Open Geospatial Consortium

Publication Date: 2014-07-16

Approval Date: 2014-06-14

Posted Date: 2014-05-15

Reference number of this OGC® project document: OGC 12-027r3

Reference URL for this document: http://www.opengeospatial.net/doc/DP/wfs-temporality-extension

Category: OGC® Discussion Paper

Editor: Timo Thomas

OGC Web Feature Service (WFS) Temporality Extension

Warning

This document is not an OGC Standard. This document is an OGC Discussion Paper and is therefore 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, an OGC Discussion Paper should not be referenced as required or mandatory technology in procurements.

Document type: OGC® Discussion Paper
Document subtype: NA
Document stage: Approved for Public Release
Document language: English

License Agreement

Contents

i. Abstract

This OGC discussion paper provides a proposal for a temporality extension for the WFS 2.0 and FES 2.0 standard. It is based on the work of and experiences made in several OWS test beds, in particular OWS-7, OWS-8 and OWS-9, Aviation threads and discussions at the 2011 OGC TC meeting in Brussels, Belgium. It partially replaces and advances the document “OWS-8 Aviation: Guidance for Retrieving AIXM 5.1 data via an OGC WFS 2.0” [4].

ii. Keywords

ogcdoc, ogc documents, wfs, fe, aviation, temporality, temporal

iii. Submission contact points

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

CONTACT	COMPANY
Timo Thomas	
Johannes Echterhoff	interactive instruments GmbH
Jeroen Dries	Luciad NV
Andreas Höfer	COMSOFT GmbH

iv. Revision history

Date	Release	Author	Paragraph modified	Description
2012-10-16	1.0	Timo Thomas, Johannes Echterhoff	All	Initial draft for general review.
2014-04-30	r3	Timo Thomas,	All	Conformance with OGC

		Johannes Echterhoff		Specification Model

Introduction

The Aeronautical Information Exchange Model (AIXM) is designed to enable the management and distribution of Aeronautical Information Services (AIS) data in digital format. The newest version of this model, AIXM 5.1, is based on GML 3.2 and features an exhaustive temporality model loosely based on the GML Dynamic Feature Model.

Various interoperability test-beds at OGC, in particular OWS-7 and OWS-8, have applied OGC's WFS 2.0 and FES 2.0 standards on AIXM 5 data. Though it could be demonstrated that a basic interoperability is possible, it turned out that some key requirements could not be fulfilled. This paper summarizes the observations made and shows that these requirements are not specific to AIXM 5, but more generally apply to any data model featuring temporality. To overcome these shortcomings, a proposal is made for an extended type of WFS query: a dynamic feature query.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. Open Geospatial Consortium Inc. shall not be held responsible for identifying any or all such patent rights. However, to date, no such rights have been claimed or identified.

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 specification set forth in this document, and to provide supporting documentation.

1 Scope

This document specifies the AIXM 5 temporality extensions of the WFS 2.0 and FES 2.0 standards. It identifies the required functionality and use cases, analyses the existing WFS and FES standards, documents their limitations and defines extensions to overcome them.

2 Conformance

2.1 Conformance Classes

The following figures outline the conformance classes defined by this specification. Details and the definition can be found in Annex E.

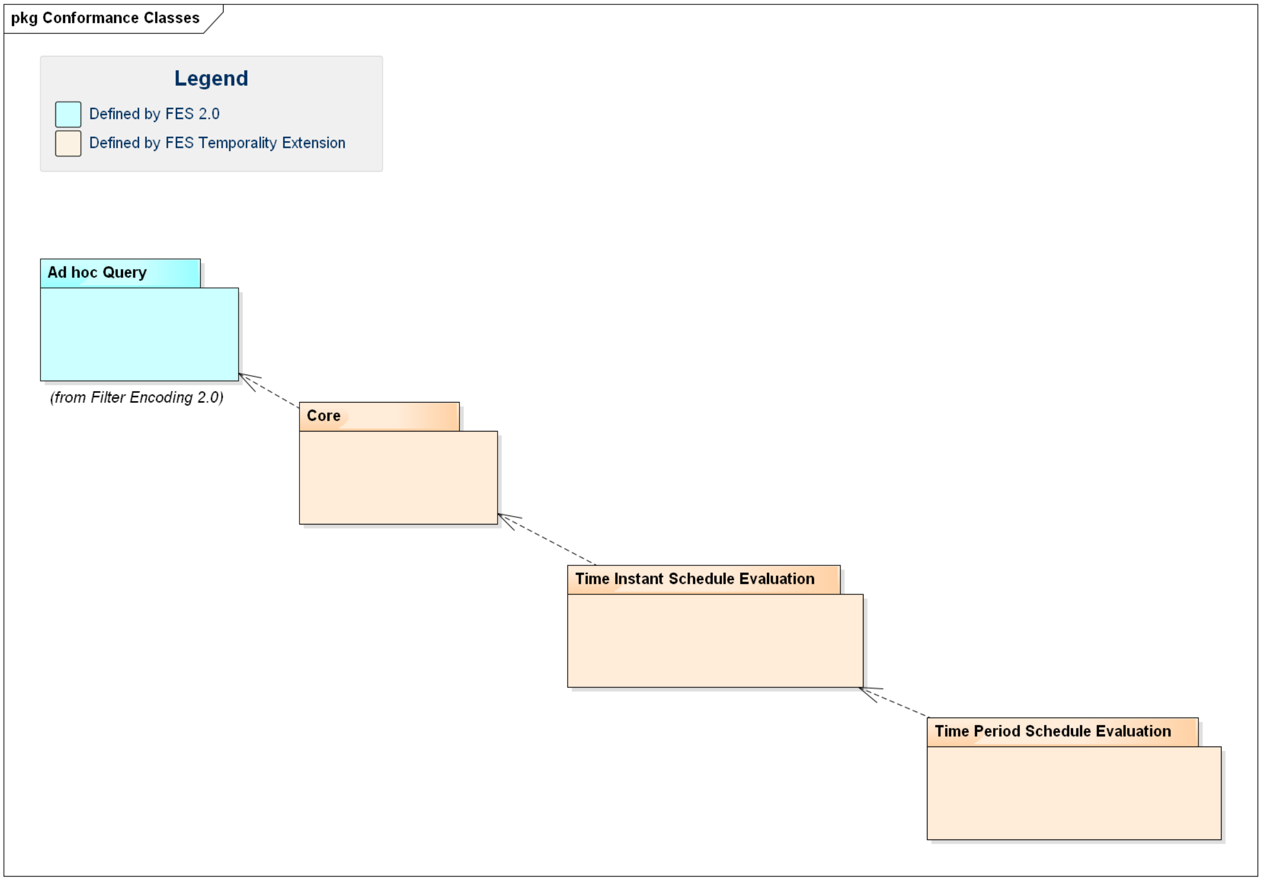

pkg Conformance Classes
Legend
Defined by FES 2.0
Defined by FES Temporality Extension
Ad hoc Query
(from Filter Encoding 2.0)
Core
Time Instant Schedule Evaluation
Time Period Schedule Evaluation

Figure 1: FES Temporality Extension Conformance Classes

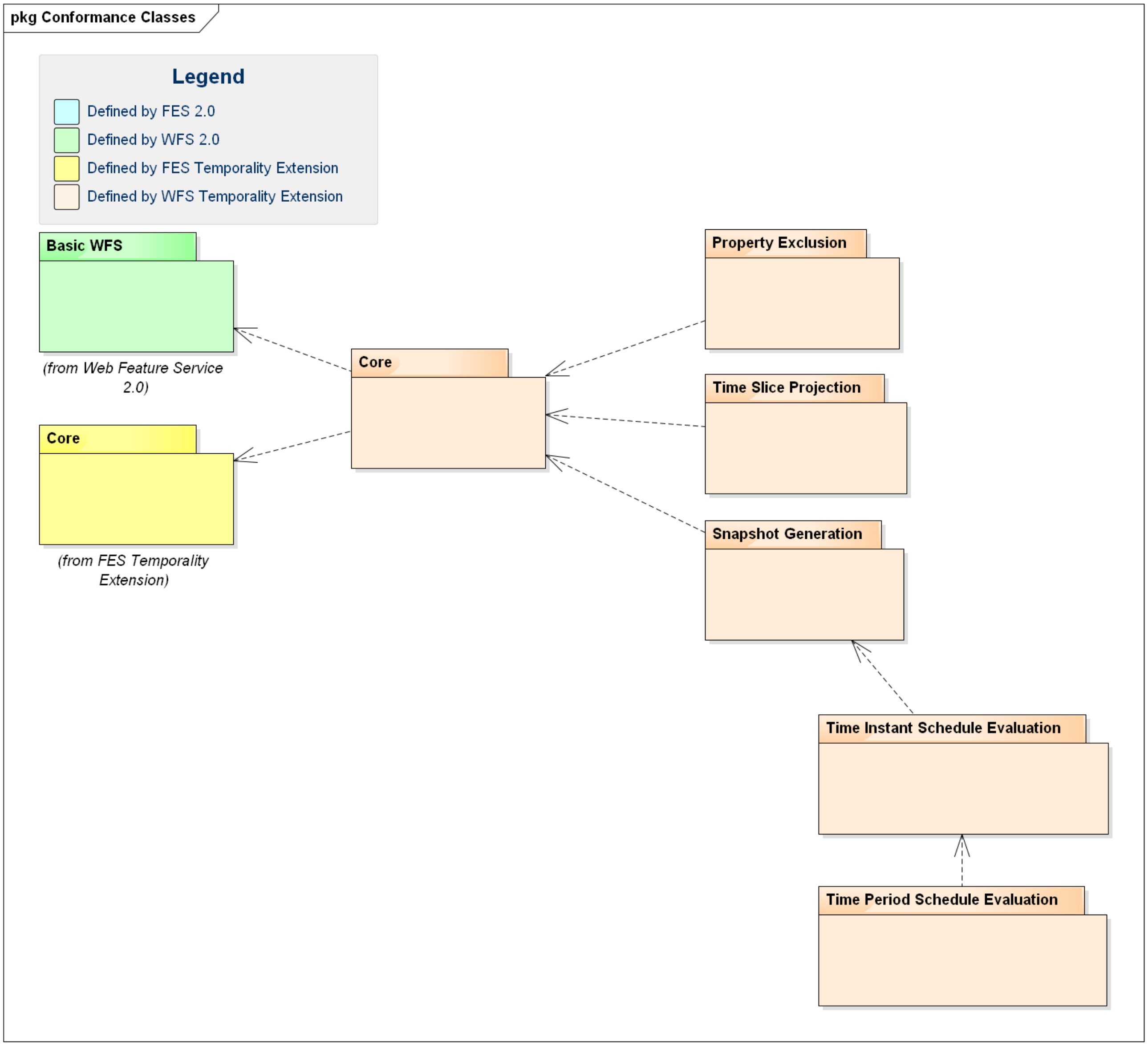

pkg Conformance Classes
Legend
Defined by FES 2.0
Defined by WFS 2.0
Defined by FES Temporality Extension
Defined by WFS Temporality Extension
Basic WFS
(from Web Feature Service 2.0)
Core
(from FES Temporality Extension)
Core
Property Exclusion
Time Slice Projection
Snapshot Generation
Time Instant Schedule Evaluation
Time Period Schedule Evaluation

Figure 2: WFS Temporality Extension Conformance Classes

3 References

The following normative documents contain provisions that, through reference in this text, constitute provisions of 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.

[1] ISO/DIS 19142 and OGC 09-025r1, OpenGIS® Web Feature Service 2.0 Interface Standard (2010-11-02)

[2] ISO/DIS 19143 and OGC 09-026r1, OpenGIS® Filter Encoding 2.0 Encoding Standard (2010-11-22)

[3] OGC 11-093r2, OGC® OWS-8 Aviation Architecture ER (2011-09-30)

[4] OGC 11-073r2, OGC® OWS-8 Aviation: Guidance for Retrieving AIXM 5.1 data via an OGC WFS 2.0 (2011-11-04)

[5] OGC 10-131r1, OGC® OWS-7 Aviation – AIXM Assessment Report (2010-08-18)

[6] OGC 05-007r7, OpenGIS® Web Processing Service (2007-06-08)

[7] XQuery 1.0: An XML Query Language (Second Edition), W3C Recommendation 14 December 2010, at http://www.w3.org/TR/2010/REC-xquery-20101214/

[8] OGC 11-171, Requirements for Aviation Metadata - OGC Best Practice (2011-11-10)

[9] OGC 11-172, Guidance on the Aviation Metadata Profile (2011-11-10)

[10] AIXM 5 Temporality Model 1.0 (2010-09-15)

[11] Lake R. et al., GML Geography Mark-Up Language, Wiley 2004

[12] “Donlon” data set, https://extranet.eurocontrol.int/http://webprisme.cfmu.eurocontrol.int/aixmwiki_public/bin/view/Main/XML_Tags#HExampleDonlondataset (2014-04-26)

4 Conventions and Abbreviated Terms

4.1 Abbreviations

AIP	Aeronautical Information Publication
AIRAC	Aeronautical Information Regulation And Control
AIXM	Aeronautical Information Exchange Model
AIXM-TM	AIXM 5 Temporality Model
API	Application Program Interface

ER	Engineering Report
FES	Filter Encoding Specification (Version 2.0 – if not stated otherwise)
GML	Geography Markup Language
HTTP	Hypertext Transfer Protocol
ISO	International Organization for Standardization
NOTAM	Notice To Airmen
OGC	Open Geospatial Consortium
UML	Unified Modeling Language
URL	Uniform Resource Locator
W3C	World Wide Web Consortium
WFS	Web Feature Service (Version 2.0 – if not stated otherwise)
WPS	Web Processing Service
XML	eXtended Markup Language
XPath	XML Path Language

4.2 Namespace bindings for XML elements

Prefix	Bound Namespace
gml	http://www.opengis.net/gml/3.2
aixm	http://www.aixm.aero/schema/5.1
fes	http://www.opengis.net/fes/2.0
wfs	http://www.opengis.net/wfs/2.0
fes-te	http://www.opengis.net/fes-te/1.0
wfs-te	http://www.opengis.net/wfs-te/1.0
ows	http://www.opengis.net/ows/1.1

5 AIXM 5 Temporality Model

AIXM 5 is based on GML 3.2. All features inherit from the type **gml:DynamicFeature** which is the base class of the GML Dynamic Feature Model (DFM). The properties of a feature are all time-variant and encoded in time slices. The only exceptions to this are global identifiers, names, metadata and a bounding box. A detailed comparison of the two models is given in 5.6.

The following section describes issues with the GML Dynamic Feature Model. Then a brief introduction is given to the AIXM Temporality Model and to what makes it special compared to the Dynamic Feature Model in GML 3.2.

5.1 Issues with GML DynamicFeature Model

The GML 3.2.1 standard [OGC 07-036] defines the UML model and XML Schema encoding of the DynamicFeature, DynamicFeatureCollection and AbstractTimeSlice types. The UML model is provided in [OGC 07-036] D.3.11 while the XML Schema is defined in section 14.5 of that document.

The GML standard provides an example (see [OGC 07-036] section 14.5.7) in which a cyclone is modeled as a DynamicFeature, with its movement status captured in time slices that together constitute the history of the cyclone.

GML does not define modeling and management aspects of dynamic features in sufficient detail:

- Modeling/encoding:
  - GML does not clearly specify the rules for modeling and encoding dynamic features.
    - Section 14.5.1 says that "… dynamic feature classes will normally be extended to suit particular applications."
    - Section 14.5.3 states that "this [the gml:dynamicProperties group] allows an application schema developer to include dynamic properties in a content model in a standard fashion." That the gml:dynamicProperties group allows the inclusion of dynamic properties in a content model causes confusion, as it may suggest that it is this group which needs to be extended by an Application Schema developer – although no UML type exists for this group.
    - Section 14.5.6 says: "A timeslice encapsulates the time-varying properties of a dynamic feature - it shall be extended to represent a time stamped projection of a specific feature."
  - These statements suggest that the DynamicFeature and AbstractTimeSlice types are extended when modeling a specific DynamicFeature type in an Application Schema by adding the same properties to both of them. This would then enable snapshots as well as the full feature history to be represented. However, this is not explicitly stated in the standard. It is also not explicitly stated that all of the dynamic feature properties would need

to be optional to enable representation of snapshots and the complete feature history.

- Management:
    - Section 14.5.4 in GML states that “each time-stamped instance [of a dynamic feature] represents a ‘snapshot‘ of a feature.” This appears to refer to the gml:validTime, which is an optional property of a GML DynamicFeature type. However, the relationship to gml:validTime is not explicitly stated there. Section 14.5.5, though, states that “a gml:DynamicFeatureCollection is a feature collection that has a gml:validTime property (i.e. is a snapshot of the feature collection)” which provides a hint that this assumption is correct.
    - GML also does not clearly define what a snapshot is if the ‘time-stamp’ (presumably the gml:validTime) is not a gml:TimeInstant (or a gml:TimeNode). Again the reader is forced to make assumptions, though an obvious assumption would be that if a gml:TimePeriod or –Edge is given for a ‘snapshot’ that then the state of the dynamic feature is constant for that time.
    - GML does not define how to solve situations in which multiple time slices contain a value for a dynamic property for a given point in time. The AIXM-TM avoids this potential ambiguity through the definition of sequence and correction numbers for time slices.
    - Other application aspects that are covered in the AIXM-TM, such as deltas for complex or multi-occurring properties, or canceling a time slice are not defined in GML.

Apparently the GML standard only defines the basic model and encoding of the dynamic feature base type. Actual rules for modeling/encoding and management of specific dynamic features that are defined in an Application Schema are neither covered in GML 3.2.1 nor in the current OGC or ISO standards baseline.

5.2 Time Slice Interpretation

AIXM 5 distinguishes four types of time slices: BASELINEs, PERMDELTAs, TEMPDELTAs and SNAPSHOTs. The time slice type is encoded in a time slice property called ‘interpretation’. From now on, the term *time slice interpretation* may be used as a synonym to *time slice type* in this document. The function of the time slice types are described in Table 1. BASELINEs and SNAPSHOTs are the direct result of PERMDELTAs and TEMPDELTAs, which means that they are only a different representation of the information carried in the delta time slices. Their generation is the result of a merging process in which overlapping deltas are ordered according to their sequence number (which is a property of all AIXM 5 time slices) to define a unique and unambiguous result.

Table 1 — Time Slice Types in AIXM 5

Time Slice Type / Interpretation	Function
PERMDELTA	Contains all properties that change permanently; valid at a time instant. Example: a change in the length of a runway due to construction works.
BASELINE	Current state of the feature as the result of permanent changes; valid for a time period (= the sum of all PERMDELTAs relevant for the time period)
TEMPDELTA	Contains all properties that change temporarily; valid for a time period. Example: a closure of a runway due to icing.
SNAPSHOT	Current state of the feature as the result of all permanent and temporary changes; valid for a time instant[A]. (= the sum of all PERMDELTAs and TEMPDELTAs relevant for the time instant)
[A] The concept of snapshots is extended to time periods later in this document	

SNAPSHOT time slices cannot be stored in a WFS and a WFS cannot retrieve them from a persistent storage. This is because in theory, an unlimited number of SNAPSHOT time slices exist as they are valid for a single point in time only.

5.3 Sequences, Corrections and Cancelations

An AIXM 5 feature never forgets its history, which includes expected events on the state of a feature to happen in the future, that were known at some point in the past. This implies that a time slice is never deleted or updated. Every change is communicated through the insertion of new time slices, for which overwrite rules apply. For this purpose AIXM 5 introduced the concepts of *sequences*, *corrections* and *cancelations*. Every time slice of a feature has a unique identifier (key) which consists of the following properties:

- the interpretation (time slice type),
- the sequence number and
- the correction number

For every sequence (i.e. the set of time slices that share a sequence number), there is always only one *active* time slice: it is the one with the highest correction number. All time slices with a lower correction number (in a sequence) are considered to be outdated. If the active time slice is a cancelation, which is indicated by a specific value of the validTime property, then the sequence is considered to be canceled[1]. This functionality

[1] A canceled sequence is always closed and cannot be re-opened by the insertion of another time slice with a higher correction number.

supports communicating both updates and deletions without losing the history. For simplification we define:

- a *sequence*: the set of time slices of a feature that share the same interpretation value and sequence number
- the *active time slice*: the time slice of a sequence with the highest correction number
- a *canceled sequence*: a sequence whose active time slice is a cancelation
- *canceled time slices*: the time slices of a canceled sequence
- *corrected time slices*: all time slices of a sequence except the active time slice

5.4 Lifetime of a feature

AIXM 5 defines two time slices that mark the beginning and the end of the feature's lifetime. They are called commissioning and decommissioning time slices, respectively. Both are PERMDELTA time slices that contain a **featureLifetime** property. For simplification we define:

- *commissioning*: the start of life / existence of a feature
- *decommissioning*: the end of life / time of withdrawal of a feature
- *lifetime of a feature*: the time period from commissioning to decommissioning, i.e. all points in time that are not before and not after that period

5.5 Properties with Schedules

In AIXM 5, a special property type exists for modeling periodic events: **PropertiesWithSchedules** [10]. It introduces an additional temporality aspect to properties which is overlaid onto the underlying temporality model. Good examples for schedules are the opening times of an airport (e.g. daily 8-18h) and the regular activation of airspaces for military operations (e.g. weekly on Saturday 8-16h). Schedules consist of a list of consecutive time sheets, where each time sheet contains a validity period and set of values.

5.6 Differences to the GML Dynamic Feature Model

AIXM 5 is only loosely based on the GML Dynamic Feature Model. Important differences exist both

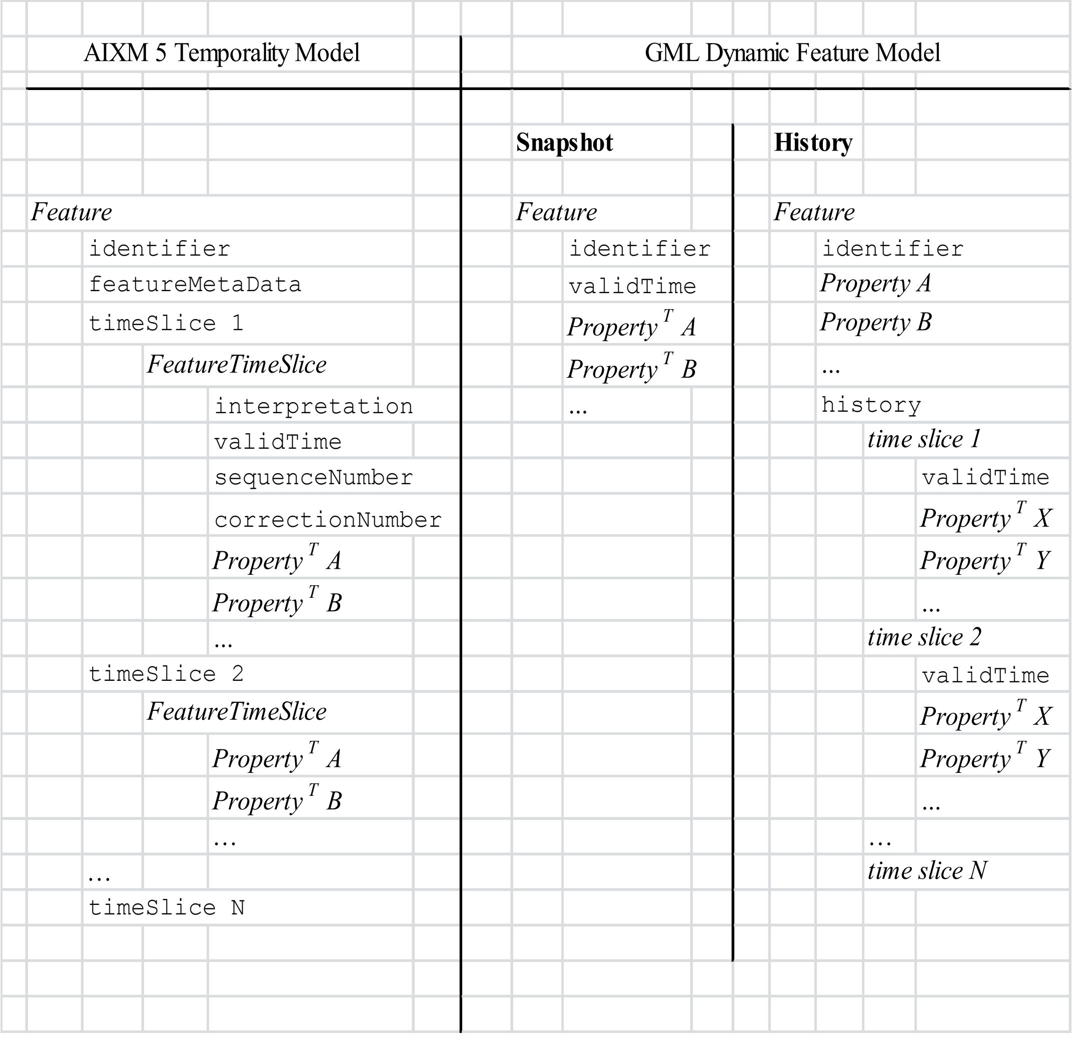
- on a conceptual level (see Table 2) as well as

- in the structure of the features (see Figure 3).

