derived from feedback resulting from the Infrastructure Roadmap development process, particularly from its Integration sessions, as described below, is considered, as well. Service Roadmaps are typically completed prior to the Infrastructure Roadmap Development process Updating phase.

## 3.2 Infrastructure Roadmap Development

To reflect the FAA's evolving system and infrastructure transition plans, the Infrastructure Roadmaps are updated annually using a four stage process: Process Evaluation, Planning, Updating, and Approving. Table 6 briefly describes the active roles within the process and their general responsibilities.

**Table 6: Roles and Responsibilities for Infrastructure Roadmap Development**

| Role | Responsibility |
|---|---|
| Investment Decision Authority | Provides final decision for Infrastructure Roadmap baseline (Joint Resources Council; ATO Executive Council) |
| Service Unit Vice President | Provides programmatic endorsement of Infrastructure Roadmaps. |
| Technology Review Board | Provides technical endorsement of Infrastructure Roadmaps (delegated responsibility by the NAS Enterprise Architecture Board). |
| NAS Chief Architect | Provides overall orchestration of Infrastructure Roadmap development activity and status reporting. |
| Solution Set Manager | Provides subject matter expertise regarding NAS Operational Improvements expressed within individual Solution Sets (ATO Integration and Implementation Office) |
| Domain Lead | Coordinates stakeholder collaboration and provides domain, system, and technical expertise for individual Infrastructure Roadmap development and integration (typically ATO Systems Engineering and Safety or Service-Unit personnel). |
| Stakeholder | Provides system and technical expertise for individual Infrastructure Roadmap development and integration (typically ATO Service-Unit and program office personnel). |
| Domain Subject Matter Expert | Provides development guidance and assistance to Lead Domain SME (typically SETA-II contract personnel under direct NAS Chief Architect authority). |

Table 7 provides a notional timeline of the process.

**Table 7: Notional Timeline for Infrastructure Roadmap Development Process**

| Phase | Month |
|---|---|
| Process Evaluation | January |
| Planning | February<br>March |
| Updating | April<br>May<br>June<br>July<br>August<br>September |
| Approving | October<br>November<br>December |

### *3.2.1 Process Evaluation*

Each process cycle begins and ends with Executive Council (EC) and Joint Resource Council (JRC) approval of the latest Infrastructure Roadmap update. Approval effectively establishes a new baseline. Shortly after cutover to the new baseline, lessons learned during its development are identified and evaluated. Evaluation results are transformed into process improvements that are implemented for subsequent development phases.

### *3.2.2 Planning*

Planning for the update phase begins after process improvement implementation resulting from the previous phase. A schedule for the remainder of the development cycle is prepared, kickoff meeting agenda items are formulated, stakeholders are identified and solicited for participation, meeting facilities are arranged, and Domain Leads and Domain Subject Matter Experts (SMEs) are identified and assigned.

### *3.2.3 Updating*

The update phase begins with the Infrastructure Roadmap kickoff meeting. The NAS Chief Architecture briefs the Domain Lead and Domain SMEs on the objectives, changes, and expectations for updates, and schedule through the Approving phase.

Each Domain Lead updates their Roadmap by coordinating with their relevant roadmap working group consisting of participating stakeholders and assigned Domain SME(s). With Domain SME assistance, Domain Leads schedule and conduct working group meetings to identify changes to the assumptions, existing and planned systems, and their related decision points. The team updates roadmaps iteratively and is expected to complete all updates within 8 to 10 weeks.

Over the next 10 to 12 week period a set of Roadmap integration sessions are prepared for and conducted. Participants in these sessions include The NAS Chief Architect, Domain Leads, Domain SMEs, stakeholders and Solution Set Managers. Solution Set Managers develop and maintain the Operational Improvements (OIs) within assigned Solution Sets. A Solution Set is a category consisting of a series of related OIs that can be managed as part of the FAA's NextGen Portfolio. OIs are understood as cross-domain statements expressing sets of anticipated benefits to be realized at some future date. The integration sessions seek to show how groups of individual Roadmap elements (Systems/Programs) are assembled to satisfy individual OIs. These assemblies are collected in a set of Integration Worksheets, organized and populated according to time frame (Mid-Term and Far-Term), then by Solution Set, and finally by OI. The relationship between individual OIs and roadmap Systems/Programs is made by identifying Sub-Capabilities. Sub-Capabilities are expressions of functionality expected to be implemented and executed on a System/Program basis. Together, all Sub-Capabilities linked to particular OIs provide cross-domain portraits for the planned realization of NAS benefits.

Over The final 2 to 4 weeks, clean-up and packaging of the Roadmaps, Decision Points, and Integration Worksheets occurs in preparation for the Approving phase.

### *3.2.4 Approving*

The Approving phase is the last step in the development process. The NAS Chief Architect provides successive briefs to the Enterprise Architecture Board (EAB)/Technology Review Board (TRB) and each ATO Service-Unit Vice President to obtain endorsement of the roadmaps. Finally, the NAS Chief Architect briefs the EC and JRC to obtain roadmap approval. A new baseline is then established and the entire process is repeated, starting with the Planning phase for the next update.

### *3.2.5 Status Reporting*

Status Reporting occurs in parallel with roadmap development and continues throughout the year. Once a new baseline is established, the progress made toward achieving the Decision Points for that year is continuously tracked. Table 8 details the criteria used to report Decision Point status.

**Table 8: Decision Point Criteria**

| | **AMS Decision Types**<br>• Concept and Requirements Definition Readiness Decision<br>• Investment Analysis Readiness Decision<br>• Initial Investment Decision<br>• Final Investment Decision<br>• Baseline Change Decision<br>• In-Service Decision | **Other Decision Types**<br>• Executive Decision<br>• FAA Policy<br>• FAA Strategy<br>• Others |
|---|---|---|
| Red | • Unsatisfactory progress is being made towards target date<br>• Target date has or is projected to be missed | • Unsatisfactory progress is being made towards target date<br>• Target date has or is projected to be missed |
| Yellow | • Progress is being made; however the target date is at risk of being missed as reported through the JRC Readiness Review Minutes | • Progress is being made; however the target date is at risk of being missed based on information provided by the Lead Organization |
| Green | • Satisfactory progress is being made towards reaching the target date as reported through the JRC Readiness Review Minutes | • Satisfactory progress is being made towards reaching the target date based on information provided by the Lead Organization |

The status of all Decision Points planned for the year, as well as those carried over from the previous year, if any, is reported regularly to the JRC. The report includes the following data for each Decision Point:

- Location on roadmap
- Identification (Identifier, Name, Domain, Related Domains, Type, CY Target Date, Owner)
- Description
- Status detail, including state (Active, Completed, Deleted, or Replaced)
- Impacts, if any, expected to occur if decision is not achieved

The NAS Chief Architect briefs this information to the JRC each calendar quarter.

## 3.3 Sub-Capability Roadmap Development

Lasting 4 to 6 weeks, the Sub-Capability Roadmap Development process extends from the Updating phase of Infrastructure Roadmap development integration sessions. As described above, these sessions result in a set of Roadmap Integration Worksheets. Shortly after final clean-up and packaging of all Infrastructure Roadmap material, the Integration Worksheets are interpreted into graphical form in accordance with section **Error! Reference source not found.** Sub-Capability Roadmap (XV-2) and Figure 4 of this document.

## 3.4 Funding Profile Development

The Funding Profile development is another process that extends from the Infrastructure Roadmap development Updating phase. The Domain Lead for each roadmap leads stakeholders and Domain SMEs in an exercise comparing the current Capital Investment Plan (CIP) against the draft roadmaps completed during the Updating phase. Variations in CIP funding streams from the funding needed to satisfy the implementation plans, as depicted in the roadmaps, are captured as Deltas. The team then develops an updated Forecast for funding based again upon their roadmap. Finally, the current CIP outlays, the Deltas and the Forecast are assembled in the format expressed in section 2.3.2.1 Funding Profile (FV-1) of this document. The complete results of the exercise are used as input into the FAA's annual budget request.

## 3.5 Enterprise-Level Architecture Development

Enterprise-Level Architecture development follows a process that is similar to the Infrastructure Roadmap Planning, Updating, and Approving phases. Table 9 briefly describes the active roles in the process and their general responsibilities.

**Table 9: Roles and Responsibilities for Enterprise-Level Architecture Development**

| Role | Responsibility |
|---|---|
| Enterprise Architecture Board | Acts as authority for the establishment of Enterprise-Level architecture baselines. |
| Technology Review Board | Provides technical review and endorsement of Enterprise-Level architectures. |
| NAS Chief Architect | Provides overall orchestration of Enterprise-Level architecture development. |
| Architecture Developers | Provides technical expertise for the development and revision of Enterprise-Level architecture views. |
| Stakeholder | Provides technical review and comment of Enterprise-Level architecture views. |

### *3.5.1 Planning*

Planning begins immediately following architecture baseline approval. Planning is limited in duration (2 to 4 weeks) and includes definition of scope, schedule development, stakeholder identification, and resource allocation. Scheduling focuses on the tasks described in following phases and includes periods for architecture review and comment, comment adjudication, architecture revision/development, and architecture approval. Resource allocation focuses on identifying and organizing resources, particularly architecture development personnel, for architecture product development in the Updating Phase.

### *3.5.2 Updating*

The Updating phase starts with a review and comment period open to FAA stakeholders. Comments are collected throughout the comment period and allocated to appropriate architecture view areas. Once the comment period ends, the architects examine and resolve comments, including clarifying comments with their originators and documenting disposition. Based on the comments as well as additional NAS Chief Architect direction, architects may revise current views or build new architecture views.

Checkpoint meetings with the developers and the NAS Chief Architect occur periodically to report progress. A final checkpoint ends the Updating process establishing the final draft architecture.

### *3.5.3 Approving*

The Approving process follows the Updating process. The final draft architecture is presented to the TRB followed by the EAB. Additional comments may be submitted from either board for disposition. Once all comments are resolved, the EAB establishes a new Enterprise-Level architecture baseline[2], the content of which represents the entire NAS and serves as the definitive context for integration with Project-Level architectures.

[2] The "To Be" *target* or end-state architecture is approved by the JRC after EAB approval. In this case, JRC approval establishes the baseline. All other Enterprise-Level architectures, "As Is" or interim "To Be", have baselines established with EAB approval.

[This page intentionally left blank.]

# 4 PROJECT-LEVEL PROCESSES AND PRACTICES

## 4.1 Project-Level Architecture Development

This section focuses on Project-Level architecture development supporting the FAA Acquisition Management System (AMS) decision-making process, from initial concept development to the start of solution implementation.

### *4.1.1 Architecture Products and Acquisition Phase*

Project-Level architecture development primarily occurs during the earliest phases of the system lifecycle: Concept and Requirements Definition (CRD); Initial Investment Analysis (IIA); and Final Investment Analysis (FIA). Developers must produce a specific set of architecture products during each phase. Table 10 lists these products. (See DoDAF for more detailed descriptions.)

**Table 10: Architecture Product Development per AMS Phase**

| Phase | Product ID | Product Name |
|---|---|---|
| All | AV-1 | Overview and Summary Information |
| | AV-2 | Integrated Dictionary |
| CRD | OV-1 | High-Level Operational Concept Graphic |
| | OV-5 | Operational Activity Model |
| | OV-6c | Operational Event-Trace Description |
| | SV-4 | Systems Functionality Description |
| IIA | SV-1 | Systems Interface Description |
| | SV-2 | Systems Communications Description |
| FIA | SV-6 | Systems Data Exchange Matrix |
| | SV-7 | Systems Performance Parameters Matrix |
| | SV-10c | Systems Event-Trace Description |
| | SV-11 | Physical Schema |

Note that DoDAF contains additional products not listed in the table above. These other products may be prescribed for project-level development by the NAS Chief Architect in addition to or as replacements for the products listed in Table 10. Otherwise, all remaining DoDAF products may be considered optional based upon program need at the discretion of Project-Level Architects.

Figure 6 illustrates the required architecture products for projects moving through the AMS acquisition phases.

