

# Testbed-12 Aviation Architecture Engineering Report

# Table of Contents

1. Introduction .....	5
1.1. Scope .....	5
1.2. Document contributor contact points .....	5
1.3. Future Work .....	5
1.4. Foreword .....	5
2. References .....	6
3. Terms and definitions .....	7
3.1. Terms and Definitions .....	7
3.2. Abbreviated terms .....	7
4. Overview .....	9
5. Component Descriptions .....	12
5.1. Data Broker [F006] .....	12
5.1.1. Status Quo .....	12
5.1.2. Requirements Statement .....	12
5.1.3. Functional Overview .....	13
5.1.4. Deployment Characteristics .....	13
5.1.5. Challenges .....	13
5.1.6. Accomplishments .....	14
5.2. Catalog [F003] .....	14
5.2.1. Status Quo .....	14
5.2.2. Requirements Statement .....	14
5.2.3. Functional Overview .....	15
5.2.4. Deployment Characteristics .....	16
5.2.5. Challenges .....	16
5.2.6. Accomplishments .....	16
5.3. Aviation Client [F004] (Aviation Client for Data Broker and CSW) .....	16
5.3.1. Status Quo .....	16
5.3.2. Requirements Statement .....	16
5.3.3. Functional Overview .....	17
5.3.4. Deployment Characteristics .....	18
5.3.5. Challenges .....	18
5.3.6. Accomplishments .....	18
5.4. Aviation Client [F008] (Aviation Client for Asynchronous Messaging) .....	19
5.4.1. Status Quo .....	19
5.4.2. Requirements Statement .....	19
5.4.3. Functional Overview .....	19
5.4.4. Deployment Characteristics .....	19
5.4.5. Challenges .....	19

5.4.6. Accomplishments .....	20
5.5. Asynchronous Messaging Server [F009] .....	20
5.5.1. Status Quo .....	20
5.5.2. Requirements Statement .....	21
5.5.3. Functional Overview .....	21
5.5.4. Deployment Characteristics .....	22
5.5.5. Challenges .....	22
5.5.6. Accomplishments .....	22
5.6. AMXM WFS [F005] .....	22
5.6.1. Status Quo .....	22
5.6.2. Requirements Statement .....	22
5.6.3. Functional Overview .....	23
5.6.4. Deployment Characteristics .....	23
5.6.5. Challenges .....	23
5.6.6. Accomplishments .....	23
5.7. AIXM WFS-TE [E006] .....	24
5.7.1. Status Quo .....	24
5.7.2. Requirements Statement .....	24
5.7.3. Functional Overview .....	24
5.7.4. Deployment Characteristics .....	25
5.7.5. Challenges .....	25
5.7.6. Accomplishments .....	25
5.8. FIXM WFS [F012] .....	25
5.8.1. Status Quo .....	26
5.8.2. Requirements Statement .....	26
5.8.3. Functional Overview .....	26
5.8.4. Deployment Characteristics .....	27
5.8.5. Challenges .....	27
5.8.6. Accomplishments .....	27
5.9. WMS/WMTS/WCS [F007] .....	27
5.9.1. Status Quo .....	27
5.9.2. Requirements Statement .....	27
5.9.3. Functional Overview .....	27
5.9.4. Deployment Characteristics .....	28
5.9.5. Challenges .....	28
5.9.6. Accomplishments .....	28
6. Engineering Report Topics .....	29
6.1. Catalog ER .....	29
6.1.1. Problem Statement .....	29
6.1.2. Overview .....	29
6.1.3. Business Value .....	30

6.1.4. Work Conducted .....	30
6.1.5. Conclusions .....	30
6.2. Data Broker ER .....	30
6.2.1. Problem Statement.....	30
6.2.2. Overview .....	31
6.2.3. Business Value.....	31
6.2.4. Work Conducted .....	31
6.2.5. Conclusions .....	32
6.3. Asynchronous Messaging ER.....	32
6.3.1. Problem Statement.....	32
6.3.2. Overview .....	32
6.3.3. Business Value.....	33
6.3.4. Work Conducted .....	33
6.3.5. Conclusions .....	34
6.4. SBVR ER .....	34
6.4.1. Problem Statement.....	34
6.4.2. Overview .....	34
6.4.3. Business Value.....	34
6.4.4. Work Conducted .....	35
6.4.5. Conclusions .....	35
6.5. Aviation Security ER .....	35
6.5.1. Problem Statement.....	35
6.5.2. Overview .....	36
6.5.3. Business Value.....	36
6.5.4. Work Conducted .....	36
6.5.5. Conclusions .....	36
6.6. Aviation Semantics ER .....	37
6.6.1. Problem Statement.....	37
6.6.2. Overview .....	37
6.6.3. Business Value.....	37
6.6.4. Work Conducted .....	37
6.6.5. Conclusions .....	37
6.7. FIXM GML ER .....	38
6.7.1. Problem Statement.....	38
6.7.2. Overview .....	38
6.7.3. Business Value.....	38
6.7.4. Work Conducted .....	39
6.7.5. Conclusions .....	39
7. Demonstration Scenarios.....	40
7.1. Scenario 0.....	40
7.1.1. Components .....	40

7.1.2. Status Quo .....	40
7.1.3. Scenario 0 Sequence .....	40
7.1.4. Result .....	41
7.1.5. Benefits .....	43
7.2. Scenario 1 .....	43
7.2.1. Components .....	43
7.2.2. Status Quo .....	43
7.2.3. Scenario 1 Sequence .....	43
7.2.4. Result .....	44
7.2.5. Benefits .....	45
7.3. Scenario 2 .....	45
7.3.1. Components .....	46
7.3.2. Status Quo .....	46
7.3.3. Scenario 2 Sequence .....	46
7.3.4. Result .....	47
7.3.5. Benefits .....	50
7.4. Scenario 3 .....	50
7.4.1. Components .....	51
7.4.2. Status Quo .....	51
7.4.3. Scenario 3 Sequence .....	51
7.4.4. Results .....	52
7.4.5. Benefits .....	54
8. Lessons Learned .....	55
8.1. Web Service Architecture .....	55
8.2. Asynchronous Messaging Architecture .....	55
8.3. Data Broker .....	55
8.4. Catalog .....	55
8.5. WFS-TE .....	56
8.6. Aerodrome Mapping Exchange Model (AMXM) .....	56
Appendix A: Revision History .....	57

Publication Date: 2017-06-15

Approval Date: 2016-09-15

Posted Date: 2016-07-31

Reference number of this document: OGC 16-018

Reference URL for this document: <http://www.opengis.net/doc/PER/t12-E005>

Category: Public Engineering Report

Editor: Charles Chen, Skymantics

Contributor: Thomas Forbes, Snowflake Software

Title: Testbed-12 Aviation Architecture ER

---

## **OGC Engineering Report**

### **COPYRIGHT**

Copyright © 2017 Open Geospatial Consortium. To obtain additional rights of use, visit <http://www.opengeospatial.org/>

### **WARNING**

This document is not an OGC Standard. This document is an OGC Public Engineering Report created as a deliverable in an OGC Interoperability Initiative and is not an official position of the OGC membership. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an OGC Standard. Further, any OGC Engineering Report should not be referenced as required or mandatory technology in procurements. However, the discussions in this document could very well lead to the definition of an OGC Standard.

Document type: OGC Engineering Report

Document subtype:

Document stage: Approved for public release

Document language: English

---

## LICENSE AGREEMENT

Permission is hereby granted by the Open Geospatial Consortium, ("Licensor"), free of charge and subject to the terms set forth below, to any person obtaining a copy of this Intellectual Property and any associated documentation, to deal in the Intellectual Property without restriction (except as set forth below), including without limitation the rights to implement, use, copy, modify, merge, publish, distribute, and/or sublicense copies of the Intellectual Property, and to permit persons to whom the Intellectual Property is furnished to do so, provided that all copyright notices on the intellectual property are retained intact and that each person to whom the Intellectual Property is furnished agrees to the terms of this Agreement.

If you modify the Intellectual Property, all copies of the modified Intellectual Property must include, in addition to the above copyright notice, a notice that the Intellectual Property includes modifications that have not been approved or adopted by LICENSOR.

THIS LICENSE IS A COPYRIGHT LICENSE ONLY, AND DOES NOT CONVEY ANY RIGHTS UNDER ANY PATENTS THAT MAY BE IN FORCE ANYWHERE IN THE WORLD. THE INTELLECTUAL PROPERTY IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE DO NOT WARRANT THAT THE FUNCTIONS CONTAINED IN THE INTELLECTUAL PROPERTY WILL MEET YOUR REQUIREMENTS OR THAT THE OPERATION OF THE INTELLECTUAL PROPERTY WILL BE UNINTERRUPTED OR ERROR FREE. ANY USE OF THE INTELLECTUAL PROPERTY SHALL BE MADE ENTIRELY AT THE USER'S OWN RISK. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR ANY CONTRIBUTOR OF INTELLECTUAL PROPERTY RIGHTS TO THE INTELLECTUAL PROPERTY BE LIABLE FOR ANY CLAIM, OR ANY DIRECT, SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM ANY ALLEGED INFRINGEMENT OR ANY LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR UNDER ANY OTHER LEGAL THEORY, ARISING OUT OF OR IN CONNECTION WITH THE IMPLEMENTATION, USE, COMMERCIALIZATION OR PERFORMANCE OF THIS INTELLECTUAL PROPERTY.

This license is effective until terminated. You may terminate it at any time by

destroying the Intellectual Property together with all copies in any form. The license will also terminate if you fail to comply with any term or condition of this Agreement. Except as provided in the following sentence, no such termination of this license shall require the termination of any third party end-user sublicense to the Intellectual Property which is in force as of the date of notice of such termination. In addition, should the Intellectual Property, or the operation of the Intellectual Property, infringe, or in LICENSOR's sole opinion be likely to infringe, any patent, copyright, trademark or other right of a third party, you agree that LICENSOR, in its sole discretion, may terminate this license without any compensation or liability to you, your licensees or any other party. You agree upon termination of any kind to destroy or cause to be destroyed the Intellectual Property together with all copies in any form, whether held by you or by any third party.

Except as contained in this notice, the name of LICENSOR or of any other holder of a copyright in all or part of the Intellectual Property shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Intellectual Property without prior written authorization of LICENSOR or such copyright holder. LICENSOR is and shall at all times be the sole entity that may authorize you or any third party to use certification marks, trademarks or other special designations to indicate compliance with any LICENSOR standards or specifications.

This Agreement is governed by the laws of the Commonwealth of Massachusetts. The application to this Agreement of the United Nations Convention on Contracts for the International Sale of Goods is hereby expressly excluded. In the event any provision of this Agreement shall be deemed unenforceable, void or invalid, such provision shall be modified so as to make it valid and enforceable, and as so modified the entire Agreement shall remain in full force and effect. No decision, action or inaction by LICENSOR shall be construed to be a waiver of any rights or remedies available to it.

None of the Intellectual Property or underlying information or technology may be downloaded or otherwise exported or reexported in violation of U.S. export laws and regulations. In addition, you are responsible for complying with any local laws in your jurisdiction which may impact your right to import, export or use the Intellectual Property, and you represent that you have complied with any regulations or registration procedures required by applicable law to make this license enforceable.



## **Abstract**

This Open Geospatial Consortium (OGC)® Engineering Report (ER) describes the architecture implemented in the OGC Testbed 12 Aviation thread. This report provides an overview of the technical architecture for the interoperable exchange of flight and aeronautical information using OGC services. The aviation architecture consists of multiple components developed by the Aviation thread, as well as specialized engineering reports per each work area. This report will provide an introduction to each work area and contain references to applicable reports. This report also describes the Aviation thread demonstration scenarios, outcomes, and benefits.

## **Business Value**

This ER documents the work activities of the OGC testbed relative to the topics of interest to the aviation thread sponsors. The outcome of this report may be referenced by future sponsors and the aviation community at large to promote the development of future work, adoption of OGC standards, and inform the public of research conducted on behalf of sponsors.

## **What does this ER mean for the Working Group and OGC in general**

This ER supports the overall objective of the OGC Aviation DWG to deliver recommendations on how to best use OGC standards in the context of System Wide Information Management (SWIM) and Air Traffic Management (ATM) information exchanges. The content of this ER is therefore expected to be used as appropriate by standardization organizations and/or governance bodies in charge of the development of ATM standards for SWIM.

## **Keywords**

ogcdocs, testbed-12, PubSub, asynchronous messaging, brokering, data broker, SBVR, Catalog, CSW, Registry, Aviation, AIXM, AMXM, FIXM, GML

# Chapter 1. Introduction

## 1.1. Scope

This OGC® document describes the architecture implemented in the OGC Testbed 12 Aviation thread. The document:

- Describes a high-level overview of the architecture, its components and their interactions;
- Contains a summary description of the various components within the architecture;
- Provides summaries for the various subject areas that the Aviation thread was concerned with and which are documented in detail in other Testbed 12 engineering reports;
- Documents lessons learned; and
- Describes the scenario as well as use cases that have been designed for the demonstration of the Aviation thread developments.

## 1.2. Document contributor contact points

All questions regarding this document should be directed to the editor or the contributors:

*Table 1. Contacts*

Name	Organization
Charles Chen	Skymanatics
Tom Forbes	Snowflake Software

## 1.3. Future Work

It is expected that the work contained within this document will inform future Aviation-related architectures and initiatives. For further details, see related OGC documents (16-024, 16-045, 16-017, 16-061, 16-040, 16-039, and 16-028).

## 1.4. Foreword

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. The Open Geospatial Consortium shall not be held responsible for identifying any or all such patent rights.

Recipients of this document are requested to submit, with their comments, notification of any relevant patent claims or other intellectual property rights of which they may be aware that might be infringed by any implementation of the standard set forth in this document, and to provide supporting documentation.

# Chapter 2. References

The following documents are referenced in this document. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. For undated references, the latest edition of the normative document referred to applies.

- OGC: OGC 06-121r9, OGC® Web Services Common Standard, 2010.

## NOTE

This OWS Common Standard contains a list of normative references that are also applicable to this Implementation Standard.

- OGC: [OGC 15-025] OGC Testbed 11 Aviation Architecture Engineering Report, 2016.
- OGC: OGC Testbed 12 RFQ Annex B, 6 November 2015
- OGC: [OGC 16-024] Testbed-12 Catalog Services for Aviation, 2017.
- OGC: [OGC 16-045] Testbed-12 Data Broker Engineering Report, 2017.
- OGC: [OGC 16-017] Testbed-12 Asynchronous Messaging for Aviation, 2017.
- OGC: [OGC 16-061] Testbed-12 SBVR Engineering Report, 2017.
- OGC: [OGC 16-040] Testbed-12 Aviation Security Engineering Report, 2017.
- OGC: [OGC 16-039] Testbed-12 Aviation Semantics Engineering Report, 2017.
- OGC: [OGC-16-028] Testbed-12 FIXM GML Engineering Report, 2017.
- OGC: [OGC 07-110r4] CSW-ebRIM Registry Service - Part 1: ebRIM profile of CSW, 2009.
- OGC: [OGC 07-144r4] CSW-ebRIM Registry Service - Part 2: Basic extension package (1.0.1), 2009.
- OGC: [OGC 13-131] OGC Publish/Subscribe Interface Standard 1.0 - Core, 2017.
- OGC: [OGC 13-133] OGC Publish/Subscribe Interface Standard 1.0 - SOAP Protocol Binding Extension, 2017.
- OGC: [OGC 09-026r2] OGC Filter Encoding 2.0 Standard - With Corrigendum, 2014.

# Chapter 3. Terms and definitions

For the purposes of this report, the definitions specified in Clause 4 of the OWS Common Implementation Standard [OGC 06-121r9] shall apply. In addition, the following terms and definitions apply.

## 3.1. Terms and Definitions

DEX - Harris system product used by the FAA SWIM program

NEMS - FAA SWIM infrastructure made up of a constellation of networked DEX systems.

PubSub - Publish Subscribe standard. When referenced as PubSub1.0, refers to OGC standard for PubSub.

## 3.2. Abbreviated terms

AIXM - Aeronautical Information Exchange Model

AMDB - Aerodrome Mapping Data Base

AMXM - Aerodrome Mapping Exchange Model

AMXS - Aerodrome Mapping Exchange Schema

API - Application Programming Interface

ATM - Air Traffic Management

BBOX - Bounding Box

CCI - Cross-Community Interoperability

COTS - Commercial Off the Shelf

CRUD - create, read, update, delete

DEX - Data Exchange

DNOTAM - Digital NOTAM

EFB - Electronic Flight Bag

EUROCAE - European Organisation for Civil Aviation Equipment

FAA - Federal Aviation Administration

FES - Feature Encoding Standard

FIXM - Flight Information Exchange Model

FNS - Federal NOTAM Service

FPS - Feature Portrayal Service

GML - Geography Markup Language

HTML - HyperText Markup Language

HTTP - HyperText Transfer Protocol

ICAO - International Civil Aviation Organization

ISO - International Organization for Standardization

JDK - Java Development Kit

NAS - National Airspace System

NEMS - NAS Enterprise Messaging Service

NOTAM - Notice to Airmen

OGC - Open Geospatial Consortium

RDF - Resource Description Framework

RTC - Radio Technical Commission for Aeronautics

SBVR - Semantics of Business Vocabulary and Business Rules  
SQL - Structured Query Language  
SWIM - System Wide Information Management  
UUID - Universally Unique Identifier  
WCS - Web Coverage Service  
WFS - Web Feature Service  
WFS-T - WFS-Transactional  
WMTS - Web Map Tile Service  
WPS - Web Processing Service  
WXXM - Weather Information Exchange Model  
XMI - XML Metadata Interchange  
XML - Extensible Markup Language  
XSLT - Extensible Stylesheet Language Transformations

# Chapter 4. Overview

The tasks for the Testbed 12 Aviation thread are to:

- Develop and document advanced use cases for an OGC Service Data Broker;
- Advance the use of Catalog Service for Web (CSW) in the context of SWIM Common Registry;
- Explore asynchronous messaging for geospatial queries of aviation data using OGC PubSub 1.0;
- Advance use of semantics for aviation;
- Advance use of SBVR using Testbed-11 SBVR work as a baseline;
- Advance use of OGC Web Service Security for the aviation domain; and
- Investigate the use of GML for the FIXM data model.

These tasks have been accomplished in Testbed 12. Chapters 5 & 6 provide summaries with references to further details. The components that enable the functionality required by each task are described in chapter 5. A high-level overview of the Aviation thread architecture and its components is given in the following figures. The Aviation thread architecture can be separated into three tiers (see Figure 1).

- The Client Tier contains the client applications.
- The Business Process Tier contains components that offer services on top of the Access Tier: Catalog, Data Broker, and PubSub Service Providers.
- The Access Tier contains Web Feature Services serving aeronautical data compliant to data models AIXM 5.1, AMXM 2.0, and FIXM 3.0.1 with GML. Also included for this thread is the first implementation of the WFS-TE for temporality extensions.

