

SC22/WG20 N956D

Title: Disposition of comments on DTR2 of 14652 (draft)

Date: 2002-06-11

Source: ISO/IEC JTC1/SC22/WG20

Status: WG20 draft

References: JTC 1 N6769, WG20 N951, JTC 1 N6721

In the following the disposition of comments is given with respect to the DTR ballot in JTC 1 N 6769, Information technology - Specifications for Cultural Conventions.

The ballot ended with 9 JTC 1 P-members voting for the draft to be approved as presented, 1 P-member for approving the draft with comments, 6 P-members disapproving the draft with comments, 3 P-members abstaining, and 9 P-members not voting. In total 10 P-members approved the draft with or without comments, and 6 P-members disapproved the draft. A majority of the voting JTC 1 P-members has thus approved the draft and according to JTC 1 directives the DTR has thus been approved.

Disposition of comments:

Comments from Germany

Germany will change its vote to approval if its comments are satisfactorily resolved.

Statement of clarification:

Germany has always opposed the development of 14652 as an IS and will continue to do so in the future, even if all of its comments on this DTR should be met and if it should in consequence change its vote to approval for the vote on this DTR.

Germany sees little use in this DTR. It has only very limited support in the industry (not even in the Linux community, cf. the comments from Ulrich Drepper in document WG20/N922). However, Germany notes that the editor has taken steps to resolving German comments of the previous rounds by marking the controversial parts of the DTR as such (altogether roughly half of the document is marked as controversial). Whatever limited use the DTR may have in the face of these controversies may come by completing it now ASAP, warts and all, and let implementors evaluate it.

Comments (with decreasing severity):

1. Section 7: Remove this section with the conformance clause altogether to avoid any mistaking of this DTR for a future IS

1. Accepted.

2. In view of the move of ISO from classical TRs of type 1 and 2 to TSs consider making this TR a TR of type 3.

Noted. This consideration could be made in due time, when more experience with implementation has been collected.

3. Section 4.5: LC_MONETARY: The double currency in one locale is the bad solution to an obsolete problem and must not be maintained

Rejected. The complete section 4.5 is marked controversial.

4. Section 4.3.2 (LC_CTYPE): The current classification is an unfortunate duplication of the work of the Unicode Consortium and may lead to confusion. At the very minimum, this section must also be marked as controversial.

Noted, the section will be marked as controversial.

5. Other comments that may be considered to have already been dealt with by marking the relevant sections as controversial. Some examples:

Section 6: The selection of the characters for the repertoire map is arbitrary. The system used to denote the symbolic character names is idiosyncratic.

The solution to transliteration (LC_XLITERATE) is inadequate for most purposes but used in practice as one (!) of several transliterations in the iconv tool (cf. Drepper's document) and can therefore be maintained for the time being.

Noted, the sections are marked as controversial.

Comments from Ireland

1. DTR 14652 was so flawed that it did not get sufficient votes a year ago, when it was presented to the JTC1 member bodies for the first time. Ireland voted against it at that time. DTR2 14652 has now been reissued with changes. However, we find that many of the technical comments from the first DTR ballot have been rejected or have not been adequately addressed. Accordingly, Ireland must vote NO again on this

Noted.

2. We have been made aware of the US NB's extensive comments regarding the flaws in this document, and we consider that they point out the flaws comprehensively and correctly.

Noted. Other comments are handled under response to the comments from the USA, as the Irish comments do not have specific technical content.

3. Ireland favours the immediate cancellation of this controversial work item.

Noted. The committee does not have consensus to cancel the work item.

Comments from Japan

The National Body of Japan disapproves ISO/IEC DTR 14652 for the reasons below.

1. Japan observes that the proposed TR does not address many technical comments from National bodies of ISO/IEC through previous DTR ballot, correctly.

Noted. The committee does not have consensus to cancel the work item.

2. For example, Germany commented that the TR should cover at least ISO/IEC 10646:2000 but the current draft still refers to ISO/IEC 10646:1993 with AM 1 through 9 and 18.

Noted. WG20 agrees that updating the TR to cover a more recent version of 10646 would be beneficial, but also time consuming, and would not be in the scope of the current work which is to publish as a TR type 1 the work that could not be approved as a standard. Furthermore the DTR refers to IS 14651 which has the same repertoire.

3. Another example is that US commented to remove LC_XLITERATE section since the proposed syntax is too weak to meet the requirement of transliteration for Asian languages, but the section is still there.

