Open Geospatial Consortium

Technology Office
4899 North Old State Road 37
Bloomington, IN 47408
Telephone: +1-812-334-0601
Facsimile: +1-812-961-2053

Request For Quotation

And

Call For Participation

In the

FEDERAL AVIATION ADMINISTRATION (FAA) SPECIAL ACTIVITY AIRSPACE (SAA) DISSEMINATION PILOT (FAA SAA DISSEMINATION PILOT)

Appendix C to Annex B – SR Web Service Description Document (WSSD)

RFQ Issuance Date: September 20, 2010
Proposal Due Date: October 18, 2008
U.S. Department of Transportation  
Federal Aviation Administration  

Web Service Description Document  

Aeronautical Information Management  
Special Activity Airspace Management Static Repository Service  

Version 0.3  
August 16, 2010
### Approval Signatures

<table>
<thead>
<tr>
<th>Participant</th>
<th>Name</th>
<th>Signature</th>
<th>Date Signed</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Revision Letter</td>
<td>Description</td>
<td>Date</td>
<td>Entered By</td>
</tr>
<tr>
<td>-----------------</td>
<td>-------------------------------------------------------</td>
<td>---------------</td>
<td>------------</td>
</tr>
<tr>
<td>Version 0.1</td>
<td>Initial Draft</td>
<td>May 28, 2010</td>
<td>Kevin Lew</td>
</tr>
<tr>
<td>Version 0.2</td>
<td>Addressed internal comments.</td>
<td>June 1, 2010</td>
<td>Kevin Lew</td>
</tr>
<tr>
<td>Version 0.3</td>
<td>Addressed comments from SWIM, the Static Repository</td>
<td>August 16, 2010</td>
<td>Kevin Lew</td>
</tr>
<tr>
<td></td>
<td>Developers, and ERAM</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table of Contents

1. SCOPE ....................................................................................................................... 6
  1.1 Background ........................................................................................................... 6

2. APPLICABLE DOCUMENTS ..................................................................................... 7
  2.1 Government Documents ....................................................................................... 7
  2.2 Non-Government Stands and Other Publications .................................................. 9

3. DEFINITIONS ............................................................................................................. 10
  3.1 Abbreviations and Acronyms ............................................................................... 10

4. WEB SERVICES PROPERTIES AND CAPABILITIES ........................................ 11
  4.1 Service Profile ...................................................................................................... 11
    4.1.1 Service Provider ............................................................................................... 11
    4.1.2 Service Consumers ........................................................................................... 11
    4.1.3 Service Functionality ....................................................................................... 12
    4.1.4 Security ............................................................................................................ 12
    4.1.5 Quality of Service ............................................................................................ 12
    4.1.6 WSDL Documents ........................................................................................... 12
  4.2 Service Interfaces .................................................................................................. 13
    4.2.1 Types ............................................................................................................... 13
    4.2.2 Messages ......................................................................................................... 18
    4.2.3 Operations ....................................................................................................... 19
    4.2.4 Lists of Interfaces ............................................................................................. 22
  4.3 Service Implementation .......................................................................................... 22
    4.3.1 End Points ........................................................................................................ 22

APPENDIX A. ACCEPTED DATA ELEMENTS ................................................................ 25

List of Tables

TBL 4-1 SAA SR Service Quality of Service Parameters .................................................. 12
TBL 4-2 SAA SR Put Service Data Elements ................................................................ 13
TBL 4-3 SAA SR Get Service Data Elements ................................................................. 15
TBL 4-4 SAA SR Put Service Message Descriptions ...................................................... 18
TBL 4-5 SAA SR Get Service Message Descriptions ...................................................... 18
TBL 4-6 SAA SR Get Service Message Descriptions ...................................................... 19
TBL 4-7 insertSaa Operation Description .................................................................... 19
TBL 4-8 updateSaa Operation Description .................................................................... 20
TBL 4-9 insertUnit Operation Description ..................................................................... 20
TBL 4-10 getNewUuid Operation Description ............................................................... 20
TBL 4-11 getSaaByUuid Operation Description ............................................................. 20
TBL 4-12 getSaaNames Operation Description ............................................................... 21
TBL 4-13 getSaaUUID Operation Description ................................................................. 21
TBL 4-14 getUnits Operation Description ..................................................................... 21
TBL 4-15 validateSaa Operation Description ................................................................. 21
TBL 4-16 List of Interfaces ............................................................................................ 22
TBL 4-17 SAA SR Put Service Quality of Service Parameters ....................................... 23
TBL 4-18 SAA SR Service Quality of Service Parameters ............................................. 23
TBL 4-19 SAA SR Notification Service Quality of Service Parameters ....................... 24
1. **Scope**

This Web Service Description Document (WSDD) was prepared in accordance with Federal Aviation Administration (FAA) Standard FAA-STD-065. It provides the design characteristics for the interface between Special Activity Airspace (SAA) Static Repository (SR) Service and its web service clients. This WSDD satisfies the interface design requirements for the Static Repository Service requirements documented in the Aeronautical Information Management (AIM) Special Activity Airspace Management Web Services Interface Requirements Document (IRD).

1.1 **Background**

This document describes the interface between the AIM SAA SR Service and web service clients.

SAA is an unofficial term of convenience to collectively describe Special Use Airspace (SUA), Air Traffic Control Assigned Airspace (ATCAA), and other airspace that can be reserved for use. The use of the term SAA in this document refers to SUAs and ATCAAs.

An SUA is a region of airspace designated to be used (typically by the military) based on a schedule.

An ATCAA is an airspace assigned by air traffic control to provide air traffic segregation between the specified activities being conducted within the assigned airspace and other Instrument Flight Rules (IFR) traffic.

The SAA SR maintains the static definitions of SAA. The static definition is a digital representation of the legal definition of the SAA using the Aeronautical Information Exchange Model (AIXM) specification. The SAA SR Service is a System Wide Information Management (SWIM)-compliant web service that allows for the ability to update SAA static definitions as well as disseminate them to users of the service.
2. Applicable Documents

2.1 Government Documents

The following documents form a part of this WSDD to the extent specified herein.

