

1Spatial Internal Feature Format (IFF) Reader/Writer

FORMAT NOTES:

This format is not available in FME Base Edition.

The 1Spatial (previously known as Laser-Scan) Internal Feature Format (IFF) Reader/Writer module allows FME to read and write IFF files. The IFF is an internal ASCII format originally created by Laser-Scan Ltd. This chapter assumes that you are familiar with the Internal Feature Format.

Overview

IFF files store both geometry and attribution for features. An IFF file has the following file name extension:

File Name Extension	Contents
.iff	Vector geometric data

The extension is added to the basename of the IFF file.

The Internal Feature reader supports the types of symbol, line, polygon, and text geometric data in `.iff` files. The IFF format also stores features with no geometry. Features that have no geometry are referred to as having a geometry of *none*. IFF files support both two- and three-dimensional geometry.

IFF Quick Facts

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

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

Reader Overview

The IFF reader produces FME features for all feature data held in IFF files that are in the specified directory.

The reader first scans the given directory specified by the `DATASET` keyword. For each IFF file found, it checks to see if that file is requested in the translation by comparison with the list specified by the `ID`'s keyword. Then the reader extracts features from an IFF file one at a time, and passes them on to the rest of the FME for further processing. When the file is exhausted, the IFF reader moves onto the next file in the directory.

Optionally a single IFF file can be specified. In this case, only that IFF file is read.

Reader Directives

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

DATASET

Required/Optional: *Required*

The value for this directive contains the directory where the IFF files will be read, or the file path to the single IFF file. A typical mapping file fragment specifying an input IFF dataset looks like:

```
IFF_DATASET /usr/data/iff/iffFile.iff
```

Workbench Parameter: [<WorkbenchParameter>](#)

DEF

Required/Optional: *Required*

Each IFF 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 an IFF `DEF` line is:

```
<ReaderKeyword>_DEF <baseName> \
```

[<attrName> <attrType>]+

The file names of the physical IFF files are constructed by using the directory specified by the `DATASET` keyword, the basename specified on the IFF `DEF` lines, and the `.iff` extension.

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 <code>YYYYMMDD</code> .
<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>smallint</code>	Small integer fields store 16-bit signed integers and therefore have a range of -32767 to +32767.
<code>integer</code>	Integer fields store 32-bit signed integers.
<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> .
<code>float</code>	Float fields store 4-byte floating point values. There is no ability to specify the precision and width of the field.
<code>double</code>	Double fields store 8-byte floating point values.

The following mapping file fragment defines a IFF file. Notice that the definition specifies the geometric type of the entities it will contain since IFF files may contain any of the valid geometry types.

```
IFF_DEF landcover \
area          number(12,3) \
landcoverType char(11) \
perimeter     float
```

Workbench Parameter: [<WorkbenchParameter>](#)

IDs

Required/Optional: *Optional*

Contains a list of IFF files to process. If more IFF files were in the directory, they are ignored. If this is not specified, then all defined IFF files in the directory are read.

This specification is used to limit the available and defined IFF files read. If no `IDs` are specified, then all defined and available IFF files are read. The syntax of the `IDs` keyword is:

```
<ReaderKeyword>_IDs<baseName1> \
<baseName2> ... \
<baseNameN>
```

The file `IDs` must match those used in `DEF` lines.

The example below selects only the `iff_data` IFF file for input during a translation:

```
IFF_IDs iff_data
```

Workbench Parameter: [<WorkbenchParameter>](#)

APPLY_ORIGIN_OFFSET

Required/Optional: *Optional*

This directive specifies whether or not to apply the origin offset found in Type 2 Map Descriptor record, to all the features.

Values: *yes | no*

Default: *no*

Workbench Parameter: [<WorkbenchParameter>](#)

Writer Overview

The IFF Writer creates and writes feature data to IFF files in the directory specified by the directive `DATASET`. Any existing IFF files are overwritten with a new IFF file. As features are routed from the FME to the IFF writer, the writer determines which file the features are to be written to and outputs them accordingly. More than one IFF file can be written in one translation.

Writer Directives

The IFF writer processes the `DATASET` and `DEF` directives as described in the *IFF Reader Overview's Reader Directives* subsection. It does not use the `IDs` directive.

DATASET

Required/Optional: *Required*

DEF

Required/Optional: *Required*

Feature Representation

IFF features consist of geometry and attributes of feature. The attribute names are defined in the `DEF` line and there is a value for each attribute in each IFF feature. In addition, each IFF feature contains several special attributes to hold the type of the

geometric entity and its display parameters. All IFF features contain the `iff_type` attribute, which identifies the geometric type.

Depending on the geometric type, the feature contains additional attributes (in addition to the generic FME feature attributes that FME Workbench adds to all features [see *About Feature Attributes* on page 7]) specific to the geometric type. These are described in subsequent sections.

The general attributes of all IFF geometric features are described in the table below.

Attribute Name	Contents
<code>iff_type</code>	The IFF geometric type of this entity. Range: iff_symbol iff_line iff_polygon iff_text iff_none Default: No default
<code>iff_history</code>	The history of the IFF file. Range: Maximum of 256 characters Default: Blank
<code>iff_map_area</code>	The map area. Range: Maximum of 256 characters Default: No default
<code>iff_map_grid</code>	The map grid representation Range: Maximum of 256 characters Default: No default
<code>iff_map_scale</code>	The map scale. Range: Maximum of 256 characters Default: No default
<code>iff_map_origin_offset</code>	The map's origin offset or local origin. (Read-only) Range: Maximum of 256 characters Default: No default
<code>iff_map_projection</code>	The map's projection as a number. (Read-only) Range: Integer Default: No default
<code>iff_map_spheroid</code>	The map's spheroid as number. (Read-only) Range: Integer Default: No default
<code>iff_map_units</code>	The map's unit. (Read-only) Range: Integer Default: No default
<code>iff_map_proj_stat</code>	The map projection status. Range: Maximum of 256 characters Default: No default

