Aeronautical Information Exchange Model (AIXM) Reader/Writer

The AIXM Reader/Writer allows the Feature Manipulation Engine (FME) to read Aeronautical Information Exchange Model format files.

This chapter assumes familiarity with the AIXM format.

Overview

The Aeronautical Information Exchange Model (AIXM) format was developed by EUROCONTROL, the European Organisation for the Safety of Air Navigation, to allow aeronautical data standardization and exchange. The role of AIXM is to enable systems to exchange aeronautical information in the form of XML-encoded data.

AIXM Quick Facts

Format Type Identifier	AIXM
Reader/Writer	Both
Dataset Type	File
Licensing Level	Professional
Dependencies	None
Feature Type	AIXM entity name
Typical File Extensions	.xml
Automated Translation Support	No
User-Defined Attributes	No
Coordinate System Support	Yes
Generic Color Support	No
Spatial Index	Never
Schema Required	No
Transaction Support	No
Geometry Type Attribute	xml_type

Geometry Support				
Geometry	**Supported?**		**Geometry**	**Supported?**
aggregate	yes		point	yes
circles	yes		polygon	yes
circular arc	yes		raster	no
donut polygon	no		solid	no
elliptical arc	no		surface	no
ellipses	no		text	no

Geometry Support				
Geometry	**Supported?**		**Geometry**	**Supported?**
line	yes		z values	no
none	yes			

Reader Overview

The AIXM reader presents features by normalizing the XML data into the entities of the AIXM Entity-Relational model. Thus, the feature representation is not equivalent to the AIXM XML format representation of the AIXM E-R model entity.

Reader Directives

The suffixes shown below are prefixed by the current `<ReaderKeyword>` in a mapping file. By default, the `<ReaderKeyword>` for the AIXM reader is `AIXM`.

DATASET

Required/Optional: *Required*

The value for this directive is the path of the AIXM file to be read. A typical mapping file fragment specifying an input AIXM dataset looks like:

```
AIXM_DATASET /usr/data/aixm.xml
```

INTERPOLATE

Required/Optional: *Optional*

The value for this directive determines whether non-linear interpolation will be performed between two vertices of an area or line geometry. This keyword will also determine the representation of geometry data. Further information on this topic can be found under the Feature Representation heading. An example mapping file fragment specifying that interpolation should be performed looks like:

```
INTERPOLATE Yes
```

Writer Overview

The AIXM writer has a fixed output schema that closely resembles the AIXM Entity-Relational model. The reader can be connected directly to the writer and the the output file will be near identical the original source file.

Writer Directives

The following table lists the directives processed by the AIXM writer. The suffixes shown will be prefixed by the current `<WriterKeyword>` in a mapping file. By default, the `<WriterKeyword>` for the AIXM writer is `AIXM`.

DATASET

Required/Optional: *Required*

The value for this keyword is the path of the output AIXM file. A typical mapping file fragment specifying an output AIXM file looks like:

```
AIXM_DATASET /usr/data/aixm.xml
```

WRITE_MODE

Required/Optional: *Optional*

The value for this keyword determines the type of AIXM file, either an AIXM Snapshot or AIXM Update, produced by the writer. Valid values are `UPDATE` and `SNAPSHOT`. The default value is `UPDATE`:

```
AIXM_WRITE_MODE UPDATE
```

ORIGIN

Required/Optional: *Optional*

The value for this keyword is a string that determines the originator of the AIXM message:

```
AIXM_ORIGIN ABC
```

CREATED

Required/Optional: *Optional*

The value for this keyword determines the date and time that the AIXM message was created. The string should be a valid XML dateTime string:

```
AIXM_CREATED 2002-10-10
```

EFFECTIVE

Required/Optional: *Optional*

The value for this keyword determines the date and time that the AIXM message becomes effective. The string should be a valid XML dateTime string:

```
AIXM_CREATED 2002-10-10
```

USE_CHG

Required/Optional: *Optional*

The value for this keyword determines whether the 'chg' XML attributes will be added to each XML element written by the writer. Valid values are `YES` and `NO`. If the value is `YES`, then XML elements whose names appear in the `aixm_update_changed` format specific attribute will have an XML attribute named 'chg' with a value of '1' inserted.

The default value for this keyword is `YES`:

```
AIXM_USE_CHG YES
```

Feature Representation

In addition to the generic FME feature attributes that FME Workbench adds to all features (see *About Feature Attributes* on page 7), special FME feature attributes are used by the AIXM reader to store the characteristics of the features it reads.

The AIXM Reader module utilizes the XML Reader module in processing the AIXM XML file. Thus, the feature reprensentation is similiar to the feature reprensentation of the XML Reader module. The format attribute, `xml_type`, which may identify the geometry type of the feature, is identical in intent to the same attribute set by the XML Reader. Details of this attribute can be found in the *XML Reader/Writer* documentation.

Attribute Name	Contents
`aixm_update_ID`	When an AIXM Update message changes the natural key that identifies an object, then this attribute will hold the old natural key of the object to be updated.
`aixm_update_group_no`	This attribute determines the order of the AIXM Group elements within the output file. The values of this attribute are integers, and identifies the feature with a specific group. When all features are received by the AIXM writer, features are grouped according to the values of their `aixm_update_group_no` attribute, and the groups are written in ascending order of through group numbers. If this attribute is not specified, then the feature will be grouped with group number zero.
`aixm_update_name`	The value of this attribute specifies the 'name' attribute of the AIXM Group element that holds the feature.
`aixm_update_subname`	The value of this attribute specifies the 'sub-name' attribute of the AIXM Group element that holds the feature.
`aixm_update_reason`	The value of this attribute specifies the 'reason' attribute of the AIXM Group element that holds the feature.
`aixm_update_type`	The value of this attribute determines the type of the AIXM Update mesasge for that particular feature: New, Update, or Withdrawn.
`aixm_noseq`	The AIXM reader normalizes the XML schema. After this transformation, child elements that composed a parent element may become independant features. If the child elements were ordered within the parent element, then this attribute will hold the sequence number that determines the child element's placement within a parent element.
`aixm_update_changed`	This is a list attribute that holds the names of attributes that are flagged as changed in an AIXM Update message.