Table 2 — Differences between AIXM and GML temporality models, conceptual level

Concept	AIXM	GML 3.2
Base types	gml:DynamicFeatureType, but restricts the inheritance to only a few elements. gml:validTime and gml:history are not among them	gml:DynamicFeatureType
Types of temporary changes	Time slice types: SNAPSHOT (current state), PERMDELTA (permanent change), TEMPDELTA (temporary change), BASELINE (current permanent state, excluding temporary changes)	Snapshots[2] (current state) and history (list of changes in time-variant properties)
Corrections	Supported through sequence and correction numbers	No such concept
Cancelations	Supported through special interpretation of validTime	No such concept
Schedules	Supported through interpretation of special timeInterval properties	No such concept

[2] What exactly a snapshot is is not clearly defined in GML 3.2.1. Here, we refer to the definition in [11] section 14.5, which is “A ‘snapshot’ is a representation of the state of a feature at a point in time or during some interval”



AIXM 5 Temporality Model

- *Feature*
  - `identifier`
  - `featureMetaData`
  - `timeSlice 1`
    - *FeatureTimeSlice*
      - `interpretation`
      - `validTime`
      - `sequenceNumber`
      - `correctionNumber`
      - *Property*^{T} *A*
      - *Property*^{T} *B*
      - ...
  - `timeSlice 2`
    - *FeatureTimeSlice*
      - *Property*^{T} *A*
      - *Property*^{T} *B*
      - …
  - …
  - `timeSlice N`

GML Dynamic Feature Model

Snapshot

- *Feature*
  - `identifier`
  - `validTime`
  - *Property*^{T} *A*
  - *Property*^{T} *B*
  - ...

History

- *Feature*
  - `identifier`
  - *Property A*
  - *Property B*
  - ...
  - `history`
    - *time slice 1*
      - `validTime`
      - *Property*^{T} *X*
      - *Property*^{T} *Y*
      - ...
    - *time slice 2*
      - `validTime`
      - *Property*^{T} *X*
      - *Property*^{T} *Y*
      - ...
    - …
    - *time slice N*

Figure 3 — Comparison of the XML structure in the AIXM 5 Temporality Model and the GML Dynamic Feature Model

Figure 3 explained: each item below "Feature" represents an XML element (including the Feature element itself). Because XML can be quite verbose, tags, attributes and namespaces have been omitted because they are of little value in this comparison. Elements in `type-writer notation` denote real existing elements, whereas elements in italic letters denote elements of a certain type or class. Feature properties specific to a concrete feature type fall into two classes: time-variant properties (*Property*^{T}) and time-invariant properties (*Property*). The latter only exist in the GML Dynamic Feature Model.

5.7 Conclusion

Despite the AIXM Temporality Model being based on the dynamic feature model of GML through the common base type gml:DynamicFeature, in practice there is little in common. AIXM-TM introduces a number of important new concepts: the distinction between temporary and permanent changes, corrections, cancelations and properties with schedules. However, both models share the approach of modeling time-variant data by the introduction of time slices, which associate properties with a validity time. Most important of all is that the AIXM Temporality Model completely defines how dynamic features with time-varying property values are modeled, encoded, interpreted and managed.

The Temporality Extension specified in this document defines the necessary additions to FES 2.0 and WFS 2.0 that support specific data retrieval use cases required by the Aviation domain. The Temporality Extension thus enables better support for retrieval of AIXM data, but in general supports retrieval of dynamic feature data that follows the principles of the AIXM Temporality Model. As outlined in [OGC 11-093r2] the Temporality Model can become a standalone standard that extends the concepts of the General Feature Model (GFM) to support modeling, encoding, interpretation and management of dynamic features.

6 Use Cases for a WFS Service Serving AIXM 5 Data

6.1 Data Retrieval

The use-cases for the retrieval of AIXM 5 data are manifold and come from different areas of applications. Table 3 categorizes them, gives examples and derives the technical requirements for a query processor.

Table 3 — Use cases for AIXM 5 data retrieval

Ref. No.	Use Case	Category	Example	Technical requirements
1	Retrieve the complete state of a feature at a point in time	Visualization, decision support	Retrieve the state of a runway at the time of arrival.	Filtering of features and generation of SNAPSHOT time slices.
2	Retrieve all features fulfilling a certain criteria at a point in time	Decision support	Retrieve all airports in a certain area that are operational at the time of arrival.	Filtering of features and generation and filtering of SNAPSHOT time slices.
3	Retrieve all time slices of a feature relevant for (i.e. affecting the state at) a point in time	Change-aware visualization	Retrieve all time slices of a runway relevant for the time of arrival. (This enables the client to process any TEMPDELTAs (e.g. digital NOTAMs) received at a later point in time).	Filtering of features and determination of the relevant time slices according to the AIXM-TM.

4	Retrieve the BASELINE of a feature valid at a point in time	AIP publication	Retrieve the BASELINE of an airport at an AIRAC effective date.	Filtering of features and filtering and aggregating the contained time slices, ignoring corrected and canceled time slices[3].
5	Retrieve all TEMPDELTAs of a feature fully or partly valid for a time period and matching a certain criteria	NOTAM communication	Retrieve all TEMPDELTAs affecting the operational status of a navigation aid until the time of arrival.	Filtering of features and filtering the contained time slices.
6	Retrieve all PERMDELTAs of a feature valid for a time period	Replication	Retrieve all PERMDELTAs of a feature valid now or in the future.	Filtering of features and filtering the contained time slices, ignoring corrected and canceled time slices.
7	Retrieve specific time slices of a feature by their unique id	Replication, technical verification	Retrieve the PERMDELTA time slice of feature X, sequence number Y, correction number Z.	Filtering of features and filtering the contained time slices.
8	Retrieve the full history of a feature	Backup	Retrieve all time slices of a feature (BASELINEs, PERMDELTAs and TEMPDELTAs) valid at any point in time.	Filtering of features.

Four technical requirements of a query operation on AIXM 5 data follow from this analysis:

a) The ability to filter features and time slices.

b) A filter expression to easily identify corrected and canceled time slices[4].

c) A filter expression to easily identify the time slices that are relevant for a point in time.[5]

d) The ability to generate SNAPSHOT time slices and to filter them.

These requirements will form the basis for the extensions introduced in sections 8 and 9.

[3] For details on what BASELINEs are and how to compute them, see the AIXM-TM.

[4] Identifying corrected time slices is not trivial using FES 2.0. The maximum correction number of a sequence has to be calculated for this purpose, which would involve a complex XPath expression.

[5] This is a special case of a), with the option to omit possibly lengthy checks on validTime in the time slice filter (to select the time slices of interest). The difference to d) is that the response does not contain snapshot time slices but rather the actually stored ones.

6.2 Data Storage

Because AIXM-TM does not allow the permanent deletion of information[6], there are only two use cases:

a) insert a feature

b) insert a time slice into a feature

This is already possible with the existing **Transaction** operation. If a feature is inserted, the **Insert** action has to be used. If a time slice is to be inserted, the **Update** action is necessary, because technically, the feature is changed (updated) by adding a new time slice to its list of properties.

EXAMPLE Insertion of a time slice into a feature of type T, with a gml:identifier I. Please note that the order of the time slices is irrelevant in AIXM.

```
<wfs:Transaction>
  <wfs:Update typeName=”T”>
    <wfs:Property>
      <wfs:ValueReference action=”insertAfter”>
        aixm:timeSlice[1]
      </wfs:ValueReference>
      <wfs:Value>
        <aixm:timeSlice>
          ...
        </aixm:timeSlice>
      </wfs:Value>
    </wfs:Property>
    <fes:Filter>
      <fes:PropertyIsEqualTo>
        <fes:ValueReference>gml:identifier</fes:ValueReference>
        <fes:Literal>I</fes:Literal>
      </fes:PropertyIsEqualTo>
    </fes:Filter>
  </wfs:Update>
</wfs:Transaction>
```

7 Limitations of the WFS/FES 2.0 standards when applied to AIXM data

In this chapter we will show that there is only little support for the identified requirements in the WFS 2.0 and FES 2.0 standards.

The WFS 2.0 interface provides two operations for retrieving data:

- **GetFeature:** returns a response containing a selection of zero or more features corresponding to the criteria defined in the request

- **GetPropertyValue:** returns a response containing zero or more of the selected feature property values corresponding to query criteria defined in the request

[6] See 5.3 for details

There is also a **GetFeatureWithLock** operation, but this does not extend the power of the query language compared to **GetFeature** and is therefore ignored in the following. Both the **GetFeature** and **GetPropertyValue** operations support two types of query:

- **wfs:Query:** these are ad hoc queries generated by a client to retrieve specified feature types or property values

- **wfs:StoredQuery**: this is a pre-defined parameterized query that has been stored on the server for re-use by clients

In the following it is shown that neither of them fully supports the requirements identified in the previous section.

7.1 GetFeature

The **GetFeature** operation request is used to retrieve features using one or more ad hoc queries (**wfs:Query**) or stored queries (**wfs:StoredQuery**).

7.1.1 Ad Hoc Queries

The **wfs:Query** element consists of two parts: a selection and a projection part.

The selection part is a FES 2.0 filter expression and selects the features matching given criteria. An additional selection criterion is given by the **typeNames** attribute on the **wfs:Query** element: it is used to restrict the types of features to be returned.

The projection part is optional and may contain one or more projection clauses. In WFS 2.0, there is only one projection clause available: **wfs:PropertyName**. By default, non-mandatory properties of a feature are not returned in the response document unless they are referenced by a **wfs:PropertyName** element in the request. In WFS, the properties of a feature are made up of all first-level child elements of the feature. For AIXM 5 data these are the gml:identifier, gml:name, gml:description, gml:boundedBy, gml:featureMetadata and aixm:timeSlice elements. Except for the list of aixm:timeSlice elements, all of these are non-mandatory. Thus, the **wfs:PropertyName** clause enables the client to include gml:identifier, gml:name, gml:description, gml:boundedBy, gml:featureMetadata in the response which are excluded otherwise.

Obviously the **wfs:PropertyName** projection clause was designed for static feature data only, where all properties are encoded in the feature's first-level child elements. In the **DynamicFeatureQuery** class introduced in 9, we also introduce **PropertyExclusion**, which replaces **wfs:PropertyName** and is capable of handling dynamic feature properties as well.

Please note that the **wfs:GetFeature** operation does not provide any means to request a modified property or a modified list of properties according to criteria. Hence, there is support neither for the generation of SNAPHOTs nor the filtering of time slices. Only the use case “Retrieve the full history of a feature“ is supported.

7.1.2 Stored Queries

From the specification:

> “A stored query expression is a persistent, parameterized, identifiable query expression. A stored query can be repeatedly invoked using its identifier with different values bound to its parameters each time.” [1] 7.9.3.1

Stored queries can use any language to implement their behavior. The given parameters can be of any type. Hence, stored queries naturally have the power to support all AIXM 5 use cases[7]. Those queries could be predefined[8] for the server and tailored to the required use cases. Table 4 shows that 3 different queries would be required. That said, what are the reasons that this paper does not simply define stored queries but instead promotes the introduction of new types for WFS 2.0 and FES 2.0? They are:

- Demanding predefined stored queries is not very different from introducing a new query type: generic WFS 2.0 clients cannot support any of them because they cannot know about the semantics of the query parameters or elements.

- Using predefined stored queries would require the definition of the overall semantics involved in the execution of these queries. Basically, this would result in the definition of a new query language or in the extension of an already existing one. The latter is what this document does. By extending the **wfs:Query** and adding functionality tailored to the AIXM-TM, queries with Temporality Extension functionality can be executed in both ad hoc and stored queries.

- The intended use of stored queries is to parameterize complex queries to improve readability, speed up server processing (by query pre-compilation on server side) and reduce the amount of text in the request document. Typical parameters are concrete values or objects like numerical values in a certain unit of measurement, a spatial object for comparison, etc.
  For temporal queries, however, the parameters include complex filters, which are used to define the query itself, and because of which the query cannot be pre-compiled.

[7] The specification says ‘A stored query expression may be used in a [,,,] GetFeature […] operation to identify a set of features to be operated upon’ ([1], section 7.9.3.1). Strictly speaking, this implies that no transformation of features can take place and hence no time slice selection or snapshot generation is possible. However, we are not sure if this was the intention of the standard and assume for the following reasoning that the result of a stored query can be an arbitrary set of features, new ones or modified ones.

[8] Predefined in the same manner as the required ‘GetFeatureById’ stored query.

Table 4 — Required Query Functionality

Description	Parameters	Return Value
Query SNAPSHOTs	- a list of feature types - a feature level filter (optional) - a time instant or time period - a time slice level filter (optional)	Collection of features containing generated (and optionally filtered) snapshot time slices
Query time slices	- a list of feature types - a feature level filter (optional) - a time instant or time period (optional)[9] - a time slice level filter (optional) - a marker (flag) whether to ignore corrected time slices or not - a marker (flag) whether to ignore canceled time slices or not[10]	Collection of features containing all or a subset of time slices

7.2 GetPropertyValue

From the specification:

> "The GetPropertyValue operation allows the value of a feature property or part of the value of a complex feature property to be retrieved from the data store for a set of features identified using a query expression." [1] 10.1

In other words, **GetPropertyValue** is a combination of a **GetFeature** request and an XPath expression. A given filter is used to select features by criteria (this is the regular **GetFeature** operation), and in a subsequent process the XPath expression is used to extract XML elements out of the result.

One of the requirements is to generate and filter SNAPSHOT time slices. This requirement cannot be met by the **GetPropertyValue** operation, because the XPath expression is only capable of identifying a feature property (or any of its sub-properties)[11] (for the discussion to use stored queries for this, see above). Another central requirement is the capability to filter the list of time slice properties contained in the feature. This is possible with **GetPropertyValue**. However, significant disadvantages are observed:

[9] Avoids a lengthy check on the valid time in the time slice filter.

[10] The two markers are technically time slice filters, but cannot be realized with the subset of XPath syntax defined for FES.

[11] Whereas in theory, custom XPath functions could be introduced to transform a list of time slices into SNAPSHOT time slices, this would be clearly against the semantics of the operation as defined by the WFS specification, which are to retrieve "the value of a feature property or part of a complex feature property". Snapshots, on the other hand, are not part of the persistent set of time slices of a dynamic feature, they are generated (or derived).

- The filter criteria for time slices are likely to be quite complex. Spatial and temporal criteria are required for almost every common use case. The existing FES operators cannot be used for this purpose, because they are not available as XPath functions. Further on, custom XPath functions have to be introduced to make the exclusion of corrected and canceled time slices possible.

- The XPath expression has to be encoded in a single string. This string cannot be validated through an XML schema. The expression will be very hard to read (and debug) for humans, especially if complex spatio-temporal functions are involved.

- The result of a **GetPropertyValue** operation is a collection of XML elements. In our case, these are timeSlice elements. A timeSlice element does not contain a reference to its parent feature. In addition, the **GetPropertyValue** operation response does not indicate which property values contained in the response belong to which features. Together this means that if property values from multiple features are contained in the result, the result is ambiguous.

7.3 Summary

This chapter showed that WFS 2.0 and FES 2.0 were designed to support static feature data, following the General Feature Model. They do not support the use cases for querying AIXM 5 data. Predefined stored queries can work around this limitation. In chapters 8.28 and 9 we will introduce extensions to FES 2.0 and WFS 2.0 as a general solution that works with ad hoc and stored queries.

8 Filter Encoding Specification Temporality Extension

The following requirements classes extend the Filter Encoding Specification by filter constraints based on time.

8.1 Terms and Definitions

In addition to the definitions from 5.3 and 5.4 we extend the concept of SNAPSHOT time slices to time periods and introduce a dummy time slice to work around schema validation issues.

8.1.1 Dummy Time Slice

The projection and transformation clauses introduced may produce features without time slices in some cases. As this is not schema-valid, and we cannot redefine or relax the schema of the features returned by the service, we need to introduce a *dummy time slice* to work around this limitation. A dummy time slice is structured as follows:

```
<aixm:timeSlice>
  <aixm:<FeatureType>TimeSlice>
  <gml:validTime nilReason=”inapplicable”>
  <aixm:interpretation “<any valid value>”>
</aixm:timeSlice>
```

The dummy time slice is similar to a cancelation time slice, but is distinguishable from it by the missing sequence and correction number.

8.1.2 Set of Snapshot Time Slices for Time Periods

A feature's minimum set of SNAPSHOT time slices for a time period *t* is a list of SNAPSHOT time slices spanning *t*, which is calculated according to the AIXM-TM and by the following rules:

- if there are no property changes in *t*, the set consists of a single snapshot time slice with a **validTime** equal to *t*. The other properties of that time slice are calculated according to AIXM-TM with a (any) time instant out of *t*.
- if there are property changes, the set consists of consecutive snapshot time slices that span *t*, ordered by time in ascending order, with disjoint **validTime** properties. The other properties for each of those time slices are calculated according to AIXM-TM with a (any) time instant out of that time slice's valid time. See Figure 4 for an illustration.
- if *t* is empty, the set consists of a single dummy time slice as defined in 8.1.1.

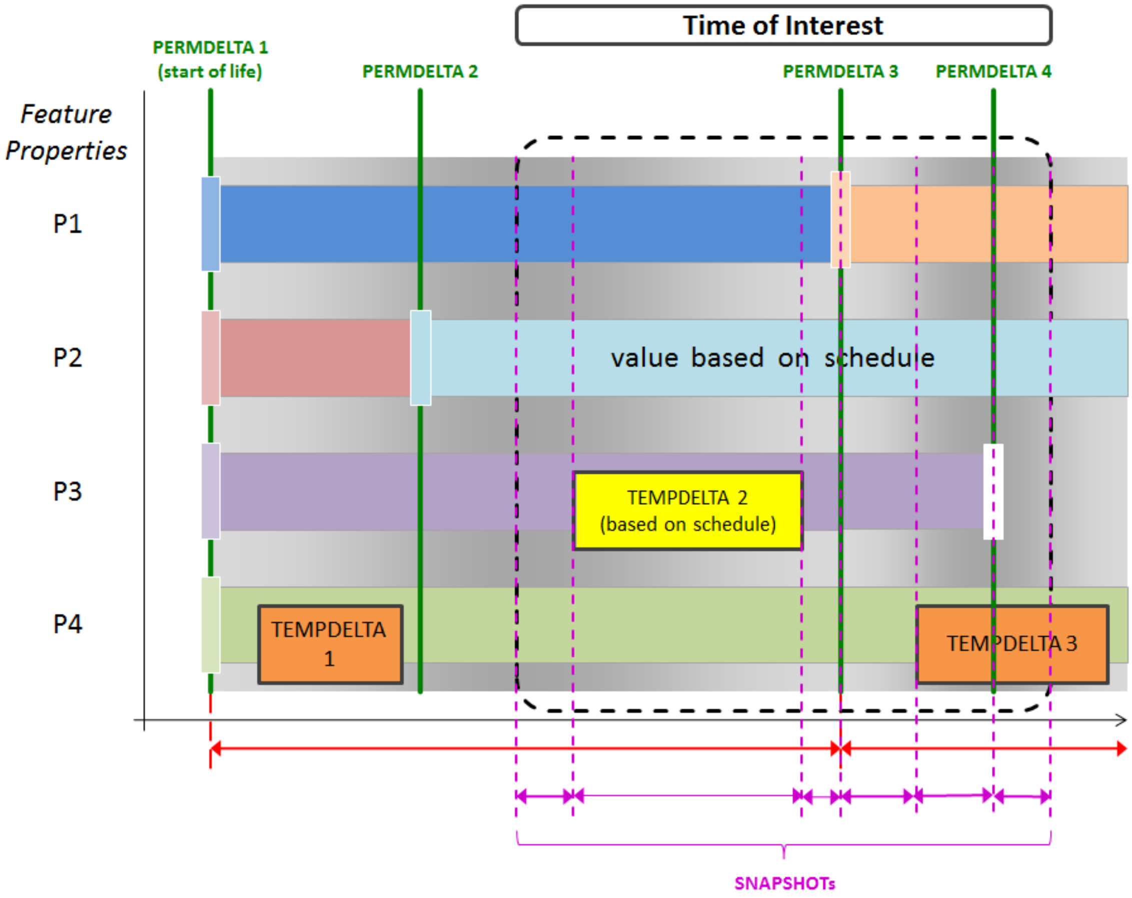


Figure 4: Creating snapshots for a time period of interest without evaluation of schedules

Figure 4 shows how a service would create snapshots of a dynamic feature for a time period of interest, without evaluation of schedules. Scheduled values are considered as ordinary, static properties with a fixed value throughout the valid time of the respective time slice. Six snapshots are created to cover the time of interest, based upon the following sequence of value changes:

1. TEMPDELTA 2 (P3, start of temporary change),
2. TEMPDELTA 2 (P3, end of temporary change),
3. PERMDELTA 3 (P1),
4. TEMPDELTA 3 (P4, start of temporary change),
5. PERMDELTA 4 (P3)

8.2 Requirements Class: FES-TE Core

Requirements Class
http://www.opengis.net/spec/fes-te/1.0/req/core

Target Type	Service
Dependency	FES ad hoc query
Requirement	**/req/core/capability** List the identifier of the corresponding conformance class of this requirements class as an ows:Profile element in the ServiceIdentification section of the Capabilities document. See also 10.
Requirement	**/req/core/dynamic-feature-filter** Support the fes-te:DynamicFeatureFilter element
Requirement	**/req/core/dynamic-feature-filter-on-aixm-features** Correctly apply the fes-te:DynamicFeatureFilter element on AIXM features
Requirement	**/req/core/dynamic-feature-filter-on-static-features** Correctly apply the fes-te:DynamicFeatureFilter element on static features
Requirement	**/req/core/dynamic-feature-filter-with-time-instant-time-indicator** Correctly apply the fes-te:DynamicFeatureFilter with a time instant set as time indicator
Requirement	**/req/core/dynamic-feature-filter-with-time-period-time-indicator** Correctly apply the fes-te:DynamicFeatureFilter element with a time period set as time indicator
Requirement	**/req/core/dynamic-feature-filter-with-evaluate-schedules-true** Raise an ‘OptionNotSupported’ exception if the evaluateSchedules attribute is set to true unless redefined

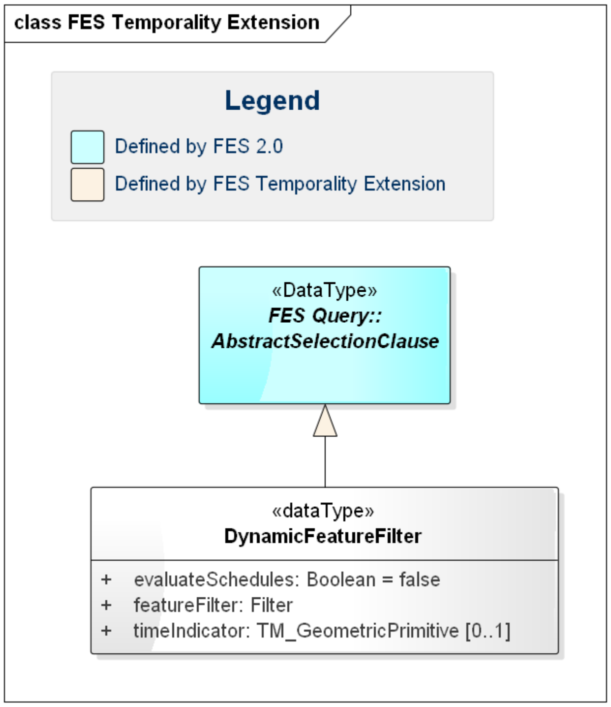

class FES Temporality Extension
Legend
Defined by FES 2.0
Defined by FES Temporality Extension
«DataType»
FES Query::
AbstractSelectionClause
«dataType»
DynamicFeatureFilter
+ evaluateSchedules: Boolean = false
+ featureFilter: Filter
+ timeIndicator: TM_GeometricPrimitive [0..1]

Figure 5: UML class diagram for DynamicFeatureFilter

This requirements class defines the **DynamicFeatureFilter** data type. It is a selection clause capable of evaluating filter constraints for a given point in time or a time period. It consists of a **featureFilter** and an optional **timeIndicator.**

If a **timeIndicator** is present, it is of type **GML::TM_GeometricPrimitive**, which can either represent a point in time or a time period. It defines the point in time or the time period when the given **featureFilter** is to be evaluated. In other words, the state of the feature at the given point in time or time period is calculated according to the AIXM-TM and the filter constraints are applied on the result[12]. The evaluation of properties with schedules is only supported if the according conformance class is supported (see 2.1). Otherwise properties with schedules are treated as any other complex property.

If a time period is used as a **timeIndicator,** the filter is evaluated for all possible states of the feature, i.e. for all points in the feature lifetime that are not before and not after the given time period.

Please note that the **featureFilter** is a filter on feature level. When used in a WFS Query and no projection clause is in place, the complete feature including the full history of time slices is returned.

[12] This equals an implicit application of the "evaluateDuring" function introduced in [4].

8.2.1 Requirement: /req/core/dynamic-feature-filter-on-aixm-features

If **DynamicFeatureFilter** is applied on dynamic feature types based on the AIXM-TM, the contained **featureFilter** property is applied on the features evaluated according to the AIXM-TM. That means *canceled* and *corrected* information is ignored. See terms and definitions for the definition of corrected and canceled information in AIXM-TM.

See examples B.1, B.2 and B.3.

8.2.2 Requirement: /req/core/dynamic-feature-filter-on-static-features

If **DynamicFeatureFilter** is applied on other feature types (static or dynamic but not based on AIXM-TM), the contained **featureFilter** is evaluated as defined by the FES.[13]

8.2.3 Requirement: /req/core/dynamic-feature-filter-with-time-instant-time-indicator

If a **timeIndicator** property is given and it is a time instant, the **featureFilter** is applied to the feature state at that point in time. The feature state is calculated as defined by the AIXM-TM algorithm for SNAPSHOT time slices. That means that the service creates an internal representation as if the generation of a SNAPSHOT time slice was requested (see 9.3). This internal representation is the target of the **featureFilter**[14]. If the feature does not exist at that point in time for any reason (e.g. because it is not yet commissioned, already decommissioned, or has its commissioning canceled) it is not selected.

