

**xSNOWTAM-TEC**

***AIXM COMPLEX TOPICS***

Version 0.1

**DOCUMENT DISTRIBUTION LIST**

NAME	COMPANY/SERVICE	ACRONYM	ROLE
Tudor IVANOV	Pulsar/Consultancy	TIV	Project Manager
Jean-Luc LIBIOUILLE	Pulsar/Consultancy	JLL	Project Leader
Ludovic LAMBRECHTS	Pulsar/Consultancy	JGR	Technical Leader
Jeremy GRUMBACH	Pulsar/Consultancy	JGR	Analyst/Developer
Nicolas PHILIPPE	Pulsar/Consultancy	NPH	Analyst/Developer

**DOCUMENT CHANGE RECORD**

Date	Author(s)	Review	Description	Sections
21 Jun 2009	JGR		V0.1: first version	All

**CONTENTS**

<b>1</b>	<b>INTRODUCTION .....</b>	<b>4</b>
<b>2</b>	<b>AIXM 5.X TEMPORALITY .....</b>	<b>5</b>
2.1	Concept overview .....	5
2.2	Temporality implementation .....	5
2.2.1	Feature identification.....	6
2.2.2	Other impacts.....	6
2.3	Snapshots and states management .....	7
<b>3</b>	<b>AIXM 5.X XML/XSD COMPLEXITIES.....</b>	<b>9</b>
3.1	Depth levels .....	9
3.2	Graphical Representation.....	10
<b>4</b>	<b>OTHER AIXM 5.X COMPLEXITIES.....</b>	<b>11</b>
4.1	Data Volume .....	11
4.2	GML .....	11
<b>5</b>	<b>AIXM 4.5 TO AIXM 5.X MAPPING .....</b>	<b>12</b>
5.1	General.....	12
5.2	Specific example.....	12
	Runway and sub-features : .....	12
<b>6</b>	<b>AIXM 5.X USAGE .....</b>	<b>13</b>
6.1	Feature Edition.....	13
6.2	Feature Search .....	13
<b>7</b>	<b>AIXM 5.0 TO AIXM 5.1 .....</b>	<b>15</b>
7.1	Change proposals.....	15

7.2 Some examples .....16

    7.2.1 *Moved an removed attributes*.....16

    7.2.2 *Inverted associations*.....16

    7.2.3 *Description and notes* .....17

7.3 Contaminations .....19

## 1 Introduction

This document intends to give an overview of the AIXM complexities encountered during the xNOTAM and xSNOWTAM projects.

xNOTAM Trial project started in 2007 and was completed in 2008. This project was born to transform standard aeronautic notes “Notes to Airman” (NOTAM) to a science technology language understood by computers, based on the general aeronautical model AIXM-XML. It uses more than twenty AIXM features (like Airport/Heliport, Airspace, Route, Navaid, Obstacle, etc...).

xSNOWTAM project started in 2008 and was completed in 2009. The application objective was to help operators with the generation of ICAO compliant SNOWTAMs (a SNOWTAM is a message describing the conditions of the runways, taxiways and apron at an aerodrome). It is mainly based on the contamination model of AIXM 5.X but it uses also more or less fifteen AIXM features (like Airport/Heliport, Apron, AircraftStand, etc...).

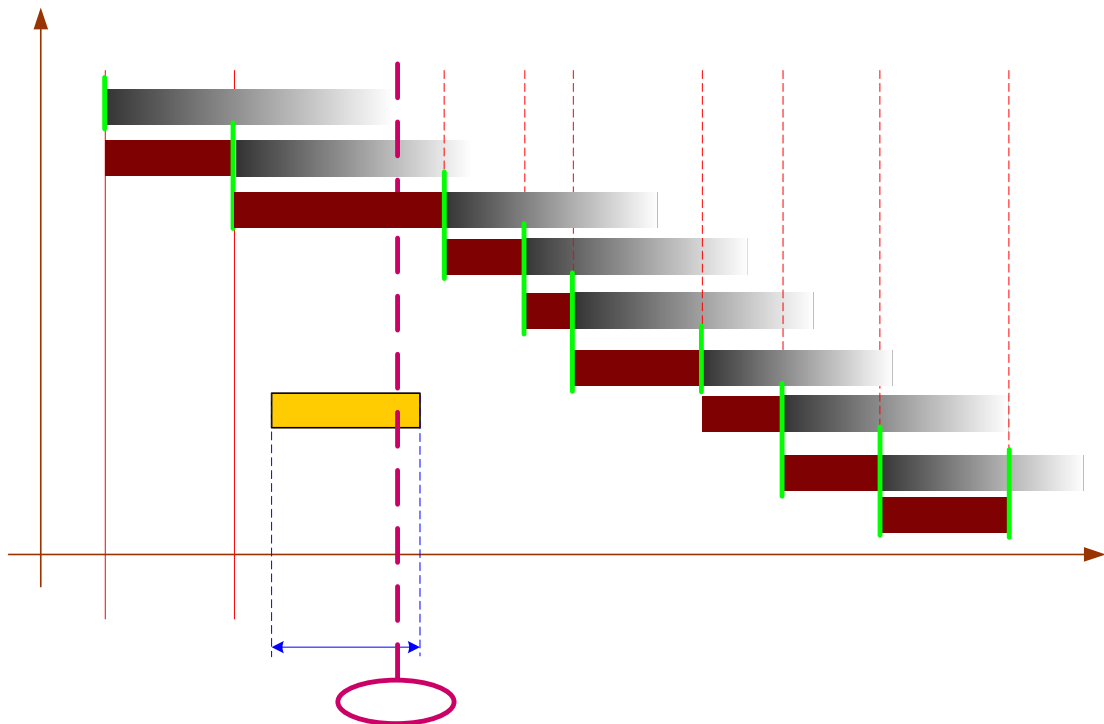
Next chapters will explain complexities of the AIXM 5 model itself (temporality, state, xsd), then will speak about the mapping AIXM 4.5 to AIXM 5 (by describing the different type of changes) and finally will present the AIXM 5.0 to 5.1 mapping.

