FAA-STD-074
July 20, 2020
SUPERSEDING
FAA-STD-070
July 12, 2012

U.S. Department
of Transportation

**Federal Aviation
Administration**

U.S. Department of Transportation
Federal Aviation Administration

Standard Practice

PREPARATION OF
SERVICE REQUIREMENTS DOCUMENTS

FOREWORD

This standard is approved for use by all Departments of the Federal Aviation Administration (FAA).

This standard specifies the minimum acceptable content for preparing FAA SOA-based service requirements documents (SRD).

This standard supersedes FAA-STD-070 [6], Preparation of Web Service Requirements Documents (WSRD). FAA-STD-070 was geared toward the Web Service (WS) paradigm [39]. Replacement of FAA-STD-070 became necessary when the FAA System Wide Information Management (SWIM) Program began implementing services with message-oriented interfaces supported by middleware technologies like Java Message Service (JMS). This superseding standard ensures that requirements for any SOA-based service are independent of technological solutions.

This standard has been prepared in accordance with FAA-STD-068, Department of Transportation Federal Aviation Administration, *Preparation of Standards* (December 2009) [4].

Comments, suggestions, or questions on this document shall be addressed to:

Federal Aviation Administration
System Wide Information Management (SWIM) Program Office, AJM-316
800 Independence Avenue, SW
Washington, DC 20591
https://www.faa.gov/air_traffic/technology/swim/contacts/

Table of Contents

List of Figures

List of Tables

1 SCOPE

This standard defines the content and structure of a Service Requirements Document (SRD). The SRD provides the details needed to specify requirements for a service that is part of the Federal Aviation Administration's (FAA) implementation of a service-oriented architecture (SOA).

- This standard does not prescribe or suggest any technological solutions for developing a service.
- This standard does not specify any configuration management procedures or policies to which the developed SRD may be subjected.

1.1 Introduction

Over the last two decades, SOA has become an accepted approach for realizing information exchange in the National Airspace System (NAS) and the FAA as a whole. The FAA's first major SOA effort, the System Wide Information Management (SWIM) program, was established in 2007 to enable information sharing by providing a communications infrastructure and architectural solutions for developing and operating a network of highly-distributed, interoperable, and reusable services.

The first services developed by the FAA followed the Web Service (WS) paradigm. To support the development of a governed, service-oriented environment, the FAA initiated a series of standards and other guidance materials for designing, developing, and documenting services consistently and uniformly. One of these standards was FAA-STD-070, Preparation of Web Service Requirements Documents (WSRD) [6], approved in July 2012.

As the application of services in the FAA continued to expand, service developers also deployed other kinds of service-based solutions, notably the Java Message Service (JMS), an approach that uses Java message-oriented middleware, and began exploring the deployment of RESTful services in the SWIM environment. These technological advancements, together with lessons learned in the intervening years, made it necessary to replace FAA-STD-070 with a new standard for preparing a solution-independent SRD.

The purpose of this standard is to establish a uniform content and structure for specifying a set of requirements, rendered as an SRD, usually created by the FAA, which identifies the necessary provisions for developing a service, usually performed by government contractors or vendors.

1.2 Intended Audience

The intended audience for this standard includes architects, decision makers, analysts, requirements developers, and implementers responsible for developing FAA services.

1.3 Basic Concepts

This section introduces major terms, concepts, and ideas relevant to this standard.

1.3.1 Service-Oriented Architecture (SOA)

For the purpose of this standard, SOA is understood to be an architecture style that is based on loosely-coupled, interoperable, and reusable software components commonly referred to as "services". The FAA has traditionally deployed tightly-coupled systems in which all system elements (e.g., programs, databases, subsystems, etc.) are highly dependent on each other and

are generally connected through individual, point-to-point interfaces. From the FAA's perspective, the major motivation for SOA is to transform tightly-coupled systems into a set of loosely-coupled services that can be created from an existing IT infrastructure and can be used and reused across the FAA to leverage existing investments and promote interoperability.

The following is a list of major SOA characteristics and their consequent relevance to the scope of an SRD. Note: although some of the principles listed below may not be pertinent to the context of an SRD, it is important for requirement developers to be aware of major architectural concepts.

- Loosely-coupled – Dependencies between interrelating SOA components are minimized. This means that "services are designed with no affinity to any particular service consumer" [34], that is, a SOA service has no intrinsic intelligence about consumers of that service.
- Interoperable – SOA components should be able to interact across heterogeneous computational platforms, operating systems, and programming languages. This is achieved through adherence to a common set of standard protocols (mainly non-proprietary, i.e., "open").
- Autonomous – SOA components encapsulate the implementation of their own business functions and, recognizing that the other interacting "components neither know nor care how services perform their function, they merely anticipate that they return the expected result" [35].
- Reusable – SOA components are realized as autonomous and self-contained IT assets and as such can be reused by an undefined number of service consumers even in contexts often unknown at design time.
- Discoverable – SOA components should be discoverable, which means that there exist processes through which a service consumer may search for and find (i.e., discover) the service. This capability is usually provided by the organization that is responsible for the support and governance of the SOA infrastructure (e.g., SWIM).
- Self-describing – SOA components should be self-describing, which means that a service should provide a service consumer with all of the relevant information (service description) that is necessary for using the service.

1.3.2 Service

Services are primary architectural assets and central artifacts of a SOA. Every SOA-based implementation encompasses a collection of services with varied architectural roles and business uses. A *service* is defined as "a mechanism to enable access to one or more capabilities, where the access is provided using a prescribed interface and is exercised consistent with constraints and policies as specified by the service description" [5]. Although services in SOA could be implemented using various enabling technology approaches (e.g., OMG CORBA, MS .NET, Java EE, etc.); a solution that leverages usage of Web and Internet-based standards (e.g., XML, HTTP, JMS, SOAP, etc.) has become a predominant approach in implementing SOA-based solutions.

Services may be identified based on many different characteristics. The following are examples of common characteristics employed in the FAA:

- Service category, e.g., Aeronautical, Weather, Security, etc.
- Service interface, e.g., Method-Oriented, Message-Oriented, Resource-Oriented, etc.
- Service criticality level, e.g., Critical, Essential, etc.

It should be noted that this standard applies to services of all types, purposes, and roles.

1.3.3 Service Interface

The *service interface* is a fundamental, definitive component of a service. In a broad sense, the term interface means "a point of communication between two or more processes, persons, or other physical entities" [8]. In information technology, an interface "might be a hardware connector used to link to other devices, or it might be a convention used to allow communication between two software systems" [18], or a "shared boundary between two functional units, defined by various characteristics pertaining to the functions, physical interconnections, signal exchanges, and other characteristics, as appropriate" [23]. For the purpose of this standard, a service interface is simply "the means by which the underlying capabilities of a service are accessed" [25].

One of the main principles of SOA is loose-coupling among SOA components. A logical consequence of loose-coupling is that "any entity that a service may interact with may not exist at the point of time the service is developed" [27]. That means that a service interface is not a binding agreement or a shared convention between two or more components, but rather it is a description of the mechanics of the message exchange (e.g., message formats, fault processing, etc.).

1.3.3.1 Service Interface Components

This standard follows the Interface Abstract Component Model described by the W3C in the Web Service Description Language (WSDL) specification [38]. This model "defines the abstract interface of a service as a set of abstract operations, each operation representing a simple interaction between the consumer and the service. Each operation specifies the types of messages that the service can send or receive as part of that operation. Each operation also specifies a message exchange pattern that indicates the sequence in which the associated messages are to be transmitted between the parties" [37].

For the purpose of this standard, the interface definition model is asserted as follows (see also Figure 1):

"An *interface* groups together operations without any commitment to transport or wire format" [38], i.e., an interface is a "logical grouping of operations" [39]. Requirements for the interface should be presented in section 7 of an SRD.

- An *operation* is "a named set of messages related to a single service action" [39] or, more specifically, "an interaction with the service consisting of a set of (ordinary and fault) messages exchanged between the service and the other parties involved in the interaction" [22]. Note: operations may also be referred to as "methods".
- A *message* is a basic unit of communication from one software agent (a service or client) to another sent in a single logical transmission (e.g., request or response) [5]. A message may include a *header* and a payload (actual data).
  - A *header* is a part of the message that precedes the message payload; it typically contains identification and routing information [5].
  - A *payload* is the actual (business) data transferred by the message [5]. Note: the payload may also be called the "message body".
- A *fault* is a message that is returned as a result of an error that prevents the service from implementing a required function. A fault usually contains information about the cause of the error. Note: faults are often referred to as "exceptions".

- *Data* is "a re-interpretable representation of information in a formalized manner suitable for communication, interpretation, or processing [32]." Data is transmitted among SOA components as the contents of messages. Data can be presented in various forms: text, maps, images, etc. The text is usually rendered in a formal language, with XML being the most popular format in the FAA.