Noted, the section are marked as controversial.

Comments from Norway

In order to preserve the work of WG20 the following work is proposed to be reinstalled from earlier drafts:

1. LC_PAPER category
2. LC_MEASUREMENT category
3. The double symbolic ellipses ..(2).. - but no changes to the data specifications.

Rejected. These specifications were controversial in earlier drafts and thus removed. The Norwegian member body is kindly invited to submit text on these unresolved issues for annex D.

Comments from Sweden

1. Sweden is of the opinion that DTR 14652 is not up to date according to e.g. ISO/IEC 10646.

Noted. Wrt to updated repertoire of IS 10646, see response 2 to Japan.

2. Also in a TR Type 1 there shall be clearly stated in the Foreword why the required support could not be obtained for the IS. If this is included in the Foreword Sweden will change the vote to Approval

Accepted. A statement of why the specification could not be approved as in International Standard will be added, it was due to lacking consensus of the participating members of the working group. (replacing line 69)

Comments from Switzerland

Justification:

1. SC20/WG20 has not been able to arrive at a reasonable level of consensus on this document and, therefore, it should not be published.

Noted, see response 3 to Ireland.

2. The character repertoire defined in this TR is completely obsolete, and completely outdated compared with ISO/IEC 10646. There is no complete and correct specification of an FDCC set, even the Euro is missing.

Noted. see response 2 to Japan.

3. The TR contains several errors (syntax, spelling, definitions, format descriptors).

Noted.

The UK provided some late comments in WG20 N951 that are responded to here at the request of the SC22 secretariat.

Late comments from the UK

Due to unfortunate circumstances, the UK's position in respect of this ballot was not received by us at BSI until after the ballot closed on 24 May 2002.

The UK's position is that we would have voted 'NO' in the ballot.

The reasons for our negative position are:

(i) The UK believes that comments submitted at previous stages of the development of this standard have not been dealt with to the UK's satisfaction.

Noted

(ii) The phrasing of the Technical Report remains as if it were a putative standard, and would be very confusing to readers who did not understand the significance of the Type classification of Technical Report

Accepted, see response 2 to Sweden and Response 1 to Germany

(iii) The purpose of the standard and its value to the wider community of users of standards is not clear.

Noted

Once again, we regret that we were unable to submit this position to the formal time scale and offer our apologies. The UK would be grateful the above position could be passed on to SC22 for consideration.

Comments from US

EXECUTIVE SUMMARY OF OBJECTIONS

The U.S. National Body still has serious objections to DTR 14652 that have not been addressed, or have been addressed inadequately, in previous drafts.

Among our major concerns are:

* Five major sections of the document and several keywords are listed as controversial because WG20 members were unable to reach agreement on the functionality. Publishing a TR for which there is so little consensus is detrimental to international standardization efforts.

- * The repertoire used in this DTR is ISO/IEC 10646 as it was defined in 1998 (equivalent to Unicode V2.1). More than 55,000 characters have been added to those universal code sets since 1998. This DTR is completely obsolete as written; it should not be published with an obsolete repertoire.
- * The functionality defined for "class combining" and "class combining_level3" violates the definition in ISO/IEC 10646.
- * The DTR provides two places to define character width. Defining one thing in two places is bad design and promotes implementation errors.
- * The LC_CTYPE section includes many errors (missing or incorrectly specified groups of characters) as well as many unexplained differences between its classifications and the de facto standard Unicode classifications.
- * There are syntactic errors in the FDCC-set "i18n" LC_COLLATE section.
- * The controversial attempt to support multiple currencies in LC_MONETARY incorrectly treats national and EU currencies as synonyms (e.g., French francs as equivalent to euros) rather than as being two separate currencies that had simultaneous use. Also, the specification includes errors that prevent correct use of those multiple currencies for some countries.
- * The controversial LC_TIME section breaks compatibility with POSIX.2 regarding weekdays. It also incorrectly includes timezone information within an FDCC-set, but without providing any way for users in countries that span multiple time zones to indicate the zone that they need to use. The TZ environment variable already provides adequate functionality in this area.
- * The controversial LC_XLITERATE section is inadequate and incomplete for most languages, including most Asian ones. It should be removed.
- * Many format descriptors in LC_NAME, LC_ADDRESS, and LC_TELEPHONE are inadequately defined.
- * There are errors in the description of charmaps, including multiple

references to a non-existent table.

