ISO/IEC JTC 1/SC22
Programming languages, their environments and system software interfaces
Secretariat: U.S.A. (ANSI)

ISO/IEC JTC 1/SC22 N3025

TITLE:
Summary of Voting on Third FCD Ballot for FCD 14651: Information technology - International String Ordering and Comparison - Method for Comparing Character Strings and Description of a Common Tailorable Ordering Template

DATE ASSIGNED:
1999-10-25

SOURCE:
Secretariat, ISO/IEC JTC 1/SC22

BACKWARD POINTER:
N/A

DOCUMENT TYPE:
Summary of Voting

PROJECT NUMBER:
JTC 1.22.30.02.02

STATUS:
WG20 is requested to prepare a Disposition of Comments Report and make a recommendation on the further processing of the FCD.

ACTION IDENTIFIER:
FYI to SC22 Member Bodies
ACT to WG20

DUE DATE:
N/A

DISTRIBUTION:
Text

CROSS REFERENCE:
N2933

DISTRIBUTION FORM:
Def

Address reply to:
ISO/IEC JTC 1/SC22 Secretariat
William C. Rinehuls
8457 Rushing Creek Court
Springfield, VA 22153 USA
Telephone: +1 (703) 912-9680
Fax: +1 (703) 912-2973
email: rinehuls@radix.net

SUMMARY OF VOTING ON

Letter Ballot Reference No:	SC22 N2933
Circulated by:	JTC 1/SC22
Circulation Date:	1999-06-16
Closing Date:	1999-10-18

SUBJECT: Third FCD Ballot for FCD 14651: Information technology - International String Ordering and Comparison - Method for Comparing Character Strings and Description of a Common Tailorable Ordering Template

---

The following responses have been received on the subject of approval:

"P" Members supporting approval without comment	9
"P" Members supporting approval with comment	2
"P" Members not supporting approval	4
"P" Members abstaining	0
"P" Members not voting	6
"O" Members supporting approval without comment	2
"O" Members not supporting approval	1

Secretariat Action:

The comments accompanying the affirmative votes from Germany and the United Kingdom are attached along with the comments accompanying the negative votes from France, Japan, the Netherlands and the United States of America.

WG20 is requested to prepare a Disposition of Comments Report and make a recommendation on the further processing of the FCD.

ISO/IEC JTC1/SC22 LETTER BALLOT SUMMARY

PROJECT NO: JTC 1.22.30.02.02

SUBJECT: Third FCD Ballot for FCD 14651: Information technology - International String Ordering and Comparison - Method for Comparing Character Strings and Description of a Common Tailorable Ordering Template

Reference Document No: N2933 Ballot Document No: N2933
Circulation Date: 1999-06-16 Closing Date: 1999-10-18

Circulated To: SC22 P, O, L Circulated By: Secretariat

SUMMARY OF VOTING AND COMMENTS RECEIVED

	Approve	Disapprove	Abstain	Comments	Not Voting
'P' Members					
Austria	( )	( )	( )	( )	(X)
Belgium	( )	( )	( )	( )	(X)
Brazil	( )	( )	( )	( )	(X)
Canada	(X)	( )	( )	( )	( )
China	(X)	( )	( )	( )	( )
Czech Republic	(X)	( )	( )	( )	( )
Denmark	(X)	( )	( )	( )	( )
Egypt	(X)	( )	( )	( )	( )
Finland	(X)	( )	( )	( )	( )
France	( )	(X)	( )	(X)	( )
Germany	(X)	( )	( )	(X)	( )
Ireland	(X)	( )	( )	( )	( )
Japan	( )	(X)	( )	(X)	( )
Netherlands	( )	(X)	( )	(X)	( )
Norway	(X)	( )	( )	( )	( )
Romania	( )	( )	( )	( )	(X)
Russian Federation	(X)	( )	( )	( )	( )
Slovenia	( )	( )	( )	( )	(X)
UK	(X)	( )	( )	(X)	( )
Ukraine	( )	( )	( )	( )	(X)
USA	( )	(X)	( )	(X)	( )
'O' Members Voting					
Australia	(X)	( )	( )	( )	( )
Korea Republic	(X)	( )	( )	( )	( )
Sweden	( )	(X)	( )	(X)	( )

___ end of detail summary; beginning of France comments __________

1. French vote on Third FCD Ballot for FCD 14651

SOURCE: AFNOR

AFNOR votes NO on Third FCD Ballot for FCD 14651.

Its vote will be reversed to YES if the following comments are satisfactorily resolved:

1.1. General introduction :

FCD 14651.3 is much more like a draft standard than the previous versions. We would like to thank the work that have been done by the editor and the working group as a whole to achieve this state of affairs. We believe that with a small number of changes in order to make the meaning of the standard clear and unambiguous, this draft can be changed to a useful standard.

The only general point that the French National Body regrets is that in this process, it seems that the French version of ISO/IEC 14651 have been lost. We are sure this is only a matter of lack of time to prepare both versions concurrently, and we would like to see both versions to be presented jointly for the FDIS draft.

1.2. Technical comments (ordered as per the FCD.3 text where possible) :

Organization of the document : it is very hard to find out what a conforming implementation is required to do. The conformance clause (2) is like a box that defers all of its task to clause 6, where the requirements for conformity are interleaved with the explications of behaviour of the reference implementation and the conditions for the various equivalencies.

We believe another organization of the document would be better: keeping in a clause all the explanations of the reference behaviour: this include most of the material present in clause 6 (but obviously with a different title, excluding 6.3.4, 6.3.5, and most of 6.4. Then make a new clause, grouping the content of clause 2, all the material in clause 6 that refers to the conditions of equivalence, and explicitly grouping all the requirements. Proper exposition would make this clause to appear after the clause 6, but strict observance of ISO/IEC rules may require such a clause to appear as soon as the actual clause 2.

1.2.1. Clause 2 (conformance):

As it stands out, the requirements appear too strong: for example, 6.2.1.2 states "These properties [forward, backward, position] can be changed." We do not believe that all implementations are required to allow any combination of the properties. But that is what is required nowadays.

This is part of the reason why we want to see a clear separation between the behaviour of the reference mechanism, and the requirements.

1.2.2. Clause 3 (normative references):

We do not believe that all amendments to ISO/IEC 10646 are normative references. In particular, Am.3 (about the deletion of UTF-1) is a strange reference. Also the inclusion of the euro character (U+20AC) in the table, while not being defined in the eight references that are given, looks like a problem.

1st sentence says "At the time of publication, the editions indicated were valid." This is very likely to be wrong, in our humble opinion.

1.2.3. Clause 4 (definitions):

"order" is not defined but is used
"ordering table" is not defined but is used

"4.7 (collation) level when used without qualification [...]" is misleading

4.11 could be rewritten as "method for ordering two character strings", which is lighter and only use defined terms

4.15 is not clear to us (to say the least)

4.16 have not been reviewed

The use of list vs. sequence vs. series, here and in clause 6, is not systematic, while it should (we are not doing good style, we are specifying things; your mileage may vary).

1.2.4. Clause 5 (symbols and abbreviations)

We fail to see the difference between <Pyyyyyyyy> and the various ranges in the UCS that are reserved for private use characters, like <UEyyy>, <U-000Fyyyy>, <U-0010yyyy>, <U-7yyyyyyy>, etc.

1.2.5. Clause 6 (requirements) is split:

Subclause 6.1 (preparation)

6.1 is no requirement (according to other parts and to 6.1 itself), so it should be moved elsewhere (annex C is a good candidate).

The only possible requirement is for Thai and Lao (the swapping for the leading vowel). Unicode requires it. The status of 14651 on this point is unclear, it should be unambiguous.

2nd paragraph, last sentence (about a further remapping) is strange: we do not see why it may be needed.

Note 1 really belongs to subclause 6.3.3 or 6.4. "Should" in a note is to be avoided.

Subclause 6.2 (key building and comparison)

Logically, this subclause should comes *after* subclause 6.3. We believe this would make the exposition much clearer, in particular by removing a number of forward references (weight, value, the underlying order relation). However, references should be kept adjusted (it appears it has not been the case in the past, this is unfortunate).

'collating-elements' are badly handled though out the whole subclauses 6.2.2 and 6.3; we infer that most of the occurrences of 'characters' and some of 'symbol occurring in symbole_definition' should in fact cover collating-elements, but that should be made clear. The BNF does not even work for them ('simple-line' accepts 'collating-elements-definition', but there are production only for 'collating-elements'.)

6.2.1.1 refers a "tailoring phase" that allows for customization of the number of levels; but such a phase cannot be located in the present draft.

6.2.1.2, 2nd paragraph : this paragraph should reference the ',position' notation for proper understanding.

Also, there is a seemly contradiction between the allowed multiple occurrences of order_start, and this sentences which states in effect that each occurrence should fix the same property for a given level throughout the table.

6.2.2, 2nd paragraph first sentence either contradicts or reformulates the definition in 4.9.

3rd paragraph, 1st sentence effectively defines "undefined". So "undefined" should be written using italics.

1st sentence of 6.2.3 is unreadable to me, but I am not familiar enough with English Mathematics jargon to say if it is correct or not. As an example, we know from external sources that incomplete comparisons (for example, where m is less than the number of levels present in the weight_table) are to be allowed, but that does not show up clearly here. Also, can someone define (i-1) when i is 0 or 1?

Subclause 6.3 (common template table: formation and interpretation)

In the BNF, the production for 'symbol_definition' should allow for "space+" between 'collating-symbol' and 'symbol_element'.

Also, 'line_completion' should be rewritten as line_completion = space* comment? EOL to allow for trailing blanks in conforming inputs.

'level_token' could be replaced by 'weight', which would decrease by one the large number of specific terms this Standard introduces.

WF1 is just plain unreadable. The intent is clarified by the note, but the words cannot be understood; as it stands, there are serious traps:
 - "shall occur in a symbol_definition in that same symbol_weight" cannot be parsed, because the production for 'symbol_weight' does not allow for

'symbol_definition';
 - what about symbols that are "defined" in other productions, like 'collating-element'?
 - then, "in the same symbol_weight" just leads to the conclusion that 'symbol_definition' is an error that is to be replaced by 'symbol_element';

this can be confirmed from the possibility of a 'level_token' to be "defined" by its presence (as a 'symbol_element') in a previous 'symbol_weight'; but the purpose of that construction is unclear (and probably wrong, since it cannot be figured how rule I8 will assign values to these symbols)

WF2 makes a forward reference to 'value' which is defined in rule I7 (something that is not very welcome: it took me more than 5 hours to understand that), but the rule I7 does not allow for a possibility for identical values; so WF2 appears as a no-op. If the intent is what the notes explains, it may be easier to specify that a given symbol should not appear twice as a 'symbol_*element*' (rewriting that to take care of ranges).

  Nothing seems to prevent (after handling of reorders)

```
      <U00C0> some_weights
      % ...
      <U-000000C0> some_other_weights
```

but that may be an artefact of another defect

WF6, WF12 and WF13 should be moved before WF3, because they do apply to both kinds of tables, while WF3 to WF5, and WF7 to WF11 only apply to 'tailored_table's.