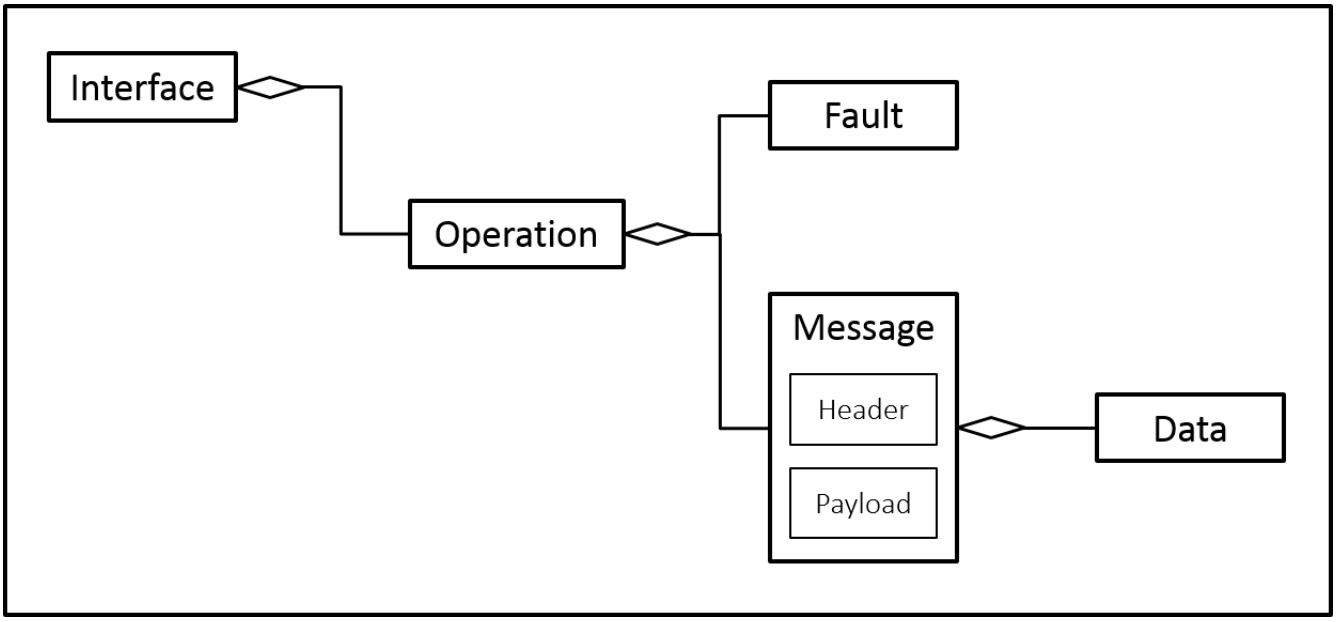

FIGURE 1. Service interface

2 APPLICABLE DOCUMENTS

2.1 Government Documents

[1] FAA Order 1000.36, FAA Writing Standards, 31 March 2003.
http://www.faa.gov/documentlibrary/media/order/branding_writing/order1000_36.pdf

[2] FAA Order 1370.121, FAA Information Security and Privacy Program & Policy, 23 December 2016.
https://www.faa.gov/regulations_policies/orders_notices/index.cfm/go/document.information/documentID/1030708

[3] FAA Order 1700.6C, FAA Branding Policy, Use of the FAA Logo, FAA Signature, and DOT Seal, 11 September 2006.
http://www.faa.gov/documentLibrary/media/order/branding_writing/Branding_Order_17006.pdf

[4] FAA-STD-068, Preparation of Standards, 4 December 2009.
http://www.tc.faa.gov/its/worldpac/standards/faa-std-068.pdf

[5] SWIM Controlled Vocabulary, March 2019.
https://semantics.aero/pages/swim-vocabulary.html

[6] FAA-STD-070, Preparation of Web Service Requirements Documents, 12 July 2012.
http://www.tc.faa.gov/its/worldpac/standards/faa-std-070.pdf

[7] Artifacts Versioning for SWIM-enabled Services, Software Specification, Version 1.0.0, 18 December 2015.
https://www.faa.gov/air_traffic/technology/swim/governance/standards/media/SWIM%20Service%20Versioning%20Spec.pdf

[8] Federal Standard 1037C, Glossary of Telecommunication Terms (now maintained as American National Standard T1.523-2001, Telecom Glossary 2000).
https://glossary.atis.org/

[9] FAA Systems Engineering Manual, Version 1.0.1, 19 June 2014.
http://everyspec.com/FAA/FAA-General/download.php?spec=FAA_SEM_V1x0_19JUN2014.052250.pdf

[10] U.S. Government Publishing Office Style Manual, 2016.
https://www.govinfo.gov/collection/gpo-style-manual?path=/gpo/U.S.%20Government%20Publishing%20Office%20Style%20Manual/2016

2.2 Non-Government Documents

[18] Free On-line Dictionary of Computing, Denis Howe, 2020.
http://foldoc.org

[19] Reference Architecture Foundation for Service Oriented Architecture Version 1.0, OASIS Committee Specification 01, 4 December 2012.
http://docs.oasis-open.org/soa-rm/soa-ra/v1.0/soa-ra.html

[20] ISO/IEC/IEEE 12207:2017, Systems and Software Engineering – Software Life Cycle Processes, November 2017.
https://www.iso.org/standard/63712.html

[21] ISO/IEC 6523-1, Structure for the Identification of Organizations and Organization Parts, 1998.
http://www.iso.org/iso/catalogue_detail?csnumber=25773

[22] ISO/IEC 7498-1, Open Systems Interconnection - Basic Reference Model, 1994.
http://www.iso.org/iso/catalogue_detail.htm?csnumber=20269

[23] ISO/IEC CD 20944-1:2013, Information Technology – Metadata Registries Interoperability and Bindings (MDR-IB) – Part 1, Framework, Common Vocabulary, and Common Provisions for Conformance, January 2013.
https://www.iso.org/standard/51914.html

[24] Glossary of Security Terms, SANS Institute, March 2020.
http://www.sans.org/resources/glossary.php

[25] OASIS Reference Model for SOA 1.0, 12 October 2006.
http://docs.oasis-open.org/soa-rm/v1.0/soa-rm.pdf

[26] OASIS Reference Ontology for Semantic Service Oriented Architectures, Public Review 1, 5 November 2008.
http://docs.oasis-open.org/semantic-ex/ro-soa/v1.0/pr01/see-rosoa-v1.0-pr01.html

[27] Position Papers for the W3C Workshop on Web Services: W3C Web Services Team: April 3, 2001.
http://www.hpl.hp.com/techreports/2001/HPL-2001-73.pdf

[28] RFC 2119, Key words for Use in RFCs to Indicate Requirement Levels, Network Working Group, March 1997.
http://www.rfc-editor.org/rfc/rfc2119.txt

[29] RFC 2828, Internet Security Glossary: Network Working Group, May 2000.
http://www.ietf.org/rfc/rfc2828.txt

[30] Java Message Service Specification Version 1.1: Sun Microsystems, Inc.: April 12, 2002
http://download.oracle.com/otndocs/jcp/7195-jms-1.1-fr-spec-oth-JSpec/

[31] RFC 3935, Mission Statement for the Internet Engineering Task Force (IETF), Network Working Group, October 2004.
http://www.ietf.org/rfc/rfc3935.txt

[32] ISO/IEC 2382:2015(en) Information technology — Vocabulary, May 2015.
https://www.iso.org/obp/ui/#iso:std:iso-iec:2382:ed-1:v1:en

[33] RFC 3986, Uniform Resource Identifier (URI): Generic Syntax, January 2005.
https://www.rfc-editor.org/rfc/rfc3986.txt

[34] SOA Definition and Principles, R. W. Schulte and Y. V. Natis, Gartner Group, 2003.
No longer available online.

[35] State of the Art in Developing Applications Using Services, Mikko Raatikainen, Helsinki University of Technology.
https://www.researchgate.net/publication/265825145_State_of_the_art_in_developing_applications_using_services_A_review_of_basic_concepts

[36] Web Services Architecture, W3C Working Group Note, 11 February 2004.
http://www.w3.org/TR/ws-arch

[37] Web Services Description Language (WSDL) Version 2.0 Part 0: Primer, W3C Recommendation, 26 June 2007.
http://www.w3.org/TR/wsdl20-primer/

[38] Web Services Description Language (WSDL) Version 2.0 Part 1: Core Language, W3C Recommendation, 26 June 2007.
http://www.w3.org/TR/wsdl20/

[39] Web Services Description Requirements, W3C Working Draft, 28 October 2002.
https://www.w3.org/TR/ws-desc-reqs/

[40] Web Services Glossary, W3C Working Group Note, 11 February 2004.
https://www.w3.org/TR/ws-gloss/

[41] Web Services Policy 1.5 – Framework, W3C Recommendation, 04 September 2007.
https://www.w3.org/TR/ws-policy/

2.3 Order of Precedence

In the event of a conflict between the text of this document and the references cited herein, the text of this document takes precedence. Nothing in this document, however, supersedes applicable laws and regulations unless a specific exemption has been obtained.

3 DEFINITIONS

3.1 Key Words

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", “MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [28]. These key words are capitalized when used to unambiguously specify requirements. When these words are not capitalized, they are meant in their natural-language sense.

All examples in the document are labeled as "non-normative", which means they are not to provide a canonical implementation for use in a registry or artifact, but merely to illustrate technical features of a particular approach.