Pseudo algorithm:

```
feature_set = [empty set]; t = timeIndicator
 for every feature f of the service's data store do
      if isDefined(f, t) then feature_set += feature_with_snapshot(f, t)
 return { original_feature(f) | f element of wfsFilter(feature_set, featureFilter) }
```

Where:

Function	Definition
isDefined(f, t)	true when feature *f* exists at time *t* and false otherwise
feature_with_snapshot(f,	returns *f* containing the calculated SNAPSHOT of *f* at time *t*

[13] It would be nicer to throw an error if applied on other dynamic feature types, but that would require the service to identify the nature of the feature type, which would be too hard for services that allow the dynamic creation of feature types at runtime.

[14] This means time slice properties are not first level properties and need to be referenced by an XPath expression following the pattern "aixm:timeSlice/*/[time slice property]". The motivation for this is that XPath expressions valid for actual SNAPSHOT time slices should also be valid for filtering.

t)	as its only time slice
wfsFilter(feature_set, filter)	returns the subset of *feature_set* that matches *filter* according to WFS 2.0
original_feature(f)	returns the feature with the same gml:identifier as *f* from the data store of the service, i.e. corresponding original feature

Informative:

- The method *original_feature(f)* is applied to emphasize that the return value of an AbstractSelectionClause is a subset of the original features of the data store
- if **evaluateSchedules** is set to false (the default), no processing of properties with schedules takes place, i.e. they are handled as any other property.

Example: see B.4.

8.2.4 Requirement: /req/core/dynamic-feature-filter-with-time-period-time-indicator

If a time period is used as a **timeIndicator,** the filter is evaluated for all possible states of the feature, i.e. for all points in time in the feature's *lifetime* that are not before and not after the given time period. The feature state(s) are calculated as defined by the AIXM-TM algorithm for SNAPSHOT time slices and internally represented by a list of SNAPSHOT time slices (see 8.2.3 for the definition of an *internal* time slice, and the consequences). If the feature does not exist in that period of time for any reason (e.g. because it is not yet commissioned, already decommissioned, or has its commissioning canceled) it is not selected.

Pseudo algorithm:

```
feature_set = [empty set]; t = timeIndicator
for every feature f of the service's data store do
    if isDefined(f, t) then feature_set += feature_with_snapshots(f, t)
return { original_feature(f)  | f element of wfsFilter(feature_set, featureFilter) }
```

Where:

Function	Definition
isDefined(f, t)	true when the intersection of the feature's *lifetime* and t is not empty and false otherwise
feature_with_snapshots(f,	returns *f* containing the calculated minimal set of

t)	SNAPSHOT time slices of *f* in time period *t* as defined in 8.1.29.3.2
wfsFilter(feature_set, filter)	returns the subset of *feature_set* that matches *filter* according to WFS 2.0
original_feature(f)	returns the feature with the same gml:identifier as *f* from the data store of the service, i.e. corresponding original feature

Due to the fact that multiple SNAPSHOT time slices exist if there are properties that change in the given time period, filter expressions targeting single-valued properties may encounter multiple values spread over several time slices, and filter expressions targeting multi-valued properties may even encounter multiple time slices containing multiple values of that property. Those cases are handled in the usual manner for collections of values, and the **matchAction** property in the filter operator applies (see [2], section 7.7.2.1.2).

Please note that there is no **matchAction** property for the operators **fes:PropertyIsLike** and **fes:PropertyIsNil**. For maximum expressiveness, this should be changed in a future version of WFS. Until this is the case, **matchAction** "Any" shall apply here.

Informative:

- the method *original_feature(f)* is applied to emphasize that the return value of an AbstractSelectionClause is a subset of the original features of the data store
- if **evaluateSchedules** is set to false (the default), no processing of properties with schedules takes place, i.e. they are handled as any other property.

Example: see B.5.

8.2.5 Requirement: /req/core/dynamic-feature-filter-with-evaluate-schedules-true

Unless a derived requirements class supports it and the service passes the corresponding conformance test class, raise an 'OptionNotSupported' exception if the **evaluateSchedules** property is set to true.

8.3 Requirements Class: FES-TE Time Instant Schedule Evaluation For Filter

Requirements Class	
http://www.opengis.net/spec/fes-te/1.0/req/time-instant-schedule-evaluation-filter	
Target Type	Service

Dependency	FES-TE Core
Requirement	**/req/time-instant-schedule-evaluation-filter/capability** List the identifier of the corresponding conformance class of this requirements class as an ows:Profile element in the ServiceIdentification section of the Capabilities document. See also 10.
Requirement	**/req/time-instant-schedule-evaluation-filter/evaluate-schedules** Support property **evaluateSchedules**=true if time indicator is a time instant.

8.3.1 Requirement: /req/time-instant-schedule-evaluation-filter/evaluate-schedules

If a **DynamicFeatureFilter** is applied on an AIXM feature type, with a **timeIndicator** set to a time instant and property **evaluateSchedules** set to true, then evaluate all properties with schedules such that all non-matching instances of the multi-occuring property in the time slices are ignored when the filter predicates are applied. Non-matching schedule properties are those whose **timeInterval** element does not contain a **TimeSheet** that temporally intersects (contains) the time instant of the **DynamicFeatureFilter**.

Please note that **TimeSheet** supports time values based on varying natural phenomena such as sunrise and sunset, or holidays. Both depend on the location of the feature as well.

Example: see A.2, B.6.

8.4 Requirements Class: FES-TE TimePeriodScheduleEvaluationForFilter

Requirements Class	
http://www.opengis.net/spec/fes-te/1.0/req/time-period-schedule-evaluation-filter	
Target Type	Service
Dependency	FES-TE Time Instant Schedule Evaluation For Filter
Requirement	**/req/time-period-schedule-evaluation-filter/capability** List the identifier of the corresponding conformance class of this requirements class as an ows:Profile element in the ServiceIdentification section of the Capabilities document. See also 10.

Requirement	**/req/time-period-schedule-evaluation-filter/evaluate-schedules** Support attribute **evaluateSchedules**=true if time indicator is a time period.

8.4.1 Requirement: /req/time-period-schedule-evaluation-filter/evaluate-schedules

If a **DynamicFeatureFilter** is applied on an AIXM feature type, with a **timeIndicator** set to a time period and attribute **evaluateSchedules** set to true, then evaluate all properties with schedules such that all non-matching instances of the multi-occuring property in the time slices are ignored when the filter predicates are applied. Thus, if there is no match, all property instances are removed. Non-matching schedule properties are those whose **aixm:timeInterval** element does not contain a **TimeSheet** that temporally intersects the time period of the **DynamicFeatureFilter**.

Please note that **TimeSheet** supports time values based on varying natural phenomena such as sunrise and sunset, or holidays. Both depend on the location of the feature as well.

Example: see B.7.

9 Web Feature Service Temporality Extension

Following the requirements identified in Table 4 and in section 6.1, a new type of query is introduced here: a query for dynamic features. This query is tailored for retrieving data based on the AIXM 5 Temporality Model[15].

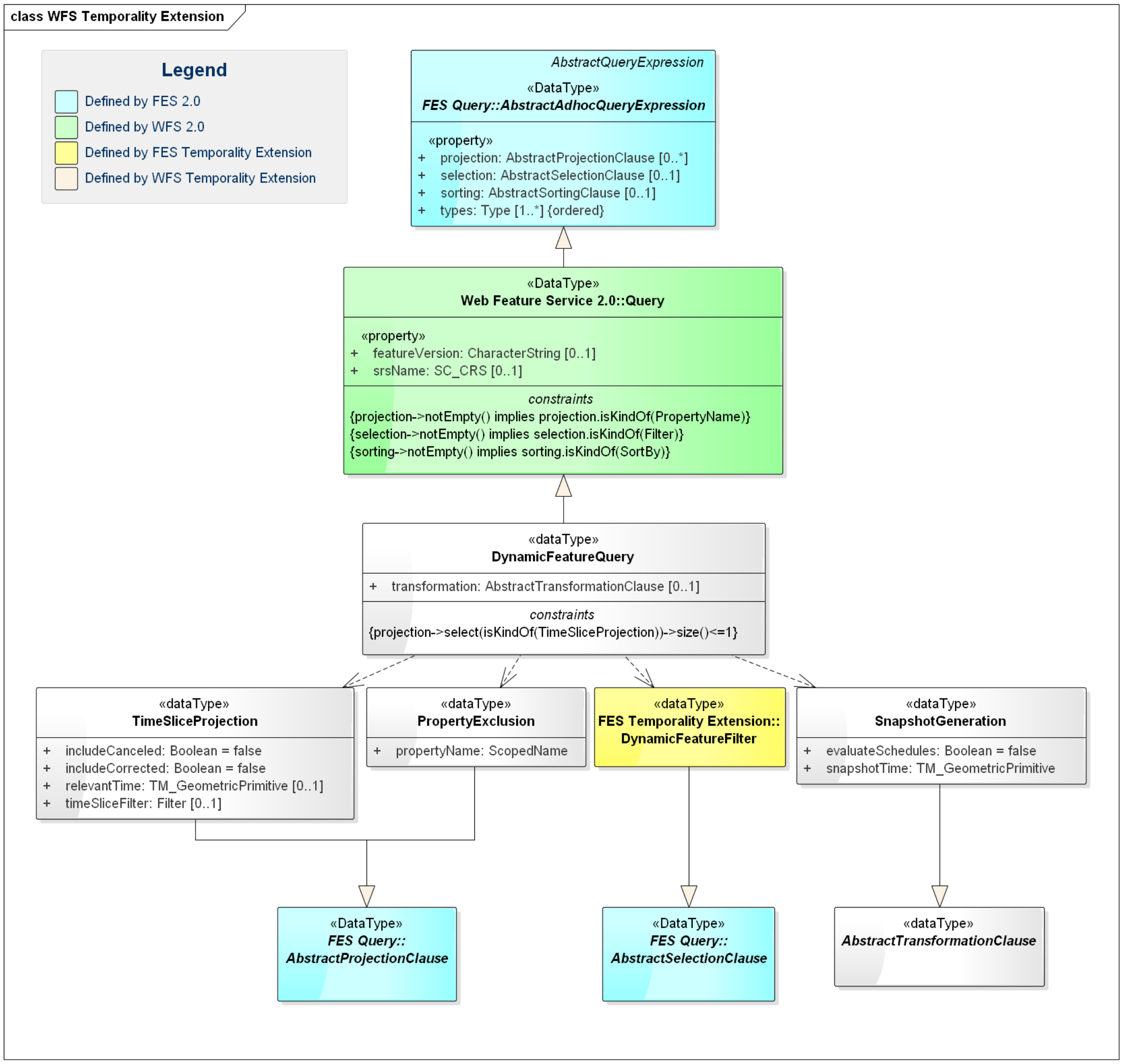
The extension is comprised of two new projection clauses, one for filtering time slices and one for excluding properties, and the introduction of transformation clauses. A transformation clause is defined that enables clients to request the state or states of a feature for a point or period in time, respectively.

9.1 Requirements Class: WFS-TE Core

Requirements Class
http://www.opengis.net/spec/wfs-te/1.0/req/core

[15] There was no experience with real-world domains applying the general GML dynamic feature model at the time of writing this document. However, AIXM-TM is general enough to form the base of any data model required to deal with temporality aspects, and not bound to the aeronautical world. It can become a domain independent concept, standardized by the relevant organizations. See also [OGC 11-093r2].

Target Type	Service
Dependency	WFS Basic
Dependency	FES-TE Core
Requirement	**/req/core/capability** List the identifier of the corresponding conformance class of this requirements class as an ows:Profile element in the ServiceIdentification section of the Capabilities document. See also 10.
Requirement	**/req/core/dynamic-feature-query** Support for the wfs-te:DynamicFeatureQuery element
Requirement	**/req/core/restrict-number-of-time-slice-projections** At most one **TimeSliceProjection** projection clause may be present on a **DynamicFeatureQuery** element. Raise an ‘InvalidParameterValue’ exception if there are multiple.
Requirement	**/req/core/include-optional-properties** Unless restricted by a projection clause, optional properties are always included in the response.
Requirement	**/req/core/transformation-clause** Support for the wfs-te:DynamicFeatureQuery/wfs-te:transformation element


class WFS Temporality Extension
Legend
Defined by FES 2.0
Defined by WFS 2.0
Defined by FES Temporality Extension
Defined by WFS Temporality Extension
AbstractQueryExpression
«DataType»
FES Query::AbstractAdhocQueryExpression
«property»
+ projection: AbstractProjectionClause [0..*]
+ selection: AbstractSelectionClause [0..1]
+ sorting: AbstractSortingClause [0..1]
+ types: Type [1..*] {ordered}
«DataType»
Web Feature Service 2.0::Query
«property»
+ featureVersion: CharacterString [0..1]
+ srsName: SC_CRS [0..1]
constraints
{projection->notEmpty() implies projection.isKindOf(PropertyName)}
{selection->notEmpty() implies selection.isKindOf(Filter)}
{sorting->notEmpty() implies sorting.isKindOf(SortBy)}
«dataType»
DynamicFeatureQuery
+ transformation: AbstractTransformationClause [0..1]
constraints
{projection->select(isKindOf(TimeSliceProjection))->size()<=1}
«dataType»
TimeSliceProjection
+ includeCanceled: Boolean = false
+ includeCorrected: Boolean = false
+ relevantTime: TM_GeometricPrimitive [0..1]
+ timeSliceFilter: Filter [0..1]
«dataType»
PropertyExclusion
+ propertyName: ScopedName
«dataType»
FES Temporality Extension::
DynamicFeatureFilter
«dataType»
SnapshotGeneration
+ evaluateSchedules: Boolean = false
+ snapshotTime: TM_GeometricPrimitive
«DataType»
FES Query::
AbstractProjectionClause
«DataType»
FES Query::
AbstractSelectionClause
«dataType»
AbstractTransformationClause

The **DynamicFeatureQuery** is in the *substitution group* of the **WFS::Query** type. Consequently, a dynamic feature query can be used in all query operations of a WFS: **GetFeature**, **GetPropertyValue**, **GetFeatureWithLock**, **LockFeature**, and also stored queries. A dynamic feature query should target a feature type based on AIXM, but this is not required (see 8.2.1 and 8.2.2 for details). The output of a dynamic feature query is a regular collection of dynamic features. The contained time slices, however,

- may represent a subset of the time slices of the dynamic feature or
- are generated in case of retrieving the complete state of the dynamic feature for a given point in time (snapshot query).

The **DynamicFeatureQuery** extends **WFS::Query** without any modification or extension. This is required to overcome and redefine constraints on applicable selection and projection clauses defined for **WFS::Query** in the underlying model (see OGC 09-025r1 figure 8).

A **FES::Filter** is still supported to select features without using time-based constraints, i.e. where the state of a feature at a point or period in time is not needed. The WFS 2.0 specification fully applies. An example is feature selection using the **gml:identifier** (see A.1 for an example).

9.1.1 Requirement: /req/core/include-optional-properties

By default optional, i.e. non-mandatory feature properties are to be included in the response. For multi-occurring properties that means that they are to be included completely. This requirement overrides clause 7.9.2.4.5.1 in [1].

Motivation: due to their time-variant nature, technically almost all properties of a dynamic feature following the AIXM-TM are optional from an XML schema point of view.

The behavior can be changed by using the **PropertyExclusion** projection clause (see 9.6).

Please note that this requirement renders **WFS::PropertyName** noneffective if used in a **DynamicFeatureQuery**.

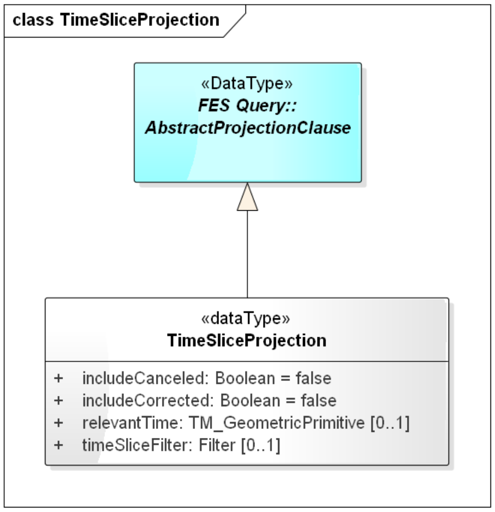
9.1.2 Requirement: /req/core/transforming-clause

If present, apply the **AbstractTransformationClause** in a **DynamicFeatureQuery** on each feature of the result set, after **AbstractProjectionClause**(s) is (are) applied and replace the result. Informative: see also 13.2.

9.2 Requirements Class: WFS-TE Time Slice Projection

Requirements Class	
http://www.opengis.net/spec/wfs-te/1.0/req/time-slice-projection	
Target Type	Service
Dependency	WFS-TE Core
Requirement	**/req/time-slice-projection/capability** List the identifier of the corresponding conformance class of this requirements class as an ows:Profile element in the ServiceIdentification section of the Capabilities document. See also 10.

Requirement	**/req/time-slice-projection/time-slice-filter** Support **TimeSliceProjection/timesliceFilter**
Requirement	**/req/time-slice-projection/relevant-time** Support **TimeSliceProjection/relevantTime**
Requirement	**/req/time-slice-projection/include-corrected** Support **TimeSliceProjection/includeCorrected**
Requirement	**/req/time-slice-projection/include-canceled** Support **TimeSliceProjection/includeCanceled**


class TimeSliceProjection
«DataType»
FES Query::
AbstractProjectionClause
«dataType»
TimeSliceProjection
+ includeCanceled: Boolean = false
+ includeCorrected: Boolean = false
+ relevantTime: TM_GeometricPrimitive [0..1]
+ timeSliceFilter: Filter [0..1]

The **TimeSliceProjection** class is a projection clause that enables the client to filter the returned list of time slices by a supplied **FES::Filter** and some other constraints.

9.2.1 Requirement: /req/time-slice-projection/time-slice-filter

If a **timeSliceFilter** property is present, for each feature in the result, apply the contained **FES::Filter** on the list of time slices, i.e. remove all time slice elements that do not match the filter.

The context node for the **timeSliceFilter** is the time slice of the feature. XPath expressions referencing time slice properties have to honor that.

EXAMPLE This reduces an XPath of

```
aixm:timeSlice/*/aixm:someProperty
```

to

```
aixm:someProperty
```

If no time slice matches the filter, technically a schema-invalid document has to be returned. To avoid this, the service shall place a dummy time slice in the affected feature(s) as defined in 8.1.1.

Example: see A.4, A.5, A.6, A.7, B.8.

9.2.2 Requirement: /req/time-slice-projection/relevant-time

The optional **relevantTime** element serves as a shortcut to an otherwise lengthy check on the **validTime** property of the time slices. If it is present, only time slices that are relevant for (not before and not after) a given time period or time instant shall be kept in the result, i.e. all time slices that define the state for that time.

Example: see A.3, A.4.

9.2.3 Requirements: /req/time-slice-projection/include-corrected and /req/time-slice-projection/include-canceled

By default, *canceled* and *corrected* time slices, as defined in 5.3, are not part of the returned set of time slices. This can be changed by setting the properties **includeCanceled** and **includeCorrected** to true, respectively. In the special case of setting **includeCorrected** to true, and setting **includeCanceled** to false, canceled time slices shall not be returned.

Example: see A.7.

9.3 Requirements Class: WFS-TE Snapshot Generation

Requirements Class	
http://www.opengis.net/spec/wfs-te/1.0/req/snapshot-generation	
Target Type	Service
Dependency	WFS-TE Core
Requirement	**/req/snapshot-generation/capability** List the identifier of the corresponding conformance class of this requirements class as an ows:Profile element in the ServiceIdentification section of the Capabilities document. See also 10.
Requirement	**/req/snapshot-generation/time-instant**

	Support the **SnapshotGeneration** clause for time instants
Requirement	**/req/snapshot-generation/time-period** Support the **SnapshotGeneration** clause for time periods
Requirement	**/req/snapshot-generation/evaluate-schedules** Raise an ‘OptionNotSupported’ exception if the **evaluateSchedules** attribute is set to true unless redefined

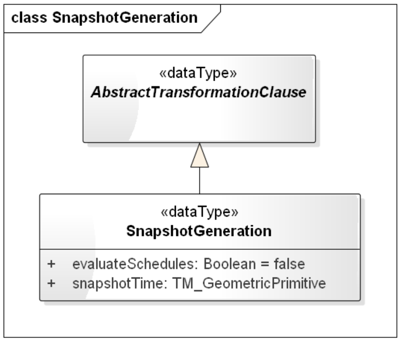

class SnapshotGeneration
«dataType»
AbstractTransformationClause
«dataType»
SnapshotGeneration
+ evaluateSchedules: Boolean = false
+ snapshotTime: TM_GeometricPrimitive

Figure 6: UML class diagram for SnapshotGeneration

The **SnapshotGeneration** class is a transformation clause that generates snapshot time slices according to AIXM-TM for the selected features. If this projection clause is present in a request, time slices with SNAPSHOT interpretation are generated for the selected features for the given time. They replace the list (history) of time slices.

The optional property **evaluateSchedules** triggers the evaluation of properties with schedule for the given time. That means that the collection of properties is replaced by a single property that contains the values valid at the given time (see A.9 for an example). Schedule evaluation for time periods requires the service to drill down into the details of each schedule.

As stated in 9.1.1, projection clauses are applied before transformation clauses. If a **TimeSliceProjection** is combined with **SnapshotGeneration**, this allows what-if-queries like ‘what would be the current state (or the state at time X) of the feature if time slice Y was not among the history of time slices?’. However, this flexibility comes at a price: with time slice filters in place, the history of time slices may become incomplete or even invalid (e.g. when regarding inter-feature references and associations). It is up to the client to formulate meaningful queries. The service is not required to perform a complete validation of the reduced time slice list and affected features. If it cannot produce a valid state, it should produce an invalid snapshot as defined by the *invalidSnapshot* function in 9.3.1.

9.3.1 Requirement /req/snapshot-generation/time-instant

For each feature in the result set, the feature's list of time slices is replaced by a snapshot time slice according to the AIXM-TM at the given time instant (**snapshotTime**).

Pseudo algorithm:

```
result = [empty set]; t = timeIndicator
for every feature f of the previous operation’s output (filtering and projecting) do
    if isDefined(f, t) then result += snapshot(f, t)
    else result +=dummySnapshot(f)
return result
```

Where:

Function	Definition
isDefined(f, t)	true when feature *f* exists and is valid at time *t* and false otherwise. A feature is valid at time *t* if its history of time slices is complete to determine its state at *t* as defined by AIXM-TM.
snapshot(f, t)	returns *f* containing the calculated SNAPSHOT of *f* at time *t* as its only time slice.
dummySnapshot(f)	returns *f* containing a single dummy SNAPSHOT time slice as defined in 8.1.1.

Examples: see A.1, A.2, B.9.

9.3.2 Requirement /req/snapshot-generation/time-period

For each feature in the result set, let *t* be the intersection of the requested time period **snapshotTime** and the feature’s *lifetime*.

Pseudo algorithm: equal to that of 9.3.1, except that function *snapshot(f, t)* is redefined for *t* being a time period, and returns the minimal set of snapshot time slices for *t* (see 8.1.2).

Example: B.10.

9.3.3 Requirement /req/snapshot-generation/evaluate-schedules

If **evaluateSchedules** is false (default), properties with schedules are passed through "as is", i.e. they are handled as any other multi-value property and the complete schedule is in the time slice.

Unless a derived requirements class supports it and the service passes the corresponding conformance test class, raise an ‘OptionNotSupported’ exception if the **evaluateSchedules** property is set to true.

9.4 Requirements Class: WFS-TE Time Instant Schedule Evaluation For Snapshots

Requirements Class	
http://www.opengis.net/spec/wfs-te/1.0/req/time-instant-schedule-evaluation-snapshots	
Target Type	Service
Dependency	WFS-TE Snapshot Generation
Requirement	**/req/time-instant-schedule-evaluation-snapshots/capability** List the identifier of the corresponding conformance class of this requirements class as an ows:Profile element in the ServiceIdentification section of the Capabilities document. See also 10.
Requirement	**/req/time-instant-schedule-evaluation-snapshots/evaluate-schedules** Support attribute evaluateSchedules=true if the snapshotTime is a time instant.

9.4.1 Requirement /req/time-instant-schedule-evaluation-snapshots/evaluate-schedules

If **evaluateSchedules** is true on a **SnapshotGeneration** and **snapshotTime** is a time instant: evaluate all properties with schedule and remove all the non-matching instances of the multi-occuring property in the time slices of the result set.

Non-matching schedule properties are those whose **timeInterval** property does not contain a **TimeSheet** that temporally intersects (contains) the time instant given as **snapshotTime**.

Please note that **TimeSheet** supports time values based on varying natural phenomena such as sunrise and sunset, or holidays. Both depend on the location of the feature as well.

Example: see A.9, B.11.

9.5 Requirements Class: WFS-TE Time Period Schedule Evaluation For Snapshots

Requirements Class	
http://www.opengis.net/spec/wfs-te/1.0/req/time-period-schedule-evaluation-snapshots	
Target Type	Service
Dependency	WFS-TE Time Instant Schedule Evaluation For Snapshots
Requirement	**/req/time-instant-schedule-evaluation-filter/capability** List the identifier of the corresponding conformance class of this requirements class as an ows:Profile element in the ServiceIdentification section of the Capabilities document. See also 10.
Requirement	**/req/time-instant-schedule-evaluation-filter/evaluate-schedules** Support attribute **evaluateSchedules**=true if **snapshotTime** is a time period.

