

# System Wide Information Management (SWIM)

## **Governance Policies**



Version Number 1.1

August 13, 2010

**SIGNATURE PAGE**

---

Jim Robb  
SWIM Requirements and Governance Lead

Date

---

Ahmad Usmani  
SWIM Program Manager

Date

Federal Aviation Administration  
800 Independence Avenue, SW  
Washington, D.C. 20591

**DOCUMENT CHANGE HISTORY**

<b>Version</b>	<b>Date</b>	<b>Description of Changes</b>
1.0	07/27/2009	Addressed comments from numerous stakeholders to drafts
1.1	08/13/2010	Address additional stakeholder comments and lessons learned from creating process documentation

## Table of Contents

<b>Table of Contents.....</b>	<b>1</b>
<b>1 Introduction.....</b>	<b>1</b>
1.1 Purpose.....	1
1.2 Scope.....	1
1.3 Document Organization.....	1
<b>2 References.....</b>	<b>2</b>
<b>3 Strategic SOA Policies.....</b>	<b>4</b>
3.1 SOA Technology Acquisition Policies.....	4
3.1.1 Purpose.....	4
3.1.2 Policy Statements.....	4
3.2 Enterprise Architecture Policies.....	4
3.2.1 Purpose.....	4
3.2.2 Policy Statements.....	5
3.3 Opportunity Management for SOA Services.....	5
3.3.1 Purpose.....	5
3.3.2 Policy Statements.....	5
3.4 Interoperability, Reuse and Standards Policies.....	5
3.4.1 Purpose.....	5
3.4.2 Policy Statements.....	6
3.5 NAS Service Registry/Repository Policies.....	8
3.5.1 Overview.....	8
3.5.2 Policy Statements.....	8
<b>4 Service Design-Time Policies.....</b>	<b>10</b>
4.1 Namespace and Schema Policies.....	10
4.1.1 Purpose.....	10
4.1.2 Policy Statements.....	10
4.2 Services Interface Design Policies.....	11
4.2.1 Purpose.....	11
4.2.2 Policy Statements.....	11
4.3 Services Technical and Design Policies.....	13
4.3.1 Purpose.....	13
4.3.2 Policy Statements.....	13
4.4 Information Security Policies.....	14
4.4.1 Purpose.....	14
4.4.2 Policy Statements.....	14
4.5 Service Development Process Policies.....	15
4.5.1 Purpose.....	15
4.5.2 Policies.....	16

4.6	Service Lifecycle Management Policies .....	17
4.6.1	Purpose.....	17
4.6.2	Policy Statements.....	17
4.7	Services Operations Readiness Policies.....	18
4.7.1	Purpose.....	18
4.7.2	Policy Statements.....	18
4.8	Service Provisioning Policies.....	19
4.8.1	Purpose.....	19
4.8.2	Policy Statements.....	19
4.9	Service Consumer Policies.....	19
4.9.1	Purpose.....	19
4.9.2	Policy Statements.....	19
<b>5</b>	<b>Runtime and Operational Policies.....</b>	<b>20</b>
5.1	Messaging and Routing Policies .....	20
5.1.1	Purpose.....	20
5.1.2	Policy Statements.....	20
5.2	Runtime Security Policies .....	20
5.2.1	Purpose.....	20
5.2.2	Policies.....	20
5.3	Service Management Policies .....	21
5.3.1	Purpose.....	21
5.3.2	Policy Statements.....	21
5.4	Maintenance and Support Policies .....	22
5.4.1	Purpose.....	22
5.4.2	Policies.....	22
<b>6</b>	<b>Glossary .....</b>	<b>23</b>
<b>7</b>	<b>Acronyms.....</b>	<b>28</b>

# 1 Introduction

## 1.1 Purpose

System Wide Information Management (SWIM) refers to both an architectural approach to integrating software components and a program within the FAA whose objective is to realize such an approach. SWIM embodies an architectural approach for a NAS service-oriented architecture (SOA) in which individual software components offer discrete, well-defined information services that can be organized in a variety of ways to provide desired functionality to the NAS enterprise. As such, SWIM will employ mechanisms commonly associated with SOAs such as a Registry/Repository for the NAS enterprise. These mechanisms will be implemented by means of core services and SWIM Program Management Office (PMO) mandatory software standards and governance.

Implementing SOA requires breaking down the system, system of systems, or enterprise into a collection of service providers and consumers and providing the necessary interoperability among them. This set of service providers and consumers is expected to behave as a community sharing information among them. Ultimately, this will benefit the enterprise by reusing services and eliminating duplicative functionality across traditional organizational boundaries, thus enabling the enterprise greater agility to adapt to changing business requirements. Therefore, interoperability and reuse are key goals of the governance effort, to facilitate the information-sharing community.

## 1.2 Scope

This document contains policy statements related to the Governance of SOA in the NAS by SWIM and associated stakeholders as described in the SWIM Governance Plan [SWIM GP]. These statements describe the organizational behaviors needed to produce the desired enterprise SOA outcome. Processes, procedures, guidelines, and requirements (i.e. the “how” and “by whom” of Governance) derived from these policies are documented elsewhere as described in Section 2 – References below.

The policies contained herein are applicable to: the SWIM program and its core services, NAS programs implementing SWIM services, and the SWIM facing NAS enterprise services themselves. Of key importance are the *service contracts* for these SWIM services.

## 1.3 Document Organization

The principal contents of this document are divided into three main governance policy sections. Section 3, “Strategic SOA Policies,” focuses on the overarching governance policies of SWIM SOA. Sections 4 and 5 contain policies to govern design-time and runtime, respectively.