* There is a 27-page "i18nrep" repertoire map that covers less than 10% of the repertoire this DTR says it supports, and no information about how to specify the actual repertoire for a given FDCC-set. Even the euro isn't in i18nrep!

* There are several references to an "i18n" FDCC-set throughout the DTR, but no full example of it, leaving many implementation details undefined.

In addition to these problems, the U.S. provided numerous comments to the previous DTR in JTC 1 N6483 (SC22/WG20 N857). We believe many of these objections were inadequately dealt with in the Disposition of Comments (SC22/WG20 N892).

Details follow on all these objections.

The executive summary of comments is expected to be elaborated in the technical comments, so there is no disposition of the executive summary comments.

DETAILED U.S. NATIONAL BODY TECHNICAL OBJECTIONS TO DTR 14652

Following are detailed technical objections. The U.S. also notes a considerable number of smaller technical issues and editorial problems in the text, but we are not enumerating them here. Rather, we are focussing on the more serious technical problems in the document.

TECHNICAL #1

Problem:

The designation of some sections and subsections of this DTR as "Controversial" is not prominent enough. Members of WG20 have been unable to reach agreement on several important sections of this DTR, and those problems should be acknowledged prominently. The sections/subsections are:

* In LC_CTYPE, the keywords "class," "width," and "map."

- * The entire LC_MONETARY section
- * The entire LC_TIME section
- * The entire LC_XLITERATE section
- * The entire REPERTOIREMAP section
- * The entire CONFORMANCE section

Action:

Add a section to the Introduction of this DTR that prominently lists and describes the controversial sections. Potential implementers need to be aware that there is no consensus for much of this functionality.

Accepted. Text will be added after line 134.

TECHNICAL #2

Problem:

The repertoire of this TR is at least four years out-of-date. According to lines 181-184, the DTR uses:

"ISO/IEC 10646-1:1993, . . . including Cor.1 and AMD 1-9 plus AMD 18. From AMD 18 only the characters U20AC EURO SIGN and UFFFC OBJECT REPLACEMENT CHARACTER are accounted for in this TR." Besides the fact that it is quite unusual to pick only certain amendments, rather than those up to a certain point-in-time, this is ISO/IEC 10646 as it was in 1998 or 1999 (same as Unicode V2.1). Over 55,000 characters have been added to ISO/IEC 10646 since that time. This DTR should match the existing repertoire, not one from four years ago.

Note also that lines 1014-1015 in the LC_CTYPE category differ from lines 181-184 ("The following is the ISO/IEC TR 14652 i18n fdcc-set LC_CTYPE category. It covers ISO/IEC 10646-1 including Cor. 1 and AMD 1 thru 9..."). There is no mention here of AMD 18.

Action:

Update the i18n fdcc-set and the repertoire to use the characters defined in ISO/IEC 10646-1:2000 and ISO/IEC 10646-2:2001. Update the references at

lines 181-184 and lines 1014-1015 to reflect the changes.

Noted. see response 2 to Japan.

TECHNICAL #3

Problem:

The definition of the classes "combining" and "combining_level3", as well as the membership of those classes in the FDCC-set "i18n" differs from what ISO/IEC 10646 defines, and thus violates that standard.

In Section 4.3.1, lines 935-946, the "class" class is defined as:

"Define characters to be classified in the class with the name given in the first operand, which is a string.. . The following two names are recognized:

combining Characters to form composite graphic symbols, such as characters listed in ISO/IEC 10646:1993 annex B.1.

combining_level3 Characters to form composite graphic symbols, that may also be represented by other characters, such as characters listed in ISO/IEC 10646-1:1993 annex B.2."

Further, the "i18n" FDCC-set includes these explanations at lines 1738-1739 and 1761-1762:

```
"% The "combining" class reflects ISO/IEC 10646-1 annex B.1
```

```
% That is, all combining characters (level 2+3).
```

```
% The "combining_level3" class reflects ISO/IEC 10646-1 annex B.2
```

```
% That is, combining characters of level 3."
```

These definitions do not match ISO/IEC 10646. It defines these three levels:

Level 1 -- most restrictive; shall not contain any characters listed in Annex B.1

Level 2 -- less restrictive; shall not contain any characters listed in Annex B.2

Level 3 -- least restrictive; can contain any coded character.

Therefore, what currently is listed as "combining" actually matches a Level 1 implementation, and what is listed as "combining_level3" actually matches a Level 2 implementation as defined in ISO/IEC 10646.