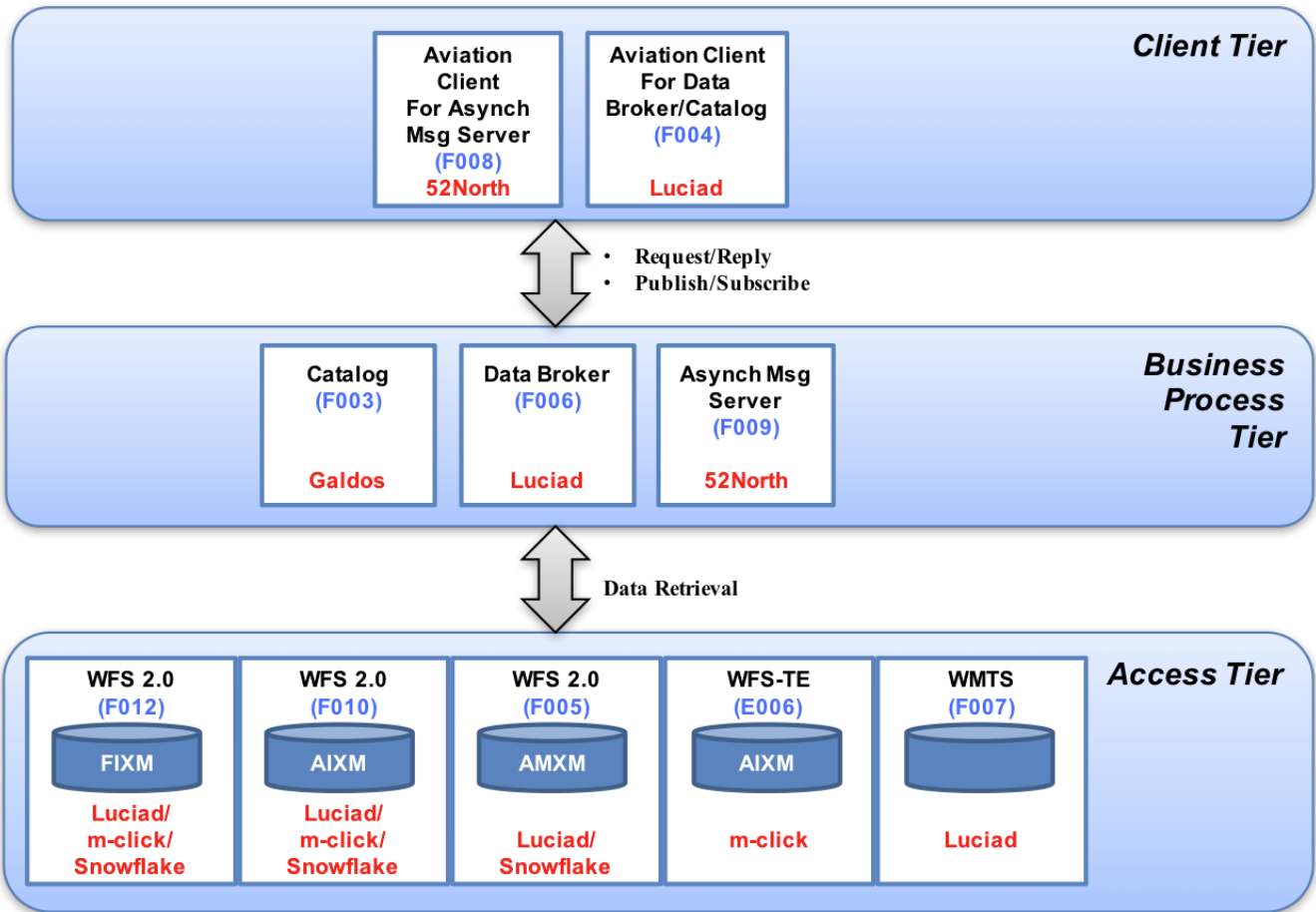


Figure 1. Testbed-12 Aviation Architecture - High-Level Overview

Figure 1 shows the links between the tiers and the general functionality that is invoked. It also shows which participants provided which components. A more detailed view of the interaction between the different components is shown in Figure 2. Each component is labeled with a deliverable number (e.g. F003 for Catalog). The "WPS" and "Tool SBVR-Schematron" components are not funded deliverables, but are carried over from OGC Testbed-11 for continuity of the SBVR effort in Testbed-12.

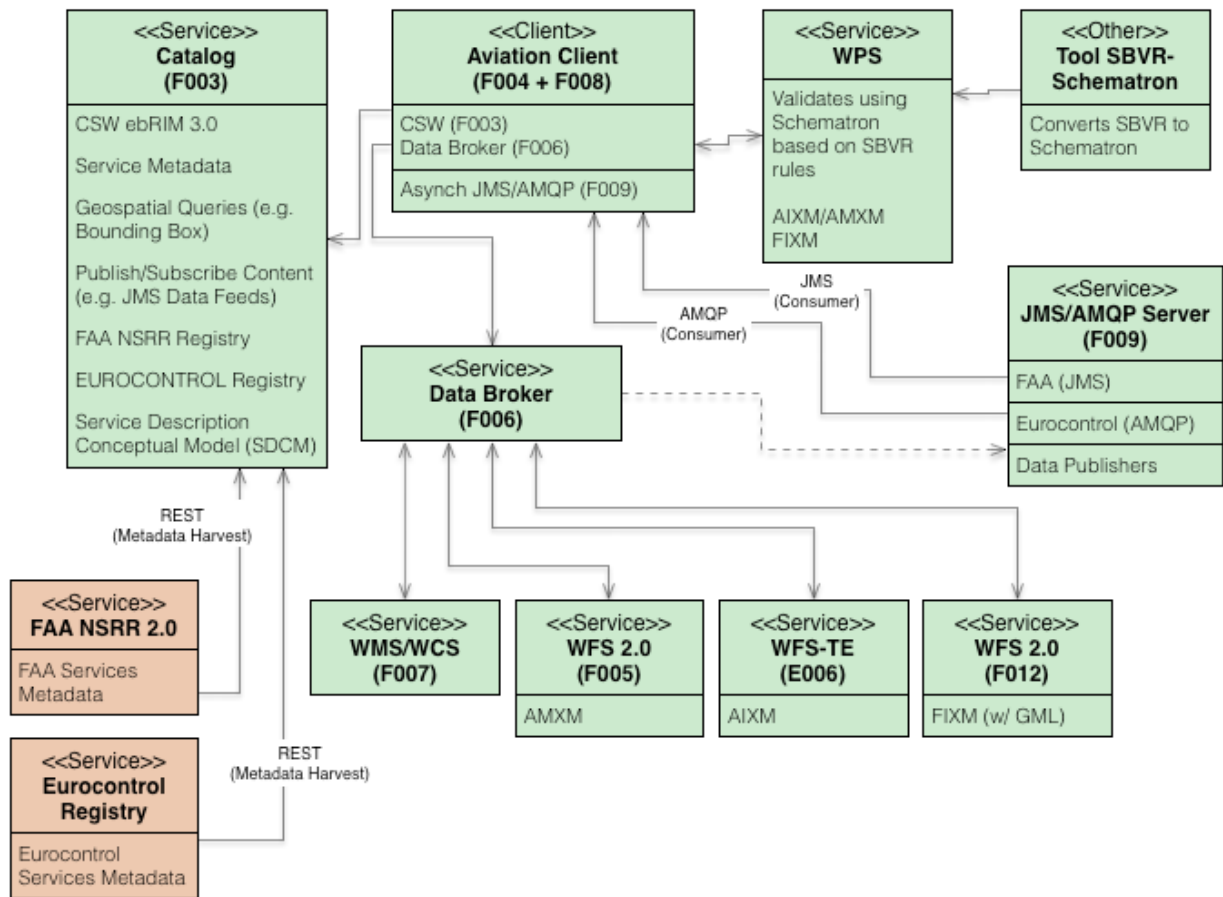


Figure 2. Components and their interactions

The Aviation Architecture references several activities occurring as a result of OGC Testbed 12. The Engineering Reports generated from these activities detail the developments, observations, and lessons learned. Chapter 6 provides overviews of each Testbed-12 Aviation thread Engineering Report.



# Chapter 5. Component Descriptions

The architecture for the Testbed 12 Aviation thread consists of components for each topic referenced in the Testbed 12 RFQ Annex B Section 4.11 Aviation Summary.

## 5.1. Data Broker [F006]

### 5.1.1. Status Quo

In OGC Testbed 11, a Data Broker concept was developed using the WFS 2.0 specification with augmented features to enable a data chain of cascading web service requests, in which the Data Broker would provide the client-interfacing web service and execute additional requests "behind-the-scenes" to other WFS instances to obtain required data.

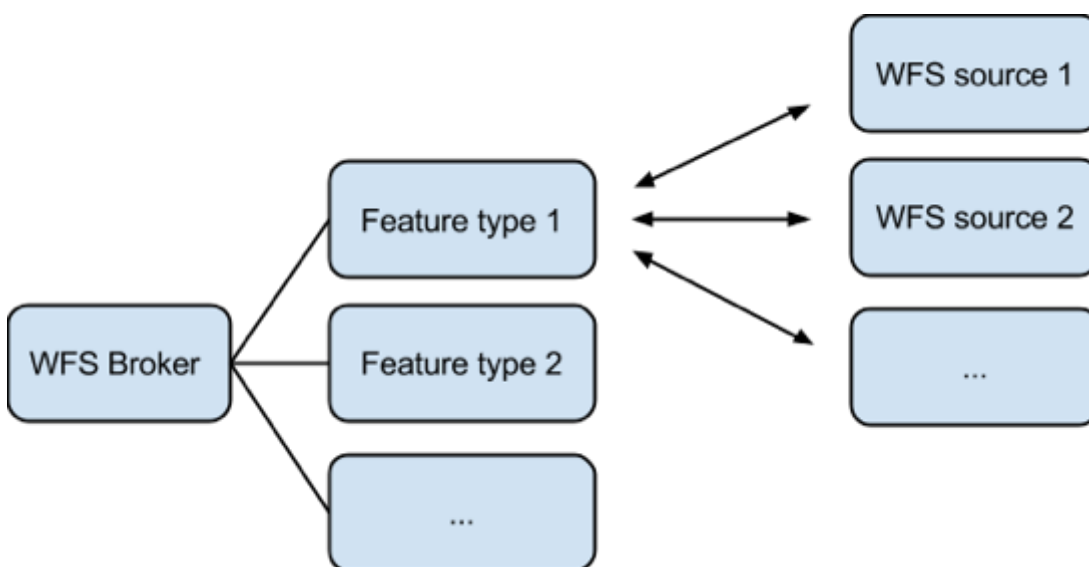


Figure 3. High-level architecture of the WFS-based Data Broker in Testbed 11

### 5.1.2. Requirements Statement

The Data Broker is intended to provide a single interface for clients to connect to multiple WFS (or WMS/WMTS) web services. The long-term vision for Data Broker is to provide a central interface for filtering, conflating, and brokering all requests to any OGC web service to simplify queries by a client and implement secure connections based on business rules functionality. For the scope of Testbed 12, the Data Broker must satisfy two requirements:

1. A request to separate WFS with heterogeneous data using AMXM and AIXM to demonstrate the conflation of two heterogeneous data sets from the same type of web service; and
2. A request to a WFS and another type of service (e.g. WMS or WCS) with a heterogeneous response. In this second approach, the participant shall determine and propose a suitable secondary service type prior to implementation according to the available data provided.

Additionally, the following requirements are derived or assumed from Testbed-12 meetings: \* Use of CSW is required for endpoint and feature discovery; and \* Use of semantic mediation is of interest to the sponsors.

### 5.1.3. Functional Overview

Luciad was selected to provide the Data Broker for Testbed 12, building off their implementation in Testbed 11.

The Data Broker provided by Luciad has the following functionality.

- OGC-compliant WFS 1.1.0 & 2.0.0 service interface with support for the following requests: GetCapabilities, DescribeFeatureType and GetFeature. Supported request encodings are HTTP GET and POST.
- OGC-compliant WMS 1.1.1 & 1.3.0 service interface with support for the following requests: GetCapabilities, GetMap, GetFeatureInfo. Support is also provided for the WMS' Styled Layer Descriptor profile: users can supply an SLD with user-defined layers and styles. Supported request encodings are HTTP GET and POST.
- Data Broker capabilities
  - Support for brokering of OGC WFS data sources complying with versions 1.1.0 and / or 2.0.0 and supporting data exchange formats AIXM 5.1 and / or AMXM 2.0.
  - Automated data source discovery by means of OGC CSW integration.
  - Conflation based on unique identifiers.
  - Aggregation of similar features types from different OGC WFS data sources in to one feature type.
  - Semantic mediation between AIXM 5.1 and AMXM 2.0 feature data.
  - Addition of provenance by integrating lineage information on the feature type level in the capabilities and on the feature level by adding ISO 19115-based lineage metadata to AIXM 5 features.

### 5.1.4. Deployment Characteristics

The Luciad Data Broker is based on Java Servlet technology. To run, it requires a Java servlet container or application server compatible with Java Servlet 3.0 or higher. Apache Tomcat 7 has been used by Luciad during Testbed 12. Other than being capable of running a Java Virtual Machine 1.7 (or higher) and an appropriate servlet container / application server, no requirements are posed on the underlying hardware or operating system.

### 5.1.5. Challenges

The Luciad Data Broker for Testbed 11 was built based on WFS 2.0 specification. For Testbed 12, the requirements include investigation in implementing additional features that are not part of the WFS 2.0 specification. For example, the inclusion of a URL in which the client may use to request a map from a Web Map Service (WMS).

Additional challenges were met when considering how to semantically search across multiple WFS of differing data models. In the demonstration use cases, one WFS contained AIXM data while the other contained AMXM data. Both data sets contain aerodrome information, and the client requests all aerodrome information for an area defined by a geospatial filter query. The Data

Broker must be able to implement a data transformation chain to convert AMXM to AIXM (or vice-versa) and return the entire feature sets back to the client.

### 5.1.6. Accomplishments

The Luciad Data Broker implements the following capabilities.

- The capability to request maps from the Data Broker using the WMS protocol.
- The capability to support mediation and brokering of semantically equivalent data formats, which enables users to gather data in a single format from the Data Broker, while the data source formats can be different.
- The capability to include portrayal information in its WFS response.
- The capability to broker across semantically similar data models such as AIXM 5.1 and AMXM 2.0.
- The capability to store information about the source of the data, known as lineage information, to inform clients about the original provider.
- The capability to mediate data responses to the broker to offer heterogenous data formats into a single format to the client.
- The capability to include feature data in a WMS response using processing instructions.

## 5.2. Catalog [F003]

The Galdos catalog service component is based on the CSW-ebRIM profile (OGC 07-110r4, OGC 07-144r4). The CSW-ebRIM profile marries the CSW catalog interface to the OASIS ebXML registry information model (ebRIM 3.0). This component provides support for publishing and discovering a wide variety of metadata resources, including descriptions of data sets and related services such as WFS 2.0 implementations (F005, E006, and F012) as well as the Data Broker component (F006). The essential purpose of the catalog in the Aviation thread is to demonstrate a central registry for finding and accessing service-related metadata in multiple aviation domains (e.g. FAA and Eurocontrol).

### 5.2.1. Status Quo

Existing SWIM registries developed by the FAA and Eurocontrol are based on the Drupal content management system and incorporate concepts from the Service Description Conceptual Model (SDCM). However, they do not currently support spatial queries and do not readily accommodate OGC service descriptions.

### 5.2.2. Requirements Statement

Section 4.5 in Appendix B of the RFQ is concerned with advancing the use of catalog services in the aviation domain. Several functional requirements are presented:

- demonstrate a CSW-based registry solution integrated with the existing FAA registry (NSRR v2) so as to provide a single entry point for service discovery;
- provide a simple user interface for executing geospatial queries such as bounding box queries

and "semantic querying of geospatial data";

- enable discovery of content provided by web services, including pub-sub content (i.e. JMS topics);
- search across multiple SWIM domain registries deployed by the FAA and Eurocontrol;
- implement the harmonized Service Description Conceptual Model (SDCM); and
- facilitate cost-effective governance of service metadata, including quality and integrity.

### 5.2.3. Functional Overview

The following figure is a simple UML component diagram summarizing the CSW-ebRIM interfaces.

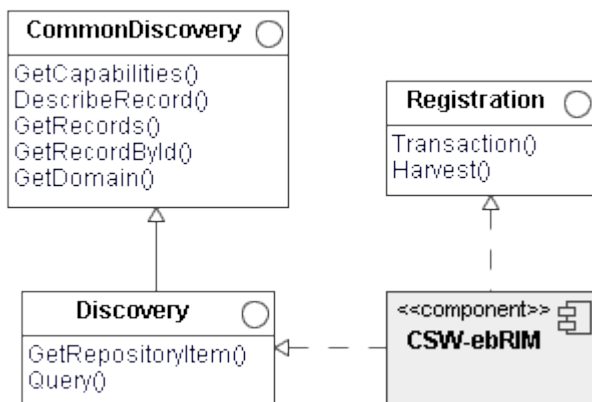


Figure 4. CSW-ebRIM interfaces

A CSW-ebRIM implementation can process the service requests summarized below, most of which are common to all CSW-based catalog services.

- **GetCapabilities:** This request allows a client to retrieve a resource that describes the essential capabilities and non-computational characteristics of the service.
- **DescribeRecord:** This request allows a client to discover the information model(s) supported by the catalog and to retrieve type definitions.
- **GetRecords:** This request allows a client to search the registry and retrieve all or some of the items in the result set.
- **GetRecordById:** This request allows a client to retrieve a representation of one or more registry objects by identifier.
- **GetDomain:** This request produces a description of the value domain for a specified data element or request parameter.
- **GetRepositoryItem:** This request allows a client to retrieve the repository item corresponding to some extrinsic object.
- **Harvest:** This request enables a "pull" style of registration whereby a resource is retrieved from some remote location and inserted into the catalog.
- **Transaction:** This request allows a client to directly insert, update, or delete catalog content.

The component provided for this thread is a complete implementation.

## 5.2.4. Deployment Characteristics

The catalog component exposes HTTP service endpoints as specified in the CSW-ebRIM specification. The implementation is Java-based and requires a Servlet container such as Apache Tomcat. It can be deployed in any operating environment for which the Java Development Kit (JDK) 8 is available.

## 5.2.5. Challenges

The initial expectation was that the catalog would also harvest metadata from existing SWIM registries provided by FAA (NAS Service Registry/Repository - NSRR 2.0) and Eurocontrol. However, an API remains under development so no harvesting mechanism was available during the testbed.

## 5.2.6. Accomplishments

Key accomplishments are listed below.

- A specialized harvester for WFS 2.0 services. The source must refer to a WFS service description (an OGC capabilities document). The corresponding resource type is "http://www.opengis.net/def/serviceType/ogc/wfs/2.0", which is the service type assigned by the OGC Naming Authority (OGCNA).
- Automatic classification of services that support temporal processing (e.g. WFS-TE).
- A periodic harvest facility that enables automatic updates.

# 5.3. Aviation Client [F004] (Aviation Client for Data Broker and CSW)

Aviation Client for Data Broker and CSW is the GUI component for interfacing with the CSW (F003) and Data Broker (F006). It is expected that the Aviation client is capable of performing queries for AIXM and AMXM data, allowing the Data Broker to conflate the response to the client.

## 5.3.1. Status Quo

The Testbed 12 Aviation Client is built on top existing LuciadLightspeed Commercial Off The Shelf (COTS) components. The Testbed 12 Aviation Client is a continuation of the Aviation Client built for previous years, capable of connecting to various OGC Services and many file formats required by the thread.

## 5.3.2. Requirements Statement

The Testbed 12 Aviation Client requires the following functionality:

- Connection and interaction with the CSW (catalogue service);
- Connection and interaction with the Data Broker to:
  - Retrieve Flight Trajectory maps;
  - Retrieve feature data of aerodromes (with OGC filters); and

- Retrieve feature data of flight trajectories (with OGC filters);
- Send temporal queries to WFS-TE services; and
- Visualization of FIXM 4.0, AIXM 5.1, AMXM data.

### 5.3.3. Functional Overview

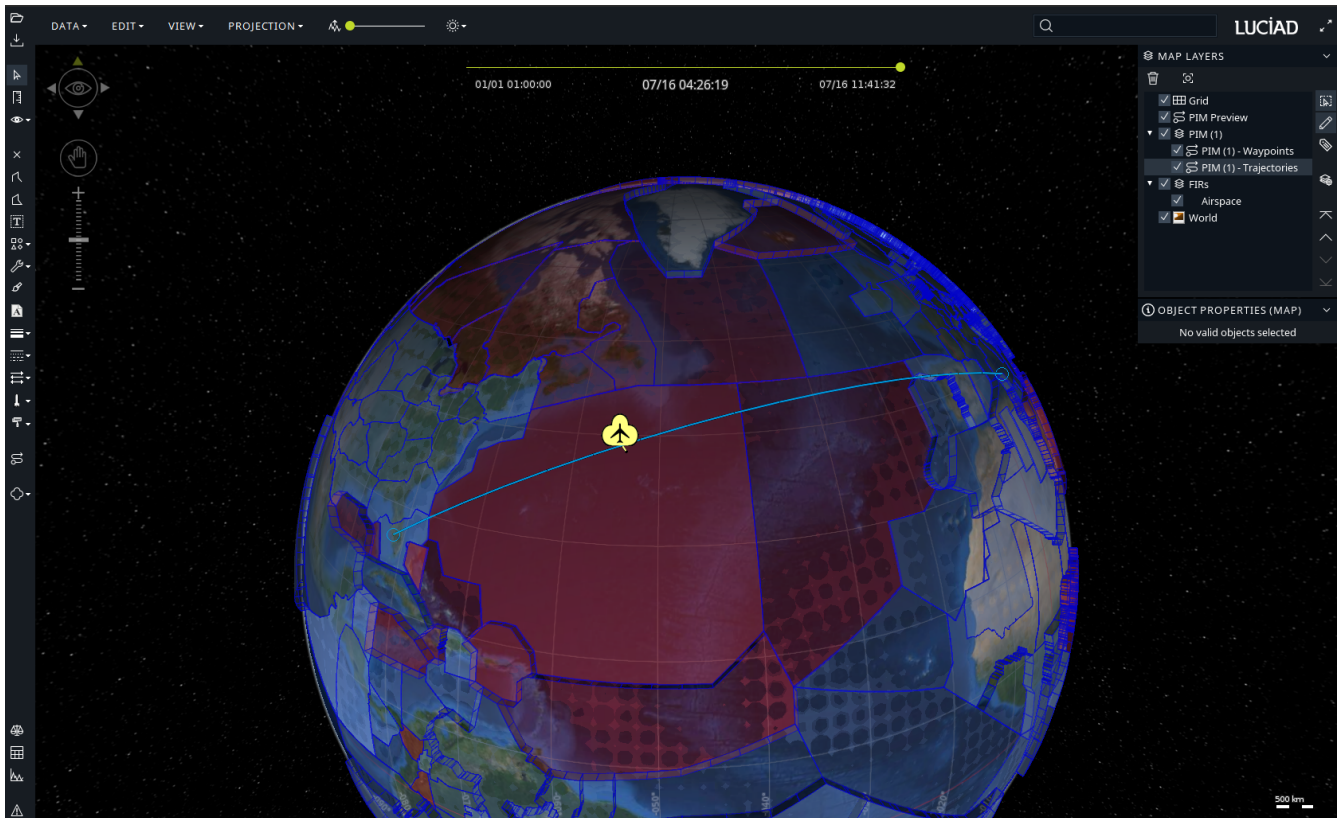


Figure 5. Luciad Aviation Client

The Luciad Aviation client provides the following functionality to support the Aviation thread in Testbed 12.