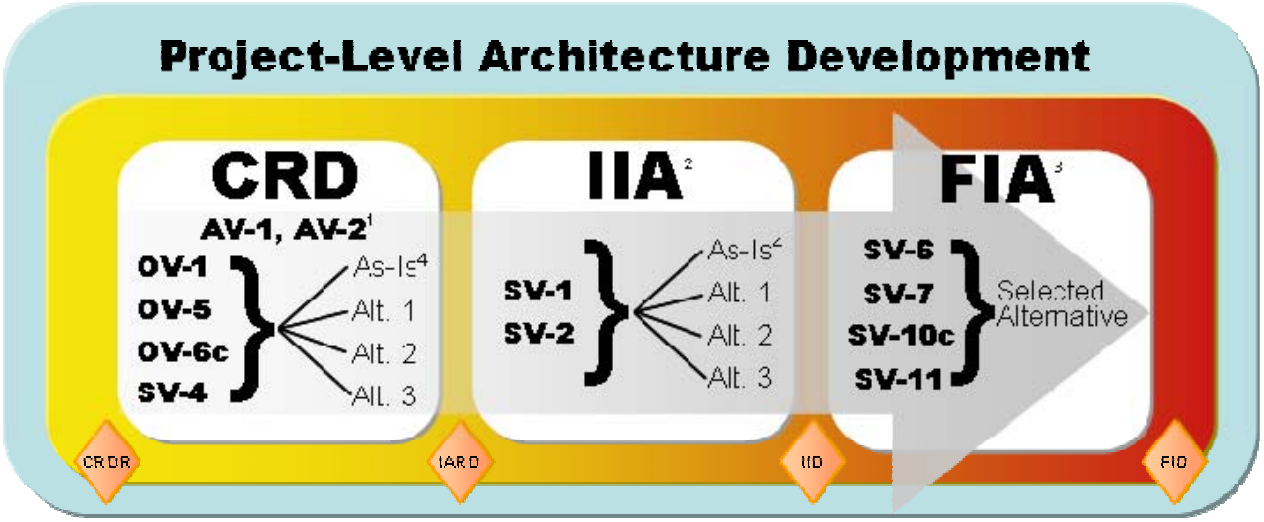



**Figure 6: Required Architecture Products per Acquisition Phase**

The following are key development constraints referenced in the above diagram by the numbers in superscript:

1. An AV-1 and AV-2 are required for each phase. The AV-1 is relevant to all the products for all alternative solutions. An AV-2 may be segmented by alternative solutions.
2. In addition to the products explicitly identified for the phase, IIA includes continued maturation of all products from CRD, as necessary.
3. In addition to the products explicitly identified for the phase, FIA includes continued maturation of all products from CRD and IIA, as necessary, for the "As Is" and the selected alternative architectures.
4. "As Is" architecture products are limited to AV-2, SV-1, SV-2, and SV-4. An "As Is" AV-1 is required only when a project represents a "legacy" system effort (e.g., Baseline Change, SLEP, Technical Refresh, etc.).

Per the AMS, Project-Level architecture development begins with Concept and Requirements Definition Readiness Decision (CRDR) and the CRD phase. In this phase, a range of alternatives is identified, and concept(s) of use and preliminary requirements are developed. Each solution within the range of alternatives, including the current, or "As Is," is represented, at minimum, by an OV-1, OV-5, OV-6c, and SV-4,. The OV-1's summarize the concept(s) of use, and the OV-5, OV-6c, and SV-4's play critical roles in organizing and identifying preliminary requirements. All architecture elements (e.g., Operational Activities, System Functions, Data Elements, etc.) used in these products are defined, and relationships between them identified, in the AV-2 per the NAS EA CDM (Figure 2, above). The AV-1 summarizes the entire architecture effort.

The Investment Analysis Readiness Decision (IARD) ends the CRD phase and initiates the IIA phase, in which comprehensive alternative analyses and lifecycle cost estimates are produced. For each alternative, an SV-1 and SV-2 are developed and integrated with the products developed during CRD. Architecture elements modeled in SV-1 and SV-2 are defined in the AV-2 and additional element-to-element relationships are captured. With system components and allocations to functions understood

architecturally, additional requirements can be defined. Comparing solution architectures contribute directly to comprehensive alternative analyses and trade studies. Further, cost figures applied against various architecture elements form the foundation for lifecycle cost estimating. Finally, the AV-1 is updated in preparation for Initial Investment Decision (IID).

Providing a down-select from alternative solutions to one preferred solution, IID ends the IIA phase and initiates the Final Investment Analysis (FIA) phase. Development of SV-6, SV-7, SV-10c, and SV-11 provides data exchange, system interface, functional sequencing, and physical data structures contribute to final requirements definition in preparation for Solution Implementation. Definitions of SV-11 data elements and their relationships to other architecture elements are populated in the AV-2. Finally, the AV-1 is again updated to reflect the efforts undertaken during FIA in preparation for Final Investment Decision (FID) and entry into the Solution Implementation phase.

### *4.1.2 Development Process*

The process for Project-Level architecture development for acquisition applies to any AMS phase. Table 11 briefly describes the active roles in the process and their general responsibilities.

**Table 11: Roles and Responsibilities for Project-Level Architecture Development**

| Role | Responsibility |
|---|---|
| Investment Decision Authority | Provides decisions leading up to investment in NAS acquisitions (Joint Resources Council; ATO Executive Council) |
| Project-Level Architect | Coordinates and develops project-level architecture development (En Route and Oceanic Services; Terminal Services; System Operations Services; Technical Operations Services) |
| NAS Chief Architect | Approval authority for all NAS Project-Level architectures (NextGen and Operations Planning Services) |
| Enterprise Architect | Provides development guidance and assistance to Program Office Project-Level architecture efforts (Typically SETA-II contract personnel under direct NAS Chief Architect authority) |
| Technology Review Board | Provides review and compliance assessment of all NAS Project-Level architectures (delegated responsibility by the NAS Enterprise Architecture Board) |

Figure 7 illustrates the process with a notional timeline that starts and ends with an AMS decision.

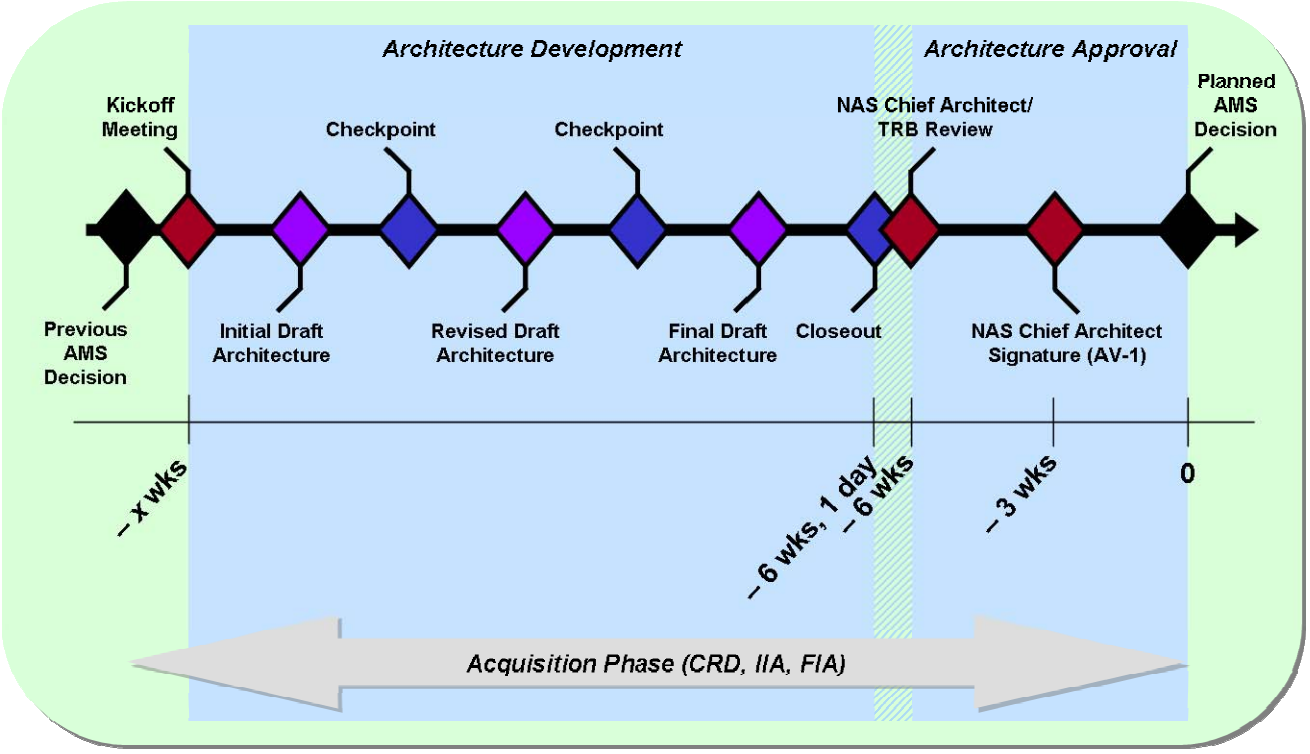



**Figure 7: Project-Level Architecture Development and JRC Decision – Notional Timeline**

The process has two stages: Architecture Development and Architecture Approval.

**Architecture Development.** This stage begins immediately following a CRDR Decision, an IARD, or an IID and is executed within the subsequent CRD, IIA, and FIA phases, respectively. The Enterprise Architect coordinates an architecture development kickoff meeting with the NAS Chief Architect and Project-Level Architect. Prior to the kickoff, a NAS Enterprise Architect coordinates a recommended architecture product set for development and a proposed high-level development schedule.

The typical product set recommendation includes the architecture products in Figure 6 above per AMS phase. However, the NAS Chief Architect may provide product set tailoring. Through tailoring, products from the prescribed list may be removed, and other relevant products added depending on a particular project's needs and constraints.

The proposed schedule is developed from the planned AMS decision date, working backwards to the kickoff meeting. Typically, architecture development occurs immediately following the meeting and continues up to 6 weeks prior to the planned AMS decision. Within this timeframe, a set of delivery and checkpoint milestones is identified for architecture development. A development milestone, shown as a purple diamond in Figure 7, indicates when the architecture is ready for review. In most cases, there should be at least three development milestones per AMS phase; an initial draft; at least one revised draft; and final draft architecture. The expected architectural content produced at each milestone is also identified.

A checkpoint milestone (blue diamond) gives the Enterprise Architect with the opportunity to consult with the NAS Chief Architect about product status and discuss issues and actions regarding architecture development. Also, the checkpoint indicate to the Project-Level Architect when the review results (in the

form of a comment/resolution matrix) may be expected from the Enterprise Architect against the developed products. The proposed schedule, including the anticipated architectural content for each draft and their respective dates, is negotiated between the Enterprise Architect and Project-Level Architect prior to the kickoff meeting.

As discussed earlier, the Kickoff meeting initiates Project-Level architecture development. The NAS Chief Architect, Enterprise Architect, and Project-Level Architect review the recommended product set and proposed schedule. They raise and deliberate issues not resolved during kickoff coordination as well as discuss any needs or constraints not previously coordinated. Finally, the NAS Chief Architect and Project-Level Architect agree on the schedule and product set, effectively initiating architecture development.

From this point, Project-Level architecture development progresses through Closeout. Within this timeframe, draft architecture products are developed and released to the Enterprise Architect for review and comment, per the agreed upon development and checkpoint milestones. The Enterprise Architect submits review comments against the products to the Project-Level Architect for resolution. The Enterprise Architect may also collect change recommendations from the Project-Level Architect against the Enterprise-Level architecture to be vetted during its updating cycle, as described in Section 3.5.2 above. After multiple iterations of product development, review, and comment, the Project-Level Architect provides a Final Draft of the architecture. The Final Draft includes all agreed-upon architecture products for the Final Draft milestone unless otherwise waived by the NAS Chief Architect. The Enterprise Architect then prepares for Closeout.

The Closeout milestone is a special checkpoint because it effectively ends architecture development for the phase and initiates the Architecture Approval stage. The Enterprise Architect prepares for Closeout by finalizing a descriptive state of the architecture. This includes the comments concerning the Final Draft architecture, highlighting the proposed resolutions, as negotiated with the Program-Level Architect, regarding all remaining open comments. Closeout becomes formal once the Enterprise Architecture delivers the architecture to the NAS Chief Architect for review and approval in the Architecture Approval stage.

**Architecture Approval.** The Architecture Approval stage begins immediately after the Closeout checkpoint with delivery of the entire architecture package to the NAS Chief Architect and the Technology Review Board (TRB). Ideally, the review occurs over a three week period. The TRB provides its recommendations and observations to the NAS Chief Architect. The NAS Chief Architect signs the AV-1 no later than three weeks before AMS decision. The AV-1 becomes the official document representing the entire architecture for the phase, indicating to the Joint Resources Council (or the Executive Council) that all AMS requirements relevant to project-level enterprise architecture development have been met.

## 4.2 Project-Level Architecture Analysis

Architecture owners (program offices for project-level architectures) perform architecture analysis in concert with an Enterprise Architect. Analysis is conducted to ensure consistent architecture development and to satisfactorily address the extent to which a particular architecture fits FAA goals and plans. Figure 8 depicts a reference structure for architecture analysis.