<table>
<thead>
<tr>
<th>FAA SPECIFICATIONS:</th>
<th>System-Wide Information Management (SWIM) Service Specification Document (SvSD) Segment 1, version 1.7 SWIM WS-I Basic Security Profile 1.1 System-Wide Information Management (SWIM) Final Program Requirements (FPR) Segment 1, Rev. 7.3</th>
</tr>
</thead>
<tbody>
<tr>
<td>December 22, 2009</td>
<td></td>
</tr>
<tr>
<td>May 23, 2007</td>
<td></td>
</tr>
<tr>
<td>FAA STANDARDS:</td>
<td></td>
</tr>
<tr>
<td>FAA-STD-063 May 1, 2009</td>
<td>Standard Practice, XML Namespaces</td>
</tr>
<tr>
<td>FAA-STD-064 May 1, 2009</td>
<td>Standard Practice, Web Service Registration</td>
</tr>
<tr>
<td>FAA-STD-065 February 26, 2010</td>
<td>Standard Practice, Preparation of Web Service Description Documents</td>
</tr>
<tr>
<td>FAA-STD-025f November 30, 2007</td>
<td>Preparation of Interface Documentation</td>
</tr>
<tr>
<td>STANDARDS FOR FAA SWIM PROGRAM COMPLIANCE:</td>
<td>Hypertext Transfer Protocol – HTTP/1.1</td>
</tr>
<tr>
<td>W3C XML Recommendation September 29, 2006</td>
<td>Universal Description, Discovery, and Integration (UDDI) Standard, version 3.0.2</td>
</tr>
<tr>
<td>OASIS UDDI v. 3.0.2 October 19, 2004</td>
<td>World Wide Web Consortium Web Services Description Language (WSDL) version 1.1</td>
</tr>
<tr>
<td>W3C WSDL v. 1.1 March 15, 2001</td>
<td>Transport Layer Security (TLS) – version 1.0</td>
</tr>
<tr>
<td>IETF INTERNET-DRAFT November 18, 1996</td>
<td>JO 7400.2G – Procedures for Handling Airspace Matters</td>
</tr>
</tbody>
</table>

OTHER FAA PUBLICATIONS:

April 10, 2008 JO 7400.2G – Procedures for Handling Airspace Matters
February 16, 2010

JO 7400.8S – Special Use Airspace

March 9, 2009

System Wide Information Management (SWIM)
Governance Plan, Version 2.1

Aeronautical Information Management (AIM) Special Activity Airspace
(SAA) Management Web Services Interface Requirements Document
(IRD) version 1.0

June 23, 2010

https://swimwiki.tc.faa.gov/download/attachments/6881348/SWIM+Special+Activity+Airspace+Requirements.doc
### 2.2 Non-Government Standards and Other Publications

The following documents form a part of this WSDD to the extent specified herein.

<table>
<thead>
<tr>
<th>STANDARDS:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Aeronautical Information Exchange Model 5.0 Specification October 2008</td>
<td>The model used for defining SAAs used in the transmission to and from the SAA SR Service.</td>
</tr>
<tr>
<td></td>
<td><a href="http://www.aixm.aero/public/standard_page/download_5_0.html">http://www.aixm.aero/public/standard_page/download_5_0.html</a></td>
</tr>
<tr>
<td>Internet Engineering Task Force (IETF) Request for Comments (RFC) 791</td>
<td>Internet Protocol (IP) as updated by RFC 1349</td>
</tr>
<tr>
<td>September 1981</td>
<td></td>
</tr>
<tr>
<td>IETF RFC 792 September 1, 1981</td>
<td>Internet Control Message Protocol (ICMP), updated by RFC 950</td>
</tr>
<tr>
<td>IETF RFC 793 September 1981</td>
<td>Transmission Control Protocol (TCP), updated by RFC 3168</td>
</tr>
<tr>
<td>IETF RFC 826 November 1, 1982</td>
<td>Ethernet Address Resolution Protocol (ARP)</td>
</tr>
<tr>
<td>IETF INTERNET-DRAFT November 18, 1996</td>
<td>The Secure Sockets Layer (SSL) Protocol Version 3.0</td>
</tr>
<tr>
<td>Web Services Security</td>
<td>Web Services Security UsernameToken Profile 1.1, OASIS Standard</td>
</tr>
<tr>
<td>UsernameToken Profile 1.1 February 1, 2006</td>
<td>Specification</td>
</tr>
</tbody>
</table>
3. Definitions

SOAP - A protocol specification for exchanging structured information in the implementation of Web Services in computer networks.

3.1 Abbreviations and Acronyms

AIM  Aeronautical Information Management
AIXM  Aeronautical Information Exchange Model
ARP  Address Resolution Protocol
ARTCC  Air Route Traffic Control Center
ATCAA  Air Traffic Control Assigned Airspace
ATM  Air Traffic Management
ERAM  En Route Automation Modernization
FAA  Federal Aviation Administration
FAA-STD  FAA Standard
GML  Geography Markup Language
HTTP  Hypertext Transfer Protocol
HTTPS  Hypertext Transfer Protocol over Secure Socket Layer
ICAO  International Civil Aviation Organization
ICMP  Internet Control Message Protocol
ID  Identifier
IETF  Internet Engineering Task Force
IP  Internet Protocol
IRD  Interface Requirements Document
JMS  Java Message Service
MEP  Message Exchange Pattern
MILOPS  Military Operations Systems
NAS  National Airspace System
OSI  Open System Interconnection
RFC  Request For Comments
SAA  Special Activity Airspace
SAA OR  Special Activity Airspace Operational Repository
SAA PR  Special Activity Airspace Project Repository
SAA SR  Special Activity Airspace Static Repository
SSL  Secure Socket Layer
STD  Standard
SUA  Special Use Airspace
SwSD  SWIM Services Specification Document
SWIM  System-Wide Information Management
TBD  To Be Determined
TCP  Transmission Control Protocol
TLS  Transport Layer Security
UDDI  Universal Description, Discovery, and Integration
UUID  Universally Unique Identifier
WSDD  Web Service Description Document
WSDL  Web Services Description Language
WS-1  Web Services Interoperability Organization
XML  Extensible Markup Language
4. Web Services Properties and Capabilities

The SAA SR Service is a collection of three different web services: the SAA SR Put Service, the SAA SR Get Service, and the SAA SR Notification Service.

The SAA Static Repository is populated through submissions of SAAs from the SAA Project Repository through calls to the SAA SR Put Service. The web service handles either the insertion of a new SAA into the Static Repository or an update to an existing SAA.

The SAA SR Get Service provides operations for service consumers to call to retrieve data related to SAAs through request/response services. These requests can include retrieving SAA definitions, Universally Unique Identifiers (UUID), names, as well as validating SAAs submitted by the service consumer.

The SAA SR Notification Service is a Java Message Service (JMS) publish/subscribe (pub/sub) service that provides notifications for when an SAA is added to or updated in the Static Repository.

