ISO/IEC JTC1/SC22/WG20 N1012
Korea JTC1/SC22 (SC22-WG20) Doc #: K158

Author: KIM, Kyongsok (GIM, Gyeongseog)
Date: 2003-02-11

Subject: Hangeul-related portion of CTT in ISO/IEC 14651 (1)

Reference: WG20 N953 (=Korea K152). Obsoletes WG20 N952 (= Korea K151).

Summary: To overcome some problems with the current CTT (Common Template Table) of ISO/IEC 14651 with regard to Hangeul, a preliminary proposal is presented in this paper. Specifically, two methods are proposed. First, a proposed revision of CTT is presented. Second, transformation at the preprocessing is proposed.

This paper may be incomplete yet. A more refined proposal will be provided in the future.

1. Five categories of Hangeul letters/syllables in UCS

Hangeul letters/syllables in ISO/IEC 10646 can be classified as follows:

	letters	syllables
1) Hangeul IPF-Johab letters - U+11xx (U+1100 - 11FF)	240	--
2) Hangeul Wanseong syllables - U+A/B/C/Dxxx (U+AC00 - D7A3)	--	11,172
3) Hangeul Compatibility CV-Johab letters - U+31xx (U+3131 - 318E)	94	--
4) Hangeul Half-width CV-Johab letters U+FFxx (U+FFA0 - FFDC)	52	--
5) Hangeul Enclosed (Parenthesized /Circled) syllables/letters) U+32xx (U+3200-321C, 3260-327B)	28	29

* IPF=initial-peak-final, CV=consonant-vowel

2. Problems with the current CTT of ISO/IEC 14651 for Wanseong syllables (U+AC00-D7A3) and IPF-Johab letters (U+11xx)

2.1 A Problem when using IPF-Johab letters and the current CTT

- When we sort data containing IPF-Johab (U+11xx) letters using the current CTT, we get the following incorrect results.

ᄀ ᅡ (U+1100 1161) line 11
ᄀ ᅡ A (U+1100 1161 0041) line 12
ᄀ ᅡ ㄱ (U+1100 1161 11A8) line 13
ᄀ ᅡ 万 (U+1100 1161 4E07) line 14

- In a correctly sorted output, "가万" (line 14) precedes "각" (line 13), as shown below:

ᄀ ᅡ (U+1100 1161) line 11
ᄀ ᅡ A (U+1100 1161 0041) line 12
ᄀ ᅡ 万 (U+1100 1161 4E07) line 14
ᄀ ᅡ ㄱ (U+1100 1161 11A8) line 13

2.2 A proposed solution to solve the problem in 2.1

1) [Decomp3] Each of SI, SP, and SF letters is represented as 3 weights

- Notation:

. simple letters: SI1, SI2, SI3, SP1, SP2, SP3, SF1, SF2, SF3
. SI: a syllable-initial letter composed of 1,2, or 3 simple letters
. SP: a syllable-peak letter composed of 1,2, or 3 simple letters
. SF: a syllable-final letter composed of 1,2, or 3 simple letters

- terms: 2-complex and 3-complex letters

. 2-complex letter a complex letter composed of 2 simple letters:
SI = SI1 + SI2 (e.g., ᄁ =ᄀ + ᄀ)
SP = SP1 + SP2 (e.g., ᅤ= ᅡ + ᅵ)
SF = SF1 + SF3 (e.g., ᆬ=ᆫ+ᆽ)
. 3-complex letter a complex letter composed of 3 simple letters
SP=SP1 + SP2 + SP3 (e.g., ᅫ = ᅩ+ ᅡ+ ᅵ, ᅰ = ᅮ+ ᅥ+ ᅵ)

- The details of Decomp3 are described below:

1a) A simple letter SI is represented as follows (SI2=SI3=0000):

SI = SI1 0000 0000 (the same for SP and SF)

e.g., ᄀ (U+1100) --> 1100 0000 0000
ᅡ (U+1161) --> 1161 0000 0000
ᆨ (U+11A8) --> 11A8 0000 0000

1b) A 2-complex letter SI composed of 2 simple letters SI1 and SI2 is represented as follows (SI3=0):

SI = SI1 SI2 0000 (the same for SP and SF)

e.g., ᄁ (U+1101) --> 1100 1100 0000
ᅢ (U+1162) --> 1161 1175 0000
ᆩ (U+11AA) --> 11A8 11BA 0000