3.2 Terms and Definitions

Access Control	Protection of system resources against unauthorized access; a process by which use of system resources is regulated according to a security policy and is permitted only by authorized entities. [29]
Asynchronous	An interaction is said to be asynchronous when the associated messages are chronologically and procedurally decoupled. [40]
Attribute-Based Access Control (ABAC)	A process in which access to system resources is granted to a user based on the value of a user’s attributes.
Audit	A process that records information needed to establish accountability for system events and for the actions of system entities that cause them. [29]
Audit Trail	A chronological record of system activities that is sufficient to enable the reconstruction and examination of the sequence of environments and activities. [29]
Authentication	The process of verifying an identity claimed by or for a system entity. [29]
Authorization	The granting of rights or permission to a system entity (mainly, but not always, a user or a group of users) to access a service.
Business Function	A characteristic action or activity that needs to be performed to achieve a desired objective, or in the context of this standard, to achieve a real world effect. (Adapted from [9])
Confidentiality	Protective measures that ensure that information is not made available or disclosed to unauthorized individuals, entities, or processes (i.e., to any unauthorized system entity). [29]
Consumer Agent	A software agent that is designed to interact with a service in order to request that a task be performed on behalf of its owner – the service consumer. [5]

Credentials	Data that is transferred to establish the claimed identity of an entity. [22]
Critical Failure	A state or condition in which a service is unable to perform the primary function for which it was designed.
Data	A reinterpretable representation of information in a formalized manner suitable for communication, interpretation, or processing. [32]
Fault	A message that is returned as a result of an error that prevents a service from implementing a required function. A fault usually contains information about the cause of the error.
Format	The arrangement of bits or characters within a group, such as a data element, message, or language.
Hyperlink	In hypertext or hypermedia, an information object (such as a word, a phrase, or an image; usually highlighted by color or underscoring) that points (indicates how to connect) to related information that is located elsewhere and can be retrieved by activating the link. [24]
Idempotent	A term used to describe an operation in which a given message will have the same effect whether it is received once or multiple times; i.e., receiving duplicates of a given message will not cause any undesirable effect.
Identifier (ID)	A sequence of characters that unambiguously indicates a particular resource. [19]
Input	Data entered into, or the process of entering data into, an information processing system or any of its parts for storage or processing. (Adapted from [23])
Integrity	Protective measures that ensure that data has not been changed, destroyed, or lost in an unauthorized or accidental manner. [29]
Interface	See Service Interface.
Internet	A large, heterogeneous collection of interconnected systems that can be used for communications of many different types between any interested parties connected to it. The term includes both the "core Internet" (Internet service provider networks) and "edge Internet" (corporate and private networks, often connected via firewalls, network address translation boxes, application layer gateways, and similar devices). [31]
Java Message Service (JMS)	A Java-based application programming interface (API) that provides a common way for Java programs to create, send, receive, and read an enterprise messaging system's messages. [5]
Loose-Coupling	A characteristic of software systems, in which dependencies among a system's constituting parts have been minimized. [5]

Message	A basic unit of communication from one software agent to another sent in a single logical transmission. [5]
Message Exchange Pattern (MEP)	A template, devoid of application semantics, that describes a generic pattern for the exchange of messages between agents. It describes the relationships (e.g., temporal, causal, sequential, etc.) of multiple messages exchanged in conformance with the pattern, as well as the normal and abnormal termination of any message exchange conforming to the pattern. [40]
Message-Oriented	An interface that exposes service capabilities through creating, sending, receiving, and reading messages exchanged by distributed systems. The middleware technologies that support this interface type include Java Message Service (JMS) and .NET WCF.
Metadata	Data that defines or describes other data.
Method-Oriented	An interface that exposes service capabilities through a set of operations. Technologies that support this interface type are Web Service framework (WS*) and OGC Web Common Services.
Non-Repudiation	Protective measures against false denial of involvement in a communication. [29]
Normative Document	A document that provides rules, guidelines, or characteristics for activities or their results. Note: The term "normative document" is a generic term that covers such documents as standards, technical specifications, codes of practice, and regulations. [23]
Operation	A set of messages related to a single service action. [39]
Organization	A unique framework of authority within which a person or persons act, or are designated to act, towards some purpose. Any department, service, or other entity within an organization which needs to be identified for information exchange. [21]
Output	Data transferred out of, or the process by which an information processing system or any of its parts transfers data out of, that system or part. (Adapted from [23])
Payload	The actual (business) data transferred by a message; also called message body. [5]
Point-to-Point (PTP)	A JMS messaging model in which messages are routed to an individual consumer that maintains a queue of incoming messages. Each message is addressed to a specific queue, and the receiving clients extract messages from the queues established to hold their messages. While any number of producers can send messages to the queue, each message is guaranteed to be delivered to and consumed by one consumer. [5]

Policy-Based Access Control (PBAC)	A process in which access to system resources is defined and enforced centrally according to formal government policy, and not by local administrators.
Processing	A set of algorithms, calculations, or business rules that operate on input data in order to produce the required output or to produce a change of internal state.
Protocol	A formal set of conventions governing the format and control of interaction among communicating functional units. [8]
Publish/Subscribe (pub/sub)	A messaging model that supports publishing messages to a particular message topic. Subscribers may register interest in receiving messages on a particular message topic. [5]
Quality of Service (QoS) Characteristic	A parameter that specifies and measures the value of a provided service. [5]
Real World Effect	An ultimate purpose associated with the interaction with a particular service. It may be the response to a request for information or the change in the state of some entities shared between the participants in the interaction. (Adapted from [26])
Requester Agent	A software agent that is designed to interact with a service in order to request that a task be performed on behalf of its owner – the service consumer. (Adapted from [36])
Resource	An identifiable entity that has value to a stakeholder. [19] Familiar examples include an electronic document, an image, a source of information (e.g., “today’s weather report for Los Angeles”), a service, and a collection of other resources. [33]
Resource-Oriented	An interface that supports the REST architectural style of interactions, that is, manipulation of XML representations of Web resources using a uniform set of stateless operations, usually a set of HTTP methods.
RESTful Web Service	An application programming interface (API) that uses standard HTTP methods (operations) such as GET, PUT, POST, and DELETE to access and manipulate resources on the Web.
Role	A collection of permissions to use resources made available by a service.
Role-Based Access Control (RBAC)	A form of identity-based access control where the system entities that are identified and controlled are functional positions in an organization or process. [29]
Security	The protection of information and data so that unauthorized persons or systems cannot read or modify them and authorized persons or systems are not denied access to them. [20]

Service	A mechanism to enable access to one or more capabilities, where the access is provided using a prescribed interface and is exercised consistent with constraints and policies as specified by the service description. [25]
Service Consumer	An organization that seeks to satisfy a particular need through the use of capabilities offered by means of a service. (Adapted from [25])
Service Description	The information needed in order to use, or consider using, a service. (Adapted from [25])
Service Interface	An abstract boundary that a service exposes. It defines the types of messages and the message exchange patterns that are involved in interacting with the service, together with any conditions implied by those messages. [36]
Service-Oriented Architecture (SOA)	A paradigm for organizing and utilizing distributed capabilities that may be under the control of different ownership domains. A SOA provides a uniform means to offer, discover, interact with, and use capabilities to produce desired effects consistent with measurable preconditions and expectations. [25]
Service Provider	An organization that offers the use of capabilities by means of a service. (Adapted from [25])
Software Agent	A running program that drives services, both to implement them and to access them. [36]
Synchronous	An interaction is said to be synchronous when the participating agents must be available to receive and process the associated messages from the time the interaction is initiated until all messages are actually received or some failure condition is determined. [40]
Taxonomy	A system or controlled list of values by which to categorize or classify objects.
Token	A data object or a portable, user-controlled, physical device used to verify an identity in an authentication process. [29]
User	A human, his/her agent, a surrogate, or an entity that interacts with information processing systems. [23] A person, organization entity, or automated process that accesses a system, whether authorized to do so or not. [24]
Web Service	A platform-independent, loosely-coupled software component designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format. Other systems interact with the Web service in a manner prescribed by its description by means of XML-based messages conveyed using Internet transport protocols in conjunction with other Web-related standards. (Adapted from [40])

3.3 Acronyms and Abbreviations

AIXM	Aeronautical Information Exchange Model
API	Application Programming Interface
DOT	Department of Transportation
FAA	Federal Aviation Administration
FIXM	Flight Information Exchange Model
FTP	File Transfer Protocol
HTTP(S)	Hypertext Transfer Protocol (Secure)
ID	Identifier
ISO/IEC	International Organization for Standardization/International Electrotechnical Commission
IT	Information Technology
Java EE	Oracle Java Platform, Enterprise Edition
JMS	Java Message Service
MEP	Message Exchange Pattern
MS .NET	Microsoft .NET Framework
NAS	National Airspace System
OASIS	Organization for the Advancement of Structured Information Standards
OGC	Open Geospatial Consortium
OMG CORBA	Object Management Group Common Object Request Broker Architecture
PNG	Portable Network Graphics
QoS	Quality of Service
RBAC	Role-Based Access Control
REST	Representational State Transfer
RFC	Request For Comment
SOA	Service-Oriented Architecture

SOAP	Originally “Simple Object Access Protocol”; the full spelling is no longer used
SRD	Service Requirements Document
SVG	Scalable Vector Graphics
SWIM	System Wide Information Management Program
URL	Uniform Resource Locator
VRTM	Verification Requirements Traceability Matrix
W3C	World Wide Web Consortium
WS	Web Service
WSDL	Web Service Description Language
WSRD	Web Service Requirements Document
WXXM	Weather Information Exchange Model
XML	eXtensible Markup Language

4 GENERAL REQUIREMENTS

This section describes requirements for the stylistic aspects of the SRD. Detailed requirements for the structure and content of the SRD are provided in section 5 of this standard.