The following sections describe the initial version of SAA SR Service, in terms of its service profile, consumer, provider, overall usage, security and quality of service attributes.

4.1 Service Profile

The SAA SR Put Service provides a service interface for the SAA Project Repository to submit new and modified SAA static definitions. The SAA SR Notification Service then notifies its subscribers via JMS pub/sub topics that new or modified SAAs are present in the SR. The users of the SAA SR Get Service can then request the static definitions of these specific SAAs.

The service namespaces are:

The service version is version 1, and the service category according to FAA-STD-066 is Air Transportation Airspace Information Service, ID 1.4.1. The SAA SR Service is in the development stage currently and will be in the operational stage upon deployment. The SAA SR Service provides a routine service, as characterized in the NAS-SR-1000.

4.1.1 Service Provider

The SAA SR Service is provided by Aeronautical Information Management (AIM). The AIM group is the authoritative government source for collecting, validating, storing, maintaining, and disseminating aeronautical data concerning the United States and its territories to support real-time aviation activities.

4.1.1.1 Point of Contact

Douglas Sage
Automation Specialist, Aeronautical Information Services (AIS) Automation, AJR-32
Doug.Sage@faa.gov

4.1.2 Service Consumers

Special Activity Airspace Project Repository, AIM.
Special Activity Airspace Operational Repository Service, AIM.
En Route Automation Modernization (ERAM).
4.1.3 Service Functionality

The SAA SR Service provides the ability to insert new and update existing static definitions of SAAs as well as provide the ability to retrieve these static definitions on request. The SAA SR Service also provides a pub/sub topic that notifies subscribers when an SAA static definition has been added or changed.

The existing management of SUA definitions exchanges the definitions using different data formats, and at times using manual methods (sending a CD through the mail). The SAA SR Service will enforce a consistent, human readable data exchange format, and an entirely electronic means of transmitting SAA definition information.

4.1.4 Security

Section redacted.

4.1.4.1 Roles

The first version of the SAA SR Service has two roles:

SAA Approver – Users who can update or submit a new SAA to the SAA SR Service.

SAA Static Data Reader – Users who can request SAA definitions from the SAA SR Service.

4.1.4.2 Access Control Mechanisms

The SAA SR Service will restrict access to its web services through user authentication on each service call. A username/password combination will be contained in the SOAP header in each request to the service.

Access control to the SR Notification Service requires the JMS client to provide username/password.

4.1.4.3 Security Policies

The SAA SR Service is in compliance to the SWIM WS-I Basic Security Profile 1.1.

4.1.5 Quality of Service

The SAA SR Service meets the following quality of service parameters:

<table>
<thead>
<tr>
<th>QoS Parameter</th>
<th>Definition</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>A measure of the lowest probability that a system or constituent piece will be operational during any randomly selected period of time, or, alternatively, the fraction of the total available operating time that the system or constituent piece is operational.</td>
<td>.9975</td>
</tr>
</tbody>
</table>

4.1.6 WSDL Documents

There are two WSDL documents, one for the SaaStaticRepositoryPutService and one for the SaaStaticRepositoryGetService. The WSDL documents are provided through the namespace listed in section 4.1 Error! Reference source not found.. They are also available through the following links:

SAA SR Put Service:
https://swimwiki.tc.faa.gov/download/attachments/6881321/SAAStatic-PutService.wsdl
SAA SR Get Service:
https://swimwiki.tc.faa.gov/download/attachments/6881321/SAAStatic-GetService.wsdl

4.2 Service Interfaces

The service interfaces section defines the data element types, messages exchanged, operations available, and interface provided in this version of the SAA SR Service.

4.2.1 Types

TBL 4-2 defines the data elements (types) used in the requests to and responses from the SAA SR Service for submissions. Each data element of a complex type, then lists out the sub data elements that it is composed of.

**TBL 4-2**  
SAA SR Put Service Data Elements

<table>
<thead>
<tr>
<th>Data Element</th>
<th>Sub Data Element</th>
<th>Element Definition</th>
<th>Occurrence</th>
<th>Obligation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fault</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>A Fault containing the reason why a service call is not successful.</td>
</tr>
<tr>
<td>Body</td>
<td>String</td>
<td>1</td>
<td>Required</td>
<td></td>
<td>A description of the fault.</td>
</tr>
<tr>
<td>InsertSaa Request</td>
<td>AIXM SaaMessage Object</td>
<td>1</td>
<td>Required</td>
<td>An AIXM object that contains the definition of the SAA. Detailed in Appendix A.</td>
<td></td>
</tr>
<tr>
<td>InsertSaa Response</td>
<td>boolean</td>
<td>1</td>
<td>Required</td>
<td>Indicates success.</td>
<td></td>
</tr>
<tr>
<td>Success</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>The Universally Unique Identifier of the SAA. A 32 character string that uniquely identifies an AIXM feature within a codespace.</td>
</tr>
<tr>
<td>Uuid</td>
<td>String</td>
<td>1</td>
<td>Required</td>
<td>A URN used to resolve the UUID.</td>
<td></td>
</tr>
<tr>
<td>Codespace</td>
<td>String</td>
<td>1</td>
<td>Required</td>
<td>The type of SAA submitted.</td>
<td></td>
</tr>
<tr>
<td>Type</td>
<td>Possible Values: SUA ATCAA SAA_COMPONENT</td>
<td>1</td>
<td>Required</td>
<td>The designator of the SAA that was submitted to the service for insertion.</td>
<td></td>
</tr>
<tr>
<td>Designator</td>
<td>String</td>
<td>1</td>
<td>Required</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data Element</td>
<td>Sub Data Element</td>
<td>Element Definition</td>
<td>Occurrence</td>
<td>Obligation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>------------------</td>
<td>--------------------</td>
<td>------------</td>
<td>------------</td>
<td>-------------</td>
</tr>
<tr>
<td>UpdateSaa Request</td>
<td>SaaMessage</td>
<td>AIXM SaaMessage Object</td>
<td>1</td>
<td>Required</td>
<td>Used to submit an SAA update to the SAA SR.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>An AIXM object that contains the definition of the SAA. Detailed in Appendix A.</td>
</tr>
<tr>
<td>UpdateSaa Response</td>
<td>success</td>
<td>boolean</td>
<td>1</td>
<td>Required</td>
<td>Indicates success.</td>
</tr>
<tr>
<td></td>
<td>uuid</td>
<td>String</td>
<td>1</td>
<td>Required</td>
<td>The Universally Unique Identifier of the SAA. A 32 character string that uniquely identifies an AIXM feature within a codespace.</td>
</tr>
<tr>
<td></td>
<td>codespace</td>
<td>String</td>
<td>1</td>
<td>Required</td>
<td>A URN used to resolve the UUID.</td>
</tr>
<tr>
<td></td>
<td>type</td>
<td>Possible String values: SUA ATCAA SAA_COMPONENT</td>
<td>1</td>
<td>Required</td>
<td>The type of SAA submitted.</td>
</tr>
<tr>
<td></td>
<td>designator</td>
<td>String</td>
<td>1</td>
<td>Required</td>
<td>The designator of the SAA that was submitted to the service for update.</td>
</tr>
<tr>
<td>InsertUnit Request</td>
<td>SaaMessage</td>
<td>AIXM SaaMessage Object</td>
<td>1</td>
<td>Required</td>
<td>An AIXM object that contains the definition of the Unit. Detailed in Appendix A.</td>
</tr>
<tr>
<td>InsertUnit Response</td>
<td>success</td>
<td>boolean</td>
<td>1</td>
<td>Required</td>
<td>Indicates success.</td>
</tr>
<tr>
<td></td>
<td>uuid</td>
<td>String</td>
<td>1</td>
<td>Required</td>
<td>The Universally Unique Identifier of the Unit. A 32 character string that uniquely identifies an AIXM feature within a codespace.</td>
</tr>
<tr>
<td></td>
<td>codespace</td>
<td>String</td>
<td>1</td>
<td>Required</td>
<td>A URN used to resolve the UUID.</td>
</tr>
<tr>
<td></td>
<td>name</td>
<td>String</td>
<td>1</td>
<td>Required</td>
<td>The name of the Unit submitted.</td>
</tr>
<tr>
<td>Data Element</td>
<td>Sub Data Element</td>
<td>Element Definition</td>
<td>Occurrence</td>
<td>Obligation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>------------------</td>
<td>--------------------</td>
<td>------------</td>
<td>------------</td>
<td>-------------</td>
</tr>
<tr>
<td>type</td>
<td></td>
<td>Possible String values: MIL MILOPS OTHER</td>
<td>1</td>
<td>Required</td>
<td>The type of Unit submitted.</td>
</tr>
</tbody>
</table>

