Open Geospatial Consortium, Inc.

Date: 2010-8-18

Reference number of this document: OGC 10-132

Category: OGC® **Public Engineering Report**

Editor: Bruno Simmenauer

OGC® OWS-7 Aviation – WXXM Assessment Engineering Report

Warning

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

Document type:	OpenGIS® Engineering Report
Document subtype:	NA
Document stage:	Approved for public release
Document language:	English

Preface

This public Engineering Report (ER) is a deliverable of the Open Geospatial Consortium (OGC) Interoperability Program Open Web Service (OWS) Testbed phase 7 (OWS-7).

The document describes the results of using OGC Web Services for accessing and using WXXM data, notably within aviation scenarios involving rerouting procedures motivated by the sudden closure of airspace areas caused by the eruption of a volcano. The focus of this document will be to evaluate the ability to encode and serve associated operational data with WXXM 1.1.1.

The document will specifically insist on all the questions raised regarding this topic, along the OWS-7 project life cycle, and will provide synthetic suggestions of improvement whenever such solutions have been found.

Suggested additions, changes, and comments on this draft report are welcome and encouraged. Such suggestions may be submitted by email message or by making suggested changes in an edited copy of this document.

If you choose to submit suggested changes by editing this document, please make sure to turn on the change tracking feature beforehand in order to ease the consolidation of change proposals.

Contents

Page

Figures

Page

Tables

Page

OGC® OWS-7 Aviation – WXXM Assessment Engineering Report

1 Introduction

1.1 Scope

In OWS-7, the Aviation Thread focused on investigating and demonstrating the applicability of AIXM and WXXM along with relevant OGC specifications and web services to applications and tools that support Airline Operations Centers and Flight Dispatch applications. Such applications provide information for representing a Common Operating Picture; supporting flight planning (including General Aviation) and preparation (MET and AIM); calculating weight and balance; estimating fuel requirements; in-flight emergency response; etc.

The document describes the results of using OGC Web Services for accessing and using WXXM data, notably within aviation scenarios involving rerouting procedures motivated by the sudden closure of airspace areas caused by the eruption of a volcano. The focus of this document will be to evaluate the ability to encode and serve associated operational data with WXXM 1.1.1.

The document will specifically insist on all the questions raised regarding this topic, along the OWS-7 project life cycle, and will provide synthetic suggestions of improvement whenever such solutions have been found.

1.2 Document contributor contact points

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

Name	Organization
Bruno Simmenauer (Editor)	Alticode
Sébastien Geindre	MeteoFrance
Frédéric Guillaud	MeteoFrance
Claude Lainard	MeteoFrance
Aaron Braeckel	NCAR
Oliver Newell	MIT Lincoln Laboratory
Johannes Echterhoff	International Geospatial Services Institute (iGSI) GmbH
Thomas Everding	University of Muenster - Institute for Geoinformatics
Robin Houtmeyers	Luciad
Jeroen Dries	Luciad
Daniel Hardwick	Snowflake Software Ltd.
Nadine Alameh	OGC (reviewer)

1.3 Revision history

Date	Release	Editor	Primary clauses modified	Description
2010-03-17	0.1.0	BS	All	Definition of initial document outline
2010-04-26	0.2.0	BS	All	Insertion of place markers to keep track of discussions held on the OWS-7 Aviation mailing list
2010-06-30	0.3.0	BS	All	Document structure reviewed, content completed. Release for formal review.

1.4 Future work

The content of this document version is still subject to formal review and might consequently be upgraded when all comments will be taken into account...

1.5 Forward

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 Inc. 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.

2 References

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

ISO 19115(all parts), *Geographic information - Metadata*

ISO 19136:2007, *Geographic information — Geography Markup Language (GML)*

ISO/DIS 19142, *Geographic information — Web feature service*

ISO/DIS 19143, *Geographic information – Filter Encoding*

ISO/DIS 19156, *Geographic information – Observations and Measurements*

OGC 06-121r3, *OpenGIS® Web Services Common Standard*

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

OGC 09-032, *OpenGIS® OWS-6-SWE Event Architecture Engineering Report*

OGC 09-050r1, *OpenGIS® OWS-6-AIM Engineering Report*

OGC 10-079, *OWS-7 Aviation Architecture Engineering Report*

OGC 10-131, *OWS-7 Aviation AIXM Assessment Report*

3 Terms and definitions

Please refer to the OWS-6 Event Architecture Engineering Report (09-032) and the OWS-6 AIM Engineering Report (09-050r1).