4.1 Text, Grammar and Style

a. The text SHALL be written in clear and simple language, free of vague terms, or those subject to misinterpretation.

b. All sentences SHOULD be complete and grammatically correct. Refer to FAA Order 1000.36, FAA Writing Standards [1] for guidance.

c. The United States Government Printing Office Style Manual [10] SHALL be used as a guide for capitalization, spelling, punctuation, syllabification, compounding words, tabular work, and other elements of grammar and style.

4.2 Key Terms

a. Requirements levied in the SRD SHALL be expressed using the terms "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", “MAY", or "OPTIONAL" to specify requirements.

b. These terms SHALL be interpreted as described in RFC 2119 [28].

c. These terms SHALL be capitalized when used to unambiguously specify requirements.

4.3 Page Numbering

a. The front cover page SHALL omit the page number.

b. All pages after the front cover page and before the page containing the first (“Scope”) section SHALL be numbered consecutively with lower-case Roman numerals, starting with ii (for example, ii, iii, and iv).

c. The first page of the first (“Scope”) section SHALL be numbered with an Arabic numeral 1.

d. All subsequent pages SHALL be numbered sequentially using Arabic numerals.

e. The page numbers SHALL be placed at the bottom center of each page.

4.4 Page Headers

a. Each page, including the front cover, SHALL contain a header in the upper right-hand corner right-justified.

b. Each header SHALL contain the SRD Identifier. Note: In most cases, the identifier is assigned by a governing or configuration management organization under whose authority the service is developed or functions.

c. If the SRD is a revision to a baselined SRD, the word “Revision” followed by the revision letter SHALL be included immediately under the SRD Identifier.

d. If the SRD has been approved by a governing or configuration management control organization, the header SHALL include the date of SRD approval on the last line.

e. If the SRD is a draft, the header SHALL include the word “DRAFT” in capital letters under the SRD identifier and the date of the draft on the next line.

4.5 Use of Hyperlinks

To improve the readability and understanding of the SRD, usage of hyperlinks is prescribed as follows:

a. Every term that is used and defined within the SRD SHOULD be linked via a hyperlink reference to the location of its definition in the SRD's "Definitions" section.

b. When the same term is used more than once within the same sentence or paragraph, only the first occurrence of the term SHOULD be referenced.

c. Every document that is cited within the SRD SHALL be linked via a hyperlink reference to the location of its bibliographic entry in the SRD's "Applicable Documents" section.

d. When a document is quoted within the SRD, the quote SHALL include a hyperlink reference to the location of the document's bibliographic entry in the SRD's "Applicable Documents" section.

4.6 Use of Diagrams

There are a number of sections in the SRD where using diagrams is suggested to enhance the understanding of a described topic.

a. Unified Modeling Language (UML) diagrams are RECOMMENDED since UML is able to concisely describe concepts without implying any specific technology. Information about UML diagrams is available at http://www.uml.org/.

4.7 Identifying Figures, Tables, and Appendices

a. Figures SHALL be identified by "Figure", the level one section number in which they appear followed by a dash and numbered sequentially using Arabic numerals within the level one section (e.g., 3-1, 3-2, 3-3, 4-1, 4-2), followed by the figure title.

b. The figure identification SHALL be placed below the figure.

c. Tables SHALL be identified by "Table", the level one section number in which they appear followed by a dash and numbered sequentially using Arabic numerals within the level one section (e.g., 3-1, 3-2, 3-3, 4-1, 4-2), followed by the table title.

d. The table identification SHALL be placed above the table.

e. Appendices SHALL be identified by "Appendix", followed by sequential capital letters (e.g., Appendix A), and the appendix title.

5 DETAILED REQUIREMENTS

This section describes requirements for the structure and content of the SRD.

5.1 Cover Page

a. The SRD SHALL include a cover page as the first page.

b. The upper left corner of the cover page SHALL include the FAA signature (the Department of Transportation triskelion figure with the words “U.S. Department of Transportation” and the words “Federal Aviation Administration” below it) in accordance with FAA Order 1700.6, FAA Branding Policy [3].

c. The line “Service Requirements Document” SHALL be centered above the title.

d. The title SHALL be the name by which the service will be known. Note: In most cases, the title will consist of the approved service’s name issued by the activity authorized to assign the name. That name will be referred to throughout the SRD as the service name.

An example of an SRD cover page is shown in Appendix A.

5.2 Approval Page (Optional)

Signatures on this page ensure that the interested parties have approved the SRD content. The approval page may not be required based on the configuration management policies established within a given organization. The following statements apply when signed approval is required.

a. The approval page SHALL be the first interior page of the SRD.

b. The approval page SHALL contain the line “Service Requirements Document” centered above the title of the service, and the line “Approval Signatures” centered below the title of the service.

c. The approval page SHALL include information for every cosigner.

d. The information SHALL include the cosigner's full name.

e. The information SHALL include the full name of the cosigner's organization followed by the acronym by which the organization is commonly recognized within FAA.

f. The information SHALL include the cosigner's signature.

g. The information SHALL include the date of the signature.

An example of an SRD Approval Page is shown in Appendix B.

5.3 Revision Record Page

a. The SRD SHALL include a revision record page.

b. The revision record page SHALL contain the centered line “Revision Record” above the revision record table.

c. Only revisions SHALL be listed.

d. The revision record page SHALL include information for every revision listed.

e. The information SHALL include the revision letter or number.

f. The information SHALL include a brief description of the revision.

g. The information SHALL include the date of the revision.

h. The information SHALL include the full name of the person who entered this revision record ("Entered by").

An example of an SRD Revision Record Page is shown in Appendix C.

5.4 Table of Contents

a. The SRD SHALL include a table of contents.

b. The SRD SHALL conform to the basic outline shown in Table I below. Note: the sections shown in italics are *optional*.

TABLE I. SRD table of contents

Cover Page
Approval Page
Revision Record Page
Table of Contents
List of Figures
List of Tables
1 Scope
1.1 Background
2 Applicable Documents
2.1 Government Documents
2.2 Non-Government Standards and Other Publications
3 Definitions
3.1 Terms and Definitions
3.2 Acronyms and Abbreviations
4 Service Information
4.1 Service Provider
4.2 Service Consumers
5 Functional Requirements
6 Non-Functional Requirements
6.1 Security Requirements
6.1.1 Authentication
6.1.2 Authorization
6.1.3 Integrity
6.1.4 Confidentiality
6.1.5 Non-Repudiation
6.1.6 Audit Capability
6.1.7 Other Security Requirements

6.2 Qualities of Service Requirements
6.3 Service Policies Requirements
6.4 Processing Requirements
6.5 Operational Environment Requirements
7 Interface Requirements
7.1 Operations Requirements
7.2 Messages Requirements
7.3 Faults Requirements
7.4 Data Requirements
8 Interoperability Requirements
8.1 Data Protocol Requirements
8.2 Message Protocol Requirements
8.3 Transport Protocol Requirements
8.4 Other Protocols Requirements
9 Quality Assurance Provisions
9.1 Responsibility for Verification
9.2 Special Verification Requirements
9.3 Verification Requirements Traceability Matrix
Appendixes

5.5 Scope

a. Section 1 of the SRD SHALL provide a scope statement that briefly describes the coverage of the SRD.

b. At a minimum, section 1 of the SRD SHALL contain the following sentences: “This SRD provides the requirements for the [*name of the service*]. It has been prepared in accordance with FAA-STD-074, Department of Transportation Federal Aviation Administration, *Preparation of Service Requirements Documents* [*cite reference or hyperlink*].”

c. The name of the service SHALL be identical with the name of the service provided on the cover page of the SRD.

d. Section 1 of the SRD MAY include paragraphs on the SRD's purpose, applicability, background, etc. as needed to give readers of the SRD a context for understanding the body of the SRD.

e. Section 1 of the SRD SHALL NOT contain requirements.

5.6 Applicable Documents

a. Section 2 of the SRD SHALL list all documents specifically cited in the SRD.

b. Only documents that are specifically cited in the SRD SHALL be listed in section 2.

c. Every document listed in section 2 of the SRD SHALL present bibliographic information about the document, as shown below:

1. The information SHALL include the full title of the document.
2. The information SHOULD include the alternate title or abbreviated name by which the document is known or recognized. This is particularly relevant for normative documents.
3. The information SHALL include the publisher of the document.
4. The information SHALL include the publication date of the document.
5. The information SHOULD include the appropriate version of the document (e.g., the latest version, the version needed for compatibility with other documents, the version of the document that is under contract by the project.)
6. The information SHALL include the address or location (preferably a persistent Web location, i.e., URL) where a copy of the document can be obtained.

d. Section 2.1 of the SRD SHALL list all types of Government standards and other publications cited in the SRD.

e. Section 2.2 of the SRD SHALL list all types of non-Government standards and other publications cited in the SRD.

f. When requirements are contained in a referenced document, the SRD SHALL specify the extent of the applicability of the referenced requirements (i.e., the section(s) or paragraph(s) that are applicable) if the whole document is not applicable.

g. When requirements are contained in a referenced document, the SRD SHALL specify the extent of any tailoring of those requirements.

h. Applicable documents MAY also be used in the SRD to provide information or clarification without imposing requirements.

i. When a document is quoted within the SRD, the quote SHOULD indicate where in the document the quote is to be found (e.g., using section number, paragraph number, page number, or other means of identification).