**TBL 4-3**
SAA SR Get Service Data Elements

<table>
<thead>
<tr>
<th>Data Element</th>
<th>Sub Data Element</th>
<th>Element Definition</th>
<th>Occurrence</th>
<th>Obligation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fault</td>
<td></td>
<td>A Fault containing the reason why a service call is not successful.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>body</td>
<td>String</td>
<td>1</td>
<td>Required</td>
<td>A description of the fault.</td>
<td></td>
</tr>
<tr>
<td>UuidKeyType</td>
<td></td>
<td>The unique key for a UUID. A string that is unique within the paired codespace that uniquely identifies a specific entity (i.e., SAA, Unit).</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>uuid</td>
<td>String</td>
<td>1</td>
<td>Required</td>
<td>The Universally Unique Identifier of the SAA. A 32 character string that uniquely identifies an AIXM feature within a codespace.</td>
<td></td>
</tr>
<tr>
<td>codespace</td>
<td>String</td>
<td>1</td>
<td>Optional</td>
<td>A URN used to resolve the UUID.</td>
<td></td>
</tr>
<tr>
<td>SaaKeyType</td>
<td></td>
<td>The key for an SAA.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>1</td>
<td>Optional</td>
<td>The name of the SAA.</td>
<td></td>
</tr>
<tr>
<td>uuid</td>
<td>String</td>
<td>1</td>
<td>Optional</td>
<td>The Universally Unique Identifier of the SAA. A 32 character string that uniquely identifies an AIXM feature within a codespace.</td>
<td></td>
</tr>
<tr>
<td>codespace</td>
<td>String</td>
<td>1</td>
<td>Optional</td>
<td>A URN used to resolve the UUID.</td>
<td></td>
</tr>
<tr>
<td>SaaUuidType</td>
<td></td>
<td>A combination of the SAA name and type.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>1</td>
<td>Required</td>
<td>The name of the SAA.</td>
<td></td>
</tr>
<tr>
<td>type</td>
<td>saaType</td>
<td>Possible values: SUA ATCAA SAA_COMPONENT</td>
<td>1</td>
<td>Required</td>
<td>The type of SAA.</td>
</tr>
<tr>
<td>UnitKeyType</td>
<td></td>
<td>The key data for a Unit.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data Element</td>
<td>Sub Data Element</td>
<td>Element Definition</td>
<td>Occurrence</td>
<td>Obligation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>------------------</td>
<td>--------------------</td>
<td>------------</td>
<td>------------</td>
<td>-------------</td>
</tr>
<tr>
<td>uuid</td>
<td>String</td>
<td>1</td>
<td>Optional</td>
<td>The Universally Unique Identifier of a Unit. A 32 character string that uniquely identifies an AIXM feature within a codespace.</td>
<td></td>
</tr>
<tr>
<td>codespace</td>
<td>String</td>
<td>1</td>
<td>Optional</td>
<td>A URN used to resolve the UUID.</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>1</td>
<td>Optional</td>
<td>The name of the Unit.</td>
<td></td>
</tr>
<tr>
<td>designator</td>
<td>String</td>
<td>1</td>
<td>Optional</td>
<td>The designator for the Unit.</td>
<td></td>
</tr>
<tr>
<td>SaaSUuid</td>
<td>Element</td>
<td></td>
<td></td>
<td></td>
<td>Used in the retrieval of the UUID of an SAA.</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>1</td>
<td>Optional</td>
<td>The name of the SAA.</td>
<td></td>
</tr>
<tr>
<td>type</td>
<td>String</td>
<td>1</td>
<td>Optional</td>
<td>The type of the SAA.</td>
<td></td>
</tr>
<tr>
<td>GetUuidReq</td>
<td>Request</td>
<td></td>
<td></td>
<td></td>
<td>Used to retrieve a requested number of new UUIDs.</td>
</tr>
<tr>
<td>count</td>
<td>int</td>
<td>1</td>
<td>Required</td>
<td>The number of UUIDs requested.</td>
<td></td>
</tr>
<tr>
<td>GetUuidRes</td>
<td>Request</td>
<td></td>
<td></td>
<td></td>
<td>Used to return a set of new UUIDs to the service consumer.</td>
</tr>
<tr>
<td>UuidKey</td>
<td>UuidKeyType</td>
<td>Maximum of 100</td>
<td>Required</td>
<td>The Universally Unique Identifier. A 32 character string that is used to uniquely identify an AIXM feature within a codespace.</td>
<td></td>
</tr>
<tr>
<td>GetSaaSByUuid</td>
<td>Request</td>
<td></td>
<td></td>
<td></td>
<td>Used to request an SAA with a specific UUID.</td>
</tr>
<tr>
<td>uuid</td>
<td>string</td>
<td>1</td>
<td>Required</td>
<td>The Universally Unique Identifier of the SAA. A 32 character string that uniquely identifies an AIXM feature within a codespace.</td>
<td></td>
</tr>
<tr>
<td>codespace</td>
<td>string</td>
<td>1</td>
<td>Required</td>
<td>A URN used to resolve the UUID.</td>
<td></td>
</tr>
<tr>
<td>startTime</td>
<td>dateTime</td>
<td>1</td>
<td>Optional</td>
<td>The start time of the interval that the SAA request is for.</td>
<td></td>
</tr>
<tr>
<td>endTime</td>
<td>dateTime</td>
<td>1</td>
<td>Optional</td>
<td>The end time of the interval that the SAA request is for.</td>
<td></td>
</tr>
<tr>
<td>GetSaaSByUuid</td>
<td>Response</td>
<td></td>
<td></td>
<td></td>
<td>Used to return a requested SAA to the service consumer.</td>
</tr>
<tr>
<td>Data Element</td>
<td>Sub Data Element</td>
<td>Element Definition</td>
<td>Occurrence</td>
<td>Obligation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>------------------</td>
<td>--------------------</td>
<td>------------</td>
<td>------------</td>
<td>-------------</td>
</tr>
<tr>
<td>SaaMessage</td>
<td></td>
<td>AIXM SaaMessage Object</td>
<td>1</td>
<td>Required</td>
<td>An AIXM object that contains the definition of the SAA. Detailed in Appendix A.</td>
</tr>
</tbody>
</table>

