

Additions to oxs format for Cross Stitch Pro Platinum

<Palette>

According to the standard OXS file specification, a thread colour has a number that includes the colour range prefix (for example “DMC 310”).

Cross Stitch Pro maintains a default colour range for a design and then only adds a prefix if a colour is from a different range. It will however add the prefix to all colours when saving an oxs file to retain compatibility with other software. In order to better re-load designs created in its own software, some optional extra parameters for the palette have been added:

Filename Allows the filename of the colour range file used as the default colour range for the design. By setting this, Cross Stitch Pro can load back the oxs format design and set the correct colour range even when multiple colour ranges on the system use the same short name prefix. Other software vendors could use this to present a choice of possible colour ranges when loading the design or could try to interpret the Cross Stitch Pro filename to a suitable range in their own software. Note that Cross Stitch Pro allows users to create there own ranges so it may not be set to a standard name.

shortname Allows the short name prefix used for colours in the default colour range to be specified. If the colour name of a thread uses this default prefix then Cross Stitch Pro will remove it from the colour name when loading the oxs file an the colour will be assumed to come from the designs default colour range.

<Palette>

Palette_item gets additional properties:

fontname If not specified then Cross Stitch Pro will assume its default symbol font. It supports up to 3 different fonts in a design and uses the normal symbol sequence number to specify the character starting with symbol=0 for ASCII character 32 (space).

metallic Set to “true” if the color represents a metallic thread.

fluorescent Set to “true” if the color represents a fluorescent thread.

locked Set to “true” if the color has been locked by the user to prevent color changes (Cross Stitch Pro also uses this flag when editing a layered object to stop palette changes from affecting the base level design)

symbolcolor rgb hex value for the font color of the symbol. Since this is a Windows color there is no cmyk version.

Example:

```
<palette_item index="1" strands="2" bsstrands="1" number="761" name=" Sunrise rose"
color="F5A3A8" bscolor="000000" printcolor="000000" symbol="1" fontname="Cross Stitch Pro
Platinum" metallic="true" fluorescent="false"/>
```

There are optional properties to specify CMYK hex colors. The standard RGB colors should always be present but for programs that support CMYK colors, it is possible to define the exact CMYK value. These are important to publishers who can specify a CMYK value in the thread color ranges

and know that their final exported color layers will correctly represent the color defined. There is nothing to be gained by converting RGB to CMYK in order to add these palette definitions so programs that cannot specify and store CMYK in the program should not add these parameters.

```
colorcmyk="F5A3A855"  
bscolorcmyk="F5A3A855"  
printcolorcmyk="F5A3A855"
```

<partstitches>

partstitch gets an optional **major** property. This is used to indicate if a $\frac{3}{4}$ stitch should show as equal parts or if one part should be shown larger than the other. If major=0 or the parameter is not present then the parts will be equal. If major=1 then palindex1 is the larger half of the stitch. If major=2 then palindex2 is the larger half of the stitch.

Example:

```
<partstitch x="8" y="5" palindex1="1" palindex2="0" major="1" direction="1"/>
```

<ornaments_inc_knots_and_beads>

A new objecttype='tent' is added that represents a diagonal stitch top left to bottom right or top right to bottom left.

Direction=1 represents \

Direction=2 represents /

Example:

```
< objecttype="tent" x1="7" y1="26" palindex="2" direction="1"/>
```

objecttype='quarter' gets an optional property **petit** which can be “true” or “false”. A value of true specifies that programs able to differentiate between $\frac{1}{4}$ stitches and petit-point should treat the stitch as petit-point. Cross Stitch Pro treats each square in the design as either regular stitches (cross, $\frac{1}{4}$, $\frac{3}{4}$, tent) or petit-point (petit, half cross vertical or horizontal, off grid cross). It is not possible to mix stitches from both groups in a single grid square. If the property is missing then Cross Stitch Pro treats the stitch as a standard $\frac{1}{4}$ stitch.

Example:

```
< objecttype="quarter" petit="true" x1="23.5" y1="26" palindex="2"/>
```

<backstitches>

The backstitch object gets an optional property **thickness** which multiplies the number of strands by the thickness when it would otherwise be the default number of strands. If there is a non-default strand value set for the palette entry then this property is ignored.

Example:

```
< x1="15" y1="28" x2="21" y2="32" palindex="4" objecttype="backstitch" thickness="1"/>
```

Note: Cross Stitch Pro will process backstitches at the time of printing or exporting to a flat design by chopping them into single stitch lengths, removing hidden lines, aligning directions and chaining them into continuous straight lengths. This means it will not respect a sequence number assigned to a backstitch when opening an oxs file but will read them in the order they appear in the file. When saving to oxs as a flat (non layered) design, Cross Stitch Pro will assign an ascending sequence number which represents the order in which they were added to the design but it does not require or use this information.

