text-width *font sequence* &key (**:start** 0) **:end :translate** Function

Returns:
width — Type **int32**.
first-not-done — Type **array-index** or **null**.

Returns the total pixel width of the given *sequence* when drawn in the given *font*. The *font* can be a **gcontext**, in which case the font attribute of the given graphics context is used. **:start** and **:end** define the elements of the *sequence* which are used. The second value returned is **nil** if all elements of the *sequence* were successfully translated; otherwise the index of the first untranslated element is returned.

font — The font (or **gcontext**) used for measuring characters.

sequence — A sequence of characters or other objects to be translated into font indexes.

:start, **:end** — Start and end indexes defining the elements to draw.

:translate — A function to translate text to font indexes. Default is **#'translate-default**.

index — An **int16** font index.

char-width *font index* Function

Returns:
width — Type **int16** or **null**.

Returns the *width* of the given font glyph. The *width* is defined to be equal to (– *right-bearing left-bearing*). Returns **nil** if the *index* is invalid or specifies an empty glyph, or if the *font* is a pseudo-font.

font — A **font** object.

index — An **int16** font index.

Querying Text Size

8.6 CLX defines functions to return the size of text drawn in a specified font. See paragraph 6.7, Drawing Text, for a description of the **:translate** function used by the functions in the following paragraphs.

text-extents *font sequence* &key (**:start** 0) **:end :translate** Function

Returns:
width — Type **int32**.
ascent — Type **int16**.
descent — Type **int16**.
left — Type **int32**.
right — Type **int32**.
font-ascent — Type **int16**.
direction — Type **draw-direction**.
first-not-done — Type **array-index** or **null**.

Returns the complete geometry of the given *sequence* when drawn in the given *font*. The *font* can be a **gcontext**, in which case the font attribute of the given graphics context is used. **:start** and **:end** define the elements of the *sequence* which are used.

The returned *width* is the total pixel width of the translated character sequence. The returned *ascent* and *descent* give the vertical ascent and descent for characters in the translated *sequence*. The returned *left* gives the left bearing of the leftmost character. The returned *right* gives the right bearing of the rightmost character. The returned *font-ascent* and *font-descent* give the maximum vertical ascent and descent for all characters in the *font*. If **:translate** causes font changes, then *font-ascent* and *font-descent* will be the maximums over all fonts used. The *direction* returns the preferred draw direction for the font. If **:translate** causes font changes, then the *direction* will be **nil**. The *first-not-done* value returned is **nil** if all elements of the *sequence* were successfully translated; otherwise the index of the first untranslated element is returned.

font — The font (or **gcontext**) used for measuring characters.

sequence — A sequence of characters or other objects to be translated into font indexes.

:start, **:end** — Start and end indexes defining the elements to draw.

:translate — A function to translate text to font indexes. Default is **#'translate-default**.

Character Attributes

8.5 The following paragraphs describe the CLX functions used to access the attributes of individual font glyphs.

char-ascent *font index* Function

Returns:
ascent — Type **int16** or **null**.

Returns the vertical distance in pixels from the baseline to the top of the given font glyph. Returns **nil** if the index is invalid or specifies an empty glyph, or if the *font* is a pseudo-font.

font — A **font** object.

index — An **int16** font index.

char-attributes *font index* Function

Returns:
attributes — Type **int16** or **null**.

Returns font-specific *attributes* of the given glyph. The interpretation of such attributes is server-dependent. Returns **nil** if the *index* is invalid or specifies an empty glyph, or if the *font* is a pseudo-font.

font — A **font** object.

index — An **int16** font index.

char-descent *font index* Function

Returns:
descent — Type **int16** or **null**.

Returns the vertical distance in pixels from the baseline to the bottom of the given font glyph. Returns **nil** if the *index* is invalid or specifies an empty glyph, or if the *font* is a pseudo-font.

font — A **font** object.

index — An **int16** font index.

char-left-bearing *font index* Function

Returns:
left-bearing — Type **int16** or **null**.

Returns the left side bearing of the given font glyph. If **draw-glyph** is called with horizontal position *x*, the leftmost pixel of the glyph is drawn at the position (+ *x left-bearing*). Returns **nil** if the *index* is invalid or specifies an empty glyph, or if the *font* is a pseudo-font.

font — A **font** object.

index — An **int16** font index.

char-right-bearing *font index* Function

Returns:
right-bearing — Type **int16** or **null**.

Returns the right side bearing of the given font glyph. If **draw-glyph** is called with horizontal position *x*, the rightmost pixel of the glyph is drawn at the position (+ *x right-bearing*). Returns **nil** if the *index* is invalid or specifies an empty glyph, or if the *font* is a pseudo-font.

font — A *font* object.

font — A **font** object.

min-char-ascent *font* Function

Returns:
ascent — Type **int16**.

Returns the minimum **char-ascent** for all characters in *font*.

font — A **font** object.

min-char-attributes *font* Function

Returns:
attributes — Type **int16**.

Returns the minimum **char-attributes** for all characters in *font*.

font — A **font** object.

min-char-descent *font* Function

Returns:
descent — Type **int16**.

Returns the minimum **char-descent** for all characters in *font*.

font — A **font** object.

min-char-left-bearing *font* Function

Returns:
left-bearing — Type **int16**.

Returns the minimum **char-left-bearing** for all characters in *font*.

font — A **font** object.

min-char-right-bearing *font* Function

Returns:
right-bearing — Type **int16**.

Returns the minimum **char-right-bearing** for all characters in *font*.

font — A **font** object.

min-char-width *font* Function

Returns:
width — Type **int16**.