- Map-centric 2D & 3D display with intuitive user interface giving access to various actions, including:
  - Map controllers to manipulate the map (zoom & pan);
  - Map layer control with access to predefined background data layers;
  - AMXM, AIXM, FIXM and WXXM-based OGC web service connectors;
  - Flight preview / simulation; and
  - Visual representation & browsing of feature properties inside a balloon.
- Client interface to query data from an OGC Web Feature Service delivering AIXM and AMXM data. Users can select the desired WFS, feature type and various filtering options (e.g., spatial filters).
- Client interface to connect with the OGC WPS-based Validation Service.
- Client interface to connect with the OGC FPS.
- Client interface to a WMS 1.1.1 & 1.3.0 service delivering background data. Users can query and

retrieve layers from a WMS.

- Rendering engine with support for SLD / SE 1.0/1.1 to render vector feature and raster data. This engine integrates extensions to support ICAO Annex 4 rendering guidelines for aeronautical data.
- Support to represent & browse ISO 19115-compatible metadata and to encode / decode to / from an ISO 19139-compatible data source. The use of metadata extensions / profiles is supported.
- Wide range of data format support, including:
  - AIXM 3/4/5, FIXM 2/3 and WXXM 1.1, including schema extensions;
  - Additional aeronautical and weather data formats like AIXM 3.3/4.0/4.5, ARINC 424, DAFIF, ASTERIX, ASDI and GRIB;
  - Raster format support (GeoTIFF, TIFF, JPEG, JPEG 2000, GMLJP2, JPIP, PNG, GIF, ECW, MrSID, CADRG/ADRG/USRP, DTED, USGS DEM, Oracle GeoRaster) for imagery and elevation background data;
  - Vector format support (ESRI Shape, MapInfo MIF/MAP, GML 2/3.1.1/3.2.1, SVG, DGN, DWG, Oracle/Informix/MSSQL databases) for vector-based background data; and
  - Other formats: OBJ, OpenFlight, OGC KML 2.2 and GeoPDF.

#### **5.3.4. Deployment Characteristics**

All development is done in Java, using the Java Development Kit (JDK) 1.7. The client application is developed on top of Luciad's COTS product LuciadLightspeed. The software runs on any operating system for which a Java Virtual Machine 1.7 or higher exists. For the 3D visualization, a graphics card with support for OpenGL 1.2 or higher is required.

#### **5.3.5. Challenges**

The main challenges for Testbed 12 Aviation Client were the integration and workflow architecture with the various services offered in the thread. The Aviation Client has to be able to make connections to WFS, WMS, FPS and catalogue services, as well as the Data Broker component service. For these protocols, the aviation client needs to be able to parse, decode and visualize the FIXM 4.0, AIXM 5.1 and AMXM formats.

For some aspects of the demo scenario, we also have also added the ability to create and upload FIXM 4.0 trajectories on the fly to a transaction WFS service.

#### **5.3.6. Accomplishments**

The key accomplishments for Luciad's Aviation client component in Testbed 12 include:

- Successful integration of the new FIXM 4.0 GML format;
- Successful integration of the AMXM data format;
- Successful integration with the new Data Broker component;
- Integration with the CSW catalogue service;
- Demonstration of an interaction with all relevant service components provided within the

Aviation thread; and

- Contribution of a rich client equipped with lots of capabilities and features that help to demonstrate the Testbed 12 aviation use cases and to perform testing and integration with a wide variety of service components.

## 5.4. Aviation Client [F008] (Aviation Client for Asynchronous Messaging)

Aviation Client for Asynchronous Messaging is the GUI component for interfacing with the Asynchronous Messaging Server (F009). It is expected that the Aviation client is capable of performing requests for subscriptions in the PubSub 1.0 format to demonstrate the PubSub 1.0 specification implemented by the Asynchronous Messaging server.

### 5.4.1. Status Quo

Previous work in OGC testbeds on asynchronous message delivery has been prototyped into existing Aviation clients (e.g. Luciad). As the requirements for this testbed were very specific on the underlying technology, 52°North was selected to develop a client that is able to interact with the Asynchronous messaging server.

### 5.4.2. Requirements Statement

The Asynchronous Aviation Client requires the following functionality:

- creation and management of subscriptions (interaction with the OGC PubSub 1.0 server);
- consumption of AMQP 1.0 messages (by connecting to an AMQP Broker software); and
- visualization of AIXM 5.1 and FIXM 3.0.1 data.

### 5.4.3. Functional Overview

Aviation Client for Asynchronous Messaging (52°North aviationFX) is the client component for interfacing with the JMS/AMQP server (F009). It is expected that the client will be implemented based on OGC PubSub 1.0 standard guidance as an exploratory effort to study publish/subscribe messaging patterns on OGC queries and data. It is assumed that that use of Flight Information Exchange Model (FIXM) data will be utilized for live update event-based messages. The client will receive messages based on a subscription query with the Asynchronous Messaging Server.

### 5.4.4. Deployment Characteristics

The client has been developed using Java 8 (including JavaFX) and is therefore deployable on all platforms that support Java. There are no more specific requirements.

### 5.4.5. Challenges

The main challenge was to set up a client that is capable of interacting with an OGC PubSub 1.0 server (in particular its SOAP binding extension). As this specification is rather new, no best practice documents and the like were available. In addition, the creation of an intuitive user



interface that allows the subscription to interesting streams of data was a challenging efforts. In particular, streamlining this with the concepts of OGC PubSub 1.0 (such as *Publications* and *Delivery Methods*) required a few test runs.

### 5.4.6. Accomplishments

The Aviation Client for Asynchronous Messaging has been developed and tested against the Asynchronous Messaging Server [F009].

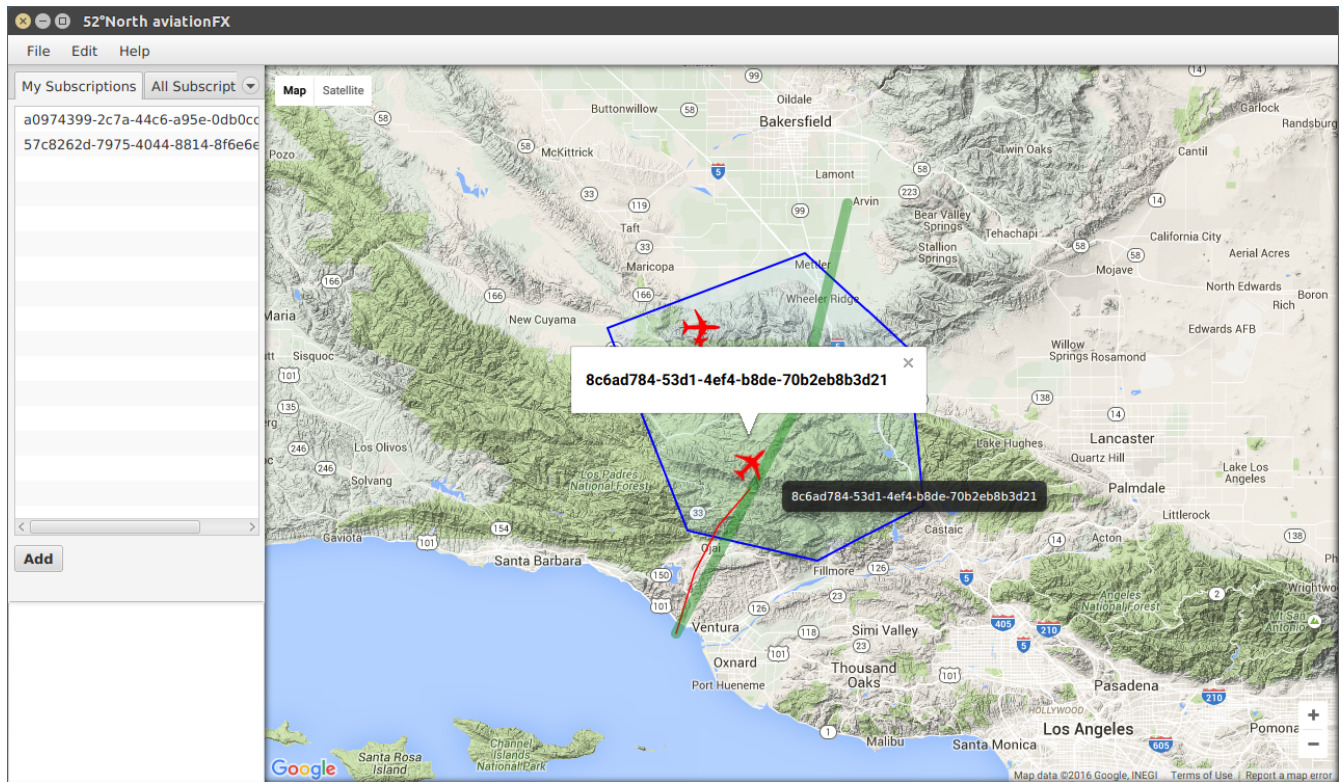


Figure 6. 52°North aviationFX client

In particular the following aspects have been accomplished:

- development of a PubSub 1.0 interaction layer;
- integration of the well-established AMQP 1.0 *Apache Qpid* library;
- visualization of updates on AIXM Airspaces; and
- visualization of the agreed flight route and the current position of an aircraft (FIXM Flight object).

## 5.5. Asynchronous Messaging Server [F009]

### 5.5.1. Status Quo

Previous work in OGC testbeds on asynchronous messaging servers focused on the implementation of the OGC Event Service (OGC 11-088r1). Recently, the requirements for message dissemination methods in the Aviation domain have been sharpened (in particular usage of the AMQP 1.0 protocol). In addition, the OGC has released the Publish/Subscribe (PubSub) 1.0 specification. These two new aspects build the foundation for a new Asynchronous Messaging Server for Aviation.



#### 5.5.4. Deployment Characteristics

52°North has developed an OGC PubSub 1.0 service (*52°North subverse*) using Java 8. It can be deployed to common Application Containers such as Tomcat, JBoss or Jetty. There are no more specific requirements.

#### 5.5.5. Challenges

The main challenge was to set up a server that implements the OGC PubSub 1.0 specification (in particular its SOAP binding extension). As this specification is rather new, no best practice documents and the like were available. An additional challenge was the implementation of the AMQP 1.0 layer for message delivery as common libraries (e.g. *Apache Qpid*) are rather limited in the documentation of implementation specifics.

An additional challenge was the integration of the Harris Data Exchange (DEX) service as a source for data streams. The DEX is a message brokering service that is in production within the FAA SWIM architecture. It supports JMS and AMQP as the message delivery protocol. In particular, the header-based routing that is used to identify the correct client destinations within DEX required a few test runs and adjustments.

#### 5.5.6. Accomplishments

In particular the following aspects have been accomplished:

- implementation of the OGC PubSub 1.0 specification (the SOAP binding extension);
- implementation of the developed AMQP 1.0 delivery profile;
- the capability for message filtering within a *Subscription* using the *OGC Filter Encoding 2.0* specification;
- support for both AIXM 5.1 and FIXM 3.0.1 including processing of geospatial queries against these formats; and
- integration of the Harris DEX as a data publisher as well as a delivery endpoint.

### 5.6. AMXM WFS [F005]

#### 5.6.1. Status Quo

The AMXM consists of the AMXM UML Model and a derived AMXM XML Schema. The schema is a ISO/OGC GML 3.2 specification for AMDB data exchange. It enables the sharing of AMDB data in accordance with EUROCAE / RTCA requirements. In combination with OGC Web Feature Service (WFS) it may be used to build SWIM information services for AMDB data. The AMXM is in a mature state of development and ready for publication.

#### 5.6.2. Requirements Statement

The scope of Testbed 12 was to deploy an AMXM 2.0 WFS 2.0 within the data access tier of the aviation architecture for use in the Data Broker use case.

Data available from the AMXM 2.0 WFS is transformed from its original source format, AIXM 5.1.

### 5.6.3. Functional Overview

The AMXM WFS provided by Snowflake Software has the following functionality:

- OGC-compliant WFS 1.1.0 and 2.0.0 service interface with support for the following requests:
  - GetCapabilities;
  - GetFeature;
  - DescribeFeatureType;
  - ListStoredQueries; and
  - DescribeStoredQueries;
- Supported request encodings are HTTP GET and POST; and
- HTTP basic access authentication enabled.

### 5.6.4. Deployment Characteristics

The Snowflake Software AMXM WFS uses COTS software to provide a mapping from database schema to AMXM 2.0 format. This mapping is used in the deployment of a read-only WFS 2.0. The WFS is deployed on a Snowflake Software hosted demonstration server on an Apache Tomcat application server and is secured using HTTP basic access authentication. Each user accessing the WFS service is provided with a username and password. The data provided by the AMXM 2.0 WFS originates from AIXM 5.1; data transformation is based upon AIXM 5.1 to AMXM 2.0 transformation rules provided by Eurocontrol.

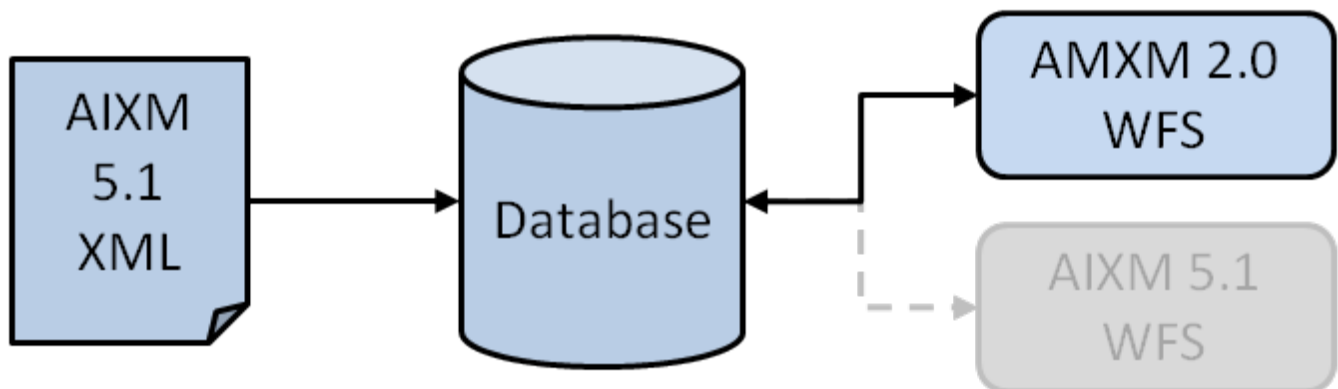


Figure 8. High-level architecture of the AMXM WFS in Testbed 12

### 5.6.5. Challenges

As the services are standard OGC compliant services and unambiguous AIXM 5.1 to AMXM 2.0 conversion rules were provided by the sponsor, no challenges were posed during setup and deployment of the AMXM 2.0 WFS 2.0.

### 5.6.6. Accomplishments

Accomplishments for Snowflake Software’s AMXM WFS include:

- Successful runtime WFS transformation from AIXM 5.1 to AMXM 2.0; and
- Successful implementation of AMXM 2.0 WFS 2.0, including spatiotemporal query capabilities using FES.

## 5.7. AIXM WFS-TE [E006]

### 5.7.1. Status Quo

The OGC-compliant WFS-TE (Web Feature Service Temporality Extension) is an aviation data repository responsible for AIXM 5.1 data management, both static documents and dynamic data such as Digital NOTAM messages. The WFS-TE service endpoint is an implementation of the OGC WFS Transactional 2.0 with additional functionalities for dealing with temporality aspects of aviation entities in accordance with the AIXM 5.1 temporality model (beyond the GML 3.2 temporality).

### 5.7.2. Requirements Statement

In Testbed 12 the AIXM WFS-TE component will demonstrate WFS Temporality Extension queries as well as serve AIXM data for the Data Broker use case.

### 5.7.3. Functional Overview

The m-click.aero WFS-TE component supports special temporality functions as described in the WFS Temporality Extension document used for the AIXM 5.1 data management.

Additionally, the m-click.aero OGC WFS Transactional 2.0 component provides standard CRUD (create, read, update, delete) data management functions based on the WFS-T 2.0 service endpoint specification. The component supports all aspects and fulfills every requirement from the official OGC WFS-T specification.

The WFS-TE service consists of subcomponents as depicted in the following figure.

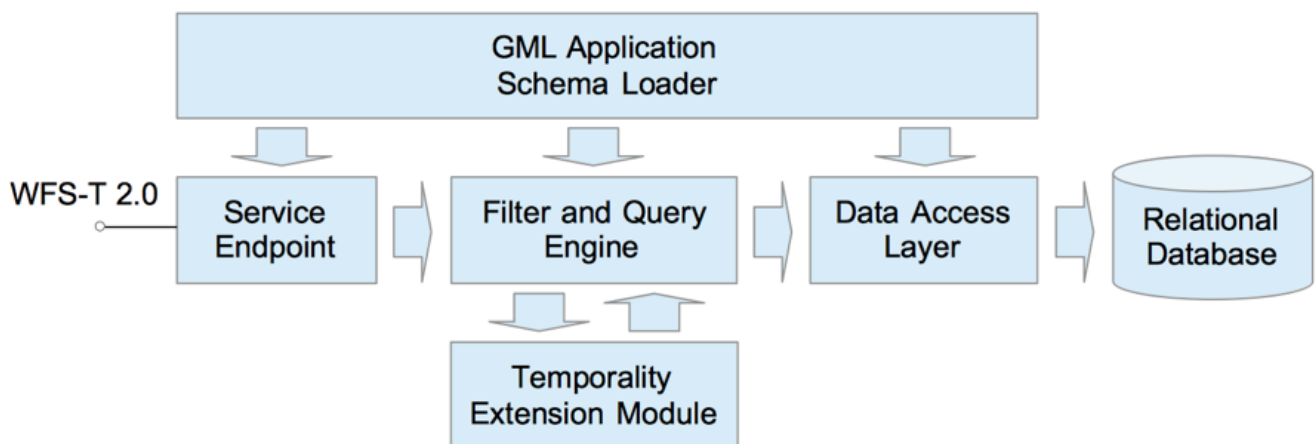


Figure 9. Component Diagram of the WFS-TE Service in Testbed 12

The WFS-TE implementation is based on the synchronous request-response message exchange pattern. Basically, the component is activated by an incoming data request call (query request). Further processing is performed in accordance with following activity diagram.