**GetSaaNames Response**  
A request to get a list of all active and pending SAA names. There are no sub data elements.

<table>
<thead>
<tr>
<th>Occur</th>
<th>Obligation</th>
<th>Description</th>
</tr>
</thead>
</table>

**GetSaaNames Request**  
A response containing the keys for all active and pending SAAs.

<table>
<thead>
<tr>
<th>SaaKey</th>
<th>SaaKeyType</th>
<th>Occurrence</th>
<th>Obligation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Unbounded</td>
<td>Optional</td>
<td>The key for an SAA.</td>
</tr>
</tbody>
</table>

**GetSaaUuid Request**  
The given SAA name and type used to get the UUID

<table>
<thead>
<tr>
<th>SaaSUuidElement</th>
<th>SaaSUuidElement</th>
<th>Occurrence</th>
<th>Obligation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1</td>
<td>Required</td>
<td>The SAA name and type for getting the UUID.</td>
</tr>
</tbody>
</table>

**GetSaaUuid Response**  
The response containing the UUID for the given SAA.

<table>
<thead>
<tr>
<th>uuid</th>
<th>String</th>
<th>Occurrence</th>
<th>Obligation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1</td>
<td>Required</td>
<td>The Universally Unique Identifier of the SAA. A 32 character string that uniquely identifies an AIXM feature within a codespace.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>codespace</th>
<th>String</th>
<th>Occurrence</th>
<th>Obligation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1</td>
<td>Required</td>
<td>A URN used to resolve the UUID.</td>
</tr>
</tbody>
</table>

**GetUnits Request**  
A request to get a list of all Units. There are no sub data elements.

**GetUnits Response**  
A list of Unit data.

<table>
<thead>
<tr>
<th>UnitKey</th>
<th>UnitKeyType</th>
<th>Occurrence</th>
<th>Obligation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Unbounded</td>
<td>Optional</td>
<td>The key information for a Unit.</td>
</tr>
</tbody>
</table>

**ValidateSaa Request**  
An SAA definition sent in to be validated.

<table>
<thead>
<tr>
<th>SaaMessage</th>
<th>AIXM SaaMessage Object</th>
<th>Occurrence</th>
<th>Obligation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1</td>
<td>Required</td>
<td>An AIXM object that contains the definition of the SAA. Detailed in Appendix A.</td>
</tr>
</tbody>
</table>

**ValidateSaa Response**  
Whether a submitted SAA is valid or not and why.

<table>
<thead>
<tr>
<th>success</th>
<th>Boolean</th>
<th>Occurrence</th>
<th>Obligation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1</td>
<td>Required</td>
<td>Indicates success.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>message</th>
<th>String</th>
<th>Occurrence</th>
<th>Obligation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1</td>
<td>Optional</td>
<td>The reason why an SAA is invalid.</td>
</tr>
</tbody>
</table>
4.2.2 Messages

TBL 4-4 defines the messages sent between the SAA SR Put Service and its users.

**TBL 4-4**

**SAA SR Put Service Message Descriptions**

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition</th>
<th>Data Elements (Parts)</th>
<th>Direction</th>
</tr>
</thead>
<tbody>
<tr>
<td>UpdateSaaRequest</td>
<td>Used by the SAA Project Repository to submit an SAA update.</td>
<td>UpdateSaaRequest</td>
<td>SAA Project Repository -&gt; SAA SR</td>
</tr>
<tr>
<td>UpdateSaaResponse</td>
<td>Used by the SAA SR Service to respond with if an SAA update is successful or not.</td>
<td>UpdateSaaResponse</td>
<td>SAA SR -&gt; SAA Project Repository</td>
</tr>
<tr>
<td>InsertSaaRequest</td>
<td>Used by the SAA Project Repository to submit an SAA insert.</td>
<td>InsertSaaRequest</td>
<td>SAA Project Repository -&gt; SAA SR</td>
</tr>
<tr>
<td>InsertSaaResponse</td>
<td>Used by the SAA SR Service to respond with if an SAA insert is successful or not.</td>
<td>InsertSaaResponse</td>
<td>SAA SR -&gt; SAA Project Repository</td>
</tr>
<tr>
<td>InsertUnitRequest</td>
<td>Used by the SAA Project Repository to submit a Unit insert.</td>
<td>InsertUnitRequest</td>
<td>SAA Project Repository -&gt; SAA SR</td>
</tr>
<tr>
<td>InsertUnitResponse</td>
<td>Used by the SAA SR Service to respond with if a Unit insert is successful or not.</td>
<td>InsertUnitResponse</td>
<td>SAA SR -&gt; SAA Project Repository</td>
</tr>
<tr>
<td>Fault</td>
<td>Used by the SAA SR Service when a call is invalid or an error occurs.</td>
<td>Fault</td>
<td>SAA SR -&gt; Service User</td>
</tr>
</tbody>
</table>

Table 4-5 defines the messages sent between the SAA SR Get Service and its users.