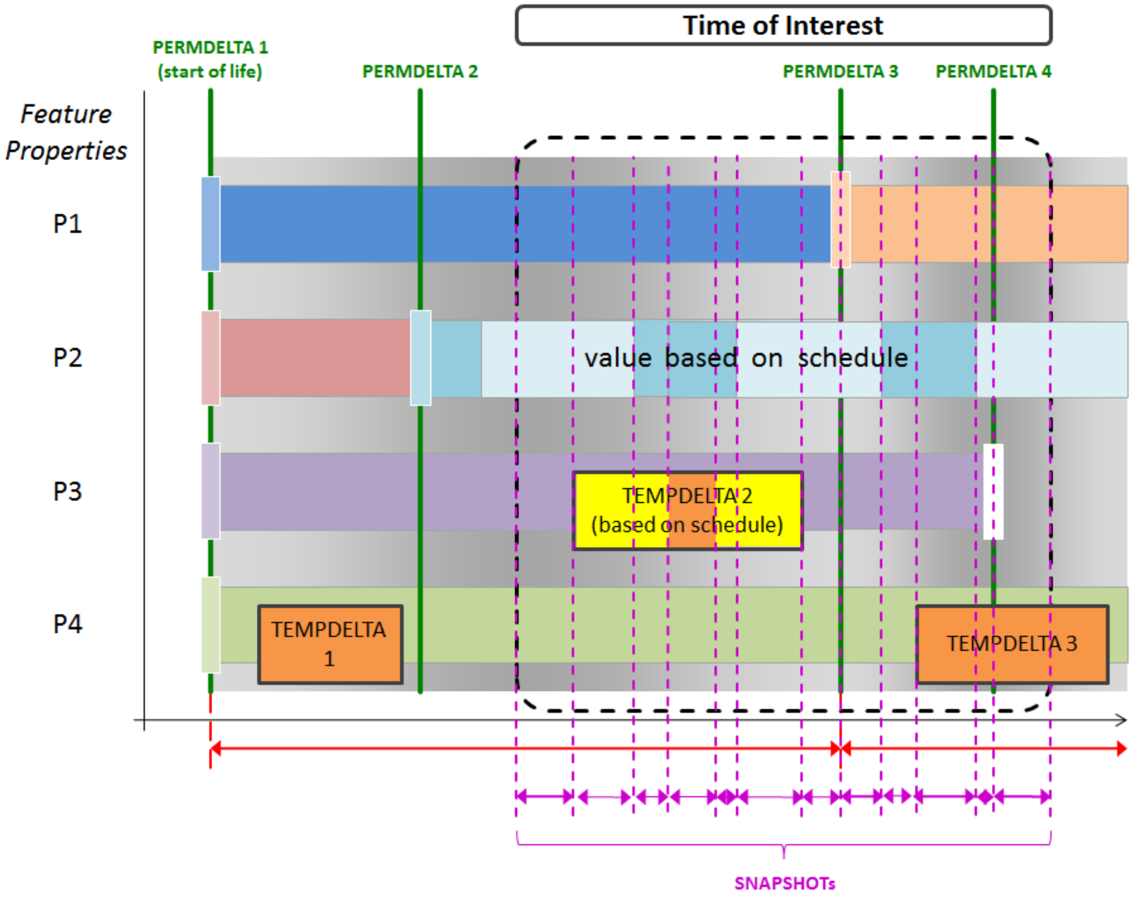


Figure 7: Creating snapshots for a time period of interest with evaluation of schedules

If **evaluateSchedules** is true in **SnapshotGeneration** and **snapshotTime** is a time period: after creating the list of SNAPSHOT time slices, evaluate all properties with schedules contained in them and remove all the non-matching instances. If after that SNAPSHOT time slices with multi-occurring properties with schedules exist, split them up by creating equivalent SNAPSHOT time slices such that no multi-occurring properties with schedules exist anymore.

Example: Figure 7 illustrates how twelve snapshots are created to cover the time of interest, based upon the following sequence of value changes:

1. TEMPDELTA 2 (P3, start of temporary change),
2. Scheduled value change (P2),
3. Scheduled value change (P3),
4. Scheduled value change (P3),
5. Scheduled value change (P2),
6. TEMPDELTA 2 (P3, end of temporary change),

7. PERMDELTA 3 (P1),
8. Scheduled value change (P2),
9. TEMPDELTA 3 (P4),
10. Scheduled value change (P2),
11. PERMDELTA 4 (P3)

As schedule evaluation for time periods might result in a great number of time slices for frequently changing properties and long time periods, support for this is considered an advanced feature of the service. In case the number grows beyond a number the service can handle, or is getting infinite (in cases where the requested time period end is indeterminate), the service may[16] raise an exception.

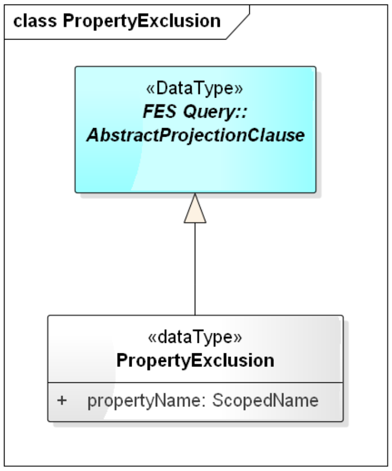
Please note that **TimeSheet** supports time values based on varying natural phenomena such as sunrise and sunset, or holidays. Both depend on the location of the feature as well.

Example: see B.12

9.6 Requirements Class: WFS-TE Property Exclusion

Requirements Class	
http://www.opengis.net/spec/wfs-te/1.0/req/property-exclusion	
Target Type	Service
Dependency	WFS-TE Core
Requirement	**/req/property-exclusion/capability** List the identifier of the corresponding conformance class of this requirements class as an ows:Profile element in the ServiceIdentification section of the Capabilities document. See also 10.
Requirement	**/req/property-exclusion/property-name** Filter top-level or time slice properties.

[16] "May" is used intentionally here, because that is easier to implement for service providers. If the service had to raise an exception instead, it would need to test on infinity (which is not easy) or abort gracefully after some defined maximum number of time slices.



This class serves as a replacement for the existing **WFS::PropertyName** projection clause, but has sort of an opposite meaning. The **PropertyExclusion** projection clause enables the client to exclude properties. The propertyName property addresses both static and dynamic properties at the same time[17].

This projection clause is useful for use cases where only a subset of the properties of a feature (or time slice) need to be retrieved from a WFS. Additional information can always be skipped by the client, but it is generally more efficient not to receive it at all for saving bandwidth and processing time. This is especially true if the unwanted data is big in size compared to the data actually used, and if it is received very often. One example for this kind of data is metadata: there is time slice metadata, feature metadata and even message metadata defined in AIXM 5. For quite a lot of use cases, such as machine-to-machine communication in an operational scenario, metadata may not be processed at all.

9.6.1 Requirement /req/property-exclusion/property-name

If an instance of the **PropertyExclusion** class is present, for each feature in the result, remove all first-level and time slice properties that match the value of the **propertyName** property.

Multiple **PropertyExclusion** elements may occur in a **DynamicFeatureQuery** to exclude multiple elements. See A.10 for an example.

It is the responsibility of the client not to exclude properties (elements) that render the resulting feature (schema-)invalid.

[17] In theory, it is possible that the XML Schema defines a static property with the same name as a time slice property. However, we assume that this does not occur in real-world schemas, because we cannot think of a property that is modeled as both static and varying over time.

10 Indicating Support for Temporality Extension

To indicate the level of support for the Temporality Extension an OGC Web Service such as a WFS instance lists the supported Temporality Extension conformance classes in its Capabilities document. More precisely, it shall list the identifier of the supported Temporality Extension conformance classes as an ows:Profile value in the ServiceIdentification section of the Capabilities document – see following example.

Listing 1: WFS Capabilities listing supported Temporality Extension conformance classes

```
<WFS_Capabilities version="2.0.0" xmlns="http://www.opengis.net/wfs/2.0"
xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:fes="http://www.opengis.net/fes/2.0"
xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:ows="http://www.opengis.net/ows/1.1"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance">
    <ows:ServiceIdentification>
        <ows:Title>OGC Member WFS</ows:Title>
        <ows:Abstract>Web Feature Service</ows:Abstract>
        <ows:Keywords>
            <ows:Keyword>Aviation</ows:Keyword>
        </ows:Keywords>
        <ows:ServiceType>WFS</ows:ServiceType>
        <ows:ServiceTypeVersion>2.0.0</ows:ServiceTypeVersion>
        <ows:Profile>http://www.opengis.net/spec/wfs-te/1.0/conf/core</ows:Profile>
        <ows:Profile>http://www.opengis.net/spec/fes-te/1.0/conf/time-period-schedule-
evaluation-filter</ows:Profile>
        <ows:Fees>NONE</ows:Fees>
        <ows:AccessConstraints>NONE</ows:AccessConstraints>
    </ows:ServiceIdentification>
</WFS_Capabilities>
```

Note that conformance classes that are implied via a direct or indirect dependency of listed conformance classes do not need to be listed in the Capabilities. For example, FES-TE-Core is not listed here because all of the other Temporality Extension conformance classes depend on it.

In this example, the service supports the minimum set of Temporality Extension functionality (as indicated by the “WFS-TE Core” conformance class). In addition the “FES-TE TimePeriodScheduleEvaluationForFilter” conformance class is supported. However, FES-TE PropertyExclusion is not supported.

11 Compatibility with Existing WFS 2.0 Based Systems

The proposed extension is established as a new XML schema which inherits from the existing WFS and FES 2.0 XML Schema. Thus, the existing WFS 2.0 standard is preserved and support for dynamic feature queries is optional and depends upon the implementation of the Temporality Extension conformance class(es).

To enable existing, non-temporality aware WFS 2.0 clients to query AIXM 5 data in a basic way, a proxy WFS could be installed which sits in-between the client and the temporality-enabled WFS and translates incoming requests to a snapshot query parameterized with the current time.

12 Alternative Approaches

Introducing a new query type is not the only way to get the required functionality of a temporality-aware query interface. In the process of finding the best solution, several approaches were evaluated. In chapter 7 the existing WFS 2.0 standard was evaluated in how far it is matching the requirements. In this chapter we evaluate alternative approaches of extending the standard or utilizing other existing standards. For each approach advantages and disadvantages over the proposed dynamic feature query are discussed.

12.1 Web Processing Services

The Web Processing Service (WPS) is an OGC standard for processing geospatial data. A WPS provides client access across a network to pre-programmed calculations and/or computation models that operate on spatially referenced data. Data inputs can be legitimate calls to OGC web services, such as WFS.

It is possible to implement a WPS that meets all of the identified requirements. The general workflow is depicted in Figure 8.

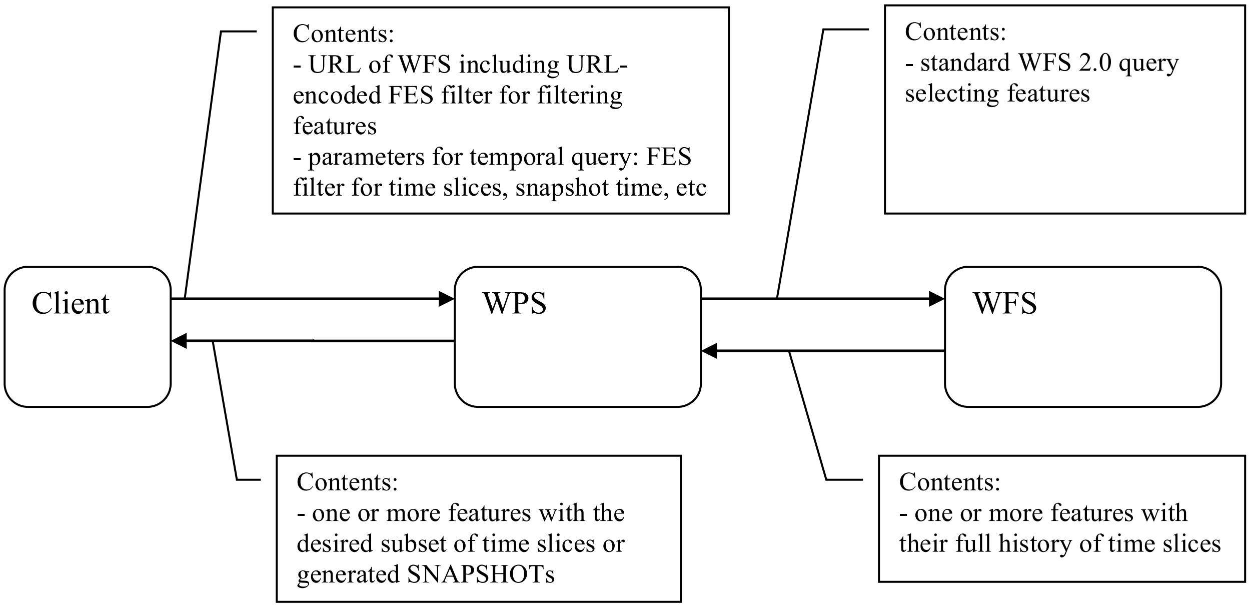

Contents:
- URL of WFS including URL-encoded FES filter for filtering features
- parameters for temporal query: FES filter for time slices, snapshot time, etc
Contents:
- standard WFS 2.0 query selecting features
Client
WPS
WFS
Contents:
- one or more features with the desired subset of time slices or generated SNAPSHOTs
Contents:
- one or more features with their full history of time slices

Figure 8 - WPS/WFS Workflow for Temporal Queries

Advantages

- Works with the existing WFS and WPS standards.

Disadvantages

- For each query, the full history of time slices of the selected features has to be transmitted between the WFS and the WPS. This is inefficient and will not be acceptable for productive, long-running systems.

- Still, though standard WFS and WPS are used, generic clients will not be able to make use of the temporal queries efficiently. In order to make use of and understand dynamic feature data clients still need to understand and support dynamic feature data, i.e. clients have to be temporality-aware.

Summary

The disadvantages clearly outweigh the advantages.

12.2 XQuery and XSLT

XQuery is a powerful query and functional language designed to query collections of XML data. XSLT is a transformation language designed for XML. Both are not OGC standards and do not define an interface for a web service. Hence, they cannot replace a WFS. However, the power of their query languages qualifies them for a possible extension of a WFS improving its filtering capabilities. For the sake of completeness we will briefly evaluate their application on AIXM 5 data.

XQuery and XSLT have the power to express all required queries including the generation of SNAPSHOT time slices. However,

- There are no spatio-temporal operators available. The powerful FES filter operators would have to be redefined using custom XPath functions

- There is no feature model as they are based on pure XML. There is no abstraction of the temporality models used in AIXM 5 or GML DFM.

The result is that even simple queries dealing with temporality aspects will require complex query expressions that are hard to build and read.

13 Future Work

While this discussion paper was developed, some features where discussed that didn’t make it in the first version of the proposed extensions. This is because they were either considered too advanced and hard to implement, or pre-mature, meaning that practical experience with existing features is required first. Those features are outlined in the following sections.

13.1 Advanced Filtering: valueFor() XPath function

A custom XPath function for evaluating a property at a given time would further enhance the power of dynamic feature queries. It could be used in **fes:ValueReference** elements and would enable the client to combine constraints on time slice property values valid at different points in time. As such, it may semantically replace the **timeIndicator** of the **DynamicFeatureFilter** (see 8.2). An example is given in A.2.2.

Related standards like the OGC Symbology Encoding may also profit by this function as it enables them to apply more powerful XPath expressions.

It can be defined as follows:

- Context node: AIXM 5 feature node containing the full history of time slices
- Parameters (Time Instant version):
    1. dateTime[18]
    2. XPath pointing to a time slice property (XPath subset as defined in [2])
- Parameters (Time Period version):
    1. dateTime (start time of the time period, inclusive[19])
    2. dateTime (end time of the time period, inclusive)
    3. XPath pointing to a time slice property (XPath subset as defined in [2])
- Return value: set of all nodes pointed to by the XPath expression and valid at the given time instant or period

Please note that the FES 2.0 specification puts constraints on the allowed XPath expressions of a **ValueReference**. Only a small subset of the XPath language is allowed. Hence, to be able to introduce the snapshot function there, the subset needs to be extended. The practical use of this function needs to be investigated further before it can be included in this extension.

13.2 Generalization of Transformation Clauses

In this paper we introduced the concept of transformation clauses, which is used to generate (or derive) snapshot time slices from a history (list) of time slices. We allow a single optional transformation clause to be used. This is a powerful concept, as it allows extensive processing of features in the service before they are transmitted to the client. It is similar to the application of Web Processing Services chained to Web Feature Services, but with an important difference: the processing happens inside the Web Feature Service, not in an external one. This saves transport overhead and bandwidth which is crucial for high performance applications or data-intensive operations (such as AIXM systems, especially when dealing with long term AIXM data; see also 12.1).

As a future work, the concept of transformation clauses could be generalized towards allowing multiple transformation clauses chained together, where each member processes the outcome of previous one in the chain.

[18] XML Schema primitive data type

[19] If needed, additional functions or parameters can be introduced to support open time intervals in a future version

Another generalization is possible towards the definition of a transformation clause based on XSLT. This is easy to implement, as XSLT processors are easy to have as free and open source software. It is also well standardized and a wide-spread technology. This could also be seen as natural advancement of the **GetPropertyValue** operation. Instead of passing an XPath expression to select some elements from the feature, the feature is transformed using XSLT, which is much more powerful. As long as the XSLT transformation does not produce a schema-invalid result (which could be defined as a requirement), this should harmonize with the WFS and FES standards.

Annex A Use Case Oriented Examples

A.1 Data Retrieval Use Case 1 (see 6.1)

Request:

```
<?xml version="1.0" encoding="UTF-8"?>

<!-- DynamicFeatureQuery example for snapshot generation. Use case: Retrieve
  the complete state of a feature at a point in time Category: Visualisation,
  decision support Example: Retrieve the state of a runway at the time of arrival. -->

<wfs:GetFeature xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:wfs="http://www.opengis.net/wfs/2.0" xmlns:wfs-te="http://www.opengis.net/wfs-
te/1.0"
  xmlns:fes-te="http://www.opengis.net/fes-te/1.0"
xmlns:fes="http://www.opengis.net/fes/2.0"
  xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:aixm="http://www.aixm.aero/schema/5.1"
  service="WFS" version="2.0.0">
  <wfs-te:DynamicFeatureQuery typeNames="aixm:Runway">
    <fes:Filter>
      <fes:PropertyIsEqualTo>
        <fes:ValueReference>gml:identifier</fes:ValueReference>
        <fes:Literal>0083defb-b42e-4417-9be2-7aba2db2674d</fes:Literal>
      </fes:PropertyIsEqualTo>
    </fes:Filter>
    <wfs-te:transformation>
      <wfs-te:SnapshotGeneration>
        <wfs-te:snapshotTime>
          <gml:TimeInstant gml:id="gmlID1">
            <gml:timePosition>2011-07-12T09:11:01.857Z
            </gml:timePosition>
          </gml:TimeInstant>
        </wfs-te:snapshotTime>
      </wfs-te:SnapshotGeneration>
    </wfs-te:transformation>
  </wfs-te:DynamicFeatureQuery>
</wfs:GetFeature>
```

A.2 Data Retrieval Use Case 2 (see 6.1)

A.2.1 Request using timeIndicator:

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- DynamicFeatureQuery example for snapshot generation with a time slice
  filter Use case: Retrieve all features fulfilling a certain criteria at a
  point in time Category: Decision support Example: Retrieve all airports in
  certain area that are operational at the time of arrival -->
<wfs:GetFeature xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:wfs="http://www.opengis.net/wfs/2.0" xmlns:wfs-te="http://www.opengis.net/wfs-
te/1.0"
  xmlns:fes="http://www.opengis.net/fes/2.0" xmlns:fes-te="http://www.opengis.net/fes-
te/1.0"
  xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:aixm="http://www.aixm.aero/schema/5.1"
  service="WFS" version="2.0.0">
  <wfs-te:DynamicFeatureQuery typeNames="aixm:AirportHeliport">
    <fes-te:DynamicFeatureFilter evaluateSchedules="true">
      <fes-te:timeIndicator>
        <gml:TimeInstant gml:id="gmlID2">
          <gml:timePosition>2011-07-12T09:11:01.857Z</gml:timePosition>
        </gml:TimeInstant>
      </fes-te:timeIndicator>
      <fes-te:featureFilter>
        <fes:Filter>
```

```
          <fes:And>
            <fes:DWithin>
              <fes:ValueReference>aixm:timeSlice/*/aixm:ARP
              </fes:ValueReference>
              <gml:Point srsName="urn:ogc:def:crs:EPSG::4326"
                gml:id="gmlID3">
                <gml:pos>50.9 4.34</gml:pos>
              </gml:Point>
              <fes:Distance uom="M">100000</fes:Distance>
            </fes:DWithin>
            <fes:PropertyIsEqualTo>

<fes:ValueReference>aixm:timeSlice/*/aixm:availability/aixm:operationalStatus
              </fes:ValueReference>
              <!-- the schedule is automatically evaluated at the given time -->
              <fes:Literal>NORMAL</fes:Literal>
            </fes:PropertyIsEqualTo>
          </fes:And>
        </fes:Filter>
      </fes-te:featureFilter>
    </fes-te:DynamicFeatureFilter>
    <wfs-te:transformation>
      <wfs-te:SnapshotGeneration>
        <wfs-te:snapshotTime>
          <gml:TimeInstant gml:id="gmlID1">
            <gml:timePosition>2011-07-12T09:11:01.857Z
            </gml:timePosition>
          </gml:TimeInstant>
        </wfs-te:snapshotTime>
      </wfs-te:SnapshotGeneration>
    </wfs-te:transformation>

  </wfs-te:DynamicFeatureQuery>
</wfs:GetFeature>
```

A.2.2 Alternative Request using the valueFor() XPath function (Future Work, see 13.1)

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- DynamicFeatureQuery example for snapshot generation with a time slice
  filter Use case: Retrieve all features fulfilling a certain criteria at a
  point in time Category: Decision support Example: Retrieve all airports in
  certain area that are operational at the time of arrival -->
<wfs:GetFeature xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:wfs="http://www.opengis.net/wfs/2.0" xmlns:wfs-te="http://www.opengis.net/wfs-
te/1.0"
  xmlns:fes="http://www.opengis.net/fes/2.0" xmlns:fes-te="http://www.opengis.net/fes-
te/1.0"
  xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:aixm="http://www.aixm.aero/schema/5.1"
  service="WFS" version="2.0.0">
  <wfs-te:DynamicFeatureQuery typeNames="aixm:AirportHeliport">
    <fes-te:DynamicFeatureFilter>
      <fes-te:featureFilter>
        <fes:Filter>
          <fes:And>
            <fes:DWithin>
              <fes:ValueReference>aixm:valueFor('2011-07-12T09:11:01.857Z','aixm:ARP')
              </fes:ValueReference>
              <gml:Point srsName="urn:ogc:def:crs:EPSG::4326"
                gml:id="gmlID3">
                <gml:pos>50.9 4.34</gml:pos>
              </gml:Point>
              <fes:Distance uom="M">100000</fes:Distance>
            </fes:DWithin>
            <fes:PropertyIsEqualTo>
              <fes:ValueReference>aixm:valueFor('2011-07-
12T09:11:01.857Z','aixm:availability/aixm:operationalStatus')
              </fes:ValueReference>
              <!-- the schedule is automatically evaluated at the given time -->
              <fes:Literal>NORMAL</fes:Literal>
            </fes:PropertyIsEqualTo>
          </fes:And>
        </fes:Filter>
      </fes-te:featureFilter>
    </fes-te:DynamicFeatureFilter>
    <wfs-te:transformation>
      <wfs-te:SnapshotGeneration>
        <wfs-te:snapshotTime>
          <gml:TimeInstant gml:id="gmlID1">
            <gml:timePosition>2011-07-12T09:11:01.857Z
            </gml:timePosition>
          </gml:TimeInstant>
        </wfs-te:snapshotTime>
      </wfs-te:SnapshotGeneration>
    </wfs-te:transformation>
  </wfs-te:DynamicFeatureQuery>
</wfs:GetFeature>
```

A.3 Data Retrieval Use Case 3 (see 6.1)

Request:

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- DynamicFeatureQuery example for retrieving relevant time slices Use
  case: Retrieve all time slices of a feature relevant for (i.e. affecting
  the state at) a point in time Category: Change-aware visualisation Example:
  Retrieve all time slices relevant for the time of arrival of a runway. -->
<wfs:GetFeature xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:wfs="http://www.opengis.net/wfs/2.0" xmlns:wfs-te="http://www.opengis.net/wfs-
te/1.0"
  xmlns:fes="http://www.opengis.net/fes/2.0" xmlns:fes-te="http://www.opengis.net/fes-
te/1.0"
  xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:aixm="http://www.aixm.aero/schema/5.1"
  service="WFS" version="2.0.0">
  <wfs-te:DynamicFeatureQuery typeNames="aixm:Runway">
    <wfs-te:TimeSliceProjection>
      <wfs-te:relevantTime>
        <gml:TimeInstant gml:id="gmlID1">
          <gml:timePosition>2011-07-12T09:11:01.857Z</gml:timePosition>
        </gml:TimeInstant>
      </wfs-te:relevantTime>
    </wfs-te:TimeSliceProjection>
    <fes:Filter>
      <fes:PropertyIsEqualTo>
        <fes:ValueReference>gml:identifier</fes:ValueReference>
        <fes:Literal>0083defb-b42e-4417-9be2-7aba2db2674d</fes:Literal>
      </fes:PropertyIsEqualTo>
    </fes:Filter>
  </wfs-te:DynamicFeatureQuery>
</wfs:GetFeature>
```

A.4 Data Retrieval Use Case 4 (see 6.1)

Request:

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- DynamicFeatureQuery example for retrieving BASELINE time slices Use
  case: Retrieve the BASELINE of a feature valid at a point in time Category:
  AIP publication Example: Retrieve the BASELINE of an airport at an AIRAC
  effective date. -->
<wfs:GetFeature xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:wfs="http://www.opengis.net/wfs/2.0" xmlns:wfs-te="http://www.opengis.net/wfs-
te/1.0"
  xmlns:fes="http://www.opengis.net/fes/2.0" xmlns:fes-te="http://www.opengis.net/fes-
te/1.0"
```

```
  xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:aixm="http://www.aixm.aero/schema/5.1"
  service="WFS" version="2.0.0">
  <wfs-te:DynamicFeatureQuery typeNames="aixm:AirportHeliport">
    <wfs-te:TimeSliceProjection>
      <wfs-te:relevantTime>
        <gml:TimeInstant gml:id="gmlID1">
          <gml:timePosition>2010-04-08T00:00:00.000Z</gml:timePosition>
        </gml:TimeInstant>
      </wfs-te:relevantTime>
      <wfs-te:timeSliceFilter>
        <fes:Filter>
          <fes:PropertyIsEqualTo>
            <fes:ValueReference>aixm:interpretation</fes:ValueReference>
            <fes:Literal>BASELINE</fes:Literal>
          </fes:PropertyIsEqualTo>
        </fes:Filter>
      </wfs-te:timeSliceFilter>
    </wfs-te:TimeSliceProjection>
    <fes:Filter>
      <fes:PropertyIsEqualTo>
        <fes:ValueReference>gml:identifier</fes:ValueReference>
        <fes:Literal>0083defb-b42e-4417-9be2-7aba2db2674d</fes:Literal>
      </fes:PropertyIsEqualTo>
    </fes:Filter>
  </wfs-te:DynamicFeatureQuery>
</wfs:GetFeature>
```

