AHELP for CIAO 3.4 **readrdb** Context: varmm

Jump to: Description Example CHANGES IN CIAO 3.1 CHANGES IN CIAO 3.0 Bugs See Also

Synopsis

S–Lang function to read a file in RDB format.

Syntax

```
Struct_Type readrdb( filename )
Struct_Type readrdb( filename, axes )
Struct_Type readrdb( filename, axes, nskip )

Error Return Value: NULL

Arguments:
filename is a String_Type variable
cols is either a String_Type variable or an Int_Type array
nskip is an Int_Type variable
```

Description

The readrdb() function allows you to read in data from RDB files into S–Lang. The ahelp page for readfile describes the features of this routine that are common to all the "read" functions provided by the Varmm module. This page describes those features that are unique to the readrdb() command.

The filename argument should be a string that contains the name of the file to be read in. Unlike the other "read" functions it can not include any Data–Model syntax. The optional parameter "cols" is a list of column names, or numbers, to be read, and "nskip" is the number of rows to be skipped. As examples:

```
chips> r1 = readrdb( "table.rdb" )
chips> r2 = readrdb( "table.rdb","1,3,5", 10 )
```

where the second example only reads in the first, third, and fifth columns and skips the first ten lines of the files.

File format

RDB files are ASCII files which use tabs to separate columns. They can contain a mixture of numeric and character columns, along with metadata and header information. Information on RDB, along with tools to access these files, is available from ftp://ftp.rand.org/pub/RDB–hobbs/.

The header of an RDB file is optional and is marked by a set of lines beginning with the "#" character. The first non–comment line gives the names of the fields as a set of names sepearated by tabs. The following line gives the type and sugested length of each column; again each volumn is separated by a tab character. Finally comes the actual data in tab–separated columns.

What does the function return?

The function returns a structure whose fields contain the data read in from the file. If an error occurred – such as the file not being found, or it is not a RDB file – then NULL is returned instead. The returned structure follows the format of the other "read" functions: metadata – i.e. information about the file – is stored in fields beginning with an underscore character followed by fields containing the input data. The initial fields are discussed in "ahelp readfile"; here we concentrate on those fields specific to RDB files.

Fields specific to tables:

Field name:	Description:
_header	This is a String_Type array that lists the header information of the RDB file. It does not use FITS format.
_ncols	The number of columns read from the file.
_nrows	The number of rows read from the file.
<column 1>	Contains the data for the first column read in. The name of the column matches the name from the rdb file.
<column 2>	Contains the data for the second column read in.

Example

If table.rdb contains:

```
# Table documentation lines. These describe and
# identify the rdbtable contents.
# They may be read by many normal UNIX utilities,
# which is useful to easily identify a file.
# May also contain RCS or SCCS control information.
NAME      COUNT    TYP      AMT      OTHER    RIGHT
6         5N       3        5N       8        8>
Bush      44       A        133      Another  This
Hansen    44       A        23       One      Is
Jones     77       X        77       Here     On
Perry     77       B        244      And      The
Hart      77       D        1111     So       Right
Holmes    65       D        1111     On       Edge
```

where the columns are separated by the tab character, rather than by spaces, then we can read it in using:

```
chips> rdb = readrdb( "table.rdb" )
chips> print( rdb )
_filename        =    table.rdb
_path            =    /data/analysis/
_filter          =    NULL
_filetype        =    2
_header          =    String_Type[5]
_ncols           =    6
_nrows           =    5
```

What does the function return?

```
    NAME            =   String_Type[5]
    COUNT           =   Double_Type[5]
    TYP             =   String_Type[5]
    AMT             =   Double_Type[5]
    OTHER           =   String_Type[5]
    RIGHT           =   String_Type[5]
```

and some of the fields contain:

```
chips> print( rdb._header )
# Table documentation lines. These describe and
# identify the rdbtable contents.
# They may be read by many normal UNIX utilities,
# which is useful to easily identify a file.
# May also contain RCS or SCCS control information.
chips> writeascii( stdout, rdb.NAME, rdb.AMT )
Bush    133
Hansen  23
Jones   77
Perry   244
Hart    1111
```

CHANGES IN CIAO 3.1

Enhanced documentation

There are now separate ahelp files for the individual functions, rather than having all the information placed into this document.

CHANGES IN CIAO 3.0

New field "_filetype"

A new field called "_filetype" has been added to the data structure which describes the type of the file read in. The contents of the field are described in the "Format of data structure" section above.

RDB support

The readrdb() routine has been added to allow reading in of RDB files. Note that readfile() does not recognise RDB files, so you have to call readrdb() directly.

Bugs

See the bugs page for the Varmm library on the CIAO website for an up–to–date listing of known bugs.

See Also

modules
> varmm

varmm
> fits_bitpix, readarf, readascii, readbintab, readfile, readimage, readpha, readrmf, writeascii, writefits

CHANGES IN CIAO 3.1