

SUN/268.0

Starlink Project  
Starlink User Note 268.0

D.S. Berry & M.J. Currie

14th November 2013

---

**CVG**  
**A Library of Format Conversion**  
**Utilities**  
**Version 1.0**  
**Programmer's Manual**

---

## **Abstract**

CVG provides utility functions for handling data format conversion tasks.

## Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>A</b>	<b>Routine Descriptions</b>	<b>1</b>
	CVG_ASSOC . . . . .	2
	CVG_BT2FT . . . . .	3
	CVG_CLEAN . . . . .	4
	CVG_CLOSE . . . . .	5
	CVG_CREAT . . . . .	6
	CVG_FC2HD . . . . .	7
	CVG_FIOER . . . . .	8
	CVG_FT2BT . . . . .	9
	CVG_HD2FC . . . . .	10
	CVG_HDATE . . . . .	11
	CVG_HECHO . . . . .	12
	CVG_NEW . . . . .	13
	CVG_OPEN . . . . .	14
	CVG_PCADC . . . . .	15
	CVG_RETR <sub>x</sub> . . . . .	16
	CVG_SCADC . . . . .	17
	CVG_SHOWHEADER . . . . .	18
	CVG_WHISR . . . . .	19
	CVG_WPROV . . . . .	20

## **1 Introduction**

This library contains routines that originally formed part of the CONVERT application package (SUN/55), but which are of more general use and so have been moved into a separate library.

## **A Routine Descriptions**

---

## CVG\_ASSOC

### Opens an existing FITS file specified by an environment parameter

---

**Description:**

This function opens an existing FITS file with a path obtained from the environment, and returns a logical unit number that can be used to access it using CVG and FITSIO functions.

**Invocation:**

```
CALL CVG_ASSOC( PARAM, MODE, FUNIT, BLOCKF, STATUS )
```

**Arguments:****PARAM = CHARACTER \* ( \* ) (Given)**

The name of the environment parameter to use.

**MODE = CHARACTER \* ( \* ) (Given)**

The access mode: ' READ' or ' UPDATE. Case insensitive. Abbreviations can be used.

**FUNIT = INTEGER (Returned)**

The logical unit number of the FITS file. Returned equal to CVG\_NOLUN if an error occurs.

**BLOCKF = INTEGER (Returned)**

The logical record blocking factor.

**STATUS = INTEGER (Given and Returned)**

The global status.

---

## CVG\_BT2FT

### Creates an AST FitsTable from a FITS binary table

---

**Description:**

This function creates a new AST FitsTable object holding the data read from a FITS binary table held in a named extension of the supplied FITS file.

**Invocation:**

```
CALL CVG_BT2FT( FUNIT, EXTNAM, EXTVER, EXTLEVEL, TABLE, STATUS )
```

**Arguments:****FUNIT = INTEGER (Given)**

The FITSIO unit number for the FITS file.

**EXTNAM = CHARACTER \* ( \* ) (Given)**

The name of the FITS extension containing the required binary table.

**EXTVER = INTEGER (Given)**

The value of the EXTVER keyword in the required binary-table HDU.

**EXTLEVEL = INTEGER (Given)**

The value of the EXTLEVEL keyword in the required binary-table HDU.

**TABLE = INTEGER (Returned)**

A pointer to the new FitsTable. A value of AST\_NULL is returned (without error) if the named extension cannot be found.

**STATUS = INTEGER (Given and Returned)**

The global status.

---

## CVG\_CLEAN

### Removes standard header cards from a FitsChan

---

**Description:**

This removes the following cards, if present, from the supplied FitsChan: SIMPLE, BITPIX, EXTEND, NAXIS, NAXISj, all WCS related cards.

**Invocation:**

```
CALL CVG_CLEAN( IPROV, FC, STATUS )
```

**Arguments:**

**FC = INTEGER (Given)**

The FitsChan to clean.

**STATUS = INTEGER (Given and Returned)**

The global status.

---

## CVG\_CLOSE

### Closes a FITS file

---

**Description:**

This function closes the FITS files specified by the supplied logical unit number. It returns without error if the supplied logical unit number is currently not attached to a file.

**Invocation:**

```
CALL CVG_CLOSE( FUNIT, STATUS )
```

**Arguments:****FUNIT = INTEGER (Given and Returned)**

The logical unit number of the FITS file. Always returned equal to CVG\_\_NOLUN.

