NAME

Xft − X FreeType interface library

DESCRIPTION

Xft is a simple library designed to interface the FreeType rasterizer with the X Rendering Extension. This manual page barely scratches the surface of this library.

HEADER FILE

#include <X11/Xft/Xft.h>

CONSTANTS

XFT_MAJOR
is the major version number of **Xft**.

XFT_MINOR
is the minor version number of **Xft**.

XFT_REVISION
is the revision number of **Xft**.

XFT_VERSION
is **XFT_MAJOR** times 10000 (ten thousand), plus **XFT_MINOR** times 100, plus **XFT_REVISION**.

XftVersion
is an alias for **XFT_VERSION**.

The following example illustrates how **Xft**'s version constants might be used:

```
#if (XFT_VERSION >= 20107)
(void) puts("Version 2.1.7 or later of the Xft library is in"
      " use.");
#else
(void) printf("Insufficient version of Xft (%d.%d.%d) installed;
       " need at least version 2.1.7.\n", XFT_MAJOR,
       XFT_MINOR,
       XFT_REVISION);
#endif
```

DATA TYPES

XftFont

```
typedef struct _XftFont {
    int        ascent;
    int        descent;
    int        height;
    int        max_advance_width;
    FcCharSet   *charset;
    FcPattern   *pattern;
} XftFont;
```

An **XftFont** is the primary data structure of interest to programmers using **Xft**; it contains general font metrics and pointers to the Fontconfig character set and pattern associated with the font. The **FcCharSet** and **FcPattern** data types are defined by the Fontconfig library.

XftFonts are populated with any of **XftFontOpen**(), **XftFontOpenName**(), **XftFontOpenXlfd**(), **XftFontOpenInfo**(), or **XftFontOpenPattern**(). **XftFontCopy**() is used to duplicate **XftFont**s, and **XftFontClose**() is used to mark an **XftFont** as unused. **XftFont**s are internally allocated, reference-counted, and freed by **Xft**; the programmer does not ordinarily need to allocate or free storage for them.

XftDrawGlyphs(), the **XftDrawString***() family, **XftDrawCharSpec**(), and **XftDrawGlyphSpec**() use **XftFont**s to render text to an **XftDraw** object, which may correspond to either a core X drawable or an X Rendering Extension drawable.

XftGlyphExtents() and the **XftTextExtents***() family are used to determine the extents (maximum dimensions) of an **XftFont**.

An **XftFont**'s glyph or character coverage can be determined with **XftFontCheckGlyph**() or **XftCharExists**(). **XftCharIndex**() returns the **XftFont**-specific character index corresponding to a given Unicode codepoint.

XftGlyphRender(), **XftGlyphSpecRender**(), **XftCharSpecRender**(), and the **XftTextRender***() family use **XftFont**s to draw into X Rendering Extension **Picture** structures. **Note: XftDrawGlyphs**(), the **XftDrawString***() family, **XftDrawCharSpec**(), and **XftDrawGlyphSpec**() provide a means of rendering fonts that is independent of the availability of the X Rendering Extension on the X server.

XftFontInfo

is an opaque object that stores information about a font. **XftF ontInfo** structures are created with **XftFontInfoCreate**(), freed with **XftFontInfoDestroy**(), and compared with **XftFontInfoEqual**(). **XftFontInfo** objects are internally allocated and freed by **Xft**; the programmer does not ordinarily need to allocate or free storage for them.

Each **XftFontInfo** structure in use is associated with a unique identifier, which can be retrieved with **XftFontInfoHash**(). An**XftF ont** can be opened based on **XftFontInfo** data with **XftFontOpenInfo**().

XftColor

```
typedef struct _XftColor {
    unsigned long   pixel;
    XRenderColor    color;
} XftColor;
```

An **XftColor** object permits text and other items to be rendered in a particular color (or the closest approximation offered by the X visual in use). The **XRenderColor** data type is defined by the X Render Extension library.

XftColorAllocName() and **XftColorAllocValue**() request a color allocation from the X server (if necessary) and initialize the members of **XftColor**. **XftColorFree**() instructs the X server to free the color currently allocated for an **XftColor**.

One an **XftColor** has been initialized, **XftDrawSrcPicture**(), **XftDrawGlyphs**(), the **XftDrawString***() family, **XftDrawCharSpec**(), **XftDrawCharFontSpec**(), **XftDrawGlyphSpec**(), **XftDrawGlyphFontSpec**(), and **XftDrawRect**() may be used to draw various objects using it.

XftDraw

is an opaque object which holds information used to render to an X drawable using either the core protocol or the X Rendering extension.

XftDraw objects are created with any of **XftDrawCreate**() (which associates an **XftDraw** with an existing X drawable), **XftDrawCreateBitmap**(), or **XftDrawCreateAlpha**(), and destroyed with **XftDrawDestroy**(). The X drawable associated with an **XftDraw** can be changed with **XftDrawChange**(). **XftDraw**s are internally allocated and freed by **Xft**; the programmer does not ordinarily need to allocate or free storage for them.