## 2 References

The following references have been cited in the body of the document:

- [COTSPMP] *SWIM COTS Product Management Plan Version 1.3, Draft - May 28, 2010*
- [NASEAF1] *National Airspace System Enterprise Architecture Framework (NASEAF) Version 2.0 Volume I: Definitions and Guidelines, September 30, 2007.*
- [NASEAF2] *National Airspace System Enterprise Architecture Framework (NASEAF) Version 2.0 Volume II: Product Descriptions, September 30, 2007.*
- [NASEAF3] *National Airspace System Enterprise Architecture Framework (NASEAF) Version 2.0 Volume III: Product Implementation Methodologies, September 30, 2007.*
- [NASSEM] NAS System Engineering Manual, Version 3.1, 11 October 2006
- [NIST800-95] *Guide to Secure Web Services, National Institute of Standards and Technology (NIST) Special Publication 800-95, August 2007.*
- [STD025F] FAA-STD-025F, US DOT FAA Standard, Preparation of Interface Documentation, November 30, 2007
- [STD063] FAA-STD-063, Standard Practice, XML Namespaces, May 1, 2009
- [STD064] FAA-STD-064, Standard Practice, Service Registration, May 1, 2009
- [STD065] FAA-STD 065, Standard Practice, Web Service Description Documents, February 26, 2010
- [STD066] FAA STD-066, Standard Practice, Web Service Taxonomies, February 26, 2010
- [SWIM CM] SWIM Configuration Management Plan v2.1, June 10, 2008
- [SWIM FPR] SWIM Final Program Requirements Segment 1 Revision 7.3, 23 May 2007
- [SWIM GP] SWIM Governance Plan Version 3.0, June 23, 2010
  
- [SWIM PLA] SWIM Program Level Agreements
- [SWIM PSP] SWIM Program Safety Plan
- [SWIM QAP] SWIM Quality Assurance Plan, Version 3, 14 April 2008
- [SWIM SEMP] SWIM System Engineering Management Plan, Version 5.1, 9 June 2008
  
- [SWIM SR] SWIM Service Roadmap
- [SWIM SRI] SWIM Service Registry/Repository IRD
- [NSRR SDM] NAS Service Registry/Repository Data Model DRAFT v0.3, March

	22, 2010
[SWIM SvSD]	SWIM Service Specification Document (SvSD) Segment 1, Release 1.6, March 31, 2009
[SWIM VMP]	SWIM Version Management Policies, v1.1, February 19, 2009.
[SWIMPSTD]	<i>SWIM Product Standardization Version 0.1</i> , February 24 2009
[SWIMRMP]	<i>SWIM Requirements Management Plan Version 3.0</i> , June 9, 2008
[SWIMSCT]	SWIM Service Contract Template (in development)
[SWIMTO]	<i>System-Wide Information Management (SWIM) Technical Overview</i> , Version 2.1, March 28, 2008.
[SIBP]	SWIM Interoperability Basic Profile, consisting of annotations to Web Services Interoperability Organization's <i>Basic Profile, Version 1.2</i> , March 28, 2007
[SIBSP]	SWIM Interoperability Basic Security Profile consisting of annotations to Web Services Interoperability Organization's <i>Basic Security Profile, Version 1.1</i> .
[TSBChart]	Test Standards Board Charter, Version 1.6, 8 June 2007
[WSISCTC]	Web Service Interoperability Organization's <i>Security Challenges, Threats and Countermeasures</i> , 6 May 2005.
[XMLGWQVL]	SWIM XML Gateway QVL Qualified Vendors List (QVL) for the XML Gateway Product, March 9, 2009

## **3 Strategic SOA Policies**

Strategic SOA policies help support the overall direction and the future state of the SOA initiative.

### **3.1 SOA Technology Acquisition Policies**

#### **3.1.1 Purpose**

The FAA Acquisition Management System (AMS) establishes policy and guidance for all aspects of acquisition management for the complete lifecycle of investment programs within the FAA. SWIM SOA Technology Acquisition Policies complement the AMS by providing specific guidelines to programs for monitoring, evaluating and vetting new Service Oriented Architecture (SOA) products and technology.

#### **3.1.2 Policy Statements**

1. For all prospective SOA technology acquisition investments, Programs shall conduct a best value evaluation by emphasizing the broader NAS enterprise scope of SOA infrastructure, rather than a single program and its immediate interfaces.
2. All new SOA technology shall be approved by the SWIM Configuration Review Board before being used to support the enterprise SOA initiative.
3. Acquisition of equipment, software, and/or services for which Qualified Vendor Lists (QVL) have been established by the SWIM Program Management Office (PMO) shall be restricted to suppliers on the QVL.
4. Service implementing programs shall acquire approved enterprise facing Commercial-Off-The-Shelf (COTS) products through only the SWIM COTS Product Repository (SCPR) in accordance with the COTS Product Management Plan [COTSPMP].

### **3.2 Enterprise Architecture Policies**

#### **3.2.1 Purpose**

SWIM Enterprise Architecture (EA) Policies provide a framework for building an FAA SOA in the most timely, economically and technically feasible manner possible. The two over-arching goals are to: (1) enable interoperability through the adoption of principles that govern implementation architectures and compliance to standards and, (2) enable service software reuse. The EA artifacts assist in aligning the information technology investment with business objectives, promote rationalization of services, support release planning across programs, and provide the framework blueprints for new systems and/or services.

To meet the above stated goals, the EA approach should have a minimal set of constraints that allows the FAA programs to realize consistency across the NAS. In addition,

architecture discipline must be aligned across SOA domains, or they will not be able to gain the maximum benefits of SOA such as service reuse, interoperability, and reduced maintenance costs.

### **3.2.2 Policy Statements**

1. FAA programs shall produce EA views for the appropriate stages in the service development life cycle in accordance with NASEAF guidance and Appendix 3 of the SWIM Requirements Management Plan [SWIMRMP].

## **3.3 Opportunity Management for SOA Services**

### **3.3.1 Purpose**

SWIM Opportunity Management for SOA Services Policies guide the evaluation of proposed new programs in the context of the NAS enterprise in order to identify where existing services may be reused to attain desired functionality. In addition, the policies guide the development of new functionality toward implementation as services for future reuse in new applications.

### **3.3.2 Policy Statements**

1. The analysis phase of the FAA AMS process for candidate programs shall include analysis of opportunities to leverage existing SWIM services and to supplement the SWIM service portfolio with additional SOA services.
2. When a new service is proposed the SWIM service portfolio shall be reviewed to determine whether the candidate services and operations are new opportunities for a SOA implementation.
3. All candidate NAS enterprise facing services shall be approved by the NAS Enterprise Architecture Board (EAB) prior to entering the proposed stage of the SWIM service lifecycle.

## **3.4 Interoperability, Reuse and Standards Policies**

### **3.4.1 Purpose**

In order to promote information sharing among heterogeneous applications across the NAS, the two primary goals of the SWIM SOA are to ensure interoperability and to provide opportunities for reuse. SOA requires individual services that conform to common design and technology standards for interoperability, reusability, and other benefits to be fully realized. The SWIM SOA interoperability, reuse, and standards policies assure service interoperability through standardization. These policies are also intended to promote the characteristics of services that support their reuse by others.