A.5 Data Retrieval Use Case 5 (see 6.1)

Request:

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- DynamicFeatureQuery example for retrieving and filtering TEMPDELTA time
  slice Use case: Retrieve all TEMPDELTAs of a feature fully or partly valid
  for a time period and matching a certain criteria Category: NOTAM communication
  Example: Retrieve all TEMPDELTAs affecting the operational status of a navigation
  aid until the time of arrival. -->
<wfs:GetFeature xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:wfs="http://www.opengis.net/wfs/2.0" xmlns:wfs-te="http://www.opengis.net/wfs-
te/1.0"
  xmlns:fes="http://www.opengis.net/fes/2.0" xmlns:fes-te="http://www.opengis.net/fes-
te/1.0"
  xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:aixm="http://www.aixm.aero/schema/5.1"
  service="WFS" version="2.0.0">
  <wfs-te:DynamicFeatureQuery typeNames="aixm:Navaid">
    <wfs-te:TimeSliceProjection>
      <wfs-te:timeSliceFilter>
        <fes:Filter>
          <fes:And>
            <fes:PropertyIsEqualTo>
              <fes:ValueReference>aixm:interpretation
              </fes:ValueReference>
              <fes:Literal>TEMPDELTA</fes:Literal>
            </fes:PropertyIsEqualTo>
            <fes:AnyInteracts>
              <fes:ValueReference>gml:validTime</fes:ValueReference>
              <gml:TimePeriod gml:id="gmlID1">
                <gml:beginPosition>2010-04-08T00:00:00.000Z
                </gml:beginPosition>
                <gml:endPosition>2010-05-08T00:00:00.000Z
                </gml:endPosition>
              </gml:TimePeriod>
            </fes:AnyInteracts>
            <fes:Not>
              <fes:PropertyIsNull>
                <fes:ValueReference>aixm:availability
                </fes:ValueReference>
              </fes:PropertyIsNull>
            </fes:Not>
          </fes:And>
        </fes:Filter>
      </wfs-te:timeSliceFilter>
    </wfs-te:TimeSliceProjection>
    <fes-te:DynamicFeatureFilter>
      <fes-te:featureFilter>
        <fes:Filter>
          <fes:PropertyIsEqualTo>
            <fes:ValueReference>gml:identifier</fes:ValueReference>
            <fes:Literal>0083defb-b42e-4417-9be2-7aba2db2674d
            </fes:Literal>
          </fes:PropertyIsEqualTo>
        </fes:Filter>
      </fes-te:featureFilter>
    </fes-te:DynamicFeatureFilter>
  </wfs-te:DynamicFeatureQuery>
</wfs:GetFeature>
```

A.6 Data Retrieval Use Case 6 (see 6.1)

Request:

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- DynamicFeatureQuery example for retrieving and filtering PERMDELTA time
  slices Use case: Retrieve all PERMDELTAs of a feature valid for a time period
  Category: Replication Example: Retrieve all PERMDELTAs of a feature valid
  now or in the future. -->
<wfs:GetFeature xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:wfs="http://www.opengis.net/wfs/2.0" xmlns:wfs-te="http://www.opengis.net/wfs-
te/1.0"
  xmlns:fes="http://www.opengis.net/fes/2.0" xmlns:fes-te="http://www.opengis.net/fes-
te/1.0"
  xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:aixm="http://www.aixm.aero/schema/5.1"
  service="WFS" version="2.0.0">
  <wfs-te:DynamicFeatureQuery typeNames="aixm:Navaid">
    <wfs-te:TimeSliceProjection>
      <wfs-te:timeSliceFilter>
        <fes:Filter>
          <fes:And>
            <fes:PropertyIsEqualTo>
              <fes:ValueReference>aixm:interpretation
              </fes:ValueReference>
              <fes:Literal>PERMDELTA</fes:Literal>
            </fes:PropertyIsEqualTo>
            <fes:AnyInteracts>
              <fes:ValueReference>gml:validTime</fes:ValueReference>
              <gml:TimePeriod gml:id="gmlID1">
                <gml:beginPosition>2010-04-08T00:00:00.000Z
                </gml:beginPosition>
                <gml:endPosition />
              </gml:TimePeriod>
            </fes:AnyInteracts>
          </fes:And>
        </fes:Filter>
      </wfs-te:timeSliceFilter>
    </wfs-te:TimeSliceProjection>
    <fes-te:DynamicFeatureFilter>
      <fes-te:featureFilter>
        <fes:Filter>
          <fes:PropertyIsEqualTo>
            <fes:ValueReference>gml:identifier</fes:ValueReference>
            <fes:Literal>0083defb-b42e-4417-9be2-7aba2db2674d
            </fes:Literal>
          </fes:PropertyIsEqualTo>
        </fes:Filter>
      </fes-te:featureFilter>
    </fes-te:DynamicFeatureFilter>
  </wfs-te:DynamicFeatureQuery>
</wfs:GetFeature>
```

A.7 Data Retrieval Use Case 7 (see 6.1)

Request:

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- DynamicFeatureQuery example for retrieving and filtering PERMDELTA time
  slice Use case: Retrieve specific time slices of a feature by their unique
  id Category: Replication, technical verification Example: Retrieve the PERMDELTA
  time slice of a feature given the sequence number and correction number -->
<wfs:GetFeature xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:wfs="http://www.opengis.net/wfs/2.0" xmlns:wfs-te="http://www.opengis.net/wfs-
te/1.0"
  xmlns:fes="http://www.opengis.net/fes/2.0" xmlns:fes-te="http://www.opengis.net/fes-
te/1.0"
  xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:aixm="http://www.aixm.aero/schema/5.1"
  service="WFS" version="2.0.0">
  <wfs-te:DynamicFeatureQuery typeNames="aixm:Navaid">
    <wfs-te:TimeSliceProjection
      includeCanceled="true" includeCorrected="true">
      <wfs-te:timeSliceFilter>
        <fes:Filter>
          <fes:And>
            <fes:PropertyIsEqualTo>
              <fes:ValueReference>aixm:interpretation
              </fes:ValueReference>
              <fes:Literal>PERMDELTA</fes:Literal>
            </fes:PropertyIsEqualTo>
            <fes:PropertyIsEqualTo>
              <fes:ValueReference>aixm:sequenceNumber
              </fes:ValueReference>
              <fes:Literal>2</fes:Literal>
            </fes:PropertyIsEqualTo>
            <fes:PropertyIsEqualTo>
              <fes:ValueReference>aixm:correctionNumber
              </fes:ValueReference>
              <fes:Literal>1</fes:Literal>
            </fes:PropertyIsEqualTo>
          </fes:And>
        </fes:Filter>
      </wfs-te:timeSliceFilter>
    </wfs-te:TimeSliceProjection>
    <fes-te:DynamicFeatureFilter>
      <fes-te:featureFilter>
        <fes:Filter>
          <fes:PropertyIsEqualTo>
            <fes:ValueReference>gml:identifier</fes:ValueReference>
            <fes:Literal>0083defb-b42e-4417-9be2-7aba2db2674d
            </fes:Literal>
          </fes:PropertyIsEqualTo>
        </fes:Filter>
      </fes-te:featureFilter>
    </fes-te:DynamicFeatureFilter>
  </wfs-te:DynamicFeatureQuery>
</wfs:GetFeature>
```

A.8 Data Retrieval Use Case 8 (see 6.1)

Request:

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Example for retrieving all time slices of a feature (standard WFS 2.0
  query, no DynamicFeatureQuery necessary) Use case: Retrieve the full history
  of a feature Category: Backup Example: Retrieve all time slices of a feature
  (BASELINEs, PERMDELTAs and TEMPDELTAs) valid at any point in time. -->
<wfs:GetFeature xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:wfs="http://www.opengis.net/wfs/2.0" xmlns:fes="http://www.opengis.net/fes/2.0"
  xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:aixm="http://www.aixm.aero/schema/5.1"
  service="WFS" version="2.0.0">
  <wfs:Query typeNames="aixm:Navaid">
    <fes:Filter>
      <fes:PropertyIsEqualTo>
        <fes:ValueReference>gml:identifier</fes:ValueReference>
        <fes:Literal>0083defb-b42e-4417-9be2-7aba2db2674d</fes:Literal>
      </fes:PropertyIsEqualTo>
    </fes:Filter>
  </wfs:Query>
</wfs:GetFeature>
```

A.9 Evaluation of Properties with Schedule (see 9.4)

A.9.1 Data Store Contents

```
<aixm:Airspace xmlns:aixm="http://www.aixm.aero/schema/5.1"
xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:xlink="http://www.w3.org/1999/xlink"
gml:id="tmp">
  <gml:identifier codeSpace="http://www.example.org">0083defb-b42e-4417-9be2-
7aba2db2674d</gml:identifier>
  <aixm:timeSlice>
    <aixm:AirspaceTimeSlice gml:id="Airspace1_TD2">
      <gml:validTime>
        <gml:TimePeriod gml:id="Airspace1_TD2_TP">
          <gml:beginPosition>2011-07-01T00:00:00.000Z</gml:beginPosition>
          <gml:endPosition>2011-07-15T23:59:59Z</gml:endPosition>
        </gml:TimePeriod>
      </gml:validTime>
      <aixm:interpretation>BASELINE</aixm:interpretation>
      <aixm:sequenceNumber>1</aixm:sequenceNumber>
      <aixm:activation>
        <aixm:AirspaceActivation gml:id="Airspace1_TD2_AACT1">
          <aixm:timeInterval>
            <aixm:Timesheet gml:id="Airspace1_TD2_TS1">
              <aixm:timeReference>UTC+2</aixm:timeReference>
              <aixm:day>ANY</aixm:day>
              <aixm:startTime>00:00</aixm:startTime>
              <aixm:endTime>24:00</aixm:endTime>
            </aixm:Timesheet>
          </aixm:timeInterval>
          <aixm:timeInterval>
            <aixm:Timesheet gml:id="Airspace1_TD2_TS2_2">
              <aixm:timeReference>UTC+2</aixm:timeReference>
              <aixm:day>ANY</aixm:day>
              <aixm:startTime>06:00</aixm:startTime>
              <aixm:endTime>17:59</aixm:endTime>
              <aixm:excluded>YES</aixm:excluded>
            </aixm:Timesheet>
          </aixm:timeInterval>
          <aixm:activity>MILOPS</aixm:activity>
          <aixm:status>INACTIVE</aixm:status>
          <!-- other AirspaceActivation properties omitted for brevity -->
        </aixm:AirspaceActivation>
      </aixm:activation>
      <aixm:activation>
        <aixm:AirspaceActivation gml:id="Airspace1_TD2_AACT2">
          <aixm:timeInterval>
            <aixm:Timesheet gml:id="Airspace1_TD2_TS2">
              <aixm:timeReference>UTC+2</aixm:timeReference>
              <aixm:day>ANY</aixm:day>
              <aixm:startTime>06:00</aixm:startTime>
              <aixm:endTime>17:59</aixm:endTime>
            </aixm:Timesheet>
          </aixm:timeInterval>
          <aixm:activity>MILOPS</aixm:activity>
          <aixm:status>ACTIVE</aixm:status>
          <!-- other AirspaceActivation properties omitted for brevity -->
        </aixm:AirspaceActivation>
      </aixm:activation>
    </aixm:AirspaceTimeSlice>
  </aixm:timeSlice>
</aixm:Airspace>
```

A.9.2 Request

```
<wfs:GetFeature xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:wfs="http://www.opengis.net/wfs/2.0" xmlns:wfs-te="http://www.opengis.net/wfs-
te/1.0"
  xmlns:fes="http://www.opengis.net/fes/2.0" xmlns:fes-te="http://www.opengis.net/fes-
te/1.0"
  xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:aixm="http://www.aixm.aero/schema/5.1"
  service="WFS" version="2.0.0">
  <wfs-te:DynamicFeatureQuery typeNames="aixm:Airspace">
    <fes-te:DynamicFeatureFilter>
      <fes-te:featureFilter>
        <fes:Filter>
          <fes:PropertyIsEqualTo>
            <fes:ValueReference>gml:identifier</fes:ValueReference>
            <fes:Literal>0083defb-b42e-4417-9be2-7aba2db2674d</fes:Literal>
          </fes:PropertyIsEqualTo>
        </fes:Filter>
      </fes-te:featureFilter>
    </fes-te:DynamicFeatureFilter>
    <wfs-te:transformation>
      <wfs-te:SnapshotGeneration evaluateSchedules="true">
        <wfs-te:snapshotTime>
          <gml:TimeInstant gml:id="gmlID1">
            <gml:timePosition>2011-07-01T08:00:00.000Z</gml:timePosition>
          </gml:TimeInstant>
        </wfs-te:snapshotTime>
      </wfs-te:SnapshotGeneration>
    </wfs-te:transformation>
  </wfs-te:DynamicFeatureQuery>
</wfs:GetFeature>
```

A.9.3 Response

```
<?xml version="1.0" encoding="UTF-8"?>
<wfs:FeatureCollection timeStamp="2008-09-07T19:00:00"
  numberReturned="1" numberMatched="1" xmlns:wfs="http://www.opengis.net/wfs/2.0"
  xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance">
  <wfs:member>
    <aixm:Airspace xmlns:aixm="http://www.aixm.aero/schema/5.1"
      xmlns:gml="http://www.opengis.net/gml/3.2" gml:id="id1">
      <gml:identifier codeSpace="http://www.example.org">0083defb-b42e-4417-9be2-
7aba2db2674d</gml:identifier>
      <aixm:timeSlice>
        <aixm:AirspaceTimeSlice gml:id="Airspace1_TD2">
          <gml:validTime>
            <gml:TimeInstant gml:id="i1">
              <gml:timePosition>2011-07-01T08:00:00.000Z
              </gml:timePosition>
            </gml:TimeInstant>
          </gml:validTime>
          <aixm:interpretation>SNAPSHOT</aixm:interpretation>
          <aixm:activation>
```

```
            <aixm:AirspaceActivation gml:id="Airspace1_TD2_AACT2">
              <aixm:timeInterval>
                <aixm:Timesheet gml:id="Airspace1_TD2_TS2">
                  <aixm:timeReference>UTC+2</aixm:timeReference>
                  <aixm:day>ANY</aixm:day>
                  <aixm:startTime>06:00</aixm:startTime>
                  <aixm:endTime>17:59</aixm:endTime>
                </aixm:Timesheet>
              </aixm:timeInterval>
              <aixm:activity>MILOPS</aixm:activity>
              <aixm:status>ACTIVE</aixm:status>
            </aixm:AirspaceActivation>
          </aixm:activation>
        </aixm:AirspaceTimeSlice>
      </aixm:timeSlice>
    </aixm:Airspace>
  </wfs:member>
</wfs:FeatureCollection>
```

A.10 Property exclusion (see 9.6)

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- DynamicFeatureQuery example for property exclusion. Use case: Retrieve
  the complete state of a feature at a point in time but exclude metadata to
  save bandwidth Category: Visualisation, decision support Example: Retrieve
  the state of a runway at the time of arrival without metadata. -->
<wfs:GetFeature xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:wfs="http://www.opengis.net/wfs/2.0" xmlns:wfs-te="http://www.opengis.net/wfs-
te/1.0"
  xmlns:fes-te="http://www.opengis.net/fes-te/1.0"
xmlns:fes="http://www.opengis.net/fes/2.0"
  xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:aixm="http://www.aixm.aero/schema/5.1"
  service="WFS" version="2.0.0">
  <wfs-te:DynamicFeatureQuery typeNames="aixm:Runway">
    <wfs-te:PropertyExclusion>
      <wfs-te:propertyName>aixm:featureMetadata</wfs-te:propertyName>
    </wfs-te:PropertyExclusion>
    <wfs-te:PropertyExclusion>
      <wfs-te:propertyName>aixm:timeSliceMetadata</wfs-te:propertyName>
    </wfs-te:PropertyExclusion>
    <fes:Filter>
      <fes:PropertyIsEqualTo>
        <fes:ValueReference>gml:identifier</fes:ValueReference>
        <fes:Literal>0083defb-b42e-4417-9be2-7aba2db2674d</fes:Literal>
      </fes:PropertyIsEqualTo>
    </fes:Filter>
    <wfs-te:transformation>
      <wfs-te:SnapshotGeneration>
        <wfs-te:snapshotTime>
          <gml:TimeInstant gml:id="gmlID1">
            <gml:timePosition>2011-07-12T09:11:01.857Z
            </gml:timePosition>
          </gml:TimeInstant>
        </wfs-te:snapshotTime>
      </wfs-te:SnapshotGeneration>
    </wfs-te:transformation>
  </wfs-te:DynamicFeatureQuery>
</wfs:GetFeature>
```

Annex B Working Examples on the DONLON Dataset

As part of the work on this document, the AIXM DONLON testing dataset was extended to be able to demonstrate all features of WFS Temporality Extension. Look at version of

April 26th, 2014 [12] to retrace the examples. All requests contain a constraint on a gml:identifier. This is not realistic for real-world requests, but ensures compatibility with future extensions of the DONLON dataset.

B.1 Dynamic Filter on gml:identifier of a canceled commissioning

B.1.1 Request

```
<?xml version="1.0"?>
<wfs:GetFeature xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:wfs="http://www.opengis.net/wfs/2.0" xmlns:wfs-te="http://www.opengis.net/wfs-
te/1.0" xmlns:fes-te="http://www.opengis.net/fes-te/1.0"
xmlns:fes="http://www.opengis.net/fes/2.0" xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:aixm="http://www.aixm.aero/schema/5.1" service="WFS" version="2.0.0">
  <wfs-te:DynamicFeatureQuery typeNames="aixm:VerticalStructure">
    <fes-te:DynamicFeatureFilter>
      <fes-te:featureFilter>
        <fes:Filter>
          <fes:PropertyIsEqualTo>
            <fes:ValueReference>gml:identifier</fes:ValueReference>
            <fes:Literal>e9ce3cc0-b41f-11e3-a5e2-0800200c9a66</fes:Literal>
          </fes:PropertyIsEqualTo>
        </fes:Filter>
      </fes-te:featureFilter>
    </fes-te:DynamicFeatureFilter>
  </wfs-te:DynamicFeatureQuery>
</wfs:GetFeature>
```

B.1.2 Response

```
<?xml version="1.0"?>
<wfs:FeatureCollection xmlns:wfs="http://www.opengis.net/wfs/2.0"
xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" timeStamp="2008-09-07T19:00:00" numberReturned="0" numberMatched="0">
</wfs:FeatureCollection>
```

B.2 Dynamic Filter on corrected data

B.2.1 Request

```
<?xml version="1.0"?>
<wfs:GetFeature xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:wfs="http://www.opengis.net/wfs/2.0" xmlns:wfs-te="http://www.opengis.net/wfs-
te/1.0" xmlns:fes-te="http://www.opengis.net/fes-te/1.0"
xmlns:fes="http://www.opengis.net/fes/2.0" xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:aixm="http://www.aixm.aero/schema/5.1" service="WFS" version="2.0.0">
  <wfs-te:DynamicFeatureQuery typeNames="aixm:VerticalStructure">
    <fes-te:DynamicFeatureFilter>
      <fes-te:featureFilter>
        <fes:Filter>
          <fes:And>
            <fes:PropertyIsEqualTo>
              <fes:ValueReference>gml:identifier</fes:ValueReference>
              <fes:Literal>8c755520-b42b-11e3-a5e2-0800500c9a66</fes:Literal>
            </fes:PropertyIsEqualTo>
            <fes:PropertyIsEqualTo>
              <fes:ValueReference>aixm:timeSlice/*/aixm:lighted</fes:ValueReference>
              <fes:Literal>NO</fes:Literal>
            </fes:PropertyIsEqualTo>
          </fes:And>
        </fes:Filter>
      </fes-te:featureFilter>
    </fes-te:DynamicFeatureFilter>
  </wfs-te:DynamicFeatureQuery>
```

```
</wfs:GetFeature>
```

B.2.2 Response

```
<?xml version="1.0"?>
<wfs:FeatureCollection xmlns:wfs="http://www.opengis.net/wfs/2.0"
xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" timeStamp="2008-09-07T19:00:00" numberReturned="0" numberMatched="0">
</wfs:FeatureCollection>
```

B.3 Dynamic Filter on canceled data

B.3.1 Request

```
<?xml version="1.0"?>
<wfs:GetFeature xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:wfs="http://www.opengis.net/wfs/2.0" xmlns:wfs-te="http://www.opengis.net/wfs-
te/1.0" xmlns:fes-te="http://www.opengis.net/fes-te/1.0"
xmlns:fes="http://www.opengis.net/fes/2.0" xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:aixm="http://www.aixm.aero/schema/5.1" service="WFS" version="2.0.0">
  <wfs-te:DynamicFeatureQuery typeNames="aixm:VerticalStructure">
    <fes-te:DynamicFeatureFilter>
      <fes-te:featureFilter>
        <fes:Filter>
          <fes:And>
            <fes:PropertyIsEqualTo>
              <fes:ValueReference>gml:identifier</fes:ValueReference>
              <fes:Literal>8c755520-b42b-11e3-a5e2-0800400c9a66</fes:Literal>
            </fes:PropertyIsEqualTo>
            <fes:PropertyIsEqualTo>

<fes:ValueReference>aixm:timeSlice/*/aixm:lightingICAOStandard</fes:ValueReference>
              <fes:Literal>YES</fes:Literal>
            </fes:PropertyIsEqualTo>
          </fes:And>
        </fes:Filter>
      </fes-te:featureFilter>
    </fes-te:DynamicFeatureFilter>
  </wfs-te:DynamicFeatureQuery>
</wfs:GetFeature>
```

B.3.2 Response

```
<?xml version="1.0"?>
<wfs:FeatureCollection xmlns:wfs="http://www.opengis.net/wfs/2.0"
xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" timeStamp="2008-09-07T19:00:00" numberReturned="0" numberMatched="0">
</wfs:FeatureCollection>
```

B.4 Dynamic Filter with time instant and without evaluation of properties with schedules

B.4.1 Request

```
<?xml version="1.0"?>
<wfs:GetFeature xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:wfs="http://www.opengis.net/wfs/2.0" xmlns:wfs-te="http://www.opengis.net/wfs-
te/1.0" xmlns:fes-te="http://www.opengis.net/fes-te/1.0"
xmlns:fes="http://www.opengis.net/fes/2.0" xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:aixm="http://www.aixm.aero/schema/5.1" service="WFS" version="2.0.0">
  <wfs-te:DynamicFeatureQuery typeNames="aixm:VerticalStructure">
    <fes-te:DynamicFeatureFilter>
      <fes-te:timeIndicator>
```

```
        <gml:TimeInstant gml:id="ID_1">
          <gml:timePosition>2012-10-02T15:30:00.000Z</gml:timePosition>
        </gml:TimeInstant>
      </fes-te:timeIndicator>
      <fes-te:featureFilter>
        <fes:Filter>
          <fes:And>
            <fes:PropertyIsEqualTo>
              <fes:ValueReference>gml:identifier</fes:ValueReference>
              <fes:Literal>8c755520-b42b-11e3-a5e2-0800500c9a66</fes:Literal>
            </fes:PropertyIsEqualTo>
            <fes:PropertyIsGreaterThan>
<fes:ValueReference>aixm:timeSlice/*/aixm:part/*/aixm:verticalExtent</fes:ValueReference>
              <fes:Literal>0</fes:Literal>
            </fes:PropertyIsGreaterThan>
          </fes:And>
        </fes:Filter>
      </fes-te:featureFilter>
    </fes-te:DynamicFeatureFilter>
  </wfs-te:DynamicFeatureQuery>
</wfs:GetFeature>
```

B.4.2 Response

```
<?xml version="1.0"?>
<wfs:FeatureCollection xmlns:wfs="http://www.opengis.net/wfs/2.0"
xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" timeStamp="2008-09-07T19:00:00" numberReturned="1" numberMatched="1">
  <wfs:member>
    <!-- omitted for brevity: the complete feature "CRANE 5" with all time slices is
returned here -->
  </wfs:member>
</wfs:FeatureCollection>
```

B.5 Dynamic Filter with time period and without evaluation of properties with schedules

B.5.1 Request