**STATUS = INTEGER (Given and Returned)**

The global status.

**Notes:**

- This function attempts to run even if an error has already occurred.

---

## CVG\_CREAT

### Create a new FITS file specified by an environment parameter

---

**Description:**

This function creates a new FITS file with a path obtained from the environment, and returns a logical unit number that can be used to access it using CVG and FITSIO functions.

**Invocation:**

```
CALL CVG_CREAT( PARAM, BLOCKF, OVRWRT, FUNIT, STATUS )
```

**Arguments:****PARAM = CHARACTER \* ( \* ) (Given)**

The name of the environment parameter to use.

**BLOCKF = INTEGER (Given)**

The blocking factor for the new file. It must be a positive integer between 1 and 10.

**OVRWRT = LOGICAL (Returned)**

If .TRUE., any existing file with the given name is silently over-written. Otherwise, an error is reported if the file already exists.

**FUNIT = INTEGER (Returned)**

The logical unit number of the FITS file. Returned equal to CVG\_NOLUN if an error occurs.

**STATUS = INTEGER (Given and Returned)**

The global status.

---

## CVG\_FC2HD

### Copies all headers from a FitsChan to the current HDU

---

**Description:**

This routine extracts all headers from the supplied FitsChan and stores them in the current HDU of the supplied FITS file. The HDU header may be emptied first (see argument CLEAR).

**Invocation:**

```
CALL CVG_FC2HD( FC, CLEAR, FUNIT, STATUS )
```

**Arguments:****FC = INTEGER (Given)**

Pointer to the FitsChan.

**CLEAR = LOGICAL (Given)**

Should the header be cleared before copying in the new cards? Otherwise, the new cards are appended to the end of the HDU.

**FUNIT = INTEGER (Given)**

The FITSIO unit number for the FITS file.

**STATUS = INTEGER (Given and Returned)**

The global status.

**Prior Requirements :**

The FITS file must already be opened with the FITSIO library.

---

## CVG\_FIOER

### Reports error messages associated with a call to a FITSIO routine

---

**Description:**

This routine writes a standard error report for an error arising within a FITSIO-library routine. The report comprises two parts. First there is a contextual error report. The error number is translated to an error string and appended to the supplied message. Thus the contextual message is of the form: <ROUTIN>: <MESSGE> Error was: or <MESSGE> Error was: when ROUTIN is a blank string. After this header message, the FITSIO error stack is flushed. If for some strange reason, there are no error messages in the stack, this routine appends the short translation of the error number, and if there is no translation, it reports the FITSIO error number and instruct the reader to consult the FITSIO User' s Guide.

The global status is set to SAI\_\_ERROR. This routine attempts to work even if the global status is bad on entry.

**Invocation:**

```
CALL CVG_FIOER( FSTAT, PARAM, ROUTIN, MESSGE, STATUS )
```

**Arguments:****FSTAT = INTEGER (Given)**

The FITSIO status.

**PARAM = CHARACTER \* ( \* ) (Given)**

The parameter name of the error report. It usually comprises the name of the routine where the FITSIO routine is called followed by some suitable suffix like ' \_ERR' .

**ROUTIN = CHARACTER \* ( \* ) (Given)**

The name of the FITSIO routine where the error occurred. It may be blank to prevent it appearing in the error report.

**MESSGE = CHARACTER \* ( \* ) (Given)**

A contextual error message to form part of the error report. It should indicate what was happening when the call to the FITSIO routine was made. No punctuation is added between this and the " Error was" text, so a full stop should appear in this message.

**STATUS = INTEGER (Given and Returned)**

The global status.

**Notes:**

The error name for the FITSIO error messages is FITSIO\_ERR.

[optional\_subroutine\_items]...

---

## CVG\_FT2BT

### Creates a FITS binary table from a AST FitsTable

---

**Description:**

This function creates a new FITS binary table holding data copied from an AST FitsTable. If a binary table extension with the given extension name already exists in the FITS file, it is replaced by the new one. Otherwise, the table is written into the current HDU. The current HDU on exit can be selected using MKCHDU.

**Invocation:**

```
CALL CVG_FT2BT( TABLE, FUNIT, EXTNAM, ASTVER, MKCHDU, STATUS )
```

**Arguments:****TABLE = INTEGER (Given)**

A pointer to the FitsTable.

