OGC Testbed-13

Quality Assessment Service ER

Table of Contents

OGC Engineering Report

COPYRIGHT

WARNING

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.

Chapter 1. Summary

This Engineering Report (ER) has been produced in conjunction with two other engineering reports from the OGC Testbed 13, the Abstract Data Quality ER [4] and the Data Quality Specification ER [5] to capture status quo, discussions, and results in the context of requirements for data quality assessment for Quality of Service in the Aviation Domain. It will, in particular, provide a Data Quality Assessment Service Specification. Much of the ER is presented in the future tense, using terms such as 'shall', in order to express requirements and constraints on future Data Quality Assessment Service implementations. The service specification includes design patterns, extension mechanisms, and service interface considerations.

In recent years, the concept of data quality has generated a notable interest among System Wide Information Management (SWIM) [17] implementers, both organization-specific and global. In the context of SWIM — and Service Oriented Architecture (SOA) implementations in general — data quality pertains to two major use cases, service advertising and service validation:

Service advertising

> a service makes known to a potential consumer the quality of the data provided by the service. Based on this information, the consumer can determine whether or not the service meets its needs.

Service validation

> assurance is given that the quality of the data provided by a service is consistent with the quality that is explicitly defined in a service contract or any kind of agreement that may exist between a service provider and service consumer.

Both use cases share two common preconditions:

1. An unambiguous definition of the concept of data quality exists.

2. A set of measurable parameters that allow specifying data quality is defined.

These are tasks that were performed as part of Testbed 13. The findings of the tasks are documented in the Abstract Data Quality ER (FA001)[4] and the Data Quality Specification ER (FA002)[5].

The following diagram depicts the context of Aviation Traffic Management (ATM) service ecosystem, in which the Assessment Data Quality Service (ADQS) shall operate:
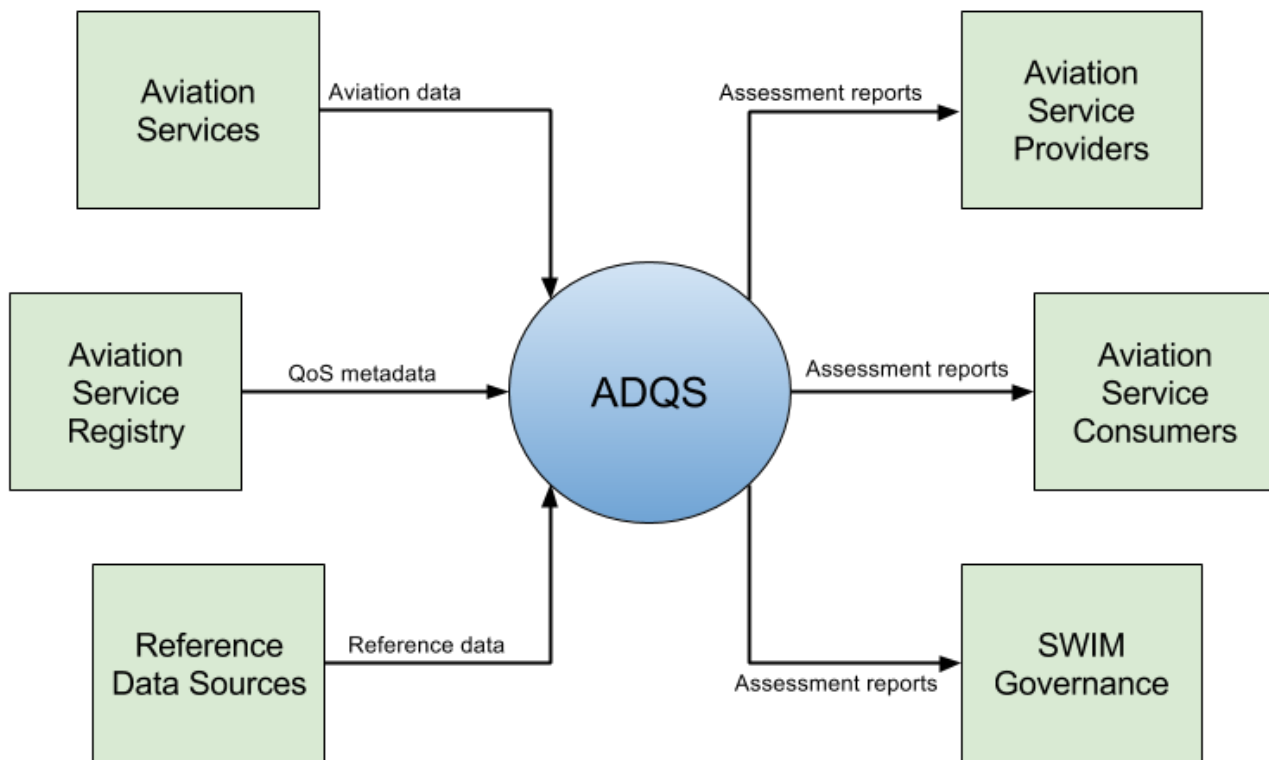
Figure 1. ADQS context diagram

Several data sources and stakeholders are relevant for an assessment service. The aviation service registry provides data quality statements, which are the subject of assessment. In order to evaluate data quality, the service might decide to collect probe data from the observed service. The data might further be compared with referential data provided by authoritative data sources. Both service providers and consumers might be interested in quality assessment reports, for example to check whether a service fulfills expected requirements.

This work shall also consider and extend a previously tested pattern for quality assessment, which separates the rules from the actual data and uses a Web Processing Service (WPS) to apply the rules to data. In detail, this pattern includes:

1. Data Service (holding the data for assessment)

2. Rules Registry (holding the rules to be applied to the data)

3. OGC WPS [3] Interface (access the Data Service and Rules Registry; applies the rules to the data; produces a report)

The ADQS has the potential to provide crucial contribution in establishing full-scale interoperability between aviation services. This can be achieved by specifying exact, machine and human readable metadata taxonomy to describe the level of data quality provided by services, as well as an innovative intermediary service responsible for assessing the claims made by service providers about quality.

1.1. Related Engineering Reports

The Abstract Data Quality ER (FA001)[4] presents an abstract or conceptual model for data quality in the context of SOA services in general and OGC-compliant services in particular. This model articulates the concept of data quality as well as associated facets, concepts and relationships. It is

based on widely-used industry standards and models, such as the Service Description Conceptual Model (SDCM) [8], the Quality Indicators Dictionary and Markup Language (QualityML)[7] and the ISO 19157[15] standard of the International Organization for standardization (ISO).

The Data Quality Specification ER (FA002)[5] presents a Data Quality Assessment Specification. This specification defines a set of data quality parameters as well as the methods and units of measure employed for measuring these parameters. The specification is information domain neutral, i.e., it specifies data quality characteristics and methods that can be applied to all aviation information domains: weather, flight, and aeronautical.

1.2. Requirements

The following considerations were included in the Call for Participation (CFP) so as to help implementers to see the "whole picture":

- The ADQS shall be an intermediary service. Its purpose is determining the quality of data produced by information services.
- The ADQS may receive data for assessment either by invocation of an information service or as directly inputted data info set. This may lead to use of the service either at design-time or run-time.
- The service shall evaluate the quality of data based on a set of criteria, which can either be specified in external document (e.g. organization-specific requirements for data quality) or can be provided as part of an ADQS request. The inputted parameters (criteria) should be derived from the Data Quality abstract model and assessment specification defined in FA001 and FA002 respectively.
- As an output, the ADQS should generate a machine-readable data quality assessment report.

This table provides the formal list of requirements:

Table 1. ADQS Requirements

Identification	Description
ADQS-1	This ER shall develop the data quality assessment service specification.
ADQS-2	The service specification shall consider outcomes derived from the Abstract Quality Model (AQM) and assessment specification defined in related Testbed 13 FA001 and FA002 engineering reports.
ADQS-3	The ADQS specification shall include design patterns, extension mechanisms, and service interface considerations.
ADQS-4	The service shall support future extensions to allow using it with domain-specific data quality parameters.
ADQS-5	The service shall receive data for assessment either by invocation of an information service or as directly inputted data info set.
ADQS-6	The service shall assess the quality of data based on a set of criteria, which can either be specified in external document or can be provided as part of request.

Identification	Description
ADQS-7	The assessment should be based on service metadata descriptions encoded using data quality abstract model (AQM) defined in Testbed 13 FA001 and FA002.
ADQS-8	The service shall generate a data quality assessment report, as an output.

1.3. Key Findings and Prior-After Comparison

Currently, neither in the aviation domain nor the OGC standards baseline, is there an adequate mechanism to describe the quality of (geospatial) data provided by a service as a subset of overall Quality-of-Service (QoS) attributes/taxonomies. It is not possible to express the expectations of service consumers regarding the quality of data or for service providers to advertise their services from the data quality perspective. The claims of service providers regarding quality of data cannot be validated nor assessed due to the lack of governance tools for data validation.

The AQM forms a basis for implementing data quality services specifically for the aviation domain. Prior to this testbed and FA001, data quality services were reliant on the standards inherited into the three aviation exchange models such as the Aeronautical Information Exchange Model (AIXM)[11], Flight Information Exchange Model (FIXM)[12] and Weather Information Exchange Model (WXXM)[13]. These vary from not having any capability to record data quality to the data quality metrics as defined in ISO 19157[15]. Although the ISO standard is recognized and provides some of the required data quality functionality, it is missing the key concepts of timeliness and precision. The abstract quality model described in FA001[4] document introduces these concepts to the AQM in a standard-compliant way, which is used in the data quality assessment service specification.

The outcome for this engineering report is the description of an Application Programming Interface (API) and design for the data quality assessment service. The ER investigates the options, proposes a solution and explains the use cases with the service interface suitable to meet all requirements put on this service. Following questions have been answered in this ER:

- How to use the assessment service?
- How to assess the data quality?
- How to tell the service which aspects of quality shall be assessed?
- How the service should know which elements/entities from a data set shall be used to assess certain quality aspect?
- How the quality assessment request and report documents should look like?
- Which API shall be used?
- How to design the assessment service to be extensible?
- Which message exchange patterns are required?
- Which binding shall be used to implement the API?

The ER also provides the WPS 2.0 profiles for data quality assessment service implementations.

1.4. What does this ER mean for the Working Group and OGC in general

In the area of aviation data quality, this engineering report, together with associated reports, provides a general blueprint for the implementation of the quality assessment service. While the ER considers aviation data formats and quality parameters specific for that business domain, it also remains general and open for further extensions.

1.5. Document contributor contact points

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

Table 2. Contacts

Name	Organization
Aleksandar Balaban	m-click.aero GmbH

1.6. Future Work

This engineering report describes the practical application of several aviation metadata and ISO standards (from the area of data quality) for an interface to access the data quality assessment service implementation. The WPS 2.0 standard plays here an essential role as the OGC Web Services (OWS) compatible, common purpose, extensible interface for accessing of geospatial computation routines. Data quality assessment fits very well into the concept of web "processing" service.

The following are recommendations for future efforts in standardization, that have been identified during this testbed, towards establishing the standard for data quality assessment service implementations:

1. A quality assessment service should be prototypically developed and tested. This service, as a component, could play a significant role in the architecture for the next testbeds. Both runtime and design-time use cases should be considered.

2. A more general service concept could be introduced in order to establish common rules for the design of a common-purpose quality-of-data validation service that is particularly applicable to OGC data types.

3. A WPS 2.0 profile concept should be used to create blueprints for different implementations of data quality of services.

4. Bloated Extensible Markup Language (XML) representation of QualityML concepts should be replaced by a more lightweight JavaScript Object Notation (JSON) version.

5. For all OGC data types, the elements of data schemas which are relevant for data quality should be identified, as well as a way of parsing data in order to prepare for the provision of assessments and application of quality metrics.

1.7. 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 normative documents are referenced in this document.

- OGC 12-080r2, OGC OWS Context Conceptual Model [https://portal.opengeospatial.org/files/?artifact_id=5518]

- OGC 06-121r9, OGC® Web Services Common Standard [https://portal.opengeospatial.org/files/?artifact_id=38867&version=2]

- OGC 14-065, OGC® WPS 2.0 Interface Standard [http://docs.opengeospatial.org/is/14-065/14-065.html]

- ISO/IEC 19510:2013, Object Management Group Business Process Model and Notation [http://standards.iso.org/ittf/PubliclyAvailableStandards/c062652_ISO_IEC_19510_2013.zip]

- Quality Indicators Dictionary and Markup Language (QualityML) [http://www.qualityml.org/]

- Service Description Conceptual Model (SDCM) [https://www.faa.gov/nextgen/programs/swim/governance/servicesemantics/media/SDCM_v2.0/SDCM_v2.0.html#3.13]

- Concept of Operations for the SWIM Common Registry (SCR) [https://www.faa.gov/nextgen/programs/swim/governance/outreach/media/SCR%20CONOPS%20FINAL%20DRAFT%2004042015.pdf]

- The Aeronautical Information Exchange Model (AIXM) [http://aixm.aero]

- The Flight Information Exchange Model (FIXM) [http://fixm.aero]

- The Weather Information Exchange Models and Schema (WXXM) [http://wxxm.aero]

- ISO/IEC 18384-3:2016, Information technology — Reference Architecture for Service Oriented Architecture (SOA RA) — Part 3: Service Oriented Architecture ontology [https://www.iso.org/obp/ui/#iso:std:iso-iec:18384:-3:ed-1:v1:en]

- ISO 19115-1:2014, Geographic information - Metadata - Part 1: Fundamentals [https://www.iso.org/standard/53798.html]

- ISO 19157:2013, Geographic information - Data quality [https://www.iso.org/standard/32575.html]

- ISO/TS 19115-3:2016, Geographic information - Metadata - Part 3: XML schema implementation for fundamental concepts [https://www.iso.org/standard/32579.html]

- FAA System Wide Information management [https://www.faa.gov/nextgen/programs/swim/]

- Publish/Subscribe Interface Standard 1.0 - Core [http://docs.opengeospatial.org/is/13-131r1/13-131r1.html]

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 [https://portal.opengeospatial.org/files/?artifact_id=38867&version=2] shall apply. In addition, the following terms and definitions apply.