<schemes> (occurs once)

This is a new section that allows different color schemes to be defined for a design. Each color scheme is contained in a scheme_item. Each scheme_item contains a new palette, name of the scheme and name of the color range.

When Cross Stitch Pro adds an alternative color scheme it always adds a master scheme plus the new alternative. So either there are no schemes or there are at least 2. The main palette is the one currently in view and the master and alternates are the ones available for selection.

<scheme_item> (occurs once for each color scheme as child of <schemes>)

- name - for example "My alternate color scheme" as shown in the design software.
- rangename – for example "dmc" or "anchor tapestry" used to show entire range.
In Cross Stitch Pro, the extension .clr is added to this name to get the filename of the thread definitions. These files could be made by the user so cannot be guaranteed standard.
- <palette> Child same as <palette> in the main chart containing palette_item's

The following information for layered designs is not yet implemented

<layers> (occurs once)

This can be ignored by software that only supports a single layer because the standard stitches should always contain the entire design on a single layer. Software like Cross Stitch Pro that supports design elements in multiple layers can instead ignore all the standard stitches and load each of the layers separately. The receiving software must be able to process the layered objects to remove hidden and partially hidden stitches.

<layered_item> (occurs once per layer as child of <layers>)

- name (should be "base" for the main design)
- x1 left side of object (eg 20.5)
- y1 top side of object
- x2 right side of object
- y2 bottom side of object
- objecttype (eg "motif" for a design within a design, "text" for text to stitches)
- <fullstitches> (top left of object is x=0 y=0) - only present for motifs
- <partstitches> - only present for motifs
- <backstitches> - only present for motifs
- <ornaments_inc_knots_and_beads> - only present for motifs
- text - "a string" - only present for text to stitches object
- fontname - name of font for text to stitches
- fontsize - size of font for text to stitches
- italic - true if font is italic
- bold - true if font is bold
- paletteindex - for text to stitches only
- priority - Lowest priority drawn first

Future object types based on current Cross Stitch Pro features

Publisher objects are not stitches but chart markup features

publisher rectangle
publisher ellipse
publisher line
publisher rounded rectangle
publisher text (rtf)
publisher fixed key
complex stitch (made in CAD drawing tools)
publisher wmf graphic
Woven bar
Kloster block
Spiders web
chart page definition
shape to stitches
font letter definition (for backstitch fonts)

Variiegated thread colors
Fade
print color type
line styles

<xsp_print>

A new section of “chart” to store print settings for Cross Stitch Professional Platinum. If none are specified then default settings are used.

pageindex="0" (the tab page of the print dialog that is currently selected)

area_to_print="autocrop" (can be autocrop, all, key, manual or predefined)

xstart="0" (x starting point of print when area is set to manual)

ystart="0" (y starting point of print when area is set to manual)

xend="500" (x ending point of print when area is set to manual)

ystart="500" (y ending point of print when area is set to manual)

singlepagenum="" (if printing a single page, this is the page number)

ptitle="Design title" (The title of the design to appear on the print)

originiscenter='true' (true if the design is numbered from 0,0 at the center)

designername="D Peters" (the name of the designer – often includes copyright)

spi="14.0" (number of stitches per unit)

spiunits="inch" (units for the SPI can be inch, cm, 2.5mm, 2.8mm)

chart_orientation="portrait" (can be portrait or landscape)

rectangular_grid="false" (true if grid is rectangular to match aspect ratio of stitches)

simple_key="own_sheet" (if a simple short key is used, can be each_sheet, own_sheet or none)

simple_key_title="My key" (the title for the short key if used);

key_orientation="portrait" (page orientation for the key sheet, can be portrait or landscape)

cover_orientation="portrait" (page orientation for the cover sheet, can be portrait or landscape)

font_name="Arial" (name of the default font for text on the chart)

font_size="12" (size of default text on chart)

font_colour="000000" (color of default text on the chart as a hex RGB value)
font_bold="false" (the default chart text is in bold if set to true)
font_italic="false" (the default chart text is in italic if set to true)

shownames="true" (the thread identification includes color names)
shownumbers="true" (the thread identification includes color numbers)
backstitch_key="true" (the simple key includes a separate key for backstitch)
superkey="true" (there will be a separate key sheet based on the designs key template)
patchkey="true" (a simple key of colour patches will be printed)
photo_key="false" (special private customer feature to use photos on the key)
bmpsize="20" (photo key bitmap width and height)

