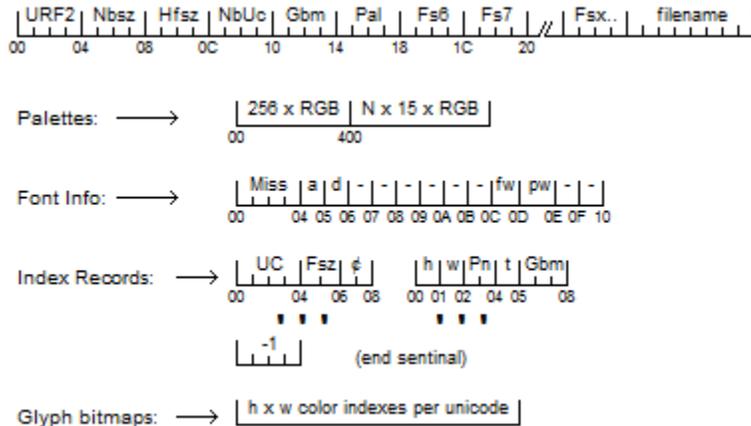## File Format layout for the URF2 Unicode Raster File

Edition: 23.12.2014

The labels in the layout have no other meaning then to describe the fields in this document.
Unless noted otherwise, numeric fields are coded in little endian.
Offsets are specified in hexadecimal.

Header fields:

URF2   A 4 byte string to identify the contents of the file.
         In memory, these 4 bytes serve as a place holder to link the data of several font files.
Nbsz   The number of font sizes in this file.
         For example, the k2r font has 10 font sizes.
Hfsz   Highest font size. For example, the highest font size in the k2r font is 20.
NbUc   Total number of unicodes for all fontsizes in this fontfile..
         For example, the k2r font contains 24848 tokens in total.
         Note that the various fontsizes may have a different number of unicodes.
Gbm   Glyph bitmap : byte offset from the beginning of the file to the glyph bitmaps.
Pal   Palette : byte offset from the beginning of the file to the 8-bit palette (palette nb. 0).
Fs6   Fs6 to Fsx are 4-byte fields for the fontsizes 6 up to the highest implemented fontsize (Hfsz).
         Each field contains the offset from the beginning of the file to the corresponding index records.
filename A zstring with the lowercase name of this fontfile (preceded with a length byte).
         For example, the name of the k2r font file is "k2r.urf".
         The value of the Hfsz field multiplied by four gives the offset of this field.

Font Info:

There is a font info record for every font size. It is located 16 bytes below the index records.

Miss   Missing character: the unicode of the suggested glyph to represent a missing character.
         For example, in the k2r font this is the unicode 0x25A1, representing a simple rectangle.
a   common ascent (pixels) for all the glyphs of this fontsize.
d   common descent (pixels) for all the glyphs of this fontsize.
fw   suggested minimal width for mono-spaced fonts.
pw   suggested width for a proportional space character.

Index Records:

The 8-byte index records are grouped per unicode group, preceded by a 8-byte group header.

UC   the first unicode of the group.
Fsz   the font size
¢   the number of subsequent unicodes in this group

h   height (in pixels) of the glyph bitmap
w   width (in pixels) of the glyph bitmap
Pn   Palette number (0=8-bit palette)
t   top (in pixels) : the number of background pixels to draw above the glyph.
gbm   glyph bitmap: offset from the beginning of the file to the glyph bitmap for this unicode character.

General description

A URF2 font file has raster data for the glyphs for several font sizes of a same font face.
For example, currently the k2a font has 10 fontsizes for a Kolibri sans serif bold italic font .
For every fontsize, a header specifies common font information such as glyph ascent and descent.
The file uses 4-byte unicodes, but the routines that use the font may be limited to handle 2-byte unicodes only.
The URF2 font file can be seen as a collection of glyph bitmaps per unicode character.
For every unicode, there is an 8-byte index record that specifies the size, type, and location of that glyph's bitmap.
The user must find this record amongst a series of records grouped in ranges of consecutive unicodes.
The fontfile has no data for none printable characters such as spaces and control characters.
The index records specify the glyph's bitmap location, it's palette, and it's width, height, and top-margin in pixels.
The palettes have 32-bit fields to specify the colour of the subpixels, using inverted 8-bit RGB values.
If the palette number is zero, then the bitmap contains 8-bit indexes in a palette of 256 RGB colours.
Otherwise, the bitmap contains 4-bit indexes into the specified palette of 14 unique colours.
The 4-bit values zero and 0x0F are used to represent the white and black colour.
The 4-bit palettes contain 14 RGB colours plus 4 zero bytes.
The 4-bit palettes (of 15x4=60 bytes each) are located behind the 8-bit palette of 1024 bytes.
A simple compression mechanism is used to avoid many zeroes in the indexes.
A value of zero marks the beginning of a sequence of one to 255 zeroes, as specified by the next value.

A note about fontsizes:

In URF2 font files, the fontsize corresponds to the pixel-height of the Latin capital letter "E".
Thus, disregarding the physical size of the pixel on the display device.
Note: for a popular 96 dpi display, the pixel-size is the point-size x 96/72.

A note about anti-aliasing:

In the font file, glyph data is present as black text on white background.
The colour of the sub-pixels is specified as direct RGB values (0 to 0xFF per colour).
An intermediate buffer may be required if the display device does not support direct colours (3x8 or 4x8).
Colour blending must be applied to produce coloured or transparent text.

A note about glyph widths:

The width of each glyph is proportional to the character width in accordance with the font height.
The user should add at least one pixel between the glyphs for proper readability.
To represent mono-spaced text, the user could center the glyphs in a fixed advance width.
Perfect centring can not be obtained because of the presence of sub-pixels.
Note that there are many unicodes that are very large.