3.1. Abstract Quality Model (AQM)

A conceptual model for data quality in the context of SOA services in general and OGC-compliant services in particular. It is based on Service Description Conceptual Model (SDCM), ISO19157, and QualityML and will deal with completeness, logical consistency, positional accuracy, temporal accuracy and thematic accuracy issues to improve quality description in the metadata.

3.2. Air Traffic Management (ATM)

Air traffic management is an aviation term encompassing all systems that assist the aircraft to depart from an aerodrome, transit airspace, and land at a destination aerodrome, including air traffic control (ATC), air traffic safety electronics personnel (ATSEP), aeronautical meteorology, air navigation systems (aids to navigation), Air Space Management (ASM), Air Traffic Services (ATS), and Air Traffic Flow Management (ATFM), or Air Traffic Flow and Capacity Management (ATFCM).

3.3. Assessment

To assess is to "evaluate or estimate the nature, ability or quality" of something, or to "calculate or estimate its value or price." Assess comes from the Latin verb assidere, meaning "to sit by." As a synonym for measurement, assessment implies the need to compare one thing to another in order to understand it. However, unlike measurement in the strict sense of "ascertain[ing] the size, amount, or degree of something by using an instrument," the concept of assessment also implies drawing a conclusion, evaluating the object of the assessment.

3.4. Aviation Data Quality Assessment Service (ADQS)

A service used to assess the quality and the fitness for purpose of the (aviation) data. It answers the questions (amongst others): How complete and how consistent are the data? How precise are the coordinates or the semantic information?

3.5. Data Governance

This describes an organization's functional approach to the management of data by actively linking integrated business and technology teams with corporate and strategic initiatives.

3.6. Data Quality

The degree to which a set of inherent characteristics of data fulfills requirements [based on ISO 9000:2005, 3.1.1].

3.7. Data Quality Assessment (DQA)

is the process of scientifically and statistically evaluating data in order to determine whether they meet the quality required for projects or business processes and are of the right type and quantity to be able to actually support their intended use. It can be considered a set of guidelines and techniques that are used to describe data, given an application context, and to apply processes to assess and improve the quality of data.

3.8. Service-Oriented Architecture

Service-Oriented architecture (SOA) is an architectural style in which business and Information Technology (IT) systems are designed in terms of services available at an interface and the outcomes of these services (ISO/IEC 18384-3:2016).

3.9. Services

Reusable software components that are "callable" by a software application or other software components to perform a business function. An SOA service may invoke other SOA services as well as other types of services such as Web services, communications services, and I/O services.

3.10. System Wide Information Management (SWIM)

SWIM is a Federal Aviation Administration (FAA) advanced technology program designed to facilitate greater sharing of Air Traffic Management (ATM) system information, such as airport operational status, weather information, flight data, status of special use airspace, and National Airspace System (NAS) restrictions. SWIM will support current and future NAS programs by providing a flexible and secure information management architecture for sharing NAS information.

Chapter 4. Abbreviated terms

Abbreviation	Meaning
ADQS	Aviation Data Quality Service
ADQS	Assessment Data Quality Service
AIM	Aeronautical Information Management
AIS	Aeronautical Information Services
AIXM	Aeronautical Information Exchange Model
API	Application Programming Interface
ANSPs	Air Navigation Service Providers
AQM	Abstract Quality Model
ATM	Air Traffic Management
COTS	Commercial Off The Shelf
ETMN	European Air Traffic Management Network
FAA	Federal Aviation Administration
FIXM	Flight Information Exchange Model
ICAO	International Civil Aviation Organization
ISO	International Organization for Standardization
NOTAM	Notification to Airman
OWS	OGC Web Services
QoS	Quality of Service
QualityML	Quality Indicators Dictionary and Markup Language
SDCM	Service Description Conceptual Model
SES	Single European Sky
SESAR	Single European Sky ATM Research
SLA	Service Level Agreement
SOA	Service Oriented Architecture
SWIM	System Wide Information Management
WSQM	Web Service Quality Model
WXXM	Weather Information Exchange Model

Chapter 5. Overview

This ER is organized as follows:

1. The chapter titled Summary provides general explanation and justification for this engineering report.

2. The chapter titled Service Design Considerations provides considerations regarding the design and the implementation of an Aviation Data Quality Assessment Service (ADQS). It also depicts a component diagram and logical decomposition.

3. The chapter titled Architecture Proposal contains a design proposal. It depicts several architectural views, such as component and deployment views.

4. The chapter titled Service Application Interface explains the choice of WPS 2.0 for an ADQS service interface and describes the possible interactions between ADQS and service consumers.

5. Based on the hierarchical profiling approach defined in the WPS 2.0 standard, the chapter titled ADQS WPS Profile defines different profiles for data quality assessment processes.

6. The chapter titled Extension Mechanism evaluates and extends the proposed design from the service extension capabilities perspective.

7. The chapter titled Examples for data quality metrics and calculations provides assessment examples explaining how the ADQS could be used for typical quality assessments based on the abstract data quality model.

Chapter 6. Service design considerations

This chapter is dedicated to service design options and alternatives concerning the architecture and implementation. It also depicts possible component diagrams and logical decomposition.

Design options which are evaluated here include analysis of the following:

1. Service orientation and microservice architectural paradigm.

2. Software frameworks vs cloud service vs customized development.

3. On premises vs cloud based deployment.

4. Availability and utilization of external cloud services.

For the service interface, we considered:

1. WPS 2.0 as the OWS compatible processing model

2. Representational State Transfer (REST) architecture

3. OGC PubSub 1.0 in case that publish/subscribe behavior has relevance and is required

The microservice paradigm is used to describe application landscapes composed of numerous autonomously deployed, self-sufficient services with complex interactions. Having said that, it is also useful for flexible quality assessment service implementation and shall be recommended. The autonomous micro "services", in the case of the ADQS architecture, would be those components which are required for interaction with service consumers, process execution management, data collecting and quality assessment metric calculations.

Extensibility is the second important requirement which shapes ADQS architecture. In order to ensure efficient development and deployment of new validation features without impact on the service availability, the service shall be developed utilizing a process-centric approach based on a Business Process Model and Notation (BPMN) process model and an execution engine for quality assessment calculation. Further, the WPS 2.0 compatible interface shall be used to abstract the access to the service's core functionalities.

While the implementation of a WPS 2.0 compatible interface from scratch would be time-consuming, simple processes can be rapidly implemented when using software frameworks such as the deegree WPS. The frameworks relieve developers of much of the overhead and the focus can be put on implementing data validation processes.

Regarding deployment and operation, the ADQS is recommended to be implemented as an on-premise solution. Some of the required processing algorithms could also be provided by external party service providers. For example, the Amazon Web Services (AWS) Cloud provides very advanced calculation services based on artificial intelligence, as well as statistical and heuristic methods. These functionalities, if useful for the data assessment service, could be easily integrated into ADQS architecture (if it was designed and implemented based on SOA and microservice paradigms).

Chapter 7. Service architecture proposal

This chapter provides design/implementation options, along with diagrams depicting architectural views such as component and deployment diagrams. Additionally, the Extensions Mechanism chapter further enriches the proposed design providing a detailed overview and explanation regarding the support for service extensions.

OGC WPS 2.0

Based on initial requirements and investigations performed as part of this testbed, the decision has been made to propose the version 2.0 of the OGC WPS standard as the service interface for ADQS. This choice has implications regarding the service architecture, since the WPS specification indirectly constrains the design decisions. The overall ADQS architecture has thus to be built around an the WPS component of an API. This component initiates and controls all execution steps required to receive an assessment service request, understand the assessment task, and perform test data retrieval and finally the assessment.

As options for the WPS binding the investigations have also been made regarding the standard, lightweight REST model and Simple Object Access Protocol (SOAP). The OGC PubSub 1.0[18] standard could be used in conjunction with the WPS. PubSub 1.0 would offer an additional, OGC-compliant publish/subscribe interface for data quality assessments.

REST service model

The WPS interface could be implemented using the REST paradigm but it does not fit well in the concept of data processing because it has been specified with the focus on the accessing of resources from the internet. Still, it could be used to implement the (abstract) WPS 2.0 interface, for example HTTP PUT could be used to "put" (initiate) the new assessment request in the service, HTTP GET could be used in combination with a Uniform Resource Locator (URL) to fetch the quality assessment result or the status of data processing in the case of asynchronous invocation, while HTTP DELETE could cancel the assessment process execution or erase an assessment report. The use of REST would be consistent with emerging versions of other OGC standards such as Web Feature Services (WFS) 3.0.

SOAP service model

A similar way to REST but with significantly more XML elements, the services described in the WPS 2.0 could be implemented using the SOAP paradigm.

Publish/Subscribe

It could be utilized for certain use cases, for example, polling the assessment service periodically to assess the data quality (for certain services under observation) and publishing the assessment results via URLs. This concept is applicable for use case scenarios involving permanent evaluation of data quality offered by observed services and delivering the assessment results to subscribed service consumers. Support for this sort of interaction is complementary to the WPS and it would allow assessment of data quality in a more real-time fashion.

7.1. User view

This section provides an introduction into stakeholders and basic functions provided by the ADQS.
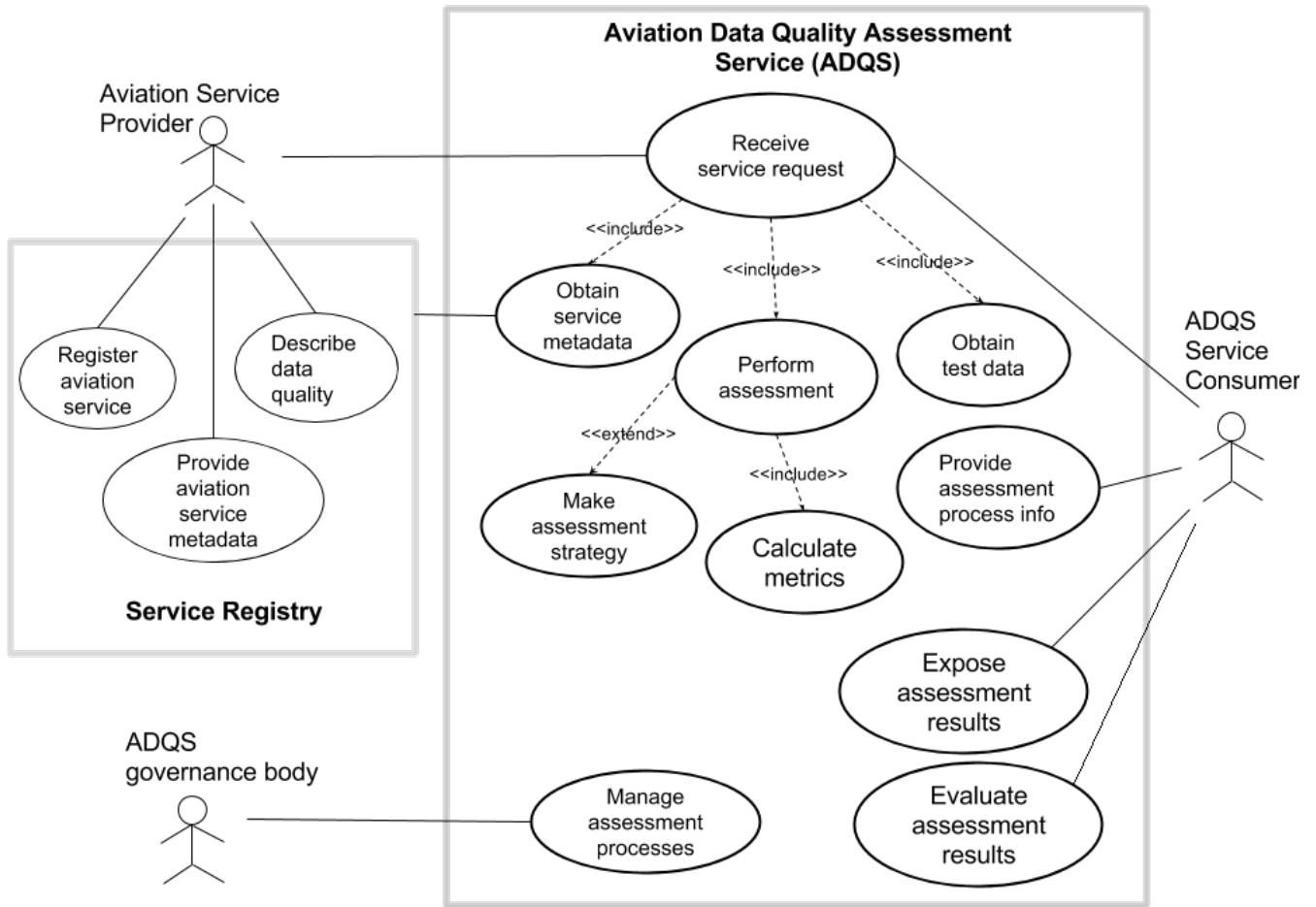
Figure 2. Use case diagram for data quality assessment process