The X **Display**, **Drawable**, **Colormap**, and **Visual** of an **XftDraw** can be queried with **XftDrawDisplay**(), **XftDrawDrawable**(), **XftDrawColormap**(), and **XftDrawVisual**(), respectively. The X Rendering Extension **Picture** associated with an **XftDraw** is returned by **XftDrawPicture**().

XftCharSpec

```
typedef struct _XftCharSpec {
    FcChar32    ucs4;
    short       x;
    short       y;
} XftCharSpec;
```

The **FcChar32** data type is defined by the Fontconfig library.

XftCharFontSpec

```
typedef struct _XftCharFontSpec {
    XftFont     *font;
    FcChar32    ucs4;
    short       x;
    short       y;
} XftCharFontSpec;
```

The **FcChar32** data type is defined by the Fontconfig library.

XftGlyphSpec

```
typedef struct _XftGlyphSpec {
    FT_UInt     glyph;
    short       x;
    short       y;
} XftGlyphSpec;
```

The **FT_UInt** data type is defined by the FreeType library.

XftGlyphFontSpec

```
typedef struct _XftGlyphFontSpec {
    XftFont     *font;
    FT_UInt     glyph;
    short       x;
    short       y;
} XftGlyphFontSpec;
```

The **FT_UInt** data type is defined by the FreeType library.

FUNCTIONS

Opening and Matching Fonts

XftFont *
XftFontOpen (Display ****dpy*,**
int *screen***,**
...);

XftFontOpen takes a list of pattern element triples of the form *field*, *type*, *value* (terminated with a NULL), matches that pattern against the available fonts, and opens the matching font, sizing it correctly for screen number *screen* on display *dpy*. The**Display** data type is defined by the X11 library . Returns NULL if no match is found.

Example:

```
    font = XftFontOpen (dpy, screen,
                XFT_FAMILY, XftTypeString, "charter",
                XFT_SIZE, XftTypeDouble, 12.0,
                NULL);
```

This opens the “charter” font at 12 points. The point size is automatically converted to the correct pixel size based on the resolution of the monitor.

XftFont *
XftFontOpenName (Display ****dpy*,**
int *screen***,**
unsigned char ****name*);**

XftFontOpenName behaves as **XftFontOpen** does, except that it takes a Fontconfig pattern string (which is passed to the Fontconfig library’s **FcNameParse**() function).

XftFont *
XftFontOpenXlfd (Display ****dpy*,**
int *screen***,**

unsigned char ****xlfd*)**

XftFontOpenXlfd behaves as **XftFontOpen** does, except that it takes a string containing an X Logical Font Description (XLFD).

FcPattern *
XftFontMatch (Display ****dpy*,**
int *screen***,**
FcPattern ****pattern*,**
FcResult ****result*);**

Also used internally by the **XftFontOpen*** functions, **XftFontMatch** can also be used directly to determine the Fontconfig font pattern resulting from an Xft font open request. The **FcPattern** and **FcResult** data types are defined by the Fontconfig library.

Determining the Pixel Extents of a Text String

void
XftTextExtents8 (Display ****dpy*,**
XftFont ****font*,**
FcChar8 ****string*,**
int *len***,**
XGlyphInfo ****extents*);**

XftTextExtents8 computes the pixel extents on display *dpy* of no more than *len* glyphs of a *string* consisting of eight-bit characters when drawn with *font*, storing them in *extents*. The**FcChar8** data type is defined by the Fontconfig library, and the **XGlyphInfo** data type is defined by the X Rendering Extension library.

void
XftTextExtents16 (Display ****dpy*,**
XftFont ****font*,**
FcChar16 ****string*,**
int *len***,**
XGlyphInfo ****extents*);**

XftTextExtents16 computes the pixel extents on display *dpy* of no more than *len* glyphs of a *string* consisting of sixteen-bit characters when drawn with *font*, storing them in *extents*. The**FcChar16** data type is defined by the Fontconfig library, and the **XGlyphInfo** data type is defined by the X Rendering Extension library.

void
XftTextExtents32 (Display ****dpy*,**
XftFont ****font*,**
FcChar32 ****string*,**
int *len***,**
XGlyphInfo ****extents*);**

XftTextExtents32 computes the pixel extents on display *dpy* of no more than *len* glyphs of a *string* consisting of thirty-two-bit characters when drawn with *font*, storing them in *extents*. The**FcChar32** data type is defined by the Fontconfig library, and the **XGlyphInfo** data type is defined by the X Rendering Extension library.

void
XftTextExtentsUtf8 (Display ****dpy*,**
XftFont ****font*,**
FcChar8 ****string*,**
int *len***,**
XGlyphInfo ****extents*);**

XftTextExtentsUtf8 computes the pixel extents on display *dpy* of no more than *len* bytes of a UTF-8 encoded *string* when drawn with *font*, storing them in *extents*. The**XGlyphInfo** data type is defined by the X Rendering Extension library.

