

**NAME**

`fits2bp` – convert a FITS event file to bpipe format

**SYNOPSIS**

`fits2bp` *options*

**DESCRIPTION**

**fits2bp** converts data from FITS table format to bpipe format. It should work with both binary and ASCII tables, but has only been tested on the former.

Explicit maps between the FITS and bpipe representations are specified in a map specification file (“Map Specification”). Only those columns specified in the map file are converted.

Alternatively, all of the columns in the input FITS file may be converted using the `auto_create` option. Maps will be automatically created, but may be overridden by specifications in a map file. It is possible that the FITS column name may not be a legal bpipe field name; in this case `fits2bp` will not continue. The user will have use a map specification to explicit rename the column.

**OPTIONS**

**fits2bp** uses an IRAF-compatible parameter interface.

**input** The input FITS file. This must be a file!

**output** The output bpipe file. If it’s the string `stdout` it is written to the standard output stream.

**map** The name of the file containing specifications on how to map the FITS data to bpipe format. See “Map Specifications” for more info.

It may have the value `none`, or may be empty, indicating that no map file is present.

**extname**

The name of the extension in the FITS file to convert (the value of the FITS `EXTNAME` header keyword).

**auto\_create**

If true, mapping for all of the columns in the FITS file will be automatically created. Mappings in the map file override the automatically create versions.

**version** Print the version of the program and exit.

**help** Print usage and exit.

**mode** A mystical value describing how parameters should be dealt with.

**MAP SPECIFICATION**

The map file contains information on how to write the FITS data in bpipe format. In particular, it allows one to fill bpipe structures from separate FITS columns.

Any line beginning with a `#` character is ignored, as are empty lines.

Lines have the format

```
<directive> <options>
```

The available directives are:

**bpipe**

```
bpipe name type [array extent]
```

This creates a bpipe data packet field with the given name and bpipe type. The field may be a one-dimensional array; if so the extent must provided. Fields need only be explicitly created if they are arrays or if they are structures (e.g. `DVector3`)

**map**

```
map fitsname bp_name
map fitsname bp_name [bp_type]
```

This indicates that the FITS column indicated by `fitsname` will be written to the bpipe column

indicated by `bp_name`. The first form may be used if the `bpipe` field has previously been created with the `bpipe` directive. The second form may be used if the mapping is from a scalar FITS column to a scalar `bpipe` field.

`bp_name` may refer to an element in an array as well as a member of a `bpipe` structure. For example,

```
map x sky.x
map y foo[3]
map z snare[3].x
```

FITS bit fields retain their same format and layout as in the FITS file. The receiving `bpipe` field must have enough space for them (FITS stores bit fields as bytes). For example, a 32X specification in FITS stores 32 bits in 4 8-bit bytes. The `bpipe` field must be large enough. In this example, the following maps will work:

```
map bitfield bitfield int
```

```
bpipe bitfield char 4
map bitfield bitfield
```

In `auto_create` mode, bit fields are mapped onto character arrays.

Here's a full example:

```
bpipe chip IVector2
map chipx chip.x
map chipy chip.y
```

```
bpipe raw IVector2
map rawx raw.x
map rawy raw.y
```

```
bpipe det DVector2
map detx det.x
map dety det.y
```

```
bpipe sky DVector2
map x sky.x
map y sky.y
```

```
bpipe tdet IVector2
map tdetx tdet.x
map tdety tdet.y
```

```
bpipe au int 3
map au1 au[0]
map au2 au[1]
map au3 au[2]
```

```
bpipe av int 3
map av1 av[0]
map av2 av[1]
map av3 av[2]
```

```
bpipe crs IVector2
map crsu crs.x
map crsv crs.y
```

**BUGS**

Mapping bit fields onto structure members may not work.

Multi-dimensional FITS elements are not explicitly handled, nor has **fits2bp** been tested with them. **fits2bp** should exit with an error if it encounters them.

**COPYRIGHT & LICENSE**

Copyright 2006 Smithsonian Astrophysical Observatory

This software is released under the GNU General Public License. You may find a copy at

<http://www.fsf.org/copyleft/gpl.html>

**VERSION**

This documents version 1.0.6 of **fits2bp**.

**AUTHOR**

Diab Jerius ( [djerius@cfa.harvard.edu](mailto:djerius@cfa.harvard.edu) )