```
<?xml version="1.0"?>
<wfs:GetFeature xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:wfs="http://www.opengis.net/wfs/2.0" xmlns:wfs-te="http://www.opengis.net/wfs-
te/1.0" xmlns:fes-te="http://www.opengis.net/fes-te/1.0"
xmlns:fes="http://www.opengis.net/fes/2.0" xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:aixm="http://www.aixm.aero/schema/5.1" service="WFS" version="2.0.0">
  <wfs-te:DynamicFeatureQuery typeNames="aixm:VerticalStructure">
    <fes-te:DynamicFeatureFilter>
      <fes-te:timeIndicator>
        <gml:TimePeriod gml:id="ID_1">
          <gml:beginPosition>2012-10-11T18:31:00Z</gml:beginPosition>
          <gml:endPosition>2012-10-13T15:31:00Z</gml:endPosition>
        </gml:TimePeriod>
      </fes-te:timeIndicator>
      <fes-te:featureFilter>
        <fes:Filter>
          <fes:And>
            <fes:PropertyIsEqualTo>
              <fes:ValueReference>gml:identifier</fes:ValueReference>
              <fes:Literal>8c755520-b42b-11e3-a5e2-0800500c9a66</fes:Literal>
            </fes:PropertyIsEqualTo>
            <fes:PropertyIsGreaterThan>
<fes:ValueReference>aixm:timeSlice/*/aixm:part/*/aixm:verticalExtent</fes:ValueReference>
              <fes:Literal>0</fes:Literal>
            </fes:PropertyIsGreaterThan>
          </fes:And>
        </fes:Filter>
      </fes-te:featureFilter>
    </fes-te:DynamicFeatureFilter>
  </wfs-te:DynamicFeatureQuery>
```

```
</wfs:GetFeature>
```

B.5.2 Response

```
<?xml version="1.0"?>
<wfs:FeatureCollection xmlns:wfs="http://www.opengis.net/wfs/2.0"
xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" timeStamp="2008-09-07T19:00:00" numberReturned="1" numberMatched="1">
    <!-- omitted for brevity: the complete feature "CRANE 5" with all time slices is
returned here -->
  </wfs:member>
</wfs:FeatureCollection>
```

B.6 Dynamic Filter with time instant and with evaluation of properties with schedules

B.6.1 Request

```
<?xml version="1.0"?>
<wfs:GetFeature xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:wfs="http://www.opengis.net/wfs/2.0" xmlns:wfs-te="http://www.opengis.net/wfs-
te/1.0" xmlns:fes-te="http://www.opengis.net/fes-te/1.0"
xmlns:fes="http://www.opengis.net/fes/2.0" xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:aixm="http://www.aixm.aero/schema/5.1" service="WFS" version="2.0.0">
  <wfs-te:DynamicFeatureQuery typeNames="aixm:VerticalStructure">
    <fes-te:DynamicFeatureFilter evaluateSchedules="true">
      <fes-te:timeIndicator>
        <gml:TimeInstant gml:id="ID_1">
          <gml:timePosition>2012-10-02T15:30:00.000Z</gml:timePosition>
        </gml:TimeInstant>
      </fes-te:timeIndicator>
      <fes-te:featureFilter>
        <fes:Filter>
          <fes:And>
            <fes:PropertyIsEqualTo>
              <fes:ValueReference>gml:identifier</fes:ValueReference>
              <fes:Literal>8c755520-b42b-11e3-a5e2-0800500c9a66</fes:Literal>
            </fes:PropertyIsEqualTo>
            <fes:PropertyIsGreaterThan>

<fes:ValueReference>aixm:timeSlice/*/aixm:part/*/aixm:verticalExtent</fes:ValueReference>
              <fes:Literal>0</fes:Literal>
            </fes:PropertyIsGreaterThan>
          </fes:And>
        </fes:Filter>
      </fes-te:featureFilter>
    </fes-te:DynamicFeatureFilter>
  </wfs-te:DynamicFeatureQuery>
</wfs:GetFeature>
```

B.6.2 Response

```
<?xml version="1.0"?>
<wfs:FeatureCollection xmlns:wfs="http://www.opengis.net/wfs/2.0"
xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" timeStamp="2008-09-07T19:00:00" numberReturned="0" numberMatched="0">
</wfs:FeatureCollection>
```

B.7 Dynamic Filter with time period and with evaluation of properties with schedules

B.7.1 Request

```
<?xml version="1.0"?>
```

```
<wfs:GetFeature xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:wfs="http://www.opengis.net/wfs/2.0" xmlns:wfs-te="http://www.opengis.net/wfs-
te/1.0" xmlns:fes-te="http://www.opengis.net/fes-te/1.0"
xmlns:fes="http://www.opengis.net/fes/2.0" xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:aixm="http://www.aixm.aero/schema/5.1" service="WFS" version="2.0.0">
  <wfs-te:DynamicFeatureQuery typeNames="aixm:VerticalStructure">
    <fes-te:DynamicFeatureFilter evaluateSchedules="true">
      <fes-te:timeIndicator>
        <gml:TimePeriod gml:id="ID_1">
          <gml:beginPosition>2012-10-11T17:30:00Z</gml:beginPosition>
          <gml:endPosition>2012-10-20T04:29:00Z</gml:endPosition>
        </gml:TimePeriod>
      </fes-te:timeIndicator>
      <fes-te:featureFilter>
        <fes:Filter>
          <fes:And>
            <fes:PropertyIsEqualTo>
              <fes:ValueReference>gml:identifier</fes:ValueReference>
              <fes:Literal>8c755520-b42b-11e3-a5e2-0800500c9a66</fes:Literal>
            </fes:PropertyIsEqualTo>
            <fes:PropertyIsGreaterThan matchAction="All">

<fes:ValueReference>aixm:timeSlice/*/aixm:part/*/aixm:verticalExtent</fes:ValueReference>
              <fes:Literal>0</fes:Literal>
            </fes:PropertyIsGreaterThan>
          </fes:And>
        </fes:Filter>
      </fes-te:featureFilter>
    </fes-te:DynamicFeatureFilter>
  </wfs-te:DynamicFeatureQuery>
  <wfs-te:DynamicFeatureQuery typeNames="aixm:VerticalStructure">
    <fes-te:DynamicFeatureFilter evaluateSchedules="true">
      <fes-te:timeIndicator>
        <gml:TimePeriod gml:id="ID_2">
          <gml:beginPosition>2012-10-11T16:30:00Z</gml:beginPosition>
          <gml:endPosition>2012-10-20T04:29:00Z</gml:endPosition>
        </gml:TimePeriod>
      </fes-te:timeIndicator>
      <fes-te:featureFilter>
        <fes:Filter>
          <fes:And>
            <fes:PropertyIsEqualTo>
              <fes:ValueReference>gml:identifier</fes:ValueReference>
              <fes:Literal>8c755520-b42b-11e3-a5e2-0800500c9a66</fes:Literal>
            </fes:PropertyIsEqualTo>
            <fes:PropertyIsGreaterThan matchAction="All">

<fes:ValueReference>aixm:timeSlice/*/aixm:part/*/aixm:verticalExtent</fes:ValueReference>
              <fes:Literal>0</fes:Literal>
            </fes:PropertyIsGreaterThan>
          </fes:And>
        </fes:Filter>
      </fes-te:featureFilter>
    </fes-te:DynamicFeatureFilter>
  </wfs-te:DynamicFeatureQuery>
</wfs:GetFeature>
```

B.7.2 Response

```
<?xml version="1.0"?>
<wfs:FeatureCollection xmlns:wfs="http://www.opengis.net/wfs/2.0"
xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" timeStamp="2008-09-07T19:00:00" numberReturned="1" numberMatched="1">
  <wfs:member>
    <wfs:FeatureCollection timeStamp="2008-09-07T19:00:00" numberReturned="0"
numberMatched="0"/>
  </wfs:member>
  <wfs:member>
    <wfs:FeatureCollection timeStamp="2008-09-07T19:00:00" numberReturned="1"
numberMatched="1">
```

```
      <wfs:member>
        <!-- omitted for brevity: the complete feature "CRANE 5" with all time slices is
returned here -->
      </wfs:member>
    </wfs:FeatureCollection>
  </wfs:member>
</wfs:FeatureCollection>
```

B.8 TimeSliceProjection selecting time slices by interpretation

B.8.1 Request

```
<?xml version="1.0"?>
<wfs:GetFeature xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:wfs="http://www.opengis.net/wfs/2.0" xmlns:wfs-te="http://www.opengis.net/wfs-
te/1.0" xmlns:fes-te="http://www.opengis.net/fes-te/1.0"
xmlns:fes="http://www.opengis.net/fes/2.0" xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:aixm="http://www.aixm.aero/schema/5.1" service="WFS" version="2.0.0">
  <wfs-te:DynamicFeatureQuery typeNames="aixm:VerticalStructure">
    <wfs-te:TimeSliceProjection>
      <wfs-te:timeSliceFilter>
        <fes:Filter>
          <fes:PropertyIsEqualTo>
            <fes:ValueReference>aixm:interpretation
              </fes:ValueReference>
            <fes:Literal>TEMPDELTA</fes:Literal>
          </fes:PropertyIsEqualTo>
        </fes:Filter>
      </wfs-te:timeSliceFilter>
    </wfs-te:TimeSliceProjection>
    <fes-te:DynamicFeatureFilter>
      <fes-te:featureFilter>
        <fes:Filter>
          <fes:PropertyIsEqualTo>
            <fes:ValueReference>gml:identifier</fes:ValueReference>
            <fes:Literal>8c755520-b42b-11e3-a5e2-0800500c9a66</fes:Literal>
          </fes:PropertyIsEqualTo>
        </fes:Filter>
      </fes-te:featureFilter>
    </fes-te:DynamicFeatureFilter>
  </wfs-te:DynamicFeatureQuery>
</wfs:GetFeature>
```

B.8.2 Response

```
<?xml version="1.0"?>
<wfs:FeatureCollection xmlns:wfs="http://www.opengis.net/wfs/2.0"
xmlns:aixm="http://www.aixm.aero/schema/5.1" xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" timeStamp="2008-09-07T19:00:00"
numberReturned="1" numberMatched="1">
  <wfs:member>
    <aixm:VerticalStructure gml:id="ID_1">
      <gml:identifier codeSpace="urn:uuid:">8c755520-b42b-11e3-a5e2-
0800500c9a66</gml:identifier>
      <!-- omitted for brevity: featureMetadata and the two TEMPDELTA time slices -->
    </aixm:VerticalStructure>
  </wfs:member>
</wfs:FeatureCollection>
```

B.9 SnapshotGeneration on time instant without evaluation of schedules

B.9.1 Request

```
<?xml version="1.0"?>
<wfs:GetFeature xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:wfs="http://www.opengis.net/wfs/2.0" xmlns:wfs-te="http://www.opengis.net/wfs-
te/1.0" xmlns:fes-te="http://www.opengis.net/fes-te/1.0"
xmlns:fes="http://www.opengis.net/fes/2.0" xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:aixm="http://www.aixm.aero/schema/5.1" service="WFS" version="2.0.0">
  <wfs-te:DynamicFeatureQuery typeNames="aixm:VerticalStructure">
    <fes-te:DynamicFeatureFilter>
      <fes-te:featureFilter>
        <fes:Filter>
          <fes:PropertyIsEqualTo>
            <fes:ValueReference>gml:identifier</fes:ValueReference>
            <fes:Literal>8c755520-b42b-11e3-a5e2-0800500c9a66</fes:Literal>
          </fes:PropertyIsEqualTo>
        </fes:Filter>
      </fes-te:featureFilter>
    </fes-te:DynamicFeatureFilter>
    <wfs-te:transformation>
      <wfs-te:SnapshotGeneration>
        <wfs-te:snapshotTime>
          <gml:TimeInstant gml:id="ID_1">
            <gml:timePosition>2012-10-11T16:30:01.857Z</gml:timePosition>
          </gml:TimeInstant>
        </wfs-te:snapshotTime>
      </wfs-te:SnapshotGeneration>
    </wfs-te:transformation>
  </wfs-te:DynamicFeatureQuery>
</wfs:GetFeature>
```

B.9.2 Response

```
<?xml version="1.0"?>
<wfs:FeatureCollection xmlns:wfs="http://www.opengis.net/wfs/2.0"
xmlns:aixm="http://www.aixm.aero/schema/5.1" xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" timeStamp="2008-09-07T19:00:00"
numberReturned="1" numberMatched="1">
  <wfs:member>
    <aixm:VerticalStructure gml:id="ID_1">
      <gml:identifier codeSpace="urn:uuid:">8c755520-b42b-11e3-a5e2-
0800500c9a66</gml:identifier>
      <!-- omitted for brevity: featureMetadata -->
      <aixm:timeSlice>
        <aixm:VerticalStructureTimeSlice gml:id="ID_2">
          <gml:validTime>
            <gml:TimeInstant gml:id="ID_3">
              <gml:timePosition>2012-10-11T16:30:01.857Z</gml:timePosition>
            </gml:TimeInstant>
          </gml:validTime>
          <aixm:interpretation>SNAPSHOT</aixm:interpretation>
          <aixm:name>EADD_CRANE_5</aixm:name>
          <aixm:type>CRANE</aixm:type>
          <aixm:lighted>YES</aixm:lighted>
          <aixm:part>
            <aixm:VerticalStructurePart gml:id="ID_5">
              <aixm:timeInterval>
                <aixm:Timesheet gml:id="ID_6">
                  <aixm:timeReference>UTC</aixm:timeReference>
                  <aixm:day>ANY</aixm:day>
                  <aixm:startTime>04:30</aixm:startTime>
                  <aixm:endTime>17:29</aixm:endTime>
                  <aixm:daylightSavingAdjust>NO</aixm:daylightSavingAdjust>
                </aixm:Timesheet>
              </aixm:timeInterval>
              <aixm:verticalExtent uom="FT">85.0</aixm:verticalExtent>
              <aixm:type>CRANE</aixm:type>
              <aixm:mobile>YES</aixm:mobile>
              <aixm:horizontalProjection_location>
                <aixm:ElevatedPoint srsName="urn:ogc:def:crs:EPSG::4326" gml:id="ID_7">
                  <gml:pos>52.3727777777778 -31.9425</gml:pos>
                  <aixm:elevation uom="FT">858</aixm:elevation>
                </aixm:ElevatedPoint>
              </aixm:horizontalProjection_location>
            </aixm:VerticalStructurePart>
          </aixm:part>
          <aixm:part>
            <aixm:VerticalStructurePart gml:id="ID_8">
              <aixm:timeInterval>
                <aixm:Timesheet gml:id="ID_9">
                  <aixm:timeReference>UTC</aixm:timeReference>
                  <aixm:day>ANY</aixm:day>
                  <aixm:startTime>00:00</aixm:startTime>
                  <aixm:endTime>24:00</aixm:endTime>
                  <aixm:daylightSavingAdjust>NO</aixm:daylightSavingAdjust>
                </aixm:Timesheet>
              </aixm:timeInterval>
              <aixm:timeInterval>
                <aixm:Timesheet gml:id="ID_10">
                  <aixm:timeReference>UTC</aixm:timeReference>
                  <aixm:day>ANY</aixm:day>
                  <aixm:startTime>04:30</aixm:startTime>
                  <aixm:endTime>17:29</aixm:endTime>
                  <aixm:daylightSavingAdjust>NO</aixm:daylightSavingAdjust>
                  <aixm:excluded>YES</aixm:excluded>
                </aixm:Timesheet>
              </aixm:timeInterval>
              <aixm:verticalExtent uom="FT">0.0</aixm:verticalExtent>
              <aixm:type>CRANE</aixm:type>
              <aixm:mobile>YES</aixm:mobile>
              <aixm:horizontalProjection_location>
                <aixm:ElevatedPoint srsName="urn:ogc:def:crs:EPSG::4326" gml:id="ID_11">
                  <gml:pos>52.3727777777778 -31.9425</gml:pos>
                  <aixm:elevation uom="FT">858</aixm:elevation>
                </aixm:ElevatedPoint>
              </aixm:horizontalProjection_location>
            </aixm:VerticalStructurePart>
          </aixm:part>
        </aixm:VerticalStructureTimeSlice>
      </aixm:timeSlice>
    </aixm:VerticalStructure>
  </wfs:member>
</wfs:FeatureCollection>
```

B.10 SnapshotGeneration on time period without evaluation of schedules

B.10.1 Request

```
<?xml version="1.0"?>
<wfs:GetFeature xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:wfs="http://www.opengis.net/wfs/2.0" xmlns:wfs-te="http://www.opengis.net/wfs-
te/1.0" xmlns:fes-te="http://www.opengis.net/fes-te/1.0"
xmlns:fes="http://www.opengis.net/fes/2.0" xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:aixm="http://www.aixm.aero/schema/5.1" service="WFS" version="2.0.0">
  <wfs-te:DynamicFeatureQuery typeNames="aixm:VerticalStructure">
    <fes-te:DynamicFeatureFilter>
      <fes-te:featureFilter>
        <fes:Filter>
          <fes:PropertyIsEqualTo>
            <fes:ValueReference>gml:identifier</fes:ValueReference>
            <fes:Literal>8c755520-b42b-11e3-a5e2-0800500c9a66</fes:Literal>
          </fes:PropertyIsEqualTo>
        </fes:Filter>
      </fes-te:featureFilter>
    </fes-te:DynamicFeatureFilter>
    <wfs-te:transformation>
      <wfs-te:SnapshotGeneration>
        <wfs-te:snapshotTime>
          <gml:TimePeriod gml:id="ID_1">
            <gml:beginPosition>2012-09-01T05:31:00Z</gml:beginPosition>
            <gml:endPosition>2012-10-19T15:42:00Z</gml:endPosition>
          </gml:TimePeriod>
        </wfs-te:snapshotTime>
      </wfs-te:SnapshotGeneration>
    </wfs-te:transformation>
  </wfs-te:DynamicFeatureQuery>
</wfs:GetFeature>
```

B.10.2 Response

```
<?xml version="1.0"?>
<wfs:FeatureCollection xmlns:wfs="http://www.opengis.net/wfs/2.0"
xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:aixm="http://www.aixm.aero/schema/5.1"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" timeStamp="2008-09-07T19:00:00"
numberReturned="1" numberMatched="1">
  <wfs:member>
    <aixm:VerticalStructure gml:id="ID_1">
      <gml:identifier codeSpace="urn:uuid:">8c755520-b42b-11e3-a5e2-
0800500c9a66</gml:identifier>
      <!-- omitted for brevity: featureMetadata -->
      <aixm:timeSlice>
        <aixm:VerticalStructureTimeSlice gml:id="ID_2">
          <gml:validTime>
            <gml:TimePeriod gml:id="ID_3">
              <gml:beginPosition>2012-10-01T05:31:00Z</gml:beginPosition>
              <gml:endPosition>2012-10-11T15:30:00Z</gml:endPosition>
            </gml:TimePeriod>
          </gml:validTime>
          <aixm:interpretation>SNAPSHOT</aixm:interpretation>
          <aixm:name>EADD_CRANE_5</aixm:name>
          <aixm:type>CRANE</aixm:type>
          <aixm:lighted>YES</aixm:lighted>
          <aixm:part>
            <aixm:VerticalStructurePart gml:id="ID_5">
              <aixm:timeInterval>
                <aixm:Timesheet gml:id="ID_6">
                  <aixm:timeReference>UTC</aixm:timeReference>
                  <aixm:day>ANY</aixm:day>
                  <aixm:startTime>04:30</aixm:startTime>
                  <aixm:endTime>14:29</aixm:endTime>
                  <aixm:daylightSavingAdjust>NO</aixm:daylightSavingAdjust>
                </aixm:Timesheet>
              </aixm:timeInterval>
              <aixm:verticalExtent uom="FT">85.0</aixm:verticalExtent>
              <aixm:type>CRANE</aixm:type>
              <aixm:mobile>YES</aixm:mobile>
              <aixm:horizontalProjection_location>
                <aixm:ElevatedPoint srsName="urn:ogc:def:crs:EPSG::4326" gml:id="ID_7">
                  <gml:pos>52.3727777777778 -31.9425</gml:pos>
                  <aixm:elevation uom="FT">858</aixm:elevation>
                </aixm:ElevatedPoint>
              </aixm:horizontalProjection_location>
            </aixm:VerticalStructurePart>
          </aixm:part>
          <aixm:part>
            <aixm:VerticalStructurePart gml:id="ID_8">
              <aixm:timeInterval>
                <aixm:Timesheet gml:id="ID_9">
                  <aixm:timeReference>UTC</aixm:timeReference>
                  <aixm:day>ANY</aixm:day>
                  <aixm:startTime>00:00</aixm:startTime>
                  <aixm:endTime>24:00</aixm:endTime>
                  <aixm:daylightSavingAdjust>NO</aixm:daylightSavingAdjust>
                </aixm:Timesheet>
              </aixm:timeInterval>
              <aixm:timeInterval>
                <aixm:Timesheet gml:id="ID_10">
                  <aixm:timeReference>UTC</aixm:timeReference>
                  <aixm:day>ANY</aixm:day>
                  <aixm:startTime>04:30</aixm:startTime>
                  <aixm:endTime>14:29</aixm:endTime>
                  <aixm:daylightSavingAdjust>NO</aixm:daylightSavingAdjust>
                  <aixm:excluded>YES</aixm:excluded>
                </aixm:Timesheet>
              </aixm:timeInterval>
              <aixm:verticalExtent uom="FT">0.0</aixm:verticalExtent>
              <aixm:type>CRANE</aixm:type>
              <aixm:mobile>YES</aixm:mobile>
              <aixm:horizontalProjection_location>
                <aixm:ElevatedPoint srsName="urn:ogc:def:crs:EPSG::4326" gml:id="ID_11">
                  <gml:pos>52.3727777777778 -31.9425</gml:pos>
                  <aixm:elevation uom="FT">858</aixm:elevation>
                </aixm:ElevatedPoint>
              </aixm:horizontalProjection_location>
            </aixm:VerticalStructurePart>
          </aixm:part>
        </aixm:VerticalStructureTimeSlice>
      </aixm:timeSlice>
      <aixm:timeSlice>
        <aixm:VerticalStructureTimeSlice gml:id="ID_12">
          <gml:validTime>
            <gml:TimePeriod gml:id="ID_13">
              <gml:beginPosition>2012-10-11T15:30:00Z</gml:beginPosition>
              <gml:endPosition>2012-10-12T00:00:00Z</gml:endPosition>
            </gml:TimePeriod>
          </gml:validTime>
          <aixm:interpretation>SNAPSHOT</aixm:interpretation>
          <aixm:name>EADD_CRANE_5</aixm:name>
          <aixm:type>CRANE</aixm:type>
          <aixm:lighted>YES</aixm:lighted>
          <aixm:part>
            <aixm:VerticalStructurePart gml:id="ID_15">
              <aixm:timeInterval>
                <aixm:Timesheet gml:id="ID_16">
                  <aixm:timeReference>UTC</aixm:timeReference>
                  <aixm:day>ANY</aixm:day>
                  <aixm:startTime>04:30</aixm:startTime>
                  <aixm:endTime>17:29</aixm:endTime>
                  <aixm:daylightSavingAdjust>NO</aixm:daylightSavingAdjust>
                </aixm:Timesheet>
              </aixm:timeInterval>
              <aixm:verticalExtent uom="FT">85.0</aixm:verticalExtent>
              <aixm:type>CRANE</aixm:type>
              <aixm:mobile>YES</aixm:mobile>
              <aixm:horizontalProjection_location>
                <aixm:ElevatedPoint srsName="urn:ogc:def:crs:EPSG::4326" gml:id="ID_17">
                  <gml:pos>52.3727777777778 -31.9425</gml:pos>
                  <aixm:elevation uom="FT">858</aixm:elevation>
                </aixm:ElevatedPoint>
              </aixm:horizontalProjection_location>
            </aixm:VerticalStructurePart>
          </aixm:part>
          <aixm:part>
            <aixm:VerticalStructurePart gml:id="ID_18">
              <aixm:timeInterval>
                <aixm:Timesheet gml:id="ID_19">
                  <aixm:timeReference>UTC</aixm:timeReference>
                  <aixm:day>ANY</aixm:day>
                  <aixm:startTime>00:00</aixm:startTime>
                  <aixm:endTime>24:00</aixm:endTime>
                  <aixm:daylightSavingAdjust>NO</aixm:daylightSavingAdjust>
                </aixm:Timesheet>
              </aixm:timeInterval>
              <aixm:timeInterval>
                <aixm:Timesheet gml:id="ID_20">
                  <aixm:timeReference>UTC</aixm:timeReference>
                  <aixm:day>ANY</aixm:day>
                  <aixm:startTime>04:30</aixm:startTime>
                  <aixm:endTime>17:29</aixm:endTime>
                  <aixm:daylightSavingAdjust>NO</aixm:daylightSavingAdjust>
                  <aixm:excluded>YES</aixm:excluded>
                </aixm:Timesheet>
              </aixm:timeInterval>
              <aixm:verticalExtent uom="FT">0.0</aixm:verticalExtent>
              <aixm:type>CRANE</aixm:type>
              <aixm:mobile>YES</aixm:mobile>
              <aixm:horizontalProjection_location>
                <aixm:ElevatedPoint srsName="urn:ogc:def:crs:EPSG::4326" gml:id="ID_21">
                  <gml:pos>52.3727777777778 -31.9425</gml:pos>
                  <aixm:elevation uom="FT">858</aixm:elevation>
                </aixm:ElevatedPoint>
              </aixm:horizontalProjection_location>
            </aixm:VerticalStructurePart>
          </aixm:part>
        </aixm:VerticalStructureTimeSlice>
      </aixm:timeSlice>
      <aixm:timeSlice>
        <aixm:VerticalStructureTimeSlice gml:id="ID_22">
          <gml:validTime>
            <gml:TimePeriod gml:id="ID_23">
              <gml:beginPosition>2012-10-12T00:00:00Z</gml:beginPosition>
              <gml:endPosition>2012-10-19T15:42:00Z</gml:endPosition>
            </gml:TimePeriod>
          </gml:validTime>
          <aixm:interpretation>SNAPSHOT</aixm:interpretation>
          <aixm:name>EADD_CRANE_5</aixm:name>
          <aixm:type>CRANE</aixm:type>
          <aixm:lighted>YES</aixm:lighted>
          <aixm:part>
            <aixm:VerticalStructurePart gml:id="ID_25">
              <aixm:timeInterval>
                <aixm:Timesheet gml:id="ID_26">
                  <aixm:timeReference>UTC</aixm:timeReference>
                  <aixm:day>ANY</aixm:day>
                  <aixm:startTime>00:00</aixm:startTime>
                  <aixm:endTime>24:00</aixm:endTime>
                  <aixm:daylightSavingAdjust>NO</aixm:daylightSavingAdjust>
                </aixm:Timesheet>
              </aixm:timeInterval>
              <aixm:verticalExtent uom="FT">0.0</aixm:verticalExtent>
              <aixm:type>CRANE</aixm:type>
              <aixm:mobile>YES</aixm:mobile>
              <aixm:horizontalProjection_location>
                <aixm:ElevatedPoint srsName="urn:ogc:def:crs:EPSG::4326" gml:id="ID_27">
                  <gml:pos>52.3727777777778 -31.9425</gml:pos>
                  <aixm:elevation uom="FT">858</aixm:elevation>
                </aixm:ElevatedPoint>
              </aixm:horizontalProjection_location>
            </aixm:VerticalStructurePart>
          </aixm:part>
        </aixm:VerticalStructureTimeSlice>
      </aixm:timeSlice>
    </aixm:VerticalStructure>
  </wfs:member>
</wfs:FeatureCollection>
```

B.11 SnapshotGeneration on time instant with evaluation of schedules

B.11.1 Request