1c) A 3-complex letter SP composed of 3 simple letters SP1, SP2 and SP3 is represented as follows:

SP = SP1 SP2 SP3

e.g., ᅪ (U+116B) --> 1169 1161 1175

2) [Decomp9] Each Hangeul syllable is represented as 9 weights

- assume that <SHG-L> (Hangeul low) is between <S10FF> and <S1100> and
<SHG-H> (Hangeul high) is between <S11FF> and <S1200>

- There are six types of Hangeul syllables (complete or incomplete) as shown below. Each of the six types of syllables is transformed into 9 weights as shown below:

a)	SI1	SI2	SI3	HG-L	0000	0000	HG-L	0000	0000 (SI only)
b)	SI1	SI2	SI3	SP1	SP2	SP3	HG-L	0000	0000 (SI + SP)
c)	SI1	SI2	SI3	SP1	SP2	SP3	SF1	SF2	SF3 (SI + SP + SF)
d)	HG-H	0000	0000	SP1	SP3	SP3	HG-L	0000	0000 (SP only)
e)	HG-H	0000	0000	SP1	SP2	SP3	SF1	SF2	SF3 (SP + SF only)
f)	HG-H	0000	0000	HG-H	0000	0000	SF1	SF2	SF3 (SF only)

- The above decomposition is called Decomp9.

2.3 How to handle Hangeul Wanseong Syllables (U+AC00-D7A3) in CTT?

- For each Hangeul syllable (complete or incomplete),
  . apply Decomp3/Decomp9 to assign 9 weights at the first level.
- Weights at the other levels can be handled easily.

- The current and proposed lines in CTT for Several Hangeul syllables are shown below:

a1) "가" <UAC00> [compare it with a2) below]
current:
<UAC00> <S1100><U1161>; <BASE><BASE>; <MIN><MIN>; <U1100><U1161>

proposed (in CTT)
<UAC00> <S1100><S0000><S0000> <S1161><S0000><S0000> <SHG-L><S0000><S0000>;
<BASE> * 9 times; <MIN> * 9 times; <UAC01> <U0000> * 8 times

b) "각" <UAC01>
current:
<UAC01> <S1100><U1161><U11A8>; <BASE><BASE><BASE>;
<MIN><MIN><MIN>; <U1100><U1161><U11A8>

proposed (in CTT)
<UAC00> <S1100><S0000><S0000> <S1161><S0000><S0000> <S11A8><S0000><S0000>;
<BASE> * 9 times; <MIN> * 9 times; <UAC01> <U0000> * 8 times

c) "꽝" <UAF5D>
current:
<UAF5D> <S1101><S116A><S11BC>; <BASE><BASE><BASE>;
<MIN><MIN><MIN>; <U1101><U116A><U11BC>
proposed (in CTT)
<UAC00> <S1100><S1100><S0000> <S1169><S1161><S0000> <S11BC><S0000><S0000>;
<BASE> * 9 times; <MIN> * 9 times; <UAC01> <U0000> * 8 times

2.4 How to handle Hangeul IPF-Johab Letters (U+11xx) in CTT?

- Since the CTT does not have a state reflecting the previous letters, we cannot apply Decomp3/Decomp9 to CTT.
- Therefore, a preprocessing should implement Decomp3/Decomp9.
  . At the preprocessing, <SHG-L> and <SHG-H> can be represented as a user-defined code positions.
- Then, in CTT, basically we should not apply any transformation to Hangeul IPF-Johab letters (U+11xx) at the level 1.
- Examples are shown below:

a2) "가" <U1100><U1161> [compare it with a1) above]
current (in CTT):
<U1100><U1161> <S1100><S1161>; <BASE><BASE>; <MIN><MIN>; <U1100><U1161>

proposed (at preprocessing, not in CTT)
<U1100><U1161>
<S1100><S0000><S0000> <S1161><S0000><S0000> <SHG-L><S0000><S0000>;
<BASE> * 9 times; <MIN> * 9 times; <UAC01> <U0000> * 8 times

d) "ᄀ " (an incomplete syllable): <U1100> <U1160>
current (in CTT)
<U1100><U1160> <S1100><S1160>; <BASE><BASE>; <MIN><MIN>; <U1100><U1160>

proposed (at the preprocessing step, not in CTT)
<U1100><U1160>
<S1100><S000><S0000> <SHG-L><S0000><S0000> <SHG-L><S0000><S0000>;
<BASE> * 9 times; <MIN> * 9 times; <UAC01> <U0000> * 8 times