Action:

Revise the text at lines 935-946 as follows:

"combining Define characters to be classified as combining characters
 for ISO/IEC 10646 Implementation Levels. The name of the
 level is given in the first operand. This keyword is optional.
 The following two level names are recognized:

level1 Combining characters prohibited from an
 Implementation Level 1 of ISO/IEC 10646 (see Annex B.1).

level2 Combining character prohibited from an
 Implementation Level 2 of ISO/IEC 10646 (see Annex B.2)."

Further, revise the text at lines 1738-1768 as follows:

```
combining     "level1" /
% Text in an Implementation Level 1 shall not contain any of these characters
% For the "i18n" locale/FDCC-set, Annex B.1 of ISO/IEC 10646 contains
% the full list. To avoid transcription mistakes, the data should be
% derived from 10646 rather than copied here. Following are the characters
% that are part of this class, but they are for information only.
%
%<U0300>..
```

```

%<U0ACB>..<U0ACD>;<U0B01>..<U0B03>;<U0B3C>;<U0B3E>..<U0B43>;<U0B47>;<U0B48>;
/
%<U0B4B>..<U0B4D>;<U0B56>;<U0B57>;<U0B82>;<U0B83>;<U0BBE>..<U0BC2>;/
%<U0BC6>..<U0BC8>;<U0BCA>..<U0BCD>;<U0BD7>;<U0C01>..<U0C03>;<U0C3E>..<U0C44
>;/
%<U0C46>..<U0C48>;<U0C4A>..<U0C4D>;<U0C55>;<U0C56>;<U0C82>;<U0C83>;/
%<U0CBE>..<U0CC4>;<U0CC6>..<U0CC8>;<U0CCA>..<U0CCD>;<U0CD5>;<U0CD6>;/
%<U0D02>;<U0D03>;<U0D3E>..<U0D43>;<U0D46>..<U0D48>;<U0D4A>..<U0D4D>;<U0D57>;
/
%<U0E31>;<U0E34>..<U0E3A>;<U0E47>..<U0E4E>;<U0EB1>;<U0EB4>..<U0EB9>;/
%<U0EBB>;<U0EBC>;<U0EC8>..<U0ECD>;<U0F18>;<U0F19>;<U0F35>;<U0F37>;<U0F39>;/
%<U0F3E>;<U0F3F>;<U0F71>..<U0F84>;<U0F86>..<U0F87>;<U0F90>..<U0F95>;/
%<U0F97>;<U0F99>..<U0FAD>;<U0FB1>..<U0FB7>;<U0FB9>;<U302A>..<U302F>;/
%<U3099>;<U309A>;<UFB1E>
%
%combining    "level2" /
% Text in an Implementation Level 2 shall not contain any of these characters
% For the "i18n" locale/FDCC-set, Annex B.2 of ISO/IEC 10646 contains
% the full list. To avoid transcription mistakes, the data should be
% derived from 10646 rather than copied here. Following are the characters
% that are part of this class, but they are for information only.
%<U0300>..<U0345>;<U0360>;<U0361>;<U1100>..<U11FF>;/
%<U20D0>..<U20E1>;<UFE20>..<UFE23>;/
%<U0483>..<U0486>;<U0591>..<U05A1>;<U05A3>..<U05AF>;<U05C4>;/
%<U093C>;<U0953>;<U0954>;<U09BC>;<U09D7>;<U0A3C>;/
%<U0A70>;<U0A71>;<U0ABC>;<U0B3C>;<U0B56>;<U0B57>;<U0BD7>;<U0C55>;<U0C56>;/
%<U0CD5>;<U0CD6>;<U0D57>;<U0F39>;<U302A>..<U302F>;<U3099>;<U309A>

```

Not accepted. The definition of combining characters is the same as in 10646.

TECHNICAL #4

Problem:

In the previous DTR, the U.S. objected to the fact that character width is specified in two places -- in LC_CTYPE (lines 950-958), and in the charmap (lines 3670-3700). The editor's response was "The reason for a

mechanism to override the default, is that in many cases the default would suffice, while there are a some exceptions from this rule. It is thus efficient to have a place to specify a default, and places to specify exceptions." Since the description in LC_CTYPE states "...A width for a character may be overridden by a WIDTH specification in a charmap...", it appears the width keyword in LC_CTYPE describes default behavior, and that WIDTH in a charmap is for the exceptions.