7.2. Component view

This section provides the list of components recommended for the ADQS implementation. The architecture is based on two pillars. The first one is the Service Endpoint, which is responsible for mediation between service consumers and quality assessment logic. The second one is the Workflow Engine, which provides high implementation flexibility and service availability.
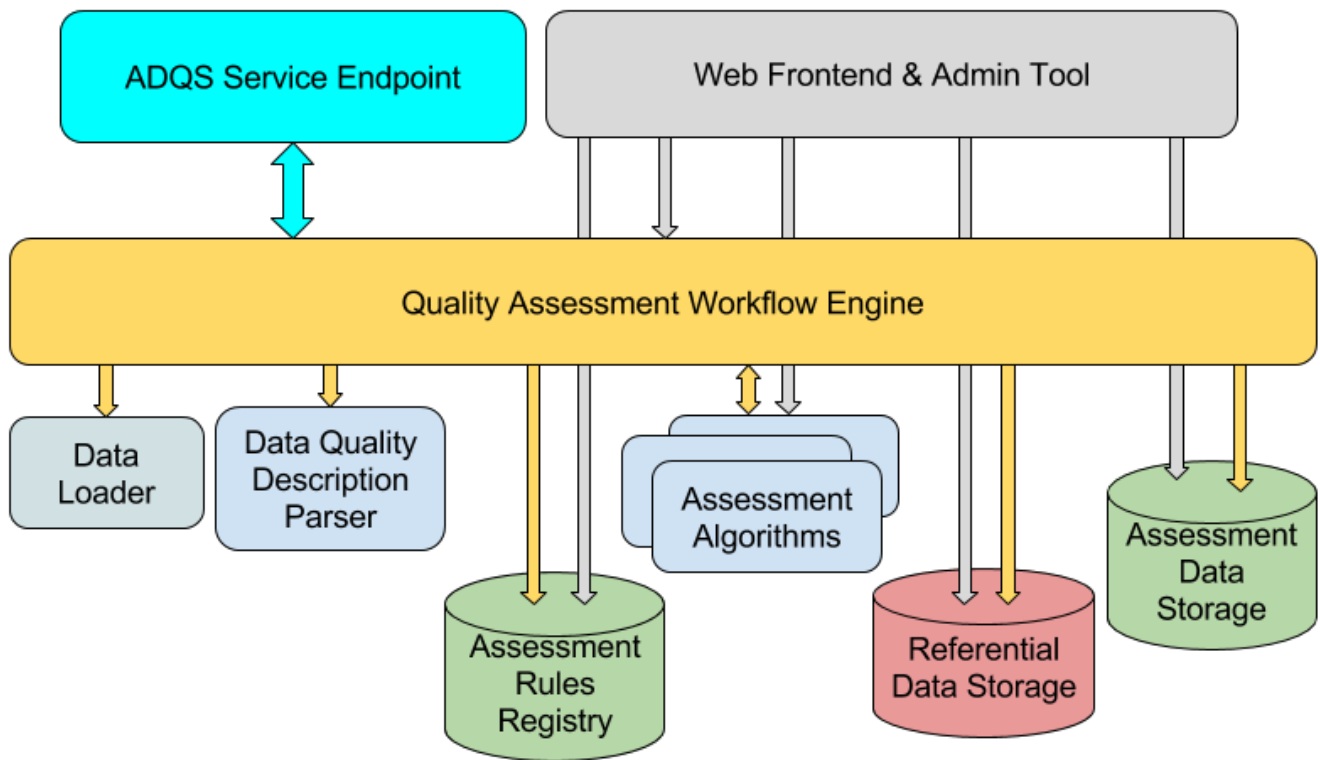
Figure 3. Components of Data Quality Assesment Service

Essential for a successful implementation is the integration of the service front-end component and the workflow engine. The WPS 2.0 specification supports synchronous and asynchronous service execution offering the operations for metadata retrieval, task execution, status check, and result retrieval. A WPS-based service interface reduces options for workflow design and implementation because WPS 2.0 does not support arbitrary interactions between assessment processes and human actors.

7.2.1. Data Quality Description Parser

This component reads data quality attributes encoded in the Abstract Quality Model (AQM) taxonomy and transfers it to an internal format suitable for assessment execution.

Quality data are available from an aviation service registry in SDCM taxonomy extended with AQM elements. The metadata description could be provided as an input parameter into the assessment. Otherwise, a URL to a service registry could also be provided, in which case this component becomes responsible for retrieving the metadata.

7.2.2. Workflow Engine

This component represents the implementation core because it coordinates all activities related to the data quality assessment. Based on the quality attributes encoded in the AQM taxonomy, it creates and executes an assessment process. This process can be executed with or without human assistance as follows:

1. Obtain the endpoint, the service endpoint, and the service registry entry.

2. Load and parse service data and data quality statements in SDCM, if available.

3. Analyze input data and derive a quality assessment workflow strategy, if required.

4. Execute the assessment workflow by invoking algorithms and calculating metrics.

5. Compare the assessment results with data quality statements.

6. Provide the assessment report as the output.

If the data quality assessment requires longer execution times involving long-running background assessment evaluations, it shall be implemented with the asynchronous message exchange pattern.

7.2.3. Assessment Algorithms and Rules Registry

The Assessment Algorithms and Rules Registry holds the set of configurable calculation rules and components, which are used in assessment workflows to perform calculations of metrics (values) and evaluations. The registered items might be defined declaratively as rules enforced as part of assessment process or be procedural as standard software components. This component also contains a set of mapping rules (related to data types) used to identify objects, which shall be used in the data quality assessment and metric values calculations. These objects are mostly geographic coordinates, altitudes or other data entities, which play roles in quality data assessment, for example regarding the data completeness.

7.2.4. Service Invoker

This component is used to invoke observed services and collect response data for quality assessment. Retrieved data is usually held in the Data Storage prior to the evaluation.

7.2.5. Assessment and Referential Data Storages

These components are used as temporary or permanent storage places for assessment and referential data sets. Referential data sets (also known as reference data sets) are well-defined data, known to be of high quality, which were obtained from authoritative sources.

7.2.6. Service Endpoint

This component enables service invocation and provides all interface-related functionalities specified by ADQS requirements. The recommendation is to implement the service endpoint based on the OGC WPS 2.0 specification. This component is responsible for managing the entire interaction between service consumers and the execution logic for different message exchange patterns used for service invocation. The endpoint shall provide support for the following assessment service specific invocation options:

- The ADQS service request shall contain service endpoint and test data obtained by a service consumer. ADQS will perform assessment based on these data.

- The ADQS WPS service request shall contain a link to a service registry containing service endpoint and data quality statements. This metadata is likely to be encoded using the SDCM taxonomy enriched for ADQ statements developed in FA001 and FA002. In this case, the ADQS will try to retrieve the data and perform the assessment based on data quality description, endpoint information and other metadata.

7.2.7. Web Front-end and Administration Tool

This component implements the administration and service execution tool. It shall be used for service management and monitoring activities. In other words, to control service access rights, referential data sets, monitor service executions, support the deployment of new assessment processes, rules, and algorithm components.

Chapter 8. ADQS application interface

The ADQS application interface shall enable the interaction between the service backend functionality (assessment processes) and the consumers. The interface shall receive data quality assessment input data, validate it, and forward it to the background computational environment (service implementation).

While this API could be designed starting from scratch using available standards and utilizing common design options, considering the requirements which prescribe a web-based interface, the state-of-the-art technology, and compatibility with the OGC OWS Common specification, the API design and implementation shall follow the 2.0 version of the WPS specification.

There are many reasons to adopt the WPS standard for ADQS. The version 2.0 of the WPS is a brand new, widely accepted OGC standard, which enables service consumers to interact with any kind of (geospatial or non-geospatial) backend computational processes. It provides an OWS-conforming way to create, operate, advertise and consume services independent on their implementations.

The WPS also supports synchronous and asynchronous invocation and service execution. Flexibility in implementation and support for two major message exchange patterns are especially valuable considering the requirement put on the assessment service to be performant, versatile and easily extensible.

As with the other OGC specifications, it is the development, publication, and adoption of profiles which defines the specific uses of this specification. Therefore, the preferable approach for adoption of the WPS as the API for the ADQS would be to first develop an abstract aviation data quality assessment profile for WPS compatible assessment processes.

For the purpose of aviation data quality assessment, under the term "process" this engineering report understands the special case of the data quality assessment process.

8.1. The WPS 2.0

The OGC WPS 2.0 Interface Standard provides rules for standardizing how inputs and outputs (requests and responses) for geospatial processing services can be structured in a standard way. WPS also defines how a client can request the execution of a process, and how the output from the process is handled. WPS also defines a standardized interface that facilitates the discovery of and binding to those processes by clients.

Processes can include any algorithm, calculation or model that operates on spatially referenced data. For example, a quality assessment process could be understood as an applicable calculation.

Publishing means making available machine-readable binding information as well as human-readable metadata that allows service discovery and use.
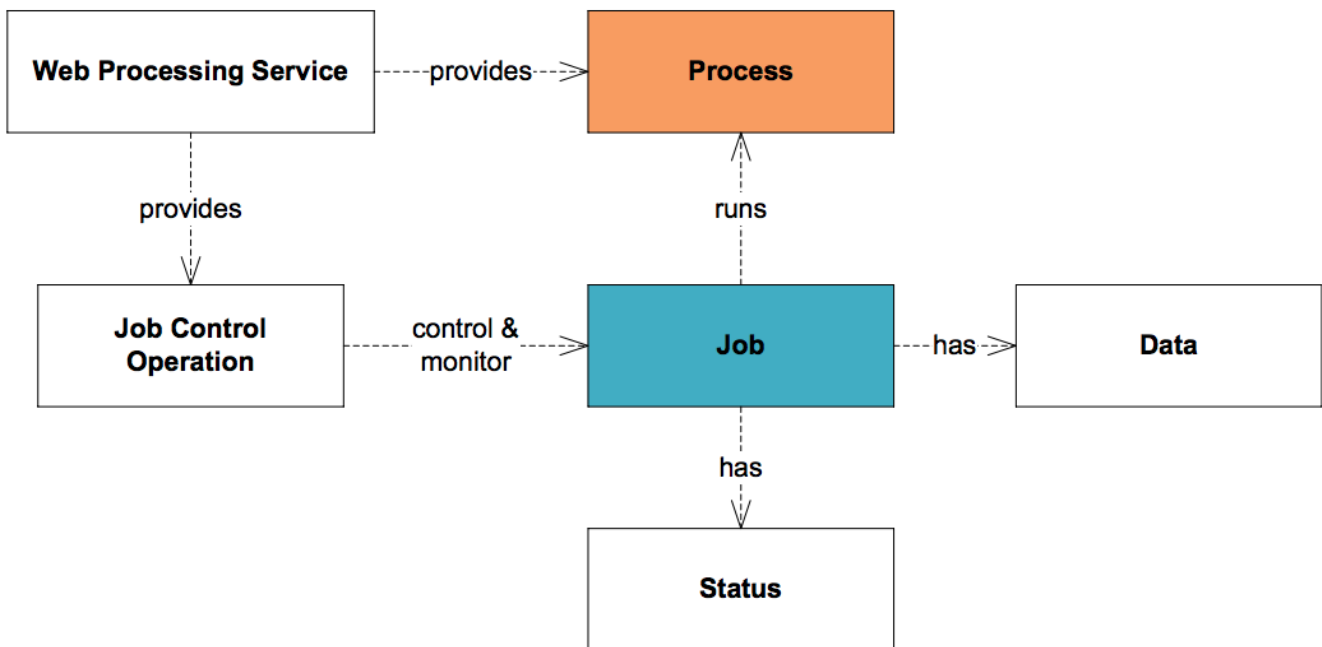
Figure 4. WPS 2.0 the core

The version 2.0 of the WPS specification allows three types of input data:

1. Complex data includes XML, imagery, and custom or proprietary data structures.

2. Literal data includes single numerical values or text strings.

3. Bounding box data includes geographic coordinates for a rectangular area.

The WPS does not specify the kind of backend business logic that could be implemented. Instead, it specifies a generic mechanism that can be used to describe and web-enable any sort of (geospatial) process. That also means that the service interface was kept generic, which makes it always applicable but also less expressive of one particular use case.

The main advantage of WPS is that it allows delivery of calculations (in this case the quality assessment) to consumers independent of the underlying software, implementation, algorithms etc. This independence helps to ensure the longevity of code.

Other advantageous WPS features are:

Multiple web service support

Both REST and SOAP bindings are possible for WPS 2.0.

Reusability

Exposing processes through WPS allows organizations to reuse the same services in multiple applications.

Flexibility

Exposing processes through WPS allows service providers to change the underlying software without impacting users and client applications.

Scalability

WPS is scalable, both in terms of the potential for unlimited parallel interfaces, and through the

exposure of gridded, cloud, or super-computers through a simple middleware interface.

Security

Access to assessment services via a WPS front-end can be restricted using a variety of standard web service security mechanisms such as firewalls and HTTPS.

8.2. Options for WPS 2.0 supported ADQS

The WPS allows several different approaches for executing a quality assessment process:

Choose process type and version

The quality assessment process might be available in many variants and versions.

Returning outputs embedded in XML

The response to an Execute request is an XML document that includes metadata about the request, as well as the outputs from the assessment process. This form of response is recommended when the size of assessment report is less than a few megabytes in size, and the user requires metadata found in the wrapper.

Storing outputs

A WPS may allow a user to request storage of the assessment results. In this case, the assessment document returned to the client will contain references to web-accessible locations from which the assessment reports can be downloaded.

Long-running processes

If an Execute request triggers a long-running process, the WPS will return a response containing references to the outputs. Also included will be a reference to a location where the Execute response document is stored. The WPS will periodically update the status indicator found in this document until processing is complete.

Access to quality assessment results

The outputs from a WPS are available to the client that initiated the Execute operation. There are several ways in which the access to the output could be provided:

- The WPS server creates an index of status documents created by the service.

- The WPS client create an index of status documents created by the service.

- The WPS server or client add the outputs to a Registry.