e) " ᅡ" (an incomplete syllable): <U115F> <U1161>
current (in CTT):
<U115F><U1161> <S115F><S1161>; <BASE><BASE>; <MIN><MIN>; <U115F><U1161>

proposed (at the preprocessing step, not in CTT)
<U115F><U1161>
<SHG-H><S000><S0000> <S1161><S0000><S0000> <SHG-L><S0000><S0000>;
<BASE> * 9 times; <MIN> * 9 times; <UAC01> <U0000> * 8 times

f) "ㄱ" (a syllable -finla letter; an incomplete syllable): <U115F> <U11A8>
current (in CTT):
<U115F><U11A8> <S115F><S11A8>; <BASE><BASE>; <MIN><MIN>; <U115F><U11A8>

proposed (at the preprocessing step, not in CTT)
<U115F><U11A8>
<SHG-H><S000><S0000> <SHG-H><S0000><S0000> <S11A8><S0000><S0000>;

<BASE> * 9 times; <MIN> * 9 times; <UAC01> <U0000> * 8 times

3. Comp., Half-widht, Enclosed Hangeul letters/syllables

3.1 Hangeul Compatibility CV-Johab (U+31xx)

3.1.1 A Problem with the current CTT RE Compatibility CV-Johab

1) A problem with Letters
- e.g., a letter "ㄱ": U+3131, which will be transformed by CTT as follows:
  --> <S1100>;<BASE>;<COMPAT>;<U3131> % HANGUL LETTER KIYEOK
- In constrast, it is not the same as a letter "ㄱ" in IPF-Johab
  <U1100> <U1160>, which will be transformed by CTT as follows:
  --> <S1100> <S1160>; <BASE> <BASE>;<MIN> <MIN>; <U1100><U1160>
- As a result, the two will not compare equal even at level 1, which is incorrect.

2) A problem with Syllables
- e.g., a syllable "가": U+3164 3131 314F 3164 U+3131, which will be transformed by CTT as follows:
  <S1160><S1100><S1161><S1160>; <BASE><BASE><BASE><BASE>;
  <COMPAT><COMPAT><COMPAT><COMPAT>; <U3164><U3131><U314F><U3164>;
- In constrast, it is not the same as a letter "가" in IPF-Johab
  <U1100> <U1161>, which will be transformed by CTT as follows:
  --> <S1100> <S1161>; <BASE> <BASE>;<MIN> <MIN>; <U1100><U1161>
- As a result, the two will not compare equal even at level 1, which is not correct.

3) The relevant portion of CTT

```
<U1100> <S1100>;<BASE>;<MIN>;<U1100> % HANGUL CHOSEONG KIYEOK
<U1160> <S1160>;<BASE>;<MIN>;<U1160> % HANGUL JUNGSEONG FILLER

<U3131> <S1100>;<BASE>;<COMPAT>;<U3131> % HANGUL LETTER KIYEOK
<U314F> <S1161>;<BASE>;<COMPAT>;<U314F> % HANGUL LETTER A
<U3164> <S1160>;<BASE>;<COMPAT>;<U3164> % HANGUL FILLER
```

3.1.2 A proposd solution RE Compatibility CV-Johab letters

- Since filler characters U+115F and U+1160 in IPF-Johab and U+3164 in Compatibility CV-Johab have drastically different usage, a blind transformation in the current CTT produces incorrect results. In other words, CTT does not take into consideration the preceding letters and, therefore, cannot transform Compatibility CV-Johab
- **A reasonable solution**

. A preprocessing applies Decomp3/Decomp9 to Compatibility CV-Johab letters

. so that they are transformed into IPF-Johab letters and

. therefore Compat. CV-Johab letters should not be processed by CTT.

3.2 Hangeul Halfwidth CV-Johab letters (U+FFxx)

3.2.1 A Problem with the current CTT RE Halfwidth CV-Johab letters

- The usage of Halfwidth CV-Joahba is different from IPF-Johab or Compatibility CV-Johab letters.
- The problem is somewhat similar to, though not identical with, that of Compatibility CV-Johab letters.
- The details are not shown here.