WF9 should allow for some 'simple_line's to appear between a 'reorder_after' and the "closing" 'reorder-xxx' line; as it stands, it defeats the purpose.

Part of WF10 is defeated by 6.4 which requires a delta to have at least one 'order_start' line.

In WF12, the term 'value_range' is poorly chosen, since it confuses things, because 'value' is used for another meaning (numeric weights).

Enhance the note by giving the 'value_range' that correspond: 20901 (or 51A5)

Add to the note: "Common prefix cannot contain any character that may be interpreted as a hex-upper: thus <Def0012>..<Def0044> is prohibited."

Definition of any 'simple_symbol' beginning with U should be prohibited, to avoid asking for trouble (and also to allow further extension).

6.3.3 interpretation of tailored tables implies the inclusion of a 'common_template_table' before processing of the 'tailored_table'; it should be said somewhere.

The example for I2 (and for I3) is wrong: the expansion should be

```
    collating-symbol <S0301>
    collating-symbol <S302>
    collating-symbol <S0303>
```

This is how I2 reads, and this is how the table in annex A behaves, by the way: it "defines" <S0200>..<S1100>, then makes use of 'symbol's of the form <Sxxx>, with only three digits.

If the intended behaviour is what the example claims, that is with the leading zeroes (and which is what PDTR 14652 claims for conformance), a number of changes are required: the common table should be adjusted, and the text from PDTR 14652 requesting the suffix to be of same length, should be drag in 14651 (somewhere near WF12). Then, I2 should be modified to explicitly produce the leading zeroes.

This will have the useful property of handling <Uxxxx> symbols nicely (it would be as if lines like

```
    collating-symbol <U0000>..<UFFFF>
    collating-symbol <U-00000000>..<U-FFFFFFFF>
```

appears before the 'common_template_table', with an additional rule meaning that corresponding 'ucs_symbol' should have the same 'value').

There is a missing rule to allow what the second note to I4 explains: that multiple 'tailoring_lines' are to be handled in sequential order.

No rules allows for a way to interpret 'tailored_table' that have more or less than four levels, while this is not prohibited otherwise.
We would expect a way to "map" the four levels of the common template to the levels used in a 'tailored_table', but there is nothing like that. Surely something is missing here (if the intent is that all tailoring tables should have four levels, a bunch of text can be dropped from WF3, WF4, WF5, etc. On the other hand, if the intent is to allow both tables to have a different number of levels, then the rules for equivalence should be deeply enhanced, since it is demonstrated [LaBonté, cited in Annex F] that it is not possible to achieve the same results as the common template table with less than four levels).

The "in general" in the note about I6 asks for trouble: is the committee aware of cases where it cannot be done? and in such cases, what is the reference behaviour? should such cases been disallowed? where is it done? If this is not the case, drop the words.

Rule I7 and I8 should be moved to a new subclause (named "evaluation"?) to highlight the difference between the interpretation of the table, and the process to transform the tables into the input for the process described in subclause 6.2.

Rule I7 effectively defines 'value's which are used in a number of other places, and in particular in 6.2.3. This should be made much more prominent, perhaps as a definition.

The handling of ranges in I7 could be made explicit.
The part enclosed in parenthesis in I7 is troublesome: either it is a paraphrase of the preceding sentence, and using a note might be a good idea; or it adds something new (we fail to see what: in particular, we do not believe that using line numbers is a requirement; but we can be wrong), and a rewriting might be a good idea; worse, it looks like it does not handle ranges as nicely as the previous sentence...

Major problem, we request a way to evaluate in I8 ucs_symbols intermixed with simple_symbols. As it stands, ucs_symbols have no value associated to them. So the reference comparison in 6.2.3 cannot work for them; alas, they are used (on level 4) in the common template table...

Also, as in other places, the injection defined in I8 does not allow for handling of collating_elements.

As we understand things (but that is deeply under-specified), m (the number of subkeys in 6.2.3) is a parameter to the equivalence relationship to be used in 6.3.4 to compare weight_table: thus it allows to have a weight_table that is equivalent to the common template (suitably tailored) when only there levels are examined, but may be different at the fourth level, because for example ',position' are not handled, or a new level is inserted here.

If we understand right, we believe the standard would be improved if this is made clear.

1st paragraph of 6.3.5 fails to request that for a implementation to be conformant, it should be equivalent to the common template table. As it stands, almost any implementation can be made conforming, since the template_table is not indicated, so any set of simple_lines can be chosen.

Also the words are poorer that the ones that are used just one subclause above (any comparison .... results in the same ordering).

What is the repertoire R which is to be used for conformance? if it is a parameter of the conformance specification, it is worth mentioning it.

2nd paragraph of 6.3.5 speaks about equivalence between a weight-table and a tailoring; but the equivalence is not defined, except by the (normative) sentence in 6.4 which says that "tailoring may be accomplished using any syntax that is equivalent to the one described in this International Standard"; the result of this is that 6.3.5 is a (partial) rewriting of 6.4. If we did not miss anything else, we suggest dropping this paragraph.

If the reorganization proposed is done, the whole text of 6.4 should be kept with the conformance part, away from the explanations of the behaviour of the reference method.

6.4 should be split in two parts: one that describe what is a delta using the reference methods and syntax (that is, the requirement to be based on the common template table and the 1st and 2nd requirements); the other that groups all the "equivalence" clauses, suitably reworded.

If the reorganisation proposed is done, the first part should be kept with the rest of clause 6, the explanations of the behaviour of the reference method, while the second should be grouped with the conformance part.

Equivalence in general sense does require both ways of implications, meaning that one should be able to demonstrate that one can pass from the implementation to the reference method *and*back* with the same results.

We believe that the intent of this standard is stricter, and that only one way is requested (namely the second): for example, an implementation may provide different backwards/forwards properties for different scripts, something that is not allowed by the reference method; but this is not a case for non-conformance.

The example at the end (perhaps purposely) avoids to deal with precomposedcharacters and combining characters; this is also elided in the more detailed examples (see below); we believe this is unfortunate.

1.3. Annex A : (common template table)

In the table: there is an obvious problem with Gurmukhi. Constant references on this subject (<URL:http://www.sikhs.org/gurmukhi.htm>) shows as order
ura(u,uu) a(aa) iri(i,ii)
   s h
then the vargs in traditional order (k kh g gh ng ... b bh m)
then
   y r l v
   rra
and nukta consonants follow their sister, this is already OK in the table. The diphthongs (e/ai and o/au) should be ordered among the basic vowels, but I cannot figure what is the rule here. Perhaps Jeroen knows.

1.4. Annex B (tailoring deltas)

B.1 : we believe the real Canadian delta requires additional handling for the correct decompositions (in particular about the ae handling). Further explanations about that would be welcome, since this is not trivial.

B.2 "The repertoire used assumes the exclusion of combining characters": this is unfortunate!
  Later reads "To also make capital letters in compatibility characters sort before lowercase, a slightly more complex tailoring is required". Something is wrong here: either the required is "slightly more complex", and we welcome the editorial committee or the working group to provide this tailoring (perhaps example 3 fits the need). Or the tailoring is really much more complex, and we would like this understatement to be remove from an International Standard, and changed to a sentence explaining what the problem really is.
  This example does not comply with the 1st requirement for a conforming delta (to have at least one order_start entry).

The example in B.3 does not comply either with the 1st requirement for a conforming delta (to have at least one order_start entry).

B.4 does not belong to annex B (this is no example, and it deals extensively with preparation). We would like to see it under annex C instead.

B.5 is neither an example, but we assume this is an artefact.

Annex E should be reworded (a lot) to take into account the newer status of TR 14652. In fact we believe this is easier to drop it completely.
  If it is kept, syntax should be harmonized with the rest of the text (use of "term" instead of 'term', for example; references to other non present parts of previous drafts of PDTR 14652 should also be dropped).

Annex F should at least name UTR10. Unicode itself is another question, but some part of the text seem to make reference to it (particularly the note in 6.1 about combining characters and normalization).

_____ end of France comments

2. German vote and comments on FCD 14651

Hereafter please find the DIN vote on on FCD 14651 with comments.
The DIN vote is YES with comments.

Approval with comments

Comments to 14651

2.1. General

The current draft is once greatly improved over the previous version. Germany congratulates the editor and sees itself in a position to approve the current draft at the FCD stage. Should, however, a number of issues (including the Cyrillic issue) not be resolved prior to the FDIS, Germany may not be able to support the draft at that stage.

Remark on the format
The current pdf-file can only be read with Acrobat Reader 4.0, and it proved impossible to print it on a variety of PostScript printers. It would be desirable if only such pdf-files were distributed that can easily be handled on different systems and printed on different printers. Many people find it very inconvenient to review lengthy drafts on screen.

Major
Annex A and Annex B.5: Cyrillic
The Cyrillic repertoire is to be aligned with that of SC22/WG20/N681 and the delta of B.5 to be used in the Common Template Table itself.

Alternatively, an entirely artificial ordering sequence can be chosen if the following conditions are met:
- this ordering makes tailoring inevitable for applications using the Cyrillic script;
- the Annex B.5 is maintained.

2.2. General and Annex E

As there is not going to be a ISO/IEC 14652, all references to this project and specifically Annex E must be removed.

General
Ordering must not produce different results from encoding differences which are invisible to the end user. E. g., (using Unicode terminology) a precomposed character and its canonically equivalent combining sequence must order identically.

2.3. Annex A

The abbreviations for diacritics and casing should be chosen according to a consistent scheme.

Minor
Introduction:
2nd §:

- some tailoring --> tailoring

Scope
2nd dash:
- "used normatively in this" --> "used normatively within this"

Note 1:
- "may be modified with a minimum of effort" --> "is to be modified"
- "no modification should be required and that the order will remain ..." --> "often no modification may be required."

alternatively, remove note altogether

Dash 11:
- "A context dependent ordering which..." --> "Context dependent ordering."

Definitions
Def. 4.15:
- "length b digit sequence" --> "digit sequence of length b" (or similar)

Def. 4.16:
- "to be completed offline": ???

Requirements
6.2.1.2, last §:
- "arbitrary name": the name is not arbitrary but must be formed following the rules set out by the BNF ("identifier"). As long as it conforms to those rules, it can be freely selected.
Change the formulation accordingly.

6.2.2, 3rd §:
- "tble" --> "table"

6.2.2.3, Level 2:
- "level_2" and "level_3" --> "level 2" and "level 3"

6.3.1:
- "symbol_ element" --> "symbol_element"

6.4, Note:
- "XML" --> "an XML conformant markup scheme" (or equivalent)

Annex A:
General:
The practice of the previous FCD to just reference a URL is much preferable over the current one. If it then is to be reproduced, a Courier font (or, at the very least, some monospaced font) should be chosen.

Note:
- "as well as in addition to be reproduced" --> "in addition to being reproduced" (or equivalent)

2.4. Annex B:

- Print code samples in Courier

Annex B.5, Note, 2nd §:
- Draw attention to the "i kratkoe" for Russian

Annex D:
1st §:
- modify 1st sentence (there are usable "commercial sort programs")

Item v, last §:
- "In Spanish and Nordic languages" --> "In some languages, including ..."