Service Composition with WPS

A WPS process is a function that performs a specific geospatial, validation or other calculation. Chaining of WPS processes facilitates the creation of repeatable workflows. WPS processes can be incorporated into bigger services in a number of ways:

- A process execution engine, for example, the Business Process Model and Notation (BPMN)[6], can be used to orchestrate a data processing chain that includes one or more data validation steps.

- A WPS process can be designed to call a sequence of external web services (for example for

8.3. Overview of WPS Core Operations

The WPS 2.0 specification can be utilized for data quality assessment service (as a service endpoint blueprint) by exposing the following operations to service consumers:
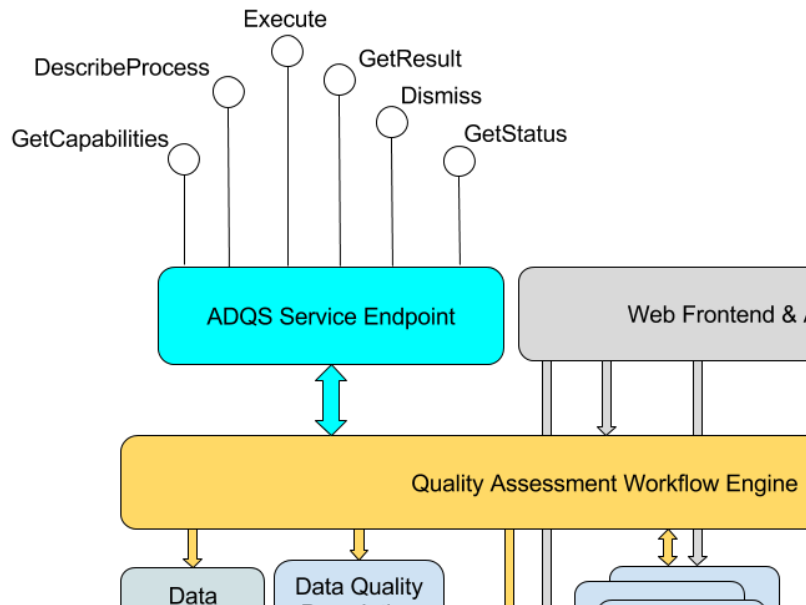
Figure 5. WPS 2.0 interface on top of ADQS implementation

The execution of quality assessment is then performed through the following interaction sequence that may be initialized by a WPS client and performed via a WPS front-end:
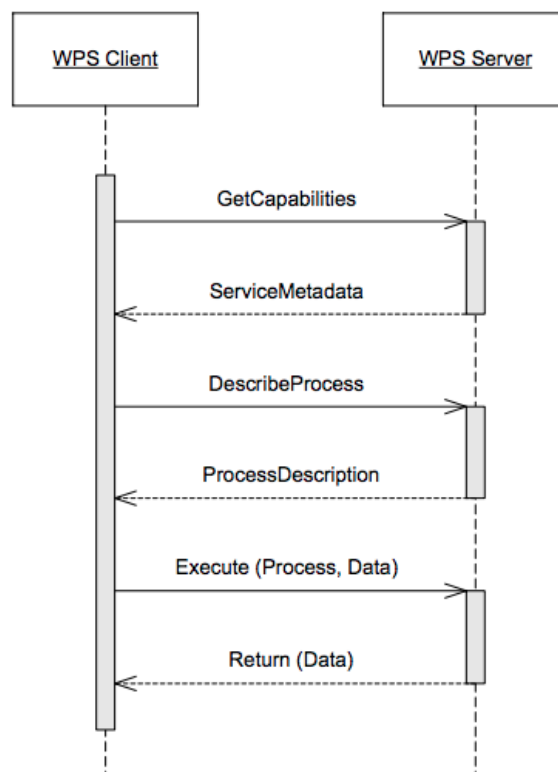
Figure 6. Basic interactions with an ADQS WPS 2.0 process

GetCapabilities

This operation allows a client to request information about the server's capabilities and processes offered.

DescribeProcess

This operation allows a client to request detailed metadata on selected processes offered by a server.

Execute

This operation allows a client to execute a process comprised of a process identifier, the desired data inputs and the desired output formats.

WPS discovery and binding mechanisms follow the OWS model [2] in that the WPS defines a GetCapabilities operation, and requests are based on HTTP Get and Post.

In case of asynchronous process execution, as it has been depicted on the second diagram, two additional operations are responsible for the execution monitoring:

GetStatus

This operation allows a client to query the status information of a processing job.

GetResult

This operation allows a client to query the results of a processing job.

Dismiss

Cancel / release running job and also release server-stored results
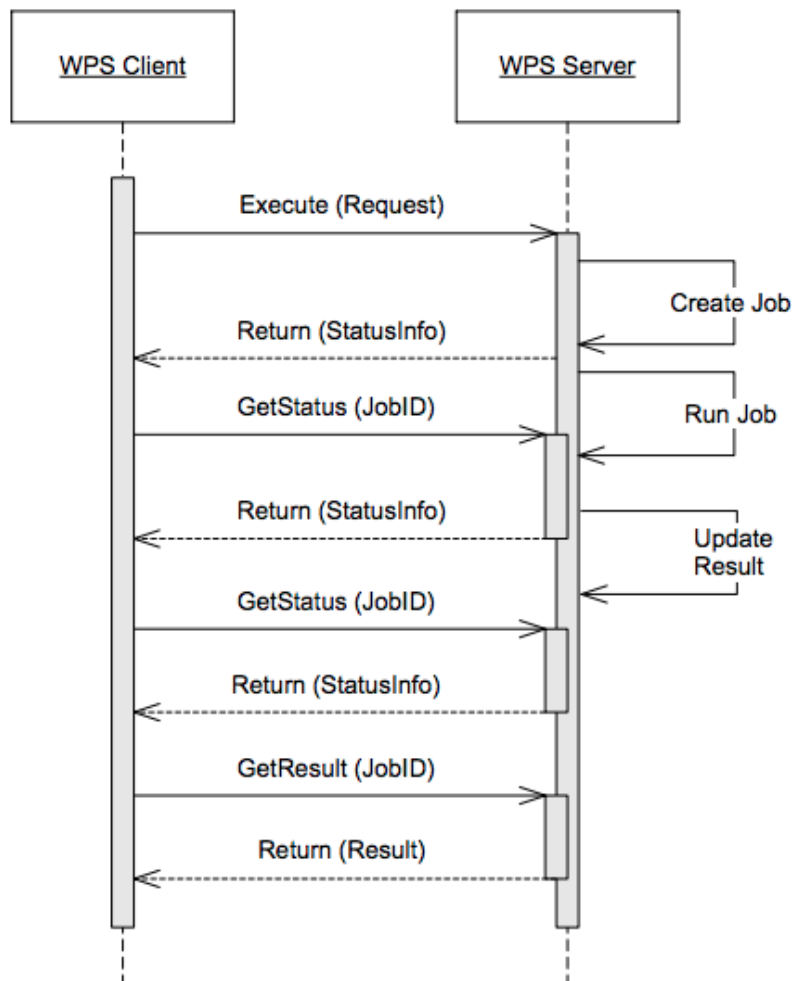
Figure 7. Complex interaction with an asynchronous ADQS WPS 2.0 based process

Translated into the ADQS terminology, the usage workflow has following typical steps:

1. A client should first issue a GetCapabilities request to the ADQS server to obtain an up-to date listing of available quality assessment processes. There could be many processes deployed and they could coexist as different process versions.

2. Then, the client may issue a DescribeProcess request to find out more details about particular processes offered, including the supported data formats.

3. To run a quality assessment, a client will issue an Execute request providing required input data. These are either embedded data sets and data quality attributes or external links and service endpoints, where such data could be obtained. If the process was run asynchronously, the Execute request will provide a JobID, which is an unique identifier for the running job.

4. In the case of a long-running assessment process invoked by specifying asynchronous message exchange pattern, the service consumer might periodically issue GetStatus with jobID as input parameter.

5. In the case of a long-running assessment process, the service consumer might issue a GetResult request with jobID as an input parameter in order to receive the data quality assessment report.

The structure of a service contract document (API) is depicted in the following UML diagram (from the WPS 2.0 specification):
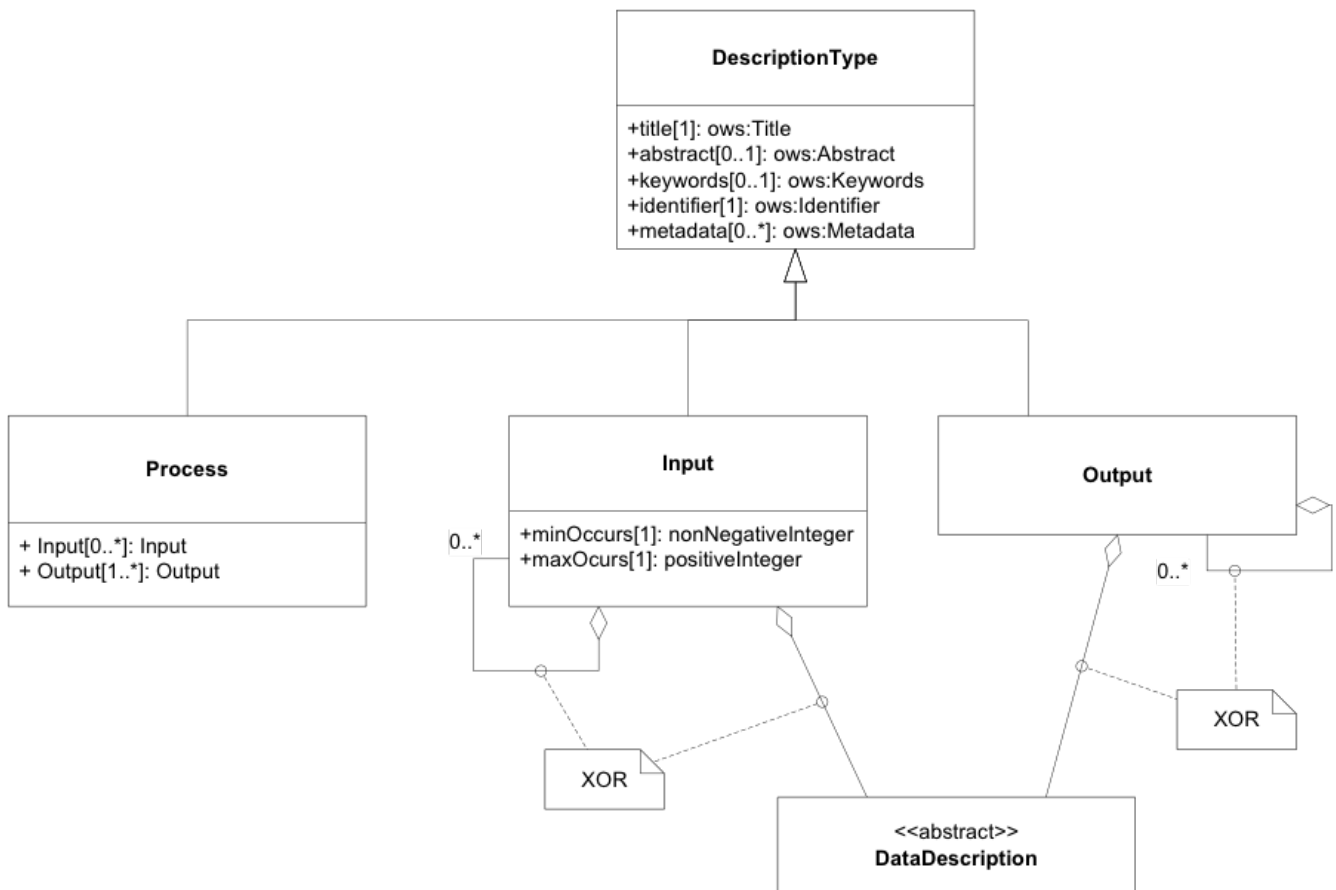
Figure 8. Process UML class diagram

To demonstrate the basic DescribeProcess syntax for ADQS, the following XML listing provides process description (response to DescribeProcess) for a minimalistic positional accuracy assessment service to assess the quality of data provided by some hypothetical flight track service. The input parameter is a flight trajectory encoded in the GML. This blueprint could easily be modified to use more appropriate AIXM 5.1 format:

DescribeProcess.xml

```xml
<wps:ProcessOffering jobControlOptions="sync-execute async-execute dismiss"
    outputTransmission="value reference">
  <ows:Title>Positional Accuracy Assessment for Tracker Service</ows:Title>
  <ows:Abstract>Simple Features geometry input in GML</ows:Abstract>
  <ows:Identifier>INPUT_TRAJECTORY</ows:Identifier>
    <ows:Metadata xlink:role=
"http://www.opengis.net/spec/wps/2.0/def/process/description/documentation"
xlink:href="http://some.host/profileregistry/aviation/adqs.html#trajectory"/>
  <wps:Input minOccurs="0"  maxOccurs="1">
      <ows:Title>Quality Assessment Input GML</ows:Title>
      <ows:Abstract>GML encoded aviation data sets</ows:Abstract>
      <ows:Identifier>INPUT</ows:Identifier>
      <ows:Metadata xlink:role=
"http://www.opengis.net/spec/wps/2.0/def/process/description/documentation"
xlink:href="http://some.host/profileregistry/aviation/adqs.html#input_gml"/>
      <wps:ComplexData>
          <wps:Format mimeType="text/xml" encoding="UTF-8" schema=
"http://schemas.opengis.net/gml/3.2.1/gml.xsd" default="true"/>
      </wps:ComplexData>
  </wps:Input>
  <wps:Output>
      <ows:Title>Quality Assessment Report</ows:Title>
      <ows:Abstract>A report describing the quality of input data.</ows:Abstract>
      <ows:Identifier>ASSESSMENT_REPORT</ows:Identifier>
      <ows:Metadata xlink:role=
"http://www.opengis.net/spec/wps/2.0/def/process/description/documentation"
xlink:href="http://some.host/profileregistry/aviation/adqs.html#assessment_report"/>
      <wps:ComplexData>
          <wps:Format mimeType="text/xml" encoding="UTF-8" schema=
"http://www.qualityml.org/schemas/qualityml/1.0/qualityml.xsd" default="true"/>
      </wps:ComplexData>
  </wps:Output>
</wps:ProcessOffering>
```

A more advanced example would also contain a second input parameter for a flight number and a bounding box with geographic coordinates for a rectangular area of interest.