```
<?xml version="1.0"?>
<wfs:GetFeature xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:wfs="http://www.opengis.net/wfs/2.0" xmlns:wfs-te="http://www.opengis.net/wfs-
te/1.0" xmlns:fes-te="http://www.opengis.net/fes-te/1.0"
xmlns:fes="http://www.opengis.net/fes/2.0" xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:aixm="http://www.aixm.aero/schema/5.1" service="WFS" version="2.0.0">
  <wfs-te:DynamicFeatureQuery typeNames="aixm:VerticalStructure">
    <fes-te:DynamicFeatureFilter>
      <fes-te:featureFilter>
        <fes:Filter>
          <fes:PropertyIsEqualTo>
            <fes:ValueReference>gml:identifier</fes:ValueReference>
            <fes:Literal>8c755520-b42b-11e3-a5e2-0800500c9a66</fes:Literal>
          </fes:PropertyIsEqualTo>
        </fes:Filter>
      </fes-te:featureFilter>
    </fes-te:DynamicFeatureFilter>
    <wfs-te:transformation>
      <wfs-te:SnapshotGeneration evaluateSchedules="true">
        <wfs-te:snapshotTime>
          <gml:TimeInstant gml:id="ID_1">
            <gml:timePosition>2012-10-11T16:30:01.857Z</gml:timePosition>
          </gml:TimeInstant>
        </wfs-te:snapshotTime>
      </wfs-te:SnapshotGeneration>
    </wfs-te:transformation>
  </wfs-te:DynamicFeatureQuery>
</wfs:GetFeature>
```

B.11.2 Response

```
<?xml version="1.0"?>
<wfs:FeatureCollection xmlns:wfs="http://www.opengis.net/wfs/2.0"
xmlns:aixm="http://www.aixm.aero/schema/5.1" xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" timeStamp="2008-09-07T19:00:00"
numberReturned="1" numberMatched="1">
  <wfs:member>
    <aixm:VerticalStructure gml:id="ID_1">
      <gml:identifier codeSpace="urn:uuid:">8c755520-b42b-11e3-a5e2-
0800500c9a66</gml:identifier>
      <!-- omitted for brevity: featureMetadata -->
      <aixm:timeSlice>
        <aixm:VerticalStructureTimeSlice gml:id="ID_2">
          <gml:validTime>
            <gml:TimeInstant gml:id="ID_3">
              <gml:timePosition>2012-10-11T16:30:01.857Z</gml:timePosition>
            </gml:TimeInstant>
          </gml:validTime>
          <aixm:interpretation>SNAPSHOT</aixm:interpretation>
          <aixm:name>EADD_CRANE_5</aixm:name>
          <aixm:type>CRANE</aixm:type>
          <aixm:lighted>YES</aixm:lighted>
          <aixm:part>
            <aixm:VerticalStructurePart gml:id="ID_5">
              <aixm:timeInterval>
                <aixm:Timesheet gml:id="ID_6">
                  <aixm:timeReference>UTC</aixm:timeReference>
                  <aixm:day>ANY</aixm:day>
                  <aixm:startTime>04:30</aixm:startTime>
                  <aixm:endTime>17:29</aixm:endTime>
                  <aixm:daylightSavingAdjust>NO</aixm:daylightSavingAdjust>
                </aixm:Timesheet>
              </aixm:timeInterval>
              <aixm:verticalExtent uom="FT">85.0</aixm:verticalExtent>
              <aixm:type>CRANE</aixm:type>
              <aixm:mobile>YES</aixm:mobile>
              <aixm:horizontalProjection_location>
                <aixm:ElevatedPoint srsName="urn:ogc:def:crs:EPSG::4326" gml:id="ID_7">
                  <gml:pos>52.3727777777778 -31.9425</gml:pos>
                  <aixm:elevation uom="FT">858</aixm:elevation>
                </aixm:ElevatedPoint>
              </aixm:horizontalProjection_location>
            </aixm:VerticalStructurePart>
          </aixm:part>
        </aixm:VerticalStructureTimeSlice>
      </aixm:timeSlice>
    </aixm:VerticalStructure>
  </wfs:member>
</wfs:FeatureCollection>
```

B.12 SnapshotGeneration on time period with evaluation of schedules

B.12.1 Request

```
<?xml version="1.0"?>
<wfs:GetFeature xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:wfs="http://www.opengis.net/wfs/2.0" xmlns:wfs-te="http://www.opengis.net/wfs-
te/1.0" xmlns:fes-te="http://www.opengis.net/fes-te/1.0"
xmlns:fes="http://www.opengis.net/fes/2.0" xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:aixm="http://www.aixm.aero/schema/5.1" service="WFS" version="2.0.0">
  <wfs-te:DynamicFeatureQuery typeNames="aixm:VerticalStructure">
    <fes-te:DynamicFeatureFilter>
      <fes-te:featureFilter>
        <fes:Filter>
          <fes:PropertyIsEqualTo>
            <fes:ValueReference>gml:identifier</fes:ValueReference>
            <fes:Literal>8c755520-b42b-11e3-a5e2-0800500c9a66</fes:Literal>
          </fes:PropertyIsEqualTo>
        </fes:Filter>
      </fes-te:featureFilter>
    </fes-te:DynamicFeatureFilter>
    <wfs-te:transformation>
      <wfs-te:SnapshotGeneration evaluateSchedules="true">
        <wfs-te:snapshotTime>
          <gml:TimePeriod gml:id="ID_1">
            <gml:beginPosition>2012-10-11T13:31:00Z</gml:beginPosition>
            <gml:endPosition>2012-10-11T18:30:00Z</gml:endPosition>
          </gml:TimePeriod>
        </wfs-te:snapshotTime>
      </wfs-te:SnapshotGeneration>
    </wfs-te:transformation>
  </wfs-te:DynamicFeatureQuery>
</wfs:GetFeature>
```

B.12.2 Response

```
<?xml version="1.0"?>
<wfs:FeatureCollection xmlns:wfs="http://www.opengis.net/wfs/2.0"
xmlns:aixm="http://www.aixm.aero/schema/5.1" xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" timeStamp="2008-09-07T19:00:00"
numberReturned="1" numberMatched="1">
  <wfs:member>
    <aixm:VerticalStructure gml:id="ID_1">
      <gml:identifier codeSpace="urn:uuid:">8c755520-b42b-11e3-a5e2-
0800500c9a66</gml:identifier>
      <!-- omitted for brevity: featureMetadata -->
      <aixm:timeSlice>
        <aixm:VerticalStructureTimeSlice gml:id="ID_2">
          <gml:validTime>
            <gml:TimePeriod gml:id="ID_3">
              <gml:beginPosition>2012-10-11T13:31:00Z</gml:beginPosition>
              <gml:endPosition>2012-10-11T14:29:00Z</gml:endPosition>
            </gml:TimePeriod>
          </gml:validTime>
          <aixm:interpretation>SNAPSHOT</aixm:interpretation>
          <aixm:name>EADD_CRANE_5</aixm:name>
          <aixm:type>CRANE</aixm:type>
          <aixm:lighted>YES</aixm:lighted>
          <aixm:part>
            <aixm:VerticalStructurePart gml:id="ID_5">
              <aixm:timeInterval>
                <aixm:Timesheet gml:id="ID_6">
                  <aixm:timeReference>UTC</aixm:timeReference>
                  <aixm:day>ANY</aixm:day>
                  <aixm:startTime>04:30</aixm:startTime>
                  <aixm:endTime>14:29</aixm:endTime>
                  <aixm:daylightSavingAdjust>NO</aixm:daylightSavingAdjust>
                </aixm:Timesheet>
              </aixm:timeInterval>
              <aixm:verticalExtent uom="FT">85.0</aixm:verticalExtent>
              <aixm:type>CRANE</aixm:type>
              <aixm:mobile>YES</aixm:mobile>
              <aixm:horizontalProjection_location>
                <aixm:ElevatedPoint srsName="urn:ogc:def:crs:EPSG::4326" gml:id="ID_7">
                  <gml:pos>52.3727777777778 -31.9425</gml:pos>
                  <aixm:elevation uom="FT">858</aixm:elevation>
                </aixm:ElevatedPoint>
              </aixm:horizontalProjection_location>
            </aixm:VerticalStructurePart>
          </aixm:part>
        </aixm:VerticalStructureTimeSlice>
      </aixm:timeSlice>
      <aixm:timeSlice>
        <aixm:VerticalStructureTimeSlice gml:id="ID_8">
          <gml:validTime>
            <gml:TimePeriod gml:id="ID_9">
              <gml:beginPosition>2012-10-11T14:29:00Z</gml:beginPosition>
              <gml:endPosition>2012-10-11T15:30:00Z</gml:endPosition>
            </gml:TimePeriod>
          </gml:validTime>
          <aixm:interpretation>SNAPSHOT</aixm:interpretation>
          <aixm:name>EADD_CRANE_5</aixm:name>
          <aixm:type>CRANE</aixm:type>
          <aixm:lighted>YES</aixm:lighted>
          <aixm:part>
            <aixm:VerticalStructurePart gml:id="ID_11">
              <aixm:timeInterval>
                <aixm:Timesheet gml:id="ID_12">
                  <aixm:timeReference>UTC</aixm:timeReference>
                  <aixm:day>ANY</aixm:day>
                  <aixm:startTime>00:00</aixm:startTime>
                  <aixm:endTime>24:00</aixm:endTime>
                  <aixm:daylightSavingAdjust>NO</aixm:daylightSavingAdjust>
                </aixm:Timesheet>
              </aixm:timeInterval>
              <aixm:timeInterval>
                <aixm:Timesheet gml:id="ID_13">
                  <aixm:timeReference>UTC</aixm:timeReference>
                  <aixm:day>ANY</aixm:day>
                  <aixm:startTime>04:30</aixm:startTime>
                  <aixm:endTime>14:29</aixm:endTime>
                  <aixm:daylightSavingAdjust>NO</aixm:daylightSavingAdjust>
                  <aixm:excluded>YES</aixm:excluded>
                </aixm:Timesheet>
              </aixm:timeInterval>
              <aixm:verticalExtent uom="FT">0.0</aixm:verticalExtent>
              <aixm:type>CRANE</aixm:type>
              <aixm:mobile>YES</aixm:mobile>
              <aixm:horizontalProjection_location>
                <aixm:ElevatedPoint srsName="urn:ogc:def:crs:EPSG::4326" gml:id="ID_14">
                  <gml:pos>52.3727777777778 -31.9425</gml:pos>
                  <aixm:elevation uom="FT">858</aixm:elevation>
                </aixm:ElevatedPoint>
              </aixm:horizontalProjection_location>
            </aixm:VerticalStructurePart>
          </aixm:part>
        </aixm:VerticalStructureTimeSlice>
      </aixm:timeSlice>
      <aixm:timeSlice>
        <aixm:VerticalStructureTimeSlice gml:id="ID_15">
          <gml:validTime>
            <gml:TimePeriod gml:id="ID_16">
              <gml:beginPosition>2012-10-11T15:30:00Z</gml:beginPosition>
              <gml:endPosition>2012-10-11T17:29:00Z</gml:endPosition>
            </gml:TimePeriod>
          </gml:validTime>
          <aixm:interpretation>SNAPSHOT</aixm:interpretation>
          <aixm:name>EADD_CRANE_5</aixm:name>
          <aixm:type>CRANE</aixm:type>
          <aixm:lighted>YES</aixm:lighted>
          <aixm:part>
            <aixm:VerticalStructurePart gml:id="ID_18">
              <aixm:timeInterval>
                <aixm:Timesheet gml:id="ID_19">
                  <aixm:timeReference>UTC</aixm:timeReference>
                  <aixm:day>ANY</aixm:day>
                  <aixm:startTime>04:30</aixm:startTime>
                  <aixm:endTime>17:29</aixm:endTime>
                  <aixm:daylightSavingAdjust>NO</aixm:daylightSavingAdjust>
                </aixm:Timesheet>
              </aixm:timeInterval>
              <aixm:verticalExtent uom="FT">85.0</aixm:verticalExtent>
              <aixm:type>CRANE</aixm:type>
              <aixm:mobile>YES</aixm:mobile>
              <aixm:horizontalProjection_location>
                <aixm:ElevatedPoint srsName="urn:ogc:def:crs:EPSG::4326" gml:id="ID_20">
                  <gml:pos>52.3727777777778 -31.9425</gml:pos>
                  <aixm:elevation uom="FT">858</aixm:elevation>
                </aixm:ElevatedPoint>
              </aixm:horizontalProjection_location>
            </aixm:VerticalStructurePart>
          </aixm:part>
        </aixm:VerticalStructureTimeSlice>
      </aixm:timeSlice>
      <aixm:timeSlice>
        <aixm:VerticalStructureTimeSlice gml:id="ID_21">
          <gml:validTime>
            <gml:TimePeriod gml:id="ID_22">
              <gml:beginPosition>2012-10-11T17:29:00Z</gml:beginPosition>
              <gml:endPosition>2012-10-11T18:30:00Z</gml:endPosition>
            </gml:TimePeriod>
          </gml:validTime>
          <aixm:interpretation>SNAPSHOT</aixm:interpretation>
          <aixm:name>EADD_CRANE_5</aixm:name>
          <aixm:type>CRANE</aixm:type>
          <aixm:lighted>YES</aixm:lighted>
          <aixm:part>
            <aixm:VerticalStructurePart gml:id="ID_23">
              <aixm:timeInterval>
                <aixm:Timesheet gml:id="ID_24">
                  <aixm:timeReference>UTC</aixm:timeReference>
                  <aixm:day>ANY</aixm:day>
                  <aixm:startTime>00:00</aixm:startTime>
                  <aixm:endTime>24:00</aixm:endTime>
                  <aixm:daylightSavingAdjust>NO</aixm:daylightSavingAdjust>
                </aixm:Timesheet>
              </aixm:timeInterval>
```

```
              <aixm:timeInterval>
                <aixm:Timesheet gml:id="ID_25">
                  <aixm:timeReference>UTC</aixm:timeReference>
                  <aixm:day>ANY</aixm:day>
                  <aixm:startTime>04:30</aixm:startTime>
                  <aixm:endTime>17:29</aixm:endTime>
                  <aixm:daylightSavingAdjust>NO</aixm:daylightSavingAdjust>
                  <aixm:excluded>YES</aixm:excluded>
                </aixm:Timesheet>
              </aixm:timeInterval>
              <aixm:verticalExtent uom="FT">0.0</aixm:verticalExtent>
              <aixm:type>CRANE</aixm:type>
              <aixm:mobile>YES</aixm:mobile>
              <aixm:horizontalProjection_location>
                <aixm:ElevatedPoint srsName="urn:ogc:def:crs:EPSG::4326" gml:id="ID_26">
                  <gml:pos>52.3727777777778 -31.9425</gml:pos>
                  <aixm:elevation uom="FT">858</aixm:elevation>
                </aixm:ElevatedPoint>
              </aixm:horizontalProjection_location>
            </aixm:VerticalStructurePart>
          </aixm:part>
        </aixm:VerticalStructureTimeSlice>
      </aixm:timeSlice>
    </aixm:VerticalStructure>
  </wfs:member>
</wfs:FeatureCollection>
```

Annex C (normative) XML Schema

This standard defines an XML Schema implementation that is compliant to the UML conceptual models defined in sections 8 and 9. XML schemas were auto generated from the models following the encoding rules in Annex E of OGC Geography Markup Language v3.2 (ISO 19136:2007).

C.1 FES Temporality Extension Schema

Table 5 - Mapping of FES-TE UML classes to XML elements

FES-TE UML	FES-TE XML
DynamicFeatureFilter	fes-te:DynamicFeatureFilter

```
<?xml version="1.0" encoding="UTF-8"?><schema xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:fes="http://www.opengis.net/fes/2.0" xmlns:fes-te="http://www.opengis.net/fes-
te/1.0" xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:xlink="http://www.w3.org/1999/xlink" elementFormDefault="qualified"
targetNamespace="http://www.opengis.net/fes-te/1.0" version="1.0">
  <import namespace="http://www.opengis.net/fes/2.0"
schemaLocation="http://schemas.opengis.net/filter/2.0/filterAll.xsd"/>
  <import namespace="http://www.opengis.net/gml/3.2"
schemaLocation="http://schemas.opengis.net/gml/3.2.1/gml.xsd"/>
  <import namespace="http://www.w3.org/1999/xlink"
schemaLocation="http://www.w3.org/1999/xlink.xsd"/>
  <!--XML Schema document created by ShapeChange - http://shapechange.net/-->
  <element name="DynamicFeatureFilter" substitutionGroup="fes:AbstractSelectionClause"
type="fes-te:DynamicFeatureFilterType"/>
  <complexType name="DynamicFeatureFilterType">
    <complexContent>
      <extension base="fes:AbstractSelectionClauseType">
        <sequence>
          <element minOccurs="0" name="timeIndicator">
            <complexType>
              <sequence>
                <element ref="gml:AbstractTimeGeometricPrimitive"/>
              </sequence>
            </complexType>
          </element>
          <element name="featureFilter">
            <complexType>
              <sequence>
                <element ref="fes:Filter"/>
              </sequence>
            </complexType>
          </element>
        </sequence>
        <attribute default="false" name="evaluateSchedules" type="boolean"/>
      </extension>
    </complexContent>
  </complexType>
  <complexType name="DynamicFeatureFilterPropertyType">
    <sequence minOccurs="0">
      <element ref="fes-te:DynamicFeatureFilter"/>
    </sequence>
    <attributeGroup ref="xlink:simpleAttrs"/>
  </complexType>
</schema>
```

C.2 WFS Temporality Extension Schema

WFS-TE UML	WFS-TE XML
DynamicFeatureQuery	wfs-te:DynamicFeatureQuery
TimeSliceProjection	wfs-te:TimeSliceProjection
PropertyExclusion	wfs-te:PropertyExclusion
SnapshotGeneration	wfs-te:SnapshotGeneration

```
<?xml version="1.0" encoding="UTF-8"?><schema xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:fes="http://www.opengis.net/fes/2.0" xmlns:fes-te="http://www.opengis.net/fes-
te/1.0" xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:wfs="http://www.opengis.net/wfs/2.0" xmlns:wfs-te="http://www.opengis.net/wfs-
te/1.0" xmlns:xlink="http://www.w3.org/1999/xlink" elementFormDefault="qualified"
targetNamespace="http://www.opengis.net/wfs-te/1.0" version="1.0">
  <import namespace="http://www.opengis.net/fes-te/1.0"
schemaLocation="fesTemporality.xsd"/>
  <import namespace="http://www.opengis.net/fes/2.0"
schemaLocation="http://schemas.opengis.net/filter/2.0/filterAll.xsd"/>
  <import namespace="http://www.opengis.net/gml/3.2"
schemaLocation="http://schemas.opengis.net/gml/3.2.1/gml.xsd"/>
  <import namespace="http://www.opengis.net/wfs/2.0"
schemaLocation="http://schemas.opengis.net/wfs/2.0/wfs.xsd"/>
  <import namespace="http://www.w3.org/1999/xlink"
schemaLocation="http://www.w3.org/1999/xlink.xsd"/>
  <!--XML Schema document created by ShapeChange - http://shapechange.net/-->
  <element abstract="true" name="AbstractTransformationClause" type="wfs-
te:AbstractTransformationClauseType"/>
  <complexType abstract="true" name="AbstractTransformationClauseType">
    <sequence/>
  </complexType>
  <complexType name="AbstractTransformationClausePropertyType">
    <sequence minOccurs="0">
      <element ref="wfs-te:AbstractTransformationClause"/>
    </sequence>
    <attributeGroup ref="xlink:simpleAttrs"/>
  </complexType>
  <element name="DynamicFeatureQuery" substitutionGroup="wfs:Query" type="wfs-
te:DynamicFeatureQueryType"/>
  <complexType name="DynamicFeatureQueryType">
    <complexContent>
      <extension base="wfs:QueryType">
        <sequence>
          <element minOccurs="0" name="transformation" type="wfs-
te:AbstractTransformationClausePropertyType"/>
        </sequence>
      </extension>
    </complexContent>
  </complexType>
  <complexType name="DynamicFeatureQueryPropertyType">
    <sequence minOccurs="0">
      <element ref="wfs-te:DynamicFeatureQuery"/>
    </sequence>
    <attributeGroup ref="xlink:simpleAttrs"/>
  </complexType>
  <element name="PropertyExclusion" substitutionGroup="fes:AbstractProjectionClause"
type="wfs-te:PropertyExclusionType"/>
  <complexType name="PropertyExclusionType">
    <complexContent>
```