Having the same thing defined in two places is bad design, and is particularly unnecessary in this case. Display width for characters in monospaced fonts is consistent; it does not differ from locale to locale or locale to charmap. There is some use in having a complete table of display widths, but the information is consistent across locales and therefore does not need to be included in an FDCC-set. For example, Han ideographs have a display width of 2 regardless of whether they are in an English, Japanese, Arabic, or Danish FDCC-set.

Action:

Remove the width keyword at lines 950-958, and also the entries in the "i18n" FDCC-set at lines 1770-1776.

not accepted. The relevant section is marked as controversial.

TECHNICAL #5

Problem:

The Japanese fullwidth ASCII and halfwidth kana characters (defined in the range <UFF01>..<>UFFEE>) are not included in the "alpha" class, or in "i18nrep."

Action:

Add the fullwidth and halfwidth characters to "alpha", and add to "i18nrep," if the full repertoire is to be defined (see TECHNICAL #19).

Not accepted. The intention of the the "alfa" class is to reflect TR 10176 annex A. Text to that effect will be added.

TECHNICAL #6

Problem:

The wrong ISO/IEC 10646 class names are used in several LC_CTYPE categories for Georgian characters. Also, there is contradictory information about the script.

At lines 1068-1069 in class "upper," there is:

```
"% COLLECTION 28 GEORGIAN EXTENDED/  
<U10A0>..
```

At lines 1092-1093 at the end of class "upper," there is:

```
"% COLLECTION 28 GEORGIAN EXTENDED is not addressed as the letters does not  
%   have a uppercase/lowercase relation"
```

And at lines 1144-1145 in class "lower", there is:

```
"% COLLECTION 28 GEORGIAN EXTENDED/  
<U10D0>..
```

It's not clear whether the comment at lines 1092-1093 applies to information in class "upper" or class "lower," but since Georgian characters are listed in both, either the comment is wrong (because those characters are addressed), or membership in one or both classes is not intended and should be removed.

Also the collection name listed in class "lower" (lines 1144-1145) is wrong. This actually is the range for Collection 27 (Basic Georgian). The range for Collection 28 (Georgian Extended) is <U10A0>..

In a related problem, at lines 1263-1264 in the "alpha" class, the incorrect definition is:

```
% COLLECTION 28 GEORGIAN EXTENDED/  
<U10A0>..
```

Actions:

Remove the comment at lines 1092-1093. Georgian *is* addressed in both "upper" and "lower."

Correct line 1144 in class "lower" as follows:

```
% COLLECTION 27 BASIC GEORGIAN/
```

Correct the information in class "alpha" as follows:

```
% COLLECTION 28 GEORGIAN EXTENDED/  
<U10A0>..  
% COLLECTION 27 BASIC GEORGIAN/  
<U10D0>..  
<U10F6>;/
```

Also, add Georgian characters to "i18nrep," if the full repertoire is to be defined (see TECHNICAL #19).

Accepted. Correct script names, comments removed. No ids added to i18nrep.

TECHNICAL #7

Problem:

Some character collections are incorrectly identified in the "alpha" and "digit" classes in LC_CTYPE. They are:

(line 1258, line 1309) TIBETAN Amendment 6

(line 1273) HANGUL amendment 5

(line 1311) FULLWIDTH

Action:

Fix the references to use the correct ISO/IEC 10646 character collections as follows:

(line 1258, line 1309) COLLECTION 72 BASIC TIBETAN

(line 1273) COLLECTION 71 HANGUL SYLLABLES

(line 1311) COLLECTION 69 HALFWIDTH AND FULLWIDTH FORMS

accepted

TECHNICAL #8

Problem:

The specification of the "i18n" LC_COLLATE category in clause 4.4.15 (lines 2330-2366) is syntactically incorrect. The specification:

```
"LC_COLLATE
...

order_start forward;forward;forward;forward,position

% Copy the template from ISO/IEC 14651
copy "ISO14651_2000_TABLE1.txt"

order_end

END LC_COLLATE"
```

is incorrect, for the following reasons:

1. ISO14651_2000_TABLE1.txt already contains a correctly specified "order_end" entry.
2. The "order_start" entry is out of place.

The *correct* way to do this is specified in ISO/IEC 14651, Annex B, where the minimal tailoring is specified as:

```
reorder-after <SFFFF>
order_start forward;forward;forward;forward
reorder-end
```

Action:

The "i18n" LC_COLLATE category in DTR 14652 should be specified as:

```
"LC_COLLATE
```

...