The diagram below repeats the WPS-based application's component diagram provided above using the ADQS terminology.
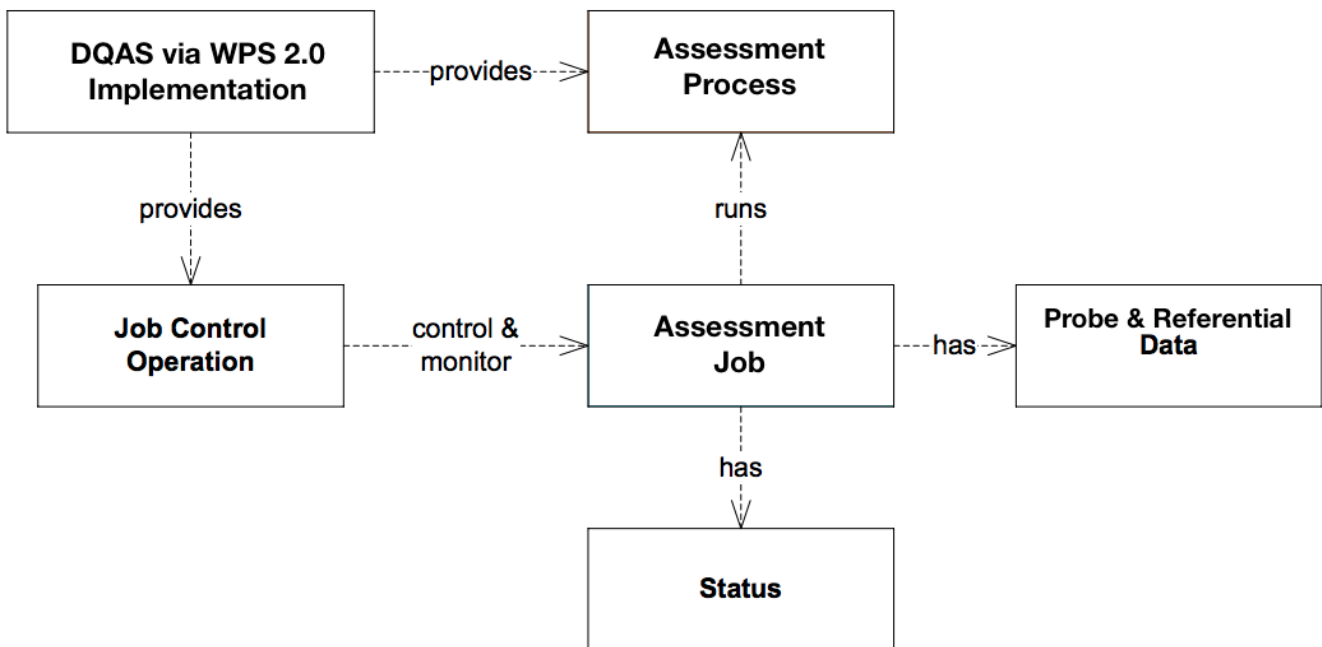
Figure 9. ADQS assessment service as WPS compatible process

A WPS 2.0 API implementation acts as a reverse proxy towards service consumers encapsulating backend service implementation details. Jobs are the WPS units of works, which are assigned to a particular assessment process instance. The WPS manages the assessment process execution lifecycle and in case of asynchronous data quality assessment evaluation offers an interface to consumers, which enables them to control and monitor running jobs.

8.4. ADQS input and output options

This ER makes the distinction between two major types of data quality processes:

- Quality estimation is calculation of quality metrics based on given quality parameters.
- Quality assessment validates the data quality claims that were made by a service provider.

An example of the quality claims are metadata documents hosted in an aviation service registry. The data quality statements are expected to be embedded into an SDCM 2.0 QoS document, as it was proposed in FA002. Two types of output document are applicable for the data quality assessment service report, one for quality estimation and another for quality assessment.

Due to the universal applicability of QualityML, the assessment output document could be created by extending the input document and adding additional elements. The input document shall be extended in the way that for every quantitative or qualitative element provided in the input, an additional element with calculated metric values is set in the output. These metric calculations could be placed in original QualityML elements or they could be integrated into the output document separately. Further, the assessment output shall provide additional data elements for encoding of assessment relevant metadata. The information regarding the input data set (which is the object of the quality assessment) shall be integrated into the output document. The following list depicts the basic structure of the assessment report (output) document:

- **Input**
  - QualityML document(s)

- ◦ Data set(s)

  - ◦ Bounding box

  - ◦ Quality elements to assess/calculate

- **Output**

  - ◦ QualityML document(s)

  - ◦ Assessment report

    - ▪ Assessed/calculated quality elements

    - ▪ Bounding box

    - ▪ Input data set info

      - ▪ Size

      - ▪ Data types

    - ▪ Assessment execution times

In the case of quality estimation, the input document shall contain all types of quality attributes that a service consumer might be interested to get the report about. The attributes shall be encoded using QualityML statements. The metric values shall remain empty. The resulting output document structure contains the QualityML elements from the input document with populated values for metric attributes. If the metrics are not specified in the input document, the assessment service might calculate values for all applicable metrics. In this case a metadata description shall describe the approach and be attached on the output document.

The code example below shows the minimalistic data quality document borrowed from the Quality ML web page. The metric for quantitative attribute accuracy used to make the quality statement is encoded using an URI (distributions/normal). This URI could be used during the service invocation to communicate the type of assessment required to be performed. A decision on which elements inside of the data set shall be considered for metric calculations has to be made based on the type of input data. For example, positional accuracy is related to geospatial information and in the case of AIXM 5.1 data quality assessment, for every single entity specified in the AIXM 5.1 the ADQS shall be filled up with information about elements that were used to encode positions.

```xml
<gmd:MD_Metadata>
   <md:dataQualityInfo>
      <gmd:MD_DataQuality>
         <gmd:report>
            <gmd:DQ_QuantitativeAttributeAccuracy>
               <gmd:result>
                  <gmd:DQ_QuantitativeResult>
                     <gmd:valueType>
                        <gco:RecordType xlink:href="http://www.uncertml.org/distributions/normal">
                              Value  for  vertical  DEM  accuracy
                        </gco:RecordType>
                     </gmd:valueType>
                     <gmd:valueUnit  xlink:href="urn:ogc:def:uom:OGC:1.0:metre"/>
                     <gmd:value>
                        <gco:Record>
                           <un:NormalDistribution>
                              <un:mean>1.2</un:mean>
                              <un:variance>3.0625</un:variance>
                           </un:NormalDistribution>
                        </gco:Record>
                     </gmd:value>
                  </gmd:DQ_QuantitativeResult>
               </gmd:result>
            </gmd:DQ_QuantitativeAttributeAccuracy>
         </gmd:report>
      </gmd:MD_DataQuality>
   </md:dataQualityInfo>
</gmd:MD_Metadata>
```

Figure 10. ADQS assessment output WPS compatible process

The QualityML statements embedded into input documents shall be used by service execution logic to identify data quality assessment activities required to be performed for an individual assessment process:

- If the metrics are not provided in the input documents at all, all assessments available in the service will be performed.

- If the metrics are available but no values are provided for them, the ADQS shall use it as a template and calculate values required for quality parameters.

- If the metrics and values are provided in the input document, the ADQS shall perform all applicable calculations and compare the results with available metric results (assessment).

The code example below demonstrates a simple QualityML output document containing a single metric. The highlighted area in the code points out to the two characteristic attributes of a normal distribution metric, which, if they are populated in the input document, represents the data quality claim made by service provider (the claim which shall be assessed by the ADQL), or if the values were omitted, that shall be the indicator for ADQS service execution logic, that those metric values shall be calculated and presented as part of data quality assessment.

```
<gmd:MD_Metadata>
   <md:dataQualityInfo>
      <gmd:MD_DataQuality>
         <gmd:report>
            <gmd:DQ_QuantitativeAttributeAccuracy>
               <gmd:result>
                  <gmd:DQ_QuantitativeResult>
                     <gmd:valueType>
                        <gco:RecordType xlink:href="http://www.uncertml.org/distributions/normal">
                              Value  for  vertical  DEM  accuracy
                        </gco:RecordType>
                     </gmd:valueType>
                     <gmd:valueUnit  xlink:href="urn:ogc:def:uom:OGC:1.0:metre"/>
                     <gmd:value>
                        <gco:Record>
                           <un:NormalDistribution>
                              <un:mean>1.2</un:mean>
                              <un:variance>3.0625</un:variance>
                           </un:NormalDistribution>
                        </gco:Record>
                     </gmd:value>
                  </gmd:DQ_QuantitativeResult>
               </gmd:result>
            </gmd:DQ_QuantitativeAttributeAccuracy>
         </gmd:report>
      </gmd:MD_DataQuality>
   </md:dataQualityInfo>
</gmd:MD_Metadata>
```

Figure 11. ADQS assessment service output

The quality assessment strategy (quality parameters) could also be specified by service consumers using control attributes in service requests. A consumer specifies what to assess using control attributes or lets the service perform all sorts of assessments/investigations it is able to calculate. Therefore, the following are options for process execution control:

- Input document structure and content

- Input assessment execution control attributes

- Assessment service implement into several WPS processes with different complexity grades.

Another approach for assessment execution control would be based on having multiple service endpoints with support for different subsets of metrics. The service implementations are therefore provided in several versions of different complexity. Each one is equipped with a subset of quality metrics so the service provider can decide which one to apply. This approach seems to be particularly useful considering the overall number of possible metrics and computational effort to calculate them all. Full-quality assessment might not always be required. Having said that, a service might offer the following three assessment process types:

- The light

- The standard

- The full

The light profile version would contain only the common metrics, the standard one would be extended with additional metrics, while the full version would host all metrics applicable for every applicable data quality assessment case. The decision of which quality parameters/metrics shall belong to which of types/categories should be made with input from domain experts.

The input/output document formats and the list of supported quality metrics shall be modeled

using the WPS 2.0 DescribeProcess document and the concept of service profiles.

8.5. WPS 2.0 profiles

The processing service specification includes the concept of an Application Profile. It is a sort of process implementation blueprint. Processes that are advertised as compliant with the same Application Profile are intended to provide identical functionality. An Application Profile is essentially the same as the ProcessDescription document that is returned in response to a DescribeProcess request. The concept of profiles shall be utilized for the design of an aviation data quality assessment profile for WPS 2.0 as it was elaborated in the next chapter.

Chapter 9. ADQS WPS Profile

The WPS specification includes the concept of an Application Profile [http://docs.opengeospatial.org/is/14-065/14-065.html#32]. Processes that are advertised as compliant with the same Application Profile are intended to provide identical functionality. An Application Profile is essentially the same as the ProcessDescription document that is returned in response to a DescribeProcess request. This concept shall be supported considering the design of an aviation data quality assessment profile for WPS 2.0.

What is the most relevant input for a WPS application profile implementation? The aviation data quality assessment service fulfills several tasks: it shall estimate the quality of input data for an input data set and quality attributes, or it shall assess the claims made by service provider regarding the quality of data provided to service consumers. The following input data is required for quality assessment:

- Aviation data sets on which quality shall be assessed

- Data quality statements (if available)

- Referential data sets (optional)
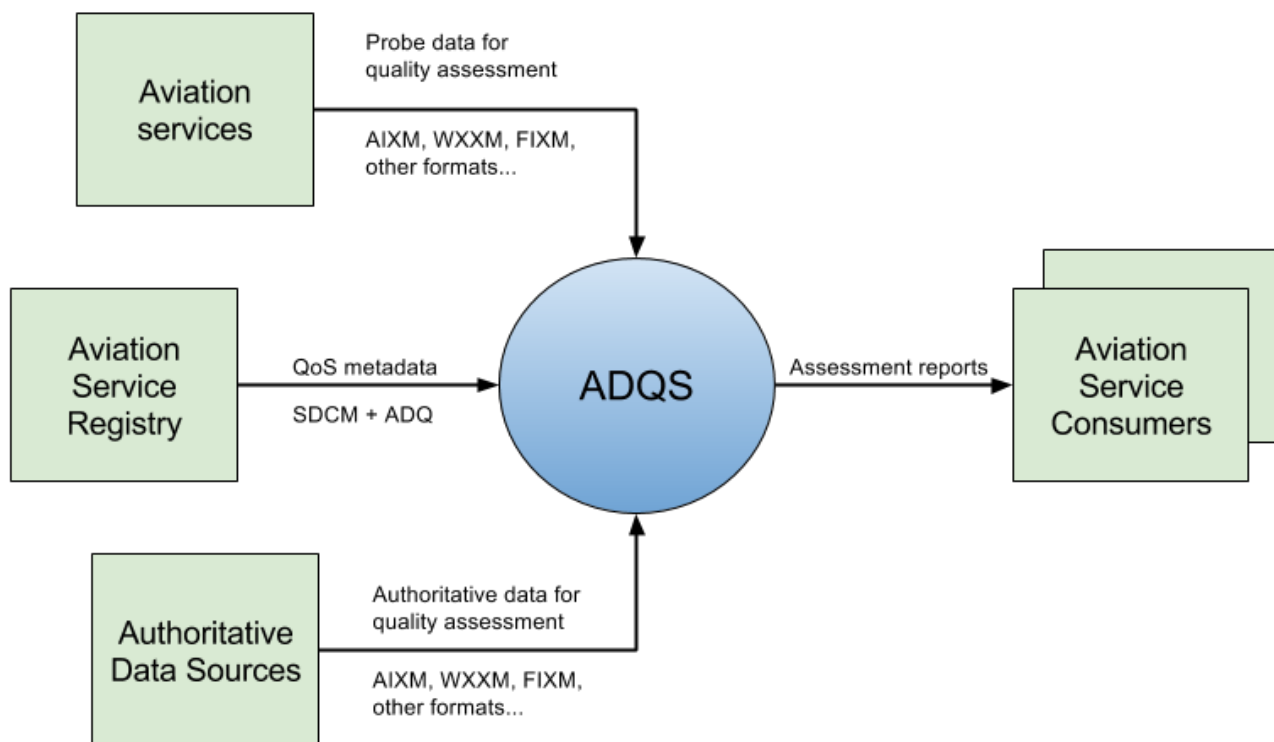
Figure 12. ADQS data input

The data for assessment could be any kind of data. However, the most frequently used data in the field of aviation and certainly the most important are the AIXM[11], FIXM[12], and WXXM[13] information exchange formats, which describe aeronautical infrastructure, flight flow and weather domains respectively. Further, the aviation domain utilizes different raster based formats such as those for encoding of terrain and obstacle data.