```
      <extension base="fes:AbstractProjectionClauseType">
        <sequence>
          <element name="propertyName" type="QName"/>
        </sequence>
      </extension>
    </complexContent>
  </complexType>
  <complexType name="PropertyExclusionPropertyType">
    <sequence minOccurs="0">
      <element ref="wfs-te:PropertyExclusion"/>
    </sequence>
    <attributeGroup ref="xlink:simpleAttrs"/>
  </complexType>
  <element name="SnapshotGeneration" substitutionGroup="wfs-
te:AbstractTransformationClause" type="wfs-te:SnapshotGenerationType"/>
  <complexType name="SnapshotGenerationType">
    <complexContent>
      <extension base="wfs-te:AbstractTransformationClauseType">
        <sequence>
          <element name="snapshotTime">
            <complexType>
              <sequence>
                <element ref="gml:AbstractTimeGeometricPrimitive"/>
              </sequence>
            </complexType>
          </element>
        </sequence>
        <attribute default="false" name="evaluateSchedules" type="boolean"/>
      </extension>
    </complexContent>
  </complexType>
  <complexType name="SnapshotGenerationPropertyType">
    <sequence minOccurs="0">
      <element ref="wfs-te:SnapshotGeneration"/>
    </sequence>
    <attributeGroup ref="xlink:simpleAttrs"/>
  </complexType>
  <element name="TimeSliceProjection" substitutionGroup="fes:AbstractProjectionClause"
type="wfs-te:TimeSliceProjectionType"/>
  <complexType name="TimeSliceProjectionType">
    <complexContent>
      <extension base="fes:AbstractProjectionClauseType">
        <sequence>
          <element minOccurs="0" name="relevantTime">
            <complexType>
              <sequence>
                <element ref="gml:AbstractTimeGeometricPrimitive"/>
              </sequence>
            </complexType>
          </element>
          <element minOccurs="0" name="timeSliceFilter">
            <complexType>
              <sequence>
                <element ref="fes:Filter"/>
              </sequence>
            </complexType>
          </element>
        </sequence>
        <attribute default="false" name="includeCanceled" type="boolean"/>
        <attribute default="false" name="includeCorrected" type="boolean"/>
      </extension>
    </complexContent>
  </complexType>
  <complexType name="TimeSliceProjectionPropertyType">
    <sequence minOccurs="0">
      <element ref="wfs-te:TimeSliceProjection"/>
    </sequence>
    <attributeGroup ref="xlink:simpleAttrs"/>
  </complexType>
</schema>
```

Annex D (normative) UML Models

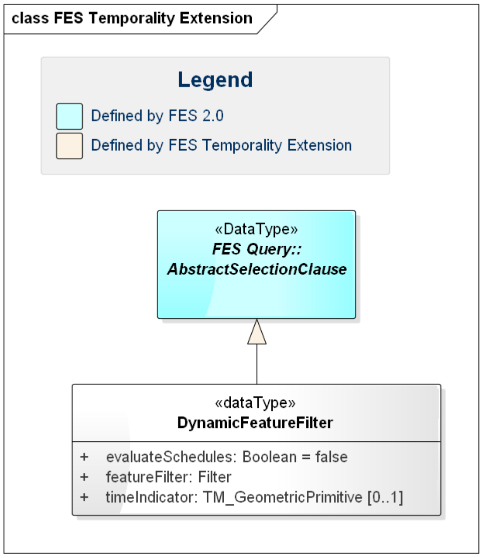

class FES Temporality Extension
Legend
Defined by FES 2.0
Defined by FES Temporality Extension
«DataType»
FES Query::
AbstractSelectionClause
«dataType»
DynamicFeatureFilter
+ evaluateSchedules: Boolean = false
+ featureFilter: Filter
+ timeIndicator: TM_GeometricPrimitive [0..1]

Figure 9: FES Temporality Extension Model

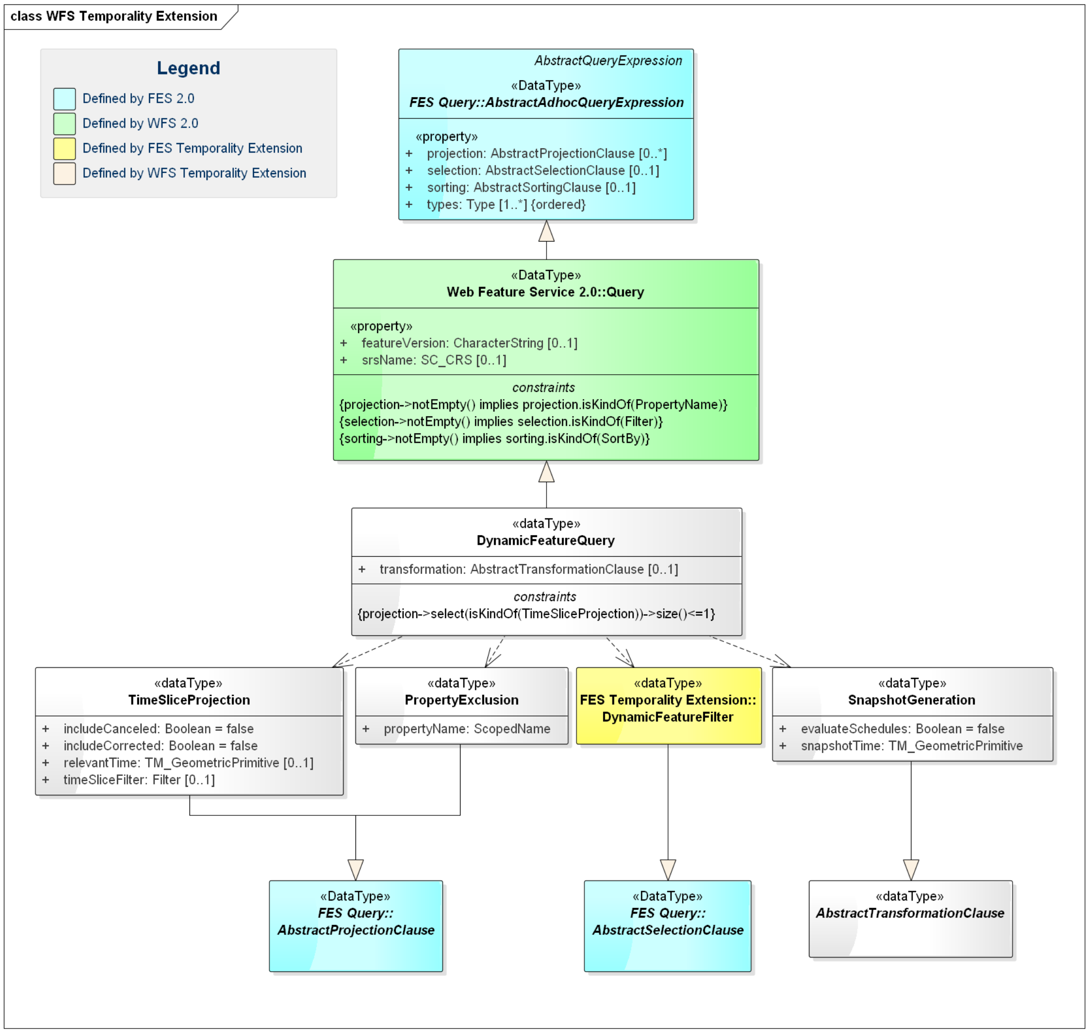

class WFS Temporality Extension
Legend
Defined by FES 2.0
Defined by WFS 2.0
Defined by FES Temporality Extension
Defined by WFS Temporality Extension
AbstractQueryExpression
«DataType»
FES Query::AbstractAdhocQueryExpression
«property»
+ projection: AbstractProjectionClause [0..*]
+ selection: AbstractSelectionClause [0..1]
+ sorting: AbstractSortingClause [0..1]
+ types: Type [1..*] {ordered}
«DataType»
Web Feature Service 2.0::Query
«property»
+ featureVersion: CharacterString [0..1]
+ srsName: SC_CRS [0..1]
constraints
{projection->notEmpty() implies projection.isKindOf(PropertyName)}
{selection->notEmpty() implies selection.isKindOf(Filter)}
{sorting->notEmpty() implies sorting.isKindOf(SortBy)}
«dataType»
DynamicFeatureQuery
+ transformation: AbstractTransformationClause [0..1]
constraints
{projection->select(isKindOf(TimeSliceProjection))->size()<=1}
«dataType»
TimeSliceProjection
+ includeCanceled: Boolean = false
+ includeCorrected: Boolean = false
+ relevantTime: TM_GeometricPrimitive [0..1]
+ timeSliceFilter: Filter [0..1]
«dataType»
PropertyExclusion
+ propertyName: ScopedName
«dataType»
FES Temporality Extension::
DynamicFeatureFilter
«dataType»
SnapshotGeneration
+ evaluateSchedules: Boolean = false
+ snapshotTime: TM_GeometricPrimitive
«DataType»
FES Query::
AbstractProjectionClause
«DataType»
FES Query::
AbstractSelectionClause
«dataType»
AbstractTransformationClause

Figure 10: WFS Temporality Extension Model

Annex E (normative) Abstract Conformance Test Suite

E.1 FES Temporality Extension

E.1.1 Conformance Class: FES-TE Core

Conformance Class		
http://www.opengis.net/spec/fes-te/1.0/conf/core		
Requirements Class	**http://www.opengis.net/spec/fes-te/1.0/req/core**	
Test	**/conf/core/capability**	
	Requirement	/req/core/capability

	Test purpose	Verify that this requirement is satisfied
	Test method	Ensure that the Capabilities document of the service lists the conformance class identifier as an ows:Profile element in its ServiceIdentification section
	Test type	Basic
Test	**/conf/core/dynamic-feature-filter**	
	Requirement	/req/core/dynamic-feature-filter
	Test purpose	Verify that this requirement is satisfied
	Test method	Execute a valid request containing a DynamicFeatureFilter. Ensure that no exception is returned.
	Test type	Basic
Test	**/conf/core/dynamic-feature-filter-on-aixm-features**	
	Requirement	/req/core/dynamic-feature-filter-on-aixm-features
	Test purpose	Verify that this requirement is satisfied
	Test method	For *corrections* according to AIXM-TM: Execute a valid request containing a DynamicFeatureFilter, targeted on an AIXM feature type. The DynamicFeatureFilter has no time indicator set, and contains an equality constraint on a property ("timeslice/propA = 'x'"). The feature store of the service does contain a single feature with timeslice/propA = 'x', and that time slice is *corrected* with a different value than 'x' according to the AIXM-TM. Ensure that the server does not return the feature. See example B.2. For *cancelations* according to AIXM-TM: a) canceled data: same setting as above, except that the equality constraint targets data on a *canceled* time slice. Ensure that the server does not return the feature. See example

		B.3. b) canceled commissioning: request a feature by its gml:identifier. The feature store contains such a feature, but its commissioning was canceled. Ensure that the server does not return the feature. See example B.1.
	Test type	Basic
Test	**/conf/core/dynamic-feature-filter-on-static-features**	
	Requirement	/req/core/dynamic-feature-filter-on-static-features
	Test purpose	Verify that this requirement is satisfied
	Test method	Execute a valid request containing a DynamicFeatureFilter, targeted on a static (non-dynamic) feature type. Ensure that the service behaves as defined by WFS 2.0, as if a Filter used instead of DynamicFeatureFilter.
	Test type	Basic
Test	**/conf/core/dynamic-feature-filter-with-time-instant-time-indicator**	
	Requirement	/req/core/dynamic-feature-filter-with-time-instant-time-indicator
	Test purpose	Verify that this requirement is satisfied
	Test method	Given time instants t1 – t5, where t1 < t2 < t3 < t4 < t5 Fill the feature store of the service with two AIXM features: feature A with a time slice property ‘propA’ of value: ‘x’, valid from t1-t2 and t4-t5 and value ‘y’, valid from t2-t4. feature B with a time slice property ‘propA’ of value: ‘y’, valid from t1-t2 and t4-t5 and value ‘x’, valid from t2-t4. Execute a valid request containing a DynamicFeatureFilter with timeIndicator set to t3

		and an equality constraint “timeslice/propA = ‘x’”. Ensure that the service returns B in its result set, but not feature A. See also example B.4
	Test type	Basic
Test	**/conf/core/dynamic-feature-filter-with-time-period-time-indicator**	
	Requirement	/req/core/dynamic-feature-filter-with-time-period-time-indicator
	Test purpose	Verify that this requirement is satisfied
	Test method	Given 5 time instants t1-t5, where t1 < t2 < t3 < t4 < t5 Fill the feature store of the service with two AIXM features: feature A with a time slice property ‘propA’ of value: ‘x’, valid from t1-t2 and t4-t5 and value ‘y’, valid from t2-t4. feature B with a time slice property ‘propA’ of value: ‘y’, valid from t1-t2 and t4-t5 and value ‘x’, valid from t2-t4. Execute a valid request containing a DynamicFeatureFilter with timeIndicator set to t2-t4 (exclusive) and an equality constraint “timeslice/propA = ‘x’”. Ensure that the service returns B in its result set, but not feature A. Execute a valid request containing a DynamicFeatureFilter with timeIndicator set to t1-t3 and an equality constraint “timeslice/propA = ‘x’”. Ensure that the service returns features A and B.

		See example B.5
	Test type	Basic
Test	**/conf/core/dynamic-feature-filter-with-evaluate-schedules-true**	
	Requirement	/req/core/dynamic-feature-filter-with-evaluate-schedules-true
	Test purpose	Verify that this requirement is satisfied
	Test method	1. Execute a valid request □ containing a DynamicFeatureFilter □ with timeIndicator set to a time instant and □ evaluateSchedules set to true against a service not implementing http://www.opengis.net/spec/wfs-te/1.0/conf/time-instant-schedule-evaluation-filter. Ensure the service returns an exception as defined by the requirements class. 2. Execute a valid request □ containing a DynamicFeatureFilter □ with timeIndicator set to a time period and □ evaluateSchedules set to true against a service not implementing http://www.opengis.net/spec/wfs-te/1.0/conf/time-period-schedule-evaluation-filter. Ensure the service returns an exception as defined by the requirements class.
	Test type	Basic

E.1.2 Conformance Class: FES-TE Time Instant Schedule Evaluation For Filter

Conformance Class	
http://www.opengis.net/spec/wfs-te/1.0/conf/time-instant-schedule-evaluation-filter	
Requirements Class	http://www.opengis.net/spec/wfs-te/1.0/req/time-instant-schedule-evaluation-filter

Test	**/conf/time-instant-schedule-evaluation-filter/capability**	
	Requirement	/req/time-period-schedule-evaluation-filter/capability
	Test purpose	Verify that this requirement is satisfied
	Test method	Ensure that the Capabilities document of the service lists the conformance class identifier as an ows:Profile element in its ServiceIdentification section
	Test type	Basic
Test	**/conf/time-instant-schedule-evaluation-filter/evaluate-schedules**	
	Requirement	/req/time-instant-schedule-evaluation-snapshots
	Test purpose	Verify that this requirement is satisfied
	Test method	Given □ time instants t_i, where t1 < t2 < t3 < t3‘ < t4 < t5, t3 falls on MO (Monday) and t3‘ falls on SA (Saturday) □ schedule s1: MO-FR value ‚x‘, SA-SU value ‚y‘ □ schedule s2: MO-FR value ‚y‘, SA-SU value ‚x‘ Fill the feature store of the service with two AIXM features: feature A with a time slice property ‘propA’ with schedule s1, valid from t1-t2 and t4-t5 and schedule s2, valid from t2-t4. feature B with a time slice property ‘propA’ with schedule s2, valid from t1-t2 and t4-t5 and schedule s1, valid from t2-t4. Execute a valid request containing a DynamicFeatureFilter with timeIndicator set to t3, evaluateSchedules set to true and an equality constraint “timeslice/propA = ‘x’”.

		Ensure that the service returns B in its result set, but not feature A. Execute a valid request containing a DynamicFeatureFilter with timeIndicator set to t3’, evaluateSchedules set to true and an equality constraint “timeslice/propA = ‘x’”. Ensure that the service returns A in its result set, but not feature B. See also example B.6
	Test type	Basic

E.1.3 Conformance Class: FES-TE Time Period Schedule Evaluation For Filter

Conformance Class		
http://www.opengis.net/spec/wfs-te/1.0/conf/time-period-schedule-evaluation-filter		
Requirements Class	http://www.opengis.net/spec/wfs-te/1.0/req/time-period-schedule-evaluation-filter	
Test	**/conf/time-period-schedule-evaluation-filter/capability**	
	Requirement	/req/time-period-schedule-evaluation-filter/capability
	Test purpose	Verify that this requirement is satisfied
	Test method	Ensure that the Capabilities document of the service lists the conformance class identifier as an ows:Profile element in its ServiceIdentification section
	Test type	Basic
Test	**/conf/time-period-schedule-evaluation-filter/evaluate-schedules**	
	Requirement	/req/time-period-schedule-evaluation-snapshots
	Test purpose	Verify that this requirement is satisfied
	Test method	Given □ time instants t_i, where t1 < t2 < t3 < t3‘ < t4 < t5, t2 falls on SU (Sunday), t3 falls on the

		following MO (Monday), t3‘ falls on the next SA (Saturday) and t4 falls on the next TU (Tuesday) □ schedule s1: MO-FR value ‚x‘, SA-SU value ‚y‘ □ schedule s2: MO-FR value ‚y‘, SA-SU value ‚x‘ Fill the feature store of the service with two AIXM features: feature A with a time slice property ‘propA’ with schedule s1, valid from t1-t2 and t4-t5 and schedule s2, valid from t2-t4. feature B with a time slice property ‘propA’ with schedule s2, valid from t1-t2 and t4-t5 and schedule s1, valid from t2-t4. Execute a valid request containing a DynamicFeatureFilter with timeIndicator set to t3’-t4, evaluateSchedules set to true and an equality constraint “timeslice/propA = ‘x’” with matchAction set to all. Ensure that the service returns A in its result set, but not feature B. Execute a valid request containing a DynamicFeatureFilter with timeIndicator set to t3’, evaluateSchedules set to true and an equality constraint “timeslice/propA = ‘x’”. Ensure that the service returns A in its result set, but not feature B. See example B.7.
	Test type	Basic

E.2 WFS Temporality Extension

E.2.1 Conformance Class: WFS-TE Core

Conformance Class		
http://www.opengis.net/spec/wfs-te/1.0/conf/core		
Requirements Class	http://www.opengis.net/spec/wfs-te/1.0/req/core	
Test	**/conf/core/capability**	
	Requirement	/req/core/capability
	Test purpose	Verify that this requirement is satisfied
	Test method	Ensure that the Capabilities document of the service lists the conformance class identifier as an ows:Profile element in its ServiceIdentification section
	Test type	Basic
Test	**/conf/core/restrict-number-of-time-slice-projections**	
	Requirement	/req/core/restrict-number-of-time-slice-projections
	Test purpose	Verify that this requirement is satisfied
	Test method	Execute a request containing a DynamicFeatureQuery with two TimeSliceProjection elements. Ensure that an 'InvalidParameterValue' exception is returned.
	Test type	Basic
Test	**/conf/core/dynamic-feature-query**	
	Requirement	/req/core/dynamic-feature-query
	Test purpose	Verify that this requirement is satisfied
	Test method	Execute a valid request containing a DynamicFeatureQuery. Ensure that no exception is returned.

	Test type	Basic
Test	**/conf/core/include-optional-properties**	
	Requirement	/req/core/include-optional-properties
	Test purpose	Verify that this requirement is satisfied
	Test method	Execute a valid request containing a DynamicFeatureQuery and no PropertyExclusion clauses. The request shall target AIXM features that contain some non-mandatory data, e.g. feature metadata. Ensure that the response contains the feature metadata in all features returned.
	Test type	Basic
Test	**/conf/core/transformation-clause**	
	Requirement	/req/core/transformation-clause
	Test purpose	Execute a valid request containing a transformation clause, e.g. a SnapshotGeneration clause. Ensure that the transformation clause is applied. See example B.9.
	Test method	Inspect the service to verify the above
	Test type	Basic

E.2.2 Conformance Class: WFS-TE Time Slice Projection

Conformance Class		
http://www.opengis.net/spec/wfs-te/1.0/conf/time-slice-projection		
Requirements Class	http://www.opengis.net/spec/wfs-te/1.0/req/time-slice-projection	
Test	**/conf/time-slice-projection/capability**	
	Requirement	/req/time-slice-projection/capability
	Test purpose	Verify that this requirement is satisfied

	Test method	Ensure that the Capabilities document of the service lists the conformance class identifier as an ows:Profile element in its ServiceIdentification section
	Test type	Basic
Test	**/conf/time-slice-projection/time-slice-filter**	
	Requirement	/req/time-slice-projection/time-slice-filter
	Test purpose	Verify that this requirement is satisfied
	Test method	Given a feature with a history of time slices consisting of TEMPDELTA and PERMDELTA time slices. Execute a request containing a feature filter matching on the feature’s gml:identifier, and a time slice filter, selecting TEMPDELTA time slices only. Ensure that the response contains that feature and its list of time slices only contains TEMPDELTA time slices. See example B.8.
	Test type	Basic
Test	**/conf/time-slice-projection/relevant-time**	
	Requirement	/req/time-slice-projection/relevant-time
	Test purpose	Verify that this requirement is satisfied
	Test method	Given a feature with a history of time slices consisting of TEMPDELTA and PERMDELTA time slices. Execute a request containing a feature filter matching on the feature’s gml:identifier, and a time slice filter with relevant time set to time instant *t*. Ensure that response contains the feature and only those time slices that are relevant for feature state at time *t*.
	Test type	Basic
Test	**/conf/time-slice-projection/include-corrected**	
	Requirement	/req/time-slice-projection/include-corrected
	Test purpose	Verify that this requirement is satisfied

	Test method	Given a feature with a history of time slices consisting of TEMPDELTA and PERMDELTA time slices, of which some a *corrected* according to AIXM-TM. Execute a request containing a feature filter matching on the feature's gml:identifier, and a time slice filter selecting all TEMPDELTA time slices with includeCorrected set to true (false). Ensure that the response contains the feature and its list of time slices contains also (no) TEMPDELTA time slices that are *corrected*.
	Test type	Basic
Test	**/conf/time-slice-projection/include-canceled**	
	Requirement	/req/time-slice-projection/include-canceled
	Test purpose	Verify that this requirement is satisfied
	Test method	Given a feature with a history of time slices consisting of TEMPDELTA and PERMDELTA time slices, of which some a *canceled* according to AIXM-TM. Execute a request containing a feature filter matching on the feature's gml:identifier, and a time slice filter selecting all TEMPDELTA time slices with includeCanceled set to true (false). Ensure that the response contains the feature and its list of time slices contains also (no) TEMPDELTA time slices that are *canceled*.
	Test type	Basic

E.2.3 Conformance Class: WFS-TE Snapshot Generation

Conformance Class		
http://www.opengis.net/spec/wfs-te/1.0/conf/snapshot-generation		
Requirements Class	http://www.opengis.net/spec/wfs-te/1.0/req/snapshot-generation	
Test	**/conf/snapshot-generation/capability**	
	Requirement	/req/snapshot-generation/capability

	Test purpose	Verify that this requirement is satisfied
	Test method	Ensure that the Capabilities document of the service lists the conformance class identifier as an ows:Profile element in its ServiceIdentification section
	Test type	Basic
Test	**/conf/snapshot-generation/time-instant**	
	Requirement	/req/snapshot-generation/time-instant
	Test purpose	Verify that this requirement is satisfied
	Test method	Execute a request containing feature filter on a feature's gml:identifier and a snapshot generation clause at time instant *t*. Let the feature's life time contain *t*. Ensure that the response contains the feature and SNAPSHOT time slice at time *t*. See example B.9.
	Test type	Basic
Test	**/conf/snapshot-generation/time-period**	
	Requirement	/req/snapshot-generation/time-period
	Test purpose	Verify that this requirement is satisfied
	Test method	Execute a request containing feature filter on a feature's gml:identifier and a snapshot generation clause at time period *t1-t2*. Let the feature's life time contain the time period in full. Ensure that the response contains the feature and one or more SNAPSHOT time slices, depending on if changes on property values exist in that period. The first SNAPSHOT time slice has its valid time begin position set to *t1* and the last SNAPSHOT time slice has its valid time end position set to *t2*. See example B.10.
	Test type	Basic
Test	**/conf/snapshot-generation/evaluate-schedules**	
	Requirement	/req/snapshot-generation/evaluate-schedules

	Test purpose	Verify that this requirement is satisfied
	Test method	1. Execute a valid request □ containing a DynamicFeatureFilter □ with a SnapshotGeneration transformation clause with a snapshotTime set to a time instant and evaluateSchedules set to true against a service not implementing http://www.opengis.net/spec/wfs-te/1.0/conf/time-instant-schedule-evaluation-snapshots. Ensure the service returns an exception as defined by the requirements class. 2. Execute a valid request □ containing a DynamicFeatureFilter □ with a SnapshotGeneration transformation clause with a snapshotTime set to a time period and evaluateSchedules set to true against a service not implementing http://www.opengis.net/spec/wfs-te/1.0/conf/time-period-schedule-evaluation-snapshots. Ensure the service returns an exception as defined by the requirements class.
	Test type	Basic

E.2.4 Conformance Class: WFS-TE Time Instant Schedule Evaluation For Snapshots

Conformance Class		
http://www.opengis.net/spec/wfs-te/1.0/conf/time-instant-schedule-evaluation-snapshots		
Requirements Class	http://www.opengis.net/spec/wfs-te/1.0/req/time-instant-schedule-evaluation-snapshots	
Test	**/conf/time-instant-schedule-evaluation-snapshots/capability**	
	Requirement	/req/time-instant-schedule-evaluation-snapshots/capability

	Test purpose	Verify that this requirement is satisfied
	Test method	Ensure that the Capabilities document of the service lists the conformance class identifier as an ows:Profile element in its ServiceIdentification section
	Test type	Basic
Test	**/conf/time-instant-schedule-evaluation-snapshots/evaluate-schedules**	
	Requirement	/req/time-instant-schedule-evaluation-snapshots /evaluate-schedules
	Test purpose	Verify that this requirement is satisfied
	Test method	Execute a valid request □ containing a DynamicFeatureFilter □ containing a feature filter on an existing AIXM feature's gml:identifier □ with a SnapshotGeneration transformation clause with a snapshotTime set to a time instant *t* and evaluateSchedules set to true The targeted feature shall contain a property with schedule *p* with multiple instances. Let the feature's life time contain *t*. Ensure that the response contains that feature with a SNAPSHOT time slice valid at time *t* and containing only that instance of property *p* that is valid at time *t*. See example B.11.
	Test type	Basic

E.2.5 Conformance Class: WFS-TE Time Period Schedule Evaluation For Snapshots

Conformance Class	
http://www.opengis.net/spec/wfs-te/1.0/conf/time-period-schedule-evaluation-snapshots	
Requirements Class	http://www.opengis.net/spec/wfs-te/1.0/req/time-period-schedule-evaluation-snapshots

Test	**/conf/time-period-schedule-evaluation-snapshots/capability**	
	Requirement	/req/time-period-schedule-evaluation-snapshots/capability
	Test purpose	Verify that this requirement is satisfied
	Test method	Ensure that the Capabilities document of the service lists the conformance class identifier as an ows:Profile element in its ServiceIdentification section
	Test type	Basic
Test	**/conf/time-period-schedule-evaluation-snapshots/evaluate-schedules**	
	Requirement	/req/time-period-schedule-evaluation-snapshots /evaluate-schedules
	Test purpose	Verify that this requirement is satisfied
	Test method	Execute a valid request □ containing a DynamicFeatureFilter □ containing a feature filter on an existing AIXM feature's gml:identifier □ with a SnapshotGeneration transformation clause with a snapshotTime set to a time period *t1*-*t2* and evaluateSchedules set to true The targeted feature shall contain a property with schedule *p* with multiple instances. Let the feature's life time contain the time period in full. Ensure that the response contains that feature with one or more SNAPSHOT time slices, depending on if changes on property values exist in that period, including changes due to properties with schedules. The first SNAPSHOT time slice has its valid time begin position set to *t1* and the last SNAPSHOT time slice has its valid time end position set to *t2*. For each property with schedule in the SNAPSHOT time slices there is only one instance.

		See example B.12.
	Test type	Basic

E.2.6 Conformance Class: WFS-TE Property Exclusion

Conformance Class		
http://www.opengis.net/spec/wfs-te/1.0/conf/property-exclusion		
Requirements Class	http://www.opengis.net/spec/wfs-te/1.0/req/property-exclusion	
Test	**/conf/property-exclusion/capability**	
	Requirement	/req/property-exclusion/capability
	Test purpose	Verify that this requirement is satisfied
	Test method	Ensure that the Capabilities document of the service lists the conformance class identifier as an ows:Profile element in its ServiceIdentification section
	Test type	Basic
Test	**/conf/property-exclusion/property-name**	
	Requirement	/req/property-exclusion/property-name
	Test purpose	Verify that this requirement is satisfied
	Test method	Execute a valid request containing a DynamicFeatureQuery and a PropertyExclusion clause selecting featureMetadata elements. The request shall target AIXM features that contain some featureMetadata. Ensure that the response contains no featureMetadata elements in all features returned.
	Test type	Basic