**FUNIT = INTEGER (Given)**

The FITSIO unit number for the FITS file.

**EXTNAM = CHARACTER \* ( \* ) (Given)**

The name of the new FITS extension containing the binary table.

**ASTVER = INTEGER (Given)**

This parameter determines what to do if the FITS file already contains an extension with the name given by EXTNAM. Normally, an error is reported, but if the extension was created by AST (i.e. if EXTNAM is equal to the symbolic constant AST\_\_TABEXTNAME), and the table version for the existing table is equal to the value of ASTVER, then no error is reported and this function returns without action.

**MKCHDU = LOGICAL (Given)**

If .TRUE., the new binary table extension is made the current HDU on exit. Otherwise, the original current HDU is unchanged on exit.

**STATUS = INTEGER (Given and Returned)**

The global status.

---

## CVG\_HD2FC

### Copies all headers from the current HDU into a FitsChan

---

**Description:**

This routine extracts all headers from the current HDU and stores them in the supplied FitsChan. The FitsChan is first emptied.

**Invocation:**

```
CALL CVG_HD2FC( FUNIT, FC, STATUS )
```

**Arguments:****FUNIT = INTEGER (Given)**

The FITSIO unit number for the FITS file.

**FC = INTEGER (Given)**

Pointer to the FitsChan.

**STATUS = INTEGER (Given and Returned)**

The global status.

**Prior Requirements :**

The FITS file must already be opened with the FITSIO library.

---

## CVG\_HDATE

### Converts the NDF history date into a more-pleasing format

---

**Description:**

This makes a few minor modifications to the date string obtained from NDF history records to make it more like UNIX and astronomical style. Specifically two hyphens around the month are replaced by spaces, and the second and third letters of the month are made lowercase.

**Invocation:**

```
CALL CVG_HDATE( DATE, STATUS )
```

**Arguments:****DATE = CHARACTER \* ( \* ) (Given and Returned)**

On input the NDF format for a date and time, namely YYYY-MMM-DD HH:MM:SS.SSS. On exit, the ISO-order format for a date and time, namely YYYY Mmm DD HH:MM:SS.SSS

**STATUS = INTEGER (Given and Returned)**

The global status.

---

## CVG\_HECHO

### Writes history text to the FITS headers

---

**Description:**

This routine appends the history text associated with an NDF to the current FITS header. It is not called directly (by CVG\_WHISR), but is passed as an external argument to routine NDF\_HOUT. (It is an equivalent to NDF\_HECHO. See SUN/33 for more details.)

**Invocation:**

```
CALL CVG_HECHO( NLINES, TEXT, STATUS )
```

**Arguments:**

**NLINES = INTEGER (Given)**

Number of lines of history text.

**TEXT( NLINES ) = CHARACTER \* ( \* ) (Given)**

The history text.

**STATUS = INTEGER (Given and Returned)**

The global status.

**Notes:**

- The argument list should not be changed.
- There is no error checking of the FITSIO status.

**Prior Requirements :**

The NDF and the FITS file must already be open. The current HDU in the FITS file should be the primary and the other headers should have been written.

---

## CVG\_NEW

### Creates a new FITS file and return a unit number for it

---

**Description:**

This function creates a new FITS file with a given path, and returns a logical unit number that can be used to access it using CVG and FITSIO functions.

**Invocation:**

```
CALL CVG_NEW( PATH, BLOCKF, OVRWRT, FUNIT, STATUS )
```

**Arguments:****PATH = CHARACTER \* ( \* ) (Given)**

The path to the file to be created. A file type of ".fit" will be added if there is no file type in the supplied string.

**BLOCKF = INTEGER (Given)**

The blocking factor for the new file. It must be a positive integer between 1 and 10.

**OVRWRT = LOGICAL (Returned)**

If .TRUE., any existing file with the given name is silently over-written. Otherwise, an error is reported if the file already exists.

**FUNIT = INTEGER (Returned)**

The logical unit number of the FITS file. Returned equal to CVG\_NOLUN if an error occurs.

**STATUS = INTEGER (Given and Returned)**

The global status.

---

## CVG\_OPEN

### Opens an existing FITS file for read or update access

---

**Description:**

This function opens an existing FITS file with a given path, and returns a logical unit number that can be used to access it using CVG and FITSIO functions.