## 2 AIXM 5.X Temporality

### 2.1 Concept overview

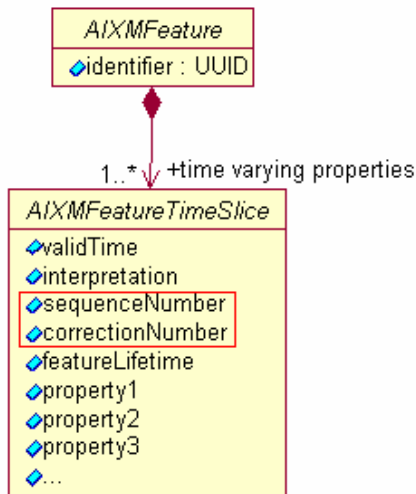
One of the main improvements of AIXM 5.X is the temporality model. It is now possible to create **permanent changes** or **temporary changes**. All the properties can change over the time except the UUID. To do so, new concepts have been introduced in AIXM 5.X:

- **BASELINE**: a kind of Time Slice that describes the feature state (the set of all feature's properties) as result of a permanent change;
- **PERMDELTA**: A kind of Time Slice that describes the difference in a feature state as result of a permanent change;
- **TEMPDELTA**: a kind of Time Slice that describes the overlay of a feature state during a temporary event;
- **SNAPSHOT**: A kind of Time Slice that describes the state of a feature at a time instant, as result of combining the actual BASELINE Time Slice (valid at that time instant) with all eventual TEMPDELTA Time Slices that are effective at that time instant;



### 2.2 Temporality implementation

As explained in the temporality document, we have implemented the temporality model by keeping only the immutable data (UUID) in the "AIXMFeature". All time varying properties have been put in a Time Slice associated to the Feature. Each Time Slice can be PERMDELTA, a BASELINE or a TEMPDELTA (we will add the possibility to have a STATE as a Time Slice in chapter 2.3):



2.2.1 Feature identification

The identification of a feature is either:

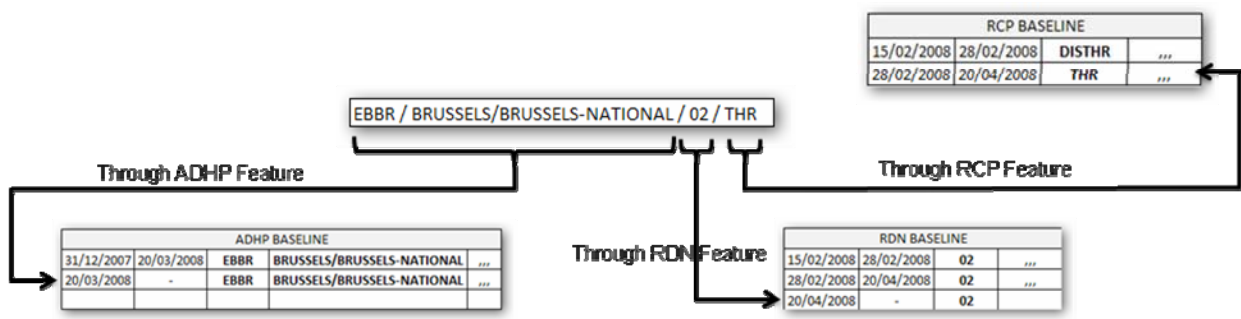
- its natural key if the feature has no parent feature (ex: *designator & name* for an airport);
- or the identification of its parent feature followed by its own natural key (ex: *airport designator & name / runway designator* for a runway).

Therefore, the identification could be something very complicated to retrieve due to the following reasons:

- required information may be missing in delta time slice (for example, a TEMPDELTA which only changes the length of a runway will not have the designator of the runway);
- natural keys may evolve in time;
- Sometimes a feature identification can be built on many parent’s natural keys, also evolving in time.

Ex: Runway Centreline Point is identified by

- ⇒ Associated Airport Heliport designator and name
- ⇒ Associated Runway Direction designator
- ⇒ Its Role



2.2.2 Other impacts

Because of the TEMPDELTA, almost all fields must be nullable in database. This implies more check in web screens and java code.

Each query in database must also be filtered by dates and interpretation: BASELINE, PERMDELTA, TEMPDELTA (or STATE – see chapter below).

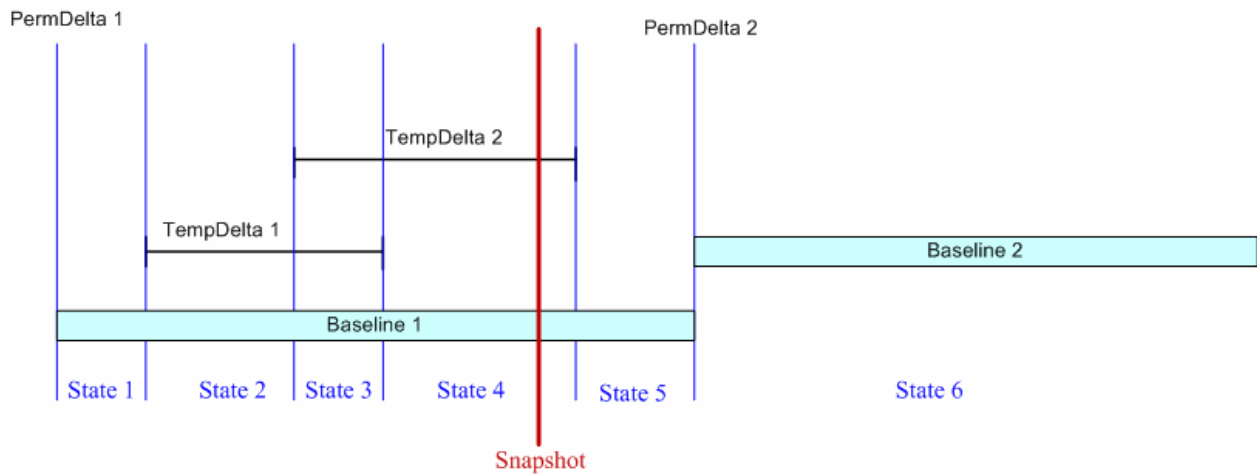
### 2.3 Snapshots and states management

BASELINEs and TEMPDELTA s describe temporal evolution. A SNAPSHOT describes a status at the specific date and time (TimeInstant).