Data quality elements are well defined statements derived from QualityML[7] and ISO

19157:2013[15] and extended for certain aviation specific concepts in the FA002[5] as an effort to establish abstract data quality model suitable for aviation and other domains. Following diagram borrowed from the FA002 engineering report depicts structure used to encode the data quality attributes:
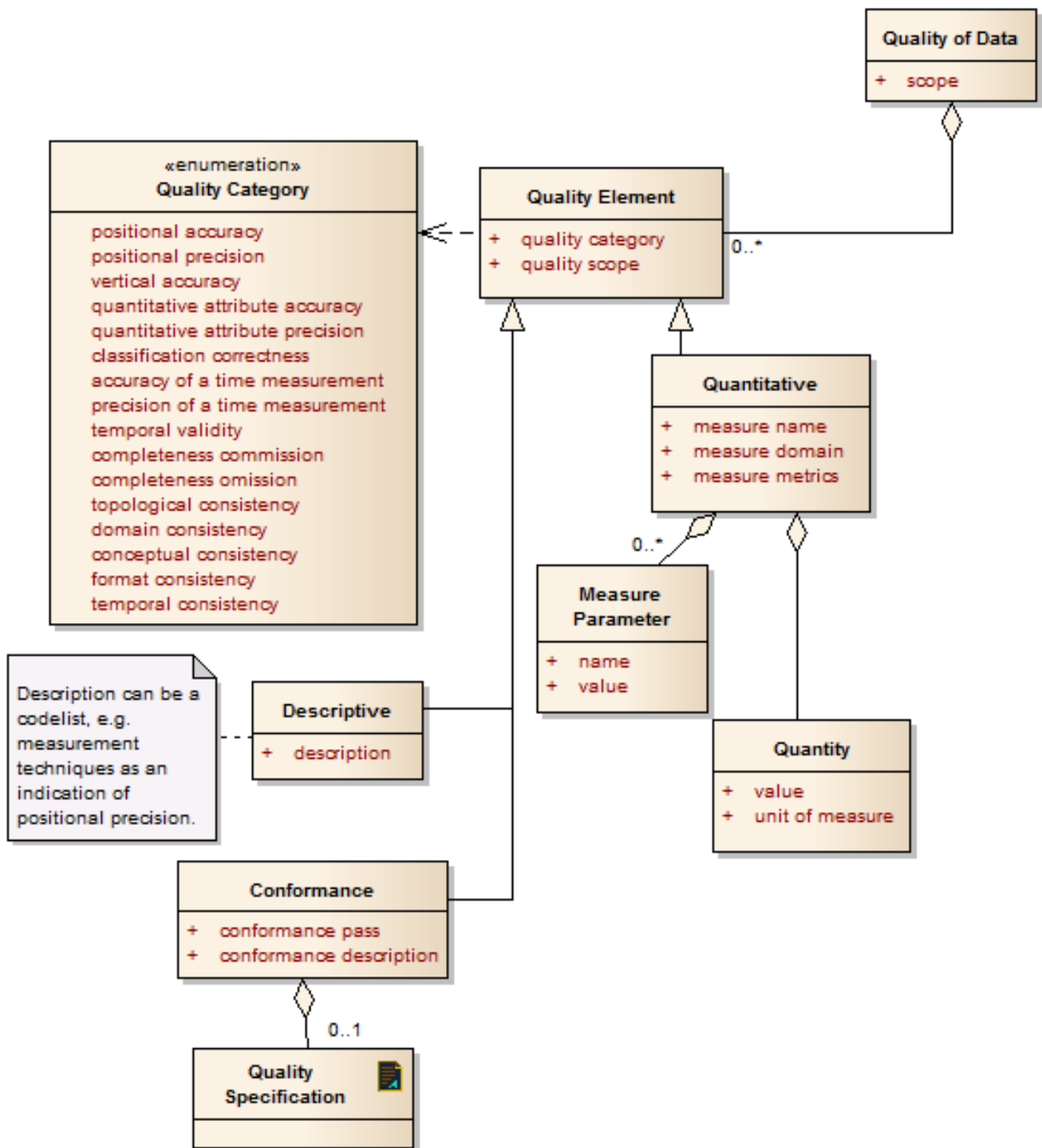
Figure 13. Data quality element

This is the format to express either the claims/expectations put on the quality of service data (input) or to encode the assessment reports (output).

Referential data (also known as reference data) is authoritative data, known to be of high quality, that is used in certain quality metrics to estimate and report the deviation between the values considered to be accurate and those provided by a service.

The input parameters for the ADQS shall obviously include all three types of information. The WPS uses concept of service descriptions to specify the input data. The operation for process description retrieval is called DescribeProcess and its return document is named ProcessOffering.

Based on the hierarchical profiling approach defined in the WPS 2.0 standard, different abstraction levels for profiles of data quality assessment processes have been defined:

- ADQS process concept
- Generic ADQS process
- ADQS process implementation profile

They help to gradually develop the proper service interface based on the WPS paradigm.

9.1. ADQS process concept

A process concept describes the functionality of a process or process group on a very high level. The WPS 2.0 standard recommends that this could be done using a Hypertext Markup Language (HTML) page. The high-level concept for data quality assessment is recommended to be based on the results of this engineering report and made publicly available as an HTML page hosted for example by an aviation service registry.

9.2. Generic ADQS process

A generic process profile shall consist of an XML document similar to a process description but without data formats specified for inputs and outputs, and an HTML page with descriptions for the process itself and for the inputs and outputs. The formal generic data quality assessment process profile description does not exist for ADQS, so the recommendation is to create and publish the description in accordance with this engineering report and the requirements from the WPS 2.0 specification.

The generic process XML description:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<wps:GenericProcess
    xmlns:ows="http://www.opengis.net/ows/2.0"
    xmlns:wps="http://www.opengis.net/wps/2.0"
    xmlns:xlink="http://www.w3.org/1999/xlink"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://www.opengis.net/wps/2.0
http://schemas.opengis.net/wps/2.0/wps.xsd">

    <ows:Title>ADQS</ows:Title>
    <ows:Identifier>http://some.host/profileregistry/aviation/adqs</ows:Identifier>
    <ows:Metadata xlink:role="http://www.opengis.net/spec/wps/2.0/def/process-
profile/concept" xlink:href="http://some.domain/wps-profileregistry/aviation/wps.html
"/>
    <ows:Metadata xlink:role=
"http://www.opengis.net/spec/wps/2.0/def/process/description/documentation"
```

```
xlink:href="http://some.domain/wps-profileregistry/aviation/adqs.html"/>

    <wps:Input>
    <ows:Identifier>Assessment BBox</ows:Identifier>
    <ows:Title>Bounding Box for spatial data quality assessment</ows:Title>
    <ows:Abstract>Bounding box to be used for quality assessment in a
polygon</ows:Abstract>
  </wps:Input>
    <wps:Input>
    <ows:Title>Quality Assessment Input AIXM</ows:Title>
        <ows:Identifier>INPUT-AIXM</ows:Identifier>
        <ows:Metadata xlink:role=
"http://www.opengis.net/spec/wps/2.0/def/process/description/documentation"
xlink:href="http://some.host/profileregistry/aviation/adqs.html#input_aixm"/>
    </wps:Input>
    <wps:Input>
        <ows:Title>Quality Assessment Input FIXM</ows:Title>
        <ows:Identifier>INPUT-FIXM</ows:Identifier>
        <ows:Metadata xlink:role=
"http://www.opengis.net/spec/wps/2.0/def/process/description/documentation"
xlink:href="http://some.host/profileregistry/aviation/adqs.html#input_fixm"/>
    </wps:Input>
    <wps:Input>
        <ows:Title>Quality Assessment Input WXXM</ows:Title>
        <ows:Identifier>INPUT-WXXM</ows:Identifier>
        <ows:Metadata xlink:role=
"http://www.opengis.net/spec/wps/2.0/def/process/description/documentation"
xlink:href="http://some.host/profileregistry/aviation/adqs.html#input_wxxm"/>
    </wps:Input>
    <wps:Input>
        <ows:Title>Quality Assessment Input GML</ows:Title>
        <ows:Identifier>INPUT-GML</ows:Identifier>
        <ows:Metadata xlink:role=
"http://www.opengis.net/spec/wps/2.0/def/process/description/documentation"
xlink:href="http://some.host/profileregistry/aviation/adqs.html#input_gml"/>
    </wps:Input>
    <wps:Output>
        <ows:Title>Data quality Assessment Output</ows:Title>
        <ows:Identifier>ASSESSMENT_REPORT</ows:Identifier>
        <ows:Metadata xlink:role=
"http://www.opengis.net/spec/wps/2.0/def/process/description/documentation"
xlink:href="http://some.domain/wps-
profileregistry/aviation/adqs.html#assessment_report"/>
    </wps:Output>
    <wps:Output>
        <ows:Title>Data Quality Description Output</ows:Title>
        <ows:Identifier>QUALITY_REPORT</ows:Identifier>
        <ows:Metadata xlink:role=
"http://www.opengis.net/spec/wps/2.0/def/process/description/documentation"
xlink:href="http://some.domain/wps-
profileregistry/aviation/adqs.html#quality_statements"/>
```

```
        </wps:Output>
</wps:GenericProcess>
```

9.3. ADQS process implementation profile

An implementation profile consists of a WPS process description and an HTML page with details about the implementation, as well as inputs and outputs of the process. The HTML page for the ADQS process profile does not exist yet and the recommendation would be to create and publish the process description in accordance with this engineering report and the WPS 2.0 specification.

The process description looks like the following:

```
<?xml version="1.0" encoding="UTF-8"?>
<wps:ProcessOfferings xmlns:wps="http://www.opengis.net/wps/2.0" xmlns:xsi=
"http://www.w3.org/2001/XMLSchema-instance" xmlns:ows="http://www.opengis.net/ows/2.0"
xmlns:xlin="http://www.w3.org/1999/xlink" xsi:schemaLocation=
"http://www.opengis.net/wps/2.0 http://schemas.opengis.net/wps/2.0/wps.xsd">
  <wps:ProcessOffering processVersion="1.0.0" jobControlOptions="sync-execute async-
execute" outputTransmission="value reference">
    <wps:Process xmlns:wps="http://www.opengis.net/wps/2.0.0" xmlns:ows=
"http://www.opengis.net/ows/2.0"
    xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:xsi=
"http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://www.opengis.net/wps/2.0.0 ../../wps.xsd">

    <ows:Title>ADQS</ows:Title>
    <ows:Abstract>Avation data quality assessment service</ows:Abstract>
    <ows:Identifier>http://some.host/profileregistry/aviation/adqs</ows:Identifier>
        <ows:Metadata xlink:role="http://www.opengis.net/spec/wps/2.0/def/process-
profile/concept" xlink:href="http://some.domain/wps-profileregistry/aviation/wps.html
"/>
    <ows:Metadata xlink:role=
"http://www.opengis.net/spec/wps/2.0/def/process/description/documentation"
xlink:href="http://some.host/profileregistry/aviation/adqs.html"/>

  <wps:Input>
    <ows:Identifier>Assessment BBox</ows:Identifier>
    <ows:Title>Bounding Box for spatial data quality assessment</ows:Title>
    <ows:Abstract>Bounding box to be used for quality assessment in a
polygon</ows:Abstract>
    <Bows:BoundingBoxData>
      <Default>
        <CRS>urn:ogc:def:crs:EPSG:6.6:4326</CRS>
      </Default>
      <Supported>
        <CRSsType>
          <CRS>urn:ogc:def:crs:EPSG:6.6:4326</CRS>
          <CRS>urn:ogc:def:crs:EPSG:6.6:4979</CRS>
        </CRSsType>
      </Supported>
```

```xml
        </ows:BoundingBoxData>
    </wps:Input>


    <wps:Input minOccurs="0" maxOccurs="1">
        <ows:Title>Quality Assessment Input AIXM</ows:Title>
        <ows:Abstract>Aeronautical infrastructure, AIXM 5.1 data sets</ows:Abstract>
        <ows:Identifier>INPUT-AIXM</ows:Identifier>
        <ows:Metadata xlink:role=
"http://www.opengis.net/spec/wps/2.0/def/process/description/documentation"
xlink:href="http://some.host/profileregistry/aviation/adqs.html#input_aixm"/>
        <wps:ComplexData>
            <wps:Format mimeType="text/xml" encoding="UTF-8" schema=
"http://www.aixm.aero/schema/5.1/AIXM_Features.xsd" default="true"/>
        </wps:ComplexData>
    </wps:Input>


