X11/95-63

MUMPS Development Committee

Extension to the MDC Standard
Type A Release of the MUMPS Development Committee

Naming string length error

February 1994

Produced by the MDC Subcommittee #15
Programming Structures

Thomas Salander, Chairman
MUMPS Development Committee

Kate Schell, Chairman
Subcommittee #15

The reader is hereby notified that the following MDC specification has been approved by the MUMPS Development Committee but that it may be a partial specification that relies on information appearing in many parts of the MDC Standard. This specification is dynamic in nature, and the changes reflected by this approved change may not correspond to the latest specification available.

Because of the evolutionary nature of MDC specifications, the reader is further reminded that changes are likely to occur in the specification released, herein, prior to a complete republication of the MDC Standard.

Anyone reproducing this release is requested to reproduce this introduction.

1. Identification of proposed change

1.1 Title

Naming string length error.

1.2 MDC Proposer and Sponsor

SC15/TG11 — Portability Size Issues
Chair: Rod Dorman
Polylogics Consulting, Inc.
136 Essex Street
Hackensack NJ 07601

phone: (201) 489-4200
fax: (201) 489-4340
email: rodd@panix.com

1.3 History

April 1995	<this document>	Changed M__ to M75
June 1994	X11/94-16	Final editorial changes made, submitted as a final MDC type A document.
February 1994	X11/SC15/TG11/93-4	Submitted to become a MDC type A. Vote passed 25-0-3.
October 1993	X11/SC15/TG11/93-4	Task group editorial changes made, submitted to become a SC15 type A. Vote passed 18-1-3.
March 1993	X11/SC15/TG11/93-2	Original proposal submitted to become a SC15 type B. Vote passed 28-3-3.

1.4 Last Set of Pros and Cons

Pro	Con
1. Standardizes error message	1. Forces a new error.
2. Standardizes behavior	2. Uses M instead of MUMPS
3. Forces a new error	

In regards to con one, the author knows of no implementation that does not raise some form of "string too big" error when its limit (if any) is exceeded. Thus this proposal is not introducing a new error, it's simply standardizing the error.

As to con two, this is an editorial issue encompassing the entire standard, not just this proposal. Since typing **M** instead of **MUMPS** saves four keystrokes this author will continue to do it unless the guidelines for document preparation explicit prohibit it.

2. Justification of proposed change.

2.1 Needs

Section one of the **M** standard defines the syntax of strings and section two defines the portability requirements. Neither identifies what should occur when the string length is exceeded. Some ludicrous possibilities would be:

- truncate excess characters from the end of the string
- truncate excess characters from the beginning of the string
- randomly remove characters until the length is equal to the limit

No implementation known to the author chooses any of these options, they all return some kind of error. However, it's currently up to the implementor as to what the ecode will be. This proposal specifies what the ecode will be for a "string length error".

2.2 Existing practice in Area of Proposed Change

Any routine wishing to detect and identify a "string length error" has to use implementation specific values.

3. Description of Proposed Change

3.1 General description of proposed change

The effect of exceeding an implementor's string length limit is defined to be an error, and the ecode for that error is specified.

3.2 Annotated examples of use

None that would be helpful to clarify the proposal.

3.3 Formalization

In Section II Clause 2.8 add a paragraph at the end to read:

> *If a string's length exceeds an implementor's limit, an error condition occurs with ecode = "M75".*

4. Implementation Effects

4.1 Effect on Existing User Practices and Investments

It will become possible to write standard error handler code for "string length errors".

4.2 Effect on Existing Vendor Practices and Investments

It's not anticipated that this proposal will have any other impact beyond specifying what ecode value a "string length error" produces.

4.3 Techniques and Costs for Compliance Verification

Attempt to create a string, the length of which would be vendors limit+1 characters, and ensure that an "M75" error occurs.

4.4 Legal Considerations

None known

5. Closely Related Standards Activities

5.1 Other X11 Proposals Under Consideration

Portability limit of string length.

5.2 Other Related Standards Efforts

None known.

5.3 Recommendations for Coordinating Liaison

None.

6. Associated Documents

None.