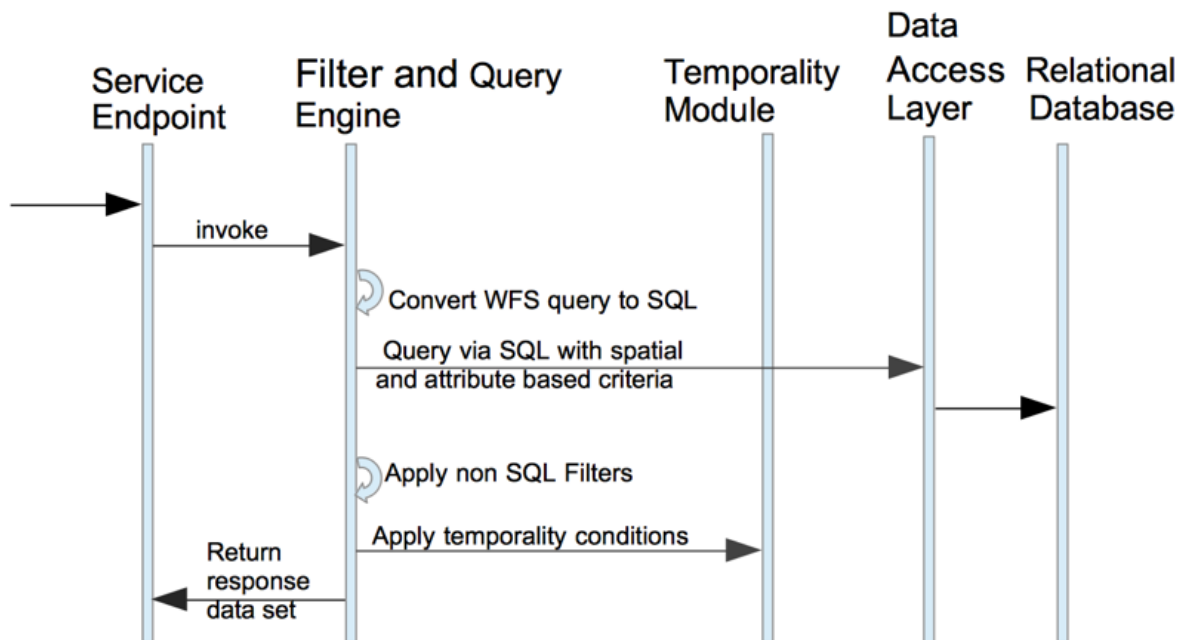


Figure 10. AIXM WFS-TE Process Flow

#### 5.7.4. Deployment Characteristics

The m-click.aero WFS is implemented based on an OGC compliant WFS data repository called deegree. The WFS runs as a Java EE web application that is bundled with a Java EE 7 compliant Web Container, which implements the Java Servlet specification. The WFS-TE is deployed on an m-click.aero hosted demonstration server.

Additionally, for purposes of access control and security a dedicated HTTP server is placed in front of the Java web container as protective reverse proxy. Each user accessing the WFS-TE service is provided with a username and password.

The deegree based solution stack is also augmented with an additional, unique m-click.aero component, responsible for dealing with WFS dynamic feature queries based on the Temporality Extension for AIXM 5.1 (OGC discussion paper). This extension helps WFS clients deal with the intricacies of the AIXM Temporality Model when requesting AIXM data.

#### 5.7.5. Challenges

No challenges for WFS-TE in Testbed-12.

#### 5.7.6. Accomplishments

- Successful demonstration of temporal transformation and snapshot generation during runtime.
- The m-click.aero AIXM WFS-TE implementation based on an open source solution stack can run reasonably fast and with unrestricted performance.

### 5.8. FIXM WFS [F012]

### 5.8.1. Status Quo

FIXM is an exchange model capturing Flight and Flow information that is globally standardized. FIXM is the equivalent for the flight domain, of AIXM and WXXM, which were developed to achieve global interoperability for AIS and MET information exchange respectively.

Built on a foundation of OGC and ISO standards, AIXM and WXXM have demonstrated that significant barriers to adoption (for example, complexity and cost of custom system builds) can be mitigated through engaging with the wider open standards community and reusing best practice patterns for application schema design and service delivery.

At the data level AIXM and WXXM share common types and patterns from ISO 19136 Geography Markup Language (GML) making them interoperable between each other as well as saving cost and time through the reuse of globally implemented types such as feature models and geometry. Being built on such a globally adopted standard as GML, AIXM and WXXM are compatible with existing tools as well as being understandable by communities across the globe.

The result has been a switch from industry specific custom software builds to off-the-shelf technology choices and with that a dramatic reduction in implementation cost. At the service level, standards such as the Web Feature Service (WFS) provide standards based web services to replace what was once a series of custom-built silo services.

Testbed 12 seeks to harmonize FIXM with AIXM and WXXM by extending it from GML. By doing so, data and service-level interoperability and standardization will be demonstrated through the component delivery of a WFS 2.0 serving GML-compliant FIXM data.

### 5.8.2. Requirements Statement

Testbed 12 seeks to integrate GML elements with FIXM for compatibility with OGC web services, in order to demonstrate the data and service-level harmonization and interoperability can be realized in FIXM, as it has been in AIXM and WXXM. To achieve this goal, FIXM will be extended from GML, reusing ISO 19136 (GML) types where applicable - thus harmonizing with AIXM and WXXM.

### 5.8.3. Functional Overview

The FIXM WFS provided by Snowflake Software has the following functionality:

- OGC-compliant WFS 1.1.0 and 2.0.0 service interface with support for the following requests:
  - GetCapabilities;
  - GetFeature;
  - DescribeFeatureType;
  - ListStoredQueries; and
  - DescribeStoredQueries;
- Supported request encodings are HTTP GET and POST; and
- HTTP basic access authentication enabled.

## 5.8.4. Deployment Characteristics

The Snowflake Software FIXM WFS uses COTS software to provide a mapping from database schema to FIXM (GML compliant) format. The mapping is used in the deployment of a read-only WFS 2.0. The WFS is deployed on a Snowflake Software hosted demonstration server on an Apache Tomcat application server and is secured using HTTP basic access authentication. Each user accessing the WFS service is provided with a username and password. The data provided by the FIXM WFS originates from FIXM 3.0.1 provided by Harris Corporation.

## 5.8.5. Challenges

As the services are standard OGC compliant services no challenges were posed during setup and deployment of the FIXM 2.0 WFS 2.0 once a GML compliant version of FIXM was developed.

## 5.8.6. Accomplishments

Accomplishments for Snowflake Software's FIXM WFS include:

- Successful runtime WFS transformation from FIXM 3.0.1 to FIXM 3.0.1 GML format;
- Successful implementation of FIXM serving WFS, including spatiotemporal query capabilities using FES; and
- References to AIXM features served from another WFS via xlink:href.

## 5.9. WMS/WMTS/WCS [F007]

WMS/WMTS/WCS component is proposed to demonstrate Data Brokering of a map using OGC Web Mapping service flows. It acts as supporting services for the testbed and its scenario.

### 5.9.1. Status Quo

The OGC compliant WMS/WMTS/WCS service is a data repository responsible for providing background data to the testbed and its scenario.

### 5.9.2. Requirements Statement

The requirements for F007 include the availability of a service that features mapping data, to be used as background imagery for any demonstrations given during the testbed. The WMS/WMTS service features data such as satellite imagery, or airport layouts that can be retrieved as maps using BBOX queries. The WCS service features satellite imagery.

### 5.9.3. Functional Overview

- OGC-compliant WCS 1.0 service interface with support for the following requests: GetCapabilities, DescribeCoverage and GetCoverage.
- OGC-compliant WMTS 1.0 service interface with support for the following requests: GetCapabilities and GetTile.
- OGC-compliant WMS 1.1.1 & 1.3.0 service interface with support for the following requests:



GetCapabilities, GetMap, GetFeatureInfo. Support is also provided for the WMS' Styled Layer Descriptor profile: users can supply an SLD with user-defined layers and styles. Supported request encodings are HTTP GET and POST.

#### **5.9.4. Deployment Characteristics**

The services have been developed using Java 7 and is therefore deployable on all platforms that support Java. There are no more specific requirements.

#### **5.9.5. Challenges**

As the proposed services are standard OGC compliant services, no challenges were posed during setup and deployment of the services.

#### **5.9.6. Accomplishments**

List of accomplishments:

- Successful deployment of WMS service that contains background imagery;
- Successful deployment of WMTS service that contains background imagery; and
- Successful deployment of WCS service that contains background imagery.

# Chapter 6. Engineering Report Topics

- [OGC 16-024] Testbed-12 Catalog Services for Aviation
- [OGC 16-045] Testbed-12 Data Broker Engineering Report
- [OGC 16-017] Testbed-12 Asynchronous Messaging for Aviation
- [OGC 16-061] Testbed-12 SBVR Engineering Report
- [OGC 16-040] Testbed-12 Aviation Security Engineering Report
- [OGC 16-039] Testbed-12 Aviation Semantics Engineering Report
- [OGC-16-028] Testbed-12 FIXM GML Engineering Report

## 6.1. Catalog ER

### 6.1.1. Problem Statement

Catalog (and registry) services play an essential "matchmaker" role in a service-oriented architecture, enabling resource producers and consumers to interact without possessing prior knowledge about service capabilities and endpoints.

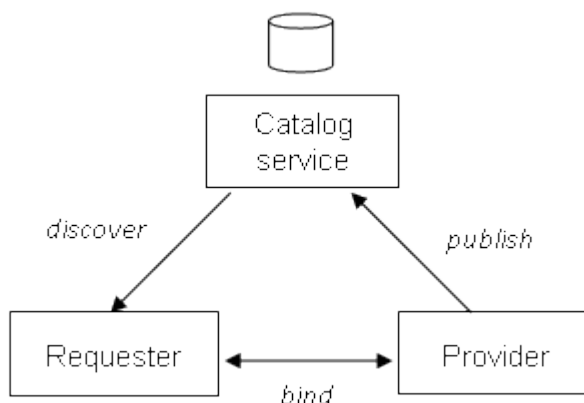


Figure 11. Role of catalog services

Within the aviation domain, the catalog role is fulfilled by a SWIM-compliant registry. However, existing SWIM registries currently under development by the FAA and Eurocontrol do not store spatial information about registered items and therefore are unable to support geospatial queries. Furthermore, they cannot readily accommodate OGC service descriptions.

### 6.1.2. Overview

This report explores the prospects for enhancing SWIM registries by a) integrating OGC catalog functionality, and b) accommodating OGC service descriptions. The technical content of this report covers two main topics:

- Using a CSW-ebRIM registry service in aviation applications; and
- Enhancing SWIM registry services:

- OpenSearch GeoTemporal services;
- Implement a CSW-SDCM profile (CSW adapter on SDCM); and
- Augmenting SDCM to accommodate OGC service descriptions.

### 6.1.3. Business Value

In the same way that one cannot stroll into a library and just pluck a desired book off a shelf, discovering and accessing services and data of interest in an operational environment generally requires the use of a catalog or registry service. This report describes how catalog services can help to enable the realization of complex workflows and to make SWIM registries more accessible to a broader community of users.

### 6.1.4. Work Conducted

Key accomplishments are listed below.

- Implemented a specialized harvester for WFS 2.0 services. The source must refer to a WFS service description (an OGC capabilities document). The corresponding resource type is "http://www.opengis.net/def/serviceType/ogc/wfs/2.0", which is the service type assigned by the OGC Naming Authority (OGCNA).
- Automatic classification of services that support temporal processing (e.g. WFS-TE).
- Implemented a periodic harvest facility that enables automatic updates.
- Investigated various options for enhancing SWIM registries so as to provide a spatial query facility.

### 6.1.5. Conclusions

Implementing the OpenSearch GeoTemporal extensions is probably the easiest way to enable a spatial search capability for a SWIM registry. However, this falls short of complying with any OGC catalog standard. Implementing a CSW adapter for the SDCM information model would be one way to accomplish this, and restricting it to read-only interactions would reduce implementation effort. The goal of such an undertaking would be to achieve a "Basic" level of conformance.

## 6.2. Data Broker ER

### 6.2.1. Problem Statement

The Data Broker Specifications Engineering Report authored in OGC Testbed 11 defined a WFS-based architecture to aggregate AIXM 5.1 data served by multiple WFS providers. In Testbed 12, the Data Broker research continues with a set of additional problem statements.

- How to offer heterogeneous services in the Data Broker, such as WFS and WMS, in a singular, uniform service that offers a single point of entry?
- How to aggregate semantically equivalent data, such as AIXM 5.1 and AMXM 2.0?
- How to embed portrayal information in WFS responses from the Data Broker?

- How to leverage a CSW Catalogue Service for endpoint and feature discovery?

### 6.2.2. Overview

The Data Broker ER builds on top of the Data Broker Specifications ER authored in OGC Testbed 11 and defines an architecture to address the new problem statements. The figure below shows the high-level architecture of the Data Broker in Testbed 12. For each problem statement, approaches are evaluated in the ER based on functional requirements, interoperability aspects and feedback gathered through implementation experiments.

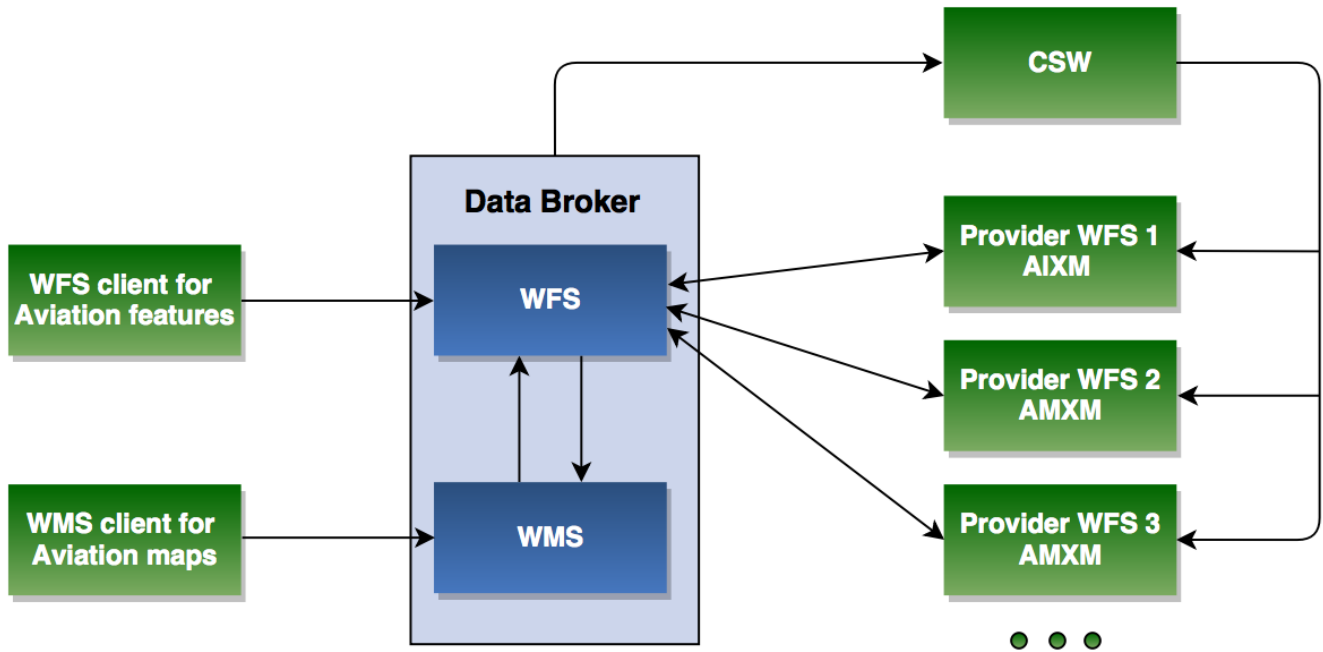


Figure 12. High-level architecture of the Data Broker in Testbed 12

### 6.2.3. Business Value

Modern-day geospatial architectures consist of a multitude of geospatial services, each dedicated to serve a specific data type to its users. This often comes with separate services to cover different areas of interest, leading to additional complexity for the user.

The Data Broker addresses this complexity by acting a single endpoint, in a hierarchy of OGC services with heterogenous data.

The business value is that using a Data Broker can drastically simplify the workflow required to look for, request and visualize data from the perspective of a client. This has a benefit for both architecture designers looking for a way to offer a set of services as a single endpoint, as well as client developers who will be able to write simpler applications with less coupling between it and the various OGC services involved in the architecture.

### 6.2.4. Work Conducted

Aligned with the Data Broker's problem statement, the following topics have been researched for the ER.

- Integration with a CSW to support WFS data provider discovery.

- Integration of a WMS in the Data Broker to support portrayal capabilities.
- Embedding of portrayal information in WFS responses.
- Aggregation of AMXM 2.0 data, including conflation and lineage provision.
- Semantic mediation between AMXM 2.0 and AIXM 5.1 data.

To support these research topics, an implementation of the Data Broker has been made during Testbed 12, building on top of the existing implementation developed in Testbed 11.

### 6.2.5. Conclusions

The results of the research and implementation experiments show that the new problem statements can be addressed in an extended Data Broker architecture. There have been a number of challenges, resulting in the following considerations and recommendations.

- To enable a WFS associating global metadata such as lineage or portrayal information with a GetFeature response, it would be convenient to have a metadata property on a WFS feature collection. Similar means are available on format-specific feature collections, such as an AIXM BasicMessage.
- The AMXM 2.0 format does not support associating feature-specific metadata on all feature types - in contrast with AIXM 5.1. This reduces the possibilities to associate feature metadata such as lineage in WFS GetFeature responses.

## 6.3. Asynchronous Messaging ER

### 6.3.1. Problem Statement

Asynchronous message dissemination played an important role in previous testbed's Aviation threads. However, no standard is available that targets the specialized requirements of the OGC world and the geospatial domain in general. In addition, no interoperable and modern approach for the actual dissemination of data messages was available.

The Asynchronous Messaging ER therefore targets the following problem statements.

- How to define an OGC compliant web service that allows the management of subscriptions and the corresponding delivery of messages?
- What protocols suite best for the dissemination of Aviation data messages from data publishers to client components?
- How to define an interoperable solution that meets all requirements in terms of geospatial filtering capabilities, data dissemination and reliability?

### 6.3.2. Overview

The Asynchronous Messaging for Aviation Engineering Report focuses on the design of an architecture to create an asynchronous messaging layer between different Aviation components such as clients, data provider instances and Data Brokers. In order to achieve interoperability among these components, the OGC PubSub 1.0 Specification forms the basis of this architecture.

The design of this architecture will cover methods for subscribing for specific subsets of data (e.g. FIXM Flights intersecting a given Airspace), managing such subscriptions as well as publishing data to Aviation Asynchronous Service instances. Different delivery patterns such as AMQP, JMS and OASIS WS-Notification are considered. In particular, their harmonization with the PubSub 1.0 specification is evaluated. Figure [Overview on the Asynchronous Messaging Architecture](#) illustrates the high-level architecture of asynchronous messaging.

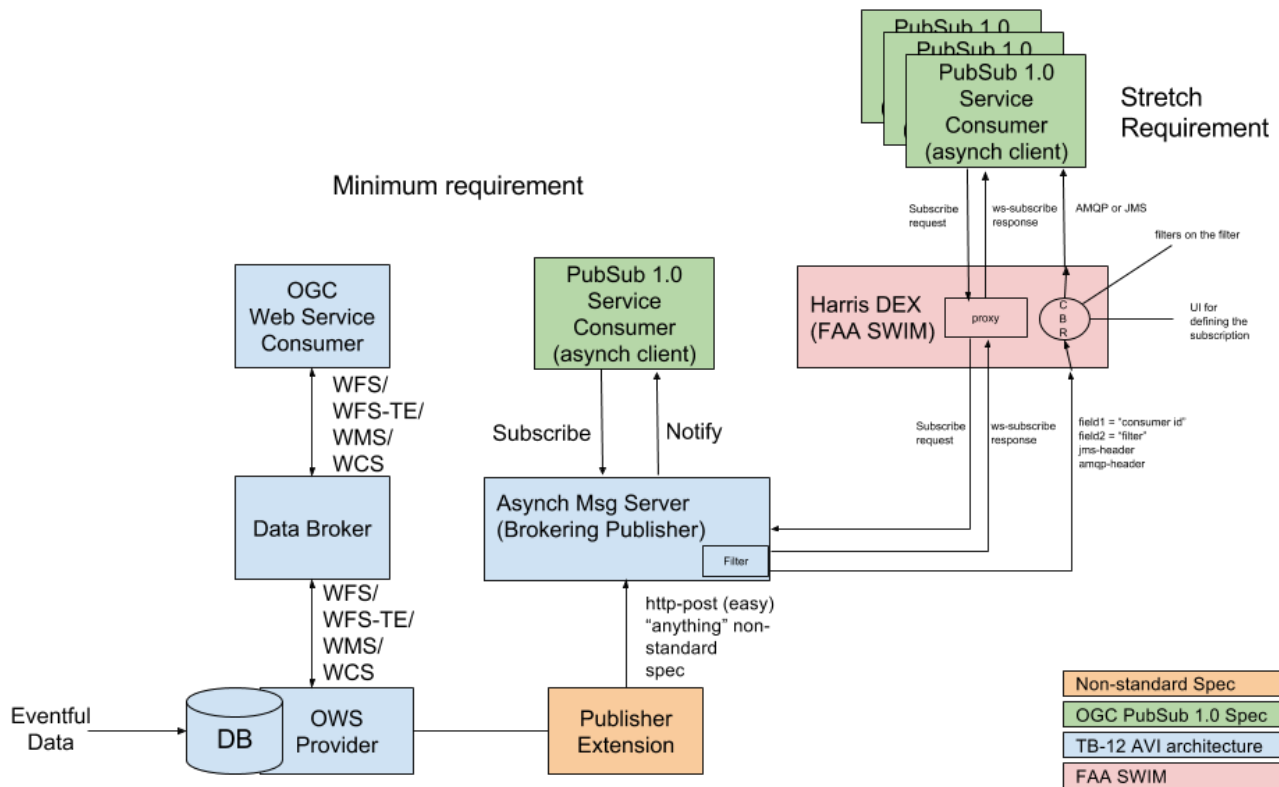


Figure 13. Overview on the Asynchronous Messaging Architecture

### 6.3.3. Business Value

ATM data as well as Flight data changes frequently. In order to achieve an efficient and reliable publication of these changes, a pull-based approach using classical request/response web services is not sufficient.