% Copy the template from ISO/IEC 14651

copy "ISO14651_2000_TABLE1.txt"

reorder-after <SFFFF>

order_start forward;forward;forward;forward,position

reorder-end

END LC_COLLATE"

accepted

TECHNICAL #9

Problem:

There are three errors in the symbol equivalences listed in the "i18n" LC_COLLATE category (lines 2340-2357). They are:

1. symbol_equivalence <NONE> <BLANK>

There is no "<BLANK>" symbol in ISO/IEC 14651. This may be a mistake for the intended equivalence to <BASE>.

2. symbol-equivalence <CAPITAL-SMALL> <COMPATCAP>

3. symbol-equivalence <SMALL-CAPITAL> <COMPAT>

These equivalences make no sense. They do not match the tertiary weight symbols <COMPATCAP> and <COMPAT> used in ISO/IEC 14651 in any meaningful way. Actual small capital letters from 10646 have a <MIN> tertiary weight.

If these symbol equivalences are intended to deal with legacy POSIX handling of mixed case digraphs, they will cause havoc in the tertiary weighting of 14651 if applied as equivalences like this indiscriminately to all the other instances of <COMPATCAP> and <COMPAT> that are not part of multiple weightings of mixed case digraphs in 14651.

Action:

Change the <BLANK> symbol name at line 2341 to <BASE>, if that is what is

intended, or to another correct, existing name from ISO/IEC 14651.

Correct the errors in equivalences for <CAPITAL-SMALL> and <SMALL-CAPITAL>.

accepted

TECHNICAL #10

Problem:

There are additional errors in the controversial LC_MONETARY section beyond those reported in previous U.S. comments. At lines 2418-2419, the keyword `mon_decimal_point` is defined as: "The operand is a string containing the symbol that is used as the decimal delimiter in monetary formatted quantities." However, this section attempts to add support for dual currencies, and other keywords are defined as allowing multiple currencies (e.g., `currency_symbol`, `int_curr_symbol`, etc.).

If an FDCC-set includes multiple values in `currency_symbol`, those currencies may have differing conventions for the monetary decimal point. Consider Italian lira and euros. The former does not use a decimal delimiter because there is no such thing as less than one lira, but the euro does use a decimal delimiter.

With this inconsistent definition, there is no way to handle multiple conventions for multiple currencies.

Action:

The support for multiple currencies is badly designed and inadequate for European needs. Take the actions described in TECHNICALS #16, 18 and 20 of the U.S. National Body's comments on the previous version of this DTR (JTC 1 N6483 = SC22/WG20 N857).

not accepted. The committee has no consensus on redesigning LC_MONETARY. The section is marked as controversial.

TECHNICAL #11

Problem:

In the controversial LC_TIME section, the U.S. still strongly objects to the change in the keywords "abday" and "day" (lines 2665-2680) to make the first day of the week be changeable. POSIX.2 defines these keywords in terms of Sunday being the first day of the week, and there are format descriptors for those who use a Monday-first week. This is not an upward compatible change; it will break existing applications.

Action:

Revise the text of "abday" as follows:

. . . The first string is the abbreviated name of the day corresponding to Sunday, the second the abbreviated name of the day corresponding to Monday, and so on. . ."

Revise the text of "day" as follows:

". . . The first string is the full name of the day corresponding to the Sunday, the second the full name of the day corresponding to Monday, and so on. . ."

not accepted. The committee has no consensus on redesigning it.
The section is marked as controversial.

TECHNICAL #12

Problem:

The U.S. still strongly objects to the inclusion of the "timezone" keyword in the controversial LC_TIME section. This functionality already exists via the TZ (timezone) environment variable, and is completely inappropriate within a locale or FDCC-set. For countries that span multiple time zones, there is no way to indicate which zone to use in what area.

Action:

Remove lines 2792-2886.

not accepted. The committee has no consensus on redesigning it.
The section is marked as controversial.

TECHNICAL #13

Problem:

The U.S. still strongly objects to the inadequate, incomplete, and

confusing LC_XLITERATE section. See TECHNICAL #32 from the previous DTR comments in document JTC 1 N6483 (SC22/WG20 N857) for details.

Action:

Remove lines 3059-3173.

Not accepted. See response 3 to japan.