    <wps:Input minOccurs="0" maxOccurs="1">
        <ows:Title>Quality Assessment Input FIXM</ows:Title>
        <ows:Abstract>Air traffic data flow, FIXM 3.0.1 data sets</ows:Abstract>
        <ows:Identifier>INPUT-FIXM</ows:Identifier>
        <ows:Metadata xlink:role=
"http://www.opengis.net/spec/wps/2.0/def/process/description/documentation"
xlink:href="http://some.host/profileregistry/aviation/adqs.html#input_fixm"/>
        <wps:ComplexData>
            <wps:Format mimeType="text/xml" encoding="UTF-8" schema=
"http://www.fixm.aero/schema/3.0.1/FIXM_Features.xsd" default="true"/>
        </wps:ComplexData>
    </wps:Input>


    <wps:Input minOccurs="0"  maxOccurs="1">
        <ows:Title>Quality Assessment Input WXXM</ows:Title>
        <ows:Abstract>Weather data, WXXM 2.0 data sets</ows:Abstract>
        <ows:Identifier>INPUT-WXXM</ows:Identifier>
        <ows:Metadata xlink:role=
"http://www.opengis.net/spec/wps/2.0/def/process/description/documentation"
xlink:href="http://some.host/profileregistry/aviation/adqs.html#input_wxxm"/>
        <wps:ComplexData>
            <wps:Format mimeType="text/xml" encoding="UTF-8" schema=
"http://www.wxxm.aero/schema/2/WXXM_Features.xsd" default="true"/>
        </wps:ComplexData>
    </wps:Input>


    <wps:Input minOccurs="0"  maxOccurs="1">
        <ows:Title>Quality Assessment Input GML</ows:Title>
        <ows:Abstract>GML encoded aviation data sets</ows:Abstract>
        <ows:Identifier>INPUT-GML</ows:Identifier>
        <ows:Metadata xlink:role=
"http://www.opengis.net/spec/wps/2.0/def/process/description/documentation"
xlink:href="http://some.host/profileregistry/aviation/adqs.html#input_gml"/>
        <wps:ComplexData>
            <wps:Format mimeType="text/xml" encoding="UTF-8" schema=
```

```
"http://schemas.opengis.net/gml/3.2.1/gml.xsd" default="true"/>
        </wps:ComplexData>
    </wps:Input>

    <wps:Input>
        <ows:Title>Quality Statements</ows:Title>
        <ows:Abstract>Input describing the quality of data in extended
SDCM.</ows:Abstract>
        <ows:Identifier>QUALITY_REPORT</ows:Identifier>
        <ows:Metadata xlink:role=
"http://www.opengis.net/spec/wps/2.0/def/process/description/documentation"
xlink:href="http://some.host/profileregistry/aviation/adqs.html#quality_statements"/>
        <wps:ComplexData>
            <wps:Format mimeType="text/xml" encoding="UTF-8" schema=
"http://www.qualityml.org/schemas/qualityml/1.0/qualityml.xsd" default="true"/>
        </wps:ComplexData>
    </wps:Input>

    <wps:Output>
        <ows:Title>Quality Assessment Report</ows:Title>
        <ows:Abstract>A report describing the quality of input data.</ows:Abstract>
        <ows:Identifier>ASSESSMENT_REPORT</ows:Identifier>
        <ows:Metadata xlink:role=
"http://www.opengis.net/spec/wps/2.0/def/process/description/documentation"
xlink:href="http://some.host/profileregistry/aviation/adqs.html#assessment_report"/>
        <wps:ComplexData>
            <wps:Format mimeType="text/xml" encoding="UTF-8" schema=
"http://www.qualityml.org/schemas/qualityml/1.0/qualityml.xsd" default="true"/>
        </wps:ComplexData>
    </wps:Output>
  </wps:Process>
 </wps:ProcessOffering>
</wps:ProcessOfferings>
```

The implementation profile extends the generic assessment profile and additionally defines the input and output formats. As a blueprint, the process description for a buffer WPS process from the WPS 2.0 examples folder was used.

To encode the atomic statements about the data quality the FA002 specifies the taxonomy called QualityML. A Quality element in QualityML is a combination of following taxonomy elements:

1. Quality class

2. Quality indicator

3. Quality domain

4. Quality metric

Quality metrics further include:

- Metrics name

- Metrics description

- Metrics parameters

- Values and units of measure

The combination of a quality domain and quality metrics are commonly known as quality measures. These concepts are taken from ISO 19139 and ISO 19115-3 standards.

Summarized, the data quality model is defined based on the QualityML, as well as ISO recommendations and extensions developed in Testbed-13 FA002[5] for the aviation domain. Here is an example of how to encode the WPS output with quality metadata in ISO 19139 metadata documents using the QualityML framework and the qualityML URIs. This format is suitable for initial data quality evaluation and not for assessment of claimed data quality by a service provider.

Here, some examples on how to encode quality metadata in ISO 19139 metadata documents using the qualityML framework and the qualityML URIs:

Example using UncertML in an ISO metadata record

```xml
<wps:Result
    xmlns:ows="http://www.opengis.net/ows/2.0" xmlns:wps=
"http://www.opengis.net/wps/2.0.0"
    xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:xml=
"http://www.w3.org/XML/1998/namespace"
    xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:xsi=
"http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://www.opengis.net/wps/2.0.0 ../wpsGetResult.xsd ">
    <wps:JobID>834593049fslis45o98d7f</wps:JobID>
    <wps:Output id="001">
        <wps:Reference xlink:href="http://foo/001"/>
        <gmd:MD_Metadata>
            <md:dataQualityInfo>
                <gmd:MD_DataQuality>
                    <gmd:report>
                        <gmd:DQ_QuantitativeAttributeAccuracy>
                            <gmd:result>
                                <gmd:DQ_QuantitativeResult>
                                    <gmd:valueType>
                                        <gco:RecordType xlink:href=
"http://www.uncertml.org/distributions/normal">
                                            Value  for  vertical DEM  accuracy
                                        </gco:RecordType>
                                    </gmd:valueType>
                                    <gmd:valueUnit xlink:href="urn:ogc:def:uom:OGC:1.0:metre
"/>
                                    <gmd:value>
                                        <gco:Record>
                                            <un:NormalDistribution>
                                                <un:mean>1.2</un:mean>
                                                <un:variance>3.0625</un:variance>
                                            </un:NormalDistribution>
                                        </gco:Record>
                                    </gmd:value>
                                </gmd:DQ_QuantitativeResult>
                            </gmd:result>
                        </gmd:DQ_QuantitativeAttributeAccuracy>
                    </gmd:report>
                </gmd:MD_DataQuality>
            </md:dataQualityInfo>
        </gmd:MD_Metadata>
    </wps:Output>
</wps:Result>
```

In a WPS profile the input parameters are optionally specified as complex types defined in external, imported schemas. For QualityML this option would mean that the corresponding XML schema shall be referenced in the profile document.

If we take a look at the WPS result above we will see that a lot of XML tags and quite complex structure is used to encode following:

- Vertical accuracy (error) of a Digital Elevation Model (DEM) was modeled using normal distribution, with "mean = 1.2 m" and "variance = 3.065 m".

Obviously, a lot of XML text was used to encode several simple attributes: the name of measurement, the method, the two values and the unit of measurement. Some information such as the measure (normal distribution) was redundantly encoded twice. The following code example, borrowed from QualityML, presents even more bloated data quality assessment in XML used to encode missing ratio of an element per 100 items (completeness omission)

```xml
<gmd:MD_Metadata>
    <md:dataQualityInfo>
        <gmd:MD_DataQuality>
            <gmd:report>
                <gmd:DQ_CompletenessOmission>
                    <gmd:nameOfMeasure>
                        <gco:CharacterString>
                            Missing
                        </gco:CharacterString>
                    </gmd:nameOfMeasure>
                    <gmd:measureIdentification>
                        <gmd:MD_Identifier>
                            <gmd:code>
                                <gco:CharacterString>
                                    http://www.qualityml.org/1.0/measure/Missing
                                </gco:CharacterString>
                            </gmd:code>
                        </gmd:MD_Identifier>
                    </gmd:measureIdentification>
                    <gmd:result>
                        <gmd:DQ_QuantitativeResult>
                            <gmd:valueType>
                                <gco:RecordType xlink:href=
"http://www.qualityml.org/1.0/metrics/items">
                                    Rate of missing items
                                </gco:RecordType>
                            </gmd:valueType>
                            <gmd:valueUnit/>
                            <gmd:value>
                                <gco:Record>
                                    <qml:Items>
                                        <qml:rate max="100">3</qml:rate>
                                    </qml:Items>
                                </gco:Record>
                            </gmd:value>
                        </gmd:DQ_QuantitativeResult>
                    </gmd:result>
                </gmd:DQ_CompletenessOmission>
            </gmd:report>
        </gmd:MD_DataQuality>
    </md:dataQualityInfo>
</gmd:MD_Metadata>
```

Considering the two examples given above, alternatively, the QualityML elements from the corresponding XML schema [http://www.qualityml.org/schemas/qualityml/1.0/qualityml.xsd] might be encoded using the WPS 2.0 taxonomy elements. The motivation is to try to utilize the same taxonomy in a more compact format.

For quality measures we need at least:

- Quality class

- Quality indicator

- Quality measure

- Domain

- Metrics

The following code provides an example of how to use WPS instead of dedicated QualityML XML schema. The concept of 'allowed values' is used to incorporate QualityML taxonomy in the WPS service endpoint description.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<wps:ProcessOfferings
    xmlns:wps="http://www.opengis.net/wps/2.0"
    xmlns:ows="http://www.opengis.net/ows/2.0"
    xmlns:gmd="http://www.isotc211.org/2005/gmd"
    xmlns:xlink="http://www.w3.org/1999/xlink"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://www.opengis.net/wps/2.0 ../wps.xsd">

    <wps:ProcessOffering jobControlOptions="sync-execute async-execute dismiss"
outputTransmission="value reference">
        <wps:Process>
            <ows:Title>Aeronautical Data Quality Assessment Service</ows:Title>
            <ows:Abstract>Asses the quality of aeronautical data</ows:Abstract>
            <ows:Identifier>http://my.wps.server/processes/proximity/Planar-
Buffer</ows:Identifier>
            <wps:Input minOccurs="0">
                <ows:Title>QoS Statements</ows:Title>
                <ows:Abstract>Data quality statement</ows:Abstract>
                <ows:Identifier></ows:Identifier>
                <ows:Metadata xlink:role=
"http://www.opengis.net/spec/wps/2.0/def/process/description/documentation"
xlink:href="http://assessment.service/processes/qos/quality-attributes"/>
        <wps:Input minOccurs="0" maxOccurs="unbounded">
                <ows:Identifier>QUALITY_CLASS_INDICATOR</ows:Identifier>
                <wps:LiteralData>
                    <LiteralDataDomain default="true">
                <ows:AllowedValues>
                  <ows:Value>PositionalAccuracy<ows:Value>
                  <ows:Value>PositionalPrecision<ows:Value>
                  <ows:Value>VerticalAccuracy<ows:Value>
                  <ows:Value>QuantitativeAttributeAccuracy<ows:Value>
                  <ows:Value>QuantitativeAttributePrecision<ows:Value>
                  <ows:Value>ClassificationCorrectness<ows:Value>
                  <ows:Value>AccuracyofTimeMeasurement<ows:Value>
                  <ows:Value>TemporalValidity<ows:Value>
                  <ows:Value>CompletenessCommission<ows:Value>
```

```
                    <ows:Value>CompletenessOmission<ows:Value>
                    <ows:Value>TopologicalConsistency<ows:Value>
                    <ows:Value>DomainConsistency<ows:Value>
                    <ows:Value>ConceptualConsistency<ows:Value>
                    <ows:Value>FormatConsistency<ows:Value>
                    <ows:Value>TemporalonsistencY<ows:Value>
                </ows:AllowedValues>
                        <ows:DataType ows:reference=
"http://www.w3.org/2001/XMLSchema#string">String</ows:DataType>
                    </LiteralDataDomain>
                </wps:LiteralData>
            </wps:Input>
                <wps:Input minOccurs="0">
                <ows:Identifier>MEASURE</ows:Identifier>
                <wps:LiteralData>
                    <LiteralDataDomain default="true">
                <ows:AllowedValues>

                                    ...

                </ows:AllowedValues>
                        <ows:DataType ows:reference=
"http://www.w3.org/2001/XMLSchema#string">String</ows:DataType>
                    </LiteralDataDomain>
                </wps:LiteralData>
            </wps:Input>
                <wps:Input minOccurs="0">
                <ows:Identifier>DOMAIN</ows:Identifier>
                <wps:LiteralData>
                    <LiteralDataDomain default="true">
                <ows:AllowedValues>
                            <ows:Value>Conformance<ows:Value>
                              <ows:Value>NonConformnce<ows:Value>
                              <ows:Value>DifferentialErrorsX<ows:Value>
                              <ows:Value>DifferentialErrorsY<ows:Value>
                              <ows:Value>DifferentialErrors2D<ows:Value>
                              <ows:Value>DifferentialErrors3D<ows:Value>
                                        ...
                </ows:AllowedValues>
                        <ows:DataType ows:reference=
"http://www.w3.org/2001/XMLSchema#string">String</ows:DataType>
                    </LiteralDataDomain>
                </wps:LiteralData>
            </wps:Input>
        <wps:Input minOccurs="0">
                <ows:Identifier>METRIC</ows:Identifier>
                <wps:LiteralData>
                    <LiteralDataDomain default="true">
                <ows:AllowedValues>
                                <ows:Value>Item<ows:Value>
                <ows:Value>Mean<ows:Value>
                <ows:Value>StandardDeviation<ows:Value>
                <ows:Value>CEP<ows:Value>
```

```xml
                        <ows:Value>MAD<ows:Value>
                                        ...
                </ows:AllowedValues>
                            <ows:DataType ows:reference=
"http://www.w3.org/2001/XMLSchema#string">String</ows:DataType>
                    </LiteralDataDomain>
                </wps:LiteralData>
            </wps:Input>
        <wps:Input minOccurs="0">
                <ows:Identifier>ADQS-VALUE</ows:Identifier>
                <wps:LiteralData>
                    <LiteralDataDomain default="true">
                        <ows:DataType ows:reference=
"http://www.w3.org/2001/XMLSchema#double">Double</ows:DataType>
                    </LiteralDataDomain>
                </wps:LiteralData>
            </wps:Input>
            </wps:Input>
        <wps:Input minOccurs="1">
                <ows:Title>Payload to be assessed</ows:Title>
                <ows:Abstract>Aeronautical Features input in AIXM 5.1</ows:Abstract>
                <ows:Identifier>INPUT_DATA</ows:Identifier>
                <ows:Metadata xlink:role=
"http://www.opengis.net/spec/wps/2.0/def/process/description/documentation"
xlink:href="http://assessment.service/processes/qos/Aeronautical-
Assessment.html#input_data"/>
                <wps:ComplexData>
                    <wps:Format mimeType="text/xml" encoding="UTF-8" schema=
"http://www.aixm.aero/schema/5.1.1/AIXM_Features.xsd" default="true"/>
                </wps:ComplexData>
            </wps:Input>
            <wps:Output>
                <ows:Title>Assessment Result</ows:Title>
                <ows:Abstract>The result of data quality assessment for the give input
data set and quality indicators</ows:Abstract>
                <ows:Identifier>ASSESSMENT_RESULT</ows:Identifier>
                <ows:Metadata xlink:role=
"http://www.opengis.net/spec/wps/2.0/def/process/description/documentation"
xlink:href="http://assessment.service/processes/proximity/Planar-
Buffer.html#buffered_geometry"/>
                <wps:ComplexData>
                    <wps:Format mimeType="text/xml" encoding="UTF-8" schema=
"http://assessment.service/processes/qos/AssessmentReport.xsd" default="true"/>
                    <wps:Format mimeType="application/json" encoding="UTF-8"/>
                </wps:ComplexData>
            </wps:Output>
        </wps:Process>
    </wps:ProcessOffering>
</wps:ProcessOfferings>
```

