No organization has endorsed
this document

Aram Airapetian
<aramhayr@hotmail.com>

RFC: EN-0003

Type: Draft

Date: 11/02/2000

Armenian Text Input, Display, and Externalization Support

This document defines console driver (keyboard, display) and application (e.g. text editor) level protocols [interfaces] to support different Armenian (Western and Eastern) orthographies. It is based on Basic Armenian Character Set described in [1].

It is a recommendation for Armenian text processing software developers.

The complexity of Eastern (Republic of Armenia) Armenian text presentation makes impossible the usage of basic Armenian characters [1] without special driver and application level support. Suggested protocols support adequate basic text processing - text data input, text data display, case change, and collating - for Western and Eastern Armenian users. Classic Armenian text processing doesn’t require special driver or application level support.

Suggested protocol solves Sign ‘Yev’ and digraph ‘U’ interpretation problem [1] for both Western and Eastern Armenian.

I. Text Types

Let’s introduce the following types of texts:

1) Classic Armenian Text (CAT) - a text which has letters of Classic Armenian Alphabet (characters 1 - 76) [1], Armenian punctuation signs (77 - 97), and doesn’t use the character 98;
2) Western Armenian Text (WAT) - a text which has letters of Classic Armenian Alphabet (characters 1 - 76), Armenian punctuation signs (77 - 97), Sign ‘Yev’ (character 98), and may not have a sequence of ‘Yech’ + ‘Vyun’;
3) Eastern Armenian Text (EAT) - a text which has letters of Classic Armenian Alphabet (characters 1 - 76) except the letter ‘Vyun’ (characters 67 - 68), Armenian punctuation signs (77 - 97), Sign ‘Yev’ (character 98), and digraph ‘U’ (letter ‘Vyun’ appears only as a part of the digraph);
4) Scientific (Linguistic) Text (SLT) - a text that has all 1 - 98 characters with no restriction.

Note that every text conforming to Classic Armenian orthography is a CAT. It is obvious that not every CAT conforms to Classic Armenian orthography. The same statements are true for WAT and EAT and respective orthographies. Western Armenian orthography allows for using sequence of Small Letter ‘Yech’ and Small Letter ‘Vyun’ instead of the Sign ‘Yev’. Hence, a text conforming to Western Armenian orthography may be a CAT. SLT is not in the scope of the document (though a brief description of possible support is made).

II. Display Buffer Support

The category of the Sign ‘Yev’ is Small Letter and it doesn’t have a Capital Letter counterpart [1]. In Western Armenian orthography it is always interpreted as a shorthand notation for the sequence of Small Letter ‘Yech’ and Small Letter ‘Vyun’. Hence, it gets capitalized into that pair of letters. In Western Armenian orthography either the sequence of Small Letter ‘Yech’ and Small Letter ‘Vyun’ or Sign ‘Yev’ may be used (not both at the same time). In Eastern Armenian orthography Sign ‘Yev’ is always interpreted as a shorthand notation for the sequence of Small Letter ‘Yech’ and Small Letter ‘Vev’. It gets capitalized into that pair of letters. However, not every sequence of Small Letter ‘Yech’ and Small Letter ‘Vev’ is a Sign ‘Yev’ (this is the case for complex words when the first ends with ‘Yech’ and the second starts with the letter ‘Vev’).

To distinguish the ways of treating the Sign ‘Yev’ let’s introduce two virtual letters: Western Armenian Letter ‘YechVyun’ and Eastern Armenian Letter ‘YechVev’. These letters have two capital forms: title and capital [2].

Correct processing of the Sign ‘Yev’ - the letters ‘YechVyun’ and ‘YechVev’ support - is based upon the idea of different representations of the sign in Text and Display buffers. The Text buffer is a buffer where actual textual data is kept for processing while the Display buffer is a buffer where the image of visible part of the text is created.

In Text buffer the Sign ‘Yev’ is always represented with the pair (sequence) of Small Letter ‘Yech’ and Small Letter ‘Vyun’. In Display buffer the Sign ‘Yev’ itself is sent. The Table below shows the content of Text and Display buffers for Western and Eastern Armenian during Sign ‘Yev’ case change.