### 3.4.2 Policy Statements

1. Services shall be implemented using SOAP/HTTP(S), XML/HTTP(S), SOAP/Java Message Service (JMS), or XML/JMS.
2. FAA Programs shall use the SWIM provided service container, in accordance with the [SWIMPSTD].
3. All web services implemented with SOAP/HTTP(S) by FAA programs shall be compliant with a SWIM program-modified WS-I Basic Profile named SWIM Interoperability Basic Profile [SIBP].
4. The [SIBP] shall tailor the clarifications, refinements, interpretations and amplifications that promote interoperability, (found in the WS-I Basic Profile including the following specifications, listed here without version numbers or section references):
  - a. Simple Object Access Protocol (SOAP).
  - b. Hyper Text Transport Protocol (HTTP).
  - c. HTTP State Management Mechanism.
  - d. WS- Addressing.
  - e. SOAP Message Transmission Optimization Mechanism (MTOM).
  - f. eXtensible Markup Language (XML).
  - g. XML Binary Optimized Packaging.
  - h. SOAP Binding for MTOM.
  - i. WS-I Attachments Profile.
  - j. Namespaces in XML.
  - k. XML Schema.
  - l. Web Services Description Language (WSDL).
  - m. SOAP Request Optional Response HTTP Binding.
  - n. WSDL Binding Extension for SOAP.
  - o. Universal Description Discovery and Integration (UDDI).
  - p. HTTP Over TLS.
  - q. The Transport Layer Security (TLS) Protocol.
  - r. The Secure Socket Layer (SSL) Protocol.
  - s. SOAP with Attachments (SwA).
  - t. Internet X.509 Public Key Infrastructure Certificates and CRL Profile.
  - u. Efficient XML Interchange (EXI).
5. The current [SIBP] instance and each included specification version shall be documented in the [SWIM SvSD].
6. Updates to the [SIBP] shall be associated with updated instances of the WS-I Basic Profile, with consideration given to:
  - a. The scope of the changes in the update.
  - b. The effect of the update on existing providers and consumers.

- c. Anticipated future compliance with the updated profile instance.
  - d. General industry adoption of the updated profile instance.
  - e. The availability of testing tools that conform to the profile instance.
7. Service Provider requests for exceptions to or deviations from the [SIBP] shall be evaluated by the SWIM PMO with the following considerations:
  - a. Anticipated future compliance with the [SIBP], i.e. how long the deviation or exception will remain in effect.
  - b. The technical effect of the exception or deviation on current and potential consumers.
  - c. Cost/Benefit analysis of the exception or deviation, including costs and benefits incurred by the provider, consumers, and SWIM Configuration Management oversight.
8. [SIBP] compliance testing shall be completed by FAA Programs in accordance with the SWIM WS-I Compliance Process.
9. Services shall use one of the following transports: HTTP, HTTPS, or JMS. Service providers may be granted an exception for an alternate transport for data delivery such as Object Management Group's (OMG) Data-Distribution Service (DDS) or File Transfer Protocol (FTP).
10. Web services implemented with XML/HTTP, SOAP/JMS, and XML/JMS shall comply with the SWIM Interoperability Basic Profile with respect to the use of SOAP, HTTP, and XML.
11. The SWIM preferred data exchange format shall be XML when technically feasible. SOAP with Attachments, SOAP with MTOM, or JMS shall be used for exchanging binary data.
12. Web services implemented with SOAP/JMS and XML/JMS shall use the SWIM PMO-approved implementation of JMS.
13. JMS message types shall be TextMessage for XML data or ByteMessage for binary data. To prevent undesirable language coupling, serialized objects should not be passed using the ObjectMessage type.
14. If XML payloads are to be exchanged, SOAP messaging shall be used whenever possible to maximize interoperability.
15. The XPath and XQuery standards shall be used for querying XML data via web services.

16. Services shall use Java Management Extensions (JMX) for managing and monitoring purposes.
17. Proposed services that duplicate functionality of operational services shall be approved for development only if the SWIM CRB finds a compelling reason to do so. For example, consideration may include but is not limited to the following:
  - a. There is significant enhancement of Quality of Service factors in the proposed service.
  - b. The operational service is nearing the end of its life cycle.
  - c. Current consumers of the operational service can migrate to the new service with minimal effort.
18. When considering a service for inclusion in the NAS portfolio of services, the potential for reuse by additional consumers shall be a primary consideration
19. Services shall be designed according to a *technical service contract* and a negotiated *Service Level Agreement (SLA)* which together comprise the *service contract*.

## 3.5 NAS Service Registry/Repository Policies

### 3.5.1 Overview

NAS Services Registry/Repository (NSRR) Policies govern both the use of the NSRR and the management of its contents. The Registry/Repository incorporates registry and repository functionality to provide human-readable and machine-processable metadata and artifacts enabling service discovery and consumption. Governance is necessary to ensure the integrity, consistency and completeness of this service meta-information.

The exposure of service meta-information in the NSRR facilitates its design-time discovery and access by authorized consumers. SWIM service discovery is assumed to occur at design-time.

### 3.5.2 Policy Statements

The following policy statements apply for NSRR governance:

1. The NSRR shall be the design-time system of record for all consumable services offered to SWIM service consumers and other qualified parties.
2. Categorization schemes shall be established for NAS service meta-information contained in the NSRR that enables its discovery, compliant with [STD-064] and [NSRR SDM].

3. Published meta-information about a service shall conform to [STD-064], [STD-065] and [NSRR SDM].
4. FAA programs shall not publish their *service contract* through any mechanism other than the NSRR. Those mechanisms prohibited include, but are not limited to, direct publish using HTTP from the service-provider site.
5. The SWIM PMO shall be the administrator of the Registry/Repository.
6. All NSRR users shall be approved by the SWIM PMO before an account is created.
7. There shall be one well known URL on the FAA administrative network for NSRR access.
8. Only authorized service providers shall publish NAS service meta-information to the NSRR.
9. NAS service meta-information shall be approved by the SWIM governance lead before its publication in the NSRR.
10. Recommendations and change requests regarding NSRR structure and architecture shall be collated by the NSRR administrator and elevated to the SWIM CRB for consideration and implementation.
11. Disputes among service providers and consumers shall be elevated to the SWIM Governance Lead who shall resolve the dispute and delegate accordingly. In the event the dispute involves issues outside the scope of SWIM, the SWIM Governance Lead shall facilitate bringing the issue to the SWIM Program Manager for escalation.
12. NAS service meta-information shall be entered into the NSRR at the appropriate lifecycle milestones in accordance with both the [NSRR SDM] and SWIM service registration process.
13. Discovery of a NAS service and its associated meta-information in the NSRR shall be limited to authorized NSRR users.
14. The Service Provider, as determined by the SWIM Governance Process, has the right to restrict access to the Service and associated meta-information in the NSRR.
15. Consumers shall provide suitable identification and contact information (both machine and human-readable) to the NSRR administrator for each registered service they are consuming.
16. All NAS services shall have an associated technical point of contact. The technical point of contact manages the technical details of the service, such as,

but not limited to, the inability to use the service, data format, and other queries of a technical nature.