numbers_font_size="12" (font size for grid numbers)
numbers_font_colour="000000" (RGB hex string for color of grid numbers)
numbers_font_bold="true" (grid numbers are in bold when set to true)
numbers_font_italic="true" (grid numbers are in italic when set to true)
numbers_font_name="Arial" (font name for grid numbers)
numbers_background_style="none" (can be none white_circle or outlined_white_circle)

pages_font_size="12" (font size for page numbers)
pages_font_colour="000000" (RGB hex string for color of page numbers)
pages_font_bold="false" (page numbers are bold)
pages_font_italic="false" (page numbers are italic)
pages_font_name="Arial" (font name for page numbers)
designer_font_size="12" (font size for designers name)
designer_font_colour="000000" (RGB hex string for color of designers name)
designer_font_bold="false" (designers name is bold)
designer_font_italic="false" (designers name is italic)
designer_font_name="Arial" (font name for designers name)

title_font_size="12" (font size for title)
title_font_colour="000000" (RGB hex string for color of title)
title_font_bold="false" (title is bold)
title_font_italic="false" (title is italic)
title_font_name="Arial" (font name for title)

```
'title_relativex'inttostr(title_relativex);  
'title_justifyx'inttostr(title_justifyx);  
'title_relativey'inttostr(title_relativey);  
'title_justifyy'inttostr(title_justifyy);  
'title_adjustx'floattostr(title_adjustx);  
'title_adjustx_units'inttostr(title_adjustx_units);  
'title_adjusty'floattostr(title_adjusty);  
'title_adjusty_units'inttostr(title_adjusty_units);
```

```
'designer_relativex'inttostr(designer_relativex);  
'designer_justifyx'inttostr(designer_justifyx);  
'designer_relativey'inttostr(designer_relativey);  
'designer_justifyy'inttostr(designer_justifyy);  
'designer_adjustx'floattostr(designer_adjustx);  
'designer_adjustx_units'inttostr(designer_adjustx_units);
```

'designer_adjusty'floattostr(designer_adjusty);
'designer_adjusty_units'inttostr(designer_adjusty_units);

'page_relativex'inttostr(page_relativex);
'page_justifyx'inttostr(page_justifyx);
'page_relativey'inttostr(page_relativey);
'page_justifyy'inttostr(page_justifyy);
'page_adjustx'floattostr(page_adjustx);
'page_adjustx_units'inttostr(page_adjustx_units);
'page_adjusty'floattostr(page_adjusty);
'page_adjusty_units'inttostr(page_adjusty_units);

'gridnum_adjustx'floattostr(gridnum_adjustx);
'gridnum_adjustx_units'inttostr(gridnum_adjustx_units);
'gridnum_adjusty'floattostr(gridnum_adjusty);
'gridnum_adjusty_units'inttostr(gridnum_adjusty_units);
'gridnum_adjustxvert'floattostr(gridnum_adjustxvert);
'gridnum_adjustx_unitsvert'inttostr(gridnum_adjustx_unitsvert);
'gridnum_adjustyvert'floattostr(gridnum_adjustyvert);
'gridnum_adjusty_unitsvert'inttostr(gridnum_adjusty_unitsvert);

'center_adjustx'floattostr(center_adjustx);
'center_adjustx_units'inttostr(center_adjustx_units);
'center_adjusty'floattostr(center_adjusty);
'center_adjusty_units'inttostr(center_adjusty_units);
'center_adjustxvert'floattostr(center_adjustxvert);
'center_adjustx_unitsvert'inttostr(center_adjustx_unitsvert);
'center_adjustyvert'floattostr(center_adjustyvert);
'center_adjusty_unitsvert'inttostr(center_adjusty_unitsvert);
'center_style'inttostr(center_style);
'xtragrid.top'inttostr(xtragrid.top);
'xtragrid.bottom'inttostr(xtragrid.bottom);
'xtragrid.left'inttostr(xtragrid.left);
'xtragrid.right'inttostr(xtragrid.right);