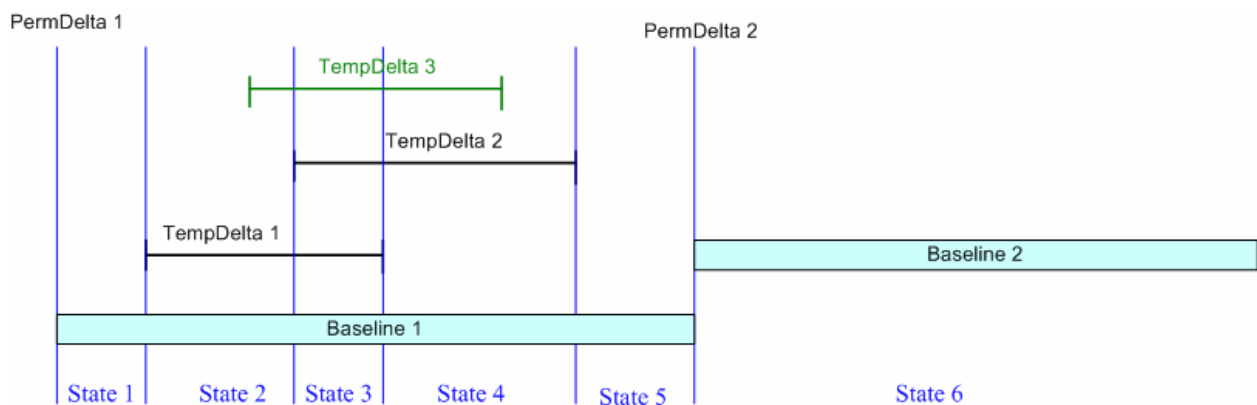
In xNOTAM, a new concept has been added which represents a SNAPSHOT valid over a TimePeriod. It has been called STATE. It was very useful for the reporting and WFS (Web Feature Service) which both export SNAPSHOTs. They were pre-calculated to improve performances of the application.

Example:

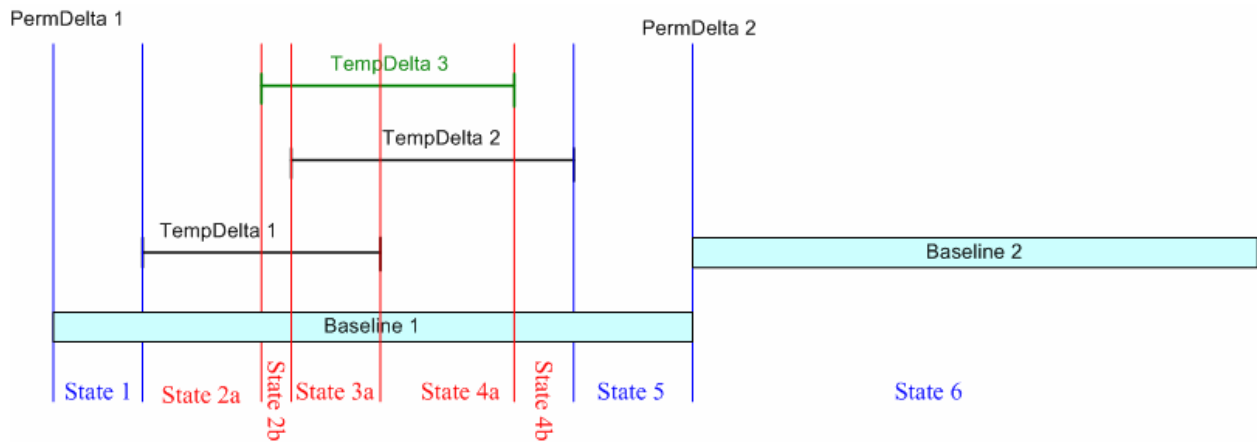
- The picture below shows the 6 states of a feature (determined by the different PERMDELTA s and TEMPDELTA s of this feature):



- A new TEMPDELTA is then created for this feature (TEMPDELTA 3):



- It implies the creation of new STATES (STATE 2, 3 and 4 have been replaced by STATE 2a, 2b, 3a, 4a and 4b):



## 3 AIXM 5.x XML/XSD Complexities

### 3.1 Depth levels

Hierarchy levels in AIXM 5 XSD are much deeper than in AIXM 4.5:

- Our mapping framework generates **2200 classes** for AIXM 4.5 and **18600 classes** for AIXM 5:
- Treatments are much more complex than in AIXM 4.5: for each needed sub-node/element/property a more complex query must be done thru the logical Tree-View;
- Code complexity and potential performance issue.

Example: How to retrieve the length of a runway?

*AIXM 4.5: 3 levels to retrieve the length of the runway*

```
<xsd:element name="Rwy" type="RunwayType"/>
<xsd:complexType name="RunwayType">
  <xsd:annotation>
    <xsd:documentation>RWY</xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="RwyUid" type="RunwayUidType">
      <xsd:annotation>
        <xsd:documentation>a version of Runway [RWY]</xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="valLen" type="valDistHorz" minOccurs="0">
      <xsd:annotation>
        <xsd:documentation>Length</xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
```



## 4 Other AIXM 5.x Complexities

### 4.1 Data Volume

- The database has a size of 11 Gigabytes. This is due to the redundancy of the data with the different timeslice types (baseline/permdelta /states). The database size implies long running conversion scripts / export / imports (and these for all our environments).
- We also encountered a problem of database corruption due to the conversion of spatial data.

### 4.2 GML

- There are no arcs and circles in a geodetic coordinates system. Therefore, arcs and circles received from AIXM 4.5 were converted into polygons: 1 circle is converted to approximately 1 000 points.
- Spatial GML generated by Oracle was not fully compliant with AIXM 5 GML:
  - Oracle generates GML 3.1.1 ;
  - AIXM needs GML 3.2 ;
    - ⇒ Translation procedures for GML 3.2 have been written.
- Lots of geometric aggregations and calculations were required (ex: in xSNOWTAM, done for AMDB, clearance removal, runway thirds...).