17. All NAS services shall have an associated administrative point of contact. The administrative point of contact manages the administrative details of the service; for example: access requests, service registration and instances of QoS failure.
18. Custody or ownership transfer of any operational service shall be negotiated and coordinated by SWIM Governance as part of the SWIM Governance Process.
19. Authorized NSRR users shall be able to access meta-information for previous versions of a service.

## 4 Service Design-Time Policies

Service design-time refers to the context in which a service (or service version) is under development prior to its availability for consumption. This section presents SWIM policies for service design-time.

### 4.1 Namespace and Schema Policies

#### 4.1.1 Purpose

SWIM Namespace and Schema Policies provide guidance to service designers involved with namespace and schema design and construction. In general, existing namespaces within the FAA should be incorporated wherever possible. Namespaces currently approved by the FAA Data Governance Board (FDGB) include:

us:gov:dot:faa:aim  
us:gov:dot:faa:atm  
us:gov:dot:faa:avs  
us:gov:dot:faa:swim

Additional guidance in the development of XML schemas can be found on the SWIM Wiki (<https://swimwiki.tc.faa.gov>).

#### 4.1.2 Policy Statements

1. Unique XML namespaces shall be registered for each service provider in accordance with [STD-063]. All XML documents for a Service Provider shall use the unique Namespace registered for that service provider.
2. Programs shall re-use an existing namespace whenever possible.

3. Any conflicts between programs over namespaces shall be resolved by the FAA Data Governance Board's (FDGB) Business Information Steward.
4. XML documents shall be specified using XML schemas and the XML Schema Definition (XSD) shall comply with the XML Schema Language. Use of Document Type Definitions (DTDs) or sample XML is not acceptable.
5. All XSD schemas shall be validated with a static analysis tool that does not allow content modification, as identified in the SWIM COTS Product Repository (SCPR).
6. The use of wild-cards, unstructured, or CDATA in schemas should be avoided.
7. Types shall be specified for all schema constructs.

## 4.2 Services Interface Design Policies

### 4.2.1 Purpose

SWIM Services Interface Design Policies address the design of the service interface. An interface specifies the functionality provided by the service and the means for interacting with the service. The SWIM program uses the Web Services Definition Language (WSDL), XML Schema, and WS-Policy to define service interfaces.

A WSDL document alone cannot describe all aspects of a service, and additional documents and meta-information are used to define a service interface, including XML schemas and WS-Policy documents (if any). This combination of WSDL document, XML schemas, and WS-Policy documents used to describe the service interface is often referred to as the *technical services contract*. The technical web services contract with the associated Service Level Agreement (SLA) are collectively referred to as the *service contract*. Universal Description Discovery and Integration (UDDI) registry metadata provides additional service description but is not considered to be part of the formal service contract, although it is a necessary part of the service description.

These policies focus on the aspects of the service interface defined by the WSDL and associated XML schemas. Additional guidance can be found on the SWIM Wiki.

### 4.2.2 Policy Statements

1. Services implementation shall be loosely-coupled to the service interface.
2. The service interface is the sole entry point into service logic and resources. Services shall be accessed only via the exposed, published interfaces.
3. All service interfaces shall be defined using a *technical service contract* that includes a WSDL definition, one or more XML schema definitions, and WS-Policy definitions as required.

4. FAA Programs shall provide a draft WSDL consisting of at least the abstract portion of the WSDL prior to development of the underlying service implementation. The full service contract shall be specified prior to the *in-service decision*.
5. Services shall have an interface that expresses a well-defined functional boundary that does not overlap with other services.
6. Each service shall accept a single XML document as an input and return a single XML document. The input and output messages will be validated against a schema at design-time that represents the data required to complete the business function. In the situations where encoding certain data types in XML is infeasible or impractical, the service provider may return data in other more efficient formats using an exception process.
7. The message schema for web services shall reside in the NSRR in the XML schema associated with the WSDL, and shall not reside in the method signature (WSDL) of the service.
8. FAA programs shall design services so they can be monitored to determine whether services become unavailable.
9. FAA programs shall design services so they can be monitored to determine whether a service has a detectable security fault.
10. FAA programs shall design services so they can be monitored to determine whether factors specified in the SLA portion of the Service Contract are out of the permitted range, including but not limited to, resource utilization and the fault behaviors and performance metrics identified in the NSRR taxonomy.
11. The service contract shall contain agreed on functional requirements. Contractual functional requirements should be derived from the appropriate service requirements document.
12. The Service Contract shall be approved by the SWIM Configuration Review Board.
13. The service contract shall contain agreed upon functional and non-functional requirements. These non-functional requirements shall include, but are not limited to:
  - Security Constraints
  - Quality of Service
  - Service Level Agreement
  - Service semantics

## 4.3 Services Technical and Design Policies

### 4.3.1 Purpose

SWIM Services Technical and Design Policies promote service design characteristics that enable interoperability and reuse. Among these design characteristics are loose coupling, abstraction, autonomy, statelessness, granularity, composability, scalability and reliability. They may be achieved by means of adopting technical standards, applying common design principles and patterns, and utilizing common software platforms.

In the area of technical standards, the SWIM Program has refined the Web Services Interoperability (WS-I) Organization profiles defining interoperability standards between web-based services and clients. A SWIM Test Assertion Document (TAD) will be available for use with the Actional® Diagnostics test tool/suite from Progress Software Corporation for compliance testing. Links to the SWIM WS-I documents and the base standards appear on the documents page of this website.

In the area of common software platforms, the SWIM Program has selected the FUSE™ product line from Progress Software Corporation to be the common software platform for all SWIM service endpoints. FUSE products are certified releases of open source software from the Apache Software Foundation, including Apache ServiceMix, Apache ActiveMQ, Apache CXF and Apache Camel.

In addition, design guidelines and best practices may be found on the SWIM Wiki.