____________ end of Germany comments; beginning of Japan comments ______

3. Japan's vote on FCD 14651.3 (N2933)

SC 22 N 2933: Third FCD Ballot for FCD 14651
Method for Comparing Character Strings and Description of a
Common Tailorable Ordering Template .

(X) Disapproved

National Body: Japan
Date: 1999-10-18
Signature: KATSUHIKO KAKEHI

Comments on FCD 14651.3

The National Body of Japan disapproves FCD 14651.3 for the reasons below.

If the comments are satisfactorily resolved, Japan will change its vote to approval.

3.1. Jp.1) Global, the lack of semantics:

The draft does not describe the indispensable semantics of the table elements, such as "IGNORE", "order-start", "collating-symbol", and "collating-element" (the detail are given afterwards).

There are three alternatives to solve this problem:

Alt.1 do piecemeal improvements to the current text,
Alt.2 systematically import the materials from PDTR 14652 or
from POSIX.2,
Alt.3 add a normative reference to ISO/IEC 9945-2 (POSIX.2)
and add a sentence

> Unless otherwise specified here, the requirements
> for LC_COLLATE in ISO/IEC 9945-2 are applied here

at the beginning of Clause 6.

Japan considers that Alt.1 will make the text much more complicated and it needs to be put back to the CD stage considering the amount of changes.

Japan also considers that the material to be imported in the case of Alt.2 is relatively small but its related changes to keep consistencies between the current text and the imported text, are huge and the draft also needs to be put back to the CD stage.

Meanwhile the decision to remove blockwise ordering direction change has reduced the difference between 14651 and POSIX.2

Therefore Japan strongly recommends Alt.3.

NOTE: the semantics to be added --

a) order_start: Define collation rules. This statement is followed by one or more collation order statements, assigning character collation values and collation weights to collating elements.

b) IGNORE: Collation shall behave as if IGNOREd elements are removed for each weight level, unless the position collation directive is specified for the corresponding level with the order_start keyword.

The special keyword IGNORE as a weight shall indicate that when strings are compared using the weights at the level where IGNORE is specified, the collating element shall be ignored; i.e., as if the string did not contain the collating element.

c) collating_symbol: This keyword (collating_symbol) shall be used to define symbols for use in collation sequence statements; e.g., between the order_start and the order_end keywords.

d) collating_element: A collating-element symbol represents a multicharacter collating element.

3.2. Jp.2) Global, CTT and the tailored table:

Japan believes that the CTT is used as an input to the tailoring process and is not used as an input for the further processing while the tailored table is used only as an input for the further processing and is not used as an input for the tailoring process.

The following text, which does not fit the principle above, should be changed.

a) 1 Scope, bullet 1:

The sentence

> This method uses transformation tables derived either from the Common Template Table defined in this International Standard or from one of its tailorings.

should be changed to

> This method uses transformation tables derived from one of the tailoring of the Common Template Table defined in this International Standard

b) 6.2.1.2 Processing properties:

The text

> a tailored table may be separated into sections for ease of tailoring

is wrong. The paragraph containing this text should be removed.

c) 6.2.2 Key formation:

The text

> where m is the maximum number of levels described in either the Common Template Table or in the tailored collation weighting table

should be changed to

> where m is the maximum number of levels described in the tailored collation weighting table.
>
> NOTE: There is still another type of error in this text as is pointed out afterward (Jp.12).

d) 6.2.2 Key formation

The text

> ... a corresponding symbol prefixed with "U" in the Common Template Table or in the tailored collation weighting table

should be changed to

> ... a corresponding symbol prefixed with "U" in the tailored collation weighting table.

3.3. Jp.3) Global, tailoring capability:

The draft pays little consideration to the kind of tailoring. Many practical cultural adaptations are impossible or very hard to do as follows:

a) adding a new "collating_symbol" is impossible in the formal delta declaration because the target of "reorder_after" seems to be limited to "symbol_weight" from the examples in Annex B.

> NOTE: The interpretation I4 in 6.3.3, which is almost impossible to understand, seems to say the target is "symbol_definition". But in that case,
> changing the "symbol_weight" is impossible.

b) adding an "order_start" is also impossible as described above,

c) swapping the blocks in the CTT is only possible by redescribing the content of all the preceding block in the delta and putting that after the following block. It is nonsense to redescribe the content of CTT without any changes.
For example, if one wants to move only one line upward, he has to redescribe all the lines from the expected position to the current position in the delta and has to reorder it after the current position. It is worth being called almost impossible.

d) let one want to redefine the order for a very small set of characters using five weight levels. In this case, he has to redefine in the delta all the symbol_weight lines in the CTT using the five weight levels, because the number of levels should be the same in the tailored table as is defined in WF3 in 6.3.2. It is worth being called almost impossible.

Solutions to the problems above:

- add some new tailoring lines for case a), b), and c),
- the condition WF3 should be replaced by an explanation

> An empty level_token shall be interpreted as the collating element itself.

in the same way as in POSIX.

> NOTE: This comment is the same as J.15-17 in FCD.2 which was not accepted without ANY rationale.

If the proposal is rejected, the sentence

> This number of levels can be extended or reduced (but not below 3 levels) in the tailoring phase

in 6.2.1.1 should be changed to

> This number of levels can be extended or reduced (but not below 3 levels) in the tailoring phase only if
> all the entries of the CTT are redefined in the delta.

3.4. Jp.4) Global, character definition:

In the case of POSIX, the characters used in LC_COLLATE are prepared in a charmap. But in this standard, there is no facility to declare the characters to be considered -- using "collating_symbol" as is done now is illegal.

A new line "collating_character" should be introduced or a new semantics for "collating_symbol" should be introduced.

> NOTE: This becomes evident by the drastic change of the CTT from FCD.2.

3.5. Jp.5) Global, Assignment of values

In the current specifications, it is not clear where the weights for symbols are defined. If it is defined in "collating_symbol"s, the weights for the characters are defined twice.

The CTT should be globally changed or a new semantics for "collating_symbol" should be introduced.

3.6. Jp.6) Global, section:

All the "section" facilities should be removed because they become no use under the current CTT while they will lay a heavy burden on users of this standard.

The script facilities, which up to FCD.1 played the same role as the section facilities does, made a sense because the CTT was divided into scripts in order to ease script-wise tailoring.

Now there is no section defined in the CTT, the tailoring using the section facilities should be started from inserting "section_definition" and the following lines using "reorder_after" with some "target_symbol. The action is done simply by using "reorder_after".

3.7. Jp.7) p.iv, Introduction, the first sentence:

The sentence

> This International Standard provides a method for ordering text data worldwide, and provides a Common Template Table whose tailoring meets the requirements of a given language and culture while retaining universal properties for other scripts.

should be changed to

> This International Standard provides a method for ordering text data worldwide, and provides a Common Template Table whose tailoring meets the requirements for the scripts used in a culture while retaining cross-cultural friendliness for other scripts. Cross-cultural friendliness, defined in TR 11017:1997, denotes the ease with which unfamiliar culturally-dependent information can be understood by persons who are not familiar with this culture.

because

- two or more languages and scripts may be used in one culture,

- the term *universal properties* suggests the orthodoxy and may invoke some unresolvable fight among the cultures sharing a script.

3.8. Jp.8) p.4, 6.1 Preparation of character strings prior to comparison:

The text in this subclause has been greatly changed from the second CD without being based on any NB comments.

The only one possibility in the disposition document (SC22 WG20 N670) relating to this change is "Text will be reorganized" in 7.1.14.

However, the disposition is the response to Japan's comments requesting to move the subclause out of Clause 6 because of its irrelevance to the subject of Clause 6 and the change is just the opposite to Japan's intent and it contains non-negligible errors as follows:

a) the first paragraph

> It may be necessary to transform character strings before the comparison method is applied to them (see annex C for an example of such preparation). Although not part of the scope of this International Standard, context-sensitive preparation may be an important part of the ordering process, as for example in telephone-book ordering, a complex case in point.

is ambiguous because

1) it says only context-sensitive preparation is not part of the standard -- some may think context free preparation is part of the standard;

2) it is not clear that "the comparison method" used here is the same as "the reference comparison methods" or a part of it.

b) the part of the second paragraph

> Where applicable, it can be an important part of the prehandling phase to map characters from a non-UCS encoding scheme to the UCS for input into the reference comparison method. This task can amongst other things encompass the correct handling of escape sequences in the originating encoding scheme, the mapping of characters without an allocated UCS codepoint to an application-defined codepoint in the private zone area and inverting strings which are not stored in UCS order

is wrong. The part suggests that a non-UCS encoding system is out of this standard because it always needs some prehandling not in a part of this standard. But we should not exclude non-UCS encoding systems.

c) the part of the second paragraph

> For example, visual order Arabic code sets must be put into logical order; bibliographic code sets with accents before base characters require reversal. The resulting string sequence may then have to be remapped into its original encoding scheme

should be removed because the terms "visual order Arabic code sets" and

"Bibliographic code sets", which are defined neither in this standard nor in any normative reference standard, appear suddenly without any explanation.

d) the NOTE 1, which describes the design principle of the CTT and the delta, should be removed because it has no relation with the title of this subclause and the main text.

Considering these problems, the subclause 6.1 should be removed or moved to Annex C.

If a link to Annex C is needed in the main text, Japan proposes to change the subclause as follows:

> 6.1 Input strings
>
> Each character used in the input to the reference comparison method shall have a one-to-one mapping to a character expressed as <Uxxxx> or <Pyyyyyyyy> and listed in the tailored table.
>
> It is not part of the scope of this International Standard how the input strings are prepared from the real application data (see annex C for an example of such preparation).

3.9. Jp.9) p.5-8, 6.2 Key building the comparison:

The beginning of this subclause

> A series of m intermediary subkeys is formed out of a character string, where m ...

should be changed to

> When two strings are compared to determine their relative order, the two strings are first broken up into a series of collating elements taking account of multi-character collating elements defined using "collating_element" statements in a tailored table. Then a series of m intermediary subkeys is formed out of a collating element string, where m ...

in order to get the intended outputs.

3.10. Jp.10) 5 Symbols and abbreviations:

The text

> By convention, if a character outside of the standard repertoire of ISO/IEC 10646 is to be used in tailored ordering tables, it is recommended that this character be identified using the form <Pyyyyyyyy>

sounds queer. If the use of <Pyyyyyyyy> is only a recommendation, it is confusing in the current way of defining characters and symbols both by "collating_symbol".

A new semantics for "collating_symbol" should be introduced or this convention should be changed to "normative" by using the word "shall".

3.11. Jp.11) 6.2.1 Preliminary considerations:

The text

> one of the tailoring possibilities is to assign a given order to each section and to change the relative order of an entire section relative to other sections

should be removed because the proposed possibility makes no sense where no section is defined in the CTT.

3.12. Jp.12) 6.2.2 Key formation:

The text

> where m is the maximum number of levels described in either the Common Template Table or in the tailored collation weighting table

is wrong. Contrary to POSIX.2 where "COLL_WEIGHT_MAX" specifies the maximum number of levels, this standard provides no room for specifying the maximum number of levels -- the number of "direction" in "order_start" should be referred simply as "number of levels".