## 5 AIXM 4.5 to AIXM 5.x Mapping

### 5.1 General

First of all, much of the complexity explained above increases the complexity of the mapping from AIXM 4.5 to AIXM 5.x.

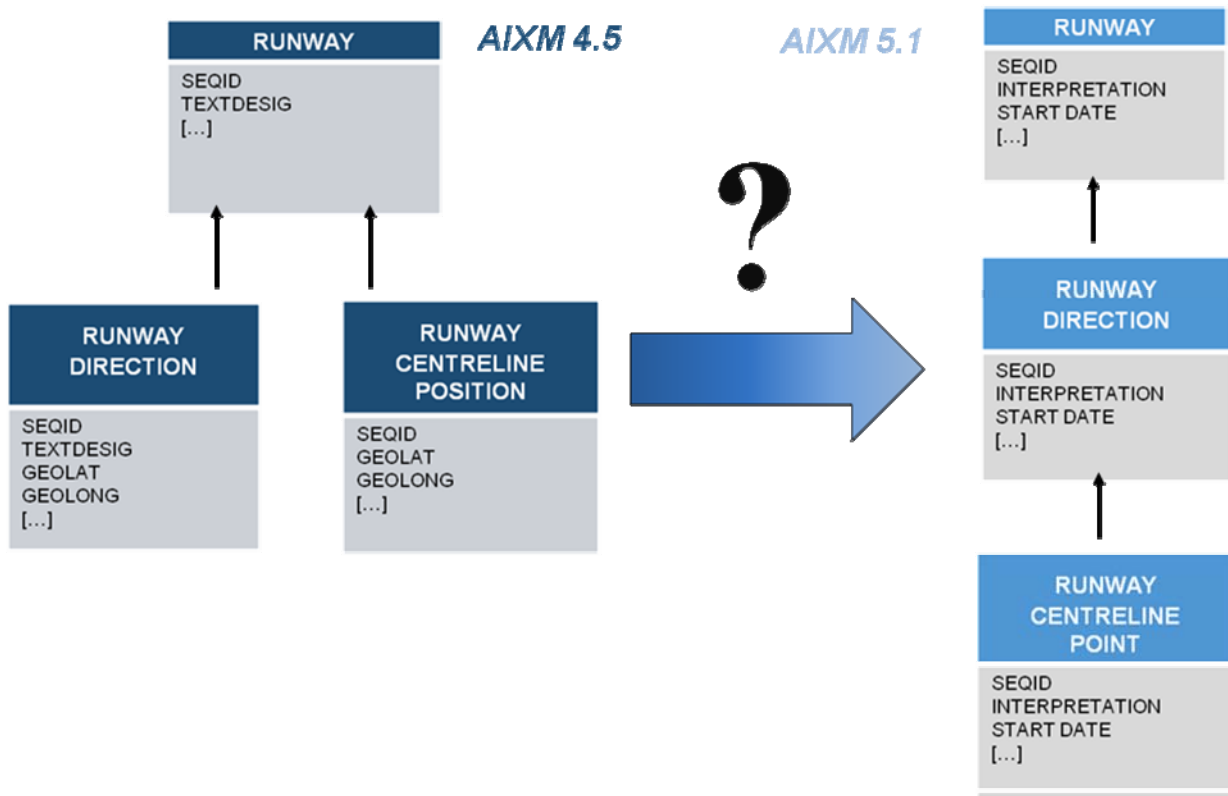
We can identify 4 “levels” of mapping:

- Changes in list of values (easy);
- Changes in name of the field (easy);
- Changes in the representation of the property (ex: elevated point) (difficult);
- More complex changes like Runway and its sub-features (very difficult).

### 5.2 Specific example

#### Runway and sub-features :

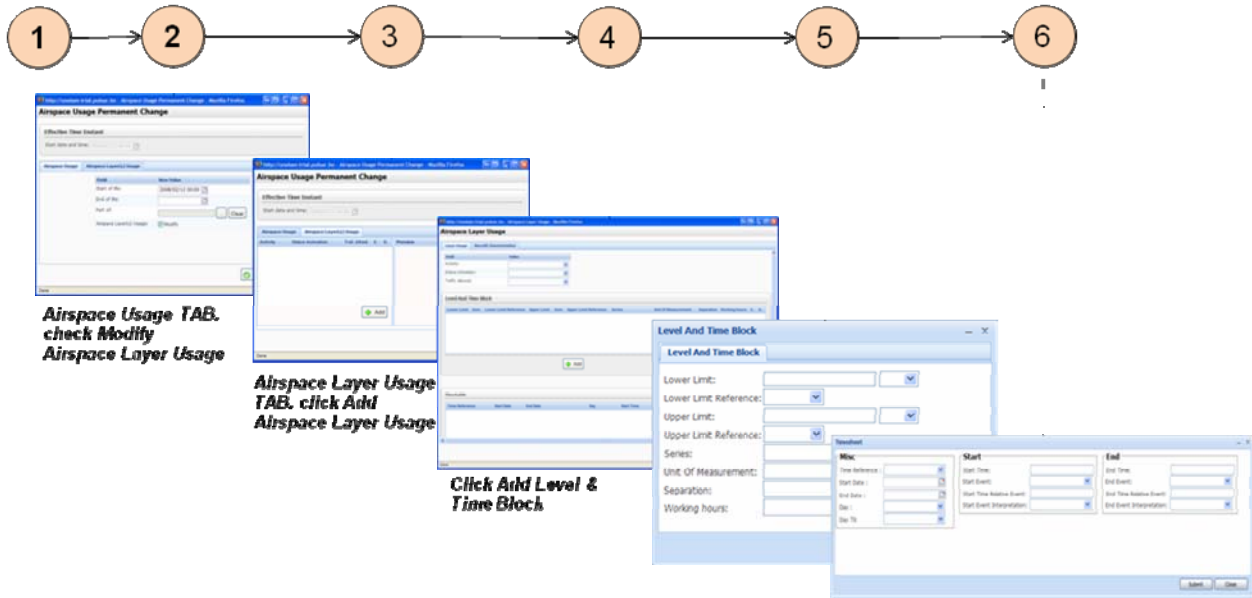
In AIXM 4.5, both *Runway Direction* and *Runway Centreline Position* were linked to the runway directly. In AIXM 5.X, the hierarchy has changed. The *Runway Centreline Position* (called *Runway Centreline Point* in AIXM 5.X) is now linked to the *Runway Direction* itself.