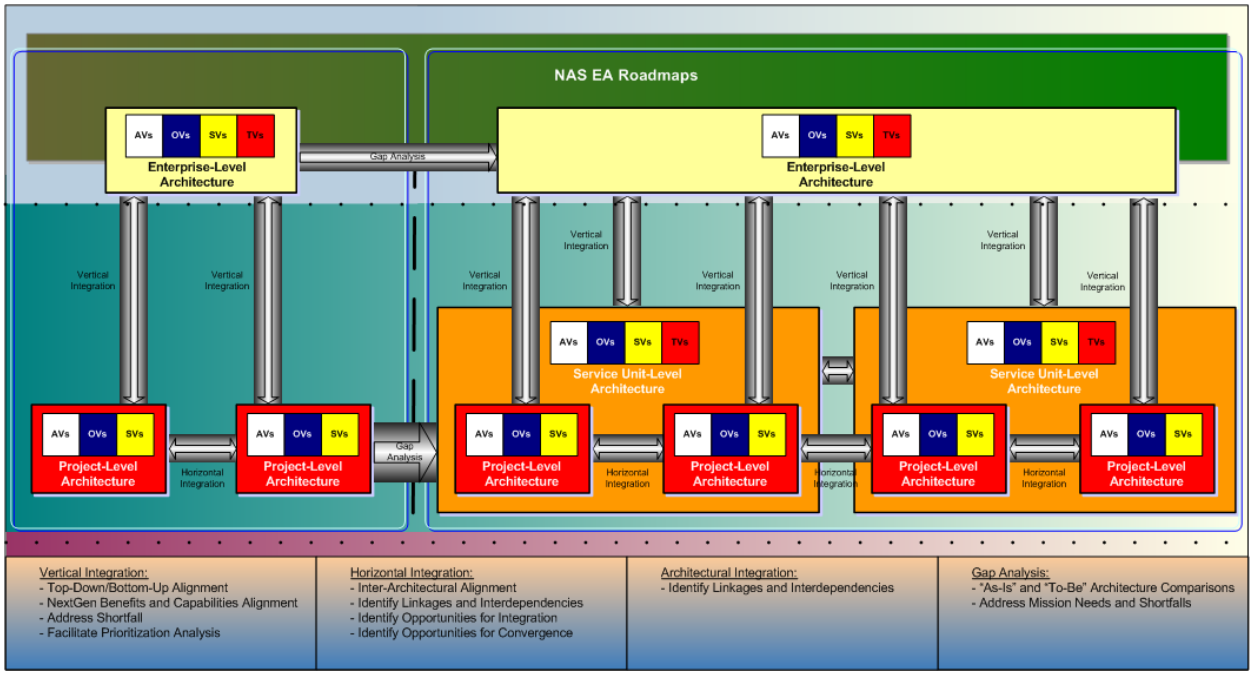



**Figure 8: Architecture Integration and Analysis**

Four principal areas govern architecture analysis: Vertical Integration, Horizontal Integration, Architectural Integration, and Gap Analysis.

**Vertical Integration** ensures that a Project-Level architecture accommodates a top-down/bottom-up alignment with architecture elements defined at the Enterprise-Level; supports its "parent" in providing NextGen benefits; aligns with NextGen Solution Set operational capabilities/improvements, addressing corresponding shortfalls; and facilitates prioritization analysis. Per Table 12, constituent Project-Level architecture elements are required to be vertically integrated with Enterprise-Level elements.

**Table 12: Required Constituent Relationships for Project-Level Vertical Integration**

| Enterprise-Level Element | Project-Level (Service Unit-Level) Element |
|---|---|
| Operational Activity | Operational Activity |
| Information Element | Information Element; Systems Data Element |
| System | System |
| Operational Node; Systems Node | Systems Node |
| System Function | System Function |

**Horizontal Integration** identifies linkages and inter-dependencies of, helps identify opportunities for integration and convergence with, and ensures consistency across interrelated Project-Level or Service Unit-Level architectures. Verification is conducted for any architecture's external element that is known to originate in another corresponding architecture. For example, in one Project-Level Architecture, an interface is identified with an external system component defined in another Project-Level architecture. Horizontal integration ensures that this interface is treated equivalently, using the same system names that serve as the source/sink for that interface in both architectures.

**Architectural Integration** identifies the linkages and inter-dependencies between architecture elements within the same architecture, ensuring consistency across interrelated Project-Level architecture products. Architectural integration is achieved by architecture element-to-element relationships. Table 13 indicates

the minimal set of element-to-element relationships required to achieve Project-Level architectural integration.

**Table 13: Minimally Required Relationships for Project-Level Architectural Integration**

| Acquisition Phase | Architecture Relationship | Cardinality |
|---|---|---|
| CRD | Operational Activity to Operational Activity Input | One-to-Many / One-to-Many |
| | Operational Activity to Operational Activity Output | One-to-Many / One-to-One |
| | Operational Activity to Actor | One-to-Many / One-to-Many |
| | System Function to System Function Input | One-to-Many / One-to-Many |
| | System Function to System Function Output | One-to-Many / One-to-One |
| | System Function to Operational Activity | One-to-Many / One-to-Many |
| IIA | System to Interface | One-to-Many / Two |
| | Systems Node to Interface | One-to-Many / Two |
| | System to System Function | One-to-Many / One-to-Many |
| | System to Communications Link | One-to-Many / One-to-Two |
| | Systems Node to Communications Link | One-to-Many / One-to-Two |
| FIA | Interface to System Data Exchange | One-to-Many / One-to-One |
| | System Data Exchange to System Function Input/Output | One-to-Many / One-to-One |
| | System Function Input/Output to System Data Element | One-to-One / One-to-Many |

Table 13 may be interpreted using the following example for "System to Interface": For a complete Project-Level architecture in the IIA Phase, any identified System must be related at least one Interface (one-to-many), and any interface must be related to an originating and a consuming system (two).

**Gap Analysis** is an element-by-element comparison of "As Is" and "To Be" architectures. It verifies that "To Be" architectures address the mission need and shortfalls recognized from its corresponding "As Is" Architecture.

Documentation of the results of the analyses described above resides in the Project-Level architecture's AV-1 Overview and Summary Information document per the template in Appendix A of this Framework.

The remainder of this section further details the analytical scope of each product per AMS phase.

### *4.2.1 Concept and Requirements Definition: Develop Initial Architecture Alternatives*

An initial set of three "To Be" Project-Level architecture alternatives is developed as a "range of architecture alternatives" during the AMS CRD process phase, in addition to an "As Is" architecture, if one does not already exist. Each architecture alternative must be sufficiently defined to distinguish it from the other alternatives, while at the same time meeting the same set of high-level requirements. Analysis is performed for vertical integration to ensure NextGen benefits and new capabilities are addressed; shortfalls are addressed; and the Project-Level architecture products align with the Enterprise-Level architecture.

Horizontal integration analysis is performed to identify linkages and interdependencies between interrelated sets of architecture products. Initial opportunities for integration and/or convergence of operations and systems are identified. An initial gap analysis is conducted on the "As Is" and "To Be" Project-Level architecture products to ensure mission need shortfalls and reuseability are addressed. Distinctions between "As Is" and "To Be" architecture product features and characteristics are identified.

Project-Level architecture Development Process findings are documented in Section 6 of the AV-1 Overview and Summary Information product.

Utilizing specific Project-Level architecture products developed during CRD, Table 14 identifies analysis items to be covered.

**Table 14: Architecture Analysis for CRD**

| NAS EA PRODUCT | ANALYSIS ITEMS |
|---|---|
| **(AV-1)**<br><br>**Overview and Summary Information** | Vertical Integration Analysis:<br><br>• Ensure that the Project-Level architecture products map to NextGen operational capabilities/improvements<br>• Ensure that Project-Level architecture products are either a sub-set or whole component of the Enterprise-Level architecture products<br>• Ensure that the Project-Level architecture products supports the scheduled execution of Enterprise-Level architecture products<br><br>Horizontal Integration Analysis:<br><br>• Ensure that the required Project-Level architecture products developed for interrelated Project-Level architectures are consistent to enable integration<br>• Assess the interrelated Project-Level architecture products schedules to ensure that interdependent operations, systems infrastructure, and technical standards are aligned<br><br>Architectural Integration Analysis:<br><br>• Assess that the shortfalls, mappings to NextGen OIs, and new services offerings, and the strategic directions are consistent across Project-Level architecture products<br><br>Gap Analysis:<br><br>• Assess that the “To Be” Project-Level architecture products address shortfalls, mappings to NextGen operational capabilities/improvements and new services offerings, and the strategic directions overcome deficiencies in the “As Is” Project-Level architecture products<br>• Assess that the scheduled migration to the “To Be” Project-Level architecture products can be realized given the complexity of the “As Is” Project-Level architecture products |
| **(AV-2)**<br><br>**Integrated Dictionary** | Vertical Integration Analysis:<br><br>• Ensure the Project-Level architecture Integrated Dictionary contains subset information of the Enterprise-Level architecture products<br><br>Horizontal Integration Analysis:<br><br>• Assess that the Project-Level architecture Integrated Dictionary contains the same glossary and component architecture relationships between interrelated Project-Level architecture<br><br>Architectural Integration Analysis:<br><br>• Ensure that Project-Level architecture elements contain required relationships with other elements per the NAS EA Conceptual Data Model (Figure 2 of this document). |

**Table 14: Architecture Analysis for CRD**

| NAS EA PRODUCT | ANALYSIS ITEMS |
|---|---|
| **(OV-1)**<br><br>**High-Level Operational Concept Graphic** | Vertical Integration Analysis:<br><br>• Ensure that the operational concept supports Enterprise-Level architecture products operational concepts<br><br>Horizontal Integration Analysis:<br><br>• Assess that the Project-Level architecture products operational concepts integrate with interrelated Project-Level architecture products operational concepts at the point(s) of interface<br><br>Architectural Integration Analysis:<br><br>• Ensure that all graphical icons and representations correspond with Project-Level architecture elements<br><br>Gap Analysis:<br><br>• Assess and provide the operational concept distinctions between the “As Is” and “To Be” Project-Level architecture products<br>• Assess that the operational concept distinctions address the operational shortfalls |
| **(OV-5)**<br><br>**Operational Activity Model** | Vertical Integration Analysis:<br><br>• Show operational activities performed are consistent as subcomponents of the Enterprise-Level architecture operational activities<br>• Ensure actors mapped to operational activities are consistent with those in the Enterprise-Level architecture<br><br>Horizontal Integration Analysis:<br><br>• Assess that the operational activities performed at operational nodes are consistent across interrelated Project-Level architecture products if the same activities are applicable<br><br>Architectural Integration Analysis:<br><br>• The following relationships are defined in accordance with Table 13:<br>  • Operational Activity to Operational Activity Input<br>  • Operational Activity to Operational Activity Output<br>  • Operational Activity to Actor<br><br>Gap Analysis:<br><br>• Assess and provide the operational activities distinctions between the “As Is” and “To Be” Project-Level architecture products<br>• Assess that the new “To Be” Project-Level architecture operational activities address the activity shortfalls |

**Table 14: Architecture Analysis for CRD**

| NAS EA PRODUCT | ANALYSIS ITEMS |
|---|---|
| **(OV-6c)**<br><br>**Operational Event-Trace Description** | Horizontal Integration Analysis:<br><br>• Assess that the operational threads and scenarios sequences and information exchanges of the interrelated Project-Level architecture products are consistent with operational nodes and information elements utilized in other architecture products<br><br>Gap Analysis:<br><br>• Assess and provide the operational scenarios distinctions between the "As Is" and "To Be" Project-Level architecture products<br>• Assess that the "To Be" Project-Level architecture products operational scenarios address the operational shortfalls |
| **(SV-4)**<br><br>**Systems Functionality Description** | Vertical Integration Analysis:<br><br>• Assess that systems functionality performed by systems in the Project-Level architecture products are consistent as subcomponents of Enterprise-Level functionality<br>• Ensure that data exchanges between systems functionality in the Project-Level architecture products are consistent with Enterprise-Level data exchanges<br><br>Horizontal Integration Analysis:<br><br>• Assess that systems functionality performed by systems in the Project-Level architecture products are consistent across interrelated Project-Level architecture products when that same functionality is used<br><br>Architectural Integration Analysis:<br><br>• The following relationships are defined in accordance with Table 13:<br>  • System Function to System Function Input<br>  • System Function to System Function Output<br>  • System Function to Operational Activity<br><br>Gap Analysis:<br><br>• Assess and provide the systems functional elements distinctions between the "As Is" and "To Be" Project-Level architecture products<br>• Assess that the new "To Be" Project-Level architecture products systems functional elements address the shortfalls |

### *4.2.2 Initial Investment Analysis: Develop Final Architecture Alternatives*

A final set of three "To Be" Project-Level architecture alternatives are developed during the AMS IIA process phase as a continuation of the initial architecture alternatives. Additional EA products are developed and continual products are further matured. Each architecture alternative must be sufficiently defined to distinguish the architectures for alternatives analysis.

Analysis is performed for vertical integration to ensure NextGen benefits and new capabilities are addressed; shortfalls are addressed; the proposed architectures are still a priority for achieving NextGen benefits; and the Project-Level architecture products align with Enterprise-Level architecture products.

Horizontal integration analysis is performed to identify linkages and interdependencies between interrelated architectures and to maintain interface consistency. Opportunities for integration and/or convergence of operations and systems are identified. A gap analysis is conducted on the “As Is” and “To Be” Project-Level architecture products to ensure that Mission Need shortfalls and component reuseability are addressed. Further distinctions between “As Is” and “To Be” features and characteristics are identified. Project-Level architecture Development Process findings are documented in section 6 of the AV-1 Overview and Summary Information product.

Table 15 identifies analysis items to be covered for specific EA products developed during IIA, in addition to those identified in Table 14.