The design of an asynchronous messaging architecture which is based on the OGC PubSub 1.0 standard and combines industry-established technologies such as AMQP 1.0 and Web Services Notification overcomes this gap and eases integration. By following a standards-based approach, an interoperable solution is available which, besides the Aviation domain, can be easily transferred to other fields that require near-real time, push-based data dissemination.

### 6.3.4. Work Conducted

In order to address the above mentioned problem statements, the following work has been conducted.

- Design of a service architecture which foundation is the OGC PubSub 1.0 Specification. This architecture involves data publishers (WFS services), the Asynchronous Server (PubSub standalone publisher) and a client that interacts with the Asynchronous Server.

- Definition of an AMQP 1.0 delivery profile for OGC PubSub 1.0 that allows the interoperable management of subscriptions and the corresponding delivery of messages using the AMQP 1.0 protocol.
- Analysis of existing architectures that involve asynchronous messaging with a focus on the usage of Message Broker Software (such as Apache ActiveMQ).
- Development of a prototypical version of the designed architecture to proof its concept and functionality.
- Integration of the developed solutions with existing established systems such as the Harris Data Exchange (DEX).

### 6.3.5. Conclusions

The overall goal of designing and establishing a asynchronous messaging system has been achieved with the above described technologies. By defining an AMQP 1.0 profile for OGC PubSub 1.0, the solution is also made interoperable. However, additional work could be carried out on the following aspects:

- integration with existing solutions (Eurocontrol NetworkManager, Harris DEX);
- PubSub-enablement of existing data publishing services (WFS services); and
- deeper analysis of topic- and queue-based delivery patterns which involve Message Broker Software and accordingly definition of delivery profiles.

## 6.4. SBVR ER

### 6.4.1. Problem Statement

The Guidance on Using Semantics of Business Vocabulary and Business Rules (SBVR) Engineering Report authored in OGC Testbed 11 described a vocabulary with a profile of core geospatial terms and concepts, which can be used to express geospatial constraints in business rules. The SBVR ER applies this vocabulary to capture rules that identify operational constraints based on aircraft characteristics and the infrastructure characteristics, including the actual status.

### 6.4.2. Overview

The SBVR standard is the base for writing AIXM business rules. They complement the structural rules that can be captured in the AIXM schemas, which are technically limited. The set of validation rules is not yet complete, and its expressiveness highly depends on the available vocabulary. The ER analyses the requested validation use cases, formulates suitable SBVR rules for them, and identifies possibilities and limits of the current vocabulary. It further documents how the existing toolset of SBVR processors (ShapeChange and m-click.aero's AIXM Validator Service) was extended to support the spatial operators proposed in Testbed 11.

### 6.4.3. Business Value

The validation of AIXM 5.1 data is important for ensuring data quality. Due to its complexity, both on technical level and domain model, it's important to use a SBVR as a language that is both human-

friendly and machine-readable. This ER advances the AIXM Business Rules based on SBVR by analyzing validation use cases that weren't previously evaluated.

#### **6.4.4. Work Conducted**

The following use cases were analyzed, and if possible, an SBVR rule formulated:

- Relating the aircraft type to the possibility to operate on a given runway, taking into consideration:
  - landing distances;
  - risk of overshooting the last exit taxiway;
  - limitations in turning radius of the aircraft; and
  - the purpose (or flight status): regular or emergency landing;
- Identification of airports along the trajectory of a flight; and
- Validation of a 'runway threshold change' D-NOTAM.

Further, the following tools were extended to support SBVR spatial operators:

- the open source ShapeChange tool; and
- m-click.aero's AIXM Validator Service.

#### **6.4.5. Conclusions**

The ER demonstrates that SBVR for AIXM validation benefits from an extended vocabulary as provided by the proposed spatial operators, and proves that their implementation in existing software is possible. However, some of the analyzed use cases revealed that the complexity of the domain model leads to very complex SBVR rules, that a very hard to write and understand, and sometimes even impossible to formulate. Further research is necessary to address these limitations.

## **6.5. Aviation Security ER**

### **6.5.1. Problem Statement**

Information security is the state of being protected against the unauthorized use of information and services, or the measures taken to achieve that. In the domain of aviation, within the system wide information management (SWIM), numerous services and data that need to be protected are exposed as OGC OWS compatible, SOA conformant web service endpoints. While the aviation modernization programs in the USA and the EU (NextGen, SESAR) have already defined security frameworks for modern aviation services, the Testbed-12 aviation architecture remains focused on its functional aspects letting the cross-cutting concerns such as information security to be evaluated somewhere else. It is the task given to this engineering report to provide an overview of possible security options, to apply the Testbed-11 recommendations regarding the OGC OWS security on the Testbed-12 Aviation Architecture, and to propose advanced security concepts related to identity management and geospatial based access control.



## 6.5.2. Overview

The document provides short introduction into the major elements of information security considering the related work performed during the aviation modernization programs in the USA, the EU, and by ICAO. The methods of information security with concepts applicable to the OGC compatible aviation web services are presented in the form of security functions such as confidentiality, integrity, and availability. Terminology and categorization has been taken from Testbed-11 recommendations and is based on the ISO standards. The report highlights the differences between transport and message based security, puts them in the relation with communication message exchange patterns, and explains how those concepts are currently applied in the aviation domain. Finally, the focus is put on the digital identity management, as well as access right control with security policy enforcement based on geospatial attributes with GeoXACML

## 6.5.3. Business Value

Geospatial enabled access control and approaches for securing OGC OWS services, as well as identity management methods proposed in this report greatly improve the overall security of aviations services and enable interoperability among regional and global aeronautical traffic management stake holders.

## 6.5.4. Work Conducted

For the concrete case of aviation architecture in the Testbed 12, this engineering report has provided an overview of and proposal for the security architecture based on the security relevant outcomes from previous engineering reports. The testbed's architecture originally doesn't specify any security requirements. Based on some operational examples (FAA SWIM and SESAR SWIM) and a couple of preconditions, constraints and requirements have been made in order to distinct between several architectural options.

## 6.5.5. Conclusions

For security implementation in aviation, the researchers came to conclusion that the most efficient single organizational domain based solution would be network level security (VPN), but it is questionable whether such architecture would be applicable for integration between multiple organizational domains (for example in case of global SWIM integration based on federation and considering the existence of numerous aviation service providers and consumers operating from distinctive security domains). On the other side, brokered identity management and security functions applied on the message level represents the most flexible solution with trade-off in complexity and security. Application of security functions on the transport level, such as the TLS, seems to be the compromise between network and message approaches. Use of PKI infrastructure authentication and confidentiality could easily be established, but such a solution lacks in end-to-end security. Still, this option is the most frequently used in the published internet and fits well with OGC OWS service endpoints based on HTTP POST approach (for end-to-end security, OGC service endpoints must be secured on the message level using a SOAP extension mechanism). Finally, an access control approach based on geospatial features is possible for geospatial services and resources from the aviation domain. Such an approach could be implemented using the concept of declarative security policies and utilizing the OGC GeoXACML.

## **6.6. Aviation Semantics ER**

### **6.6.1. Problem Statement**

WSDOM is a formal ontological model based on OWL-S that is being developed by FAA for use in building applications that process and exchange web service information. It can be utilized to create, manage and query/discover service metadata instances using semantic technology. While it provides good basis for extensions and already contains some examples for concepts specified in the FAA Service Taxonomy, the ontology doesn't contain enough metadata, for example to fully classify services according to the taxonomy or to express their geospatial properties.

### **6.6.2. Overview**

This ER analyses the options for improvements of semantic description for (OGC compatible) web services with a focus on the aviation domain. It evaluates different options for further utilization of semantic technology to classify, describe, and discover OGC OWS compatible aviation services.

### **6.6.3. Business Value**

Geospatial enabled service discovery as proposed in this ER together with advanced semantic technologies, such as axioms and reasoning, improves the expressiveness of service catalogues and contributes to the interoperability among distinctive, regional, national, and global service catalogue formats, service providers, and consumers.

### **6.6.4. Work Conducted**

This ER provides briefly explanation of aviation domain and services, lists stake holders and aviation use case(s), and provides an overview into the service catalogues created as part of aviation modernization programs in the USA and the EU. It has evaluated approaches for semantic service discovery based on the FAA WSDOM ontology and demonstrated the application of FAA Service Taxonomy and OGC gGeoSPARQL on the service description methodology. Finally, the report has proposed an application's solution stack and performed an investigation about how the semantic approach fits into overall service registry and repository infrastructure proposing possible integration scenarios.

### **6.6.5. Conclusions**

WSDOM ontology based on the OWL-S provides basis to develop semantic service descriptions and operate service metadata repository based on semantic technology. The enhancements proposed here (together with those yet to be identified as part of future investigations) will eventually enable successful operational use adding missing concepts from the aviation domain. Service discovery is all about reasoning over and querying of semantic data. The fact that WSDOM schema has been created using OWL actually implies that SPARQL shall be used to express queries for service discovery. This ER has demonstrated a couple of such queries in conjunction with FAA Service Taxonomy. Using the FAA Service Taxonomy classification and GeoSPARQL extension it was possible to achieve basis for the service discovery of the same or better quality as it was the case with standard service catalogues having better extensibility capabilities. It is especially true for the semantic discovery approach based on axioms and reasoning. Thus, several important activities

remain to be done in order to gain full ontological power required for efficient semantic service discovery beyond the standard approach based on pattern matching and attribute values.

## 6.7. FIXM GML ER

### 6.7.1. Problem Statement

The Flight Information Exchange Model (FIXM) resides within a family of ATM exchange models including AIXM and WXXM. Both AIXM and WXXM are based upon GML, whereas FIXM is based upon XML - this inconsistency contributes additional implementation costs. Aligning FIXM with AIXM and WXXM by basing it upon GML provides implementation benefits at the data, service and application levels.

- Data Level - interoperability and harmonization between FIXM, AIXM and WXXM would be aided through the reuse of globally implemented types.
- Service Level - standards such as the WFS provide a standardized approach to data exchange, this would negate the need for custom-built silo services.
- Application Level - standards such as GML are understandable by a global community and supported by a plethora of existing tools.

Testbed 12 aims to advance the discussion of GML integration into FIXM by building upon previous OGC IP initiative and FIXM stakeholder inputs. The GML integration process will be documented in the FIXM GML ER, whilst a WFS deployment serving a GML compliant version of FIXM will act as a technical proof of concept.

### 6.7.2. Overview

FIXM is an exchange model capturing Flight and Flow information that is globally standardized. The need for FIXM was identified by the ICAO Air Traffic Management Requirements and Performance Panel (ATMRPP) in order to support the exchange of flight information as prescribed in Flight and Flow Information for a Collaborative Environment (FF-ICE). FIXM is the equivalent, for the Flight domain, of AIXM (Aeronautical Information Exchange Model) and WXXM (Weather Information Exchange Model), both of which were developed in order to achieve global interoperability for, respectively, AIS and MET information exchange. FIXM is therefore part of a family of technology-independent, harmonized, and interoperable information exchange models designed to cover the information needs of Air Traffic Management.

The OGC Testbed 12 FIXM Engineering Report seeks to harmonize FIXM with AIXM, and to permit the exchange of FIXM via a WFS. Testbed 12 seeks to build upon previous OGC IP initiatives and to demonstrate the integration of the ISO/OGC standard GML in a manner that meets the requirements of the FIXM community. Work undertaken in previous OGC IP initiatives seeks to integrate GML to FIXM via a model-driven approach.

### 6.7.3. Business Value

Built on a foundation of OGC and ISO standards, AIXM and WXXM have demonstrated that significant barriers to adoption (aka, complexity and cost of custom system builds) can be mitigated

through engaging with the wider open standards community and reusing best practice patterns for application schema design and service delivery. At the data level AIXM and WXXM share common types and patterns from ISO 19136 GML making them interoperable between one another as well as saving cost and time through the reuse of globally implemented types. At an application level, being built on globally adopted standards like GML, means AIXM and WXXM are compatible with a plethora of existing tools as well as being understandable by a global community. The result has been a switch from industry specific custom software builds to off-the-shelf technology choices and with that a dramatic reduction in implementation cost. At the service level, standards such as the Web Feature Service (WFS) provide standards based web services to replace what was once a series of custom-built silo services. This interoperability and openness is at the heart of the net-centric vision of SWIM.

#### **6.7.4. Work Conducted**

To implement GML in FIXM whilst maintaining a compact data encoding, the following areas have been investigated:

- Object-Property Model (including the UML to XSD process);
- GML Abstract Feature;
- Temporal Values;
- Use of `xsi:type` Attribute;
- Harmonize FIXM references with AIXM and WXXM using `xlink`; and
- Remove restrictions on providing natural keys and coordinates.

#### **6.7.5. Conclusions**

The engineering report provides a set of recommendations for FIXM based on the FIXM 3.0.1. These recommendations were further developed into Change Requests which are expected to be brought forth to the FIXM Change Control Board (CCB) to be voted upon and enacted as changes for future versions of FIXM.

# Chapter 7. Demonstration Scenarios

## 7.1. Scenario 0

This scenario demonstrates the cooperation of the OGC with the RTCA organization, specifically the TFR task Group. In working with the TFR Task Group, data quality and validation was identified as a concern. The gaps identified include the need for the following:

- a standard data model such as AIXM to provide NOTAM data;
- a template for NOTAM language to avoid ambiguity of free-text;
- solution for maintaining conformance with FAA policy orders;
- formatting and encoding compliance; and
- graphical verification.

### 7.1.1. Components

- m-click.aero Data Validation Service
- m-click.aero WFS-TE Client
- E006 AIXM WFS-TE

### 7.1.2. Status Quo

Currently, NOTAMs contain free text in a legacy format that requires a human to decipher. With the introduction of Digital NOTAMs, stakeholders are trying to determine a methodology to transfer the fluidity of free-text NOTAMs into a machine-readable format. NOTAMs are often generated without graphical verification resulting in incorrect depiction of NOTAM boundaries in pilot briefings. NOTAMs also violate FAA policy by remaining active beyond an expiration, or are created to depict a non-temporary restriction that should be represented in another format.

### 7.1.3. Scenario 0 Sequence

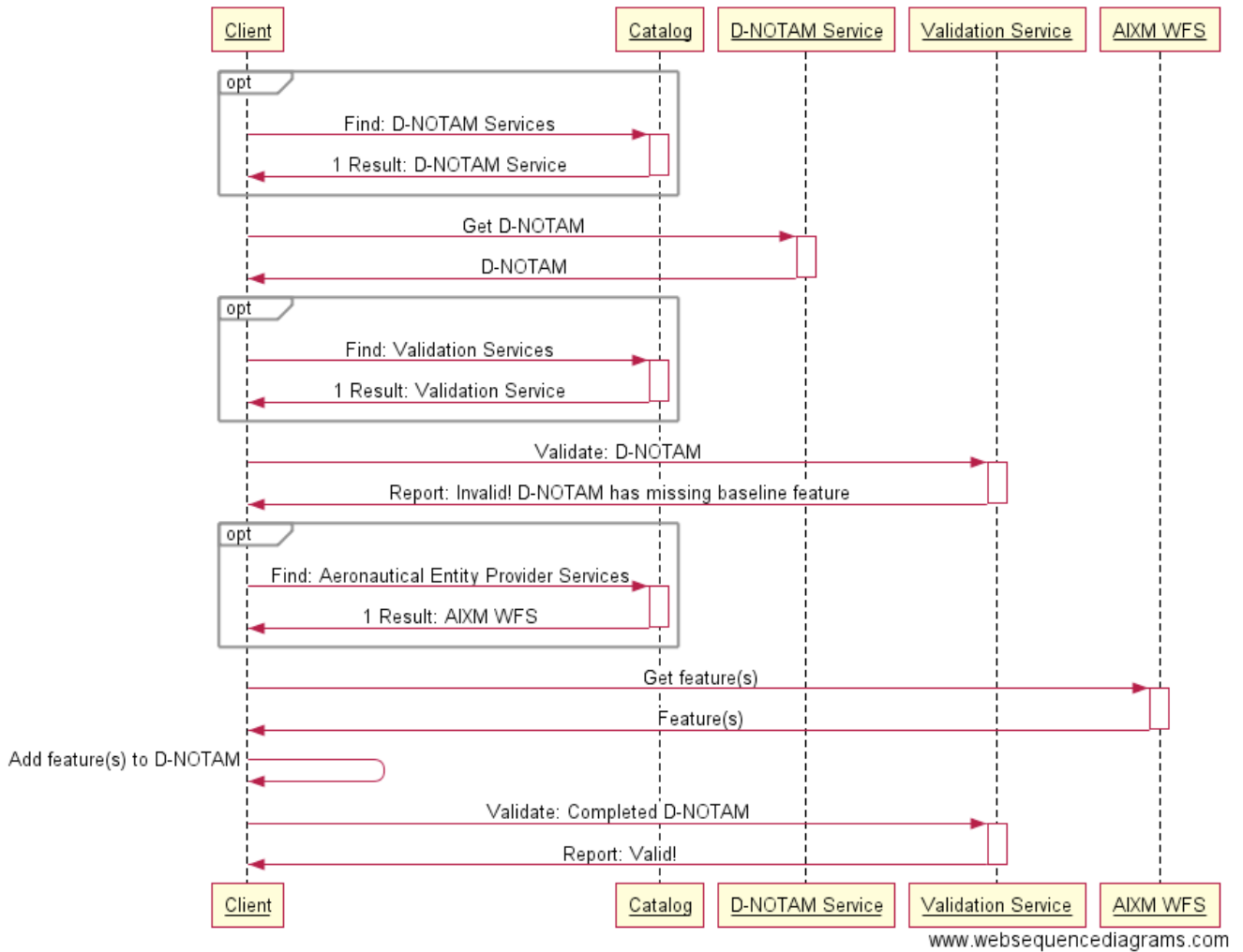


Figure 14. Sequence Diagram for Aviation Thread Scenario 0

### 7.1.4. Result

A real TFR NOTAM was used for this demonstration, taken from the FAA NOTAM website. The image below depicts the TFR NOTAM for aerial fire-fighting on the FAA NOTAM website.

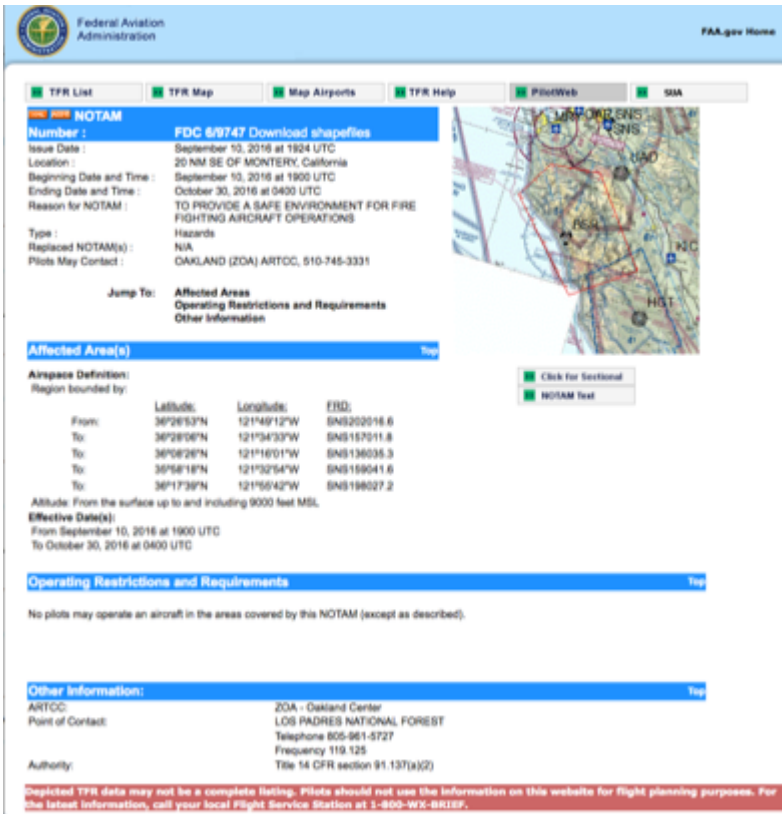


Figure 15. TFR NOTAM for OAKLAND (ZOA) ARTCC

The NOTAM was transferred from legacy format to digital AIXM format and loaded into the WFS-TE service. This service then allows the m-click.aero validation service to process FAA policy in the form of business rules. These business rules are encoded in Semantic Business Vocabulary and Rules (SBVR) format, an OMG standard. The business rules are then converted to Schematron format, and then used to validate the NOTAM data.

