MUMPS Development Committee

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

Portable length limit of strings

March 19, 1997

Produced by the MDC Subcommittee #15
Programming Structures

Ed de Moel, Chairman
MUMPS Development Committee

Art Smith, 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.

1. Identification of proposed change

1.1 Title

Portable length limit of strings.

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) 646-1422
email: rodd@panix.com

1.3 Motion

None, final MDC type A document superseding X11/SC15/97-4.

1.4 History

August 1997	X11/97-23	Final version of this document.
March 1997	X11/SC15/97-4	History updated, submitted for task group discussion and elevation to MDC type A. Passed 27-0-0
September 1996	X11/SC15/96-18	Modified as per task group discussion, submitted for additional task group discussion and elevation to SC15 type A. Passed 22-3-5
March 1996	X11/SC15/96-6	Modified as per task group discussion, submitted for additional task group discussion and reaffirmation as SC15 type B. Passed 17-1-4
October 1995	X11/SC15/95-36	Submitted for task group discussion and elevation to SC15 type A. Task group decided not to present it.
April 1995	X11/SC15/TG11/95-2	Original proposal submitted for task group discussion and adoption as a SC15 Type B document. Passed 9-7-3
January 1995		Task group discussion on increasing limit

1.5 Dependencies

None

2. Justification of proposed change.

2.1 Needs

The current portable string length limitation (510 characters) is out of date when compared to other programming environments.
The MWAPI has a portability limit requirement of 32,767 for longchars

2.2 Existing practice in Area of Proposed Change

Some implementations allow increased string lengths. If an application takes advantage of the increase it becomes non-standard and possibly non-portable.

3. Description of Proposed Change

3.1 General description of proposed change

The portability limit for the maximum length of a string is changed from one limit to separate limits by string length category. Some of the limits are bring increased. A vendor may still exceed any portability limit if they desire.

3.2 Annotated examples of use

```
SET X=$J("",2**15-1)
FOR I=1:1:$L(X)\500 SET ^FOO(SomeNode,I)=$E(X,500*(I-1)+1,500*I)
SET:$L(X)#500 I=$L(X)\500+1,^FOO(SomeNode,I)=$E(X,500*(I-1)+1,$L(X))
```

3.3 Formalization (References are to Canvass Document for ANSI/MDC X11.1-1994 as amended by X11/94-28)

In Section II Clause 2.3.2 (Number of subscripts), first paragraph, second sentence, replace the final phrase which currently reads:

the total length of the array reference must not exceed the maximum string length (see 2.8).

with:

the total length of the array reference must not exceed 510.

Editorial note: this is not a change in value, 510 was established in X11/94-28

In Section II Clause 2.4.2 (Number of subscripts), first paragraph, first sentence, replace the final phrase which currently reads:

the total length of the array reference must not exceed the maximum string length (see 2.8).

with:

the total length of the array reference must not exceed 510.

Editorial note: this is not a change in value, 510 was established in X11/94-28

In Section II Clause 2.8 (Character strings), first paragraph, first sentence, replace the entire sentence which reads:

Character string length is limited to 510 characters.

with:

Character string length is limited to 32,767 characters for local variables, 510 characters for global variables, 32,767 characters for structured system variables, .

In Section II Clause 3.2 (Results), first paragraph, first sentence, replace the entire sentence which reads:

Any result, whether intermediate or final, which does not satisfy the constraints on character strings (see 2.8) is erroneous.

with these two sentences:

Any final result which does not satisfy the constraints on character strings (see 2.8) is erroneous. Any intermediate result which does not satisfy the constraints on local variable character strings (see 2.8) is erroneous.

In Section II Clause 4.1 (Command lines), first paragraph, first sentence, replace the entire sentence which reads:

A command line (line) must satisfy the constraints on character strings (see 2.8).

with:

A command line (line) must satisfy the constraints on global variable character strings (see 2.8).

In Section II Clause 8 (Storage space restrictions), first paragraph, first sentence, replace the entire sentence which reads:

The size of a single routine must not exceed 10,000 characters.

with:

The size of a single routine must not exceed 20,000 characters.

In Section II Clause 8 (Storage space restrictions), first paragraph, second sentence, replace the entire sentence which reads:

The size of local variable storage must not exceed 10,000 characters.

with:

The size of local variable storage must not exceed 100,000 characters.

4. Implementation Effects

4.1 Effect on Existing User Practices and Investments

Programmers using MWAPI may create longchars's in a portable manner. They may have to purchase additional memory.

4.2 Effect on Existing Vendor Practices and Investments

Implementers may have to modify their data structures to accommodate longer local variable strings. Some Implementers already have. This will probably be mitigated by the fact that other language extensions (e.g. multinational or multibyte character sets) will also necessitate rethinking of internal data structures. Two vendors have stated that 32767 would not be a problem.

4.3 Techniques and Costs for Compliance Verification

```
KILL  SET X=$J("",2**15-1)
```

and ensure no "M75" error occurs.

4.4 Legal Considerations

None known

5. Closely Related Standards Activities

5.1 Other X11 Proposals Under Consideration

The SC13 Local Variable Storage proposal (X11/SC13/TG15/95-6).

5.2 Other Related Standards Efforts

None known.

5.3 Recommendations for Coordinating Liaison

None.

6. Associated Documents

X11/94-28 Portable String Length, published in the July 94 mailing, page 187.

7. Issues, Pros and Cons, and Discussion

June 1995 MDC meeting

Pro	Con
1. Addresses MWAPI's portability requirement	1. Incredibly difficult to implement (esp. re globals and subscripts) 2. Should use several different limits. 3. Goes too far

Re cons 1 and 2, the task group will explore different limits for different "strings".
Re con 3, no discussion or straw poll explored alternate values for the limit, the idea behind discussion is to give guidance to the task group, if no one mentions or suggests an alternate the task group will assume acceptance of its selection.

October 1995 MDC meeting

No formal pros & cons were produced however much discussion took place in the task group. Valuable vendor input pointed out that the cost of increasing the limit for gvn's was much greater than for lvn's which resulted in the task group decision to "split" the string length limit into separate categories.

March 1996 MDC meeting

Pro	Con
1. More degrees of freedom for string length 2. Increases lvn string length	1. lvn → gvn may fail

Re con 1, yes, this is a possibility. The task group feels however that the benefits outweigh this con. Note that there are already existing implementations that have lvn limits greater than gvn limits.

September 1996 MDC meeting

There was much discussion of David Marcus' comments which highlighted the fact that X11/94-28 has already raised the length of an array reference to 510. As a result, a proposal may be forthcoming that would reduce it back to the former value of 255. Note that this (current) proposal (if passed) will make that task easier.

Pro	Con
1. Addresses MWAPI's portability requirement	1. Reaffirms unexpected side effect of raising array ref to 510

Re con 1, this is a roundabout con to X11/94-28. As mentioned in the above discussion this proposal would simplify the changing of 510 back down to 255.

March 1997 MDC meeting

Final list of pros and cons.

Pro	Con
1. addresses MWAPI's portability requirement 2. frequent user requests	