**TBL 4-5**

**SAA SR Get Service Message Descriptions**

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition</th>
<th>Data Elements (Parts)</th>
<th>Direction</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetSaaNamesRequest</td>
<td>Used by a user to request a list of SAA names.</td>
<td>GetSaaNamesRequest</td>
<td>Service User -&gt; SAA SR</td>
</tr>
<tr>
<td>GetSaaNamesResponse</td>
<td>Used by the SAA SR Service to return a list of SAA names.</td>
<td>GetSaaNamesResponse</td>
<td>SAA SR -&gt; Service User</td>
</tr>
<tr>
<td>GetUnitsRequest</td>
<td>Used by a user to request a list of Units.</td>
<td>GetUnitsRequest</td>
<td>Service User -&gt; SAA SR</td>
</tr>
<tr>
<td>GetUnitsResponse</td>
<td>Used by the SAA SR Service to return a list of Units.</td>
<td>GetUnitsResponse</td>
<td>SAA SR -&gt; Service User</td>
</tr>
<tr>
<td>GetSaaSByUuidRequest</td>
<td>Used by a user to request an SAA with a given UUID.</td>
<td>GetSaaSByUuidRequest</td>
<td>Service User -&gt; SAA SR</td>
</tr>
<tr>
<td>GetSaaSByUuidResponse</td>
<td>Used by the SAA SR Service to return an SAA with a given UUID.</td>
<td>GetSaaSByUuidResponse</td>
<td>SAA SR -&gt; Service User</td>
</tr>
<tr>
<td>Name</td>
<td>Definition</td>
<td>Data Elements (Parts)</td>
<td>Direction</td>
</tr>
<tr>
<td>----------------------</td>
<td>---------------------------------------------------------------------------</td>
<td>-----------------------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>GetSaaSUuidRequest</td>
<td>Used by a user to request the UUID of an SAA with the given name and type.</td>
<td>GetSaaSUuidRequest</td>
<td>Service User -&gt; SAA SR</td>
</tr>
<tr>
<td>GetSaaSUuidResponse</td>
<td>Used by the SAA SR Service to return the UUID of an SAA.</td>
<td>GetSaaSUuidResponse</td>
<td>SAA SR -&gt; Service User</td>
</tr>
<tr>
<td>GetUuidRequest</td>
<td>Used by a user to request a set of unused UUIDs.</td>
<td>GetUuidRequest</td>
<td>Service User -&gt; SAA SR</td>
</tr>
<tr>
<td>GetUuidResponse</td>
<td>Used by the SAA SR Service to return a set of unused UUIDs.</td>
<td>GetUuidResponse</td>
<td>SAA SR -&gt; Service User</td>
</tr>
<tr>
<td>ValidateSaaSRequest</td>
<td>Used by a user to validate a submitted SAA definition.</td>
<td>ValidateSaaSRequest</td>
<td>Service User -&gt; SAA SR</td>
</tr>
<tr>
<td>ValidateSaaSResponse</td>
<td>Used by the SAA SR Service to return whether or not a submitted SAA definition is valid.</td>
<td>ValidateSaaSResponse</td>
<td>SAA SR -&gt; Service User</td>
</tr>
<tr>
<td>Fault</td>
<td>Used by the SAA SR Service when a call is invalid or an error occurs.</td>
<td>Fault</td>
<td>SAA SR -&gt; Service User</td>
</tr>
</tbody>
</table>

4.2.2.1 Error Codes

The error codes used in response messages from the service are To Be Determined (TBD), but will be listed in TBL 4-6.

### TBL 4-6 SAA SR Get Service Message Descriptions

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

4.2.3 Operations

4.2.3.1 SAA SR Put Service Operations

The following tables describe the operations provided by the SAA SR Put Service.

### TBL 4-7 insertSaa Operation Description

<table>
<thead>
<tr>
<th>Operation Name</th>
<th>insertSaa</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Allows the user to submit a static definition for a new SAA to the SAA Static Repository.</td>
</tr>
<tr>
<td>MEP</td>
<td>In-Out (FAA-STD-066 Message Exchange Pattern Taxonomy)</td>
</tr>
<tr>
<td>Precondition</td>
<td>Service consumer, the SAA Project Repository, has a correctly implemented application for creating the SAA insert request. Service provider SAA SR Service has the service consumer identified as an authorized service consumer.</td>
</tr>
<tr>
<td>Input</td>
<td>Message InsertSaaSRequest.</td>
</tr>
<tr>
<td>Output</td>
<td>Message InsertSaaSResponse, or SOAP Fault Message.</td>
</tr>
<tr>
<td>Effect</td>
<td>The submitted SAA static definition is inserted into the SAA Static Repository.</td>
</tr>
<tr>
<td>Faults</td>
<td>TBD, but will be identified from the pre-defined error messages in section 4.2.2.1.</td>
</tr>
</tbody>
</table>
### updateSaa Operation Description

<table>
<thead>
<tr>
<th>Operation Name</th>
<th>updateSaa</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Allows the user to submit a static definition for an existing SAA to the SAA Static Repository.</td>
</tr>
<tr>
<td>MEP</td>
<td>In-Out (FAA-STD-066 Message Exchange Pattern Taxonomy)</td>
</tr>
<tr>
<td>Precondition</td>
<td>Service consumer, the SAA Project Repository, has a correctly implemented application for creating the SAA update request. Service provider SAA SR Service has the service consumer identified as an authorized service consumer.</td>
</tr>
<tr>
<td>Input</td>
<td>Message UpdateSaaRequest.</td>
</tr>
<tr>
<td>Output</td>
<td>Message UpdateSaaResponse, or SOAP Fault Message.</td>
</tr>
<tr>
<td>Effect</td>
<td>The submitted SAA static definition is updated into the SAA Static Repository.</td>
</tr>
<tr>
<td>Faults</td>
<td>TBD, but will be identified from the pre-defined error messages in section 4.2.2.1.</td>
</tr>
</tbody>
</table>

### insertUnit Operation Description

<table>
<thead>
<tr>
<th>Operation Name</th>
<th>insertUnit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Allows the user to submit a definition for a new Unit to the SAA Static Repository.</td>
</tr>
<tr>
<td>MEP</td>
<td>In-Out (FAA-STD-066 Message Exchange Pattern Taxonomy)</td>
</tr>
<tr>
<td>Precondition</td>
<td>Service consumer, the SAA Project Repository, has a correctly implemented application for creating the Unit insert request. Service provider SAA SR Service has the service consumer identified as an authorized service consumer.</td>
</tr>
<tr>
<td>Input</td>
<td>Message InsertUnitRequest.</td>
</tr>
<tr>
<td>Output</td>
<td>Message InsertUnitResponse, or SOAP Fault Message.</td>
</tr>
<tr>
<td>Effect</td>
<td>The submitted Unit definition is inserted into the SAA Static Repository.</td>
</tr>
<tr>
<td>Faults</td>
<td>TBD, but will be identified from the pre-defined error messages in section 4.2.2.1.</td>
</tr>
</tbody>
</table>

### 4.2.3.2 SAA SR Get Service Operations

The following tables describe the operations provided by the SAA SR Get Service.