**Table 15: Architecture Analysis for IIA**

| NAS EA PRODUCT | ANALYSIS ITEMS |
|---|---|
| **(SV-1)**<br><br>**Systems Interface Description** | Vertical Integration Analysis:<br><br>• Assess that the systems utilized in the Project-Level architecture products are consistent as subcomponents of the Enterprise-Level systems<br>• Ensure that system interfaces established between systems in the Project-Level architecture products are consistent with the Enterprise-Level<br>• Ensure that the systems utilized in the Project-Level architecture products are consistent with NAS EA XV-3 Infrastructure Roadmaps<br><br>Horizontal Integration Analysis:<br><br>• Assess that the systems utilized in the Project-Level architecture products are consistent across interrelated Project-Level architecture products and have a common interface reference<br><br>Architectural Integration Analysis:<br><br>• The following relationships are defined in accordance with Table 13:<br>  • System to System Interface<br>  • System Node to Interface<br>  • System to System Function<br><br>Gap Analysis:<br><br>• Assess and provide the system elements distinctions between the “As Is” and “To Be” Project-Level architecture products<br>• Assess that the “To Be” Project-Level architecture products systems elements are needed and address the shortfalls |
| **(SV-2)**<br><br>**Systems Communications Description** | Vertical Integration Analysis:<br><br>• Assess that the communications services and communications systems (communications systems, links, and networks) means of exchanging data between systems interfaces for the Project-Level architecture products are consistent with the Enterprise-Level<br>• Ensure that the communications services and communications systems utilized in the Project-Level architecture products are consistent with NAS EA XV-3 Infrastructure Roadmaps<br><br>Horizontal Integration Analysis:<br><br>• Assess that the communications services and communications systems (communications systems, links, and networks) means of exchanging data between systems interfaces are consistent across interrelated Project-Level architecture products<br>• Ensure that interrelated Project-Level architecture products have an instantiated communication paths that link the architectures products for communications<br><br>Architectural Integration Analysis:<br><br>• The following relationships are defined in accordance with Table 13:<br>  • System to Communications Link<br>  • Systems Node to Communications Link<br><br>Gap Analysis:<br><br>• Assess and provide the communications services and communications systems (communications systems, links, and networks) distinctions between the “As Is” and “To Be” Project-Level architecture products<br>• Assess that the “To Be” Project-Level architecture products communications services and communications systems address the shortfalls |

### *4.2.3 Final Investment Analysis: Complete Preferred Alternative*

Analysis of the "preferred" alternative is executed during the AMS FIA phase. The "preferred" Project-Level architecture has been down-selected by the JRC, but is only partially completed after IIA. Completion of the "preferred" architecture is achieved by developing additional EA products and updating mature products previously developed.

Analysis is performed for vertical integration to ensure NextGen benefits and new capabilities are addressed; shortfalls are addressed; the "preferred" architecture is a priority for achieving NextGen benefits; and the EA products are in alignment with Enterprise-Level architectures.

Horizontal integration analysis is performed to identify linkages and inter-dependencies between interrelated architectures. Final opportunities for integration and/or convergence of operations and systems are identified. A Final gap analysis is conducted on the "As Is" and "To Be" Architectures products to ensure mission need shortfalls and reuseability are addressed. Distinctions between "As Is" and "To Be" Architectures features and characteristics are identified. Analysis findings are documented in the Section 6 of the AV-1 Overview and Summary architecture product.

Table 16 identifies analysis items to be covered for specific EA products developed during FIA, in addition to those identified in Table 14 and Table 15.

**Table 16: Architecture Analysis for FIA**

| NAS EA PRODUCT | ANALYSIS ITEMS |
|---|---|
| **(SV-6)**<br><br>**Systems Data Exchange Matrix** | Vertical Integration Analysis:<br><br>• Assess that the data exchanged between systems in the Project-Level architecture products are consistent with the Enterprise-Level<br><br>Horizontal Integration Analysis:<br><br>• Ensure that system data exchanged between interrelated Project-Level architecture products system elements are consistent<br><br>Architectural Integration Analysis:<br><br>• The following relationships are defined in accordance with Table 13:<br>  • Interface to System Data Exchange<br>  • System Data Exchange to System Function Input/Output<br><br>Gap Analysis:<br><br>• Assess that the "To Be" Project-Level architecture products system data exchanges are an extension of the "As Is" Project-Level architecture products system data exchanges |

**Table 16: Architecture Analysis for FIA**

| NAS EA PRODUCT | ANALYSIS ITEMS |
|---|---|
| **(SV-7)**<br><br>**Systems Performance Parameters Matrix** | Horizontal Integration Analysis:<br><br>• Assess that the Project-Level architecture products systems functional characteristics for processing and interface characteristics for system data exchanges meet the specified criteria (required performance)<br>• Assess that the Project-Level architecture products systems required performance meet the requirements for the architecture required operational performance<br>• Ensure that the performance requirements for interrelated Project-Level architecture products are sufficient to meet overall performance requirements<br><br>Gap Analysis:<br><br>• Assess distinctions between “As Is” and “To Be” Project-Level architecture products performance requirements and ensure that the “To Be” Project-Level architecture products are sufficient to meet future planning needs |
| **(SV-10c)**<br><br>**Systems Event-Trace Description** | Horizontal Integration Analysis:<br><br>• Assess that the system threads, scenarios sequences, and data exchanges of interrelated Project-Level architecture products are consistent with other architecture products<br><br>Gap Analysis:<br><br>• Assess the distinctions between the “As Is” and “To Be” Project-Level architecture products systems, systems functions, and human roles and ensure that the new event trace descriptions address the mission need shortfalls |
| **(SV-11)**<br><br>**Physical Schema** | Vertical Integration Analysis:<br><br>• Ensure that Project-Level architecture products data elements align with Enterprise-Level OV-7 Logical Data Model<br><br>Horizontal Integration Analysis:<br><br>• Ensure that data elements are consistently represented across Project-Level architecture products<br><br>Architectural Integration Analysis:<br><br>• The following relationships are defined in accordance with Table 13:<br>  • System Function Input/Output to System Data Element<br><br>Gap Analysis:<br><br>• Assess the distinction between “As Is” Project-Level architecture data elements data elements and “To Be” Project-Level architecture data elements and application to relevant “To Be” functionality. |

# A APPENDIX: TEMPLATES

Templates are provided to guide the development of Project-Level Architectures, Roadmaps, and Enterprise Financials. These templates are listed below:

PROJECT-LEVEL ARCHITECTURE PRODUCTS

- AV-1 Overview and Summary Information
- AV-2 Integrated Dictionary
- OV-5 Operational Activity Model
- OV-6c Operational Event-Trace Description
- SV-1 Systems Interface Description
- SV-2 Systems Communications Description
- SV-4 Systems Functionality Description
- SV-6 Systems Data Exchange Matrix
- SV-7 Systems Performance Matrix
- SV-10c Systems Event-Trace Description
- SV-11 Physical Schema

ROADMAPS

- Infrastructure Roadmaps
- Sub-Capability Roadmaps

ENTERPRISE FINANCIAL

- Funding Profile

Architecture product templates are provided to guide architects in the development of products that are meaningful to other architects, decision makers, and general stakeholders. It should be noted that these templates provide a minimal starting point for product development. Products may be embellished to reflect or relate to other architectural aspects that are not necessarily shown on any individual template. This is at the discretion of the architect.

Some templates appear to suggest particular methodologies: IDEF0 for OV-5; UML for OV-6c, SV-10c, and SV-11. Architects may consider using these methodologies or opt for using other methodologies for their particular architectures. This is also at the discretion of the architect as long as what is selected meets the relational aspects of the NAS EA Conceptual Data Model depicted in section 2.2 of the main document.

Finally, templates have been provided for required Project-Level products only. This does not preclude the development of other DoDAF-described products that may provide value to the architect, decision makers, or general stakeholder communities. Project-Level architects may elect to develop other products in addition to those required where their value is deemed appropriate.

## A.1 AV-1 Overview and Summary Information

[program name] Architectures
Overview and Summary Information (AV-1)

Version [#.#]
[Month ##, 20##]

**National Airspace System Enterprise Architecture (NAS EA)**

**Air Traffic Organization, Directorate of [name]**

**(ATO-[letter])**

**Version [#.#]**

**Overview and Summary Information (AV-1)**

**For**

**[program name] Proposed Architecture**

**[Month ##, 20##]**

Approved By:_____ Date:_____
NAS Chief Architect, Operations Planning, ATO

## A.1 AV-1 Overview and Summary Information (continued)



### Version History

| VERSION | PUBLICATION DATE | PRIMARY AUTHOR(S) | DESCRIPTION OF CHANGE |
|---|---|---|---|
| 2.0 | [Month ##, 20##] | [Author Name(s)]<br>and/or<br>[Organization Name(s)] | A. [major change 1]<br>B. [major change 2]<br>X. [major change n] |
| 1.1 | [Month ##, 20##] | [Author Name(s)]<br>and/or<br>[Organization Name(s)] | DRAFT REVISION |
| 1.0 | [Month ##, 20##] | [Author Name(s)]<br>and/or<br>[Organization Name(s)] | APPROVED ORIGINAL |

## A.1 AV-1 Overview and Summary Information (continued)



### Table of Contents

HOW TO USE THIS PRODUCT GUIDE ..... III

1 ARCHITECTURE PROJECT IDENTIFICATION ..... 1-1
- 1.1 ARCHITECTURE NAME ..... 1-1
- 1.2 ARCHITECT AND DEVELOPING ORGANIZATIONS ..... 1-1
- 1.3 APPROVAL AUTHORITY ..... 1-1
- 1.4 ARCHITECTURE DEVELOPMENT SCHEDULE ..... 1-1

2 SCOPE: ARCHITECTURE VIEWS AND PRODUCTS ..... 2-1
- 2.1 VIEWS AND PRODUCTS DEVELOPED ..... 2-1
- 2.2 TIME FRAMES ADDRESSED ..... 2-2
- 2.3 ORGANIZATIONS WITHIN SCOPE ..... 2-2

3 PURPOSE AND VIEWPOINT ..... 3-1
- 3.1 PURPOSE, ANALYSIS, AND QUESTIONS TO BE ANSWERED ..... 3-1
- 3.2 VIEWPOINT ..... 3-1

4 CONTEXT ..... 4-1
- 4.1 MISSION ..... 4-1
- 4.2 GOALS, OBJECTIVES, AND VISION ..... 4-1
- 4.3 RULES, CRITERIA, AND CONVENTIONS FOLLOWED ..... 4-2
- 4.4 TASKING AND LINKAGES TO OTHER ARCHITECTURES ..... 4-2

5 ASSUMPTIONS AND CONSTRAINTS ..... 5-1

6 TOOLS AND FILE FORMATS USED ..... 6-2

7 FINDINGS ..... 7-1
- 7.1 ANALYSIS RESULTS ..... 7-1
- 7.2 RECOMMENDATIONS ..... 7-1

## A.1 AV-1 Overview and Summary Information (continued)



### HOW TO USE THIS PRODUCT GUIDE

All sections of this product guide contain an explanatory paragraph that is highlighted in *blue italicized text*. This text provides overall guidance on the information that is required for each section and may be deleted in the final version of the specific AV-1.

Many of the paragraphs in this product contain example text typical for describing various architectures. Authors are encouraged to consider the relative appropriateness of this text and tailor it as necessary. Placeholders for variable terms are highlighted by [blue bracketed text].

This product guide is intended to be flexible enough to be used to describe architectures at any level of maturity for any purpose. In the initial phases of architecture development, the AV-1 serves as a planning guide. Upon completion of the architecture, AV-1 provides summary textual information concerning the architecture.

Note that the identification of multiple architectures may be necessary when the effort has a system acquisition focus. The Concept and Requirement Development process and the Initial Investment Analysis phase of the FAA Acquisition Management System require at least four technical alternatives. One alternative is the status quo, represented by an As-Is architecture with its own AV-1. The remaining three alternatives are viable candidate solutions to identified shortfalls. Each alternative is represented by an individual To-Be architecture, all three of which may be summarized in one, all-encompassing "To-Be" AV-1. Once a candidate solution is selected for full requirements development and design maturation at Initial Investment Decision, the To-Be AV-1 is revised in deference to the selected To-Be architecture.

## A.1 AV-1 Overview and Summary Information (continued)



# 1 ARCHITECTURE PROJECT IDENTIFICATION

*This section identifies the Architecture name, the architect, and the organization developing the Architecture. It also includes assumptions and constraints, identifies the approving authority, and the completion date.*

## 1.1 ARCHITECTURE NAME

*This section identifies the name of the architecture to be addressed.*

This Overview and Summary Information (AV-1) document provides executive-level summary information for the [Architecture Name].

## 1.2 ARCHITECT AND DEVELOPING ORGANIZATIONS

*This section identifies the architect responsible for architecture development. The architect may be a program's Project Lead or designate. Name of individual is optional. Use appropriate organizational titles when referencing individuals by name.*

The architect responsible for the [Architecture Name] is the Air Traffic Organization, [directorate name] Directorate, Office of [office name], [program name] Project Lead.

## 1.3 APPROVAL AUTHORITY

The FAA Joint Resources Council (JRC) is the final approval authority of all [program name] architectures.