4 Conventions

4.1 Abbreviated terms

AIM	Aeronautical information Management
AIXM	Aeronautical Information Exchange Model
EFB	Electronic Flight Bag
ER	Engineering Report
FE	Filter Encoding
FES	Filter Encoding Specification
FIR	Flight Information Region
FL	Flight Level
FPS	Feature Portrayal Service
GML	Geography Markup Language
GMU	George Mason University
HTTP	HyperText Transport Protocol
ICAO	International Civil Aviation Organization
ISO	International Standardization Organization
METAR	METorological Air Report
NNEW	NextGen Network Enabled Weather
NOTAM	NOTice to AirMen
OGC	Open Geospatial consortium
OWS	OGC Web Services
OWS-7	OWS testbed phase 7
O&M	Observation & Measurement
PIREP	PIlot REPort
SIGMET	SIGnificant METeorological information
SLD	Styled Layer Descriptor
TAF	Terminal Aerodrome Forecast
UCAR	University Corporation for Atmospheric Research
UCUM	Unified Code for Units of Measure
VA	Volcanic Ash

WFS	Web Feature Service
WFS-T	Web Feature Service -Transactional
WMS	Web Mapping Service
WXXM	Weather Information Exchange Model
XML	Extensible Markup Language

5 WXXM Assessment Overview

5.1 WXXM overview

WXXM, the Weather Data Model, is a UML-based structural definition for the exchange of weather information, featuring an Aviation-specific layer. It was designed by EUROCONTROL, in partnership with NNEW.

WXXM is not a piece of software, nor does it have any function on its own. It defines a common vocabulary for exchanging weather information between organizations, but it does not inherently provide any sort of functionality to facilitate that exchange. It is, fundamentally, a set of guidelines for how to think about weather data.

The Weather Data Model is in fact a set of three tiered data models, only one part of which is actually called WXXM: Weather Exchange Model. Together, the three models provide conceptual, structural, and physical representations of weather data:

- The Weather Conceptual Model (WXCM) provides a high-level, implementation-independent look at how weather data concepts are connected.
- The Weather Exchange Model (WXXM) provides a more logical and structural (if still implementation-independent) perspective of the same data, in more complete detail — the interrelationships of every weather data concept are spelled out.
- The Weather Exchange Schema (WXXS) is a machine-generated, XML-formatted implementation of the Exchange Model — a "physical" code version of it.

As it has become a common usage, "generic" term WXXM will be used in the following sections of the present document, regardless of what part of the model is actually referred to.

The different layers of the WXXM model are represented on the figure below:

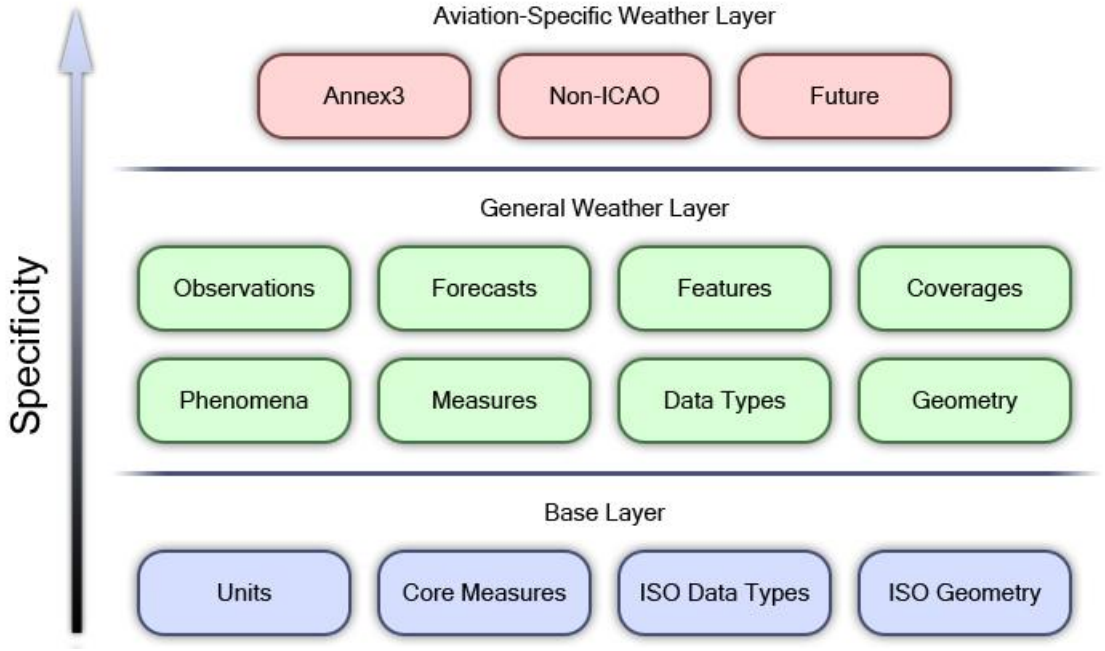

Figure 1: WXXM Model Layers

5.2 WXXM versions

Version 1.1 of WXXM was officially released on Jan. 12, 2010, shortly before the OWS-7 kick-off meeting took place. Issues affecting that version led to the release of version 1.1.1 on Mar. 19, 2010. This initially introduced a bit of confusion for some participants, but it can be stated that most of the OWS-7 work dealing with WXXM has been produced with reference to version 1.1.1.

Suggestions for improvements and corrections designed to solve issues, which resulted from OWS-7 experience is proposed to the NNEW team – currently responsible for the WXXM evolution – for inclusion into the 1.2 version, planned to be released in Fall 2010.

The latest version of the WXXM model is accessible both from the EUROCONTROL OneSky website and from the UCAR wiki.
(at https://wiki.ucar.edu/display/NNEWD/Downloads)

5.3 WXXM data in OWS-7

WXXM has been used in OWS-7 to encode the following data types:

- Volcanic ash advisory and SIGMET, and – marginally – a few turbulence and icing SIGMET samples (MeteoFrance)
- TAF, METAR and PIREP (NNEW)

- Peak point, sink point and thermal couplet features (GMU)

These correspond to diverse uses (and challenges) and the experience gained would at least partially be valid for other WXXM usages, but it should be kept in mind that the overall OWS-7 WXXM experimentation and validation attempt nevertheless remain limited in their scope.

6 WXXM 1.1.1 Data Handling

6.1 OWS-7 WXXM Components

6.1.1 Overview

The three tier OWS-7 architecture (see. e.g. OGC 10-079) encompasses components which are either partially or completely devoted to the management of WXXM aspects, as depicted on Figure 2 below:

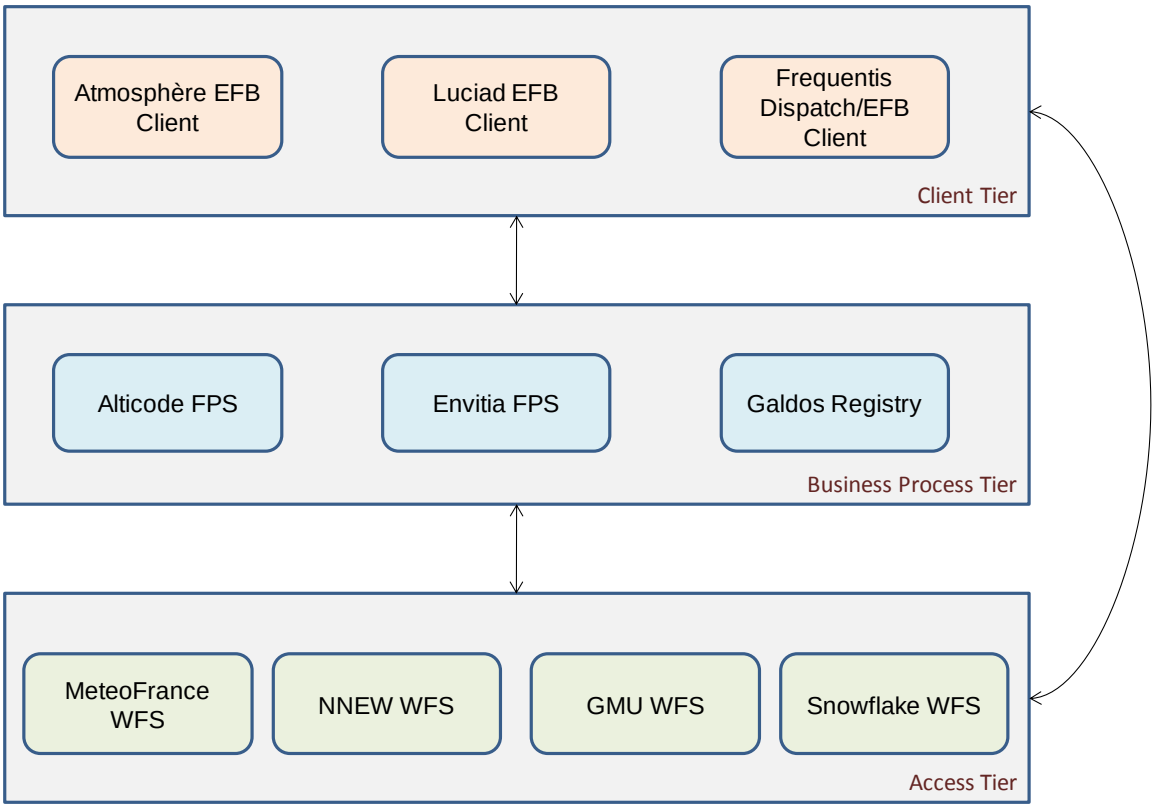

Figure 2: Overview of the OWS-7 WXXM Components Architecture

Details on these components can be found in OGC 10-079 and, as far as the Snowflake WFS is concerned, in OGC 10-131.

6.1.2 Interrelationship

The major data flow amongst the three levels of components is illustrated on Figure 3 below:

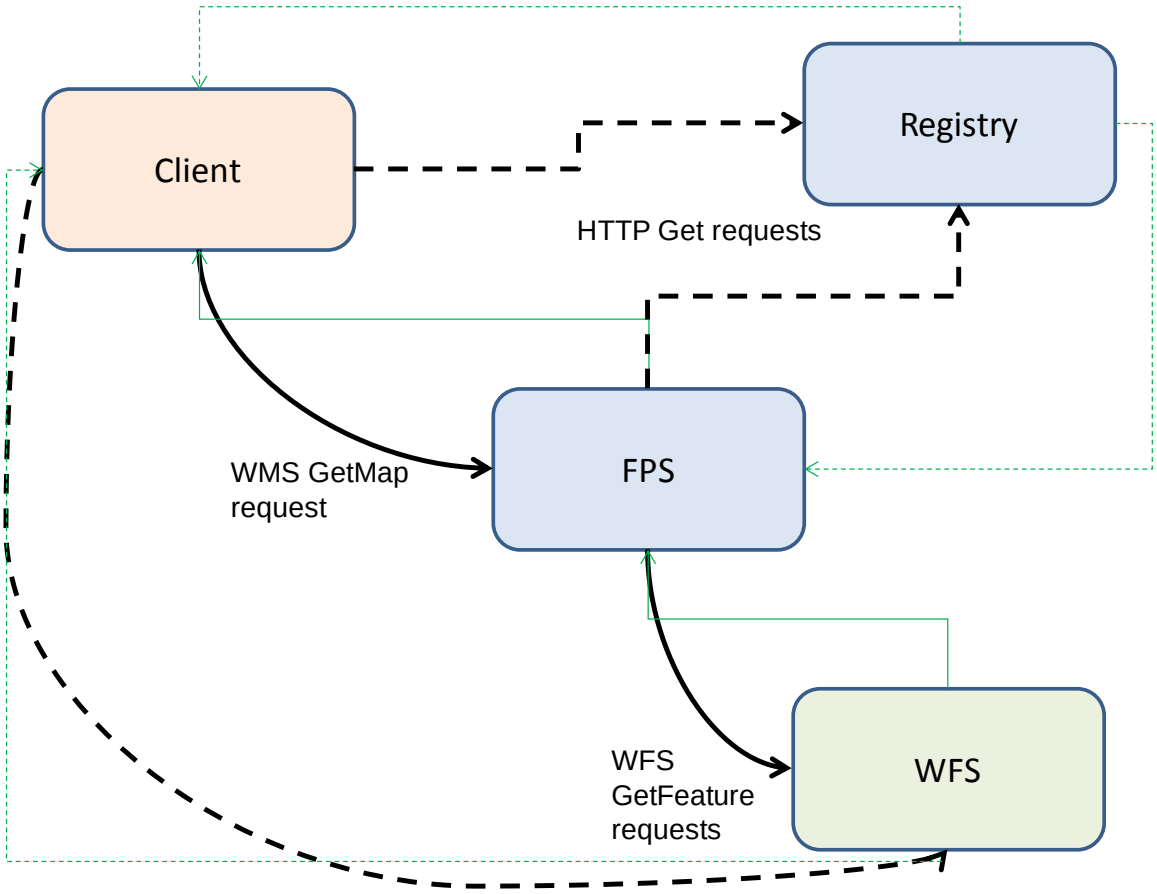

Figure 3: Main Data Flows amongst WXXM Components

The following conventions have been adopted in the graph above:

- Plain lines represent the most classical interaction schema amongst the components whereas dashed ones stand for less conventional ones: either introducing a new use of functionality (registry) or as they involve a direct relationship between the Client and Access Tiers;
- Thick black lines correspond to requests, and thin green lines to the associated responses, which consist in XML encapsulated in HTTP responses, except for the arrow from the FPS toward the Client, which is also an HTTP response, nominally containing the requested map, hence under the requested format (e.g. JPEG, GIF...);
- Requests following specific OGC standards are depicted as curved lines (strait lines stand for "common" HTTP requests and responses)

6.2 Encoding WXXM Data

6.2.1 Convergence and Consistency with other Schemas

6.2.1.1 Measures and Units

Measures, *Units* and *WX_Measures* packages are modeled in such a way that some schema validation (like *mininclusive* and *maxinclusive*) or allowed units per phenomenon can be strictly enforced. It has been questioned whether that capability was at the origin of these packages creation. Also, this approach ties a set of phenomena to a defined set of *uoms*, and this differs from an approach like the one chosen by SWE Common – especially the Quantity type – or at least gml:Measure, which could be more efficient: phenomenon and *uom* are distinct properties there. Actually, the selected approach has been chosen to loosely mirror what is done in AIXM.

As far as standardization is concerned, opening up *uom* definitions to be closer to GML (i.e. UCUM) would be preferable, but this change remains to be studied, discussed and agreed amongst the AIXM and WXXM developers.

6.2.1.2 Use of Codelists instead of Enums

The use of codelists instead of enums has been advocated under "best practices" considerations for all Aviation thread schemas. WXXM developers did not see a priori obstacles to this evolution.

6.2.1.3 *wx:Procedure* Element

It has been noted that the introduction of a *wx:procedure* is not mandatory. The *om:procedure/om:Process* could indeed by used, adding there the *wx:ObservationOrForecastProcess* (as om:Process is of any type).

6.2.1.4 DiscreteCoverage

The GML encoding of *DiscreteCoverage* is defined in OGC 06-188r1, but a different *DiscreteCoverage* is defined by the WXXM model. It was initially thought that the GML definition had a "selective" support of either a spatial domain or a temporal domain, which does not suit aviation domain needs (where e.g. trajectories can cover both space and time) and would therefore justify the creation of a WXXM-specific discrete coverage representation that could support that. The WXXM team even planned to submit a Change Request for GML 3.3 precisely to overcome this identified shortcoming.

The support of both spatial and temporal domain might actually be possible with the GML *DiscreteCoverage*. But since the situation of coverage related developments within OGC did not seem crystal clear to OWS-7 participants (e.g. would coverage aspects be kept within GML or defined by WCS?), and since the schedules did not seem to be compatible anyway, it was thought that keeping the specific definition in WXXM was the soundest solution at least for short to mid-term perspectives.

6.2.1.5 WXXM – AIXM link

It was noted that some characteristics are common to the two domains (e.g. runways description, at least as far as their designation and direction are concerned) and defined – sometimes differently – by the two models. Consistency considerations naturally lead to wonder whether WXXM should be made dependent on AIXM (or even whether the two models should be merged). This interrelationship question has been raised in quite a few other environments than OWS-7 and has far reaching consequences.

In broad terms, the overall conclusion reached on this matter is to keep the models separate while favoring the collaboration amongst their developers and ensuring that each model can benefit from experience gained with the other. All in all, the potential duplication of definitions between the two models can be seen as marginal.

6.2.2 Time Properties

This section about the various time properties usage could as well have been placed within the previous one (*6.2.1 Convergence and Consistency with other Schemas*) as the consistency with the O&M (Observation and Measurement) model indeed plays a central role.

The various phenomena which can be addressed through the WXXM model involve quite a few different time characteristics. Not surprisingly, WXXM users – and especially first time ones – may end up wondering which is the appropriate WXXM time property to be used to encode a given time parameter (which often used to be manipulated routinely under other models or formats).

The associated issue is actually linked to ontology considerations, or, in raw terms, as ensuring that the WXXM developers and users indeed agree on time properties names and on their respective definitions. In order to concretely illustrate this point, the following table lists the time properties of *WX:Observation* and *WX:Forecast* elements, their intended usage and the issues raised by users (when any):

	Property Name	Intended Usage	Comments and/or Issue(s)
WX:Observation	samplingTime	Sampling time for observation	
	validTime	Period when observation is intended to be used	