In this listing, WPS literal data types are used to define the taxonomy of metrics and measurements methods instead of QualityML schema with semantically equivalent elements. This approach might be preferable in combination with WPS because it better utilizes the expressiveness of the WPS and provides better readable service contract document.

Chapter 10. Extension mechanism

In the context of a quality assessment service, 'extension mechanism' refers to a characteristic of the design which facilitates design and development of the service, gradually extending its overall functionality. This is important because new capabilities might be required to be implemented during the assessment service life cycle. The services could also operate many versions of assessment processes in parallel. Having said that, from the extensibility point of view the fit-for-purpose service design shall have the following features:

- Support for deployment, configuration and management of quality assessment calculation metrics (provided as deployable, configurable software modules).

- Different data formats for inputs and outputs

- Variable message exchange patterns

- Deployment, configuration and management of multiple assessment process versions

- Assessment process discovery, in case many possible options are available

Obviously, a simple service interface cannot fulfill all these requirements.

From the design perspective and considering what has been described in the chapter titled Architecture Proposal, three components of ADQL service architecture are of crucial importance in ensuring that the purpose of ADQS is realized:

- WPS 2.0 compatible front-end component (API) used to coordinate interaction between service consumers and service backend business logic

- Process flow engine responsible for executing assessment processes

- Extensible assessment metric components

- Service management tool for configuration

What are the major characteristics of an extensible quality assessment service? First of all, it should be capable of supporting different concepts of quality assessment. This includes adoption of new data quality attributes with components for parsing, data collecting and data quality calculation metrics for new attributes. For service consumers, all features of an assessment service shall be exposed via a standardized application interface flexible enough to remain untouched while it fully supports changes in assessment workflows. In other words, it shall decouple the assessment process implementation from service end-users.

The following elements of design and architecture address the requirements regarding the extensibility:

The OGC WPS 2.0 interface

enables introduction of new assessment processes without interruption of service availability and changes of service contract. This is possible thanks to a flexible architecture and management functions which enable users to search for and address new assessment processes using a predefined request attribute (process name). The WPS interface also offers meta description for new assessment processes and supports rudimentary service discovery. An assessment service can be implemented either as synchronous or asynchronous processes with

standardized support for process instance creation, execution, status querying and asynchronous fetching of assessment results.

Assessment workflows

Based on a standard process model, such as the BPMN 2.0 process description taxonomy, the task is to specify and execute an assessment processes. Workflows are particularly useful for defining processes with multiple computational steps and interactions with external systems. Such a process-centric approach enables a process instance to maintain its own process state and control the execution. A standardized way to design and deploy an assessment process is important for extensibility.

Metric calculation components

They are atomic building blocks of an assessment service implementation. These components implement quality assessments based on quality metrics. They shall be designed, developed and deployed without service interruptions, which significantly contributes to the service flexibility. If they are reused in more than one assessment process, they would also contribute to the reusability and efficiency. In order to achieve these effects, quality metric components shall have standardized interfaces.

Service management tool

A front-end application for metrics configurations and deployment, as well as the assessment process design, deployment and monitoring. Such tools help to enable flexible process implementation.

Non-functional requirements and supporting architectural components:

Reusability

Metric calculation components, assessment workflows

Extensibility

WPS 2.0, workflows, service management tool

Flexibility

WPS 2.0, workflows

Versioning

WPS 2.0

10.1. WPS 2.0 as extensible application interface

WPS 2.0 offers a standardized service interface which has two characteristics of interest from the extensibility point of view. The WPS was designed using a generic interface specification, which basically allows any kind of input data.

The specification defines three types of input data:

1. Literal Data (includes single numerical values or text strings).

2. Bounding Box Data (includes geographic coordinates for a rectangular area).

3. Complex Data (includes XML, imagery, and custom or proprietary data structures).

The bounding box based input is standardized as the smallest common denominator between all potential OWS services.

The literal data, simple data types, varies depending on the service implementation and its contract. Literal data types can be used for interface definition in the DescribeProcess document. Simple data types are standardized in the sense that there is an enumerated set of allowed types (integer, double, datum, string) and values they could take.

Complex data, provided as input data, is a placeholder for any sort of structured data. This could be a document described by an XML schema (for example an AIXM or FIXM document), for example.

Combinations of all three input data types to describe the input are allowed and definitively imaginable. For example, we could have a service with simple data definitions used for quality attributes, any kind of structured document might be used for assessment input data and the bounding box could carry the spatial information in order to reduce quality assessment evaluation to one specific area.

Obviously, a WPS endpoint allows the coexistence of several different service contracts bound on different background quality assessment processes. Service instances could be configured to work with the same background process but to be described using a different service contract. They also could be deployed in several operational versions. As part of the deployment process, every process instance will get a dedicated DescribeProcess document with a well-defined input data set.

The data inputs feature from DescribeProcess allows a lot of flexibility but also introduces a certain overhead and makes the service consumption more difficult. This is due to the need to invoke metadata operations such as GetCapabilities and DescribeProcess in order to check the options and figure out how to consume the service.

Chapter 11. Examples for data quality metrics and calculations

This chapter shall provide some concrete examples about how to perform an assessment for data quality claims expressed using standard taxonomy. While some quality parameters such as completeness are assessed by searching and identifying data types delivered in a service response (in order to assess whether anything is missing), other quality metrics are much more of a mathematical nature and include calculations of statistical parameters (average values, deviations, pattern matching). The metrics can be calculated in time and spatial domains.

Metrics for precision, accuracy, and timelines applicable to describe the data quality (for both temporal and spatial dimensions) include, for example:

1. Ratio

2. Mean

3. Mean Variance

4. Median

5. Standard deviation

11.1. Digital NOTAM completeness

Digital NOTAMs are expected to be delivered in an AIXM 5.1 message envelope containing aeronautical objects they refer to. Since the NOTAMs have different originators and could be delivered with or without infrastructure entities or aeronautical objects, some of them might be considered incomplete for particular use cases (fit for purpose).

Quality assessment would follow certain predefined rules of completeness and monitor the absence of certain objects in Digital NOTAM messages.

Possible outcome: At the end, the completeness could be described as a missing ratio, for example 2%, which would signal that 2% of all NOTAM messages delivered by service have been incomplete, because they were missing certain related elements.

11.2. Timeliness assessment for commercial airline flight tracker

Metadata could, for example, specify (among others) a data quality attribute such as the mean absolute deviation: MAD =.025 sec. for timeliness.

Let's assess the timeliness for a live flight tracker (of commercial airline flights) service. Assume that the service claims (in its data quality attributes) for a chosen flight to provide position information every second with mean absolute deviation in timeliness of MAD=0.25. In other words, more than 99% of positions are going to be delivered (pushed) to service requestors within an interval between 0.75 and 1.25 seconds.

Quality assessment: In order to assess the timeliness performance, several flight tracks will be collected (in some observation time or permanently). As part of the assessment process, an algorithm component of DQAS responsible for timeliness assessment will calculate the times between provided positions (or retrieved from position items if they are part of them). Mean values will also be calculated, as well as the mean absolute deviation from that value.

Possible outcome: the service provides positions with time accuracy of one second (mean value) with dispersion of 0.23 seconds. That means that 99.3% of values were sent within a time range of 0.76 - 1.23 seconds. Final conclusion of the assessment process would be that the result was in accordance with claimed quality of data!

11.3. Positional accuracy for commercial airline flight tracker

Quality metadata could also specify (among others) a data quality attribute such as the positional accuracy for a flight tracker service. It is about the deviation of a provided flight trajectory from the flown one.

Quality assessment

In order to assess the positional accuracy performance several flight trajectories will be collected (more trajectories provide better results). For the same flights (using their unique flight number) the flown trajectories will be fetched from an authoritative track service of known (high) quality. As part of the assessment process, a metric calculation component responsible for positional accuracy (in this case we talk about curves similarity metric or curve matching) will calculate average deviation of trajectories in relation to referential trajectories. Several algorithms might be used for assessment, for example:

1. Pompeiu–Hausdorff distance, measures how far two subsets of a metric space are from each other. It is a mathematical construct to measure the "closeness" of two sets of points that are subsets of a metric space.

2. Frechet distance between curves, measures similarity between curves taking into account the location and ordering of the points along the curves.

The quality assessment might also include the common flight trajectory characteristics of commercial airline flights. For example, the trajectory cannot (it is very uncommon) have big differences in speed, altitude or heading between time intervals. Such deviations might indicate bad quality of flight tracking positional information.

Appendix A: Revision History

Table 3. Revision History

Date	Release	Editor	Primary clauses modified	Descriptions
June 15, 2017	.1	A. Balaban	all	Initial version
November 11, 2017	.7	A. Balaban	all	Review comments integrated
December 15, 2017	.9	A. Balaban	various	Preparation for publication
January 15, 2018	1.0	A. Balaban	various	Final version
February 09, 2018	1.1	A. Balaban	various	Review comments integrated

Appendix B: Bibliography

[1]

OGC 12-080r2, OGC OWS Context Conceptual Model [https://portal.opengeospatial.org/files/?artifact_id=5518]

[2]

OGC 06-121r9, OGC® Web Services Common Standard [https://portal.opengeospatial.org/files/?artifact_id=38867&version=2]

[3]

OGC 14-065, OGC® WPS 2.0 Interface Standard [http://docs.opengeospatial.org/is/14-065/14-065.html]

[4]

OGC 17-032r1, OGC Testbed 13: FA001 Abstract Data Quality ER [https://portal.opengeospatial.org/files/?artifact_id=76113&version=1]

[5]

OGC 17-032r1, OGC Testbed 13: FA002 Data Quality ER [https://portal.opengeospatial.org/files/?artifact_id=76113&version=1]

[6]

Object Management Group Business Process Model and Notation (ISO/IEC 19510:2013) [http://standards.iso.org/ittf/PubliclyAvailableStandards/c062652_ISO_IEC_19510_2013.zip]

[7]

Quality Indicators Dictionary and Markup Language (QualityML) [http://www.qualityml.org/]

[8]

Service Description Conceptual Model (SDCM) [https://www.faa.gov/nextgen/programs/swim/governance/servicesemantics/media/SDCM_v2.0/SDCM_v2.0.html#3.13]

[9]

Concept of Operations for the SWIM Common Registry (SCR) [https://www.faa.gov/nextgen/programs/swim/governance/outreach/media/SCR%20CONOPS%20FINAL%20DRAFT%2004042015.pdf]

[10]

Testbed-12 WPS Conflation Service Profile Engineering Report [http://docs.opengeospatial.org/per/16-022.html]

[11]

The Aeronautical Information Exchange Model (AIXM) [http://aixm.aero]

[12]

The Flight Information Exchange Model (FIXM) [http://fixm.aero]

[13]

The Weather Information Exchange Models and Schema (WXXM) [http://wxxm.aero]

[14]

ISO 19115-1:2014, Geographic information - Metadata - Part 1: Fundamentals

[15]

ISO 19157:2013, Geographic information - Data quality

[16]

ISO/TS 19115-3:2016, Geographic information - Metadata - Part 3: XML schema implementation for fundamental concepts

[17]

FAA System Wide Information management [https://www.faa.gov/nextgen/programs/swim/]

[18]

Publish/Subscribe Interface Standard 1.0 - Core [http://docs.opengeospatial.org/is/13-131r1/13-131r1.html]