5.7 Definitions

a. Section 3.1 of the SRD SHALL define all terms used in the SRD to provide for clarity, unless the terminology is generally accepted and not subject to misinterpretation. See Appendix D of this standard for guidance on writing good definitions.

b. Only terms that are specifically used in the SRD SHALL be listed in section 3.1.

c. Definitions MAY be included by reference to another document.

d. It is RECOMMENDED that the SWIM Controlled Vocabulary [5] be used wherever possible as a source for definitions.

e. Terms and their definitions SHALL be listed in alphabetical order.

f. At a minimum, section 3.1 of the SRD SHALL contain the following paragraph: “The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", “MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [*cite reference or hyperlink, i.e.,* http://www.rfc-editor.org/rfc/rfc2119.txt]. These keywords are capitalized when used to unambiguously specify requirements. When these words are not capitalized, they are meant in their natural-language sense.”

g. Section 3.2 of the SRD SHALL include a list of acronyms and abbreviations used in the SRD, together with their full spelling.

h. Only acronyms and abbreviations that are specifically used in the SRD SHALL be listed in section 3.2.

i. Acronyms and abbreviations SHALL be listed in alphabetical order.

5.8 Service Information

Section 4 of the SRD provides information about the service as a whole, the service provider, and the service consumers.

a. Section 4 of the SRD SHALL NOT contain requirements.

b. Section 4 of the SRD SHALL provide information about the service that is needed to (1) gain a proper understanding of the business objectives of the service as a part of its operational environment, and (2) support semantic consistency in service registration and future service description.

c. The information SHALL include the name of the service.

d. The name SHALL be identical with the name of the service provided on the cover page of the SRD.

e. The information SHALL include a brief plain language description of the service.

f. The information SHALL include a service identifier that uniquely identifies the service. Note: The SWIM Governance team supports creation and maintenance of unique identifiers for SWIM-enabled services[1]. Providers of other services should follow their appropriate organizational policy.

g. The information SHALL include a service version. For guidance on service versioning, see [7].

h. The information SHALL include the criticality level for the service.

i. The single value representing the service's criticality level SHALL be selected from the Service Criticality Taxonomy described in Appendix E of this standard.

j. The information SHALL include a service category, i.e., a classification of the service according to the capabilities it offers.

k. One or more values representing the service category SHOULD be selected from the Service Category Taxonomy described in Appendix E of this standard.

l. If the values in the Service Category Taxonomy in Appendix E do not apply to the service, appropriate new values SHOULD be defined and selected.

m. The information SHALL describe the geographical extent of the data provided by the service.

5.8.1 Service Provider

a. Section 4.1 of the SRD SHALL provide information about the service provider, that is, the FAA organization ultimately responsible for service development and future operation.

[1] A "SWIM-enabled service" is understood as a service that uses SWIM computing and infrastructure assets implemented in compliance with SWIM governance guidelines.

b. The information SHALL include the name of the service provider organization.

c. The name SHALL consist of the full name spelled out followed by the acronym by which it is commonly recognized within FAA.

d. The information SHALL include a brief plain language description of the organization.

e. The information MAY include an accessible reference to the Web page that supplies information about the organization.

5.8.2 Service Consumers (Optional)

a. Section 4.2 of the SRD MAY provide information about consumers that are expected to use the service.

b. The information MAY include each known consumer organization's full name (and acronym if applicable).

c. The information MAY include a brief plain language description of each known organization.

d. The information MAY include an accessible reference to the Web page that supplies information about each known organization.

e. The information MAY include a brief description of the expected consumer audience in lieu of a specific organization, e.g., aviation safety analysts, operations research specialists, etc.

5.9 Functional Requirements

Section 5 of the SRD addresses requirements for service functions from a business point of view, that is, from the point of view of consumer organizations that will use the service to conduct their business. Such requirements do not deal with the mechanics of interacting with a service (this aspect is addressed in the Service Interface section of the SRD), but rather with what activities need to be performed in order to achieve a "real world effect" or specific purpose associated with interacting with the service. For example, a real world effect could be that a flight was rerouted, weather information was received, etc. Requirements in this section are written with emphasis on the functionality of the entire service and not on the specific operations, which are addressed in section 7.1 of the SRD.

a. Section 5 of the SRD SHALL specify requirements for the service's business functions.

b. If the service business functions have also been described in architectural or other requirements documents, the requirements SHOULD contain specific references to these documents.

5.10 Non-Functional Requirements

Section 6 of the SRD specifies requirements for properties that exhibit constraints over the functionality of the service.

5.10.1 Security Requirements

Security is one of the most formidable and unavoidable concerns in the development of services. Service security is generally understood to be collective measures that enable the service to provide protection against security threats. Security threats to a service may include (but are not limited to): unauthorized access to the service information; unauthorized disclosure, modification and destruction of information; unknown status and repudiation in execution; and denial of

service.

Service security requirements encompass the elements of security for all stages of information exchange and modification, that is, while the data is being processed by a service or while it is in transit between components.

Section 6.1 of the SRD specifies the requirements for the security measures, procedures, and policies to be implemented by the service.

5.10.1.1 Authentication

Authentication is "the process of verifying an identity claimed by or for a system entity" [29] so as to ensure that both message sender and message recipient are who they claim to be. Three methods for authentication of services are usually deployed: transport layer authentication, token authentication using the WS-Security specification, and SOAP authentication headers.

a. Section 6.1.1 of the SRD SHALL specify the authentication requirements to be implemented by the service.

b. When authentication is implemented by using a standard protocol or specification document, the information about this document SHALL be presented as prescribed in section 5.6 of this standard.

c. When custom implementation of authentication is required, section 6.1.1 of the SRD SHALL specify requirements for the structure and content of credentials presented to the service.

d. If the SRD requires the use of a password, then the SRD SHALL specify the requirements for password complexity and change management in accordance with FAA Order 1370.121, FAA Information Security and Privacy Program & Policy Document Information [2].

e. If there are no authentication requirements, section 6.1.1 of the SRD SHALL include the statement "This SRD does not impose any authentication requirements."

5.10.1.2 Authorization

In the context of this standard, the authorization process is the granting of rights or permission to a system entity (mainly but not always a user or a group of users) to access a service. There are several authorization models, of which the most relevant for SOA are: role-based, attribute-based and policy-based access control. In role-based access control (RBAC), the set of rights and permissions are associated with a particular role, usually corresponding to work functions of an entity. Because RBAC is the most popular authorization model in FAA (it allows for more granular access control and reduces the amount of administrative effort), RBAC specific requirements are included in this section.

a. Section 6.1.2 of the SRD SHALL specify the authorization requirements to be implemented by the service.

b. When the RBAC model is deployed, section 6.1.2 of the SRD SHALL list all roles defined for the service.

c. Every description of a role listed in section 6.1.2 of the SRD SHALL include the name of the role.

d. Every description of a role listed in section 6.1.2 of the SRD SHALL include a description of all rights or permissions associated with that role.

e. If there are no authorization requirements, section 6.1.2 of the SRD SHALL include the statement "This SRD does not impose any authorization requirements."

5.10.1.3 Integrity

In the context of this standard, integrity is understood to be protective measures that ensure "that data has not been changed, destroyed, or lost in an unauthorized or accidental manner" [29] either in transit or in storage.

a. Section 6.1.3 of the SRD SHALL specify the integrity requirements to be implemented by the service.

b. When integrity is implemented by using a standard protocol or specification document (e.g., XML Signature), the information about this document SHALL be presented as prescribed in section 5.6 of this standard.

c. If there are no integrity requirements, section 6.1.3 of the SRD SHALL include the statement "This SRD does not impose any integrity requirements."

5.10.1.4 Confidentiality

In the context of this standard, confidentiality is understood to be protective measures that ensure "that information is not made available or disclosed to unauthorized individuals, entities, or processes (i.e., to any unauthorized system entity)." [29]

a. Section 6.1.4 of the SRD SHALL specify the confidentiality requirements to be implemented by the service.

b. When confidentiality is implemented by using a standard protocol or specification document (e.g., XML Encryption), the information about this document SHALL be presented as prescribed in section 5.6 of this standard.

c. If there are no confidentiality requirements, section 6.1.4 of the SRD SHALL include the statement "This SRD does not impose any confidentiality requirements."

5.10.1.5 Non-Repudiation

In the context of this standard, non-repudiation is understood to be protective measures "against false denial of involvement in a communication" [29], that is, assurance that the sender or recipient of a message cannot legitimately claim that they did or did not receive the message. The non-repudiation functionality is often provided through the use of a digital signature or, more typically for SOA, XML Signature.

Note: non-repudiation cannot be established without authentication.

a. Section 6.1.5 of the SRD SHALL specify the non-repudiation requirements to be implemented by the service.

b. When non-repudiation is implemented by using a standard protocol or specification document (e.g., WS-Security's standard XML Signature), the information about this document SHALL be presented as prescribed in section 5.6 of this standard.

c. When an audit capability is required to support non-repudiation, the requirements for audit trail SHALL be described in a separate Audit Capability subsection of the SRD. See section 5.10.1.6 below.

d. If there are no non-repudiation requirements, section 6.1.5 of the SRD SHALL include the statement "This SRD does not impose any non-repudiation requirements."