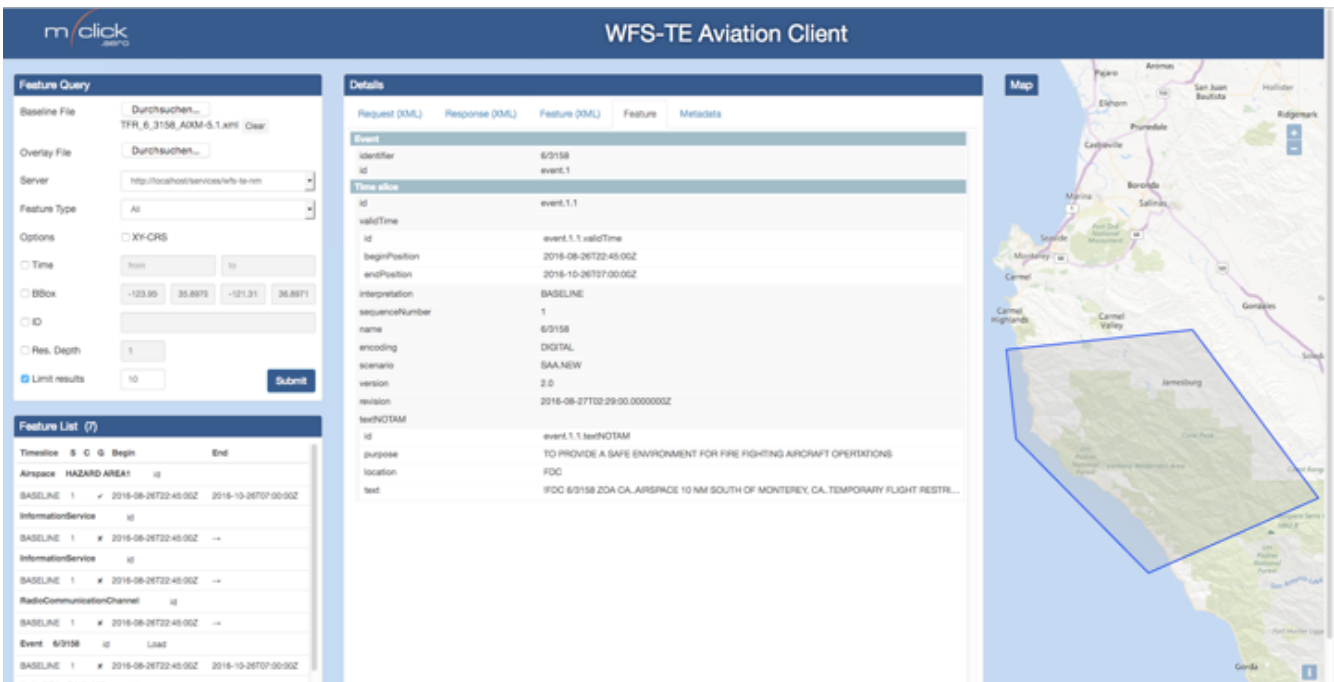


Figure 16. Data Validation Service discovering a TFR NOTAM

The result is the validation of the TFR NOTAM graphically, while ensuring validation prior to submission. The firefighting TFR NOTAM can now be submitted to the FAA with confidence.

### 7.1.5. Benefits

- Data conforms with AIXM 5.1 providing machine readable formats.
- Data validation using semantic rules provides a mechanism for testing conformance with FAA Policy Orders including testing before submission and testing routinely for compliance.
- TFR NOTAMS can be viewed graphically for accuracy with the use of existing OGC compliant AIXM and WFS client viewers.

## 7.2. Scenario 1

Flight plans are typically recurring flights scheduled according to an airline schedule. Typically these flight plans are submitted days in advance to be scheduled, and amendments are submitted to update a flight plan prior to going active according to any flight restrictions and weather. The International Civil Aviation Bureau (ICAO) has developed a future concept called Flight and Flow in a Collaborative Environment, also known as FF-ICE. The first iteration of this concept, FF-ICE Step 1, is focused on Pre-Departure Flight Planning. FF-ICE Step 1 utilizes standard SWIM services and data structures such as the Flight Information Exchange Model (FIXM) and Aeronautical Information Exchange Model (AIXM) to support operations pre-flight planning. OGC services provide service interface standards and advanced semantic and geospatial search capabilities.

### 7.2.1. Components

- F003 Catalog
- E006 AIXM WFS-TE
- F012 FIXM WFS
- F004 Client

### 7.2.2. Status Quo

Currently, flight plans are developed without verification and then submitted via either the Aeronautical Fixed Telecommunications Network (AFTN) or the Aeronautical Message Handling System (AMHS), both of which precede SWIM. These services are only message handlers and require alternate means to verify the flight plans. Flight plans currently are in ICAO FPL2012 format, and do not correlate aeronautical data in a machine-processable manner. The flight plan identifiers tend to be correlated to an aircraft ID which is not unique and may be re-used multiple times a day, making it difficult to identify a single unique flight.

### 7.2.3. Scenario 1 Sequence



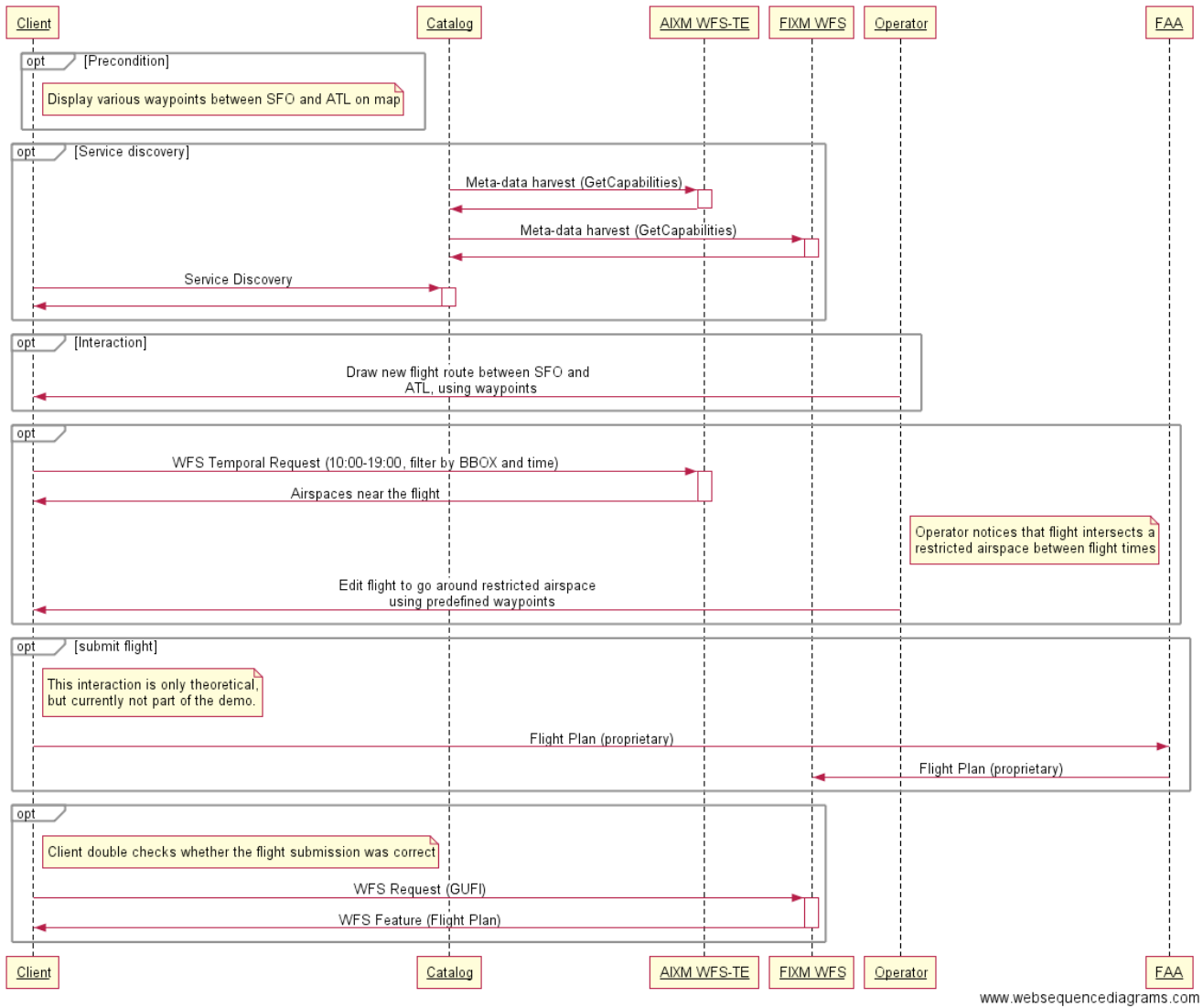


Figure 17. Sequence Diagram for Aviation Thread Scenario 1

## 7.2.4. Result

In this scenario, service discovery is demonstrated for all services containing the relevant information for flight planning. A direct request to AIXM WFS-TE is made to retrieve features containing aeronautical geometries and constraints for relevant to a flight plan. By using the FIXM flight plan information containing the temporal information for active flight time, the route of flight can be used to query for any TFR NOTAMS within the constraints of the route and time. The FIXM schema was modified to include a minimal set of GML elements such that the information can be made available via a WFS. This enabled the correlation of the route information to the aeronautical information geography. The image below shows the result of the query using the WFS-Temporality Extension service which enables additional temporal query parameters. The image below shows the TFR based on the route of flight.

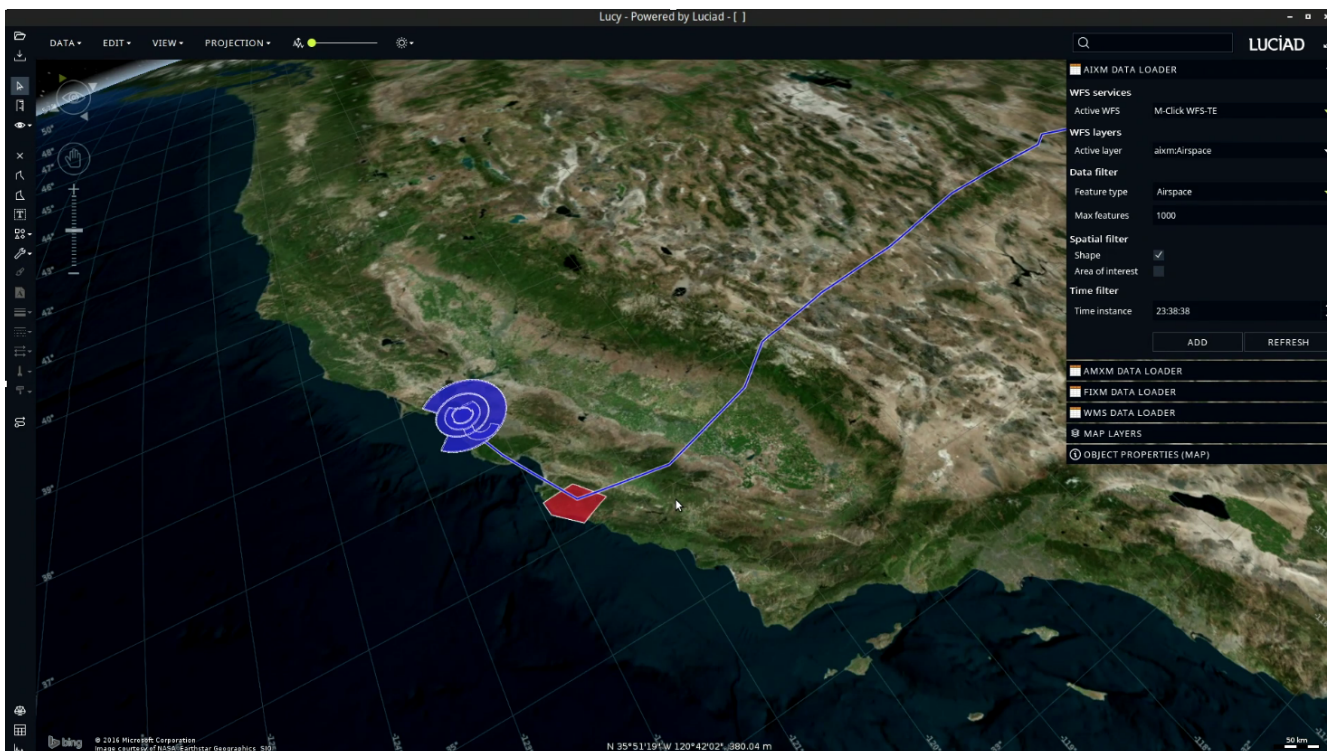


Figure 18. Flight Plan intersecting a TFR NOTAM

The results are compared to the current planned flight plan to determine an optimal flight plan trajectory. Then the flight plan is submitted to FAA and a direct request to FIXM WFS is made for the submitted flight plan as a query on the Globally Unique Flight Identifier.

### 7.2.5. Benefits

- Use of Catalog capabilities demonstrates OGC service standards can support SWIM Common Registry concepts such as Service Description Conceptual Model (SDCM) for OGC services.
- Use of WFS Temporality Extension (WFS-TE) to make specific temporal queries demonstrates the first implementation of a WFS-TE service. This implementation will inform future development of WFS standards and extensions as well as relevant use cases in Aviation such as in AIXM and FIXM. Specifically, the use of WFS-TE could support Aeronautical Information Systems (AIS) in SESAR and FAA (e.g. NAS Common Reference).
- The FIXM WFS is the first implementation of a FIXM-based WFS. Development of a FIXM schema with GML properties enables the use of OGC WFS standards to serve FIXM data. This enables clients currently supporting WFS for AIXM to also consume FIXM using the same standard interfaces.

## 7.3. Scenario 2

An earthquake has caused several city fires in the vicinity of the airspace causing lower atmosphere smoke, constraining terminal airspace departures. A pilot needs to download a pre-flight briefing to their Electronic Flight Bags (EFB). On the EFB client, the pilot can query for their flight plan. A Data Broker provides a single point of access and orchestration to assist in the retrieval of information for the client application. OGC web services are used to provide feature information on the airport, flight plan, and map the information in a format that the client can display to the pilot. A FIXM flight plan contains GML elements including a reference to the

departure aerodrome. The Data Broker is capable of requesting information from multiple data sources such as from an Aerodrome Mapping Data Base (AMDB) in the Aerodrome Mapping Exchange Model (AMXM) format and converting the data into a single data set in AIXM format. Using a pre-processing instruction, the Data Broker can also convert the AIXM data using a built-in Web Mapping Transaction Service (WMTS) to convert the XML data into a displayable map. The map is sent back to the pilot's EFB client to display their pre-flight briefing.

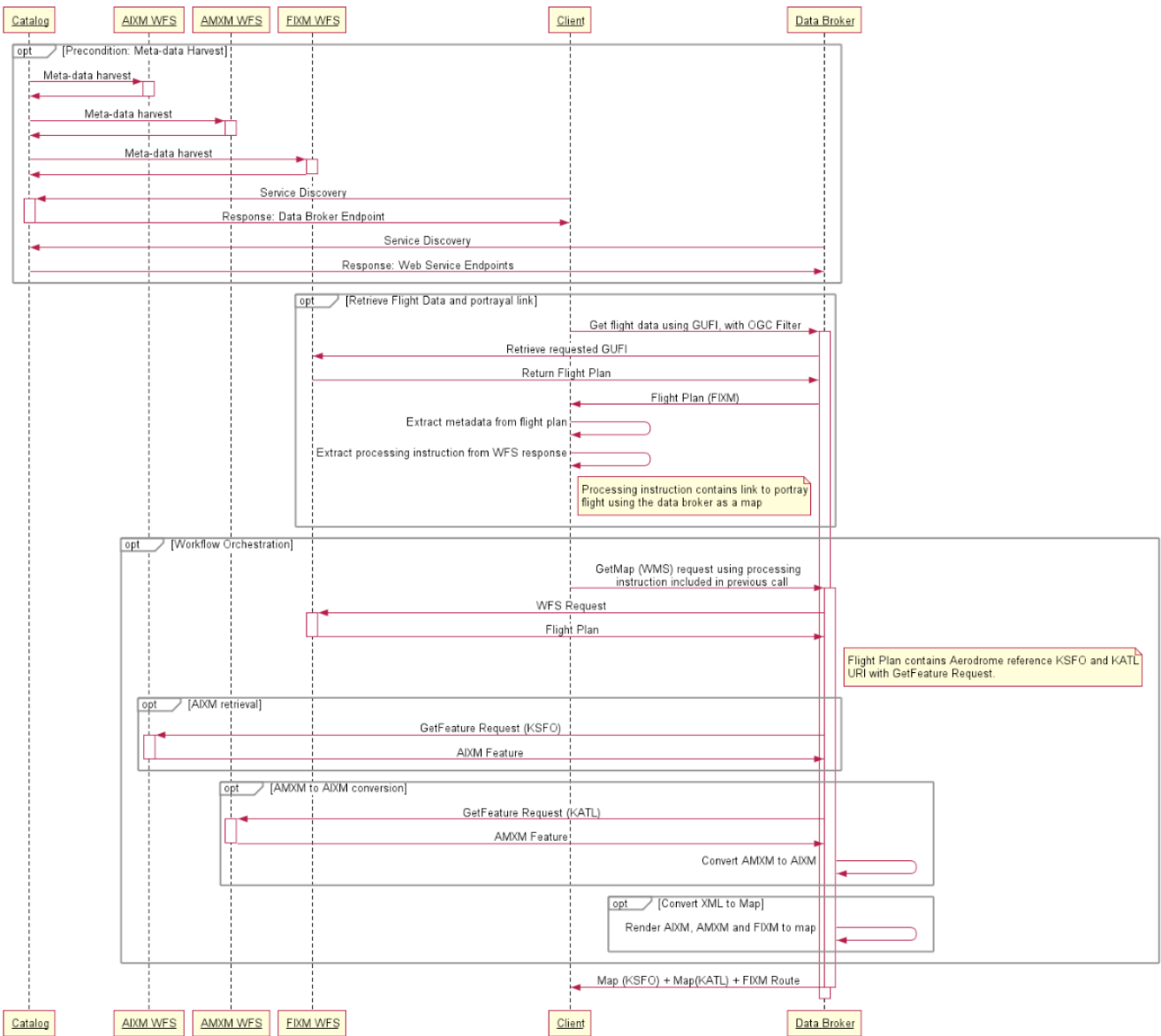
### **7.3.1. Components**

- F003 Catalog
- E006 AIXM WFS-TE
- F012 FIXM WFS
- F004 Client
- F006 Data Broker
- F007 WMTS
- F010 AIXM WFS
- F005 AMXM WFS
- F012 FIXM WFS

### **7.3.2. Status Quo**

Currently, aviation is in a transition period in which some pilots are able to retrieve digital pre-flight briefings, while others must call into an FAA service operations desk to get briefed by an operator. However, electronic pre-flight briefings may not have the latest information due to information delay, and this information is subject to change once in-flight.

### **7.3.3. Scenario 2 Sequence**



www.websequencediagrams.com

Figure 19. Sequence Diagram for Aviation Thread Scenario 2

### 7.3.4. Result

In Testbed 12, the Data Broker was augmented with a WMS service along with its already existing WFS service. This service provides the ability to orchestrate a conversion of WFS retrieved XML data into a graphical image for display to the pilot. The figure below depicts the architecture of the Data Broker.

