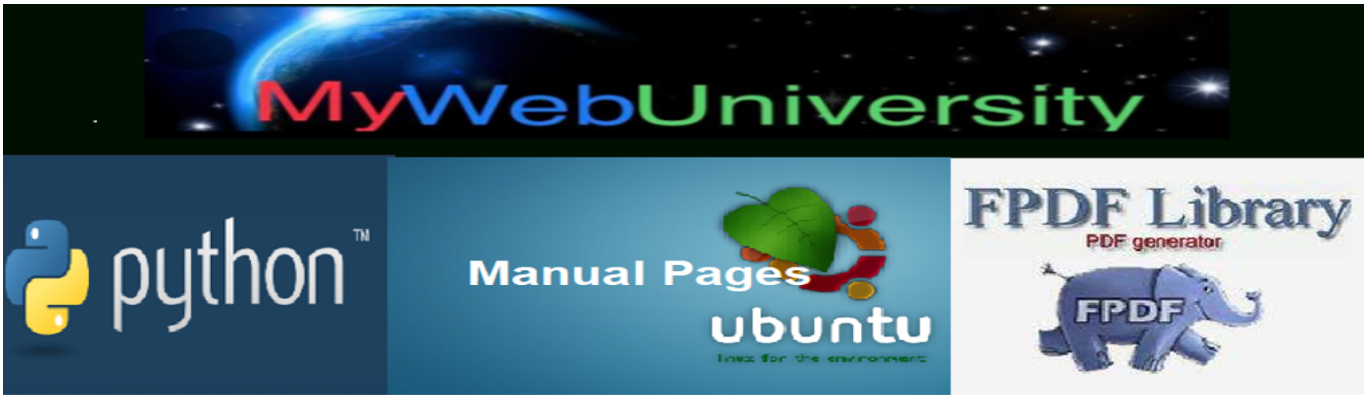

Rocky Enterprise Linux 9.2 Manual Pages on command 'X11::Protocol::Ext::RENDER.3pm'

$ man X11::Protocol::Ext::RENDER.3pm

Protocol::Ext::RENDER(3pm) User Contributed Perl Documentation Protocol::Ext::RENDER(3pm)

NAME

X11::Protocol::Ext::RENDER - Perl module for the X Rendering Extension

SYNOPSIS

```
use X11::Protocol;
$x = X11::Protocol->new($ENV{'DISPLAY'});
$x->init_extension('RENDER') or die;
```

DESCRIPTION

The RENDER extension adds a new set of drawing primitives which effectively represent a replacement of the drawing routines in the core protocol, redesigned based on the needs of more modern clients. It adds long-desired features such as subpixel positioning, alpha compositing, direct specification of colors, and multicolored or animated cursors. On the other hand, it omits features that are no longer commonly used: wide lines, arbitrary polygons (only triangles and horizontally-aligned trapezoids are supported), ellipses, bitwise rendering operations, and server-side fonts (in favor of "glyphs" that are rendered on the client side and transmitted once).

As of this writing (early 2004), the specification and implementation both have rough edges, but there are relatively few alternatives for offloading fancy graphics processing to the server, as is necessary over slow links or if the client is written in a slow language. Another possibility you might consider is the 2D subset of OpenGL, though it doesn't yet have an X11::Protocol-compatible interface.

SYMBOLIC CONSTANTS

This extension adds the constant types 'PictType', 'PictOp', 'SubPixel', 'PolyEdge', and

'PolyMode', with values as defined in the standard.

REQUESTS

This extension adds several requests, called as shown below:

```
$x->RenderQueryVersion($major, $minor)
=>
($major, $minor)
$x->RenderQueryPictFormats()
=>
([[$id, $type, $depth,
   $red, $red_m, $green, $green_m, $blue, $blue_m,
   $alpha, $alpha_m, $cmap], ...],
 [[$fallback, [$depth, [$visual, $format], ...], ...], ...],
 [$subpixel, ...])
$x->RenderQueryPictIndexValues($pict_format)
=>
([$index, $red, $green, $blue, $alpha], ...)
$x->RenderQueryFilters($drawable)
=>
([@filters], [@aliases])
$x->RenderCreatePicture($picture, $drawable, $format,
                        'attribute' => $value, ...)
$x->RenderChangePicture($picture, 'attribute' => $value, ...)
$x->RenderSetPictureClipRectangles($pic, $x_origin, $y_origin,
                                   [$x, $y, $width, $height], ...)
$x->RenderSetPictureTransform($pict, $m11, $m12, $m13,
                                $m21, $m22, $m23,
                                $m31, $m32, $m33);
$x->RenderSetPictureFilter($pict, $filter, @args)
$x->RenderComposite($op, $src, $mask, $dst, $src_x, $src_y,
                    $mask_x, $mask_y, $dst_x, $dst_y,
                    $width, $height)
$x->RenderFillRectangles($op, $dst, [$red, $green, $blue, $alpha],
                         [$x, $y, $width, $height], ...)
```

```
$x->RenderTrapezoids($op, $src, $src_x, $src_y, $dst, $mask_format,
                     [$top, $bottom, $lx1, $ly1, $lx2, $ly2,
                                $rx1, $ry1, $rx2, $ry2] ,...)
$x->RenderTriangles($op, $src, $src_x, $src_y, $dst, $mask_format,
                    [$x1, $y1, $x2, $y2, $x3, $y3])
$x->RenderTriStrip($op, $src, $src_x, $src_y, $dst, $mask_format,
                    [$x, $y], [$x, $y], [$x, $y], [$x, $y], ...)
$x->RenderTriFan($op, $src, $src_x, $src_y, $dst, $mask_format,
                  [$x, $y], [$x, $y], [$x, $y], [$x, $y], ...)
$x->RenderCreateGlyphSet($gsid, $format)
$x->RenderReferenceGlyphSet($gsid, $existing)
$x->RenderFreeGlyphSet($gsid)
$x->RenderAddGlyphs($gsid, [$glyph, $width, $height,
                            $x, $y, $x_off, $y_off, $data], ...)
```

Warning: with some server implementations (including XFree86 through 4.4) passing more than one glyph to AddGlyphs can hang or crash the server. So don't do that.

```
$x->RenderFreeGylphs($gsid, @glyphs)
$x->RenderCompositeGlyphs8($op, $src, $dst, $mask_format, $gsid,
                           $src_x, $src_y,
                           [$delta_x, $delta_y, $str], ...)
$x->RenderCompositeGlyphs16($op, $src, $dst, $mask_format, $gsid,
                            $src_x, $src_y,
                            [$delta_x, $delta_y, $str], ...)
$x->RenderCompositeGlyphs32($op, $src, $dst, $mask_format, $gsid,
                            $src_x, $src_y,
                            [$delta_x, $delta_y, $str], ...)
```

In these three requests, new GlyphSetIDs can also be interspersed with the array references.

```
$x->RenderCreateCursor($cid, $source, $hot_x, $hot_y)
$x->RenderCreateAnimCursor($cid, [$cursor, $delay], ...)
```

AUTHOR

Stephen McCamant <SMCCAM@cpan.org>.

SEE ALSO

perl(1), X11::Protocol, The X Rendering Extension (XFree86 draft standard).