TECHNICAL #14

Problem:

Keywords lang_name, lang_ab2, lang_ab3_term, and lang_ab3_lib in LC_ADDRESS (lines 3261-3273) define natural languages and abbreviations. These have no direct tie on LC_ADDRESS, and the values are not used by any of the LC_ADDRESS format descriptors.

Language information may be useful for an FDCC-set, but not within the LC_ADDRESS section. Such information might be more valuable in the LC_IDENTIFICATION section.

Action:

Remove lines 3261-3273. Consider adding them to LC_IDENTIFICATION.

Not accepted. The LC_ADDRESS category has been implemented as in the TR and the committee has no consensus to change this design.

TECHNICAL #15

Problem:

The new %n format descriptor in LC_ADDRESS (line 3281) is defined as "Person's name, possibly constructed with LC_NAME." This descriptor was added in response to previous U.S. objections to the lack of any explicit way to identify the addressee in an LC_ADDRESS format. While we are glad that the need for identifying the addressee is recognized, the new descriptor does not explain how it can be "constructed with LC_NAME". That category does not have an %n descriptor. As LC_NAME shows, individual names can include many variations, so how, for example, how does one specify such

addressees as:

Joan Smith

Herr Dieter Klein

Dr. Jessica W. O'Brien, Esq.

etc.

using the %n descriptor?

Action:

Add text explaining how to include an addressee within LC_ADDRESS.

15. accepted. The text will be clarified to say that %n can use the result from applying LC_NAME information.

TECHNICAL #15a

Problem:

When an LC_ADDRESS field is not present, the only mechanism for dealing with that is (lines 3288-3291):

"- %N Insert an <end-of-line> if the previous descriptor's value was not an empty string; otherwise ignore.

- %t Insert a <space> if the previous descriptor's value was not an empty string; otherwise ignore."

This is inadequate. There are a number of circumstances where punctuation and other characters between two fields should be deleted if either of them is empty. Take "John Smith, Esq.; Mail-Stop 3; AT&T....". If the title and mailstop are empty, one doesn't want: "John Smith,;; AT&T....".

Action:

Provide a mechanism that allows the removal of a string, containing any sequence of characters, under different conditions (including that either of the adjacent fields is empty). User-test this formulation by investigating what is used by companies to formulate address fields in practice, to ensure that it actually covers the variety of addresses

used around the world.

Not accepted. The LC_ADDRESS category has been implemented as in the TR and the committee has no consensus to change this design.

TECHNICAL #16

Problem:

The description of the %l format descriptor in LC_TELEPHONE is defined as "local number (within area code)" at line 3397. This still does not specify whether it can include digits only (e.g., 5551212) or formatted numbers (e.g., 555-1212 or 12-34-56). The response to the U.S.'s previous objection about this states "The strings are not meant to be restricted to digits", but that information is not in the text itself.

The most useful capability for formatting telephone numbers would be the ability to take a series of digits as typed in by the user, and display those digits with the appropriate format for a given locale. E.g. "12345678901" => "+1 (234) 567-8901". While it is recognized that this is not a simple task, given the variety of different conventions around the world, the limitations of the current descriptors are severe. Nobody wants to split up telephone numbers into 4 database fields, for example, merely to have the above formatting; it is a lot less costly simply to store a formatted string. And if the same digits were used for a number in a different country, the digits might need to be allocated to different fields.

Action:

Revise the format descriptors in lines 3395-3398 to accommodate the full telephone number, and to explain the formatting implications of these values.

Not accepted. The LC_TELEPHONE category has been implemented as in the TR and the committee has no consensus to change this design.

TECHNICAL #17

Problem:

The description of the <repertoiremap> keyword in the Charmap section (lines 3468-3471) is incorrect. It states:

"<repertoiremap> The name of the repertoiremap used to define the symbolic character names in the charmap. The characters of the name are taken from the set of characters with visible glyphs defined in Table 1."

There is no "Table 1" in the DTR. Also, the second sentence "The characters of the name..." probably intends to say "The names of the characters..."

Action:

Fix the faulty second sentence, and also add the information from the non-existent Table 1 into this section.

Partially accepted. Table 1 is defined in 3.2.3, added as "defined in 3.2.3". The "characters of the names" is correct.

TECHNICAL #18

Problem:

More incorrect references to the phantom Table 1. Lines 3517-3525 in the Charmap description state:

"In the first syntax, the line of the character set mapping definition starts with the symbolic name, immediately preceded by a <less-than> character and immediately followed by a <greater-than> character. Symbolic names only contain characters from the set shown with a visible glyph in Table 1.