5.10.1.6 Audit Capability

In the context of this standard, a security audit is a process that "records information needed to

establish accountability for system events and for the actions of system entities that cause them" and an audit trail is "a chronological record of system activities that is sufficient to enable the reconstruction and examination of the sequence of environments and activities" [29].

a. Section 6.1.6 of the SRD SHALL specify the audit requirements to be implemented by the service.

b. Section 6.1.6 of the SRD SHALL specify the structure and content of the audit trail (e.g., user ID, date, time, event type, security object, success/failure status, etc.).

c. If there are no audit requirements, section 6.1.6 of the SRD SHALL include the statement "This SRD does not impose any audit requirements."

5.10.1.7 Other Security Requirements

It is a common practice for the security functionality of a service to be implemented by an intermediary security service, that is, a service that receives a request, performs the security steps, and passes the request to the service provider. (The most typical example of such a service is an XML Gateway.)

a. When the service described in the SRD will use a separately established security service, section 6.1.7 of the SRD SHALL provide the name of the security service as well as a brief description of that service and its functionality.

b. When the service described in the SRD will use a separately established security service, the SRD SHALL NOT include requirements for that security service.

c. If there are no other security requirements, section 6.1.7 of the SRD SHALL include the statement "This SRD does not impose any other security requirements."

5.10.2 Qualities of Service Requirements

Qualities of Service (QoS) are measurable characteristics that the service is expected to meet or possess. These characteristics, or parameters, are documented by specifying a parameter's name, definition, required value or range of values it is expected to meet or possess, the method to be used to measure or calculate its values, and the unit of measure in which its values are expressed. The QoS requirements are a combination of performance requirements and information necessary for the developers to size the service software in addition to allowing the service consumers to understand the level of service they can expect.

a. Section 6.2 of the SRD SHALL specify requirements to be met by the service for all QoS parameters that are deemed by the SRD author to be relevant to the service.

b. Section 6.2 of the SRD SHALL specify the name, definition, method of calculation, unit of measure, and required value(s) for all QoS parameters listed. See Appendix D of this standard for guidance on how to write good definitions.

c. If a QoS parameter is relevant but its required value cannot be determined at this time, a description of how the value will be determined SHALL be specified in place of its required value.

See Appendix F of this standard for examples of properly defined QoS parameters.

5.10.3 Service Policies Requirements

Service Policies describe the constraints on the allowable actions of a service's consumers or software agents. Note: A policy may be defined in a separate document (e.g., common policies

for a business or organizational domain) or as part of some document, and it may be expressed in machine-readable or human-readable languages. For example, an organization's security policy that prescribes performing cryptographic operations or use of key-based security tokens can be described using XML as shown in [41].

a. Section 6.3 of the SRD SHALL specify all policy documents that levy requirements on the service's consumers or software agents.

b. Each document that contains these requirements SHALL be documented as prescribed in section 5.6 of this standard.

c. If there are no service policies requirements, section 6.3 SHALL include the statement "This SRD does not impose any service policy compliance requirements."

5.10.4 Processing Requirements

Section 6.4 of the SRD specifies the processing requirements, that is, steps or actions which must be taken on data that is received as part of a service request (input) in order to produce the desired output. Actions on data might be (but are not limited to) transformations, algorithms, unique logic, or business rules. Examples include special validation that goes beyond checking the XML request as being valid or well-formed (e.g., making sure the ZIP code is valid for the city and state in an address), priority processing requirements, overload handling requirements, or timing limitations (e.g., the maximum time to wait for a database response).

a. Section 6.4 of the SRD SHALL specify the processing requirements for the service.

b. If there are no processing requirements, section 6.4 of the SRD SHALL include the statement "This SRD does not impose any processing requirements."

5.10.5 Operational Environment Requirements

Although services are defined as platform and implementation independent, the software components that realize a service are usually subject to the constraints of their potential or existing operational environment. A service may be required to be developed in a specific development environment in compliance with the provider organization's established practices, or to run on an existing enterprise network, or to be hosted by a specific set of hardware and/or operating system. Examples of these constraints include the nature of the enterprise network (e.g., whether it is NAS or non-NAS), firewalls, physical computing resources, etc. Section 6.5 of the SRD specifies such requirements.

a. Section 6.5 of the SRD SHALL specify all requirements relative to the environment in which the service will be developed, operated, and maintained.

b. Section 6.5 of the SRD SHALL specify configuration requirements for the service for a given operational environment.

c. If there are no configuration requirements, section 6.5 of the SRD SHALL include the statement "This SRD does not impose any configuration requirements."

d. If it is expected that the service will be enhanced after its initial implementation, section 6.5 of the SRD SHALL specify all enhanceability requirements identifying the capabilities to add, delete, and adapt software components to provide new or improved functionality, capacity, and/or performance relative to their impact on the system architecture.

e. If the usage of the service is expected to increase after its initial implementation, section 6.5 of the SRD SHALL specify all scalability requirements identifying the capabilities to add, delete, and adapt resources in order to meet the capacity, functionality, and performance

required to meet the increased system loads.

f. If there are no operational environment requirements, section 6.5 of the SRD SHALL include the statement “This SRD does not impose any operational environment requirements.”

5.10.6 Interface Requirements

Section 7 of the SRD specifies requirements for the service interface. This section also includes requirements for operations, messages, faults, and data associated with the interface.

a. Section 7 of the SRD SHALL specify requirements for the interface to be implemented by the service.

b. Section 7 of the SRD SHALL provide a plain language description of the interface.

c. Section 7 of the SRD SHALL specify the service interface type. Common interface types include, but are not limited to, method-oriented, message-oriented, and resource-oriented.

d. If the interface type is message-oriented, section 7 of the SRD SHALL specify a messaging model (point-to-point or publish/subscribe).

e. Section 7 of the SRD SHALL indicate the specification or standard to which the interface is required to conform; e.g., the version of the JMS API specification; the version of HTTP, etc.

f. The interface specification SHALL provide a list of names of all the operations (methods) that are to be implemented by the interface.

g. All operations that constitute the interface SHALL be defined in section 7.1 of the SRD as prescribed in section 5.10.6.1 of this standard.

It is possible, although rarely recommended, that a service may have more than one interface.

h. When a service will expose multiple interfaces, each interface SHALL be specified in section 7 of the SRD as described above in requirements (a) through (g).

5.10.6.1 Operations Requirements

Section 7.1 of the SRD specifies the requirements for the operations that will be offered by the service. Every operation that is listed in SRD section 7.1 Operations represents the patterns and content of interactions of messages described in SRD section 7.2 Messages.

It is important to note that the concept of “operation” cannot be effectively applied for all services. For example, while the Web service interface provides a well-defined structure for operations, the Java Message Service specification [30] does not reference the notion of an operation; and in HTTP-based interfaces (e.g., OGC Web Common, REST), methods may be described as operations, but not always explicitly.

Requirements (b) through (l) apply to service interfaces for which one or more operations can be identified.

a. Section 7.1 of the SRD SHALL specify every operation that will be a part of the interface described in section 7 of the SRD.

b. Each operation specification SHALL provide a name that uniquely identifies the operation throughout the SRD.

c. Each operation specification SHALL provide a plain language description of the operation. For example, “allows consumer to retrieve current status of a specified flight.”

d. Each operation specification SHALL indicate the Message Exchange Pattern (MEP) that will be implemented by the operation by using a single value selected from the Message Exchange Pattern Taxonomy described in Appendix E of this standard.

e. Each operation specification SHALL provide the type of the operation by indicating whether it is to be "synchronous" or "asynchronous".

f. Each operation specification SHALL state if the operation is to be idempotent or non-idempotent.

g. Each operation specification SHALL describe an input, that is, the information that initiates interaction, including the name of the relevant input message defined in section 7.2 of the SRD.

h. Each operation specification SHALL describe an output, that is, the information that is produced in response to a request, including the name of the relevant output message defined in section 7.2 of the SRD.

i. Each operation specification SHALL describe the fault(s), that is, the information to be produced in response to conditions that result in operation failure, including the name of the corresponding fault message defined in section 7.3 of the SRD.

j. All messages referenced in requirements for service operations SHALL be specified in section 7.2 of the SRD as prescribed in section 5.10.6.2 of this standard.

k. All faults referenced in requirements for service operations SHALL be specified in section 7.3 of the SRD as prescribed in section 5.10.6.3 of this standard.

l. Each operation specification SHOULD include a diagram that shows how and in what order messages are exchanged within the context of the operation.

m. For interfaces for which the notion of an Operation is not applicable, section 7.1 of the SRD SHALL state "This SRD does not impose any operations requirements" followed by a brief explanation of why the notion of an operation is not applicable for this specific service interface.

5.10.6.2 Messages Requirements

a. Section 7.2 of the SRD SHALL specify requirements for all input and output messages that will be exchanged between the service and a requester agent. Note: fault messages are specified separately as described in section 5.10.6.3 below.

b. Each message specification SHALL provide a name that uniquely identifies the message throughout the SRD.

c. Each message specification SHALL provide a plain language description of the message.

d. Each message specification SHALL specify a message direction that indicates whether the message is coming to the service ("in") or going from the service ("out").

e. Each message specification SHALL specify the message body (payload) type.

f. The single value representing the message body type SHALL be selected from the Message Body Type Taxonomy described in Appendix E of this standard.