3.13. Jp.13) 6.3.1, BNF:

The term "collating_element_definition" should be changed to "collating_element".

3.14. JP.14) misc.

A NOTE for removing the syntax like 'collating-element <ll> from "ll" ', which is allowed in POSIX and PDTR 14652 should be given in some place.

3.15. Jp.15) 6.3.2, WF4:

The condition

> A tailored_table may not contain a multiple_level_direction if it does not also contain a weight_list consisting of more than one level_token

is wrong. A tailored table must have a order_start statement which shall have a multiple_level_direction by BNF

```
        order_start = 'order_start' space+ identifier semicolon
                multiple_level_direction (',position')?
                line_completion ;
```

NOTE: A multi_level_direction may have only one direction if all the collating entry identifiers contain a weight_list consisting of only one level_token.

3.16. Jp.16) 6.3.2, WF4 NOTE:

The sentence here

> No order_start statement shall be used in a table which defines no multi-level weights.

does not explain the main text.

3.17. Jp.17) 6.3.2, WF5.

The sentence here

> A multiple_level_direction in a tailored_table shall contain the same number of direction's as the number of level_token's of any weight_list in that tailored_table.

still remains the problem that how to do with the multiple order_start where the number of direction's are equal but the contents differ. The number of order_start in a tailored table should be declared as only one.

3.18. Jp.18) 6.3.3, I2:

The sentence

> The number of simple_line's thus generated is equal to one more than the value_range of the symbol_range.

is not understandable because the term "value_range" is not defined. Does this mean, in the example of NOTE, the value _range of the symbol_range is equal to 2?

3.19. Jp.19) 6.3.3, I4:

The explanation here is not understandable.

--- comments on Annex A ---

3.20. Jp.20) Annex A, KATAKANA-HIRAGANA PROLONGED SOUND MARK:

The line

```
    <U30FC> <S2A3>;<BASE>;<MIN>;<U30FC> % KATAKANA-HIRAGANA ....
```

and

```
    <UFF70> <S2A3>;<BASE>;<NARROW>;<UFF70> % HALFWIDTH KATAKANA-...
```

should be changed to

```
        <U30FC> <IGNORE>;<IGNORE>;<IGNORE>;<U30FC> % ...
```

and

```
        <UFF70> <IGNORE>;<IGNORE>;<IGNORE>;<UFF70> % ...
```

respectively as are defined in FCD.1 (see the disposition SE.11 in SC 22/WG 20 N 568 -- Disposition of comments on ballot JTC1/SC22 N N2719).

NOTE: Japan agreed in the disposition meeting in Dublin to replace the content of Annex 1 with the symbolic information in the UNICODE symdump2.txt table hearing that the information in use by vendors which implement the Unicode Collation Algorithm. Therefore, we gave only syntactical comments on the CTT in the second FCD ballot believing the UNICODE symdumpx.txt was in use and stable enough.

But the changes of the CTT from FCD.2 to FCD.3 prove that the information in symdump*.txt is not stable enough to inhibit the amendments. Therefore Japan has decided to investigate the CTT not only in syntax but in semantics without paying attention to whether the material is changed from FCD.2 or not.

3.21. Jp.21) Annex A, weight assignments for symbol characters:

The current CTT contains many troublesome weight assignments for symbol characters as are pointed out in the following comments. Japan considers it will take too much time to settle them and the best solution at this point of time is to put them back to those in FCD.1 -- ordering by code point or all IGNOREd in the first three levels. If this proposal is accepted, many of the following comments need not be investigated.

3.22. Jp.22) The symbols defined in the line

```
        collating-symbol <S0200>..<S1100> % Alphabetics & syllabics
```

are never used and many symbols of the pattern <Sxxx> are used without definitions. The line above should be corrected.

3.23. Jp.23) The following lines in the CTT

```
        % order_start <TABLE>;forward;forward;forward;forward,position
                ...
        % order_start Latin;forward;backward;forward;forward,position
```

should be changed to

```
        % order_start forward;forward;forward;forward,position
                ...
        % order_start forward;backward;forward;forward,position
```

considering the change of the table syntax and contents.

3.24. Jp.24) Annex A, the letterlike symbols and number forms:

The current CTT is based on the principle that letterlike symbols should be decomposed as far as possible. But the principle will confuse users in the following cases;

case 1: the symbol <U2173>, SMALL ROMAN NUMERAL FOUR, is decomposed to <i>+<v> while the symbol <U249C>, PARENTHESIZED LATIN SMALL A, is not decomposed -- the former, used to express one meaning "four", should be considered more tightly coupled than the latter, usually handled as a ligature.

> NOTE: if <U249C> is decomposed into '(' 'a' ')' where the pattern for the first and the third is
>
> ```
> IGNORE;IGNORE;IGNORE;...
> ```
>
> then the rule should be
>
> ```
> <U249C> <S6CF>;<BASE>;<MIN>;<U249C> ...
> ```
>
> instead of the current line
>
> ```
> <U249C> <S6CF>;<BASE>;<COMPAT>;<U249C> ...
> ```

case 2: the symbol <U2114>, L B BAR SYMBOL, is not decomposed,

case 3: the symbols <U2400>..<U2424>, CONTROL PICTURES, are not decomposed,

> NOTE: Control characters themselves should be IGNOREd, but the pictures for representing them should not be IGNOREd.

case 4: only looking at the symbol <U3300> and <U337F>, most users cannot decide the orders of decomposing -- column first (and right precedence) for the former and or row first for the latter.

Moreover it also put users into confusion that <U2108>, SCRUPLE, does not correspond to <e> although it looks very similar to <U212F>, SCRIPT SMALL E, corresponding to <e>.

Considering those, all character like symbols, which are not used to form a word, should be ordered by its code point or be IGNOREd in the first three levels un the same way as </>, <@> etc.

3.25. Jp.25) Annex A, parenthesized letters and digits:

In just the same way as the "case a - NOTE" in the last comment, all the third level weight for the parenthesized letters (including <U3200>..<3243>) not limited to Latin!) and digits, should be changed to that of the base character if the decompose-as-far-as-possible principle still holds.

3.26. Jp.26) Annex A, repeat and iteration:

The four lines

```
        <U309D> <S2A1>;<BASE>;<MIN>;<U309D> % HIRAGANA ITERATION MARK
        <U309E> <S2A1>;"<BASE><KNVCE>";"<MIN><MIN>";<U309E>
                % HIRAGANA VOICED ITERATION MARK
        <U30FD> <S2A4>;<BASE>;<MIN>;<U30FD> % KATAKANA ITERATION MARK
        <U30FE> <S2A4>;"<BASE><KNVCE>";"<MIN><MIN>";<U30FE>
                % KATAKANA VOICED ITERATION MARK
```

should be changed to

```
        <U309D> <S2A1>;<BASE>;<MIN>;<U309D> % HIRAGANA ITERATION MARK
        <U30FD> <S2A1>;<BASE>;<MIN>;<U30FD> % KATAKANA ITERATION MARK
        <U309E> <S2A1>;"<BASE><KNVCE>";"<HIRA><MIN>";<U309E> ...
        <U30FE> <S2A1>;"<BASE><KNVCE>";"<KATA><MIN>";<U30FE> ...
```

in order to be consistent with other HIRAGANA/KATAKANA handling.

3.27. Jp.27) Annex A, repeat and iteration:

```
        <U3031> <S29C>;<BASE>;<MIN>;<U3031> % VERTICAL KANA REPEAT MARK
        <U3032> <S29D>;<BASE>;<MIN>;<U3032> % VERTICAL KANA REPEAT ...
        <U3033> <S29E>;<BASE>;<MIN>;<U3033> % VERTICAL KANA REPEAT ...
        <U3034> <S29F>;<BASE>;<MIN>;<U3034> % VERTICAL KANA REPEAT ...
```

should be changed to

```
        <U3031> <S29C>;<BASE>;<MIN>;<U3031> % VERTICAL KANA REPEAT MARK
        <U3032> <S29C>;"<BASE><KNVCE>";"<MIN><MIN>";<U3032> % ...
        <U3033> <S29E>;<BASE>;<MIN>;<U3033> % VERTICAL KANA REPEAT ...
        <U3034> <S29E>;"<BASE><KNVCE>";"<HIRA><MIN>";<U3034> % ...
```

in order to be consistent with other HIRAGANA/KATAKANA handling.

3.28. Jp.28) Annex A, CJK MISCELLANEOUS:

The weight list for the characters <U3190>..<319F> should be

```
        <Uxxxx> IGNORE;IGNORE;IGNORE;<Uxxxx>
```

because they acts as annotations and should not be used for ordering.

3.29. Jp.29) p.17, Annex B.1, Canadian delta and benchmark:

The text

```
        Alternate formal ISO/IEC 14652 tailoring equivalent
```

should be changed to

```
        Alternate formal ISO/IEC 14651 tailoring equivalent
```

and the line

```
        order_start TABLE;forward;backward;forward;forward,position
```

should be changed to

```
        order_start forward;backward;forward;forward,position
```

```
    NOTE: the original line does not conform even to PDTR 14652
        because TABLE is not enclosed by '<' and '>' and
        there is no section definition anywhere.
```

3.30. Jp.30) p.17, Annex B.2, Example 2 - Danish delta and benchmark:

This is a wrong example because it contains no valid order_start entry.

3.31. Jp.31) Annex E -- Description of a collating sequence definition (informative)

The item

> (9) Easy reordering of sections. The template in ISO/IEC 14651 gives an ordering of the sections that may not be culturally acceptable in certain cultures.

should be removed because it is very hard to reorder some block of lines (sections) in the current tailoring capability and the current CTT includes no section.

3.32. Jp.32) The following items are all typographic errors.

p01] 1 Scope, bullet 1: "two characters strings" >> "two character strings" >>

p02] 4.11: "see clause 6.1" >> "see clause 6"

p03] 6.2.2, 2nd paragraph: "weights. formed by" >> "weights formed by"

p04] 6.2.2, NOTE: "codes.6.2.2.1" >> "codes. (CRLF)6.2.2.1"

p05] 6.2.3: "in clauses 6.2.1 and 6.2.3" >> "in subclause 6.2.2"

p06] 6.4: "ISO/IEC 14652" >> "ISO/IEC PDTR 14652"

p07] Annex C.2.3: "Louis 5 V" >> "Louis 05 V" (or "Louis 0005 V")

p08] Annex C.2.9: (see the section C.2.10) >> (see the subclause C.2.10).

```
p09] Annex E: "ISO/IEC 9945-2 and ISO/IEC 14652"
                >> "ISO/IEC 9945-2 and ISO/IEC PDTR 14652"
```

_____ end of Japan comments;

4. Vote and comments from the Netherlands

The NNI votes NO on FCD 14651:1999 for many of the same reasons that the NNI has voted no on earlier versions of this document.
The NNI is of the opinion that during the successive revisions of this document not enough progress has been made and that too many of the issues raised on earlier documents (not only by the NNI !) have not had appropriate attention from WG20.
As a result, the current document is again considered to be of insufficient quality and stability.