## 6 AIXM 5.x Usage

### 6.1 Feature Edition

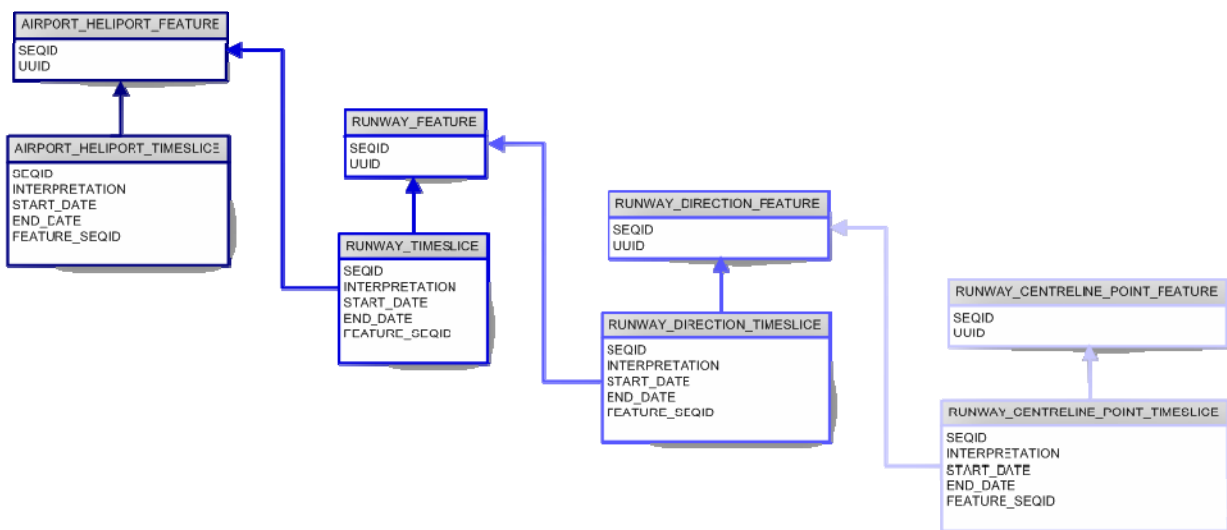
Due the complexity of AIXM 5.x, the edition of some features can be very complex.  
 Example: Airspace Timesheets Edition (6), coming from Event processing screen (1)



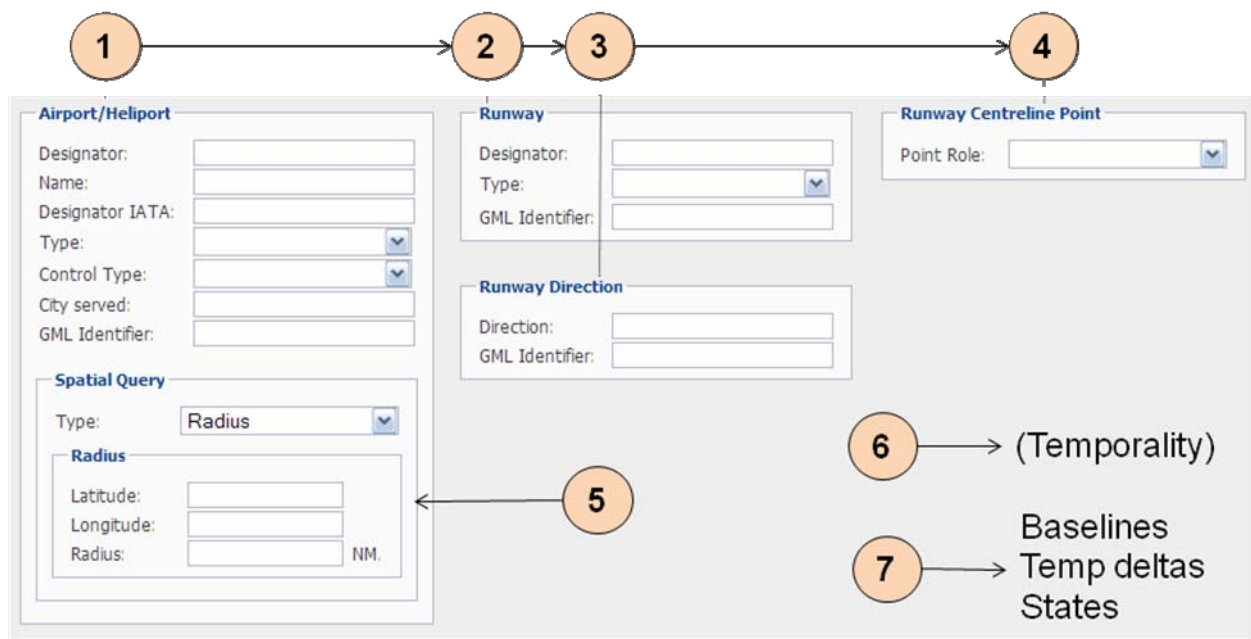
⇒ A deeper analysis must be done with the clients to avoid too many levels of edition.