'chart2'inttostr(chart2);
'singlebackthick'booleantostring(singlebackthick);
{ tabbednotebook1.pageindex:=2;}
'symbolscale'inttostr(symbolscale);
'quarterscale'inttostr(quarterscale);
'pagetitle'utf8encode(pagetitle);
'symspacing'inttostr(symspacing);
'linewidth'inttostr(linewidth);
'longwidth'inttostr(longwidth);
'knotsize'inttostr(knotsize);
'beadsize'inttostr(beadsize);
'majorgrid'inttostr(majorgrid);
'middlegrid'inttostr(middlegrid);
'minorgrid'inttostr(minorgrid);
'bordergrid'inttostr(bordergrid);
'bordergap'floattostr(bordergap);
'centergrid'inttostr(centergrid);

```
'scale'intostr(scale);
'majorcolour'cmyktohexstring(majorcolour);
'minorcolour'cmyktohexstring(minorcolour);
'middlecolour'cmyktohexstring(middlecolour);;
'bordercolour'cmyktohexstring(bordercolour);
'centercolour'cmyktohexstring(centercolour);
'available'intostr(available);
```

```
{ tabbednotebook1.pageindex:=3;}
'psymbol'booleantostring(psymbol);
'colsyms'booleantostring(colsyms);
'pcolour'booleantostring(pcolour);
'pcrosses'booleantostring(pcrosses);
'pboth'booleantostring(pboth);
'cutouts'booleantostring(cutouts);
'backstyle'intostr(backstyle);
{ specialcolour:=specialcolour;}
'specialcolour'booleantostring(specialcolour);
'specialstyle'intostr(specialstyle);
'longstyle'intostr(longstyle);
'knotstyle'intostr(knotstyle);
'beadstyle'intostr(beadstyle);
'halfstyle'intostr(halfstyle);
```

```
'hardcolour'booleantostring(hardcolour);
'crossscheme'intostr(crossscheme);
'backscheme'intostr(backscheme);
'knotscheme'intostr(knotscheme);
'beadscheme'intostr(beadscheme);
'specialscheme'intostr(specialscheme);
'publisherscheme'intostr(publisherscheme);
'longscheme'intostr(longscheme);
'halfscheme'intostr(halfscheme);
'allscheme'booleantostring(allscheme);
'movecenter'booleantostring(movecenter);
'white_symbols_on_dark_colors'booleantostring(white);
'intensity'intostr(intensity);
'backstyle'intostr(backstyle);
if backstyle=1 then backcolour:=true;
if backstyle>2 then linesblack:=true; { default old settings to black for load into old version }
'outlines'booleantostring(outlines);
'colourtogether'booleantostring(colourtogether);
'blacksilk'booleantostring(blacksilk);
'coloursilk'booleantostring(coloursilk);
'symbolsilk'booleantostring(symbolsilk);
'silkgap'intostr(silkgap);
'silkoverlap'floattostr(silkoverlap);
'circulargrid'booleantostring(circulargrid);
'blankcircular'booleantostring(blankcircular);
```

```
{ tabbednotebook1.pageindex:=4;}
'cmark'booleantostring(cmark);
```

```

'gridon'booleanstring(gridon);
'pagenumber'booleanstring(pagenumber);
'separator'booleanstring(separator);
'separatorq'booleanstring(separatorq);
'shade34'booleanstring(shade34);
'separators'booleanstring(separators);
'plastic'booleanstring(plastic);
'gridbehind'booleanstring(gridbehind);
'title1'booleanstring(title1);
'titleall'booleanstring(titleall);
'designerbox'booleanstring(designerbox);
'designerall'booleanstring(designerall);
'notes'booleanstring(notes);
'printlengths'booleanstring(printlengths);
'cover'booleanstring(cover);
'pagemap'booleanstring(pagemap);
'singlepage'booleanstring(singlepage);
'pback1'booleanstring(pback1);
'pcons'booleanstring(pcons);
'pimage'booleanstring(pimage);
'pallother1'booleanstring(pallother1);
{ tabbednotebook1.pageindex:=5;}
'margintop'inttostr(margintop);
'marginbottom'inttostr(marginbottom);
'marginleft'inttostr(marginleft);
'marginright'inttostr(marginright);
if portrait_notes then 'notes_orientation'"portrait'
  else 'notes_orientation'"landscape';

'center_page'booleanstring(center_page);
'dual_overlap'booleanstring(dual_overlap);
'alt_single_overlap'booleanstring(alt_single_overlap);
'shadeblank'booleanstring(shadeblank);
'symformat'inttostr(symformat);
'backformat'inttostr(backformat);
'overlap_style'inttostr(ord(overlap_style));
'overlap_color'cmyktohexstring(overlap_color);
'overlap_stitches'inttostr(overlap_stitches);
'break_major'booleanstring(break_major);
'numbers'booleanstring(numbers);
'numberlines'booleanstring(numberlines);
'squashed'booleanstring(squashed);

'petitareasize'inttostr(petitareasize);
'petit_shade_colour'cmyktohexstring(petit_shade_colour);
'petit_shade_style'inttostr(ord(petit_shade_Style));
'petit_area_title'petit_area_title;
'petit_area_suffix'inttostr(petit_area_suffix);
'petit_scale'floattostr(petit_scale);

```