	Text Buffer	Display Buffer
Small ‘YechVyun’	Small ‘Yech’+Small ‘Vyun’	Sign ‘Yev’
Title ‘YechVyun’	Capital ‘Yech’+Small ‘Vyun’	Capital ‘Yech’+Small ‘Vyun’
Capital ‘YechVyun’	Capital ‘Yech’+Capital ‘Vyun’	Capital ‘Yech’+Capital ‘Vyun’
Small ‘YechVev’	Small ‘Yech’+Small ‘Vyun’	Sign ‘Yev’
Title ‘YechVev’	Capital ‘Yech’+Small ‘Vyun’	Capital ‘Yech’+Small ‘Vev’
Capital ‘YechVev’	Capital ‘Yech’+Capital ‘Vyun’	Capital ‘Yech’+ Capital ‘Vev’

The accent signs [1] in Text buffer always go next to the letter ‘Yech’. For small letters ‘YechVyun’ and ‘YechVev’ the accent sign goes next to the Sign ‘Yev’ in Display buffer.

Display driver must support Text Buffer content to the Display buffer 1). unchanged copy mode. It may also support 2) virtual Letter ‘YechVyun’ mode or 3) virtual Letter ‘YechVev’ mode or both (obviously not at the same time).

III. Keyboard Support

Keyboard may support virtual letters ‘YechVyun’, ‘YechVev’, and ‘U’ input and handling. If the keyboard has separate buttons for Sign ‘Yev’ and digraph ‘U’ then the following rules apply:

- “Sign ‘Yev’” button push inputs the sequence of Small ‘Yech’+Small ‘Vyun’;
- “Shift” + “Sign ‘Yev’” button push inputs the sequence of Capital ‘Yech’+Small ‘Vyun’;
- If “Caps Lock” on and “Sign ‘Yev’” button is pushed then the sequence of Capital ‘Yech’+ Capital ‘Vyun’ is entered;
- “Digraph ‘U’” button push inputs the sequence of Small ‘Vo’+Small ‘Vyun’;
- “Shift” + “Digraph ‘U’” button push inputs the sequence of Capital ‘Vo’+Small ‘Vyun’;
- If “Caps Lock” on and “Digraph ‘U’” button is pushed then the sequence of Capital ‘Vo’+ Capital ‘Vyun’ is entered.

All data is entered into Text buffer. Display buffer behaves according to its own settings (see Section II, modes 1 to 3).

Keyboard navigation and deletion buttons must start routines that consider these three virtual letters as one unit of processing:

- “Right Arrow” button must pass both characters of virtual letter and position the cursor at the character next to trailing ‘Vyun’;
- “Left Arrow” button must pass both characters of virtual letter and position the cursor at the character in front of starting ‘Yech’ or ‘Vo’;
- “Delete” button push at the first character of virtual letter (‘Yech’ or ‘Vo’) must delete both characters: starting ‘Yech’ or ‘Vo’ and trailing ‘Vyun’;
- “Backspace” button push at the next to virtual letter character must delete both characters: trailing ‘Vyun’ and starting ‘Yech’ or ‘Vo’.

In this mode of keyboard operation the cursor can never be at trailing ‘Vyun’ position of the virtual letters.

IV. Console Driver support

The console driver by default doesn’t support any virtual letter. This is the CAT mode. Thus, the mode supports Classic Armenian text input. In this mode it may support virtual letter buttons (to input two characters by one button push), but it must not support any specific cursor navigation or Display buffer handling. In this mode at user option it may input code value of Sign ‘Yev’ into Text buffer (e.g. then the user types “Alt” + “Sign ‘Yev’” or via preset parameters).

The driver may support two more modes: WAT mode and/or EAT mode. In WAT mode it supports virtual Letter ‘YechVyun’ input (and navigation) and Display. In EAT mode it supports virtual Letter ‘YechVev’ and virtual Letter ‘U’ input (and navigation) and Display of the Letter ‘YechVev’.

Accent application to virtual letters needs special handling too. As far as virtual letter is a unit of processing accent characters cannot be applied in between two characters. In most cases (except Sign ‘Yev’) it must appear that way [1] though.

The user enters accent character next to virtual letter. Console driver places accent character next to the first character of virtual letter in Text buffer. In case of Small Letter ‘YechVyun’ or Small Letter ‘YechVev’ the sequence of Sign ‘Yev’ and the accent character is sent to Display buffer. Console driver treats accented virtual letters as one unit of processing. The user cannot delete accent or virtual letter separately. Even in case of accented Sign ‘Yev’ an attempt to delete only the Sign ‘Yev’ with “Delete” button or only the accent mark with “Backspace” the whole combination of letter and accent disappears. This makes perception of accented virtual letter different from regular.