The same symbolic name may occur several times, with different values. The first value is the one used when generating an encoding, while the other values are accepted in decoding. Symbolic names may be included to identify values that can overlap with each other or with the values of the symbolic names shown in Table 1. . ."

Action:

Add the information that is supposed to be available in the currently non-existent Table 1.

Partially accepted. See 17.

TECHNICAL #19

Problem:

The 27-page repertoiremap "i18nrep" in Section 6 includes entries for about 2,300 out of the 38,000+ characters in the 1998 ISO/IEC 10646 repertoire. All of the following characters are in various sections of LC_CTYPE in the FDCC-set "i18n", but are not in i18nrep:

- * the euro <U20AC>
- * Cyrillic characters in the range <U0492>..<>U04F9>
- * Armenian characters in the range <u0531>..<>U0587>
- * Devanagari characters in the range <U0901>..<>U0963>
- * Georgian characters in the range <U10A0>..<>U10F6>
- * many others. . .

The repertoiremap is defined in lines 3707-3709 as "...the repertoire of characters defined for a FDCC-set, and the symbolic character names and corresponding abstract character (by a reference to ISO/IEC 10646)."

Lines 3729-3731 do specify predefined symbolic names for repertoiremaps. ("The set of <U0000>..<>UFFFF> and <U00000000>..<>U7FFFFFFF> symbolic names... are predefined and refer to the corresponding code points of ISO/IEC 10646 with the same short identifier.") The DTR is silent on whether predefined symbolic names that are not then listed in a repertoiremap form part of the repertoire.

One might assume a repertoiremap provides a set of additional symbolic names and does not need to contain the entire repertoire. However, the "i18nrep" repertoiremap consumes 27 pages in the DTR, implying that it is a complete list of the repertoire. But, as noted, it actually includes less than 10% of the characters used within the FDCC-set "i18n."

Action:

Add wording that explains whether names (including the predefined <U0000>..<>UFFFF> ones) must appear in a repertoiremap for characters to be

considered part of the active repertoire. Then take one of the following actions:

1. If the predefined names must appear, "i18nrep" (lines 3747-6066) must be expanded to include the complete list of characters used in this repertoire. It is *not* acceptable to add one line stating that <U0000>..<UFFFF> are part of the repertoire. The DTR states that it is adhering to a specific version of ISO/IEC 10646, and characters are not assigned to all entries in that range.

If this action is adopted, "i18nrep" should be moved to an appendix.

2. If the predefined names do not have to appear, then the repertoiremap simply is an example showing how alternate names can be defined. There is no need to list 2,318 example names while omitting the remaining 35,000+ other characters. In that case, reduce "i18nrep" to a one-or-two-page example. Also, add information explaining how to determine which of the predefined symbolic names are part of a given repertoire (e.g., <U0000>..<U007F> is included, but <U0600>..<U060B> is not [because no characters are assigned in the latter range]).

Not accepted. The i18nrep is not intended to be complete. Words will be added to that effect. See also response 4 to Germany.

TECHNICAL #20

Problem:

The discussion of the "i18nrep" repertoiremap in Section 6 makes reference to "the 'i18n' FDCC-set" (line 3744). However, nowhere in DTR2 14652 is "the 'i18n' FDCC-set" actually defined. All the relevant categories are defined:

Section 4.2, "the 'i18n' LC_IDENTIFICATION category"

Section 4.3.2, "'i18n' LC_CTYPE category"

Section 4.4.15, "'i18n' LC_COLLATE category"

and so on. But where is the actually *FDCC-set* definition for "i18n" that would include the crucial specification of whether the "i18nrep" repertoiremap is actually part of that FDCC-set or not? We only see a full

attempt in B.1.3.3, the "Sample FDCC-set specification for Danish", which includes the "i18nrep" repertoiremap and the "ISO_8859-1:1987" charmap. This is just quietly skipped over for the "i18n" FDCC-set itself in the main text. Without any indication of a repertoiremap or charmap, how are the symbols in the "i18n" categories to be resolved?

This is another hole in the specification.

Action:

Add the full "i18n" FDCC-set specification to the DTR. This could be in an appendix.

accepted. Text will be added to define "i18n" as "i18n" specifications in the various categories, but not as a textual copy of these specifications.