3.2.2 A proposed solution RE Halfwidth CV-Johab letters

. A preprocessing applies Decomp3/Decomp9 to Halfwidth CV-Johab letters

. so that they are transformed into IPF-Johab letters and

. therefore Halfwidth CV-Johab letters should not be processed by CTT

3.3 Enclosed (Parenthesized, Circled) Hangeul letters (U+32xx)

3.3.1 Problems with the current CTT RE Enclosed Hangeul letters

1) e.g.,

<U3200> --> <S1100>;<BASE>;<COMPAT>;<U3200> % PARENTHESIZED HANGUL KIYEOK
<U3260> --> <S1100>;<BASE>;<CIRCLE>;<U3260> % CIRCLED HANGUL KIYEOK

2) Independent Hangeul letter KIYEOK is represented in IPF-Johab as
U+1110 1160

- Therefore, two enclosed Hangeul letters will not be equal to independent letters at level 1.

3.3.2 A proposd solution RE Enclosed letters

- **A reasonable solution:**
  incorporate Decomp3/Decomp9 in the relevant lines in CTT as follows:

a) PARENTHESIZED HANGUL KIYEOK
current in CTT
<U3200> --> <S1100><S1160>;<BASE><BASE>;<COMPAT><COMPAT>;<U3200>

proposed (in CTT)
<S1100><S000><S0000> <SHG-L><S0000><S0000> <SHG-L><S0000><S0000>;
<BASE> * 9 times; <MIN> * 9 times; <UAC01> <U0000> * 8 times

b) PARENTHESIZED HANGUL KIYEOK A
current in CTT
<U320E> --> <S1100><S1161>;<BASE><BASE>;<COMPAT><COMPAt>;<U320E>

proposed (in CTT)
<U320E> <S1100><S0000><S0000> <S1161><S0000><S0000> <SHG-L><S0000><S0000>;
<BASE> * 9 times; <MIN> * 9 times; <UAC01> <U0000> * 8 times

4. Other issues

4.1 Old Hangeul complex letters not included in UCS

- Korean scholars claim that they found tens of old complex letters not included in UCS.
- We can apply Decomp3/Decomp9 to newly found old complex letters.
- Therefore, newly found old complex letters do not pose any problem.
- Note. In my previous paper (WG20 N953 (=Korea K152)), I proposed that newly found Old Hangeul complex letters be defined as collating-element as follows:

collating-element <Uxxxx_yyyy> from "<Uxxx><Uyyyy>"
collating-element <Uxxxx_yyyy_zzzz> from "<Uxxx><Uyyyy><Uzzzz>"

However, due to Decomp3/Decomp9 proposed in this paper, we do not treat newly found old complex letters as collating-element any longer.

4.1x complex letters included in UCS (under construction) ??

- we may have to define coll. elts for the complex letters included in UCS ??
  . ㅱ = ㄱ + ㄱ

- 2-complex letters:
collating element <U1101> from <U1100> <U1100>

- 3-complex letters:
collating element <Uㅂ ㅅ ㄱ > from <Uㅂ ><Uㅅ ><Uㄱ >
collating element <Uㅂ ㅅ ㄱ > from <Uㅂ ㅅ ><Uㄱ >
collating element <Uㅂ ㅅ ㄱ > from <Uㅂ ><Uㅅ ㄱ >

4.2 Old Hangeul Tone marks (Bangjeom)

- There is no widely accepted collating sequence for Old Hangeul letters and therefore the way to treat tone marks is not well defined yet either.
- We need a further investigation.

5. Relationship between CTT and UAX #15

5.1 The exact relationship between CTT in ISO/IEC 14651 and UAX #15.

- The author does not know the exact relationship between CTT in ISO/IEC 14651 and UAX #15. I have two questions.

1) It is quite clear that CTT does not have a state reflecting the preceding letters.
- However, I don't know exactly whether or not UAX #15 has the concept of state.

2) I wonder if there is any relationship between CTT and Decomp/Comp in UAX #15.

5.2 KIM's paper about Hangeul decomposition/compsition

- Comments Re: UAX #15 can be found in a separate paper, WG20 N953 (Korea K152), A summary of a paper by Kim, K: New Canonical Decomposition and Composition processes for Hangeul.

. A full paper can be found in WG20 N954 (= Korea K153), Paper by KIM, K: New Canonical decomposition and composition processes for Hangeul. This paper was published in CSI (Computer Standarads & Interfaces) Vol. 24 (2002), pp. 69-82. The full paper can be also found at

http://asadal.cs.pusan.ac.kr/hangeul/i18n/sc22wg20-K153.PDF

* * *