

SUN/185.2

Starlink Project
Starlink User Note 185.2

A J Chipperfield
B K McIlwrath
4 November 1994

MESSEGEN

Starlink Facility Error Message Generation Version 1.2

Abstract

This document describes the Starlink MESSGEN program which creates files required for proper operation of the system for associating messages with Starlink facility status values on UNIX platforms. It should be read by authors of Starlink subroutine libraries.

The utility program CREMSG is also described. It can be used to produce source files for MESSGEN.

Contents

1	Introduction	1
2	MESSEGEN Source Files	1
3	Running MESSEGEN	2
4	Running CREMSG	3
5	Using the Facility Message Files	3
6	Changes in Version 1.1	4
7	Changes in Version 1.1-1	4
8	Changes in Version 1.2	4

1 Introduction

The Starlink MESSGEN facility is a system devised and written by Brian McIlwrath to enable error messages to be associated with Starlink STATUS values on UNIX systems. This feature is available with the VAX/VMS error system and has proved useful.

MESSGEN can create a Fortran INCLUDE file and a C header file defining symbolic names for the status values and a *facility message file* which is used by the Starlink Error Message Service (EMS) to associate messages with the status values.

A utility program CREMSG is also provided in the MESSGEN release. It may be used to create a MESSGEN source file from an existing, standard-format Fortran error INCLUDE file.

2 MESSGEN Source Files

The starting point for the Starlink MESSGEN facility is a source file conventionally named *facility_err.msg*. The format is very similar to that for the VAX/VMS MESSAGE facility. This is most easily described by an example:

```
.TITLE          PAR          ADAM/SSE parameter routines
.FACILITY       PAR,190/PREFIX=PAR__
.IDENT         'Version 0.5'
!author        REVAD:BDK 18Nov84 (B.D.Kelly)

.SEVERITY      INFORMATIONAL
.BASE 100

NOUSR          <No user available for input>
PARNF         <Parameter does not exist>
EOL           <End of parameter list>
NULL          <Null parameter value>
ABORT         <Parameter request aborted>

.END
```

Notes:

- (1) Lines may be up to 100 characters long but may not be continued to the next line.
- (2) ! indicates that the remainder of the line is comment, except on a .TITLE line.
- (3) The .TITLE directive is optional but recommended. The given text is displayed as the file is read by MESSGEN.
- (4) The .FACILITY directive specifies the facility name, the facility number (which is allocated by the Starlink Infrastructure Support Group) and the prefix to be used for symbolic constants for the status values. In the example, the facility name is PAR, the facility number is 190 and the symbolic constant for the first error value will be PAR__NOUSR.

- (5) The .IDENT directive is optional and has no effect.
- (6) The .SEVERITY directive defines the severity of the error. In addition to INFORMATIONAL, any of the values permitted by the VAX/VMS MESSAGE facility are allowed. The severity value has no meaning on UNIX, or in most cases on VAX/VMS, but it does affect the error value generated for subsequent errors in the file. There may be more than one .SEVERITY directive in the file or it may be omitted, in which case WARNING is assumed.
- (7) The .BASE directive specifies a message number to be associated with the next message and is optional. In the absence of any .BASE directives, message numbering is sequential (from 1) within each facility.
- (8) Message lines specify the ident of the message and the text (enclosed in <>).
- (9) SUN/104 gives a formula for calculating unique error numbers. This formula corresponds with .SEVERITY WARNING.
- (10) The .END directive is optional and has no effect.
- (11) If the VAX/VMS MESSAGE facility was used to generate facility INCLUDE files which were subsequently ported to UNIX, the original .MSG source files should be used to generate the error files for UNIX. This will ensure that the same error values are generated.

3 Running MESSGEN

To produce the required files from MESSGEN source files, on UNIX, type:

```
% messgen -[cfFev] files
```

where *files* may be any number of MESSGEN source files and the options may be one or more of:

- c Generate a C header file (*facility_err.h*).
- f Generate a Fortran INCLUDE file (*facility_err*).
- F As for option -f, but with an uppercased output filename (*FACILITY_ERR*).
- e Generate a facility message file (*fac_facnum_err*).
- v Output diagnostic information.
- V Output the names of generated files to stdout.

where *facility* is the facility name, e.g. PAR, and *facnum* is the facility number, e.g. 190 for PAR.

4 Running CREMSG

If a MESSGEN source file is not available but there is an existing, standard-format Fortran error INCLUDE file, program CREMSG can create a MESSGEN source file from it. Standard-format means that the error numbers are defined by statements of the form:

```

        INTEGER PAR__NOUSR
        PARAMETER ( PAR__NOUSR = 146703139 )
*   No user available for input

        INTEGER PAR__PARNF
        PARAMETER ( PAR__PARNF = 146703147 )
*   Parameter does not exist

... etc ...

```

i.e. the format produced by the VAX/VMS MESSAGE/ERRGEN system.

To produce the corresponding MESSGEN source file, type:

```
% cremsg [-v] filename
```

where *filename* is the name of the INCLUDE file and option *-v* may be used to obtain diagnostic information.

Notes:

- (1) The file produced will be named *fac_err.msg* file, where *fac* is the lower case of the facility name determined from the error idents in the INCLUDE file (**par** in the above example). This will not always give the correct name.
- (2) The .TITLE directive produced will contain only the facility name. It may be desirable to add a short description by hand.
- (3) CREMSG will ignore any initial heading comment lines in the INCLUDE file.
- (4) CREMSG will accept the message comment line either first or last in each definition sequence in the INCLUDE file but a blank line must be present between each sequence.

5 Using the Facility Message Files

Normally the facility message file for each Starlink library will be installed in **/star/help** when the library is installed. It is recommended that the message file and error include files are included in the source tar file rather than building them from the MESSGEN source file during the **make build**. This will avoid an unnecessary dependency.

The internal EMS source file **ems1_fcerr.c** contains C code for obtaining the message given the status value – this may be copied by other systems requiring to do the same job on UNIX platforms but note that the system is not fully portable and Starlink reserve the right to alter it. See SSN/4 (Appendix on Portability) for a description of how the routine operates on UNIX.

6 Changes in Version 1.1

- The CREMSG utility is added (Section 4).
- The first issue of this document gave the wrong .SEVERITY value corresponding with the formula in SUN/104 – this has been corrected (Section 2 Note 9).

7 Changes in Version 1.1-1

- Generated C header files now contains a conditional statement to prevent repeat definitions.
- A hypertext version of this document is released. Apart from the addition of this section, there are no changes to the text.

8 Changes in Version 1.2

Added the -F option to generate Fortran output files with uppercase filenames.