**Invocation:**

```
CALL CVG_OPEN( PATH, MODE, FUNIT, BLOCKF, STATUS )
```

**Arguments:****PATH = CHARACTER \* ( \* ) (Given)**

The path to the file to be created. A file type of ".fit" will be added if there is no file type in the supplied string.

**MODE = CHARACTER \* ( \* ) (Given)**

The access mode: ' READ' or ' UPDATE. Case insensitive. Abbreviations can be used.

**FUNIT = INTEGER (Returned)**

The logical unit number of the FITS file. Returned equal to CVG\_NOLUN if an error occurs.

**BLOCKF = INTEGER (Returned)**

The logical record blocking factor.

**STATUS = INTEGER (Given and Returned)**

The global status.

---

## CVG\_PCADC

### Writes CADC-style provenance records to the current FITS header

---

**Description:**

This creates headers in the current FITS header that record the number and names of all the immediate parents in the supplied NDG provenance structure. It also records the number of root parents—those without ancestors—and their observation identifiers from component OBIDSS within the MORE component of the supplied provenance structure. These are the observations.

The names follow CADC convention as follows. For the immediate parents:

PRVCNT = \_INTEGER / Number of parents PRV1 = \_CHAR / Name of the first parent PRV2 = \_CHAR / Name of the second parent : : : : PRVn = \_CHAR / Name of the PRVCNTth parent  
for the root provenance: OBSCNT = \_INTEGER / Number of root-ancestor headers OBS1 = \_CHAR / First observation identifier : : : : OBSn = \_CHAR / OBSCNTth observation identifier  
and the output file name: FILEID = \_CHAR / Filename

The above headers are prefaced by a blank header and a title " Provenance:" comment.

The PRODUCT keyword's value is modified for FITS extensions. It has '\_<extname>' appended where <extname> is the lowercase name of the standard EXTNAME keyword.

**Invocation:**

```
CALL CVG_PCADC( IPROV, FUNIT, STATUS )
```

**Arguments:****IPROV = INTEGER (Given)**

The identifier of the PROVENANCE that is to be written to the FITS headers. If NDG\_\_NULL is supplied, no provenance is stored in the header, but the PRODUCT keyword is still updated.

**FUNIT = INTEGER (Given)**

The logical unit number of the output FITS file.

**STATUS = INTEGER (Given and Returned)**

The global status.

**Notes:**

- Ancestors that have been flagged as " hidden" are ignored.
- A warning is issued if the OBSIDSS component cannot be found for a root ancestor. The value of OBSCNT gives the number of ancestors with an OBSIDSS value.

**Prior Requirements :**

The FITS file must already be open. The current HDU in the FITS file should be the primary and the standard headers should be present.

---

## CVG\_RETRx

### Retrieves a value from an array

---

**Description:**

The value stored at a given index within the supplied array is returned.

**Invocation:**

```
CALL CVG_RETRx( EL, INDEX, DATA, VALUE, STATUS )
```

**Arguments:****EL = INTEGER (Given)**

The number of elements in the array.

**INDEX = INTEGER (Given)**

The index within the array of the required value.

**DATA( EL ) = ? (Given)**

The input array.

**VALUE = ? (Returned)**

The returned value.

**STATUS = INTEGER (Given and Returned)**

The global status.

**Notes:**

- There is a routine for all numeric data types: replace " x" in the routine name by B, D, I, R, UB, UW, or W as appropriate. The VALUE and DATA arguments must have the data type specified.

---

## CVG\_SCADC

### Writes CADC-style provenance to a FITS file specified by an environment parameter

---

**Description:**

This opens the FITS file associated with the specified parameter, and adds headers to the primary FITS header that record the number and names of all the immediate parents in the supplied NDG provenance structure. It then closes the FITS file. It also records the number of root parents—those without ancestors—and their observation identifiers from component OBIDSS within the MORE component of the supplied provenance structure. These are the observations.

The names follow CADC convention as follows. For the immediate parents:

PRVCNT = `_INTEGER` / Number of parents PRV1 = `_CHAR` / Name of the first parent PRV2 = `_CHAR` / Name of the second parent : : : : PRVn = `_CHAR` / Name of the PRVCNTth parent

for the root provenance: OBSCNT = `_INTEGER` / Number of root-ancestor headers OBS1 = `_CHAR` / First observation identifier : : : : OBSn = `_CHAR` / OBSCNTth observation identifier

and the output file name: FILEID = `_CHAR` / Filename without extension

The above headers are prefaced by a blank header and a title " Provenance:" comment.