5.10.6.3 Faults Requirements

a. Section 7.3 of the SRD SHALL specify requirements for all faults (also called errors or exceptions) that will be generated in response to conditions that result in failure of an

operation or service function.

b. Each fault specification SHALL provide a name that uniquely identifies the fault throughout the SRD.

c. Each fault specification SHALL provide a plain language explanation of the cause of the fault.

d. Each fault specification MAY provide a fault originator. For example, the fault originator might be a database to which the service passes information received from a consumer that prompts an error.

e. Each fault specification MAY describe an underlying cause that resulted in this fault. Note: there could be an unlimited number of causes.

f. Section 7.3 of the SRD SHOULD include requirements for information (e.g., a textual description of the error) and data elements (e.g., fault code, timestamp, etc.) to be provided with every fault message returned by a service which are not addressed above in requirements (b) through (e).

5.10.6.4 Data Requirements

Data is defined by [32] as "reinterpretable representation of information in a formalized manner suitable for communication, interpretation, or processing." Data is transmitted among SOA components via messages. Section 7.4 of the SRD contains requirements for the data that constitutes the body of messages to be sent or received by the service.

a. Section 7.4 SHALL specify requirements for the data to be provided by the service.

b. Section 7.4 SHALL provide a plain language description of the data.

In most cases a document that describes the data exchanged by a service is produced separately, often by an organization other than the organization responsible for developing the SRD. Usually such a document is developed for use by multiple services and not just for the service described in the SRD. For the purpose of this standard, these documents are referred to as Data Description Documents. Data Description Documents can include XML schemas, information exchange models (e.g., AIXM, FIXM, WXXM), data model diagrams, data dictionaries, or other data description documents.

c. Section 7.4 of the SRD SHOULD provide references to the Data Description Document(s) that defines the data structures, names, constraints, rules governing allowable values, semantics, etc. that the data must conform to.

d. The Data Description Document SHALL maintain a versioning policy.

e. The Data Description Document SHALL have a persistent Web location (URL).

f. The Data Description Document SHALL be documented as prescribed in the Applicable Documents section 5.6 of this standard.

5.11 Interoperability Requirements

Section 8 of the SRD establishes requirements for the data protocol, message protocol, and transport protocol that will be used by the service, as well as any other protocols that cannot be classified as being one of those three types.

a. Section 8.1 of the SRD SHALL specify requirements for the data protocol that the service will use, as described in section 5.11.1 of this standard.

b. Section 8.2 of the SRD SHALL specify requirements for the message protocol that the service will use, as described in section 5.11.2 of this standard.

c. Section 8.3 of the SRD SHALL specify requirements for the transport protocol that the service will use as described in section 5.11.3 of this standard.

d. Section 8.4 of the SRD SHALL specify requirements for other protocols that the service will use, as described in section 5.11.4 of this standard.

5.11.1 Data Protocol Requirements

In order to exchange data between SOA components, an agreed-upon format must be used. A data protocol is a formal set of rules governing data encoding and coordination for data exchange among SOA components.

For the purpose of this standard, two categories of data are considered: text-based data (e.g., an XML document) and binary-encoded data (e.g., a PNG image, a Microsoft Excel spreadsheet). The XML format is the protocol most often employed for exchanging textual data via services. Besides transmitting textual data, an important use for XML is serializing data structures according to domain-specific conceptual models; e.g., the Geography Markup Language (GML) used to serialize information about geographical features or the Aeronautical Information Exchange Model (AIXM) used for transmitting aeronautical information.

Binary-encoded data is data that is converted using a code, frequently consisting of binary numbers or two-dimensional arrays of pixels (a graphical-based encoding). Both XML-based and graphical-based formats are used in today's FAA SOA implementation and, in some scenarios, within the same service implementation; e.g., a Web Map service uses XML to request a map rendered as an SVG image.

a. Section 8.1 of the SRD SHALL specify the data protocol.

b. The normative document that regulates the data protocol SHALL be specified and documented as prescribed in section 5.6 of this standard.

c. If an accessible reference to the normative document is not available, the document itself SHALL be included in an Appendix of the SRD.

d. The data protocol SHALL be compatible with the data prescribed in section 7.4 (“Data Requirements”) of the SRD.

e. If there are no data protocol requirements, section 8.1 of the SRD SHALL include the statement “This SRD does not impose any data protocol requirements.”

5.11.2 Message Protocol Requirements

In a SOA environment, the communication and interaction between components is performed by exchanging messages of predefined content. A message protocol is a formal set of rules and conventions governing procedure calls and responses among communicating SOA components. A widely used message protocol is SOAP, a specification that defines an XML-based common message format. Services may also rely on other application layer protocols for message definitions and transmission, such as HTTP and JMS.

Generally, a message consists of a header part and a message-specific payload. The message header may include directives or contextual information related to the message delivery (e.g. security or addressing information). The message payload consists of instances of service-defined data (see section 5.10.6.4 of this standard for requirements for describing payload data).

a. Section 8.2 of the SRD SHALL specify the message protocol.

b. The normative document that establishes the message protocol SHALL be specified and documented as prescribed in section 5.6 of this standard.

c. If an accessible reference to the normative document is not available, the document itself SHALL be included in an Appendix of the SRD.

d. The message protocol SHALL be compatible with the data protocol specified in section 8.1 of the SRD.

e. If there are no message protocol requirements, section 8.2 of the SRD SHALL include the statement “This SRD does not impose any message protocol requirements.”

5.11.3 Transport Protocol Requirements

A transport protocol is a formal set of rules governing message transmission and port handling among communicating SOA components. The most prevalent example of a transport protocol is the Transmission Control Protocol/Internet Protocol (TCP/IP).

a. Section 8.3 of the SRD SHALL specify the transport protocol.

b. The normative document that establishes the transport protocol SHALL be specified and documented as prescribed in section 5.6 of this standard.

c. If an accessible reference to the normative document is not available, the document itself SHALL be included in an Appendix of the SRD.

d. The transport protocol SHALL be compatible with the message protocol specified in section 8.2 of the SRD.

e. If there are no transport protocol requirements, section 8.3 of the SRD SHALL include the statement “This SRD does not impose any transport protocol requirements.”

5.11.4 Other Protocols Requirements

Some protocols may combine data definitions with messaging conventions or messaging and transport governing conventions and cannot be unambiguously classified as strictly a data, message or transport protocol. This section of the standard describes requirements for specifying such protocols.

a. Section 8.4 of the SRD SHALL specify any other protocols which cannot be clearly identified as a data protocol, a transport protocol, or a message protocol.

b. The normative document that establishes each protocol SHALL be specified and documented as prescribed in section 5.6 of this standard.

c. If an accessible reference to any normative document is not available, the document itself SHALL be included in an Appendix of the SRD.

d. If there are no other protocol requirements, section 8.4 of the SRD SHALL include the statement “This SRD does not impose any other protocol requirements.”

5.12 Quality Assurance Provisions

a. Section 9 of the SRD SHALL specify the process for verification of the requirements contained in the SRD.

b. The test and evaluation process specified in section 4.7, Verification and Validation, of the

Systems Engineering Manual [9] SHALL be used, and tailored as necessary for the levels and methods of verification identified in the Verification Requirements Traceability Matrix (VRTM).

5.12.1 Responsibility for Verification

a. Section 9.1 of the SRD SHALL state: “The FAA is responsible for developing and implementing the verification of requirements for each project. The FAA may delegate verification activities to other organizations, independent contractors, and/or the prime contractor.”

5.12.2 Special Verification Requirements

This section describes any special verification requirements necessary to verify the technical requirements imposed within the SRD.

a. If testing a consumer agent requires special arrangements (e.g., establishment of a test database), section 9.2 of the SRD SHALL state the test requirements and provide the point of contact information to make the arrangements.

b. If testing the consumer agent does not require special arrangements, section 9.2 of the SRD SHALL specify how to report problems with the provider application encountered during testing the consumer application and how they will be resolved.

5.12.3 Verification Requirements Traceability Matrix

There is a one-to-one correspondence between each requirement statement (i.e., the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", “MAY", and "OPTIONAL") and each entry in the Verification Requirements Traceability Matrix (VRTM). An example of a VRTM is shown in Table II below.

a. Section 9.3 of the SRD SHALL contain the following statement: “Verification shall be in accordance with Table 9-1, Verification Requirements Traceability Matrix (VRTM).”

b. The content and structure of the VRTM SHALL comply (and be tailored if necessary) with the Systems Engineering Manual (SEM) [9] section 4.7.

c. The contents of the VRTM SHALL provide verification of each technical requirement contained in the SRD with the appropriate verification method(s).

d. The verification methods used in the VRTM SHALL be identified at the top of the VRTM.

The appropriate verification levels and methods for use in the VRTM are defined in the following paragraphs.

5.12.3.1 Verification Levels

a. The levels of verification are Service and Integration. All requirements imposed by the SRD SHALL be verified at one or both of these levels.

1. Service level (Development). This level is usually conducted at the developer or contractor's facility and culminates in the formal acceptance of a contractual end-item.

2. Integration level (Quality Assurance). This level is conducted by the service consumer. The verification conducted determines if the consumer application software performs with the service provider application software and service specified in the SRD in accordance with the operational and functional requirements.

5.12.3.2 Verification Methods

a. The four verification methods that can be used at either of the verification levels are shown in the Verification Method Taxonomy described in Appendix E of this standard. All requirements imposed by the SRD SHALL be verified using one or more of the values selected from this taxonomy.

