Title: Conversion of Cyrillic script to Score with SipXML2Score
Author: Jan de Kloe
Version: 2.00
Date: June 28th, 2003, last updated January 24, 2007

## Scope

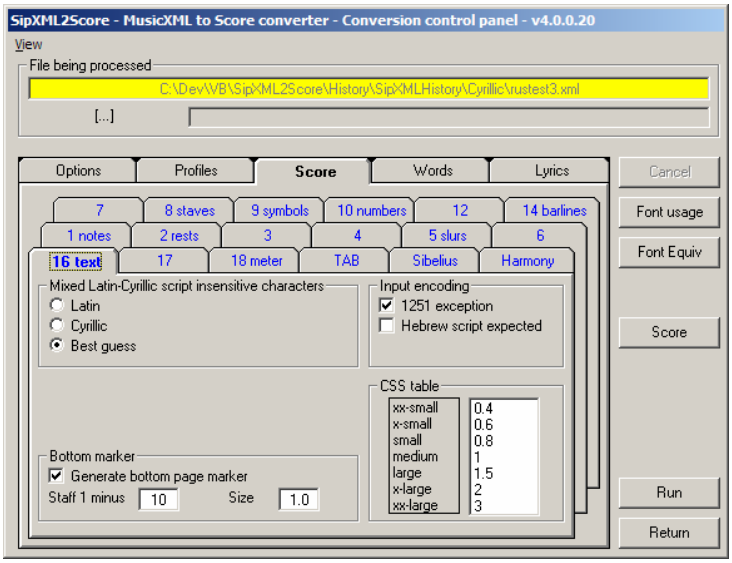
There is no limitation in MusicXML to the encoding of characters which means that the Latin alphabet with all its specific accents and variations as well as Cyrillic script can be stored. Theoretically, Hebrew, Greek, Arabic, and other scripts can be encoded but the current version of SipXML2Score only processes Latin, Cyrillic and Hebrew as these constitute the primary market for the converter to Score.
Only the characters used in modern Russian are subject of this article. There are many other languages which use Cyrillic with additional characters not covered here.
This chapter explains various aspects of encoding, conversion, and fonts. Experience with MusicXML is currently limited to Dolet output.
As the user base interested in Cyrillic is limited, the specific issues are separately documented here rather than in the User Guide.

## File format

Files written in MusicXML come in various formats. SipXML2Score supports UTF-8 and UTF-16. Both formats can be created by Dolet. UTF-8 uses from 1 to 6 bytes for each character while UTF-16 uses two bytes for each character. The standard for representation of characters in MusicXML is called Unicode. For the engraver, it is not necessary to know about the details of these encoding standards.

## Code page 1251

Early versions of Dolet only produced Code page 1251 format for Cyrillic. Even the current combination of FinWin2006c.r1 Dolet3.4 erroneously generates 1251 and this is currently being investigated by Recordare. It is the intention of Recordare to drop non-Unicode encoding.
The converter attempts to differentiate between Unicode and 1251 and needs a little help from the user: In the 'Text' panel, the switch needs to be set if 1251 encoding is still being used as shown here:




## Input character encoding

To obtain a maximum of flexibility in the conversion of characters, equivalencies are defined in a table outside the program. The character equivalence table comes with the program.
In that table one finds the internal values which can occur in an input file and the characters to which it converts. The character equivalence file is named SIPXML16.TXT and comes with converter.
Though the character equivalence table can be maintained by the user, this is discouraged.

## *Fonts*

Any font of which there is a .PSC file (part of the Score font library) in your LIB directory can be selected. It is not possible to select an undefined font because SipXML2Score needs to consults the .PSC files for character sizes when centering or justifying.
Alphabetic information in music is either words, lyrics, harmony, or fingering.
The font equivalence table defines which XML fonts correspond to which Score fonts.
To find out which fonts occur in the input, use the 'font usage' button.
Loading of fonts to the printer is the responsibility of the user.

**Lyrics**
Font selection of lyrics for Latin and Cyrillic scripts is done by definition of fonts in the font equivalence table where the font name must be 'lyric'. There can be only one target font per script. So when XML uses Arial for lyrics and both Latin and Cyrillic refrains occur, there are two entries in this table (one L line and one C line) equating this to Helvetica as Score target font.

**Words**
For other text (in MusicXML the term 'words' is used for text which is not lyrics), any Score font can be used. Again, when XML uses a single font for multiple scripts, multiple equivalences need to be defined as in Score there is no single font with multiple scripts.

**Harmony**
The term 'harmony' in MusicXML is used to name chords such as in guitar accompaniment. For harmony, only a Latin font should be selected since it is not subject to conversion.

**Fingering**
Fingering is converted to Score numbers (Code10 items). Any font can be selected to be generated as Code10Par07.

**Mixed scripts**
It is entirely possible to have single text items with multiple scripts. These are converted correctly as Score allows change of font within a text. However, when an item has mixed scripts and there are characters for which the font cannot be established (such as a space or a period) the user needs to specify his preference (either one or the other, or the converter's best guess).
Note that in mixed situations, the equivalence table can specify different target sizes. Score does not support different sizes in a single text item, so the converter take the size based on the number of characters of either script.

## *Refrains*

The vertical position of a refrain cannot be deducted from the input and is therefor assumed. MusicXML contains a refrain number but it is not necessarily 1, 2, 3, etc. but can be 3, 6, 9, etc.
The user has the option to pre-scan the input to find out which refrains occur and then modify the assumed vertical position.