### 6.2 Feature Search

To search some features, a lot of nested search screens are required. For example, the Runway Centreline Point is determined by its runway direction, its runway and its airport heliport:



The associated search screen is therefore quite complicated:



## 7 AIXM 5.0 to AIXM 5.1

### 7.1 Change proposals

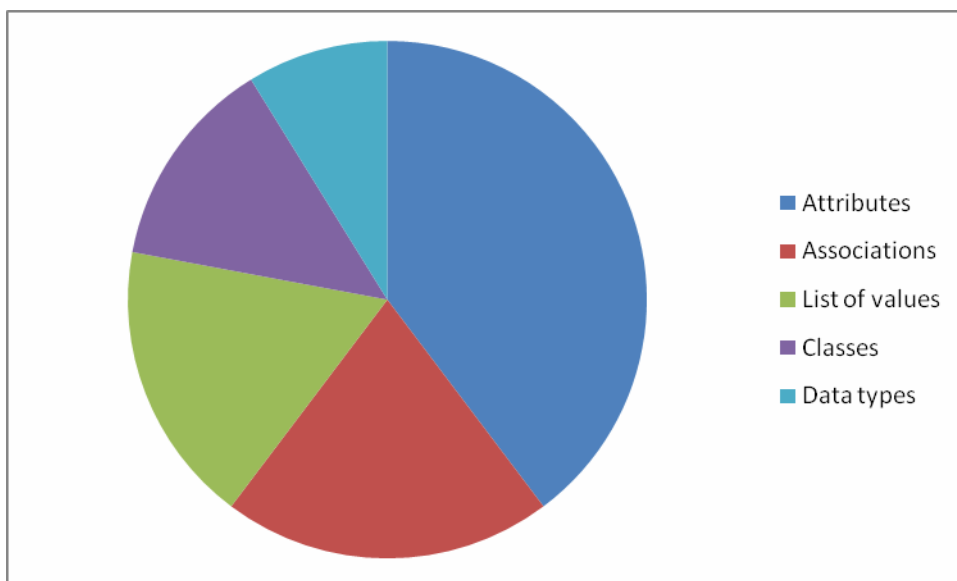
Changes between AIXM 5.0 and AIXM 5.1 imply more or less 50 published proposals. They have been published on the AIXM Forum. EUROCONTROL has foreseen to deliver XSD files which will do the mapping between AIXM 5.0 and AIXM 5.1 XML (in both directions).

**More detailed view of the changes:**

In the table below, you can find all the possible changes to take into account for the AIXM 5.0 to AIXM 5.1 mapping:

Classes <sup>1</sup>	Attributes	Data Types	Values	Associations
New	New	New	New	New
Removed	Removed	Changed	Removed	Removed
Renamed	Changed Type		Renamed	Association Class
	Renamed		Need mapping	Inverted
	Converted to object		Moved	
	Removed and Converted to Note			
	Changed type and converted to Note			
	Moved			

The following graph gives a summary of the aforementioned types:

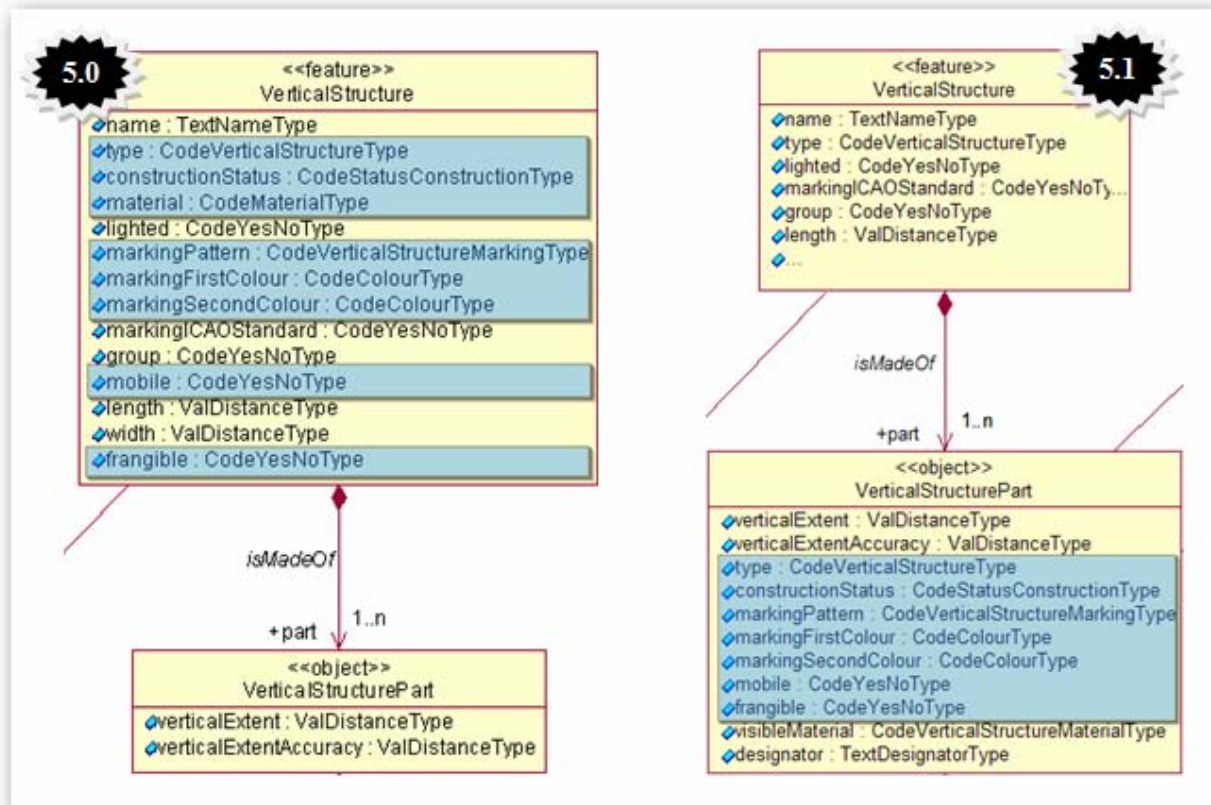


<sup>1</sup> By class, we mean <<feature>> and/or <<object>>.