## 1.4 ARCHITECTURE DEVELOPMENT SCHEDULE

*The schedule provides a set of key dates of the development effort. For Project-Level architecture, this includes the start date, dates of completion for initial draft, revised drafts, and final draft architecture and applicable decision date(s). (Typically, these are FAA Acquisition Management System decisions, e.g., Investment Analysis Readiness Decision; Initial Investment Decision, Final Investment Decision, etc.)*

**[Architecture Name] Development Schedule**

| | |
|---|---|
| Start: | [Date] |
| Architecture Initial Draft: | [Date] |
| Architecture Revised Draft 1: | [Date] |
| Architecture Revised Draft *n*: | [Date] |
| Architecture Final Draft: | [Date] |
| [AMS Decision]: | [Date] |

## A.1 AV-1 Overview and Summary Information (continued)



# 2 SCOPE: ARCHITECTURE VIEWS AND PRODUCTS

*This section identifies the views and products that have been developed and the temporal nature of the proposed architecture, such as the time frame covered, whether by specific years or by designations such as current, target, transitional, and so forth. Scope also identifies the organizations that fall within the scope of the architecture.*

## 2.1 VIEWS AND PRODUCTS DEVELOPED

*This section identifies the views and products applicable to the development of an architecture. Note that the sample table below is representative of the full suite of architecture products currently prescribed by the NASEAF. Tailoring of this table is expected depending upon the nature of a specific architecture. In acquisition phases requiring alternatives analysis, architectural representation of the preferred and non-preferred alternative solutions is identified here, as well.*

Architecture products applicable to the development of [Architecture Name] are identified in table 2.1.

| Product ID | Title |
|---|---|
| AV-1 | Overview and Summary Information |
| AV-2 | Integrated Dictionary |
| OV-1 | High-Level Operational Concept Graphic |
| OV-5 | Operational Activity Model |
| OV-6c | Operational Event-Trace Description |
| SV-1 | Systems Interface Description |
| SV-2 | Systems Communications Description |
| SV-4 | Systems Functionality Description |
| SV-6 | Systems Data Exchange Matrix |
| SV-7 | Systems Performance Parameters Matrix |
| SV-10c | Systems Event-Trace Description |
| SV-11 | Physical Schema |

## A.1 AV-1 Overview and Summary Information (continued)



### 2.2 TIME FRAMES ADDRESSED

*This section identifies the timeframe expressed by a specific architecture. Text may vary depending on whether the architecture is As-Is or To-Be.*

*As-Is Text:*

The As-Is [Architecture Name] described in this document covers the current period through the end of FY [current fiscal year].

*To-Be Text:*

The proposed To-Be architecture(s) described in this document covers the current period as follows:

[Architecture Name(s)]: FY [starting fiscal year] to FY [ending fiscal year]

*Additional To-Be architecture timeframes may follow depending upon acquisition phase.*

### 2.3 ORGANIZATIONS WITHIN SCOPE

*This section identifies all ATO organizational entities that are involved in the architecture effort.*

Organizations that are involved in this architecture effort are:

- *[Add organizations as appropriate.]*

## A.1 AV-1 Overview and Summary Information (continued)



# 3 PURPOSE AND VIEWPOINT

## 3.1 PURPOSE, ANALYSIS, AND QUESTIONS TO BE ANSWERED

### 3.1.1 Purpose

*This section describes the intent of or need for the architecture effort and what the architecture should demonstrate. For example, the purpose might be "to satisfy the architecture requirements of the FAA Acquisition Management System (AMS) during its Initial Investment Analysis Phase. In so doing, the architecture demonstrates the need to replace [name of legacy system]." Other types of purpose outside of AMS may also apply.*

The purpose for conducting the architecture effort described in this document is…

### 3.1.2 Analysis and Questions to Be Answered by Analysis of the Architecture

*This section describes the analytical techniques, specific or general, applied to achieve the architecture purpose, as stated above. At a minimum, this includes questions relevant to the architecture's integration, with other architectures and within its own architecture, and questions on gap analysis. Answers to the questions posed in this section are the subject of AV-1 section 6,* Findings. *Additionally, this section describes who is expected to perform the analysis, what decisions are expected to be made based on each form of analysis, who is expected to make those decisions, and what actions are expected to result.*

An [Analysis Technique] is applied to the architecture to assist in answering the following questions:

- *[Add question(s) as appropriate.] (may be in tabular form)*

Examination of these questions will be conducted by [e.g., organization, team, individual], the results of which help in making the following decisions and the expected course(s) of action…

- *[Add decisions and expected action(s)] (may be in tabular form)*

## 3.2 VIEWPOINT

*This section identifies the developmental viewpoint or perspective of the architecture (e.g., planner or decision maker).*

The [Architecture Name] is developed from the perspective of the [individual/organizational role(s)].

## A.1 AV-1 Overview and Summary Information (continued)



# 4 CONTEXT

*This section describes the setting in which the program architecture exists. It includes such things as mission, relevant goals, objectives, and vision statements, concepts of operation, etc. This section also identifies authoritative sources for the rules, criteria, and conventions that were followed. The tasking for the program architecture and known or anticipated linkages to other architectures is identified. Mission*

## 4.1 MISSION

*This section states the programmatic mission of the architecture under consideration.*

The mission of the [Architecture Name] is to…

## 4.2 GOALS, OBJECTIVES, AND VISION

### 4.2.1 Goals

*This section states the general intentions established for achieving the above stated mission.*

The goals of the architecture described in this document are:

- *[Add goals as appropriate.]*

### 4.2.2 Objectives

*This section states the specific, concise, and measurable objectives of the architecture for achieving the above stated goals.*

The objectives of the architecture described in this document are:

- *[Add objectives as appropriate.]*

### 4.2.3 Vision

*This section provides a general summary of the operational concept of the architecture under consideration (may be equivalent to the Description Section of OV-1). If the intent of this document is to describe multiple to-be architectures (i.e., Alternative Solutions), separate Vision descriptions may be necessary.*

## A.1 AV-1 Overview and Summary Information (continued)



### 4.3 RULES, CRITERIA, AND CONVENTIONS FOLLOWED

*This section identifies prescribed constraints for adherence to architecture development.*

The [Architecture Name] is developed consistent with the following rules, criteria, and conventions:

- *[Add rule set, criteria, or convention as appropriate.] Examples:*
- *Office of Management and Budget (OMB) planning and budgeting requirements*
- *FAA Acquisition Management System (AMS) policy and processes*
- *JRC Procedures and scheduling constraints*

### 4.4 TASKING AND LINKAGES TO OTHER ARCHITECTURES

#### 4.4.1 Tasking for Architecture Project

*This section identifies the cross-organizational teams, individuals, etc. assigned to specific roles in the development of the architecture.*

The [Named Architect from Section 1.2 above] has tasked the following entities to develop the architecture described in this document:

| ASSIGNEE | ROLE |
|---|---|
| [e.g., organizational team, individual, etc. 1] | [role 1] |
| [e.g., organizational team, individual, etc. 2] | [role 2] |

#### 4.4.2 Linkages to Other Architectures

*This section lists all architectures related to or linked in context to the architecture described in this document. Note: the FAA Capital Investment Plan (CIP) designators may be used when linkages are known, but not represented as a formal architecture.*

Linkages from the architecture under consideration to other architectures are as follows:

- *[List identified architectures as appropriate.]*

## A.1 AV-1 Overview and Summary Information (continued)



# 5 ASSUMPTIONS AND CONSTRAINTS

### 5.1.1 Assumptions

*The assumptions in this section are expressed in terms of the architecture (subject system) under development. Assumptions may address subject system's dependencies upon the expected availability of resources, the implementation plans of other systems, timeliness of results of R&D efforts, etc. For example, system X will be fully operational in order for certain functionality in subject system to work as intended.*

The [Architecture Name] was developed under the following assumptions:

- *[Add assumptions as appropriate.]*

### 5.1.2 Constraints

*The constraints in this section are expressed in terms of the architecture under development. For example, constraints may address the effects of known time or resource limitations, other system's dependencies upon the implementation plans of subject system, etc. For example, planned implementation of system Y is dependent upon implementation of subject system by a certain date.*

The [Architecture Name] was developed within the following constraints:

- *[Add constraints as appropriate.]*

## A.1 AV-1 Overview and Summary Information (continued)



# 6 TOOLS AND FILE FORMATS USED

*The Tools and File Formats Used section identifies the tool suite used to develop the program architecture and related products. Specify the tool names and file formats used in building and documenting the architecture.*

## A.1 AV-1 Overview and Summary Information (continued)



# 7 FINDINGS

*This section discusses the findings and recommendations that have been developed based on the architecture effort. Examples of findings include identification of shortfalls, recommended system implementations, and opportunities for technology insertion that are specific to the program.*

## 7.1 ANALYSIS RESULTS

*This section provides a summary of the analytical results that answer the questions posed from Section 3.1.2 above, as well as the assessment of these results.*

## 7.2 RECOMMENDATIONS

*This section provides the set of recommendations formulated from the assessment of analysis results and the potential courses of action identified in Section 3.1.2.*

## A.2 AV-2 Integrated Dictionary

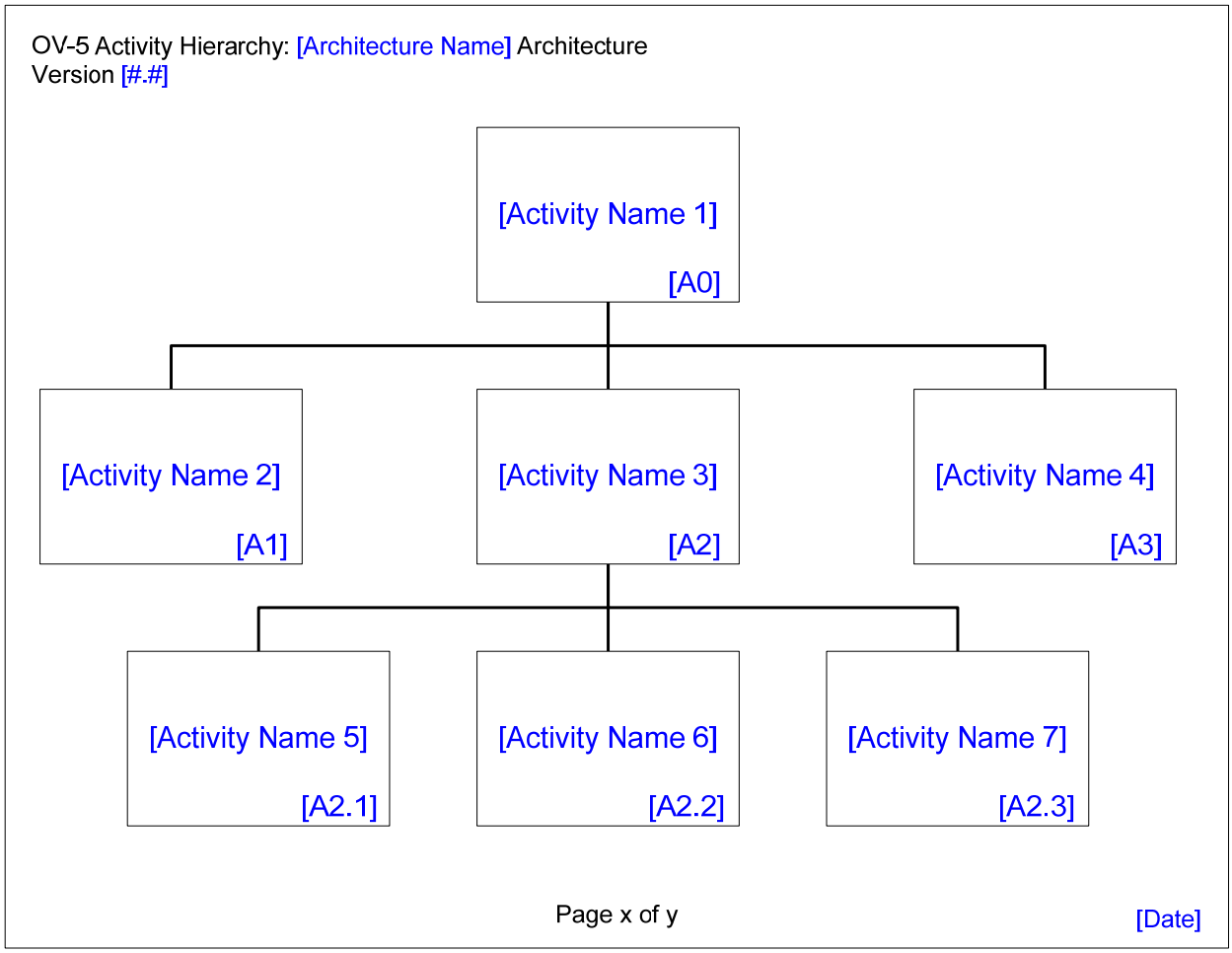
AV-2 Integrated Dictionary: [Architecture Name] Architecture
Version [#.#]

| Element Type | Identifier | Name | Description |
|---|---|---|---|
| [element type] | [element ID] (if applicable) | [element name] | [description] |
| [element type] | [element ID] (if applicable) | [element name] | [description] |
| [element type] | [element ID] (if applicable) | [element name] | [description] |

Page *x* of *y* [Date]

