

5. APPLICATIONS

However, the message

More than MAXBOOK bookmarks requested

is not 'fatal', in the sense that the function `bkmrk_` returns `.false.` to permit appropriate action before termination. This is effectively a fatal error for which recompilation with a larger value of `MAXBOOK` is necessary. However, this is usually the result of a logic error in the application, and the error has been made non-fatal to allow the programmer to insert debugging code, if desired. The application should clean up and exit promptly.

5.4.11.2. Fatal errors: data sequence, syntax and file construction

Dict_ must precede ocif_

Dictionary files must be loaded before an input CIF is opened because some checking occurs during the CIF loading process.

Illegal tag/value construction

Data name (*i.e.* a 'tag') and data values are not matched (outside a looped list). This usually means that a data name immediately follows another data name, or a data value was found without a preceding data name. The most likely cause of this error is the failure to provide '.' or '?' for missing or unknown data values or a failure to declare a `loop_` when one was intended.

Item miscount in loop

Within a looped list the total number of data values must be an exact multiple of the number of data names in the `loop_` header.

Prior save-frame not terminated

Save-frame terminator found out of context. Save frames must start with `save_framecode` and end with `save_`. These messages will be issued if this does not occur.

Syntax construction error

Within a data block or save frame the number of data values does not match the number of data names (ignoring loop structures). This message should occur only if there is an internal logic error in the library. Normally the program will terminate on Item miscount in loop first.

Unexpected end of data

When processing multi-line text the end of the CIF is encountered before the terminal semicolon.

5.4.11.3. Fatal errors: invalid arguments

The following messages are generated by calls with invalid arguments.

Call to find_ with invalid arguments
Internal error in putnum

5.4.11.4. Warnings: input errors

Category <cat-code> first implicitly defined in cif

The category code in the DDL2 data name is not matched by an explicit definition in the dictionary. This may be intentional but usually indicates a typographical error in the CIF or the dictionary.

Data name <name> not in dictionary!

The data item name <name> was used in the CIF but could not be found in the dictionary.

Data block header missing

No `data_` or `global_` was found when expected.

Duplicate data item <name>

Two or more identical data names <name> have been detected in a data block or save frame.

Exponent overflow in numeric input

Exponent underflow in numeric input

The numeric value being processed has an exponent that cannot be processed on this machine. If the string involved is not intended as a number, then surrounding it with quotes may resolve the problem.

Heterogeneous categories in loop <new cat-code>

vs <old cat-code>

Looped lists should not contain data items belonging to different categories. This error occurs if the category of a new data item fails to match the category of a prior data item. A special category (none) is used to denote item names for which no category has been declared. Warnings are not issued on this level for a loop for which all data items have no declared category.

Input line length exceeds line_

Non-blank characters were found beyond the value given by the variable `line_`. The default value for `line_` is 80 (optionally increased to 2048 in *CIFtbx 2.7* and later for CIF 1.1 compliance). The extra characters in column positions `line_ + 1` through `MAXBUF` will be processed but the input file may need to be reformatted for use with other CIF-handling programs.

Missing loop_ items set as DUMMY

A looped list of output items was truncated with an incomplete loop packet (*i.e.* the number of items did not match the number of `loop_` data names). The missing values were set to the *character string* 'DUMMY'.

Numb type violated <name>

The data item <name> has been processed with an explicit dictionary type `numb`, but with a non-numeric value. Note that the values '?' or '.' will *not* generate this message.

Quoted string not closed

Character values may be enclosed by bounding quotes. The strict definition of a 'quoted-string' value is that it must start with a <wq> digraph and end with a <qw> digraph, where `w` is a white-space character blank or tab and `q` is a single or double quote, and the same type of quote mark is used in the terminal digraph as was used in the initial digraph. This message is issued if these conditions are not met.

5.4.11.5. Warnings: output errors

Converted pchar_ output to text for <string>

An attempt was made to write a string with `pchar_` instead of `pTEXT_`, but the string contains a combination of characters for which `pTEXT_` must be used.

ESD less than precision of machine

Overflow of esd

Underflow of esd

A call to `pnumb_` or `numb_` was made with values of the number and standard uncertainty (e.s.d) which cannot be presented properly on this machine. A bounding value of 0 or 99999 is used for the e.s.d.

Invalid value of `esdlim_` reset to 19

In processing numeric output, a value of `esdlim_` less than 9 or greater than 99999 was found. `esdlim_` is then set to 19.