	Property Name	Intended Usage	Comments and/or Issue(s)
	resultTime	Time of observation result if different than the sampling time (after the fact observation)	
WX:Forecast	samplingTime	Time of forecast	Might also be called the forecast target time, and to be renamed 'phenomenonTime' in O&M 2.0. It is the time that the estimates apply in the world (e.g. if it is a forecast of the weather next Tuesday, then phenomenonTime='next Tuesday'). Q: Should a SIGMET validity period be encoded as a samplingTime/TimePeriod? A: No, rather validTime (see below)
	validTime	Period when forecast is intended to be used	validTime is generally a time period that may include the samplingTime. As per O&M 2.0, it is the time when the estimates are claimed to be usable. e.g. if the forecast was made today but an improved one is expected tomorrow, the validTime='today (but not tomorrow)'
	resultTime	Time when forecast process was run, if different from the forecastAnalysisTime	It is the time when the estimate became available.
	forecastAnalysisTime	Analysis time for the forecast.	This is sometimes also referred to as the reference time of the forecast. If a 2:00 PM forecast is generated at 12:00 PM, the analysis time is 12:00 PM. Note that the analysis time represents the time in the same frame of reference as the forecast itself. In other words, when generating forecasts in non-real time, the analysis time will still be consistent with the original times produced when the forecast was run in real time.

Table 1 – WX:Observation and WX:Forecast Time Properties

6.2.3 Volcanic Ash SIGMET

6.2.3.1 Encoding both an Obervation and a Forecast

The latest WXXM version does not allow encoding both an observation and a forecast in a single SIGMET message. This is nevertheless generally required in VA SIGMETs. This shortcoming did not enable the full encoding of operational VA SIGMETs. The content of WXXM encoded VA SIGMETs had to be restricted to either an observation or a forecast.

6.2.3.2 Phenomenon Type

SIGMETs address several types of weather phenomena (e.g. icing, thunderstorms, clear air turbulence...). It was noted that the phenomenon type is not directly accessible within a tag and generally has to be derived e.g. from the presence of given fields. Adding such an element would considerably simplify filtering and is therefore recommended for next WXXM releases.

6.2.3.3 CompassDirection

The enumerated item (*Compass16*) selected for *avwx:VolcanicAshCloud / wx:movement / wx:MovementDescription / wx:compassDirection* does not correspond to ICAO Annex III specification: *Compass8* should be used instead.

This recommendation will have to be considered also in the light of section 6.2.1.2.

6.2.3.4 Coverage

Several geographical areas are generally associated to a VA SIGMET:

- The FIR concerned with the SIGMET;
- The extension of the single – or often multiple – part(s) of ash cloud(s) which can be encountered within that FIR airspace.

Both kinds of areas are modeled as polygons. Whereas it was thought natural to encode ash cloud polygons as *SIGMET / airSpaceAreaForecast / Forecast / result / volcanicashcloud / extentof*, the encoding of the FIR extension was more problematic. Two possible parameters had indeed been identified:

- *SIGMET / airSpaceAreaForecast / Forecast / featureOfInterest / avwx:Airspace*
- *SIGMET / issuedFor / avwx:Airspace*

This will be brought up with the broader WXXM team to try to trace some of the history and intent, and the relationship to O&M and ICAO representations. In the meantime, it

was decided (somewhat arbitrarily) to opt for the second one at least for the OWS7 developments.

Using the first option had in fact been recommended by some participants, but others noticed that it could not be used directly, because the model of *avwx:Airspace* does not have a property (which is required in the O&M model, see e.g. https://www.seegrid.csiro.au/twiki/bin/view/AppSchemas/ObservationsAndSampling#Unknown_features). The solution would then be to select *om:SamplingFeature* as *featureOfInterest.* The *sampledFeature* inside the sampling feature could then be the affected airspace and the *SamplingFeature* could contain the following reference: <sa:sampledFeature xlink:href="urn:ogc:def:nil:OGC:unknown"/>. Nevertheless, the appropriate *featureOfInterest* should relate to the whole SIGMET (used as a "product"), not uniquely to an observation (or forecast) included in that message.

It had also been contemplated to encode the observation or forecast results as a feature collection associated with a boundary box (3 dimensional, or even 4D if the forecast horizon was to be encoded simultaneously), thus encompassing all the feature geometries in this collection. But the *FeatureCollection* element is deprecated in GML 3.2, this collection would therefore have to be defined directly in the WXXM model. No definitive solution has been agreed at the time of report writing.

Finally, the necessity to systematically include the description of the FIR boundary in all SIGMETs was also questioned (mostly for bandwidth optimization considerations, as it can generate "heavy" content for characteristics which are mostly of static nature). No definitive work around mechanism has been designed yet, but a possibility could lie with turning the use of a *name* element mandatory (within *airspace*) and requesting that it contain the FIR's ICAO 4 character designator (using gml:id for that purpose does not seem to be an adequate solution).