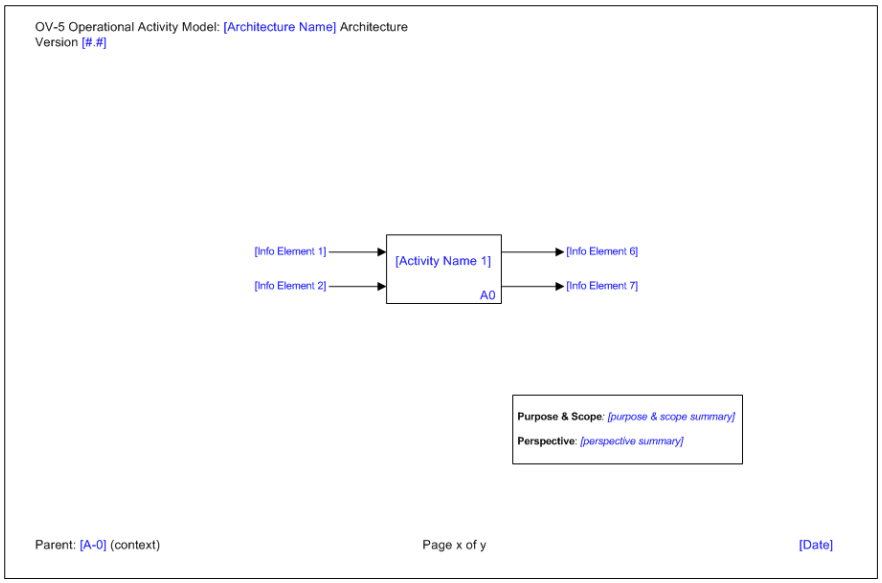
## A.3 OV-5 Operational Activity Model

ACTIVITY HIERARCHY

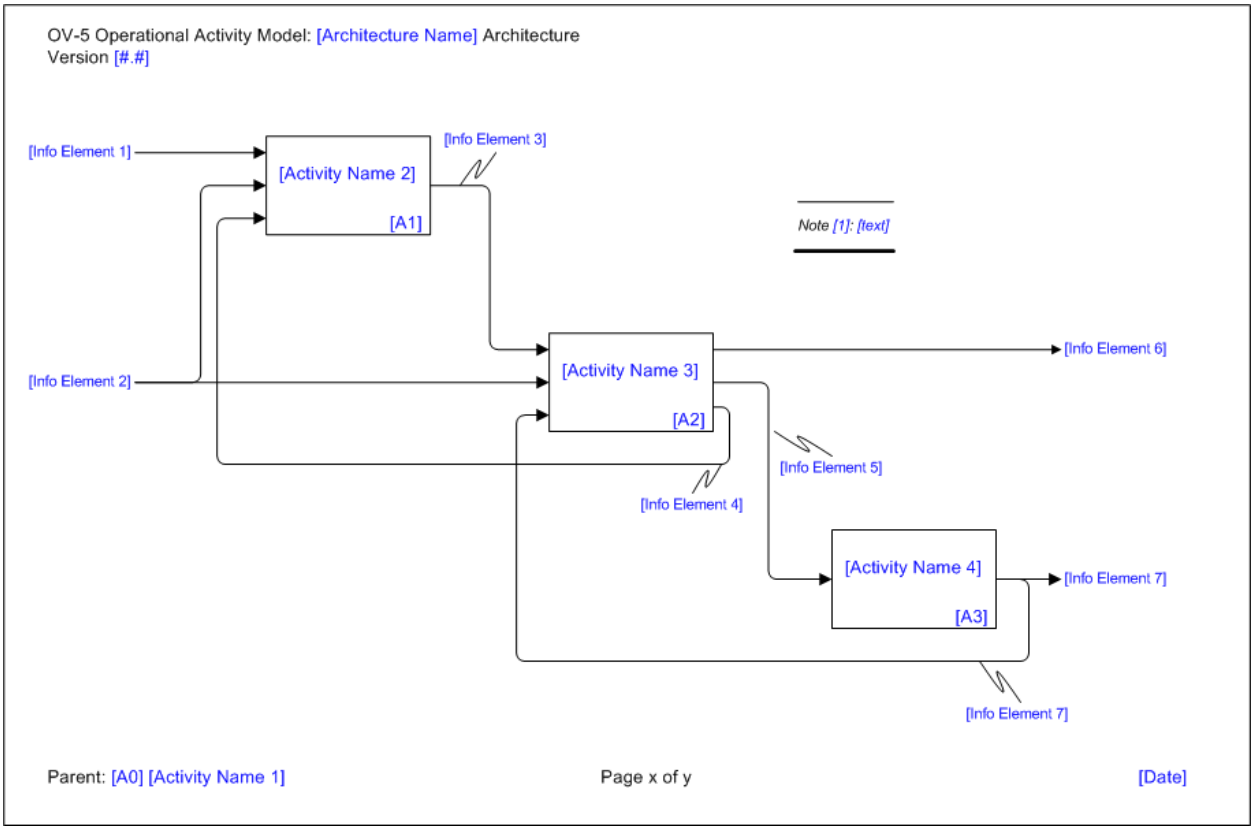


## A.3 OV-5 Operational Activity Model (continued)

ACTIVITY MODEL CONTEXT

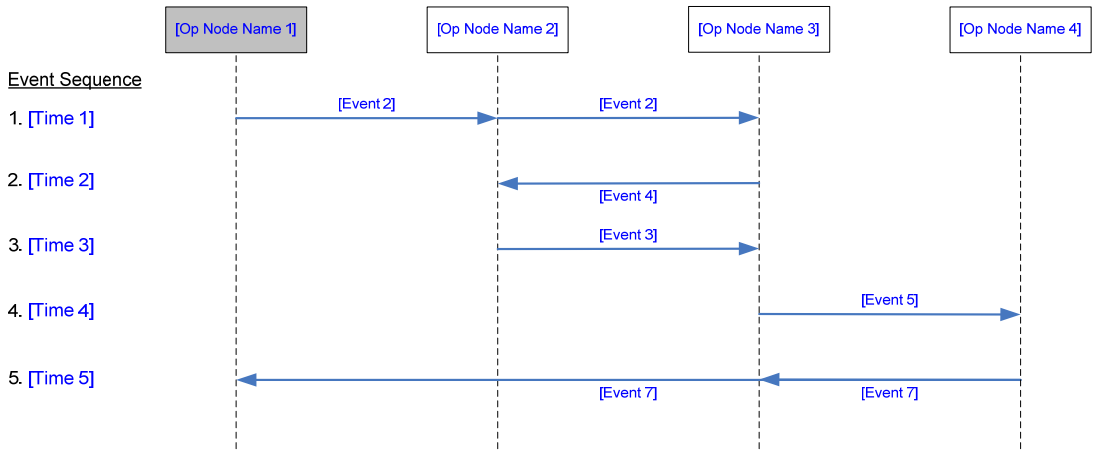



ACTIVITY MODEL



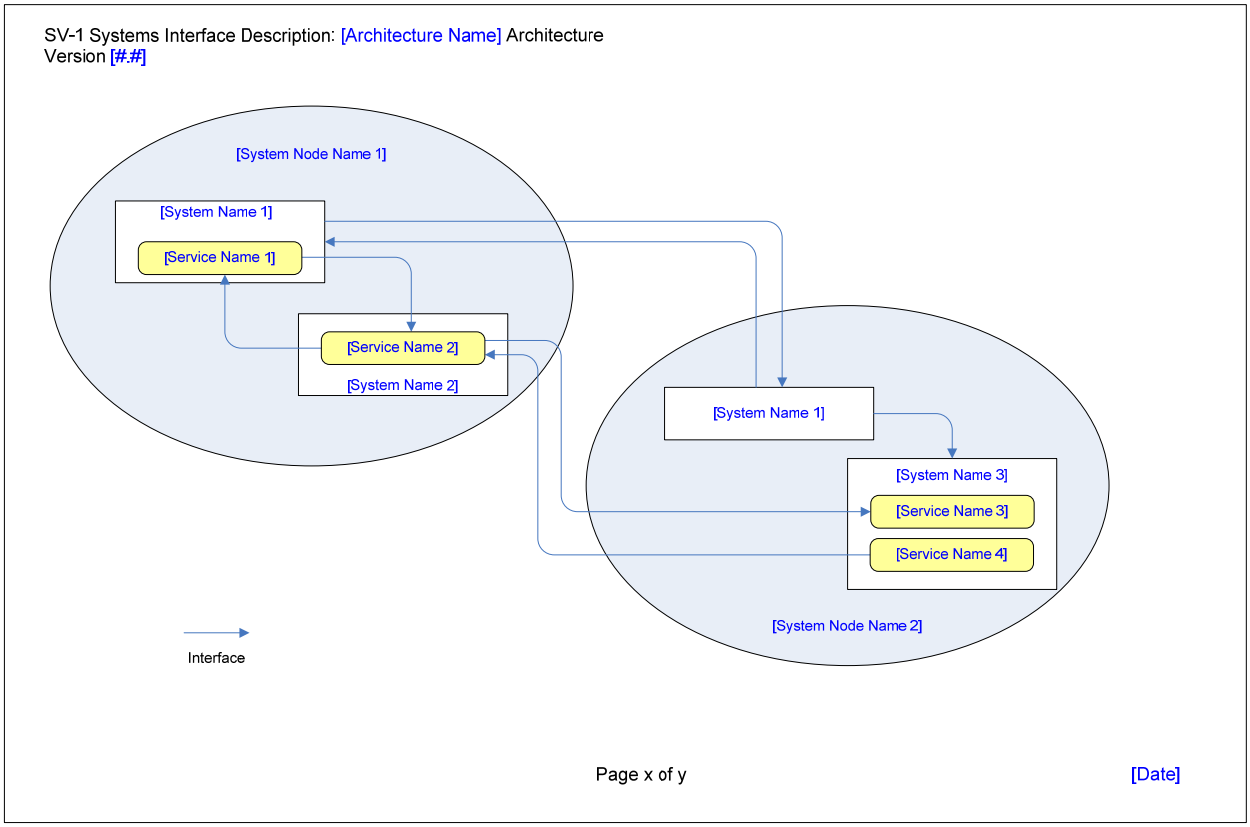
## A.4 OV-6c Operational Event-Trace Description

OV-6c Operational Event-Trace Description: [Architecture Name] Architecture
Version [#.#]

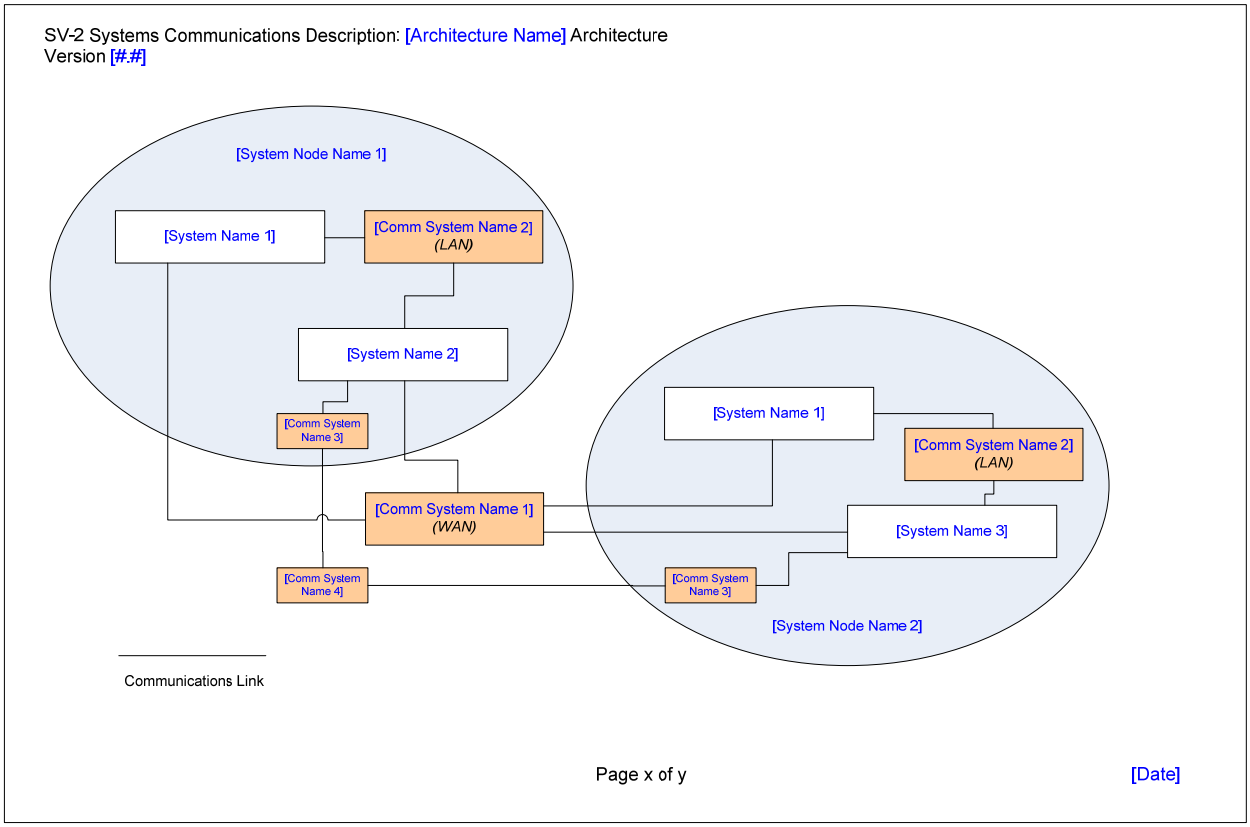



Page x of y [Date]

## A.5 SV-1 Systems Interface Description



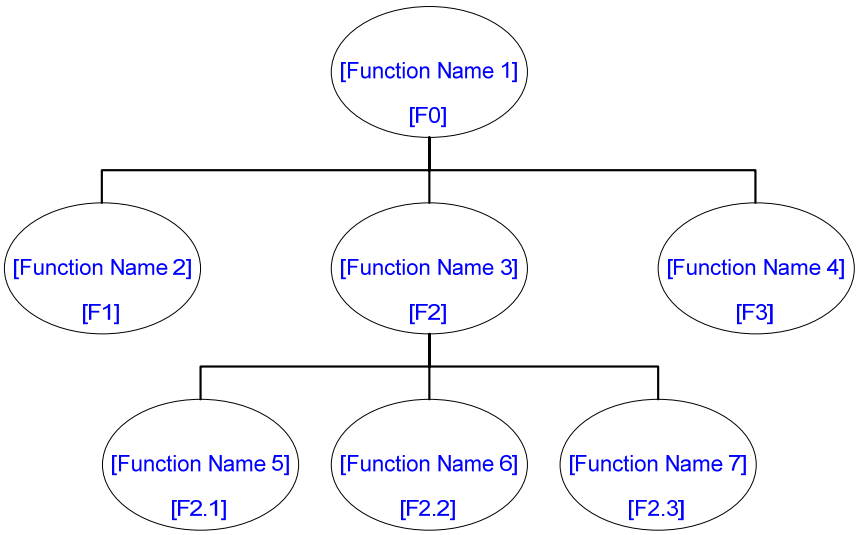
## A.6 SV-2 Systems Communications Description