### 4.3.2 Policy Statements

1. Static (e.g. hard coded) service addresses shall not be used. Dynamic addressing is preferred for the purpose of location transparency and failover.
2. The service logic exposed by the service shall handle concurrent access without deadlock or loss of data integrity.
3. Services shall be designed in a loosely-coupled fashion.
4. Service design shall comply with the [SWIM SvSD].
5. Services shall be implemented in a manner that does not require consumers to use a specific language (e.g. Java only) to access the service.
6. Services shall be categorized according to the SWIM Taxonomy described in the [NSRR SDM], so that they may be appropriately registered.
7. In the event of exceptions, services shall provide fault content to the consumer and the audit log, without compromising security, which shall include sufficient information for consumer recovery.
8. Code used to implement services shall be written with the ability to refactor as needs dictate. This shall include, but not be limited to, refactoring to allow for

a greater level of abstraction, refactoring to achieve more modularity, and refactoring to relocate code segments, modules and methods.

9. Services shall support at least one of the WSDL message exchange patterns in accordance with WSDL standards in [SIBP].
10. SWIM services shall implement SOAP binding extensions that are compatible with the WSDL message exchange patterns in use.

## 4.4 Information Security Policies

### 4.4.1 Purpose

The FAA maintains information and information systems that support the agency, aviation safety and security, and the National Airspace System (NAS). Information security is necessary for proper operation of FAA information systems. SWIM Information Security Policies ensure FAA program conformance with SWIM-related security requirements and guidance, including FAA Order 1370.82A and the Federal Information Security Management Act of 2002 (FISMA) [ref. NIST Special Publication 800-95, Guide to Secure Web Services].

SWIM is based on a Service Oriented Architecture (SOA) and includes service design- and run-time security controls to enforce security policies at the service and message level, including providing authorization-based access to data and services. The SWIM security functional architecture includes functions that enforce security policies at the NAS service level and the core SWIM services level including authentication, authorization, and access controls. It includes security controls that span the overall information sharing architecture that is envisioned for SWIM SOA-based services.

### 4.4.2 Policy Statements

1. FAA programs shall implement security consistent with NIST Special Publication 800-95 Guide to Secure Web Services [NIST800-95].
2. All web services implemented with SOAP/HTTP by FAA programs shall implement security compliant with a SWIM Program modified WS-I Basic Security Profile named SWIM Interoperability Basic Security Profile [SIBSP].
3. The SWIM Interoperability Basic Security Profile [SIBSP] shall tailor the clarifications, refinements, interpretations and amplifications that promote interoperability found in the WS-I Basic Security Profile including the following specifications, listed here without version numbers or section references:
  - a. HTTP over TLS.
  - b. The Transport Layer Security (TLS) Protocol.

- c. The Secure Socket Layer (SSL) Protocol.
  - d. Web Services Security.
  - e. Simple SOAP Binding Profile.
  - f. XML Signature Syntax and Processing.
  - g. XML Encryption Syntax and Processing.
  - h. Internet X.509 Public Key Infrastructure Certificates and CRL Profile.
  - i. Attachments Profile.
  - j. ITU-T X.509 CORR 1 Information technology – Open Systems Interconnection – The Directory: Public-key and attribute certificate frameworks Technical Corrigendum 1.
4. Web services security implemented for XML/HTTP, SOAP/JMS, and XML/JMS shall comply with the SWIM Interoperability Basic Security Profile with respect to their use of SOAP, HTTP, XML, WS-Security, TLS, and SSL.
  5. Web services implemented with SOAP/JMS and XML/JMS shall use the appropriate security controls available in the SWIM program-approved implementation of JMS.
  6. FAA programs shall provide to the SWIM program the security portion of their design documentations for review by the SWIM program.
  7. XML gateways used by FAA programs for security controls shall be one of those specified on the SWIM XML Gateway Qualified Vendor List (QVL) [XMLGWQVL].
  8. Extensible Access Control Markup Language (XACML) shall be used by FAA Programs beyond SWIM Segment 1 as the access control policy language and XML processing model for policy interpretation.
  9. Security Assertion Markup Language (SAML) shall be used by FAA Programs beyond Segment 1 for describing and exchanging authentication, attribute, and authorization information between security domains.

## **4.5 Service Development Process Policies**

### **4.5.1 Purpose**

SWIM Service Development Process Policies guide the exchange of information between the implementing program and the SWIM PMO to ensure adherence to SOA design principles applicable during the development stage of the SWIM service lifecycle. This includes plans for version management, quality assurance and testing, and demonstration of requirements traceability.

## 4.5.2 Policies

1. FAA Programs shall ensure service requirements traceability to [SWIM SvSD].
2. FAA Programs shall validate that upgrades or changes to existing applications or systems comply with SWIM Version Management Process [SWIM VMP].
3. The SWIM Program Management Office and FAA programs implementing SWIM-compliant services shall execute their service requirements management activity in accordance with the SWIM Requirements Management Plan [SWIM RMP].
4. The SWIM PMO and FAA programs implementing SWIM-compliant services shall execute quality assurance activities in accordance with the SWIM Quality Assurance Plan [SWIM QAP].
5. Programs shall execute QA activities using existing program resources and provide information to the SWIM QA plan per the Program Level Agreement.
6. The SWIM PMO and FAA Programs implementing SWIM-compliant services shall execute safety activities in accordance with the SWIM Program Safety Plan (PSP) [SWIM PSP].
7. Services shall verify compliance with requirements in accordance with the Verification section of the SWIM System Engineering Management Plan [SWIM SEMP].
8. FAA programs shall install, configure, and test all infrastructure components required for user acceptance, quality assurance and performance verification environments.
9. FAA programs shall successfully execute user acceptance and performance verification for all services.
10. FAA programs shall successfully execute security verification on services.
11. When a SWIM-compliant service is created or updated by a major version, the service provider shall create a model of consumption that captures all likely consumers.
12. Based on this model, the service provider shall submit in writing to the SWIM PMO a report that contains a set of likely or foreseeable impacts to participants in the SWIM infrastructure. The report shall include expected deltas in resource requirements caused by the changes, and a probability score of the occurrence of the impact itself.
13. All service components shall have fully documented and implemented unit test cases.

14. All unit test cases involving the service interface shall succeed before moving to the in-service decision.
15. The service provider should provide a reference consumer application to prospective consumers via the NSRR that exercises all features of the service interface.
16. The service provider shall perform a test which exercises all features of the service interface. The test description, procedure and results shall be stored in the NSRR.

## **4.6 Service Lifecycle Management Policies**

### **4.6.1 Purpose**

SWIM Service Lifecycle Management Policies provide guidelines to service developers in order to track progress through all of the service lifecycle stages – proposed, development, operational, deprecated and retired. This includes the categorization and tracking of multiple versions of services as they are modified and upgraded.