The PRODUCT keyword's value is modified for FITS extensions. It has '`_<extname>`' appended where `<extname>` is the lowercase name of the standard EXTNAME keyword.

**Invocation:**

```
CALL CVG_SCADC( IPROV, PARAM, STATUS )
```

**Arguments:****IPROV = INTEGER (Given)**

The identifier of the PROVENANCE that is to be written to the FITS headers. If NDG\_NULL is supplied, no provenance is stored in the header, but the PRODUCT keyword is still updated.

**PARAM = CHARACTER \* ( \* ) (Given)**

The name of the environment parameter associated with the FITS file.

**STATUS = INTEGER (Given and Returned)**

The global status.

**Notes:**

- Ancestors that have been flagged as " hidden" are ignored.
- A warning is issued if the OBSIDSS component cannot be found for a root ancestor. The value of OBSCNT gives the number of ancestors with an OBSIDSS value.

---

## CVG\_SHOWHEADER

### Displays all headers from one or all HDUs on standard output

---

**Description:**

This routine displays headers from the current HDU, or all HDUs, on standard output.

**Invocation:**

```
CALL CVG_SHOWHEADER( FUNIT, ALL, STATUS )
```

**Arguments:****FUNIT = INTEGER (Given)**

The FITSIO unit number for the FITS file.

**ALL = LOGICAL (Given)**

If .TRUE., list headers from all HDUs. Otherwise, list only the current HDU.

**STATUS = INTEGER (Given and Returned)**

The global status.

Copyright (C) 2013 Science & Technology Facilities Council.

All Rights Reserved.

---

## CVG\_WHISR

### Appends NDF history records to the current FITS header

---

**Description:**

This appends all the NDF HISTORY records in an easy-to-read format to the current FITS header.

**Invocation:**

```
CALL CVG_WHISR( NDF, FUNIT, STATUS )
```

**Arguments:****NDF = INTEGER (Given)**

The identifier of the NDF whose HISTORY records are to be written to the FITSD headers.

**FUNIT = INTEGER (Given)**

The logical unit number of the output FITS file.

**STATUS = INTEGER (Given and Returned)**

The global status.

**Prior Requirements :**

The NDF and the FITS file must already be open. The current HDU in the FITS file should be the primary and the other headers should have been written.

---

## CVG\_WPROV

### Writes general provenance records to the current FITS header

---

**Description:**

This creates headers in the current FITS header that record the supplied provenance information. The tabulated indexed headers below, all with string values, are written for each ancestor NDF. The keyword index *n* is the provenance identifier for each NDF, starting at 0 corresponding to the current NDF. All have value ' <unknown>' if the information could not be found, except for MORE information; the PRVM*n* header is omitted if there is no MORE information to record.

Keyword Comment

- ——— PRVP*n* Path of the <nth> parent PRVIn Identifiers of direct parents for <nth> ancestor PRVD*n* Creation date of <nth> parent PRVC*n* Creator software of <nth> parent PRVM*n* Contents of MORE of <nth> parent

where <nth> is the appropriate ordinal string, e.g. 1st, 2nd, 3rd, 4th.

The above headers are prefaced by a blank header and a title " Provenance:" comment. There is a blank header between each set for improved legibility.

**Invocation:**

```
CALL CVG_WPROV( IPROV, FUNIT, STATUS )
```

**Arguments:****IPROV = INTEGER (Given)**

The identifier of the PROVENANCE that is to be written to the FITS headers. If NDG\_\_NULL is supplied, this function returns without action.

**FUNIT = INTEGER (Given)**

The logical unit number of the output FITS file.

**STATUS = INTEGER (Given and Returned)**

The global status.

**Notes:**

- Supports up to 9999 ancestors.
- The PRVP*n*, PRVIn, and PRVM*n* keywords may often need more than the maximum 68 characters allowed in a keyword character value. Therefore this routine uses the non-standard Long-string Keyword Convention that permits long strings to continue over multiple headers. If any of the aforementioned keywords' values exceeds 68 characters, this routine adds a warning in some COMMENT headers and writes a LONGSTRN header, whose value is the version of the convention being used.

**Prior Requirements :**

The FITS file must already be open. The current HDU in the FITS file should be the primary and the standard headers should have been written.