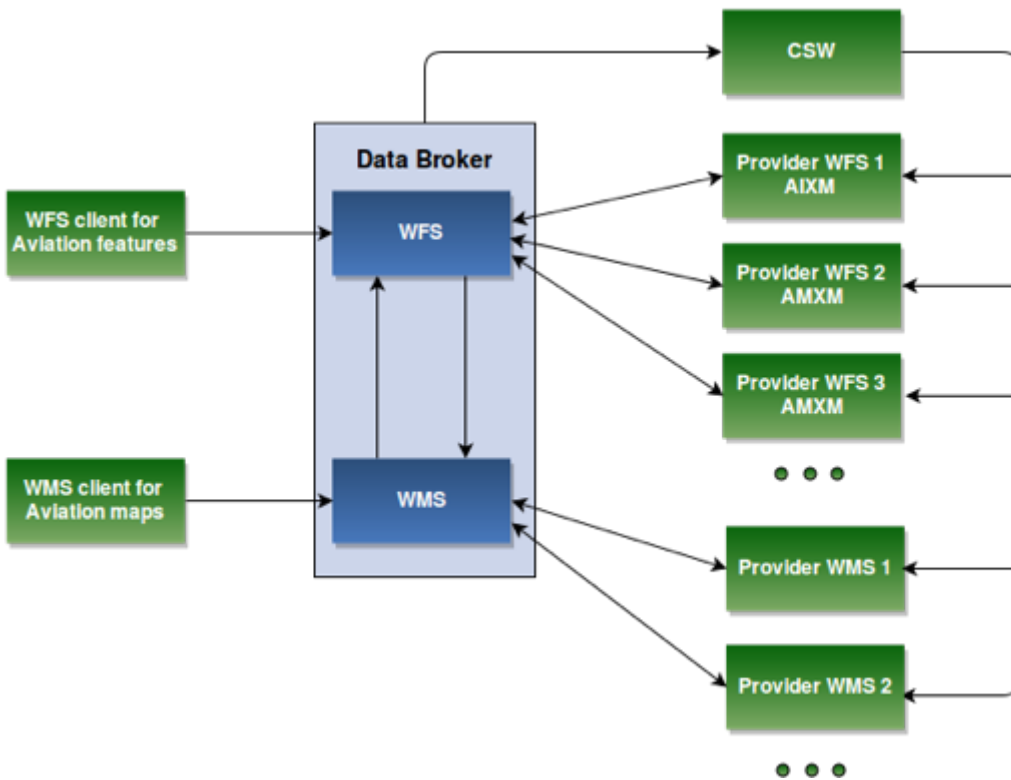


Figure 20. Data Broker Architecture

First, the pilot enters the flight GUFi and the data broker is able to retrieve the flight information and route.

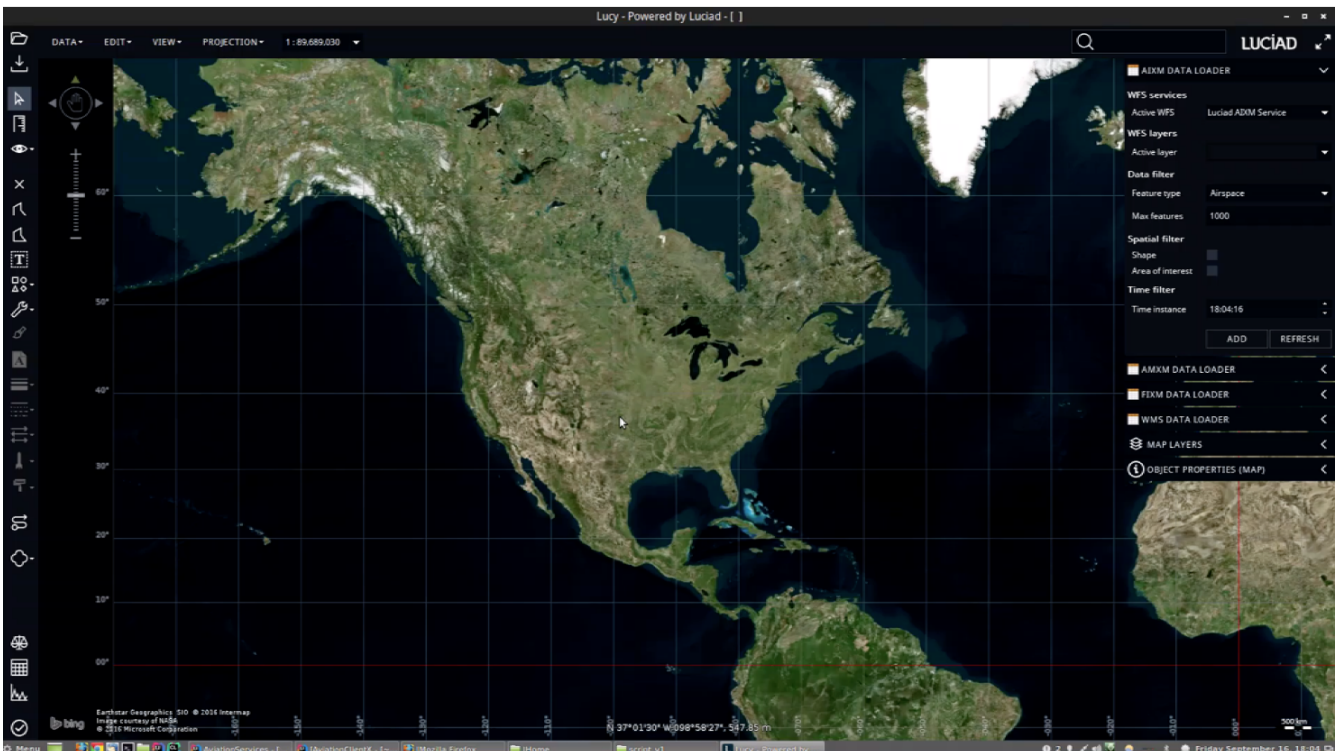


Figure 21. Data Broker retrieves flight information

Next, the data broker determines the departure and arrival airports and uses the embedded Xlink information to retrieve corresponding aeronautical geography from other WFS services. In this case, the departure airport is KSFO and the arrival airport is KATL. The WFS containing KSFO data is in AMXM format. The data broker is orchestrated to convert AMXM formatted data to AIXM format. The next figure shows the KSFO runway information with the lineage information and

provenance showing that it was 'Transformed from AMXM 2.0'.

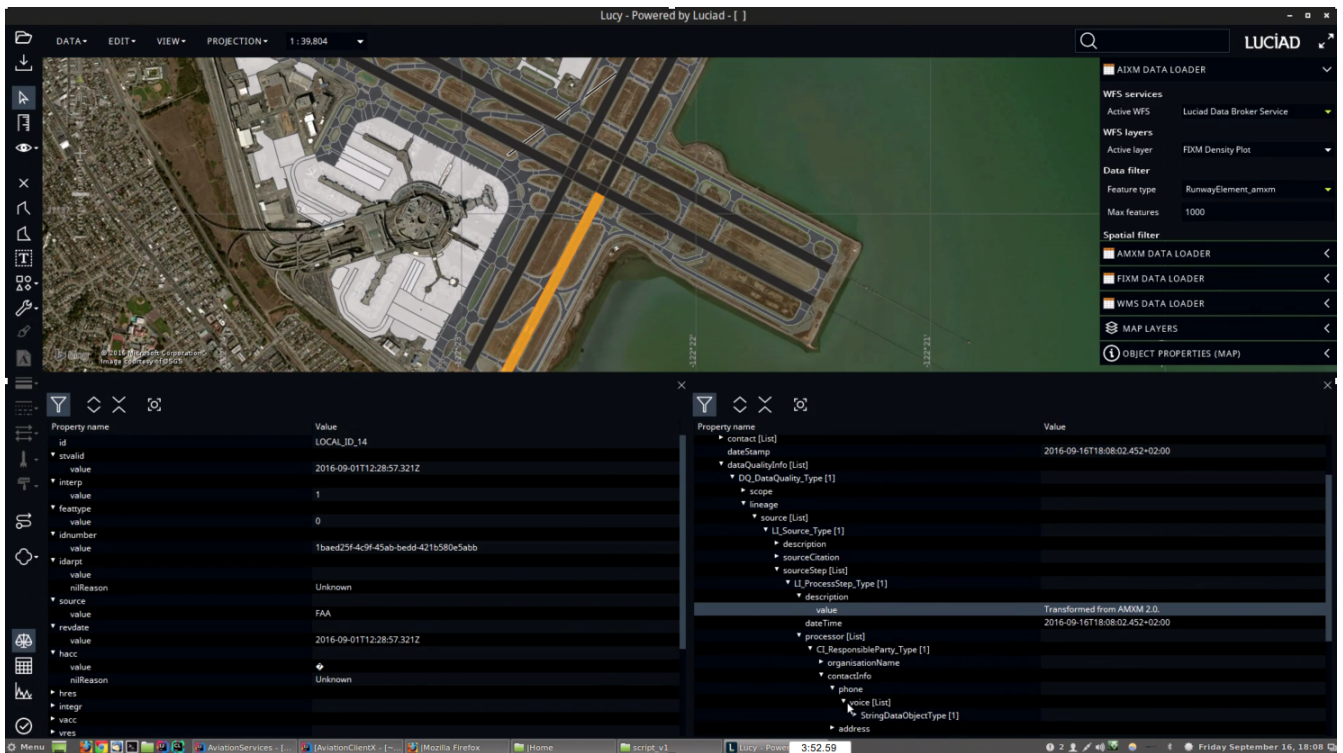


Figure 22. Data Broker converts AMXM data to AIXM

The data broker can then retrieve the KATL data in AIXM format.

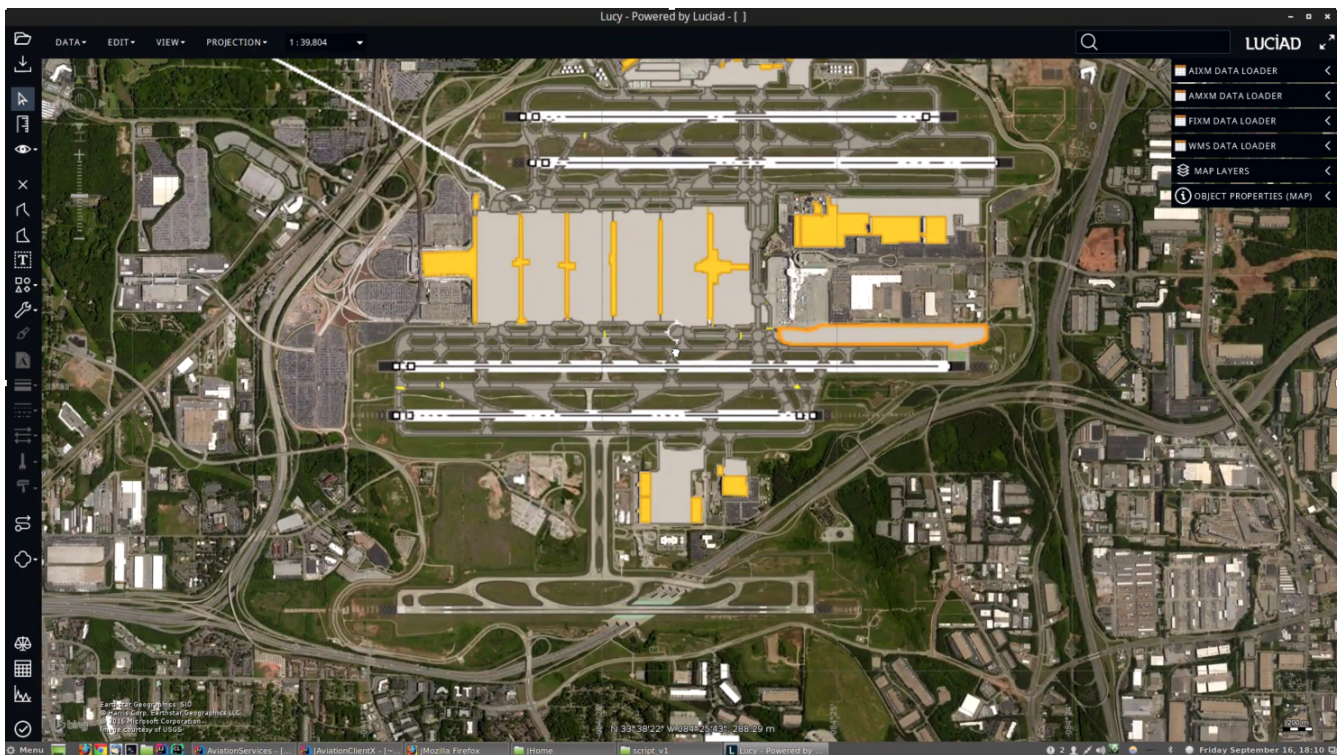


Figure 23. Data Broker retrieved AIXM format data

Finally, all this information is converted to a graphical image by the WMS service and displayed to the pilot.

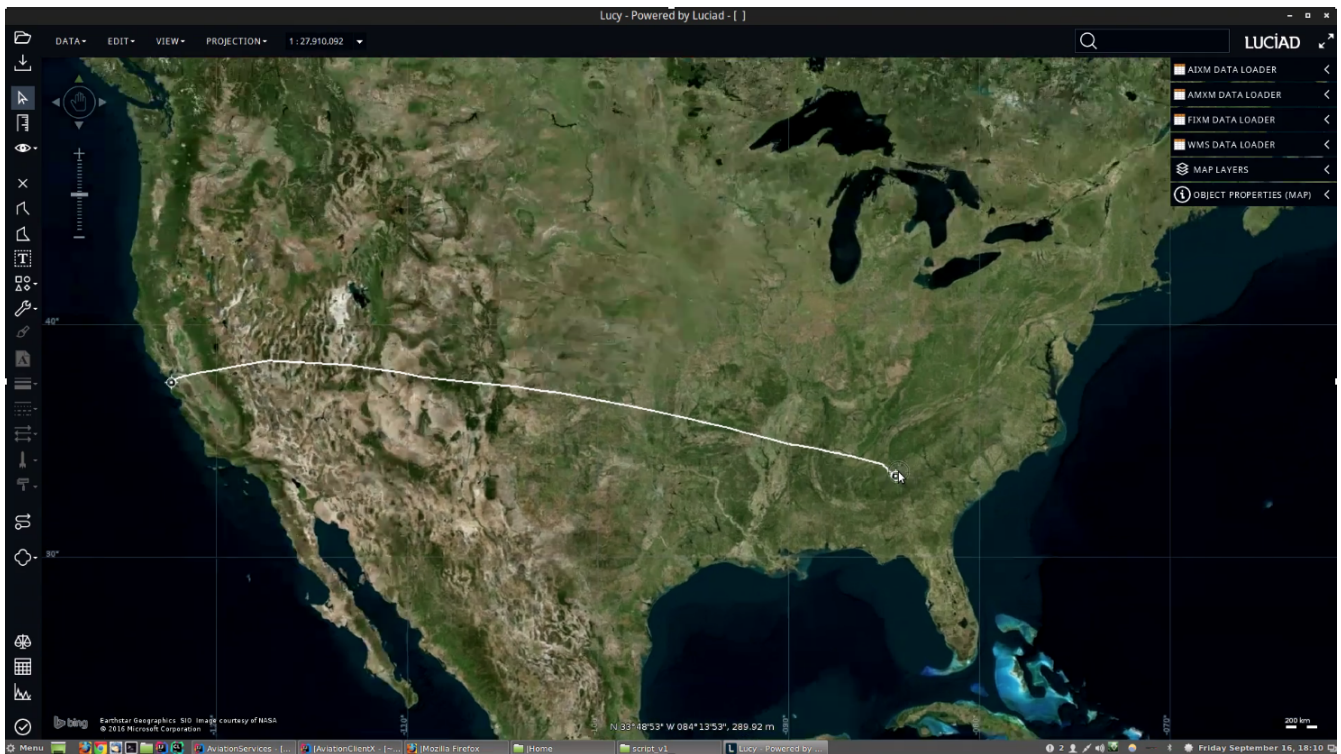


Figure 24. Data Broker orchestration of XML to Image

### 7.3.5. Benefits

- Use of Data Broker capabilities demonstrates simplified operations by using enhanced OGC services.
- Unified data models using WFS-enabled FIXM data containing GML + xlink references to AIXM/AMXM. The automatic conversion of XML based data into a map and layering the maps for a client such as an EFB demonstrates the flexibility of workflow management in an aviation use case.
- Better interoperability using Data Broker centralized service with AIXM/AMXM transformative capabilities. The automatic conversion of AMXM to AIXM demonstrates the improvement in service management across disparate data models using predefined business rules.
- Semantics used in the context of ISO 19119, classified as resources with spatial constraints. This is demonstrated using Catalog Service for Web to search across multiple services using semantic search capabilities.

## 7.4. Scenario 3

The fires have spread, putting the United Airline's Technical Operations Center (located in San Francisco) at risk. An evacuation has been called and remote AOC facilities are setup at an alternate location in Los Angeles. The Airline Operations Center (AOC) system must subscribe to live FAA flight data for the SFO airspace over SWIM. The AOC is a consumer of FAA NEMS, and they need to subscribe to AIXM and FIXM data in the SFO airspace using a Geospatial filter. The data is provided using the OGC PubSub 1.0 service specification, which enables the Virtual AOC facility to subscribe to a geospatial-filtered feed of Flight data and tracks.

### 7.4.1. Components

- F008 Asynchronous Messaging Client
- F009 Asynchronous Messaging Server
- Harris DEX Server (represents FAA NEMS Producer)
- Harris DEX Consumer (represents FAA NEMS Consumer)

### 7.4.2. Status Quo

FAA SWIM provides publish-subscribe data feeds in the Java Message Service (JMS) protocol. This protocol was abandoned by the European aviation community due to lack of wire-level control over the messaging. AMQP was adopted, but no operational SWIM services have yet been deployed with AMQP 1.0. Messaging protocols only provide a means for transporting the data, but a means for managing the subscriptions and content has yet to see widespread adoption. The OGC PubSub 1.0 specification enables content management and subscription management in a publish-subscribe pattern with any protocol binding desired.