#### getNewUuid Operation Description

<table>
<thead>
<tr>
<th>Operation Name</th>
<th>getNewUuid</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Allows the user to request a given number of unused UUIDs.</td>
</tr>
<tr>
<td>MEP</td>
<td>In-Out (FAA-STD-066 Message Exchange Pattern Taxonomy)</td>
</tr>
<tr>
<td>Precondition</td>
<td>Service provider SAA SR Service has the service consumer identified as an authorized service consumer.</td>
</tr>
<tr>
<td>Input</td>
<td>Message GetUuidRequest.</td>
</tr>
<tr>
<td>Output</td>
<td>Message GetUuidResponse, or SOAP Fault Message.</td>
</tr>
<tr>
<td>Effect</td>
<td>A set of unused UUIDs are sent to the service consumer for their use.</td>
</tr>
<tr>
<td>Faults</td>
<td>TBD, but will be identified from the pre-defined error messages in section 4.2.2.1.</td>
</tr>
</tbody>
</table>

#### getSaaByUuid Operation Description

<table>
<thead>
<tr>
<th>Operation Name</th>
<th>getSaaByUuid</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Allows the service consumer to get the static definition of the SAA with a given UUID for a given time period.</td>
</tr>
<tr>
<td>MEP</td>
<td>In-Out (FAA-STD-066 Message Exchange Pattern Taxonomy)</td>
</tr>
<tr>
<td>Precondition</td>
<td>Service provider SAA SR Service has the service consumer identified as an authorized service consumer.</td>
</tr>
<tr>
<td>Input</td>
<td>Message GetSaaByUuidRequest.</td>
</tr>
<tr>
<td>Output</td>
<td>Message GetSaaByUuidResponse, or SOAP Fault Message.</td>
</tr>
</tbody>
</table>
### Effect
The static definition of the requested SAA for the requested time period is returned to the service consumer.

### Faults
TBD, but will be identified from the pre-defined error messages in section 4.2.2.1.

### TBL 4.12
getSaaNames Operation Description

<table>
<thead>
<tr>
<th>Operation Name</th>
<th>getSaaNames</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Allows the service consumer to get a collection of names for all active and pending SAAs in the Static Repository.</td>
</tr>
<tr>
<td>MEP</td>
<td>In-Out (FAA-STD-066 Message Exchange Pattern Taxonomy)</td>
</tr>
<tr>
<td>Precondition</td>
<td>Service provider SAA SR Service has the service consumer identified as an authorized service consumer.</td>
</tr>
<tr>
<td>Input</td>
<td>Message GetSaaNamesRequest.</td>
</tr>
<tr>
<td>Output</td>
<td>Message GetSaaNamesResponse, or SOAP Fault Message.</td>
</tr>
<tr>
<td>Effect</td>
<td>A collection of all SAA names in the SAA Static Repository is returned to the service consumer.</td>
</tr>
<tr>
<td>Faults</td>
<td>TBD, but will be identified from the pre-defined error messages in section 4.2.2.1.</td>
</tr>
</tbody>
</table>

### TBL 4.13
getSaaSUID Operation Description

<table>
<thead>
<tr>
<th>Operation Name</th>
<th>getSaaSUID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Allows the service consumer to submit key information of an SAA (name and type) to retrieve the SAA’s UUID.</td>
</tr>
<tr>
<td>MEP</td>
<td>In-Out (FAA-STD-066 Message Exchange Pattern Taxonomy)</td>
</tr>
<tr>
<td>Precondition</td>
<td>Service provider SAA SR Service has the service consumer identified as an authorized service consumer.</td>
</tr>
<tr>
<td>Input</td>
<td>Message GetSaaSUIDRequest.</td>
</tr>
<tr>
<td>Output</td>
<td>Message GetSaaSUIDResponse, or SOAP Fault Message.</td>
</tr>
<tr>
<td>Effect</td>
<td>The UUID of the SAA with the matching key information is returned to the service consumer.</td>
</tr>
<tr>
<td>Faults</td>
<td>TBD, but will be identified from the pre-defined error messages in section 4.2.2.1.</td>
</tr>
</tbody>
</table>

### TBL 4.14
getUnits Operation Description

<table>
<thead>
<tr>
<th>Operation Name</th>
<th>getUnits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Allows the service consumer to get a collection of all active and pending Unit objects in the SAA Static Repository.</td>
</tr>
<tr>
<td>MEP</td>
<td>In-Out (FAA-STD-066 Message Exchange Pattern Taxonomy)</td>
</tr>
<tr>
<td>Precondition</td>
<td>Service provider SAA SR Service has the service consumer identified as an authorized service consumer.</td>
</tr>
<tr>
<td>Input</td>
<td>Message GetUnitsRequest.</td>
</tr>
<tr>
<td>Output</td>
<td>Message GetUnitsResponse, or SOAP Fault Message.</td>
</tr>
<tr>
<td>Effect</td>
<td>A collection of all the Unit objects in the SAA Static Repository is returned to the service consumer.</td>
</tr>
<tr>
<td>Faults</td>
<td>TBD, but will be identified from the pre-defined error messages in section 4.2.2.1.</td>
</tr>
</tbody>
</table>

### TBL 4.15
validateSaaS Operation Description

<table>
<thead>
<tr>
<th>Operation Name</th>
<th>validateSaaS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Allows the user to submit a static definition of an SAA and have the SAA SR Service validate it.</td>
</tr>
<tr>
<td>MEP</td>
<td>In-Out (FAA-STD-066 Message Exchange Pattern Taxonomy)</td>
</tr>
<tr>
<td>Precondition</td>
<td>Service provider SAA SR Service has the service consumer identified as an authorized service consumer.</td>
</tr>
</tbody>
</table>
### 4.2.4 Lists of Interfaces

The SAA Project Repository submits static definitions of SAAs to the SAA Static Repository via web service-based message transfer.

