

CITS Data Transfer Format (QLF) Reader/Writer

FORMAT NOTES:

This format is not supported by FME Base Edition.

The Centre for Topographic Information Sherbrooke (CITS) Data Transfer Format (QLF) Reader and Writer modules provide the Feature Manipulation Engine (FME) with the ability to read and write QLF import and export files. The QLF is a published ASCII format used by CITS for import and export.

Overview

QLF files store both feature geometry and attribution. A QLF file has the following file name extension:

File Name Extension	Contents
.qlf	Vector geometric data
.qlf.gz	Same as above but in compressed gzip format.

The extension is added to the basename of the QLF file. Optionally adding `.gz` to the extension will output a compressed gzip QLF format file; conversely, the reader can directly read files with the extension `.qlf.gz`.

The QLF reader and writer supports the storage of point, line, and polygon geometric data in `.qlf` files. The QLF format also stores features with no geometry. Features having no geometry are referred to as having a geometry of *none*.

QLF Quick Facts

Format Type Identifier	QLF
Reader/Writer	Both
Licensing Level	Professional
Dependencies	None
Dataset Type	File
Feature Type	qlf_record type
Typical File Extensions	.qlf, .qlf.gz
Automated Translation Support	Yes
User-Defined Attributes	No
Coordinate System Support	No
Generic Color Support	No
Spatial Index	Never
Schema Required	No
Transaction Support	No
Geometry Type	qlf_type

Geometry Support			
Geometry	Supported?	Geometry	Supported?
aggregate	no	point	yes
circles	no	polygon	yes
circular arc	no	raster	no
donut polygon	yes	solid	no
elliptical arc	no	surface	no
ellipses	no	text	no
line	yes	z values	yes
none	no		

Reader Overview

The QLF reader extracts features from a file one at a time, and passes them on to the rest of the FME for further processing. The reader finishes when it reaches the end of the file.

Reader Directives

The directives processed by the QLF reader are listed below. The suffixes shown are prefixed by the current <ReaderKeyword> in a mapping file. By default, the <ReaderKeyword> for the QLF reader is QLF.

DATASET

Required/Optional: *Required*

The value for this keyword is the file path of the QLF file to be read. A typical mapping file fragment specifying an input QLF data set looks like:

```
QLF_DATASET /usr/data/qlf/qlffile.qlf
```

DEF

Required/Optional: Required

Each QLF file may optionally be defined before it is read. The definition specifies the base name of the file, and the names and the types of all attributes. The syntax of a QLF DEF line is:

```
<ReaderKeyword>_DEF <baseName> \
[<attrName> <attrType>]+
```

The basename specified on the QLF DEF lines is constructed by using either the file name without the extension specified by the DATASET keyword or `qlf_record` (used only when QLF is the source).

QLF files require at least one attribute to be defined. The attribute definition given must match the definition of the file being read. If it does not, translation is halted and the true definition of the QLF file's attributes gets logged to the log file.

The following table shows the attribute types supported.

Field Type	Description
<code>char(<width>)</code>	Character fields store fixed-length strings. The <code>width</code> parameter controls the maximum number of characters that can be stored by the field. No padding is required for strings shorter than this width.
<code>date</code>	Date fields store dates as character strings with the format YYYYMMDD.
<code>number(<width>,<decimals>)</code>	Number fields store single and double precision floating point values. The <code>width</code> parameter is the total number of characters allocated to the field, including the decimal point. The <code>decimals</code> parameter controls the precision of the data and is the number of digits to the right of the decimal.
<code>logical</code>	Logical fields store TRUE/FALSE data. Data read or written from and to such fields must always have a value of either <code>true</code> or <code>false</code> .

The following mapping file fragment defines a QLF file def line when QLF is the source file format.

```
QLF_DEF qlf_record \
  F1 char(20) \
  F2 char(20) \
  F3 char(20) \
  F4 char(20) \
  F5 char(20) \
  F6 char(20) \
```

F7 char(20)
F8 char(20)
F9 char(20)

\
\

Writer Overview

The QLF writer creates and writes feature data to a QLF file specified by the `DATASET` keyword. As with the reader, the directory must exist before the translation occurs. Any existing QLF files in the directory are overwritten with the new feature data. Only one QLF file can be written during a single FME session. Optionally a `.prj` file will also be written if the the coordinate system's (projection) information is available. Output `.prj` files comply with ESRI's shape format projection file specification.

Writer Directives

The directives that are processed by the QLF writer are listed below. The suffixes shown are prefixed by the current `<WriterKeyword>_` in a mapping file. By default, the `<WriterKeyword>` for the QLF writer is `QLF`.

Note: By default, the QLF writer will write the coordinates with 15 digits of precision. If this is not desirable or it is causing problems, then the precision can be easily changed by editing the value of the `QLF_PRECISION` macro in the `qlf_write.fmi` file in the directory `[FME_HOME]\pipeline`. For example, if your FME installation is in `C:\Program Files\FME` the file `qlf_write.fmi` can be found in `C:\Program Files\FME\pipeline` directory.

DATASET, DEF

These directives are processed as described in the Reader Directives section.

PRECISION

Required/Optional: *Optional*

Define the precision of output coordinates. To be precise, the value of this keyword will determine the number of significant digits after the decimal for the output coordinates.

Range: 1 to 15

Default: 15

NUMFIELD

Required/Optional: *Optional*

Defines the number of user defined fields. This should be set during mapping file generation.

Range: 1 to 512

Default: 9

Feature Representation

QLF features consist of geometry and attributes. The attribute names are defined in the DEF line and there is a value for each attribute in each QLF feature.

In addition to the generic FME feature attributes that FME Workbench adds to all features (see *About Feature Attributes* on page 7), this format adds the format-specific attributes described in this section.

Attribute Name	Contents
qlf_type	The QLF geometric type of this entity. Range: qlf_point qlf_polygon qlf_line qlf_none Default: No default
F1- Fn	Represents a feature attribute where 'n' is the number set via the keyword NUMFIELDS during mapping file generation. If NUMFIELDS is set to 5 then there will be 5 attributes F1, F2, F3, F4 and F5. Range: Maximum of 20 characters Default: Blank

Points

qlf_type: qlf_point

QLF point features specify a single x and y coordinate in addition to any associated user-defined attributes. There are no special FME attributes for the QLF line type.

Lines

qlf_type: qlf_line

QLF line features specify linear features defined by a sequence of x and y coordinates. There are no special FME attributes for the QLF lines type.

Polygon

qlf_type: qlf_polygon

QLF polygon features specify area (polygonal) features. The areas that make up a single feature may or may not be disjoint, and may contain polygons that have holes. There are no special FME attributes for the QLF region type.