## A.7 SV-4 Systems/Services Functionality Description

FUNCTIONAL HIERARCHY

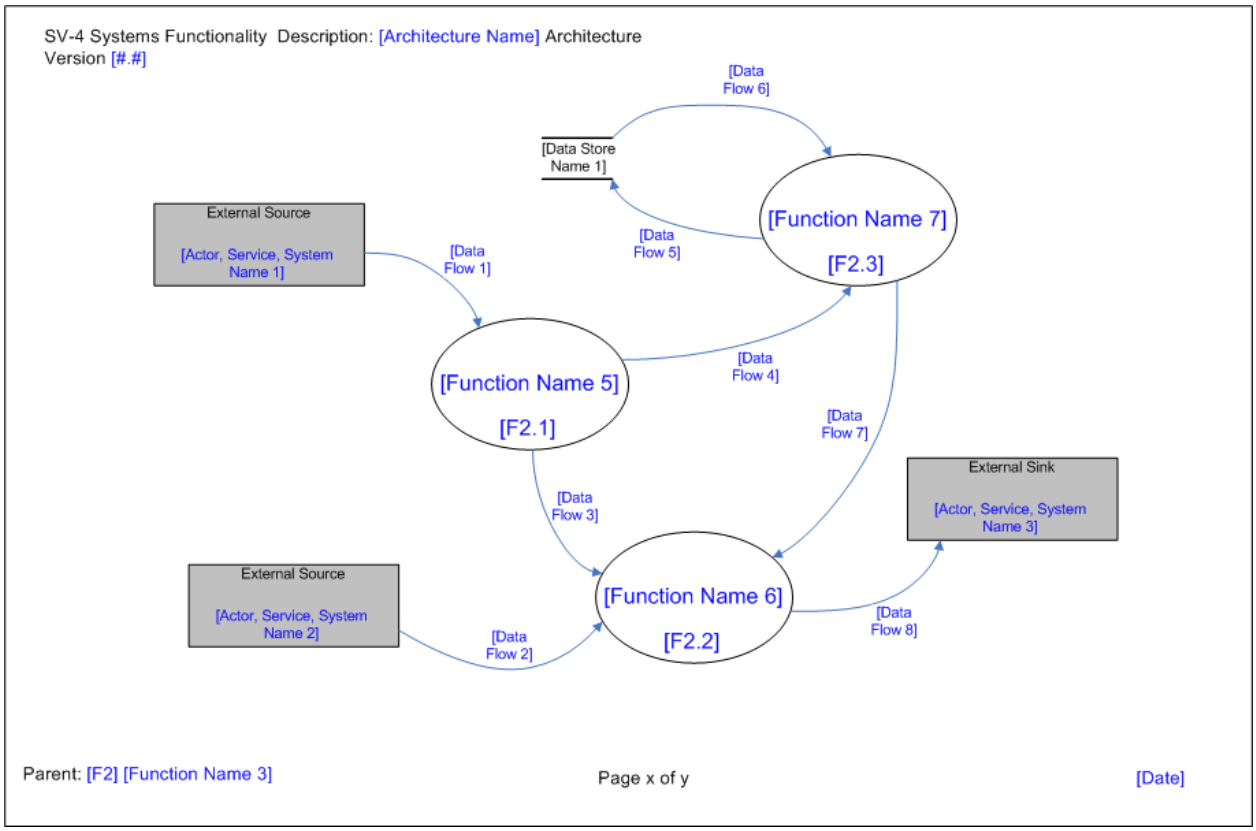
SV-4 Systems (Services) Functionality Description: [Architecture Name] Architecture
Version [#.#]




Page x of y [Date]

## A.7 SV-4 Systems/Services Functionality Description (continued)

FUNCTIONALITY MODEL



## A.8 SV-6 Systems Data Exchange Matrix

SV-6 Systems Data Exchange Matrix: [Architecture Name] Architecture
Version [#.#]

| Systems Data Exchange | Interface | Needline | Systems Data Element | Sending System | Sending Systems Function | Receiving System | Receiving Systems Function | Content Description |
|---|---|---|---|---|---|---|---|---|
| [SDX 1] | [Interface 1] | [Needline 1] | [Data Element 1] | [System Name 1] | [Function name 1] | [System Name 2] | [Function name 2] | [SDX 1 content description] |
| [SDX *n*] | [Interface *n*] | [Needline *n*] | [Data Element *n*] | [System Name *n*] | [Function name *n*] | [System Name *x*] | [Function name *x*] | [SDX *n* content description] |

Page *x* of *y* [Date]

## A.9 SV-7 Systems Performance Parameters Matrix

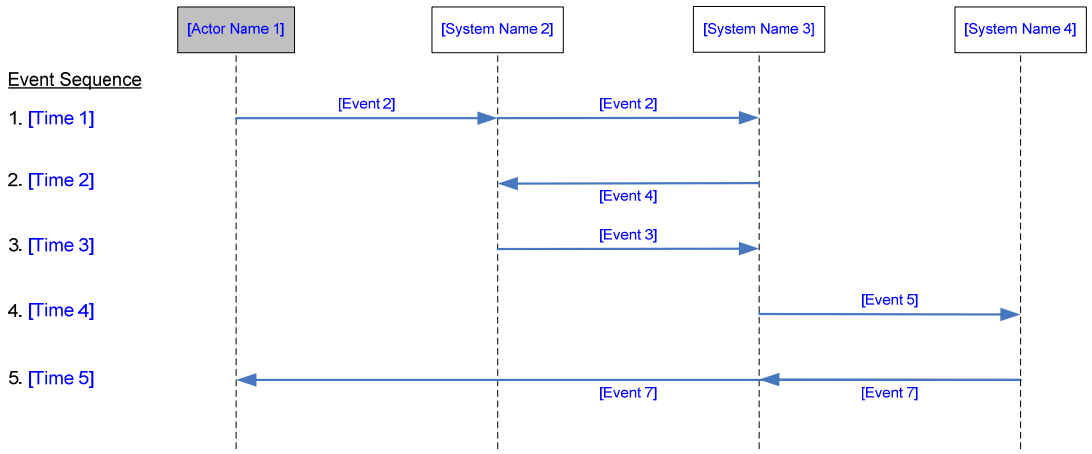
SV-7 Systems Performance Matrix: [Architecture Name] Architecture
Version [#.#]

| Resource | Performance Measure Type 1] | | Performance Measure Type 2] | | Performance Measure Type *n* ] | |
|---|---|---|---|---|---|---|
| | Threshold | Objective | Threshold | Objective | Threshold | Objective |
| [Systems Data Exchange 1] | [Threshold Measure 1] | [Objective Measure 1] | [Threshold Measure 3] | [Objective Measure 3] | [Threshold Measure *i* ] | [Objective Measure *i* ] |
| [Systems Data Exchange 2] | [Threshold Measure 2] | [Objective Measure 2] | [Threshold Measure 4] | [Objective Measure 4] | [Threshold Measure *n* ] | [Objective Measure *n* ] |
| [Systems Data Exchange *n* ] | [Threshold Measure *a* ] | [Objective Measure *b* ] | [Threshold Measure *c* ] | [Objective Measure *c* ] | [Threshold Measure *x* ] | [Objective Measure *x* ] |

Page *x* of *y* [Date]

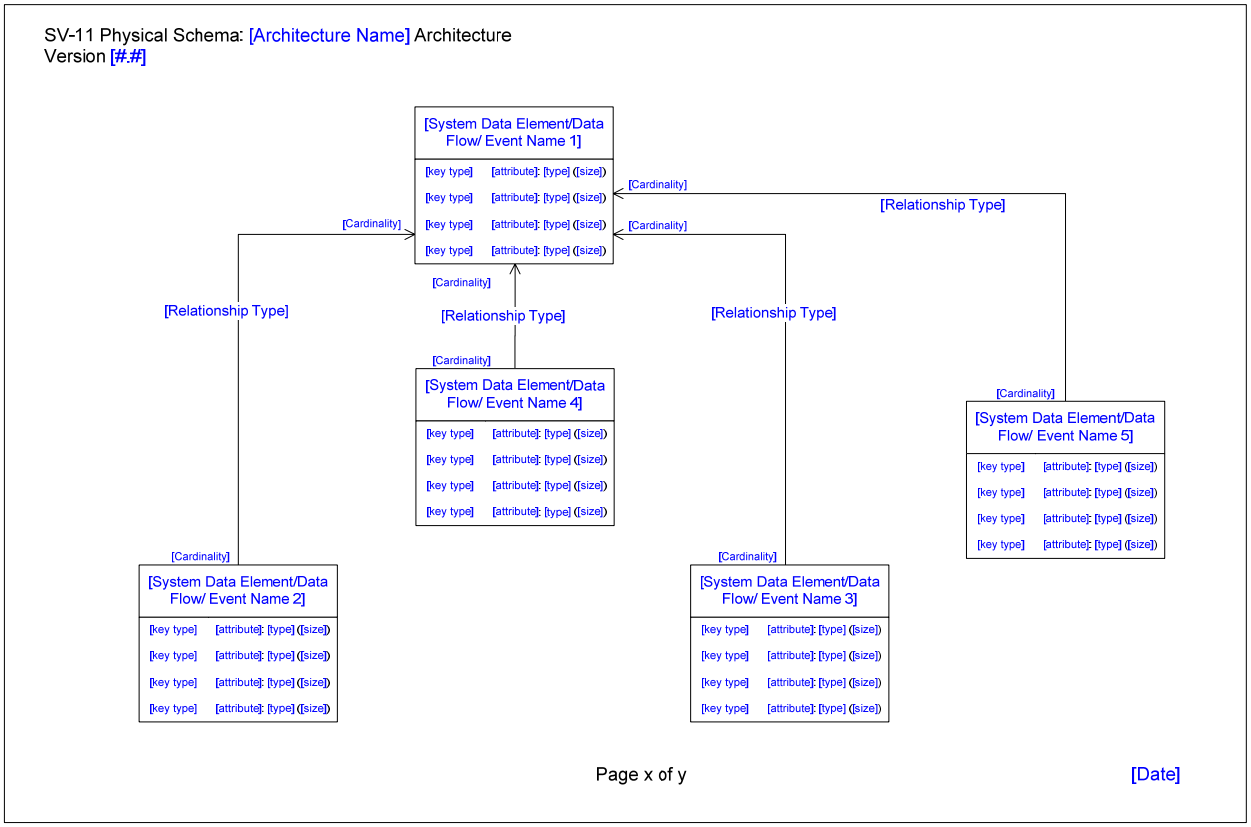
## A.10 SV-10c Systems Event –Trace Description

SV-10c Systems Event-Trace Description: [Architecture Name] Architecture
Version [#.#]



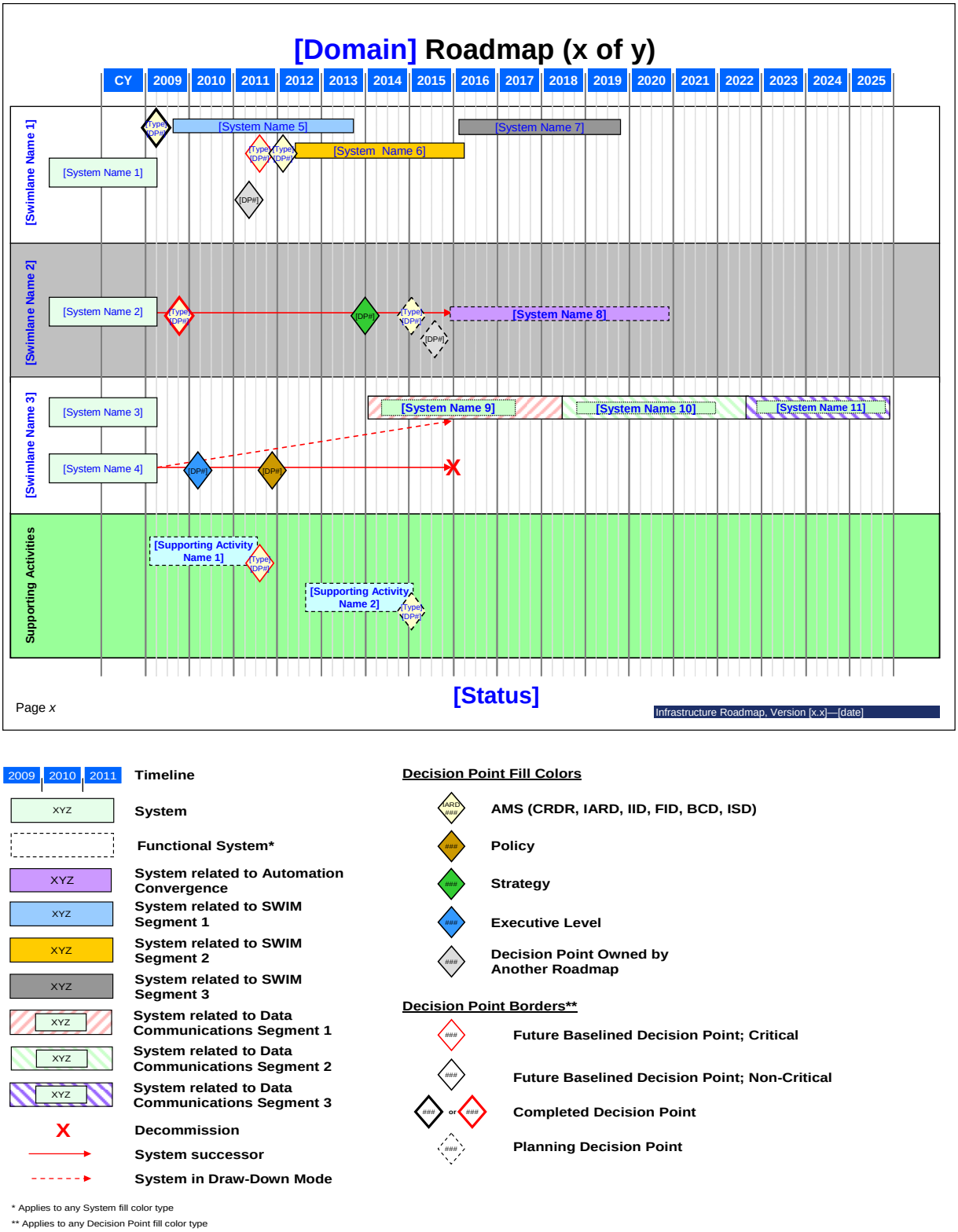

Page x of y [Date]