void

XftTextExtentsUtf16 (Display ***dpy*****,**
XftFont ***font*****,**
FcChar8 ***string*****,**
FcEndian *endian***,**
int *len***,**
XGlyphInfo ****extents*);**

XftTextExtentsUtf16 computes the pixel extents on display *dpy* of no more than *len* bytes of a UTF-16LE- or UTF-16BE-encoded *string* when drawn with *font*, storing them in *e xtents*. The endianness of*string* must be specified in *endian*. The**FcEndian** data type is defined by the F ontconfig library, and the **XGlyphInfo** data type is defined by the X Rendering Extension library.

void
XftGlyphExtents (Display ***dpy*****,**
XftFont ***font*****,**
FT_UInt ***glyphs*****,**
int *nglyphs***,**
XGlyphInfo ****extents*);**

Also used internally by the **XftTextExtents*** functions, **XftGlyphExtents** computes the pixel extents on display *dpy* of no more than *nglyphs* in the array *glyphs* drawn with *font*, storing them in *e xtents*. The **FT_UInt** data type is defined by the FreeType library, and the **XGlyphInfo** data type is defined by the X Rendering Extension library.

Drawing Strings (and Other Things)

XftDraw *
XftDrawCreate (Display ***dpy*****,**
Drawable *drawable***,**
Visual ***visual*****,**
Colormap *colormap***);**

XftDrawCreate creates a structure that can be used to render text and rectangles using the specified *drawable*, *visual*, and *colormap* on *display*. The**Drawable**, **V isual**, and **Colormap** data types are defined by the X11 library.

XftDraw *
XftDrawCreateBitmap (Display ****dpy*,**
Pixmap *bitmap***);**

XftDrawCreateBitmap behaves as **XftDrawCreate**, except it uses an X pixmap of color depth 1 instead of an X drawable. The**Pixmap** data type is defined by the X11 library .

XftDraw *
XftDrawCreateAlpha (Display ****dpy*,**
Pixmap *pixmap***,**
int *depth***);**

XftDrawCreateAlpha behaves as **XftDrawCreate**, except it uses an X pixmap of color depth *depth* instead of an X drawable. The**Pixmap** data type is defined by the X11 library .

void
XftDrawChange (XftDraw ***draw*****,**
Drawable *drawable***);**

XftDrawChange changes the X drawable association of the existing Xft draw object *draw* from its current value to *drawable*.

Display *
XftDrawDisplay (XftDraw ****draw*);**

XftDrawDisplay returns a pointer to the display associated with the Xft draw object *draw*.

Drawable
XftDrawDrawable (XftDraw ****draw*);**

XftDrawDrawable returns the X drawable associated with the Xft draw object *draw*.

```
Colormap
XftDrawColormap (XftDraw *draw);
```

XftDrawColormap returns the colormap associated with the Xft draw object *draw*.

```
Visual *
XftDrawVisual (XftDraw *draw);
```

XftDrawVisual returns a pointer to the visual associated with the Xft draw object *draw*.

```
Picture
XftDrawPicture (XftDraw *draw);
```

XftDrawPicture returns the picture associated with the Xft draw object *draw*. If the the X server does not support the X Rendering Extension, 0 is returned.

```
Picture
XftDrawSrcPicture (XftDraw *draw,
            XftColor *color);
```

This function is never called if the X server doesn't support the X Rendering Extension; instead, **XftGlyphCore** is used.

```
void
XftDrawDestroy (XftDraw *draw);
```

XftDrawDestroy destroys *draw* (created by one of the **XftCreate** functions) and frees the memory that was allocated for it.

```
void
XftDrawString8 (XftDraw       *d,
      XftColor       *color,
      XftFont        *font,
      int            x,
      int            y,
      unsigned char  *string,
      int            len);
```

XftDrawString8 draws no more than *len* glyphs of *string* to Xft drawable *d* using *font* in *color* at position *x*, *y*.

```
void
XftDrawRect (XftDraw       *d,
     XftColor      *color,
     int           x,
     int           y,
     unsigned int  width,
     unsigned int  height);
```

XftDrawRect draws a solid rectangle of the specified *color*, *width*, and *height* at position *x*, *y* to Xft drawable *d*.

COMPATIBILITY

As of version 2, **Xft** has become relatively stable and is expected to retain source and binary compatibility in future releases.

Xft does provide a compatibility interface to its previous major version, Xft 1.*x*, described below.

Xft 1.x Compatibility Header File

#include <X11/Xft/XftCompat.h>

Xft 1.x Compatibility Data Types

XftPattern

holds a set of names with associated value lists; each name refers to a property of a font. **XftP attern**s are used as inputs to the matching code as well as holding information about specific fonts.

XftFontSet

contains a list of **XftPattern**s. Internally, **Xft** uses this data structure to hold sets of fonts. Externally, **Xft** returns the results of listing fonts in this format.

XftObjectSet

holds a set of names and is used to specify which fields from fonts are placed in the the list of returned patterns when listing fonts.

AUTHOR

Keith Packard

SEE ALSO

Fontconfig Developers Reference
FreeType API Reference
Xlib − C Language Interface