6.3 Serving WXXM Data

6.3.1 VA SIGMET retrieval based on intersection with a 4D trajectory

As mentioned in section 6.2.3.1 above, the spatial extensions associated to ash cloud model within a VA SIGMET can be multiple:

- The dispersion of a volcano plume potentially generating several clouds;
- An ash cloud being modeled possibly at different flight level layers;
- An ash cloud contour being forecast at different time horizons.

The requirement for an adequate *featureOfInterest*, also mentioned in that section, is essential in the context of the most typical use case, where a user would like to request the WFS for "those VA SIGMET(s) whose ash cloud(s) intersect a given 4D flight trajectory".

The problem that there is no distinct spatial property in an observation (and thus currently also in a forecast) is known and there is work ongoing to define an O&M extension to add this as an observation parameter (which web services may or may not support). The problem is due to the O&M concept where there needs to be some knowledge about the structure of the feature of interest for which an observation was made. One solution might be to add a property to the *avwx:AreaReport*, which could be the union of the spatial extent of all the ash cloud layouts within the concerned FIR. Whether or not this would make up the perfect solution isn't clear at the time of document writing (e.g. would it allow handling appropriately the temporal dimension as well?).

In order to handle the case of multiple clouds to be modeled in a given FL range, the *wx:ExtentOf* element definition might usefully be upgraded to include *gml:MultiSurface* as an additional possibility.

6.3.2 "*AnyType*" Issue

Snowflake's GO Loader® provides the functionality to automatically configure the schema translations required to translate the data into the database schema. However, the WXXM schema does not substitute the *anyType* data type for *om:result* property for a specified data type (e.g. the different WXXM features). This means GO Loader® cannot automatically define the database schema for these properties and discards the property. Manual intervention was needed to configure the load and schema generation of this property.

It is recommended that for future releases of WXXM, that feature types extended from observation feature types should extend form specialized observation feature types classified by their result type as recommended in O&M (ISO/DIS 19156 Geographic information – Observations and Measurements). This will enable generic tools such as GO Loader® to parse and load the result properties.

Within the test bed, seemingly valid WXXM data was created where the only geometry properties populated were within the *om:result anytype* property. This data structure would be very difficult for generic WXXM clients to predict and query upon, as the *anyType* could contain anything.

6.4 Portraying WXXM Data

6.4.1 Compliancy and interoperability issues

6.4.1.1 Feature collection definition

The definition of the *wx:FeatureCollectionType* in the XML Schema of WXXM 1.1.1 is not compliant with the OGC GML 3.2 specification. According to this specification, a feature collection should be defined as follows (OGC 07-036, paragraph 9.9.1):

"A GML feature collection is any GML feature with a property element in its content model whose content model is derived by extension from gml:AbstractFeatureMemberType (see 9.9.2)."

An example illustrating this principle:

```
<complexType name="MyFeaturesType">
   <complexContent>
      <extension base="gml:AbstractFeatureType">
         <sequence>
            <element name="myMember"
               type="ex:MyFeaturesMemberType"
               minOccurs="0" maxOccurs="unbounded"/>
         </sequence>
         <attributeGroup ref="gml:AggregationAttributeGroup"/>
      </extension>
   </complexContent>
</complexType>

<complexType name="MyFeaturesMemberType">
   <complexContent>
      <extension base="gml:AbstractFeatureMemberType">
         <sequence minOccurs="0">
            <element ref="gml:AbstractFeature"/>
         </sequence>
         <attributeGroup ref="gml:AssociationAttributeGroup"/>
      </extension>
   </complexContent>
</complexType>
```

Applications should be able to rely on this pattern to recognize feature collections in custom application schemas. In the WXXM 1.1.1 schema however, the *FeatureCollectionType* is defined as follows:

```
<complexType name="FeatureCollectionType">
   <complexContent>
      <extension base="gml:AbstractFeatureType">
         <sequence>
            <element maxOccurs="unbounded" minOccurs="0"
               name="featureMember"
               type="gml:FeaturePropertyType"/>
         </sequence>
</extension>
```

```
    </complexContent>
</complexType>
```

As the predefined pattern is not implemented, it is no longer possible to automatically detect that the wx:FeatureCollectionType is a GML feature collection, which is key information in order to know how to process the data correctly.

This problem has been noted by the WXXM developers' team, and will be fixed in the next version.

