NAME

pds-to-cdf -- converts PDS label/table datasets to CDF datasets

SYNOPSIS

pds-to-cdf [**-H, --help**] [**-V, --version**] [**-O, --overwrite**] [**-D, --debug**] [**-c, --cdaweb**] [**-o, --output** *cdf-file-or-directory*] [**-t, --template** *template-cdf-file*] [**-e, --epoch** *time-type-spec*] **pds-label-file...**

DESCRIPTION

The **pds-to-cdf** program reads data from PDS-format label and table files, and generates a corresponding CDF file.

OPTIONS

-H, --help

Shows help information, briefly describing options.

-V, --version

Shows the version of the **pds-to-cdf** program.

-O, --overwrite

Allows an existing CDF file to be overwritten.

-D, --debug

Displays information on the PDS data file being processed. Multiple **--debug** options can be supplied to generate increasingly voluminous amounts of debugging output.

-o, --output *cdf-file-or-directory*

Specifies the file or directory into which the generated CDF files should be written. Without an **--output** option, the CDF files will be created in the same directory and with the same filename as the PDS file being processed, but with a *.cdf* filename extension. If the **--output** option specifies a directory, the CDF files will be written to the specified directory using name of the PDS file being processed, but with a *.cdf* filename extension. If the **--output** option does not specify a directory, whatever is specified will be used as is as the name of the output CDF file.

-c, --cdaweb

Derives the CDF filename and several CDAWeb-mandated CDF global attributes based on the first 'Epoch' variable as determined by the **--epoch** option, and other CDF global attributes which must be supplied using a CDF file specified by the **--template** option.

-e, --epoch *time-type-spec*

Specifies the name of the column and method of designating observation times that are to be used to generate the "Epoch" variables in the generated CDF file. The argument supplied must be one of "epoch" to extract observation times from an "EPOCH" column containing time values expressed as ISO-8601 character strings; "julian" to extract observation times from a "JULIAN_DATE" column containing time values expressed as floating point julian day numbers; "apollo15" to extract observation times from a "GROUND_ELAPSED_TIME" column containing time values expressed as a floating point number representing the time in elapsed seconds from the launch of Apollo 15, nomimally at 13:34:00 UT on 26 July 1971; or "apollo16" to extract observation times from a "GROUND_ELAPSED_TIME" column containing time values expressed as a floating point number representing the time in elapsed seconds from the launch of Apollo 16, nominally at 17:54:00 UT on 16 April 1972.

-t, --template *template-cdf-file*

Uses the specified *template-cdf-file* as the basis for the generated CDF file. This is generally used to supply global parameters to a CDF file in order to satisfy the requirements of the CDAWeb software.

EXAMPLES

To generate a CDF file in the current working directory with a filename of *wago.cdf* based on input files of *wago.lbl* and *wago.tab*, use:

```
$ pds-to-cdf --output=. /media/MGSC_1001/index/wago.lbl
```

BUGS

Currently, only PDS label/table files and image files can be used as input to the **pds-to-cdf** program. This may be corrected in future releases. Processing of image files is limited to those types of images that the PDS Object Access Library can read. Attempting to overwrite a malformed CDF file fails with a confusing message of "Read failed - error from file system." The file system is likely just fine, but you'll have to manually delete the CDF file before proceeding.

SEE ALSO

pds-to-cdaw(7), **cdfdump**(1), **skeletoncdf**(1), **skeletontable**(1)