### **4.6.2 Policy Statements**

1. SWIM services shall be categorized according to service lifecycle stage as follows: Proposed, Development, Operational, Deprecated, and Retired.
2. All SWIM Proposed, Development, Operational, Deprecated, and Retired services and associated information shall be accessible via the NSRR.
3. Programs shall comply with the SWIM Version Management Process [SWIM VMP], which provides guidelines for version numbering and the management of major, minor, and revision releases for all transitions in the service development lifecycle.
4. There shall be no more than one version of a service identified as ‘Operational’ in the NSRR at any point in time. When an updated version of a service is promoted to the ‘Operational’ state, the existing version shall automatically be transitioned to the ‘Deprecated’ state.
5. The SWIM CRB shall approve major, minor and revision version releases of services throughout the service development lifecycle and placed under the appropriate level of configuration control as specified by the SWIM Configuration Management Plan [SWIM CM].
6. Service Providers shall conduct an analysis of impact on current consumers for review and approval by the SWIM CRB prior to scheduling a version of a Service for Retirement. .

7. Service Updates which require a change to the Technical Service Contract, and also require changes by consumers (i.e. not backward-compatible) shall be categorized as Major updates.
8. Service Updates which require a change to the Technical Service Contract, but do not require changes by any known consumers (i.e. backward-compatible) shall be categorized as Minor updates.
9. Service Updates which do not require a change to the Technical Service Contract (i.e. bug fixes, or Quality of Service enhancements) shall be categorized as Revision updates.
10. Programs shall ensure that the NSRR contains information on all inter-dependencies between service producers and consumers to ensure that a complete impact analysis can be performed.
11. The SOA suitability analysis process will be used to identify and prioritize retirement, replacement, rationalization, refactoring or service-enablement of legacy systems.
12. FAA Programs implementing SOA services shall publish their service roadmap to enable legacy service consumers to efficiently plan transition to the planned SOA-based service.

## **4.7 Service Operations Readiness Policies**

### **4.7.1 Purpose**

SWIM Service Operations Readiness Policies support the processes for reviewing and assessing the operational readiness of a service prior to releasing it into production, publishing its meta-information to the NSRR, or deploying it.

### **4.7.2 Policy Statements**

1. A decision from a competent reviewing authority shall be recorded for each service designated as “Operational” in the SWIM Service Registry/Repository. Decisions and the reviewing authority are recorded in the NSRR by the SWIM Governance Lead, and may include: (1) an In-Service Decision (ISD) by a designated ISD authority and (2) a Configuration Control Decision (CCD) by the NAS Configuration Control Board (CCB).
2. Operational support processes for services shall be fully defined and documented. The definitions shall be reviewed and approved by the SWIM Governance Lead prior to service introduction.
3. Operational reports and the criteria for their creation and distribution shall be defined, documented, and tested. The definitions shall be reviewed and approved by the SWIM Governance Lead before service introduction.

## 4.8 Service Provisioning Policies

### 4.8.1 Purpose

SWIM Service Provisioning Policies provide specific guidelines to FAA programs supporting the processes of provisioning services and monitoring service consumption and usage by user account.

### 4.8.2 Policy Statements

1. Provisioning for services to authorized users shall consist of offer, request, negotiation and approval workflows for service access, security, capacity, SLAs and policy contracts.
2. Administrators with an enterprise-wide scope of authority or delegated administrators with limited scope of authority shall be able to update user access to the NSRR and multiple managed services from a single system administration user account.
3. Requests to provision for new access, to change existing user profiles or to deactivate consumers shall be routed to those with suitable authority for approval or disapproval.
4. Once a change to a user profile has been approved, the applicable administrator shall update that profile.

## 4.9 Service Consumer Policies

### 4.9.1 Purpose

SWIM Service Consumer Policies support the processes of negotiating contracts for the utilization of SWIM services, developing service clients, and consuming services in accordance with the applicable contracts.

### 4.9.2 Policy Statements

1. SWIM service consumers shall consume services offered by a service provider in accordance with the *service contract*. The SLA component of the service contract shall be negotiated with the service provider. Hence, there shall be no anonymous consumers. The SLA need not be unique to the service consumer and indeed it is preferable that it not be unique.
2. If required by the service provider, service consumers shall be approved by the service provider.
3. The service consumer shall use an emulated service for integration testing if offered by the service provider.

4. Consumers shall remain aware of the service lifecycle stage of consumed services. Consumers of Deprecated Services for which there is a new Operational version shall migrate to the Operational version within the time periods set forth in [SWIM VMP].

## **5 Runtime and Operational Policies**

Runtime refers to the context in which a service (or service version) is available for consumption, regardless of its lifecycle stage. Runtime and operational policies address non-functional requirements such as access, security, and performance.

### **5.1 Messaging and Routing Policies**

#### **5.1.1 Purpose**

SWIM Messaging and Routing Policies provide for failover, load-balancing, delivery guarantees, prioritized delivery, synchronous and asynchronous delivery, and transformations and translations of the transmitted information.

#### **5.1.2 Policy Statements**

1. The SWIM service provider/consumer shall process data stream address faults pursuant to the W3C WS-Addressing and Organization for the Advancement of Structured Information Standards (OASIS) WS-Reliable Messaging specifications.
2. The server-to-client data-stream messaging service shall provide for a server/client failover mechanism to recover messages and to reconnect to another available message broker as deemed necessary.

### **5.2 Runtime Security Policies**

#### **5.2.1 Purpose**

The FAA maintains information and information systems that support the agency, aviation safety and security, and the National Airspace System (NAS). SWIM Runtime Security Policies address runtime considerations – such as password creation and reset, security credential distribution, configuration of system elements, etc. – that are not explicitly addressed in Information Security. Runtime Security Policies ensure FAA program conformance with SWIM-related security requirements and guidance, including FAA Order 1370.82A and the Federal Information Security Management Act of 2002 (FISMA) [ref. NIST Special Publication 800-95, Guide to Secure Web Services].

#### **5.2.2 Policies**

1. FAA programs shall implement runtime security consistent with NIST Special Publication 800-95 Guide to Secure Web Services [NIST800-95].

## 5.3 Service Management Policies

### 5.3.1 Purpose

SWIM Service Management Policies provide for common management of FAA program SWIM-compliant services for availability, security faults, and SLA conformance.