Additionally, the NNI is of the opinion that indicating shortcomings and suggesting improvements on this and earlier documents takes too much effort from the international standards preparing community. With the previous 14651 document, the total length of the comments was larger than the length of the document to comment upon!

The NNI therefore strongly suggests that either this effort is halted and the corresponding Unicode document is adopted by SC22, or, this document is withdrawn until a high quality document becomes available from WG20.
To obtain such a high quality document, it is suggested that WG20 raises funds to attract professional scientific journalists, experienced standards authors (within ISO or IEEE or elsewhere) or staff members of university departments were computer and formal languages are studied. Staff members from such departments have the appropriate training to construct and formulate such documents in a clear and unambiguous way.

The NNI will change its vote into YES only when a document of at least the same quality as the Unicode document has become available.

We will give additional reasons for the NO-vote below:

-1-

In our comment on the earlier FCD it was indicated that a Unicode document of similar scope and better quality existed. Reasons have been given for not wanting two (almost) equal standards.
The NNI is of the opinion that these reasons given earlier still hold and that the WG20 DoC did not appropriately address the issue raised.

-2-

In our comment on the earlier document the NNI suggested that the document was to be re-issued as a CD, not as an FCD. As was expected, the current document shows again a large delta. The same reasoning as presented then, holds now again.

-3-
Textual ambiguities galore; many old ones removed, many new ones introduced.
Looking at the document from a somewhat larger distance one may notice that:

- the use of the English language is complex and cumbersome and still leaves much to be desired.
- in many cases it has been tried to compress too much information in one sentence or paragraph.
- there is not always a good textual separation between:
  - the normal case and the exceptional case
  - the definition, the construction and the use of an item
  - Below, we discuss some (!) of the textual comments.

We will give extracts from the 14651 text, followed by our comment in italics.

4.1. Introduction:

This International Standard provides a method for ordering text data worldwide, and provides a Common Template Table whose tailoring meets the requirements of a given language and culture while retaining universal properties for other scipts.
This is a typical example of saying too much in one sentence:
This sentence relates language, culture and script without making clear what relations between these notions exist (or not).
Also this sentence is a typical example of not distinguishing between normal use, construction and adaptation of the CTT.
Additionally, it is unclear why this sentence talks about 'text data' wheras the rest of the document calls these 'strings'. What kind of text data is intended? Books?
Additionally, it is unclear what 'ordering text data worldwide' means.
...
However, conformance to this International Standard requires that all deviations from the Template, called "deltas", be declared to document result discrepancies.
However, <== comma missing
What is the 'Template'?
Crippled English.

This Standard describes a method to order text data independently of context.
Why not 'International Standard'?

What is the purpose of a clause named 'Introduction'?
A well written Introduction should convince the reader that he/she wants to invest money in this standard or product.
Would you do so, given this 'Introduction'?
...
A reference comparison method applicable to two character strings in order to determine their respective order in a sorted list.
Why 'reference'? Are there also non-reference comparison methods?
It is unclear what a sorted list has to do with all this.
What is a 'respective' order? An order respecting some criterion; which criterion?

The method can be applied on strings exploiting the full repertoire of ISO/IEC 10616-1.
applied TO
Strings do not exploit a repertoire; may be the strings contain characters that exploit a repertoire. However, the next sentence states that repertoires are sets, so one could perhaps simply say 'characters from the repertoire'.

This method is also applicable to subsets of that repertoire, such as, for example, those of the different ISO/IEC 8-bit standard character sets or any other character set, standardized or private, to produce ordering results valid (after tailoring) for a given set of languages for each script.
'such as, for example' seems doubly said.
again this an example of trying to say three things in one sentence.
firstly, the character set;
secondly, the tailoring
thirdly, the languages and scripts.
'standardized or private' seems irrelevant; only the repertoire seems to be relevant.
It is unclear from this sentence whether tailoring should be used (if necessary) for those subsets, or for standardized and private characters sets only.

This method uses transformation tables derived either from the CTT defined in this International Standard or from one of its tailorings.
It is unclear what 'transformation tables' are.
Furthermore, this sentence mixes up defining an item and using an item.
The whole purpose of the paragraph is to define/announce the comparison method and the kind of data that the method applies to. Nothing more.

.....
A specific CTT used by the reference comparison method.
Why 'specific'?
That the table will be used somewhere seems understandable; Again mixing up definition and use.

This table describes a basic order for all characters encoded in the first edition of ISO/IEC 10646-1 up to Amendment 7.
What is a 'basic' order? Are there non-basic (complex, composite) orders?

It allows for a further specification of a fully deterministic ordering.
What is meant by 'further'?
What is meant by 'specification'? Nothing has been specified by now.
What is a 'fully deterministic ordering'? Are there non-full, non-deterministic orders somewhere?
WHAT is being ordered in a fully deterministic way?
Again this is an example of mixing up definition and use of the table.
Again this is an example of mixing up things, this paragraph is about the table, not about properties of the comparison method.

The table is a starting point for enabling the specification of an international string ordering adapted to different cultures, without requiring an implementor to have knowledge of all the different scripts encoded in the UCS.
'starting point for enabling' seems doubly said
What is an international string ordering? Are there national string orders? Interplanetary string orders? Or is this what has been called 'worldwide' in the 'Introduction'?
Why only cultures here and no languages and no scripts?
What is the implementor implementing? Again this is mixing up definition and implementation.
Have scripts been encoded in the UCS? Earlier it was stated that characters were encoded in the UCS.

This CTT may be modified with minimal effort to suit the needs of a local environment. The main benefit, worldwide, is that for other scripts, no modification should be required and that the order will remain as consistent as possible and predictable from an international point of view.
What differentiates a local environment from a culture, a script or a language? Why yet another notion?
So, suiting the needs of my local environment, will have a worldwide benefit. That's great!
Apparently my local environment needs a script?
The order (of what?) will remain consistent (with what property?)?
So, suiting my local environment will provide an order that is predictable from an international point of view. Great!

The character repertoire described in …
There is no character repertoire described in this IS. There is a CTT derived from the UCS.

Requirements for a declaration of the differences (delta) between the comparison table used in processes and the CTT.
It is unclear wat a 'comparison table' is; is it the same as the transformation table mentioned earlier?
It is unclear what 'processes' are.

This standard does **not** mandate:

- A specific comparison method; .

????? But the first paragraph states that this IS defines: A reference comparison method????

I'm lost in the dark.
This is only one page, and there are so many pages to go.
This will take up too much of my valuable time.
I quit!
There is no need to react on these textual comments individually.
Please rewrite and restructure the whole of the document before presenting it again.

______ end of Netherlands comments; beginning of Sweden comments _______

5. COMMENTS ACCOMPANYING SWEDEN NEGATIVE VOTE ON SC22 LETTER BALLOT N2933

5.1. Comment 1:

replace the definitions section with the following (here in a more or less logical order, in some vague sense; should (must?) perhaps be put in alphabetical order...):

Character: a datum used as an elementary building block for representing text.

Character string: a sequence of characters.

Collation preparation: a process in which given character strings are mapped to (other) character strings logically before the calculation of the collation key for each of the strings.

Collation or ordering: sorting (ascending or descending) of character strings according to a collation key assigned to each of the strings. A collation key is calculated from a string (after collation preparation) and a collation table. All strings that have a Not-a-Key collation key are put in an unspecified order at the end of the resulting ordering. Other strings that have the same collation key are put in an unspecified order amongst themselves at the place indicated by their (common) key.

Collation key or ordering key: a value, that can be compared to other collation key values, constructed from a given number of collation subkeys. If appropriate collation subkeys cannot be obtained, a special Not-a-Key value will be produced. The construction must be such that subkeys at different levels do not interfere in the collation comparison.

Note: Not-a-Key will be produced only when entries are missing in the collation table relative to the string for which a collation key is to be calculated.

Collation subkey (of level n): a digit sequence that is a concatenation of a sequence of (0 or more) collation weights, in turn formed by concatenating a number of sequences of collation weights at level n, formed from a given string (after collation preparation).

Collation key comparison: a process by which two collation keys are determined to be in exactly one of the relationships less than, greater than, equal, or unordered. Unordered shall happen exactly when a Not-a-Key collation key value is involved in the comparison.

Note: Unordered will happen only when entries are missing in the collation table relative to the strings to be compared.

Collation weight: a digit string, of a given length and radix, whose value, when regarded as an integer value, reflects the relative order in which a collating element is to be placed relative to other collating elements.

Collating element: a sequence of one of more characters that have an entry in the collation table.

Collation level: the sequence number for a collation subkey.

Collation table: an unambiguous mapping from a sequence of one or more characters to a weighting element.

Collation table delta: differences from another given collation table. The given collation table, together with a given collation table delta forms a new collation table.

Collation weighting element: a given number of sequences of weights. All collation weighting elements of a collation table must have the same number of sequences of weights. Each sequence of weights is at a collation level. All weights in a collation table must use the same radix. All weights at a given level for a collation table must have the same number of digits.

Collation weight symbol: a name bound to a collation weight. This name may be used when specifying a collation table or collation table delta.

Collating element symbol: a name bound to a collating element. This name may be used when specifying a collation table or collation table delta.

Collation key reference method: the method defined in clause 6 to compute and compare collation keys.

Stable: A sorting process is stable if entries that have the same sort key are kept in the same relative order in the result as they were initially. This is a useful property when sorting multi-field items, but the sort key is built only from a subset of the fields, or if some preparation before building the sort keys looses information.

5.2. SE Comment 2: clause 5:

"(followed by exact location of syntax)"???
Delete.

5.3. SE Comment 3: clause 5:

Delete paragraph 2; this naming is not used, and shall not be used, in 14651. There seems to be no point in keeping that paragraph.

5.4. SE Comment 4: clause 6.1,

second paragraph: this paragraph needs some (minor) clarification "inverting strings"? ; "visual order", in whose eyes?; "UCS order"? Even if "I understand what you mean", please write what is meant, rather than let us guess.

5.5. SE Comment 5: clause 6.1, note 2:

"reintroduced afterwards" does not make sense.

5.6. SE Comment 5B: clause 6:

An implementation should somehow declare (in
documentation at least) whether the sort method applied on the collation
keys obtained is stable. Maybe it should even be required to be stable.

5.7. SE Comment 6: clause 6.2.1:

delete headings at level 4 (but not (all of)
their contents).

5.8. SE Comment 7: clause 6.2.1 (ex-clause 6.2.1.1):

delete second sentence of first paragraph. This sentence does not belong in
normative text
("Normally, ...sometimes called.....").

5.9. SE Comment 8: clause 6.2.1,

paragraph beginning "An optional
property...": delete that paragraph, this option implies no user benefits,
and thus adds complication (albeit optional) for no useful purpose.
It still complicates 14651 for no useful purpose.

5.10. SE Comment 9: clause 6.2.1, NOTE:

a) This is several independent notes,
and should be so split. b) Some encodings use left-to-right storage for
Arabic. This should not be done if the encoding is one of 10646.