Attribute Name	Contents
iff_map_aux_grid	The map auxiliary grid. Range: Maximum of 256 characters Default: No default
iff_cubic_coef{#}	The matrix for coordinate transformation. Each list attribute of this type contains 2 numbers that are part of the transformation matrix for the feature. Range: Maximum of 256 characters. Default: No default
iff_ctrl_pt_nw	The Northwest control points. Range: Maximum of 256 characters Default: No default
iff_ctrl_pt_sw	The Southwest control points. Range: Maximum of 256 characters Default: No default
iff_ctrl_pt_se	The Southwest control points. Range: Maximum of 256 characters Default: No default
iff_ctrl_pt_ne	The Northeast control points. Range: Maximum of 256 characters Default: No default
iff_sec_descr	The section description. Range: Maximum of 256 characters Default: Blank
iff_layer_num	The layer number of the feature. (0 is a reserved layer number that is often ignored) Range: 0...32767 Default: 1
iff_layer_stat	The layer status flag (currently not used). Range: Integer Default: 0
iff_layer_ptr	The layer's pointer to location of matching end of layer marker. Range: Maximum of 256 characters Default: Blank
iff_serial_num	The feature serial number. This number is often the same as the iff_seq_num. Range: 0...65535 Default: No default
iff_seq_num	The feature internal sequence number. This number is unique and corresponds with creation order. Range: 0...65535 Default: 1 and increment for each new feature
iff_feat_code	The feature code number. Range: 0...32767 Default: 0

Attribute Name	Contents
<code>iff_feat_stat</code>	The feature status. Range: 0...32767 Default: 0
<code>iff_proc_code</code>	The feature type or process code. The last two bits specify the feature type (0 = line, circle, area or symbol string feature, 1 = symbol feature, 2 = text feature, and 3 = value reserved). Range: 0...32767 Default: 0
<code>iff_user_word</code>	The user defined word, this is a reserved field for use by users. Range: Smallint Default: No default
<code>iff_anc_code{#}.type</code>	The ancillary code type. Range: 0...32767 Default: No default
<code>iff_anc_code{#}.value</code>	The ancillary code value. Range: Maximum of 256 characters Default: No default
<code>iff_anc_code{#}.text</code>	The ancillary code text. Range: Maximum of 255 characters Default: No default
<code>iff_anc_code</code>	The comma-separated ancillary code list. Range: Maximum of 256 characters Default: No default
<code>iff_pen_stat</code>	The pen status. Range: 0 - pen up, 1 - pen down Default: 0
<code>iff_junc_blk{#}.sec_num</code>	The junction block section number. Range: integer Default: No default
<code>iff_junc_blk{#}.next_jb</code>	The junction block pointer to the next junction block. Range: Maximum of 256 characters Default: No default
<code>iff_junc_blk{#}.offset</code>	The junction block offset number. Range: Maximum of 256 characters Default: No default
<code>iff_junc_blk{#}.arms_num</code>	The junction block number of arms. Range: Maximum of 256 characters Default: No default
<code>iff_junc_blk{#}.x_coord</code>	The junction block x-coordinate. Range: Double Default: No default

Attribute Name	Contents
<code>iff_junc_blk{#}.y_coord</code>	The junction block y-coordinate. Range: Double Default: No default
<code>iff_junc_blk{#}.pnt_no</code>	The junction block string vertex number. Range: Integer Default: No default
<code>iff_junc_blk{#}.addr</code>	The junction block address location of the arm coordinates. Range: Maximum of 256 characters Default: No default
<code>iff_junc_blk</code>	The comma separated junction block list. (Nested junction block lines are separated by '\\s'). Range: Maximum of 256 characters Default: No default
<code>iff_junc_ptr{#}.offset</code>	The junction pointer junction block offset. Range: Maximum of 256 characters Default: No default
<code>iff_junc_ptr{#}.addr</code>	The junction pointer junction block address. Range: Maximum of 256 characters Default: No default
<code>iff_junc_ptr</code>	The comma separated junction pointer list. Range: Maximum of 256 characters Default: No default
<code>iff_void_size</code>	The size of the void. Range: Integer Default: No default

Symbol

iff_type: iff_symbol

IFF symbol features are point features that specify a single x and y coordinate in addition to any associated user-defined attributes.

There are no special FME attribute names used to control the IFF symbol settings.

Lines

iff_type: iff_line

IFF line features specify linear features defined by a sequence of x and y coordinates.

The following table lists the special FME attribute names used to control the IFF line settings.

Attribute Name	Contents
<code>iff_size</code>	The line thickness. Range: Integer Default: No default

Text

iff_type: iff_text

IFF text features are used to specify annotation information. Although IFF files can have features that have more than one set of annotation information, only simple text features can be written. The IFF reader is able to read features with more than one set of annotation information by splitting them into separate features. But the IFF writer will not merge these separated features when writing, only features with only a set of annotation information are written.

The following table lists the special FME attribute names used to control the IFF text settings.

Attribute Name	Contents
iff_rot	The text label's rotation. Range: 0.00...360.00 Default: 0
iff_text_string	The text label. Range: Maximum of 256 characters Default: Blank
iff_size	The text size. Range: Integer Default: No default
iff_text_code	The text code value. Range: Integer Default: No default
iff_text_cmpnt	The text component. Range: Integer Default: No default
iff_text_res1	Reserved space for future use. Range: Maximum of 256 characters Default: No default
iff_text_res2	Reserved space for future use. Range: Maximum of 256 characters Default: No default