The NAS SOA environment consists of heterogeneous but interconnected systems that are governed or managed according to different policies, rules, or principles that meet individual FAA program information management needs. This interconnected network of systems requires agreement and mediation as to which policies will govern information as it flows among systems.

### 5.3.2 Policy Statements

1. FAA programs shall monitor their SWIM-compliant services to determine whether services become unavailable.
2. FAA programs shall monitor their SWIM-compliant services to determine whether a service experiences a detectable security fault.
3. FAA programs shall monitor their SWIM-compliant services to determine if factors specified in Service Level Agreements (SLAs) are out of the permitted range, including but not limited to resource utilization, and the fault behaviors and performance metrics identified in the NSRR taxonomy.
4. FAA programs shall comply with the monitoring and control policy specified in the SWIM Product Standardization [SWIMPSTD] document.
5. Service providers shall be responsible for ensuring that services are monitored.
6. All abnormal conditions that cannot be corrected automatically shall send an alert through the enterprise service management infrastructure, allowing operational support teams to correct the problem in a timely manner. This policy is not implemented in SWIM Segment 1, but service providers should be aware that it will be in effect in Segment 2 and beyond, and should plan accordingly.
7. All alerts that may be sent to the enterprise service management infrastructure shall have documented escalation procedures and if possible, the process to address the abnormal condition. This policy is not implemented in SWIM Segment 1, but service providers should be aware that it will be in effect in Segment 2 and beyond, and they should plan accordingly.

## **5.4 Maintenance and Support Policies**

### **5.4.1 Purpose**

The SWIM maintenance and support policies ensure that services are maintained and supported to remain in compliance with their service contracts.

### **5.4.2 Policies**

1. Programs shall monitor actual service consumption and usage based on service metrics such as availability, capacity, service latency, and scheduled downtime in accordance with FAA-STD-065.
2. Programs shall support the operation of their service(s) to provide the performance specified in the service contract.
3. Maintenance updates of services shall adhere to the [SWIM VMP].

## 6 Glossary

The following definitions are assumed in this document.

<b>Administrator</b>	A person responsible for maintaining and administering the <i>SWIM Enterprise Services Registry/Repository</i> . The <i>SWIM Administrator</i> .
<b>artifact</b>	A <i>repository</i> item attached to a <i>service</i> that describes the service or may be used to access the service, such as a WSDL file, an XML schema file, sample code or an IRD. Artifacts are one of the two <i>meta-information</i> types (the other is <i>metadata</i> ).
<b>authentication</b>	The process of verifying an identity claimed by or for a system entity. [RFC 2828]
<b>authorization</b>	Permission to engage in a specific activity. A Registry/Repository user must be authorized to access <i>meta-information</i> contained in the <i>Registry/Repository</i> .
<b>Browsing</b>	Viewing the contents of the <i>Registry/Repository</i> guided by its <i>taxonomic</i> structure, much like browsing a website; one of the two ways to navigate a <i>registry</i> (the other is <i>querying</i> ).
<b>CDATA</b>	A CDATA section starts with “<![CDATA[“ and ends with “]”>. Everything inside a CDATA section is ignored by the parser.
<b>client</b>	A service client.
<b>Consumer</b>	A person that consumes a service; the <i>Service Consumer</i> .
<b>deregistration</b>	The act of deleting all <i>meta-information</i> for a <i>NAS service</i> from the <i>Registry/Repository</i> .
<b>Designated Data Authority (DDA)</b>	A senior FAA management official, appointed in writing by a Management Team member, who is responsible for the Data Management Program within their organization. Each DDA is a permanent member of the FAA Data Governance Board. [FAA Order 1375.1D] [FAA-STD-063]
<b>design-time</b>	The pre-runtime mode of a <i>service</i> . Design-time considerations for a service include software development, testing and publication.
<b>developer</b>	A person responsible for developing a <i>service</i> ; in the context of the <i>Registry/Repository</i> , a developer could be a <i>consumer</i> , a <i>publisher</i> or both.
<b>discoverable</b>	A characteristic of a <i>service</i> where its interface features may be discovered by a potential user.

<b>discovery</b>	The act of locating and accessing the <i>meta-information</i> for a <i>service</i> .
<b>Enterprise Service Management (ESM)</b>	ESM provides the ability to monitor, manage, and scale services within the enterprise to ensure the capability offerings are available, responsive and scalable to the operational environment supported.
<b>expose</b>	To make <i>service meta-information discoverable</i> . In SWIM, services are exposed via the <i>Registry/Repository</i> .
<b>governance</b>	SWIM SOA governance is characterized by the people, policies, and processes required for leading, communicating, guiding, and enforcing the organizational behaviors needed to produce the desired enterprise SOA outcome; supporting governance is one of the primary objectives of the NSRR.
<b>Governance lead</b>	An abstraction of the set of FAA person(s) responsible for defining and enforcing the policy rules that govern activities in the SWIM environment.
<b>group</b>	A logical grouping of <i>users</i> , such as FAA, non-FAA, airline, DoD, etc.
<b>metadata</b>	Data that defines or describes other data [ISO 11179]. A <i>registry</i> item defined by <i>taxonomy</i> whose usage is defined by policies. One of the two types of <i>NAS service meta-information</i> (the other is <i>artifact</i> ), metadata describes NAS services and the organizations that offer them.
<b>meta-information</b>	Information describing a <i>NAS service</i> , including <i>metadata</i> and <i>artifacts</i> .
<b>NAS service</b>	A <i>service</i> within the NAS that encapsulates a distinct set of operation logic within a well-defined functional boundary. A NAS service provides <i>Service Consumers</i> access to one or more NAS applications or systems by means of the SWIM core services.
<b>notification</b>	An indication presented to a user regarding the status of a system or an element in a system.
<b>publish</b>	The act of <i>exposing service meta-information</i> to <i>users</i> in the NSRR. This is done by the <i>SWIM Administrator</i> .
<b>query</b>	A directed request for specific information done in a request-response manner; one of the two ways to navigate a registry (the other is <i>browsing</i> ).
<b>refactor</b>	Improving a part of a computer program without affecting the parts of the program you want to leave unchanged.