5.11. SE Comment 10: clause 6.2.2, title:

change to "Collation key formation,
reference method".

5.12. SE Comment 11: clause 6.2.2:

a) there is no clause 6.2.2.1...; b) delete
also the heading for clause 6.2.2.2 (but not the contents), and delete
both the heading and contents of 6.2.2.3; c) the note in ex-6.2.2.3
appears to belong with clause 6.2.1 and should be moved there (not
deleted).

5.13. SE Comment 12: clause 6.2.2:

this is supposed to be a reference method.
However, the text leaves too much to be guessed by the reader, and leaves
much to be desired in terms of clarity.

5.14. SE Comment 13: ex-clause 6.2.2.2:

it is not the subkey that should be
reversed in this case, it is the sequence of weights used to form the
subkey that should be reversed before these weights are concatenated into
a subkey.

5.15. SE Comment 14: clause 6.2.3, title:

change to "Collation key comparison,
reference method"; and delete the first sentence.

5.16. SE Comment 15: clause 6.2.3:

a) all of the collation key construction
should be described in the preceding clause; b) this clause should be about the comparison only; c) "complete ordering key", the word "complete" appears to be overdoing the reference here, "collation key" or "ordering key" is sufficient.

5.17. SE Comment 16: clause 6.2.3:

the collation key comparison method is
highly overcomplexified, and is hard to understand. 14651 has no reason to try do define its own comparison, and the reference method already uses digits. Everyone is familiar with comparing numbers, including numbers that are not integers. Why not take advantage of that? If you don't want to make the entire collation key a single numeral (with value between 0 and 1), you can make each subkey a single numeral (with value between 0 and 1) by 0.<sequence of digits from weights>.

5.18. SE Comment 17: clause 6.3:

why do we need conditions for considering two
[collation] tables as equivalent?

5.19. SE Comment 18: clause 6.3:

collation weight symbols must be defined only
for a particular level, since different levels should be insulated from each other, and different levels often have different number of digits in the weights. The given syntax does not have provisions for such insulation, and separation of levels, and is thus inadequate.

5.20. SE Comment 19:

since this syntax is not required for conformity, neither
for implementations, nor for other standards/similar that tailor the CTT, it is hard to see why a lot of syntax that is not used in the actual CTT as given in Annex A is specified. The syntax should be simplified to ONLY cover what is needed for Annex A.

5.21. SE Comment 20: clause 6.3.1:

the meaning of the word "token" is not
given. It is apparent the meaning is not the one usually used in connection with parsing. Probably a correction of the text is better than the introduction of a new definition...

5.22. SE Comment 21: BNF:

the syntax should divide the CTT format into two
separate parts: 1) weight symbol declarations, 2) collating element to weighting element mapping description.

5.23. SE Comment 22: BNF (if tailoring syntax kept):

the syntax should allow
only "reorder after" to refer to the weight symbol declarations (saying "reorder after" with a reference to the mapping description part appears meaningless). Between reorder/reorder_end should only weight symbol declarations occur. The symbols must be new or of the same level as the symbol declaration "reordered after".   The following is incomplete, but corrects a number of errors/problems with the syntax given in the draft standard:

```
base_table ::=
          'table' table_name c? EOL
               weight_symbol_level_definition+
               collating_element_definition*
               (table_entry | table_entry_ranged)+
          'table-end' c? EOL

weight_symbol_level_definition ::=
          'level' c? EOL
               symbol_definition+
          ('level-end' c? EOL)?

delta_table ::=
          'table' table_name c? EOL
          'delta-from' table_name c? EOL
               weight_symbol_redefinition*
               collating_element_definition*
               (table_entry | table_entry_ranged)*
          'table-end' c? EOL

weight_symbol_redefinition ::=
          'reorder-after' simple_symbol c? EOL
               symbol_definition+
          ('reorder-end' c? EOL)?

weight_symbol_definition ::= 'collating-symbol' simple_symbol c? EOL
collating_element_definition ::=  'collating-element' simple_symbol
'from' collating_element c? EOL
table_entry ::= collating_element space+ weights_list c? EOL
table_entry_ranged ::= _

collating_element ::=  simple_symbol | ucs_symbol | '"' (simple_symbol |
ucs_symbol)* '"'
weights_list ::= weights (';' weights)*  (';' ucs_symbol+)?
weights ::=  simple_symbol | '"' simple_symbol* '"' | 'IGNORE'

ucs_symbol ::= _
simple_symbol ::= _
```

The last "level ...level-end" is for level 1, the ones preceding that one are for higher levels in order. Any symbol_weight-line in a tailoring takes priority over any corresponding entry (same collating element) in the table it is a delta from.

5.24. SE Comment 23:

"UCSsymbols" should not be allowed in the symbol declarations section; they are already declared implicitly and what they are bound to cannot be changed. It is however, unclear if a UCSsymbol stands for the UCS identifier regarded as a weight (somehow; UTF-8? UTF-16? Identification number for that character?), or that character's code in the "current encoding" (compare point 1 of clause 1) regarded as a weight (somehow).

5.25. SE Comment 24:

Some of the "well-formedness" rules are better suited to be expressed in the BNF syntax.

5.26. SE Comment 25:

The first level weight symbols for a script should indicate the script in the weight symbols: digit0..digit9, lat000..latYYY, kana00..kanaYY, greek00..greekYY, cyr00..cyrYY, ... This is in order to make any tailoring declarations that use the weight symbols of the CTT much less sensitive to additions of scripts/characters. This is a worry for instance for the EOR, or any national standard ordering based on 14651.

5.27. SE Comment 26: clause 6.3.3:

Rule I1 is syntax, not interpretation.

5.28. SE Comment 27: clause 6.3.3:

It should be said explicitly that IGNORE is equivalent to the empty list of weight symbols.

5.29. SE Comment 28: clause 6.4:

"tailoring shall be based on the CTT in Annex A" must be changed. Tailoring must be 1) chainable: e.g. EOR (when a proper minimal tailoring of the CTT, which it isn't yet) should be usable as a basis for further tailoring to e.g. Swedish; and 2) there will be new versions of the CTT, and "one should investigate the possibility of using the latest version..." without clause 6.4 preventing that.

5.30. SE Comment 29: clause 6.4:

There should be a strong recommendation that any tailoring only changes what must be changed, and does not do nonce tailorings.

5.31. SE Comment 30: clause 6.4,

note: the tailoring example is wrong. It should be something like:

```
     table ex1
     delta-from CTT1
          reorder-after <la t344> % assumed weight for z in CTT1 in this
          example
               collating-symbol <lat344A> % here assumed unused...
               collating-symbol <lat344B> % here assumed unused...
          reorder-end
          <U00E5> <lat344A>;<BASE>;<MIN>;<U00E5> % †
          <U00E4> <lat344B>;<BASE>;<MIN>;<U00E4> % „
     table-end
```

5.32. SE Comment 31: clause 6.5:

the name of the table should be part of the
file describing the table. See modified syntax above. Clause 6.5 can then
be deleted.

5.33. SE Comment 32:

The table should cover the same repertoire as
10646-1:2000/Unicode 3.0.

5.34. SE Comment 33: Annex A,

first level collating symbols: Each script should
have its own set of first level weights so as to increase the stability of
the weight symbols used for scripts as new scripts are added. This is
essential for standard documents describing minimal tailorings of the
CTT. Without very stable weight names such standards will not do miminal
tailorings, and the importance of 14651 diminishes not nearly nothing.

```
   level % 1
     collating-symbol <sym00>..<symXX> % first level significant symbols
     collating-symbol <digit0>..<digit9> % digits
     collating-symbol <latin000>..<latinXXX> % Latin letters
     collating-symbol <greek000>..<greekXXX> % Greek letters
     collating-symbol <cyr000>..<cyrXXX> % Cyrillic letters
     ...
     collating-symbol <thai00>..<thaiXX> % Thai
     ...
     collating-symbol <kana00>..<kanaXX> % Hiragana/Katakana syllables
     ...
     collating-symbol <final> % heaviest level 1 weight
    level-end
```

(the number of weights needed for each script must be determined; with a
margin)

```
     ...
     <U0030> <digit0>;<BASE>;<MIN>;<U0030> % DIGIT 0
     ...
     % <latin000> is unused, just in case someone want to put something
     before a.
     <U0061> <latin001>;<BASE>;<MIN>;<U0061> % LATIN SMALL LETTER A
     ...
     <U3041> <kana01>;<BASE>;<HIRA-SMALL>;<U3041> % HIRAGANA LETTER SMALL
A
     ...
```

5.35. SE Comment 34:

Greek small sigma(s) should have the following entries:

```
     <U03C3> <greekYYY><MIN>;<MIN>;<U03C3> %GREEK SMALL LETTER SIGMA
     <U03C2> <greekYYY><MIN>;<AFINAL>;<U03C2> %GREEK SMALL LETTER FINAL
SIGMA
```

(with an appropriate YYY, same in both lines)

5.36. SE Comment 35: Annex B.1:

Item lists starts at number 5?

5.37. SE Comment 36: Annex B.1:

The "formal" tailoring (according to 14651 syntax) should be something like:

```
   table canadian1
   delta-from CTT1
     <U00E6>
"<latin001><latinXXX>";"<BASE><VRNT1><BASE>";"<MIN><MIN><MIN>";<U00E6> %
ae
     <U00C6>
"<latin001><latinXXX>";"<BASE><VRNT1><BASE>";"<CAP><MIN><CAP>";<U00C6> %
AE
     <U00F0> "<latinXXX>";"<BASE><VRNT1>";"<MIN><MIN>";<U00F0> % eth
     <U00D0> "<latinXXX>";"<BASE><VRNT1>";"<CAP><MIN>";<U00D0> % ETH
     <U00FE>
"<latinXXX><latinXXX>";"<BASE><VRNT1><BASE>";"<MIN><MIN><MIN>";<U00FE> %
th
     <U00DE>
"<latinXXX><latinXXX>";"<BASE><VRNT1><BASE>";"<CAP><MIN><CAP>";<U00DE> %
TH
   table-end
```

(ignoring the 'order-start' in this comment)
(no reorder-after needed, since no new or changed weight symbols are used)

Where each XXX is replaced properly according to new stable weight symbols.
The comments in the delta should be the full 10646 names as well.

5.38. SE Comment 37: Annex B.3:

Each of the lines between the "reorder-after" and "reorder-end" should begin with "collating-symbol".

5.39. SE Comment 38: Annex B.4:

This is very hard to read for those (implementers) that are not fluent in Thai_ And many implementers might not be_   The important thing that is not already covered by the CTT (character rearrangement) should be clarified with code point references.

5.40. SE Comment 39: Annex B.4 (editorial comment):

there are two unnumbered subheadings, plus one subheading numbered as "2.1", and another as "2.2". Probably not what one wants_

5.41. SE Comment 40: Annex B.5:

the two lines with "reorder-after" and "reorder-end" should be deleted.

5.42. SE Comment 41: Annex C.1:

"phonetic"? You mean spelled-out as a word, not phonetic.

5.43. SE Comment 42: Annex C.2:

The item list numbering has gone astray again (problem with Word).