6.4.1.2 Interpretation of observation result

WXXM 1.1.1 is based upon the OGC Observations & Measurements specification (O&M). The XML Schema associated with this specification defines an object type (*ObservationType*) to model observations, which in turn has a property 'result' to model the observation results. This 'result' property is of the XML Schema type 'anyType', meaning that it can contain anything (see also section 6.3.2 above). The documentation in the O&M schema explicitly mentions that custom application schemas may choose to constraint the type of the result.

In the WXXM schema, the result type is not constrained, resulting in a variety of contents in the result encoding when working with different data providers. This makes it very difficult for applications to detect the type of the observation result when decoding WXXM data. To increase interoperability when working with different data vendors, it should be considered to either constrain the result type, or to clearly specify how the value of the *observedProperty* element maps to a specific XML Schema type for the result.

6.4.2 Generic Portraying Issues

The following considerations are derived from the experience gained during the implementation of an FPS solely based on the use of OGC standards. They are not WXXM specific, and, as far as the second sub-section is concerned, not even FPS specific...

6.4.2.1 Decoupling WFS GetFeature and Styling Parametrization

The solution selected to store WFS GetFeature parameters (notably the filtering ones) within the *sld:LayerFeatureConstraints* element, which seems to be the common practice amongst the OWS-7 participants, This is not consistent with OGC 09-015 (OWS6 ER about SLD changes), which proposed to remove that element, considering that it was redundant with the Rule filtering capabilities.

Using the *sld:UserStyle / se:FeatureTypeStyle / se:Rule* element to store WFS GetFeature request parameters, as suggested by OGC 09-015, will no longer permit to fully decouple

the styling profile from the data retrieval one. As an example of the usefulness of this decoupling capability, one could consider the situation where a client wishes:

- To implement a spatial filter for the selection of features lying in the vicinity of a given flight plan. This filter will be included in the GetFeature request transmitted to the WFS);

- To define various styling options depending on the actual airspace properties (e.g. property filters which filter on airspace classification to address specific style requirements). These will be implemented in the *sld:UserStyle / se:FeatureTypeStyle / se:Rule* element, used by the FPS to style the data retrieved from the WFS.

6.4.2.2 Schemas "Anachronism"

In Alticode's FPS implementation, namespaces checks are strictly enforced. This was initially thought to be a sound practice, but ended up causing many problems when attempting to interconnect with clients and servers using different versions of services specifications.

The main difficulty encountered was caused by clients' use of schemas versions which were posterior to those specified in the services implemented (WMS, SLD, SE, FE). Local work-around solutions, consisting in allowing alternate namespaces, were implemented. However, these are not considered to be satisfactory, mostly because assumptions made on the structure of elements, consistent with the version specified in the implemented standards, might no longer be valid for later versions of the schemas.

6.5 Conclusion

6.5.1 Suggested WXXM Evolutions

The following table is intended to provide a synthetic view of the different suggestions for WXXM evolution, which resulted from OWS-7 experience.

Topic	Status or summary of proposed evolution	Document section(s) for details
Further clarify time elements usage	Provide additional use examples to improve users' understanding (and appropriate selection). Maintain naming consistency with O&M	§ 6.2.2
Further clarify (coverage) areas usage	For the particular case of VA SIGMET: • Clarify respective usage of the two *avwx:Airspace* elements • Further investigate what type of *featureOfInterest* would suit the	§ 6.2.3.4, § 6.3.1

	need (study the interest of adding gml:MultiSurface option, and whether a *FeatureCollection* kind of element could be helpful)	
Allow encoding of obs. and forecast in a given SIGMET	Modify the element definition to allow the presence of both sub-components	§ 6.2.3.1
Get rid of *anyType* usage	Use specialiezd observation feature types instead	§ 6.3.2, § 6.4.1.2
Alignment with O&M	Use om:procedure / om:Process instead of defining wx:procedure	§ 6.2.1.3
Ease SIGMET filtering based on phenomenon type	Add a phenomenonType element to SIGMET definition	§ 6.2.3.2
Correct *wx:FeatureCollectionType* definition	Noted, will be fixed in next WXXM version	§ 6.4.1.1
Correct wx:compassDirection value	Use Compass8 instead of Compass16	§ 6.2.3.3

Table 2 – Summary of Suggested WXXM Evolutions

6.5.2 Ideas for Future Work

As mentioned in section 5.3 above, OWS-7 could only address a small subset of what WXXM potentially offers. Even though some of the results obtained can clearly be generalized, the limited scope of validation which could be carried out during OWS-7 should always be kept in mind when interpreting the conclusions.

OWS-7 participants expressed an unambiguous interest in going on with the process of testing WXXM utilization with operational data.