## 7.2 Some examples

### 7.2.1 Moved an removed attributes

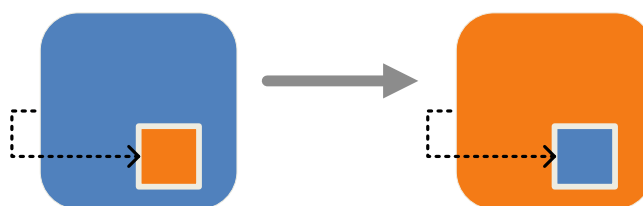
In the change about VerticalStructure, some attributes are moved (*type* or *constructionStatus*) and some attributes are removed (*material*):



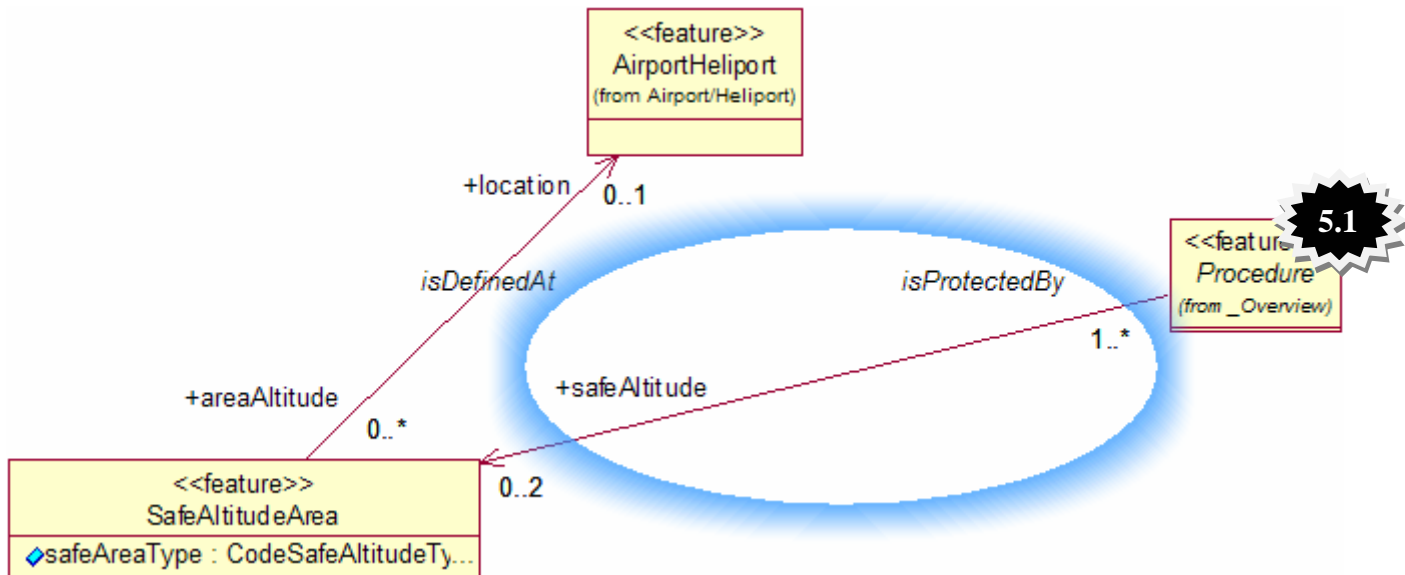
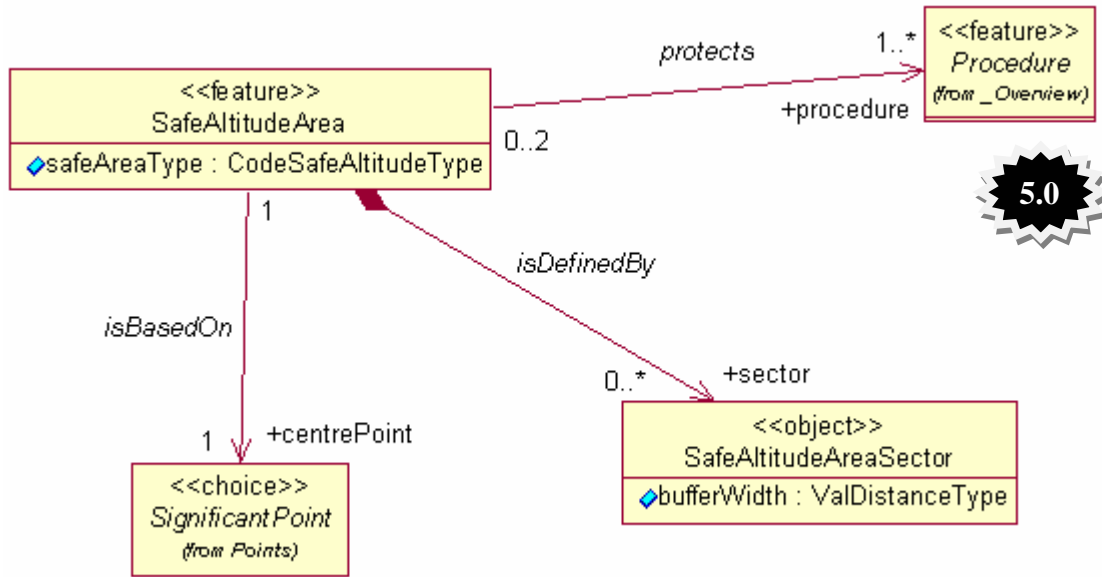
### 7.2.2 Inverted associations

In AIXM 5.1, some associations have been inverted. A class referencing another one is now referenced by the other one.

When exporting the formerly referencing class (A), the link to the referenced class (B) will be omitted. When exporting the referenced class, the mapping will add a link to the formerly referencing class.