5.44. SE Comment 43: Annex E:

Delete. This is taken from another exposition, and does not belong in 14651.

_____ end of Sweden comments;

6. UK comments accompanying an affirmative vote on ISO/IEC FCD 14651.3

The UK notes that many of its comments on ISO/IEC FCD 14651.2 have been accommodated. On ISO/IEC FCD 14651.3, the UK votes YES with comments, and asks that these comments be accommodated.

As some of the comments on ISO/IEC FCD 14651.3 refer back to earlier UK comments on ISO/IEC FCD 14651.2, the same numbering is retained, in case it helps the editor also to refer to the previous UK comment, and to his disposition of comments.

Comment 9 may be ignored at this time, if the agenda does not permit looking at the ordering of the repertoire of ISO/IEC 10646-1:2000, which is now stable and known, but not yet published (publication is anticipated in the first quarter of 2000).

Some other comments can be ignored: where previous UK comments have been accommodated this is merely noted, as in GB1, GB2, GB3 and GB7.

These comments should be printed/displayed in a non-proportional (monospace) font so that some of the table entries can be seen easily.

6.1. GB1. Cyrillic letters used in Old Church Slavonic and Macedonian:

The UK notes that its previous comments have been accommodated in ISO/IEC FCD 14651.3, and that the whole of the Cyrillic repertoire is ordered in a consistent manner, taking account of predominant language use.

6.2. GB2. Greek

The UK notes that previous comments on ordering Greek combining characters have been accommodated.

6.3. GB3. Naming conventions

The UK notes that many of its comments on Notation relating to the use of BNF syntax have been accommodated.

However, UK comments on conventions for describing fields within tables have not been dealt with: these points are made in comment GB6 below.

6.4. GB4. Inconsistencies (spacing and non-spacing versions of characters)

It should be made clear why Currency characters and other symbols are significant at Level 1, while other symbols are ignored at Level 1. There appears to be an implicit difference, for some characters, but this should be stated explicitly.

It _will_ also be important to explain the general pervasive UCS-order within various sub-sections of the Common Template Table, to explain why this means that various punctuation characters are not ordered together (e.g. various non-combining forms of accents are separated from their combining equivalents) while in comparison different forms of DIGITS are linked together (see comment GB 6.4).

For example note the relative differences in ordering between:

```
  <U007E> IGNORE;IGNORE;IGNORE;<U007E> % TILDE
  <U00A8> IGNORE;IGNORE;IGNORE;<U00A8> % DIAERESIS
  <U0384> IGNORE;IGNORE;IGNORE;<U0384> % GREEK TONOS
  <U0385> IGNORE;IGNORE;IGNORE;<U0385> % GREEK DIALYTIKA TONOS
```

on the one hand and

```
  <U0308> IGNORE;<TREMA>;<MIN>;<U0308> % COMBINING DIAERESIS
                                         [UCS has no COMBINING TONOS]
  <U0344> IGNORE;"<TREMA><AIGUT>";
                "<MIN><MIN>";<U0344> % COMBINING GREEK DIALYTIKA TONOS
  <U0303> IGNORE;<TILDE>;<MIN>;<U0303> % COMBINING TILDE
```

on the other hand.

Differences may be justified, but the rationale should be explicitly stated.

It _may_ also be useful to explain the general pervasive UCS-order within various sub-sections of the Common Template Table, to explain why various punctuation characters are not together (e.g. the following are separated from their Latin equivalents, while different forms of DIGITS are linked together.

```
  <U037E> IGNORE;IGNORE;IGNORE;<U037E> % GREEK QUESTION MARK
  <U0387> IGNORE;IGNORE;IGNORE;<U0387> % GREEK ANO TELEIA
  <U055A> IGNORE;IGNORE;IGNORE;<U055A> % ARMENIAN APOSTROPHE
  <U055C> IGNORE;IGNORE;IGNORE;<U055C> % ARMENIAN EXCLAMATION MARK
  <U055D> IGNORE;IGNORE;IGNORE;<U055D> % ARMENIAN COMMA
  <U055E> IGNORE;IGNORE;IGNORE;<U055E> % ARMENIAN QUESTION MARK
```

6.5. GB5. Ordering of SPACE

There seems to be some minor work to be done regarding explanations of ordering of SPACE, and similar "white space" characters. In the former versions of ISO/IEC FCD 14651, a toggle was forced, so that the user had to decide one way or the other, by decommenting the relevant field. The draft standard had additional comment fields to assist the user in this.

It makes a difference whether SPACE is ignored in filing or treated as a blank character. Compare ISO/IEC FCD 14651 and the Unicode Collation Algorithm. Many users will have been used to space being counted as at level 1 in many operating systems and applications, and will be surprised to see ISO/IEC FCD 14651 ordering it differently.

Not ordering it at level one may indeed be the preferred solution (it certainly makes ordering of some Southeast Asian scripts easier, where spaces are not used between words) but further explanation of this point is needed in the standard.

6.6. GB6. Conventions for describing fields within the Common Template Table

Conventions for describing fields in the tables of ISO/IEC FCD 14651.3 and its equivalents in the Unicode Ordering Algorithm SYMDUMP2.TXT and EOR - the European Ordering Rules (prENV 13710) - all vary to some degree. Given that these are supposed to be harmonised, and as it is likely that some users will use some of these standards in conjunction with each other, any differences need to be explained. A description of the conventions used need not be lengthy.

GB6.1 - GB6.4 deal with specific issues here.

6.6.1. GB6.1

For example, prENV 13710 uses conventions based on
ISO/IEC 10646 names:

```
<U01DF> <a>;"<DIAERESIS><MACRON>";<SMALL>;<U01DF> % LATIN SMALL
                        LETTER A WITH DIAERESIS AND MACRON
```

ISO/IEC FCD 14651.3 (and the Unicode Collation Algorithm) use different naming conventions:

```
<U01DF> <S6CD>;"<TREMA><MACRO>";<MIN>;<U01DF> % LATIN SMALL
                        LETTER A WITH DIAERESIS AND MACRON
```

A brief description of these uses is requested (a single paragraph explaining that conventions used are different to those in ISO/IEC 10646-1, without going into detail on each term, would suffice).

6.6.2. GB6.2

There are also other unexplained differences between them as in [1], [2], and [3] below:

```
[14651]   <U0041> <S6CD>;<BLANK>;<CAP>; <U0041> % LATIN CAPITAL LETTER A
[Unicode] <U0041> <S6CD>;<BLANK>;<CAP>; <@0041> % LATIN CAPITAL LETTER A
[EOR]     <U0041> <a>;<BLANK>;<CAPITAL>;<U0041> % LATIN CAPITAL LETTER A
                  [1] (weight)    [2]   [3]
```

A brief paragraph on such differences is requested, just saying that there may be differences in detail between the Common Template table in ISO/IEC FCD 14651 and some of its implementations.

6.6.3. GB6.3

In ISO/IEC FCD 14651, the records in the default table use <COMPAT> etc: compatibility characters are defined in Unicode but not in ISO/IEC FCD 14651 or in ISO/IEC 10646: therefore their use in the tables of ISO/IEC FCD 14651.3 requires some explanation to the user.

These explanations need not be lengthy, but there should be more detail, in a section or subsection of the standard entitled "Notation" on the conventions used (as in many ISO standards).

6.6.4. GB6.4

With DIGITS, unnecessary notation is introduced at Level 2, when this is merely informative: it is clear that the distinction is at level 4. There would be no difference if Level 2 annotations were all left as <BASE> in the appropriate parts of the DIGITS section of the Common Template Table. As it stands the information can hinder the user. Relying on the character name, which is already in the entry, to supply this information would be far more helpful and much less confusing.

```
<U0030> <S6C5>;<BASE>;<MIN>;<U0030> % DIGIT ZERO
<UFF10> <S6C5>;<BASE>;<WIDE>;<UFF10> % FULLWIDTH DIGIT ZERO
<U24EA> <S6C5>;<BASE>;<CIRCLE>;<U24EA> % CIRCLED DIGIT ZERO
<U2070> <S6C5>;<BASE>;<MNN>;<U2070> % SUPERSCRIPT ZERO
<U2080> <S6C5>;<BASE>;<MNS>;<U2080> % SUBSCRIPT ZERO
<U0660> <S6C5>;<ARABIC>;<MIN>;<U0660> % ARABIC-INDIC DIGIT ZERO
<U06F0> <S6C5>;<EXTARABIC>;<MIN>;<U06F0> % EXTENDED ARABIC-INDIC DIGIT ZERO
<U0966> <S6C5>;<NAGAR>;<MIN>;<U0966> % DEVANAGARI DIGIT ZERO
<U09E6> <S6C5>;<BENGL>;<MIN>;<U09E6> % BENGALI DIGIT ZERO
<U0A66> <S6C5>;<GURMU>;<MIN>;<U0A66> % GURMUKHI DIGIT ZERO
<U0AE6> <S6C5>;<GUJAR>;<MIN>;<U0AE6> % GUJARATI DIGIT ZERO
```

```
<U0B66> <S6C5>;<ORIYA>;<MIN>;<U0B66> % ORIYA DIGIT ZERO
<U0C66> <S6C5>;<TELGU>;<MIN>;<U0C66> % TELUGU DIGIT ZERO
<U0CE6> <S6C5>;<KNNDA>;<MIN>;<U0CE6> % KANNADA DIGIT ZERO
<U0D66> <S6C5>;<MALAY>;<MIN>;<U0D66> % MALAYALAM DIGIT ZERO
<U0E50> <S6C5>;<THAII>;<MIN>;<U0E50> % THAI DIGIT ZERO
<U0ED0> <S6C5>;<LAAOO>;<MIN>;<U0ED0> % LAO DIGIT ZERO
<U0F20> <S6C5>;<BODKA>;<MIN>;<U0F20> % TIBETAN DIGIT ZERO
<U3007> <S6C5>;<CJKVS>;<MIN>;<U3007> % IDEOGRAPHIC NUMBER ZERO
<U3358> "<S6C5><S70B9>";
              "<BASE><BASE>";
                      "<COMPAT><COMPAT>";
                             <U3358> % IDEOGRAPHIC TELEGRAPH SYMBOL
                                       FOR HOUR ZERO
```

6.7. GB7. Apparent inconsistencies in ordering in the default table

The UK is grateful for a more consistent ordering of LATIN SMALL LETTER TONE TWO, FIVE and SIX, and also awaits similar allocation of remaining tone letters in a future version of UCS, and their reordering in a future version of ISO/IEC FCD 14651 alongside LATIN SMALL LETTER TONE TWO, FIVE and SIX.

No action is necessary on ths comment at this time.

6.8. GB8. Korean, and other CJK ordering

The UK is grateful for explicitly stating the relevant jamo range (U+1100..U+11F9) when building weights for Hangul syllables, in response to its earlier comment.

