

Voting Summary on JTC 1/SC 22 N 3673

P-Member	Approve	Approve w/Comments	Disapprove	Abstain
Austria				
Belgium				
Brazil				
Canada				
China				
Czech Republic				
Denmark				X
Egypt				
Finland				
France				
Germany			x	
Ireland	X			
Italy				
Japan	X			
DPR of Korea				
Republic of Korea	X			
Netherlands	X			
Norway				
Romania				
Russian Federation				
Slovenia				
Switzerland				Abstain
Ukraine				
United Kingdom	X			
United States		X (see below)		

Comments:**Germany:**

TR 10176 deals with the general principles of writing programming language standards and is thus at the core of SC22's scope. Maintaining an essential part of that standard in a

different SC will make the work on this important TR next to impossible and will not yield any benefits.

United States:

The U.S. supports with comment, proposal N3673 that SC22 make a request to SC2/WG2, such that when SC2/WG2 add new characters to ISO 10646, that SC2/WG2 also specifies those new characters that would also be acceptable as identifiers.

The comments are:

1) Having the JTC1/SC2 program of work include this project (as outlined in SC22 N3673) would allow for timely and continuous synchronization with the repertoire of ISO 10646/Unicode, as future amendments of 10646 are adopted.

2) The work of identifying extended characters is likely to have broad interest to JTC1 subcommittees, such as SC7, SC11, SC22, SC23, SC25, SC28, SC31, SC32, SC34, SC35, and SC36. It is recommended that this work be circulated at the JTC1 level so that JTC1 subcommittees may be better informed of these SC2 activities, and so that JTC1 subcommittees may harmonize their work with the SC2 activities.

In addition to identifying extended identifier characters, it may be important for many ICT applications to know what characters are also considered extended digit characters, extended letter characters, and other classes of extended special characters.

In prior work on extended identifiers, knowledgeable experts provided advice and guidance that helped properly classify characters. It would be extremely helpful to capture this knowledge in a "rationale/criteria annex" or a companion technical report so that users of these classifications could be better informed about the intended uses, the special circumstances, the limitations, and so on.

While it is not necessarily a classification function, certain relationships are important for common operations on identifiers, such as the familiar upper-lower case usage in Latin letters (e.g., leading capitals, "camel case", etc.). It has been discovered that additional character transformations for identifiers are in use in non-Latin scripts in ICT applications (e.g., full vs. half width for Japanese identifiers). It would be helpful if the character set experts in SC2 could identify some of these popular conventions so that they could be better understood and processed more consistently in ICT environments.

3) As long as this change involves

* asking SC2 to specify those characters which they believe are suitable for identifiers, but

* leaves each programming language standard (and SC22 and other SCs and work programs generally) free to specify its own identifier-character list after considering the tradeoffs between its requirements and the advantages of a consistent, single, identifier specification, then this proposal and position seems entirely appropriate. However, this is not clear from either the draft letter nor from the material sent out with the ballot -- those can be read to imply that programming languages can extend, but not restrict, the identifier character list (10176 is, apparently, fairly clear on this matter, but that material is not cited here)