The proposed change on SafeAltitudeArea is an example of an inverted association:



### 7.2.3 Description and notes

Some elements appearing as attributes in AIXM 5.0 may be replaced by a Note in AIXM 5.1. Two cases must be distinguished:

- when the attribute itself is completely removed:

```
<aixm:locationDescription>045°, 9 KM from Donlon</aixm:locationDescription>
```

5.0



```
<aixm:annotation>
  <aixm:Note>
    <aixm:definition>descriptions</aixm:definition>
    <aixm:translatedNote>
      <aixm:LinguisticNote>
        <aixm:note lang="eng">045°, 9 KM from Donlon</aixm:note>
      </aixm:LinguisticNote>
    </aixm:translatedNote>
  </aixm:Note>
</aixm:annotation>
```

5.1

- when the note keeps a reference to the attribute:

Example for AirportHeliport.altimeterCheckLocation converted from AIXM 5.0 to AIXM 5.1:

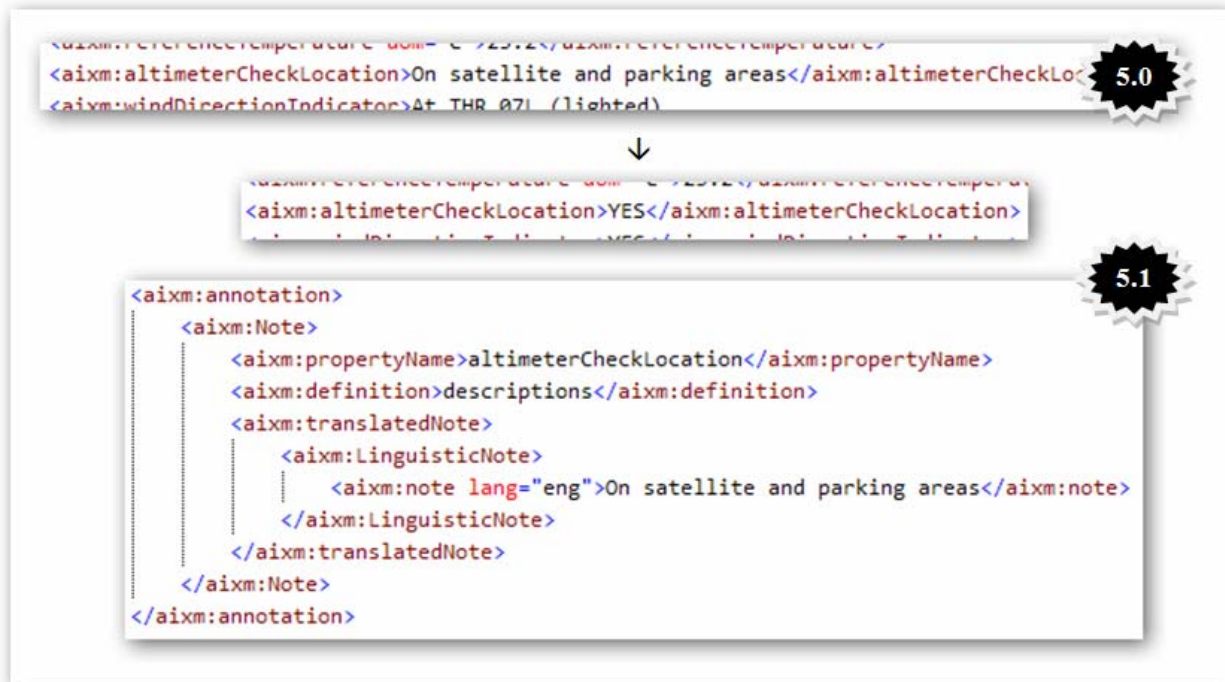
5.0

```
<<feature>>
  AirportHeliport
  ♦designator : CodeAirportHeliportDesignatorType
  ♦name : TextNameType
  ♦locationIndicatorICAO : CodeCAOType
  ♦designatorATA : CodeATAType
  ♦type : CodeAirportHeliportType
  ♦certifiedICAO : CodeYesNoType
  ♦privateUse : CodeYesNoType
  ♦controlType : CodeMilitaryOperationsType
  ♦referencePointDescription : TextDescriptionType
  ♦fieldElevation : ValDistanceVerticalType
  ♦fieldElevationAccuracy : ValDistanceVerticalType
  ♦verticalDatum : CodeVerticalDatumType
  ♦locationDescription : TextDescriptionType
  ♦magneticVariation : ValMagneticVariationType
  ♦magneticVariationAccuracy : ValAngleType
  ♦dateMagneticVariation : DateYearType
  ♦magneticVariationChange : ValMagneticVariationChangeType
  ♦referenceTemperature : ValTemperatureType
  ♦altimeterCheckLocation : TextDescriptionType
  ♦secondaryPowerSupply : TextDescriptionType
  ♦windDirectionIndicator : TextDescriptionType
  ♦landingDirectionIndicator : TextDescriptionType
  ♦transitionAltitude : ValDistanceVerticalType
  ♦transitionLevel : ValFLType
  ♦lowestTemperature : ValTemperatureType
  ♦abandoned : CodeYesNoType
```

5.1

```
<<feature>>
  AirportHeliport
  ♦designator : CodeAirportHeliportDesignatorType
  ♦name : TextNameType
  ♦locationIndicatorCAO : CodeCAOType
  ♦designatorATA : CodeATAType
  ♦type : CodeAirportHeliportType
  ♦certifiedCAO : CodeYesNoType
  ♦privateUse : CodeYesNoType
  ♦controlType : CodeMilitaryOperationsType
  ♦fieldElevation : ValDistanceVerticalType
  ♦fieldElevationAccuracy : ValDistanceVerticalType
  ♦verticalDatum : CodeVerticalDatumType
  ♦magneticVariation : ValMagneticVariationType
  ♦magneticVariationAccuracy : ValAngleType
  ♦dateMagneticVariation : DateYearType
  ♦magneticVariationChange : ValMagneticVariationChangeType...
  ♦referenceTemperature : ValTemperatureType
  ♦altimeterCheckLocation : CodeYesNoType
  ♦secondaryPowerSupply : CodeYesNoType
  ♦windDirectionIndicator : CodeYesNoType
  ♦landingDirectionIndicator : CodeYesNoType
  ♦transitionAltitude : ValDistanceVerticalType
  ♦transitionLevel : ValFLType
  ♦lowestTemperature : ValTemperatureType
  ♦...
```

Changed to CodeYesNoType



### 7.3 Contaminations

The xSNOWTAM project has identified holes and important improvements in the contamination model of AIXM 5.0. It has been there completely reviewed as you can see with the two following diagrams (differences are in blue):