However, following the adjacent comment:

```
% Weights for unified Han characters follow the Unified Repertoire and
%   Ordering, which is a language-neutral, traditional radical-stroke order.
```

it would be valuable to also add a further comment like "for many purposes, specific tailorings of Han character ordering for Chinese, Japanese or Korean use are likely to be required. These would be related to the relevant portions of the character ranges above for ordering by pinyin (Latin characters), Chinese bopomofo, Japanese kana, or Korean jamo ordering. Specifications for linking these with the language-neutral, traditional radical-stroke order in

```
 <U4E00>..<U9FA5> <S4E00>..<S9FA5>;<BLANK>;<MIN>;<U4E00>..<U9FA5> % Han
```

is outside the scope of this standard."

6.9. GB9. Script-by-script ordering of the ISO/IEC 10646-1:2000 repertoire.

Given the timescale involved, it may not be feasible to deal with the comment below in the upcoming November 1999 meeting of SC22/WG20. However, the UK expects that this should be dealt with at the meeting after that.

The UK considers that a reasonably predictable order should be explicit in the ISO/IEC FCD 14651 default table, and should take on board the ordering of the repertoire of ISO/IEC 10646-1:2000 and Unicode version 3.0.

This should be West through East by the point of origin of each script, an order broadly similar to, although not completely identical with, that in BMP of ISO/IEC 10646-1:2000 (subdivided where necessary North through South, as in South Asian scripts in ISO/IEC 10646-1).

Users who are using printed or computer-held multilingual/multiscript indexes or other data sources can imagine this in relation to the scripts in which they are interested. They should not need to refer to ISO/IEC 10646-1:2000 or some other standard.

This is fairly easy to achieve with only a very small number of differences between script order in ISO/IEC 10646-1:2000 and ISO/IEC FCD 14651, and has already been done for Georgian.

Such ordering was implicit in earlier drafts of ISO/IEC FCD 14651, as noted in the earlier comments by the UK (see UK comments, section 3.A.2. Order of scripts, in earlier UK comments) but is no longer specified in any single area of ISO/IEC FCD 14651.

The UK proposes that the order adopted in the early drafts of ISO NP 15921: Generalized conversion methods, being developed in ISO/TC46/SC2/WG8: Transliteration and Computers, be used.

There is also an additional question of whether minority scripts or historical scripts that are not used in official languages should be ordered separately from other scripts, or interfiled (ordering (a) and (b) below in a single sequence) - there are arguments either way.

(a) Scripts used in official languages worldwide (at country level) [1] [2]

Americas/Europe:	Latin, Greek, Cyrillic, Georgian, Armenian;
Near East:	Hebrew;
West Asia/North Africa:	Arabic;
Northeast Africa:	Ethiopic;
South Asia:	Devanagari, Bengali/ Assamese, Gurmukhi, Gujarati, Oriya, Tamil, Telugu, Kannada, Malayalam, Sinhala; Thaana;
Southeast Asia:	Thai, Lao, Myanmar (Burmese), Khmer;
Inner Asia:	Dzongka/Tibetan, Mongolian;
East Asia:	Korean, Japanese, Chinese.

(b) Scripts used in official languages below country level [1] by minorities within countries, and in religious/historical texts [2]

Americas:	Cherokee, Canadian Aboriginal Syllabics;
Europe:	Ogham, Runic;
Near East:	Syriac;
East Asia:	Yi (Southwest China),

_______ end of UK comments

7. US vote and comments

Document SC22 N2933
October 10, 1999

The US votes NO on the Third FCD Ballot for FCD 14651: Information technology International String Ordering and Comparison - Method for Comparing Character Strings and Description of a Common Tailorable Ordering Template, but will gladly change the vote to YES, if the comments below are accommodated.

7.1. Technical Comments

7.1.1. p. 1, NOTE 2.

This note references the Unicode Standard Version 2.1, but the appropriate reference occurs neither in the Normative References nor in the Bibliography. We suggest that the appropriate reference for the Unicode Standard, Version 2.1, be added to the Bibliography.

7.1.2. p. 4, definition 4.16.

This definition is incomplete in the text and must be fixed.

7.1.3. p. 5, NOTE 1.

This note refers to Unicode normalization, but the appropriate reference occurs neither in the Normative References nor in the Bibliography. We suggest that the appropriate reference for Unicode Technical Report #15, Unicode Normalization, be added to the Bibliography, and a more complete reference be added at this note.

7.1.4. p. 9, BNF syntax.

The "line_completion" tokens in the production rules for order_start, order_end, reorder_section_after, reorder_after, and reorder_end should be removed. They are redundant with the line_completion token in the production rule for tailoring_line.

7.1.5. p. 14, NOTE.

This note refers to the Unicode collation algorithm, but the reference occurs neither in the Normative References nor in the Bibliography. We suggest that the appropriate reference for Unicode Technical Report #10, Unicode Collation Algorithm, be added to the Bibliography, and a more complete reference be added at this note.

7.2. Technical Changes to Annex A -- Common Template Table

7.2.1. Fixes for Thai

To match cultural expectations for a correct Thai sort, the following changes should be made to the Thai entries in the Common Template Table. Incidentally, these changes will put the Common Template Table in synch with the principles explained in Annex B.4

a. The Thai vowel indicator U+0E47 THAI CHARACTER MAITAIKHU should be treated exactly like the Thai tone marks, rather than being given a primary weight as for other Thai vowels. This implies that:

i. collating symbol <D0E47> for THAI CHARACTER MAITAIKHU be added just before the collating symbol <D0E46>.

ii. a weight entry for THAI CHARACTER MAITAIKHU be added:
<U0E47> IGNORE;<D0E47>;<MIN>;<U0E47> just before <U0E46>.

iii. the current weight entry for THAI CHARACTER MAITAIKHU be removed from the table.

b. U+0E33 THAI CHARACTER SARA AM and U+0EB3 LAO VOWEL SIGN AM should be treated as units, rather than as combinations of the weights for the NIKHAHIT and the vowel SARA AA. This implies that:

i. the current weight entry for THAI CHARACTER SARA AM be changed to
<U0E33> <SE20>;<BASE>;<MIN>;<U0E33> % THAI CHARACTER SARA AM

ii. the current weight entry for LAO VOWEL SIGN AM be changed to
<U0EB3> <SE4F>;<BASE>;<MIN>;<U0EB3> % LAO VOWEL SIGN AM

c. The change for MAITAIKHU impacts the auto-generated primary weight symbols, so the table should be regenerated to correct the resulting sequence of primary weight symbols.

7.2.2. Fixes for archaic Greek letter case

The third-level weights for several archaic Greek letters that have no case pairs in the Unicode 2.1 repertoire were misassigned to <MIN> instead of <CAP>. Those should be corrected. (Note that the lowercase correspondents of those letters were added by 10646 amendment Amendment 30, and will appear, appropriate weighted in future revisions to the 14651 Common Template Table, so the uppercase forms currently in the table should be correctly weighted.)

Affected characters are:

<U03DC> GREEK LETTER DIGAMMA
<U03DA> GREEK LETTER STIGMA
<U03DE> GREEK LETTER KOPPA
<U03E0> GREEK LETTER SAMPI

7.2.3. Case fix for Palochka

As for the 4 Greek characters, one Cyrillic character with no case pair should have its third-level weight corrected from <MIN> to <CAP>:

```
<U04C0> CYRILLIC LETTER PALOCHKA
```

7.2.4. Misuse of symbol <BLANK>.

The following two lines at the end of the table:

```
<U4E00>..<U9FA5> <S4E00>..<S9FA5>;<BLANK>;<MIN>;<U4E00>..<U9FA5> % Han
% <UAC00>..<UD7A3> <SAC00>..<SD7A3>;<BLANK>;<MIN>;<UAC00>..<UD7A3> % Hangul
```

have an undefined symbol <BLANK> in them. That should be corrected to use the symbol <BASE>, which is otherwise used in that position in the table:

```
<U4E00>..<U9FA5> <S4E00>..<S9FA5>;<BASE>;<MIN>;<U4E00>..<U9FA5> % Han
% <UAC00>..<UD7A3> <SAC00>..<SD7A3>;<BASE>;<MIN>;<UAC00>..<UD7A3> % Hangul
```

7.3. Technical Issue, Annex B.5 Cyrillic

The U.S. would strongly object to the inclusion of the B.5 tailorings for Cyrillic into the Common Template Table for the following reasons:

1. To do so would very significantly complicate the autogeneration of the Common Template Table, which will be a maintenance and quality problem for future editions of 14651 that add more characters.

2. Adding this material to the Common Template Table would introduce baseform + combining mark weightings into the CTT, something that is currently not required, but which would significantly increase the complexity of implementations of the table before tailorings. (That would be an additional implementation penalty to be carried around by all implementations, including those which are not primarily concerned with Cyrillic.)

3. The actual tailorings required for Russian are quite a bit less than that indicated in Annex B.5. Common Cyrillic requires only slightly more. Only a full tailoring for all Cyrillic extensions requires addition of all the information of Annex B.5.

Our preferred solution for this issue is to retain B.5 as an annex describing Cyrillic tailoring, but to divide it up into three parts, to show the Russian, the Common Cyrillic (i.e. Serbian, Macedonia, Bulgarian, Byelo-Russian, Ukrainian) tailoring, and the extended Cyrillic tailoring. This will make it clear that the tailoring required for Russian, for example, is no more formidable than the Canadian tailoring of Annex B.1.

7.4. Technical Issue, Annex E

The U.S. objects to the inclusion of this Annex, which is an attempt to reinject a dependency between 14651 and PDTR 14652, from which most of the text for Annex E derives.

The inappropriateness of the addition of this material here is illustrated by the fact that it includes a number of editorial and other errors that the U.S. committee has commented on in the context of ballot comments on PDTR 14652. By replicating that material into an Annex in 14651, those errors would need to be corrected once again in this text, with allowances for the edited down version of the text that appears in Annex E.

Furthermore, the suggestions made in Annex E change the syntax of at least one keyword in ways incompatible with that described in the normative BNF of Section 6.3 of 14651 (viz. order_start). This might be appropriate in PDTR 14652, but is not appropriate in an informative annex to 14651 itself, since it is more likely to just confuse rather than elucidate there.

This problem is not fixed simply by labelling Annex E "informative". Annex E should be removed entirely, with the focus being on the correction of its corresponding content in PDTR 14652, rather than to try once again to hitch 14652's wagon to 14651.

If WG20 cannot reach consensus regarding the removal of Annex E, the U.S. delegation will provide a long list of suggested editorial changes to make its inclusion less objectionable in the context of 14651.

7.4.1. Editorial Comments

p. iv. 2nd paragraph. result ==> resultant

p. 1, 2nd paragraph. "two characters strings" ==> "two strings"

p. 4, definition 4.8. remove extraneous "-" in definition

p. 4, section 5, first paragraph. "(followed by exact location of syntax)" is apparently incomplete. This should, presumably constitute a reference to Amendment 9, which should then also be included in the normative references for 14651.

p. 5, 1st paragraph. Remove extra quotation mark at end of the paragraph.

p. 7, section 6.2.2.1. Correct the line break and style for this section header.

p. 13, NOTE to I6. I1 and I2 should be corrected to I4 and I5, respectively.

p. 15, NOTE. "too long comments" ==> "long line lengths"

_____________________ end of USA comments _____________________
_____________________ end of SC22 N3025 _____________________