## *Cyrillic fonts families*

Score has two families of Cyrillic fonts that I know of (ScoreCyr and TimeScore) and both are supported by SipXML2Score. The internal values of the two families are however different so the target values are also defined in the equivalence file. By selecting a font, the user automatically selects the proper conversion.
As there are two families to be supported, and since we must leave the liberty of font names and Score font numbers to the user, there is a file which defines the families. Currently, the contents of this text file (which can be maintained via the View menu of the font equivalence table) looks like this:

```
* Font family table.
* Field1 - digit indicating the family.
* Field2 - font name as it appears in the PCS file.
* Use SipAnlib to find out the correct spelling.
* The fields need to be separated by commas.
* Records starting with an asterisk are comment lines.
*
2,ScoreCyr
3,TimeScore
```

```
3,TimeScore-Italic
3,TimeScore-Bold
3,TimeScore-BoldItalic
3,HelveCyr
3,HelveCyr-Bold
3,HelveCyr-Italic
3,HelveCyr-BoldItalic
```

With SipEdit, font prefixes can be changed in a series of MUS files in a matter of seconds.

## *Cyrillic font ScoreCyr*

This font was obtained from Matanya Ophee of Editions Orphée (Columbus, Ohio) and contains the following characters:

ScoreCyr character set

| | |
|---|---|
| Lowercase Russian alphabet | абвгдежзийклмнопрстуфхцчшщъыьэюя ё |
| Uppercase Russian alphabet | АБВГДЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ Ё |
| Digits | 0123456789 |
| Non-alphabetic characters | ( ) [ ] ! ? ; : & (not all shown) |

The internal values of this alphabet are based on the Russian keyboard layout. Inputting Cyrillic is not necessary when converting from an external source but in case you want to enter characters in Score and you do not have such keyboard, use the equivalence table information to type Cyrillic. For example:

```
Unicode=x410,C"F", C"A" 'uppercase a
```

means that to make Cyrillic ‘uppercase a’ you need to type an ‘F’. Take the ScoreCyr value from the second argument of the Unicode equivalent definition.

Note that this set misses common characters such as comma, period, etc. When encountered, the converter automatically switches to the alternate Latin font, so even when the music does not contain Latin text, a font should be selected. When not, then the program defaults to Times Roman. So assuming the Score fonts 49 and 00, the Russian string “собьюсь.” will be converted to “_49cj,m.cm_00.” and in Score screen show as “cj,m.cm.”.

There is no bold, italic, or bold-italic version of this font.

## *Cyrillic font family TimeScore*

The font family was downloaded from the Score site years ago and was made available by Sergey Lebedev of the Moscow Conservatory. There is a shadow copy on my web-site.
The set is shown here in its basic form, the variants bold, italic, and bold-italic are not shown.

| TimeScore character set | |
|---|---|
| Lowercase Russian alphabet | абвгдежзийклмнопрстуфхцчшщъыьэюя ё |
| Uppercase Russian alphabet | АБВГДЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ Ё |
| Digits | 0123456789 |
| Non-alphabetic characters | . , ( ) : ; ? ! + - * = № " [ ] { } – (not all shown) |

Internal values of this alphabet are based on phonetic or visual resemblance to Latin script and therefor easier if you need to input them with a Latin keyboard. For example:

```
Unicode=x416,C":", C"W" 'uppercase zhe
```

means that to make Cyrillic 'Ж' you need to type a 'W'. Take the TimeScore value from the third argument of the Unicode equivalent definition. Assuming the fonts, the Russian string "собьюсь." will be converted to "_50sob!h?ts!h." and in Score screen shows as "sobXXsX." because the equivalents for "!h" (right guillemet) representing "ь" and "?t" (trade mark) representing "ю" cannot be visualized.

Note that there are a few special cases here:
- the character '/' exists in the font and needs to be coded as '\'. The PDF however does not show it.
- the apostrophe exists in the font and needs to be coded as '!8'. The PDF will show 'э'.
- character 'Ё' is a little smaller than the other capitals. It hardly occurs as a capital though.

This family of fonts also has all vowels with the acute diacritical as required in ethnomusicology. Since these do not exist in MusicXML as single characters ('precomposed') but can be made by combining letters and diacritical marks ('composite sequences', the marks follow the character), it should be possible to foresee them in the conversion but until samples and requests come along this is not yet supported.

Besides the family of four fonts, this collection contains font TimesScoreAcc. This is a font with only three letters which are not part of Cyrillic but occur in some Balkan languages which use the Latin alphabet (ISO Latin2). They are 'Č' (capital C hacek), 'č' (lowercase c hacek), and 'ć' (lowercase c acute).

## *Editing Cyrillic*

Cyrillic text can be edited by the utility SipText provided its Cyrillic option is installed.

## *Transliteration*

SipXML2Score also contains a phonetic conversion module which is used to communicate to the user. Music output will never have phonetic output from this routine.
Phonetic conversion occurs in two situations:
- when the user needs to define a vertical value for a refrain, he needs to know which refrain it is. By clicking on the vertical position in the list generated by pre-scan ('set refrain levels'), the transliteration is given if the refrain was in Cyrillic.
- the summary report contains verses assembled from the input if there were any lyrics. Cyrillic input is transliterated.

Transliteration is also defined in the equivalence table. So while the transliteration table is delivered according to the Modified Library of Congress standard, a non-English user could select to modify this table to get Tschaikowski (German) or Rajmaninof (Spanish).

## *Fonts in this document*

For Russian characters (other than in the two figures), this document uses the 'Times New Roman' character set. It is an extended version different from the standard which comes with WindowsXP.

### *Music symbols in MusicXML*

The Unicode standard also defines codes for music symbols but these have not been encountered in any of the files which I have at my disposal. SipXML2Score cannot yet handle them.


### *Acknowledgment*

Without the help of the following people, this information could not have been complete: Tom Brodhead, Sergey Lebedev, Timo Leskelä, and Matanya Ophee.