Other service consumers submit requests for information related to SAAs to the SAA SR Service via web service-based message transfer.

The SAA SR Notification Service publishes notifications of SAA updates to a JMS topic. Users that subscribe to that JMS topic then receive these notifications. Note: The interface description for the JMS topic is not written, but will be referenced here when it is.

TBL 4-16 below lists the two interfaces being implemented in the SAA SR Service.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>SaaStaticRepositoryPutPort</td>
<td>Allows the SAA Project Repository to submit SAAs and Units to the SAA SR Service to store in the Static Repository.</td>
<td>insertSaa, updateSaa, insertUnit</td>
</tr>
<tr>
<td>SaaStaticRepositoryGetPort</td>
<td>Allows service consumers to retrieve information on SAAs as well as have the service evaluate a SAA definition they submit to see if it is a valid definition.</td>
<td>getNewUuid, getSaByUuid, getSaaNames, getSaaUuid, getUnits, validateSaa</td>
</tr>
<tr>
<td>SaaStaticRepositoryNotificationPort</td>
<td>Allows users to subscribe to JMS topics in order to receive notifications that new or modified SAAs are available through the SAA SR Get Service.</td>
<td></td>
</tr>
</tbody>
</table>

### 4.3 Service Implementation

The SAA SR Service complies with cited SWIM Segment 1 Compliance standards for web service protocols listed in Section 2.1, Standards for FAA SWIM Program Compliance.

The SAA SR Service will be hosted at the Mike Monroney Aeronautical Center in Oklahoma City. The backup of the system will be hosted at the William J. Hughes Technical Center in Atlantic City. The hardware and its configuration is currently TBD, so the specific end point addresses are also TBD.

The SAA SR Service interface complies with web services interface standards Web Service Definition Language (WSDL), and SOAP. The SAA SR Service transports messages over a HyperText Transfer Protocol with SSL (HTTPS) session.

### 4.3.1 End Points

The following sections describe the associated interface, communication protocol, messaging protocol, and network address.
4.3.1.1 SAA SR Put Service End Points

4.3.1.1.1 Associated Interface
For all SAA SR Put Service end points, the interface name is SaaSStaticRepositoryPutPort.

4.3.1.1.2 Communication Protocol
For all SAA SR Put Service end points, the communications protocol is HTTPS.

4.3.1.1.3 Messaging Protocol
For all SAA SR Put Service end points, the messaging protocol is SOAP.

4.3.1.1.4 Network Address
The network address for the SAA SR Put Service interface is currently TBD.

4.3.1.1.5 End Point-Specific Qualities of Service

\textit{TBL 4-17}
SAA SR Put Service Quality of Service Parameters

<table>
<thead>
<tr>
<th>QoS Parameter</th>
<th>Definition</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Response Time for insertSaa and updateSaa operations.</td>
<td>A measure of the longest average time period required to complete a service request, from the time the requester invokes the service to the time the requester receives the last byte of the response.</td>
<td>A 95\textsuperscript{th} percentile response time of 10 seconds per airspace volume per timeslice.</td>
</tr>
<tr>
<td>Response Time for insertUnit operation.</td>
<td>A measure of the longest average time period required to complete a service request, from the time the requester invokes the service to the time the requester receives the last byte of the response.</td>
<td>A 95\textsuperscript{th} percentile response time of 10 seconds.</td>
</tr>
</tbody>
</table>

4.3.1.2 SAA SR Get Service End Points

4.3.1.2.1 Associated Interface
For all SAA SR Get Service end points, the interface name is SaaSStaticRepositoryGetPort.

4.3.1.2.2 Communication Protocol
For all SAA SR Get Service end points, the communications protocol is HTTPS.

4.3.1.2.3 Messaging Protocol
For all SAA SR Get Service end points, the messaging protocol is SOAP.

4.3.1.2.4 Network Address
The network address for the SAA SR Get Service interface is currently TBD.

4.3.1.2.5 End Point-Specific Qualities of Service

\textit{TBL 4-18}
SAA SR Service Quality of Service Parameters

<table>
<thead>
<tr>
<th>QoS Parameter</th>
<th>Definition</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Response Time for get</td>
<td>A measure of the longest average time period</td>
<td>A 95\textsuperscript{th} percentile response time</td>
</tr>
<tr>
<td>QoS Parameter</td>
<td>Definition</td>
<td>Value</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>services</td>
<td>required to complete a service request, from the time the requester invokes the service to the time the requester receives the last byte of the response.</td>
<td>of 5 seconds.</td>
</tr>
<tr>
<td>Response Time for the validateSaa operation</td>
<td>A measure of the longest average time period required to complete a service request, from the time the requester invokes the service to the time the requester receives the last byte of the response.</td>
<td>A 95th percentile response time of 10 seconds per airspace volume per timeslice.</td>
</tr>
</tbody>
</table>

4.3.1.3 SAA SR Notification Service End Points

4.3.1.3.1 Associated Interface
For all SAA SR Notification Service end points, the interface name is SaaStaticRepositoryNotificationPort.

4.3.1.3.2 Communication Protocol
For all SAA SR Notification Service end points, the communications protocol is TCP/IP (with SSL/TLS).

4.3.1.3.3 Messaging Protocol
For all SAA SR Notification Service end points, the messaging protocol is SOAP.

The JMS client must be a FUSE message broker in order to receive updates from the service.

4.3.1.3.4 Network Address
The network address for the SAA SR Notification Service interface is currently TBD.

4.3.1.3.5 End Point-Specific Qualities of Service

TBL 4-19
SAA SR Notification Service Quality of Service Parameters

<table>
<thead>
<tr>
<th>QoS Parameter</th>
<th>Definition</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Response Time</td>
<td>A measure of the longest average time period required to complete a service request, from the time the requester invokes the service to the time the requester receives the last byte of the response.</td>
<td>Responds within 5 seconds or less of an update, insert, or invalidation of an SAA.</td>
</tr>
</tbody>
</table>
Appendix A. Accepted Data Elements

The accepted AIXM elements for the SAA SR Service are defined in Appendix C of the AIM SAA Management Web Services IRD.

The accepted ways to define geometries for SAAs in GML are defined in Appendix D of the AIM SAA Management Web Services IRD.