<b>register</b>	The act of <i>publishing service meta-information</i> . A registered service has had its meta-information published. This is done by the <i>SWIM Administrator</i> .
<b>Registry /Repository</b>	The <i>SWIM Enterprise Services Registry/Repository</i> .
<b>registry</b>	An enabling infrastructure that uses a formal registration process to store, catalog and manage <i>service metadata</i> . A registry supports search and query capabilities for and understanding of <i>resources</i> .
<b>repository</b>	A collection of <i>resources</i> accessible over an internet. It contains <i>artifacts</i> such as technical reference documents, reports, policy descriptions, WSDL files, XML schema files, etc.
<b>resource</b>	An object of information that is available on an internet and identified by a unique <i>Uniform Resource Identifier</i> .
<b>role</b>	A mode of <i>user</i> interaction with the <i>NSRR</i> . Registry/Repository user roles are <i>publisher, consumer, administrator</i> and <i>governance lead</i> . Registry/Repository users may have more than one role.
<b>runtime</b>	The executing mode of a <i>service</i> . A service has both runtime and <i>design-time</i> modes.
<b>security</b>	The SWIM core service responsible for the protection of information, operation, assets and participants from unauthorized access or attack.
<b>service</b>	An implementation-independent reusable operational function that may be discovered as self-describing interfaces, and invoked using open standard protocols across networks. Services can be combined and orchestrated to produce composite services and operations processes, in accordance with predefined policies, security and SLAs.
<b>service binding</b>	Specifies concrete message format and transmission protocol details for an interface, and must supply such details for every operation and fault in the interface.
<b>service client</b>	An executing software entity that interacts with a service.
<b>Service Consumer</b>	Anyone authorized to access a SWIM-based service.

<b>service contract</b>	A service contract is comprised of one or more published documents that express meta-information about a service. The service contract consists of the Service Level Agreement (SLA) in addition to the <i>technical service contract</i> which includes WSDL, XML schema, and WS-Policy definitions. [adapted from Thomas Erl, et. al., 2008, Web Service Contract Design and Versioning for SOA, Prentice Hall]
<b>service deregistration</b>	The act of deleting an entry from the <i>Registry/Repository</i> .
<b>service level agreement</b>	The Service Level Agreement (SLA) is a human-readable document describing non-functional service features such as Quality of Service (QoS), behaviors, or limitations. Some SLA-related requirements may also be expressed as machine-readable WS-Policy definitions. [adapted from Thomas Erl, et. al., 2008, Web Service Contract Design and Versioning for SOA, Prentice Hall]
<b>service lifecycle</b>	The life stages of a SWIM service which consist of proposed, development, operational, deprecated, and retirement.
<b>Service Oriented Architecture (SOA)</b>	Policies, practices and frameworks that enable application functionality to be provided and consumed as services that can be invoked, published and discovered, and are abstracted away from implementation using a single, standards-based form of interface. [SWIM Glossary]
<b>Service Provider</b>	Any FAA organization providing a SWIM-based service.
<b>service registration</b>	The act of <i>exposing</i> a <i>NAS</i> service by <i>publishing</i> its <i>meta-information</i> in the <i>Registry/Repository</i> . Same as <i>registration</i> .
<b>submit</b>	To offer service meta-information for publication in the <i>Registry/Repository</i> . Submission is done by <i>Service Providers</i> . Submitted meta-information that has been approved is subsequently <i>published</i> by the <i>SWIM Administrator</i> .
<b>SWIM</b>	A service-oriented environment for implementing and operating NAS software-based systems that enables information-sharing.
<b>SWIM Administrator</b>	The FAA person(s) responsible for administering SWIM infrastructure elements, including the NSRR.

<b>SWIM core services</b>	The fundamental SWIM mechanisms that enable information sharing: Interface Management, Messaging, Enterprise Service Management (ESM) and Security. These services are solution-agnostic (not limited to a single process or solution environment) and have a high degree of autonomy so that they support reuse within the NAS. Also referred to as “core services.”
<b>SWIM Governance Lead</b>	See “Governance lead” above.
<b>SWIM- implementing program (SIP)</b>	A program responsible for one or more applications or systems participating in the SWIM environment.
<b>SWIM NAS Services Registry /Repository</b>	A logical system consisting of one or more instances of commercial service registry/repository products providing a vehicle for <i>discovery</i> of and access to <i>NAS services</i> . The <i>SWIM NAS Services Registry/Repository</i> provides two basic functions—a <i>registry</i> and a <i>repository</i> —that work together in a unified manner.
<b>SWIM Security Authority</b>	The FAA person(s) responsible for establishing and enforcing policies that maintain a secure SWIM environment.
<b>technical service contract</b>	The fundamental part of a service contract consisting of documents expressing its technical interface of the service, essentially establishing a technology-independent API into the functionality offered by the service. The most common service description documents are the WSDL definition, with may link multiple XML schema and WS-Policy definitions. [adapted from Thomas Erl, et. al., 2008, Web Service Contract Design and Versioning for SOA, Prentice Hall]
<b>Uniform Resource Identifier (URI)</b>	A compact string of characters for identifying an abstract or physical resource. [RFC 2396]
<b>User</b>	Person that accesses the SWIM Enterprise Services Registry/Repository. A user has one or more <i>roles</i> (publisher, consumer, administrator or governance lead) and may be a member of one or more <i>groups</i> (FAA, non-FAA, etc.).

## 7 Acronyms

AMS	Acquisition Management System
CCB	Configuration Control Board
CCD	configuration control decision
COTS	Commercial Off The Shelf
CRB	Configuration Review Board
DDS	Data-Distribution Service
DTD	Document Type Definition
EA	Enterprise Architecture
EAB	Enterprise Architecture Board
EXI	Efficient XML Interchange
FAA	Federal Aviation Administration
FDGB	FAA Data Governance Board
FTP	File Transfer Protocol
FISMA	Federal Information Security Management Act
HTTP	Hyper Text Transport Protocol
HTTPS	Hyper Text Transfer Protocol Secure
ISD	In-Service Decision
JBI	Java Business Integration
JMX	Java Management Extensions
JMS	Java Message Service
MTOM	Message Transmission Optimization Mechanism
NAS	National Airspace System
NIST	National Institute of Standards and Technology
NSRR	NAS Services Registry/Repository
OASIS	Organization for the Advancement of Structured Information Standards
OSGI	Open Services Gateway Initiative
PMO	Program Management Office
QA	Quality Assurance
QoS	Quality of Service
QVL	Qualified Vendor List
SCPR	SWIM COTS Product Repository

SOA	Service-Oriented Architecture
SOAP	Simple Object Access Protocol
SSL	Secure Socket Layer
SwA	SOAP with Attachments
SWIM	System-Wide Information Management
TLS	Transport Layer Security
UDDI	Universal Description Discovery and Integration
URL	Uniform Resource Locator
WSDL	Web Services Description Language
XML	eXtensible Markup Language
XSD	XML Schema Definition