TABLE II. Sample verification requirements traceability matrix

A = Analysis; D = Demonstration; I = Inspection; T = Test; X = Not Applicable

Section Number	Requirement Title	Requirement ID	Verification Level	
			Service Level	Integration Level
6.1	Security Requirements			
		1	D	D
		2	A	A
		3	I	I
		n	D	D
7	Interface Requirements			
		1	D	D
		2	D	D
		3	I, D	D
		n	I	X

6 NOTES

(This section contains information of a general or explanatory nature that may be helpful, but is not mandatory.)

This section is not applicable to this standard.

APPENDIXES

Appendix A. Example of an SRD Cover Page

This Appendix is not a mandatory part of the standard. The information contained herein is intended for guidance only. The following template is offered to show what is expected in terms of format and content.

U.S. Department of Transportation
Federal Aviation Administration

FAA-X-XXX
Revision A
September 4, 2020

Service Requirements Document
Special Activity Airspace Management Web Service,
Aeronautical Information Management (AIM)

Appendix B. Example of an SRD Approval Signature Page

This Appendix is not a mandatory part of this standard. The information contained herein is intended for guidance only. The following template is offered to show what is expected in terms of format and content.

FAA-X-XXX
Revision A
September 4, 2020

Service Requirements Document
Special Activity Airspace Management Web Service,
Aeronautical Information Management (AIM)

Approval Signatures

Name	Organi-zation	Signature	Date Signed

i

Appendix C. Example of an SRD Revision Record Page

This Appendix is not a mandatory part of this standard. The information contained herein is intended for guidance only. The following template is offered to show what is expected in terms of format and content.

FAA-X-XXX
Revision A
September 4, 2020

Revision Record

Revision Letter	Description	Revision Date	Entered By

ii

Appendix D. Writing Good Definitions

This Appendix is not a mandatory part of this standard. The information contained herein is intended for guidance only; it is not normative.

The purpose of a definition is to define a concept with words or phrases that describe, explain, or clarify its meaning. Precise and unambiguous definitions are one of the most critical aspects of ensuring interoperability. When two or more parties use a term, it is essential that all be in explicit agreement on the meaning of that term.

ISO/IEC 11179 Part 4 (http://metadata-standards.org/11179/#A4) provides rules for writing good definitions. There are mandatory requirements with which all definitions must comply, and there are recommendations that should be followed when writing a definition. Note the difference between requirements and recommendations: compliance with the requirements can be objectively tested, whereas compliance with the recommendations can only be evaluated subjectively. The rules cited below are abstracted from this document.

Requirements

A definition *shall*:

1. Be stated in the singular.
2. State what the concept is, not only what it is not (i.e., never exclusively in the negative).
3. Be stated as a descriptive phrase or sentence(s).
4. Contain only commonly used abbreviations.
5. Be expressed without embedding definitions of other underlying concepts.

Recommendations

A definition *should*:

1. State the essential meaning of the concept.
2. Be precise and unambiguous.
3. Be concise.
4. Be able to stand alone.
5. Be expressed without embedding rationale, functional usage, domain information, or procedural information.
6. Avoid circular reasoning.
7. Use the same terminology and consistent logical structure for related definitions.

For further explanations of these rules as well as examples of definitions that pass and fail the tests, see the "**Rules for Writing Good Definitions**" section of the *Guidelines for Using the SWIM Vocabulary*, located at https://semantics.aero/SWIM%20CV%20Guidelines.pdf.

Appendix E. Taxonomies

The taxonomies in this Appendix are a mandatory part of this standard. The information contained herein is intended for compliance.

Criticality Level – A single value used to classify the service based on the level of significance given to a functional failure of the service. Values are:

Critical	Loss of this service would significantly raise the risk associated with providing safe and efficient operations.
Essential	Loss of this service would raise the risk associated with providing safe and efficient operations to an unacceptable level.
Routine	Loss of this service would have a minor impact on the risk associated with providing safe and efficient operations.

Service Category – One or more values used to classify the service based on the capabilities it offers. Note: the *Information* and *Core* categories are used only for logical grouping and should not be selected as values for SRD purposes.

Information	*A service that offers capabilities for generating, making available, storing, managing, and analyzing information.*
Aeronautical	A service that provides information used to describe, manage, and control aeronautical facts, concepts or instructions such as special use airspace restrictions, airport configuration, and Notices to Airmen (NOTAMS).
Flight	A service that provides information used to describe, manage, and control the safe movement of aircraft in the airspace, including information such as flight itinerary, flight identification, flight planning, flight events and status, and air traffic management (ATM) control events associated with a single flight, where a flight normally includes one takeoff and one landing.
Weather	A service that provides information used to describe current or predicted atmospheric conditions, including terminal and airborne weather observations, forecasts, and reports of weather phenomena.
Infrastructure	A service that provides information used to describe the infrastructure and resources supporting a flight such as landing facilities, air space partitions, communications systems, navigation systems, surveillance systems, automation tools, etc.
Surveillance	A service that provides information produced by technologies (e.g., radar, beacon interrogator, automatic dependent surveillance-broadcast) for detecting and locating airborne and taxiing aircraft and ground support vehicles.
World Features	A service that provides information used to describe natural and manmade features (terrain, time, coordinate systems, geopolitical boundaries, etc.) utilized in the management of air transportation assets or flight activities.
Core	*A service that offers capabilities by which to interconnect, adapt, and facilitate services provided by other parties.*

Discovery	A service that provides capabilities to a service consumer to obtain information about available services.
Security	A service that provides capabilities for protecting systems from unauthorized access or interference with data transfers.
Messaging	A service that provides capabilities for distributing messages exchanged by interacting components.
Mediation	A service that provides capabilities for connecting systems that deploy different data representations, formats, or protocols.

Message Exchange Pattern (MEP) – A single value that indicates the pattern of message exchange between interacting components. Values are:

In-Only	Indicates an operation which has only an input message, that is, a message is sent to the service and service does not produce any output message.
In-Out	Indicates an operation where an input message is sent to the service first and an output message (or a fault message) is generated in response.
Out-Only	Indicates an operation which has only an output message, that is, the service generates the output message but does not expect to receive any response message or fault messages.
Out-In	Indicates an operation where the service generates the output message and in return the input message (or a fault message) is received.

Message Body Type – A single value that indicates the nature of the actual (business) data transferred by the message. Values are:

Text	The message body contains a text, e.g., XML.
Stream	The message body contains a stream of primitive values that are written and read sequentially.
Map	The message body contains a set of name-value pairs, where names are strings, and values are primitives.
Object	The message body contains a serialized object.
Byte	The message body contains an array of primitive bytes.

Verification Method – One or more values that indicate the method used to verify a given requirement. Values are:

Analysis	A method in which hardware or software designs are compared with known scientific and technical principles, procedures, and practices to estimate the capability of the proposed design to meet the mission and system requirements.

Demonstration	A method in which qualitative determination of properties is made for a configuration item, including software and/or the use of technical data and documentation. The items being verified are observed, but not quantitatively measured, in a dynamic state.
Inspection	A method used to determine compliance without using special laboratory equipment, procedures, or services and consists of a nondestructive static-state examination of hardware, software, and/or technical data and documentation.
Test	A method in which performance is measured during or after the controlled application of functional and/or environmental stimuli. Quantitative measurements are analyzed to determine the degree of compliance. The process uses standardized laboratory equipment, procedures, and/or services.

Appendix F. Examples of Quality of Service Parameters

This Appendix is not a mandatory part of this standard. The information contained herein is intended for guidance only. The table below provides non-normative examples of QoS parameters. Service requirements developers may reuse these parameters or provide their own, as well as their own values or range of values. Information on QoS is also available in the FAA Reliability, Maintainability, and Availability (RMA) Handbook (http://www.tc.faa.gov/its/worldpac/standards/faa-hdbk-006c.pdf).

QoS Parameter Name	Definition	Method	Unit of Measure	Required Value or Range of Values
Accuracy	Number of errors produced by the service over a period of time.	Simple count. Measurements are taken daily and apply to the preceding 24-hour period.	Whole positive number.	
Availability	Probability that the service will be operational during any randomly selected period of time, or, alternatively, the fraction of the total available operating time that the service is operational.	(Total Time in an interval - Total Outage Time in the interval) / Total Time in the interval.	Probability expressed to 3 decimal places.	
Capacity	Number of service requests that the service can accommodate within a given time period.	Simple count.	Whole positive number, per period of time.	
Mean Time Between Failure (MTBF)	Average time for service interruptions that result in the service being automatically restored within “T” seconds.	The sum of the individual times between failures that result in the service being automatically restored within “T” seconds divided by the number of those failures.	Hours.	
Mean Time Between Critical Failure (MTBCF)	Average time for service interruptions that are not successfully restored to the available state within “T” seconds.	The sum of the individual times for service interruptions that are not successfully restored to the available state within “T” seconds divided by the number of those failures.	Hours.	
Mean Time To Restore (MTTR)	Average time required to return the service to a pre-determined (available) state after a failure.	The sum of the times to restore service after failures divided by the number of times the service was restored.	Seconds.	

QoS Parameter Name	Definition	Method	Unit of Measure	Required Value or Range of Values
Response Time	Maximum time required to complete a service request.	Measured from the time the service provider receives the request to the time the service provider transmits the response.	Seconds.	