### 7.4.3. Scenario 3 Sequence

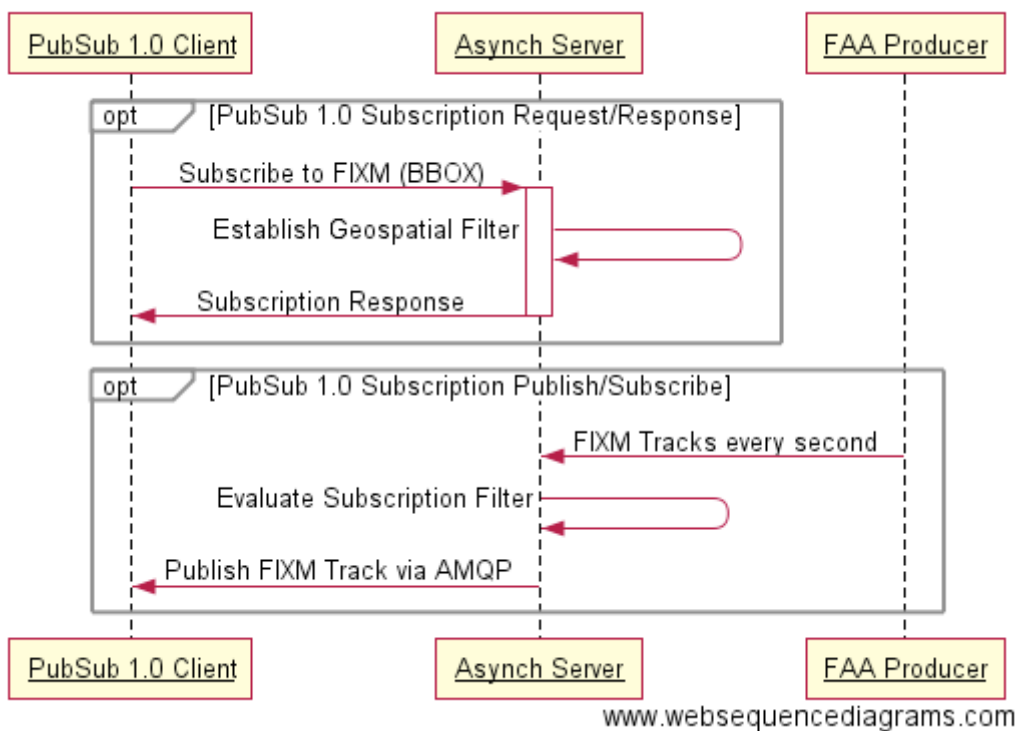


Figure 25. Sequence Diagram for Aviation Thread Scenario 3 - PubSub 1.0



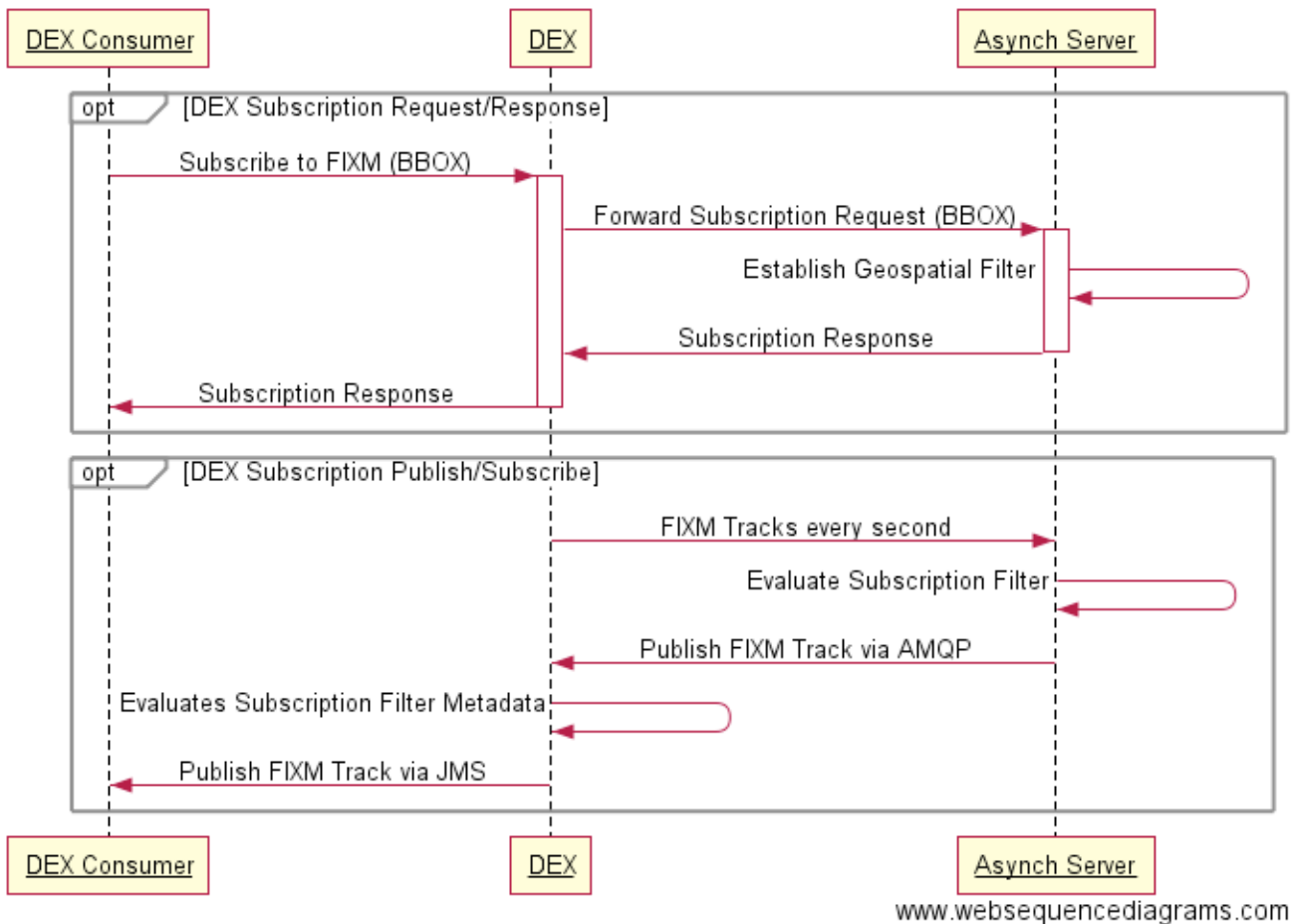


Figure 26. Sequence Diagram for Aviation Thread Scenario 3 - DEX Integration

#### 7.4.4. Results

In this scenario, a client is connected to the Harris DEX service, which represents the FAA SWIM NEMS. The figure below depicts the messaging flow.

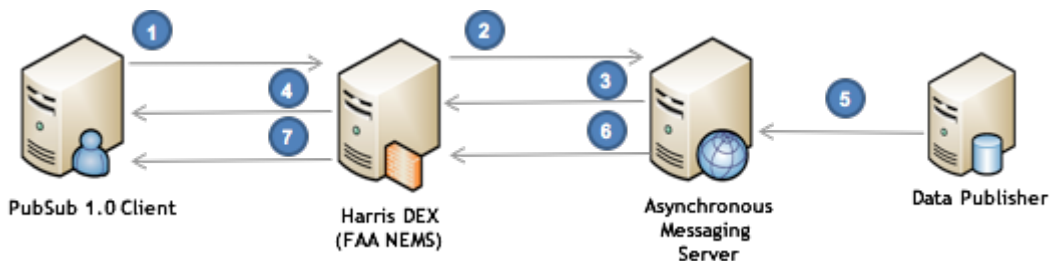


Figure 27. Interaction with Harris DEX

1. The PubSub 1.0 Client connects (via Secure VPN) to the Harris DEX and makes a web service request to the Asynchronous Messaging Server (PubSub 1.0 Server) containing a subscription request.
2. The Harris DEX proxies the request to the subscription server.
3. The subscription server establishes a subscription filter and responds with a confirmation.
4. The response is proxied back to the client by the Harris DEX.
5. The data publisher is publishing information to the Asynchronous Messaging Server. The server detects a match on a message against the subscription filter.

6. The server forwards a copy of the message to the AMQP subscription queue on the Harris DEX.
7. The Client retrieves the copy of the message off the AMQP queue.

The 52North aviation client, AviationFX, is used to demonstrate setting up a subscription.

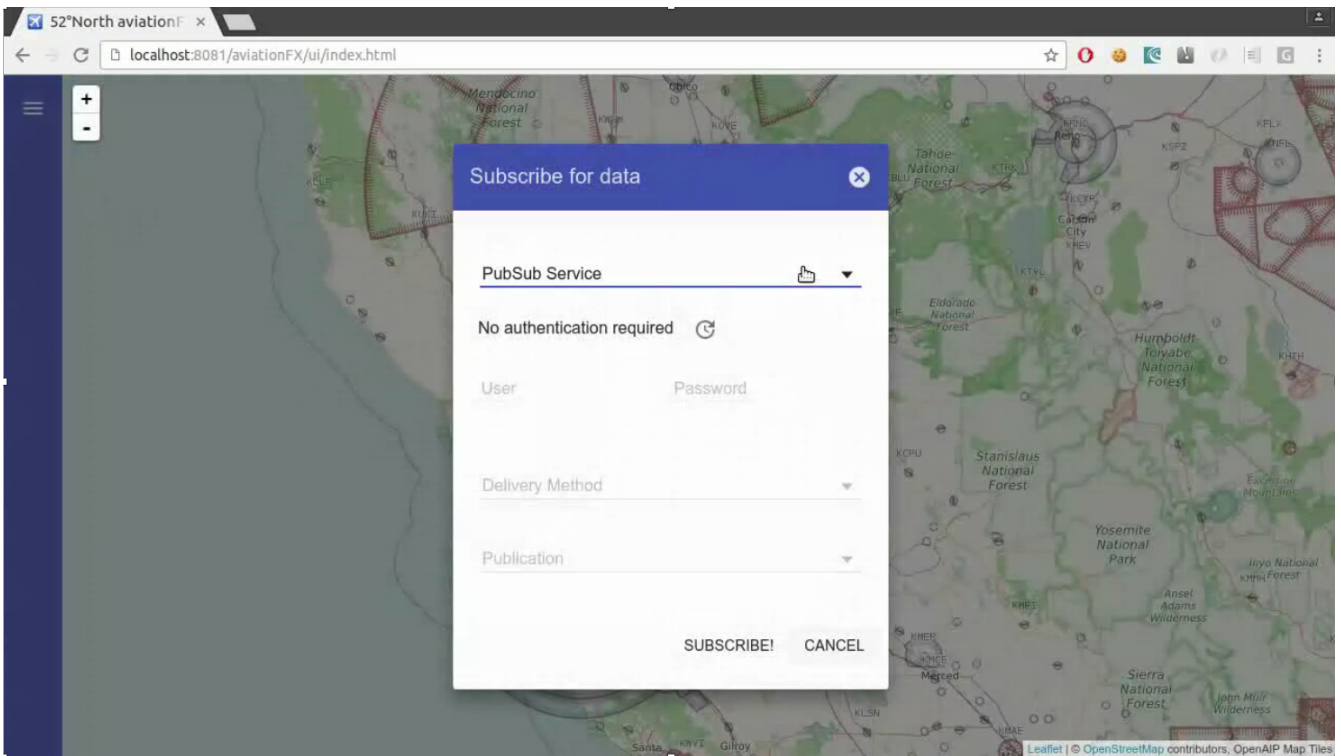


Figure 28. Subscribing to data in PubSub 1.0

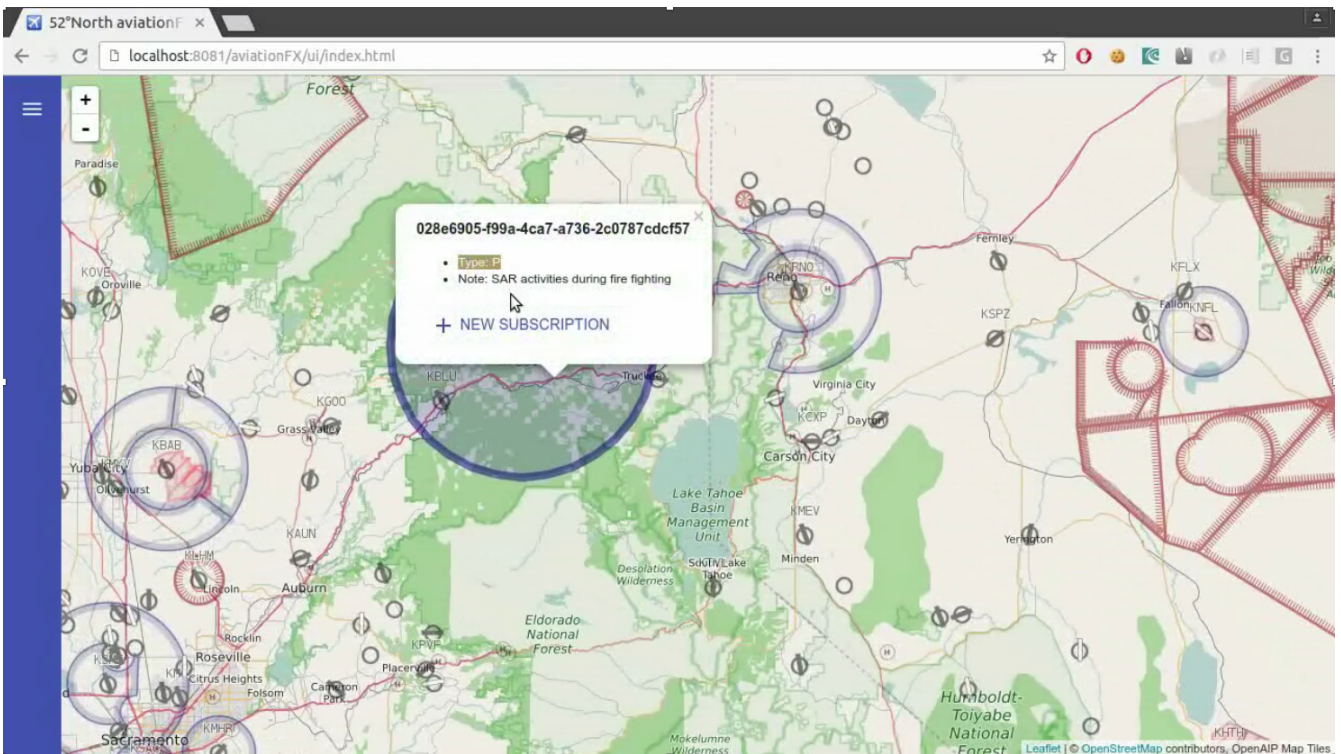


Figure 29. Subscription established

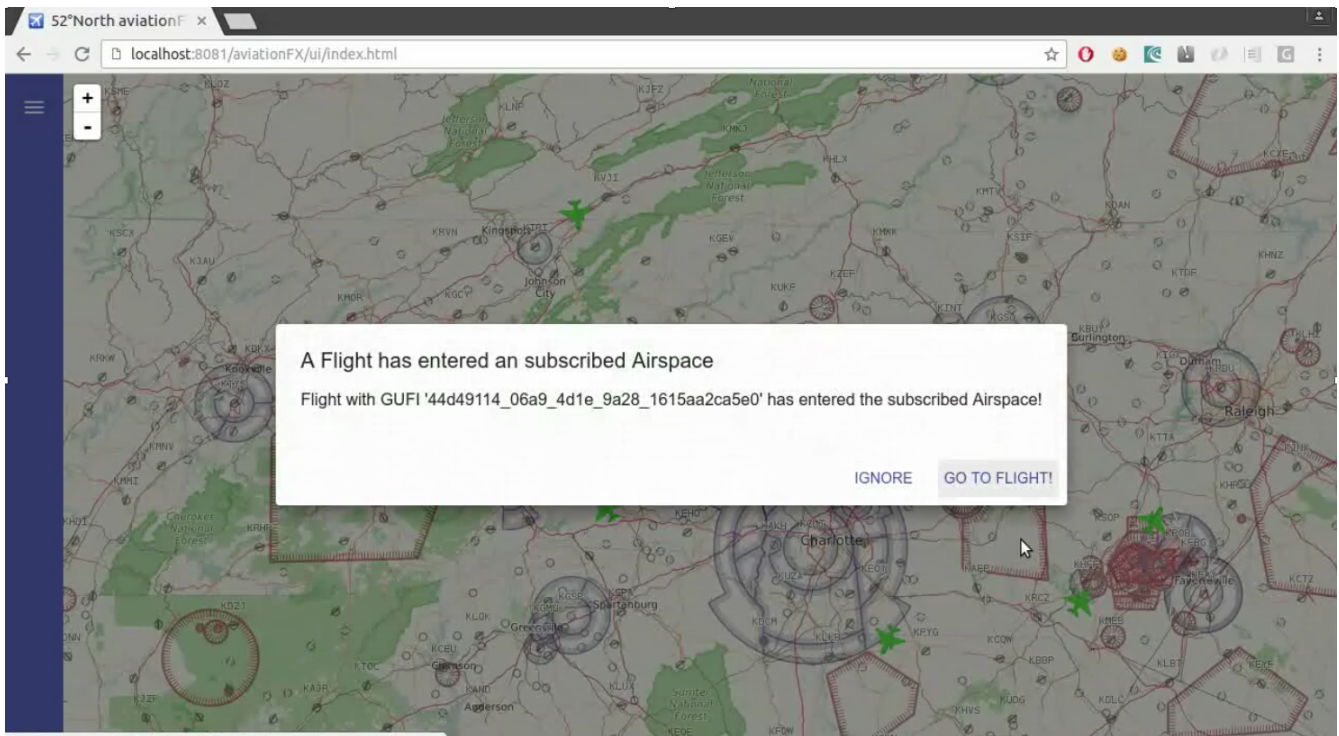


Figure 30. Match found according to the subscribed airspace filter

In this way, the airline operations center facility in San Francisco threatened by fire can be migrated to the Los Angeles location, and the subscription filters can be reconfigured to include both the San Francisco and Los Angeles areas.

#### 7.4.5. Benefits

- The publish/subscribe messaging pattern is better suited for eventful data such as aircraft tracks.
- Interoperability with OGC services is demonstrated through the use of the OGC PubSub 1.0 specification.
- Interoperability with FAA SWIM NEMS services is demonstrated using well-known protocol bindings and OGC standards.
- Data filtering using geospatial queries using OGC Filter Encoding Specification demonstrates the flexibility of OGC service specifications as compared to current statically configured data producers.

# Chapter 8. Lessons Learned

The lessons learned described in this section are from a high level testbed architect's perspective. The lessons learned contained herein are concerned with integration between components and are not meant to override the detailed lessons learned per each ER topic.

## 8.1. Web Service Architecture

In Testbed 12, the prominent theme was the use of enhanced features to extend the capabilities of current OGC web service standards. These features created challenges in how to best manage and describe these enhanced services.

## 8.2. Asynchronous Messaging Architecture

Testbed 12 is the first time OGC PubSub 1.0 specification was explored. The PubSub 1.0 Standards Working Group developed the PubSub standard 1.0 for managing data exchange using the Publish/Subscribe messaging pattern. FAA uses the publish/subscribe messaging pattern widely and can benefit from the use of PubSub standards to take advantage of the filter encoding specification (FES). The FES enables typical PubSub clients to specify geospatial bounded queries to filter event-based data. The Testbed 12 explores the interoperability between PubSub 1.0 specification and the FAA sponsor's publish/subscribe infrastructure called the FAA NEMS.

## 8.3. Data Broker

In Testbed 11, the use of Data Broker focused on the use of a WFS to extend to multiple WFS servers. In Testbed 12, the Data Broker was augmented with additional capabilities such as an embedded Feature Portrayal Service and Web Map Service to process XML feature data into graphical maps that can be presented to the client. In order to implement this flow, the workflow orchestration is required. Several solution options arose such as the use of processing instructions. In developing these options, it was determined that few standards exist in workflow management. Further investigation in workflow management can simplify process oriented operations that exist such as in aviation.

## 8.4. Catalog

The testbed activity explored use of catalog for semantic descriptions, notably the Service Description Conceptual Model (SDCM) from FAA and Eurocontrol sponsors. Furthermore, the use of the service descriptions for semantic queries needs further investigation. SDCM version 2.0 was release mid-way through the testbed and has not yet been investigated with Catalog.

During integration, it was learned that the service description for WFS-TE does not provide the necessary artifacts to enable the client to search specifically for a WFS that provides the temporality extension feature. The issue was raised in the Aviation Domain Working Group, only to be determined that this is possible a Catalog issue. Further investigation is necessary in the respective Catalog working group to determine how to best manage these extensions.

## 8.5. WFS-TE

The WFS-TE implementation used in the testbed is based on the current WFS-TE specification draft. It was tested whether the new flexibility gained by combining classic WFS filters with temporal filters, projections and transformations.

During integration, it was learned that clients can be adjusted quickly to support the most important types of temporal queries. However, extending the Data Broker turned out to be too difficult for implementation during the testbed. In consequence, WFS-TE had to be accessed by clients directly. However, non-temporal WFS queries still worked through the Data Broker, because WFS-TE extends WFS in a backwards compatible way. This demonstrated that extending existing protocols has a value in its own and is preferable to creating entirely new protocols.

Moreover, it was learned that the WFS "gml:id issue" (contradictory requirements that are already present in the classic non-temporal WFS standard) exists for real and is amplified in the WFS-TE temporal queries. In close cooperation with the Aviation DWG that works on finalizing the WFS-TE draft, this issue was raised again and a solution will be specified in the WFS-TE draft and hopefully be backported to the original WFS standard as well.

## 8.6. Aerodrome Mapping Exchange Model (AMXM)

The use of AMXM data in Testbed 12 propagates the data model standard to help widen the audience and potential users of this data model. During the development aviation thread scenarios, it was seen that there is a lack of sufficient data in AMXM. Current data models are in AIXM or ArcGIS. AMXM provided benefits in being less complex than AIXM, yet defined aerodrome geometries with the same benefits of GML-based schemas.

# Appendix A: Revision History

Table 2. Revision History

Date	Editor	Revision	Primary clauses modified	Descriptions
February 15, 2016	C. Chen	.1	all	initial version
June 22, 2016	C. Chen	.2	all	Draft ER Deliverable updates
June 28, 2016	T. Forbes	.3	all	Draft ER Deliverable updates
July 11, 2016	T. Forbes	.4	component-descriptions	FIXM WFS component
July 12, 2016	M. Rieke	.5	er-topics	Asynch Messaging ER overview
July 13, 2016	T. Thomas	.6	er-topics	SBVR ER overview
July 15, 2016	D. Balog	.7	er-topics, er-components	Data Broker ER & components
July 19, 2016	M. Pohl	.8	component-descriptions	WFS-TE component
July 20, 2016	A. Balaban	.9	er-topics	Semantic ER and Security ER
July 21, 2016	R. Martell	.91	component-descriptions	Catalog component
July 22, 2016	D. Balog	.92	component-descriptions	WMS/WMTS/WC S component
July 22, 2016	V. Diels-Grabsch	.93	all	typo fixes
July 25, 2016	C. Chen	1.0	all	Final Draft Release