## A.11 SV-11 Physical Schema



## A.12 Infrastructure Roadmaps

ROADMAP



## A.12 Infrastructure Roadmaps (continued)

DECISION POINTS

### [Domain] Roadmap: Decision Points

| DP # | Target Date | High Priority | Domain | Name |
|---|---|---|---|---|
| [DP ID 1] | [CY] | Y | [Domain] | [Decision Point Name] |
| [DP ID 2] | [CY] | Y | [Domain] | [Decision Point Name] |
| [DP ID 3] | [CY] | N | [Domain] | [Decision Point Name] |
| [DP ID 4] | [CY] | N | [Domain] | [Decision Point Name] |
| [DP ID 5] | [CY] | Y | | [Decision Point Name] |
| [DP ID 6] | [CY] | N | | [Decision Point Name] |
| [DP ID 7] | [CY] | N | | [Decision Point Name] |
| [DP ID 8] | [CY] | N | | [Decision Point Name] |

Page *x*

**[Status]**

Infrastructure Roadmap, Version [x.x]—[date]

## A.12 Infrastructure Roadmaps (continued)

ASSUMPTIONS

### Aircraft Roadmap: Assumptions

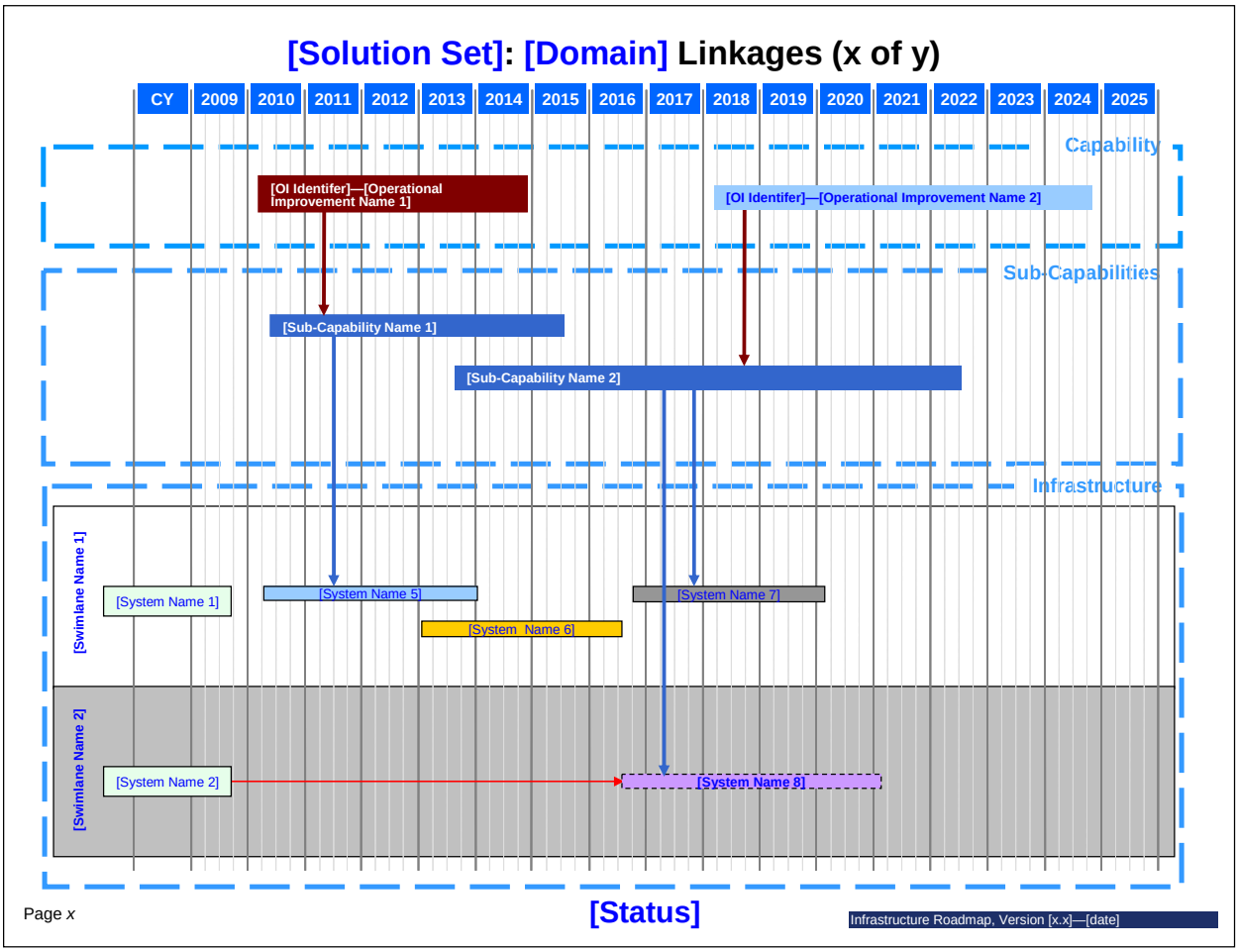
| Identifier | Description |
|---|---|
| [ID #] | [Assumption Description] |
| [ID #] | [Assumption Description] |
| [ID #] | [Assumption Description] |
| [ID #] | [Assumption Description] |

Page *x*

**[Status]**

Infrastructure Roadmap, Version [x.x]—[date]

## A.13 Sub-Capability Roadmaps




### CAPABILITIES/SUB-CAPABILITIES

- XXXXXX — Mid-Term Operational Improvement Title and Initial Operating Capability Date Range
- XXXXXX — Far-Term Operational Improvement Title and Initial Operating Capability Date Range
- ABC — Sub-Capability and IOC Date Range

### LINKAGES

- Operational Improvement to Sub-Capability Linkage
- Sub-Capability to Infrastructure Roadmap linkage

### TIMELINE

CY | 2008 | 2009 | 2010 | 2011 | 2012 | 2013 | 2014 | 2015

### INFRASTRUCTURE

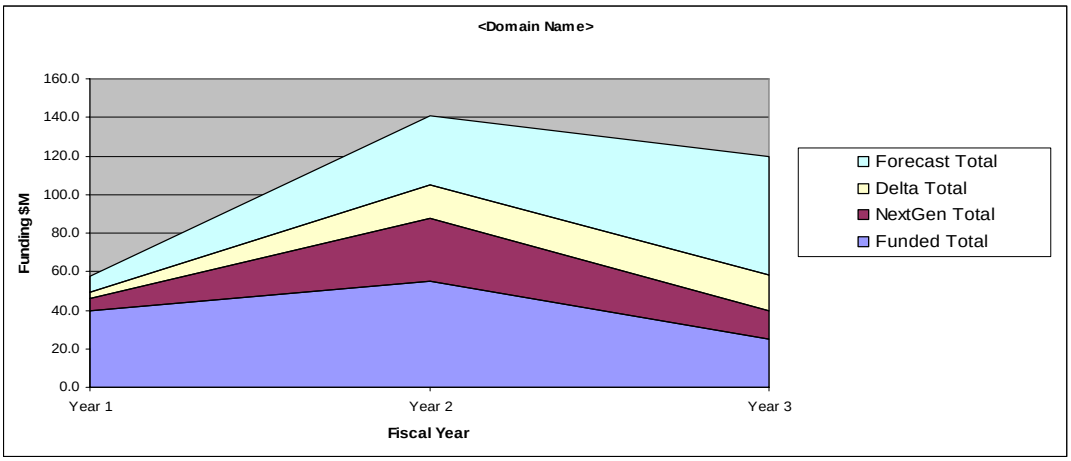
- XYZ — System
- Functional System*
- XYZ — System related to Automation Convergence
- XYZ — System related to SWIM Segment 1
- XYZ — System related to SWIM Segment 2
- XYZ — System related to SWIM Segment 3
- XYZ — System related to Data Communications Segment 1
- XYZ — System related to Data Communications Segment 2
- XYZ — System related to Data Communications Segment 3
- System successor
- System in Draw-Down Mode
- X — Decommission

## A.14 Funding Profile

<Domain Name> Funding
Version *x.x*

**<Domain Name> Summary**

| Name | Year 1 | Year 2 | Year 3 | Total |
|---|---|---|---|---|
| **Funded Total** | <$ amt> | <$ amt> | <$ amt> | **<$ amt>** |
| **NextGen Total** | <$ amt> | <$ amt> | <$ amt> | **<$ amt>** |
| **Delta Total** | <$ amt> | <$ amt> | <$ amt> | **<$ amt>** |
| **Forecast Total** | <$ amt> | <$ amt> | <$ amt> | **<$ amt>** |




| | **Funded Programs** (Per August 2008 CIP) | | | | | |
|---|---|---|---|---|---|---|
| **CIP #** | **Name** | **DP #** | **Year 1** | **Year 2** | **Year 3** | **Total** |
| <CIP ref 1>* | <Program Name 1> | <DP ID #> | <$ amt> | <$ amt> | <$ amt> | **<$ amt>** |
| <CIP ref 2> | <Program Name 2> | <DP ID #> | <$ amt> | <$ amt> | <$ amt> | **<$ amt>** |
| <CIP ref *n*> | <Program Name *n*> | <DP ID #> | <$ amt> | <$ amt> | <$ amt> | **<$ amt>** |
| | **Funded Total** | | **<$ amt>** | **<$ amt>** | **<$ amt>** | **<$ amt>** |

| | **NextGen** | | | | | |
|---|---|---|---|---|---|---|
| **CIP #** | **Name** | **DP #** | **Year 1** | **Year 2** | **Year 3** | **Total** |
| <CIP ref 1> | <Program Name 1> | <DP ID #> | <$ amt> | <$ amt> | <$ amt> | **<$ amt>** |
| <CIP ref 2> | <Program Name 2> | <DP ID #> | <$ amt> | <$ amt> | <$ amt> | **<$ amt>** |
| <CIP ref *n*> | <Program Name *n*> | <DP ID #> | <$ amt> | <$ amt> | <$ amt> | **<$ amt>** |
| | **Funded Total** | | **<$ amt>** | **<$ amt>** | **<$ amt>** | **<$ amt>** |

| | **Delta** | | | | | |
|---|---|---|---|---|---|---|
| **CIP #** | **Name** | **DP #** | **Year 1** | **Year 2** | **Year 3** | **Total** |
| <CIP ref 1> | <Program Name 1> | <DP ID #> | <$ amt> | <$ amt> | <$ amt> | **<$ amt>** |
| <CIP ref 2> | <Program Name 2> | <DP ID #> | <$ amt> | <$ amt> | <$ amt> | **<$ amt>** |
| <CIP ref *n*> | <Program Name *n*> | <DP ID #> | <$ amt> | <$ amt> | <$ amt> | **<$ amt>** |
| | **Funded Total** | | **<$ amt>** | **<$ amt>** | **<$ amt>** | **<$ amt>** |

| | **Forecasted Programs** | | | | | |
|---|---|---|---|---|---|---|
| **CIP #** | **Name** | **DP #** | **Year 1** | **Year 2** | **Year 3** | **Total** |
| <CIP ref 1> | <Program Name 1> | <DP ID #> | <$ amt> | <$ amt> | <$ amt> | **<$ amt>** |
| <CIP ref 2> | <Program Name 2> | <DP ID #> | <$ amt> | <$ amt> | <$ amt> | **<$ amt>** |
| <CIP ref *n*> | <Program Name *n*> | <DP ID #> | <$ amt> | <$ amt> | <$ amt> | **<$ amt>** |
| | **Funded Total** | | **<$ amt>** | **<$ amt>** | **<$ amt>** | **<$ amt>** |

*<Referenced Notes, e.g., CIP ref 1…>

**Assumptions:**
1. <assumption 1 text>
2. <assumption 2 text>
3. <assumption *n* text>