V. Text Externalization

The technique for handling different Armenian texts in essence is based upon storing the CAT in Text buffer and simulating (or presenting) WAT and EAT. The presentation is done on console driver level. The reason for this is that the CAT has all information for correct case change, collating, sorting, and representing the text as WAT or EAT. WAT can be always converted into CAT while EAT cannot. It is impossible without semantic analysis of the word (or table lookup) due to Sign ‘Yev’ interpretation ambiguity [1]. That’s why CAT is recommended as normal (standard) form of any Armenian text representation.

3 different formats for text externalization may be supported by higher level software applications (editors, word processors, network communication, etc.):

- normalized - CAT;
- semi-normalized - WAT;
- de-normalized - EAT.

Normalized Armenian text must be used for persistence (files, databases, etc.) and network communication. Semi-normalized Armenian text may also be used for persistence, network communication as well because it can be transformed to normalized without loss of information. Internalization routine (e.g. reading from the ‘stream’) must normalize the text: convert the Sign ‘Yev’ to the sequence of Small ‘Yech and Small Vyun’. Semi-normalized text may be used for Western Armenian text externalization to Display or to Printer buffer. De-normalized text must be used for Eastern Armenian text externalization to Display or to Printer buffer. It may not be used for persistence or interprocess communication.

VI. Software Application Operating Modes

Armenian text processing application - for example, a text editor - can be designed to support both: West and East Armenian users. The following is the list and the description of all reasonable modes for such application:

1) CAT-1 (mode name)

- Usage (dialects/orthographies): Classic and West Armenian.
- Keyboard: has characters 1 - 97 [1];
- Unit of processing (input, navigation, highlighting increment, deletion): character;
- Externalization (persistent storage, network communication): normalized;
- Visualization (display buffer, printer): normalized;
- Collation: Classic [1]

2) CAT-2

- Usage: Classic and West Armenian.
- Keyboard: has characters 1 - 98;
- Unit of processing: character (exception: inputs two characters for Sign 'Yev');
- Externalization: normalized;
- Visualization: normalized;
- Collation: Classic.

3) WAT

- Usage: West Armenian.
- Keyboard: has characters 1 - 98;
- Unit of processing: character and virtual Letter 'YechVyun';
- Externalization: normalized (semi-normalized);
- Visualization: semi-normalized;
- Collation: Classic.

4) EAT

- Usage: East Armenian.
- Keyboard: has characters 1 - 99 (except Letter 'Vyun');
- Unit of processing: character and virtual letters 'YechVev' and 'U';
- Externalization: normalized (semi-normalized);
- Visualization: de-normalized;
- Collation: East Armenian (the pair 'Yech' + 'Vyun' is treated as 'Yech' + 'Vev', the pair 'Vo' + 'Vyun' - as 'Vyun' [1]).

For all above-mentioned modes the content of Text Buffer is normalized and the case change is conducted in the Text Buffer according to the Classic Alphabet [1]. Collation is also conducted upon Text Buffer content. Internalization routine for all above-mentioned modes may [must] normalize external text though it makes sense for semi-normalized text input. For normalized text it becomes null (dummy) operation.

5) SLT

- Usage: Scientific (Linguistic).
- Keyboard: has characters 1 - 98;
- Unit of processing: one character;
- Externalization: semi-normalized,
- Internalization: semi-normalized (no transformation of Sign 'Yev');
- Visualization: semi-normalized;

- Collation: Special West Armenia (Sign ‘Yev’ is small ‘YechVyun’) or Special East Armenian (Sign ‘Yev’ is small ‘YechVev’ and the pair ‘Vo’ + ‘Vyun’ is ‘Vyun’);
- Case change: Sign ‘Yev’ remains unchanged.

For all above-mentioned modes Search/Replace is conducted upon the content of Text Buffer. Search/Replace user interface must be implemented in Text/Display buffer paradigm with the corresponding (to the settings) rules applied.

VII. References

1. RFC EN-001. Armenian Character Names, Descriptions, Classification, Collating, and Searching.
2. The Unicode Standard Version 3.0. Addison-Wesley. Reading, MA. Jan 2000.