Returns the minimum **char-width** for all characters in *font*.

font — A **font** object.

font-properties *font* Function

Returns:
properties — Type **list**.

Returns the list of font *properties* recorded by the X server. The returned list is a property list of keyword/value pairs. The set of possible font property keywords is implementation-dependent.

font — A **font** object.

font-property *font name* Function

Returns:
property — Type **int32** or **null**.

Returns the value of the font *property* specified by the *name* keyword. The property value, if it exists, is returned as an uninterpreted 32-bit integer.

font — A **font** object.

name — A font property keyword.

max-char-ascent *font* Function

Returns:
ascent — Type **int16**.

Returns the maximum **char-ascent** value for all characters in *font*.

font — A **font** object.

max-char-attributes *font* Function

Returns:
attributes — Type **int16**.

Returns the maximum **char-attributes** value for all characters in *font*.

font — A **font** object.

max-char-descent *font* Function

Returns:
descent — Type **int16**.

Returns the maximum **char-descent** value for all characters in *font*.

font — A **font** object.

max-char-left-bearing *font* Function

Returns:
left-bearing — Type **int16**.

Returns the maximum **char-left-bearing** value for all characters in *font*.

font — A **font** object.

max-char-right-bearing *font* Function

Returns:
right-bearing — Type **int16**.

Returns the maximum **char-right-bearing** value for all characters in *font*.

font — A **font** object.

max-char-width *font* Function

Returns:
width — Type **int16**.

Returns the maximum **char-width** value for all characters in *font*.

font-max-char *font* Function

Returns:
index — Type **card16**.

Returns the maximum valid value used for linear encoded indexes. This function is not meaningful for fonts that use matrix index encoding.

font — A **font** object.

font-min-byte1 *font* Function

Returns:
min-byte1 — Type **card8**.

Returns zero if the *font* uses linear index encoding. Otherwise, if the *font* uses matrix index encoding, a value between 1 and 255 is returned that specifies the minimum value for the most significant byte of font indexes.

font — A **font** object.

font-min-byte2 *font* Function

Returns:
min-byte2 — Type **card8**.

Returns zero if the *font* uses linear index encoding. Otherwise, if the *font* uses matrix index encoding, a value between 1 and 255 is returned that specifies the minimum value for the least significant byte of font indexes.

font — A **font** object.

font-min-char *font* Function

Returns:
index — Type **card16**.

Returns the minimum valid value used for linear encoded indexes. This function is not meaningful for fonts that use matrix index encoding.

font — A **font** object.

font-name *font* Function

Returns:
name — Type **string** or **null**.

Returns the name of the *font*, or **nil** if *font* is a pseudo-font.

font — A **font** object.

font-p *font* Function

Returns:
font-p — Type **boolean**.

Returns true if the argument is a **font** object and **nil** otherwise.

font-plist *font* Function

Returns:
plist — Type **list**.

Returns and (with **setf**) sets the property list for the specified *font*. This function provides a hook where extensions can add data.

font — A **font** object.

font — A **font** object.

font-descent *font* Function

Returns:
descent — Type **int16**.

Returns the vertical *descent* of the *font* used for interline spacing. The *descent* defines the nominal distance in pixels from the baseline to the top of the next line of text. Some font glyphs may actually extend beyond the font *descent*.

font — A **font** object.

font-direction *font* Function

Returns:
direction — Type **draw-direction**.

Returns the nominal drawing *direction* for the *font*. The font drawing direction is only a hint that indicates whether the *char-width* of most font glyphs is positive (**:left-to-right** direction) or negative (**:right-to-left** direction). Note that X does not provide any direct support for vertical text.

font — A **font** object.

font-display *font* Function

Returns:
display — Type **display**.

Returns the **display** object associated with the specified *font*.

font — A **font** object.

font-equal *font-1 font-2* Function

Returns true if the two arguments refer to the same server resource and **nil** if they do not.

font-1, *font-2* — The **font** objects.

font-id *font* Function

Returns:
id — Type **resource-id**.

Returns the unique resource ID assigned to the specified *font*.

font — A **font** object.

font-max-byte1 *font* Function

Returns:
max-byte1 — Type **card8**.

Returns zero if the *font* uses linear index encoding. Otherwise, if the *font* uses matrix index encoding, a value between 1 and 255 is returned that specifies the maximum value for the most significant byte of font indexes.

font — A **font** object.

font-max-byte2 *font* Function

Returns:
max-byte2 — Type **card8**.

Returns zero if the *font* uses linear index encoding. Otherwise, if the *font* uses matrix index encoding, a value between 1 and 255 is returned that specifies the maximum value for the least significant byte of font indexes.

font — A **font** object.

list-font-names *display pattern* &key (**:max-fonts** 65535) (**:result-type 'list**) Function

Returns:
font-name — Type **sequence** of **string**.

Returns a sequence of strings containing the font names that match the *pattern*. The fonts available are determined by the font search path; see **font-path**). The maximum number of font names returned is determined by **:max-fonts**.

display — A **display** object.

pattern — A string used to match font names. Only font names that match the pattern are returned.

:max-fonts — The maximum number of font names returned. Default is 65535.

:result-type — The type of sequence to return. Default is '**list**.

list-fonts *display pattern* &key (**:max-fonts** 65535) (**:result-type 'list**) Function

Returns:
font — Type **sequence** of **font**.

Returns a sequence of pseudo-fonts corresponding to the available fonts whose names match the *pattern*. The fonts available are determined by the font search path; see **font-path**). The maximum number of **font** objects returned is determined by **:max-fonts**.

display — A **display** object.

pattern — A string used to match font names. Only fonts whose name matches the pattern are returned.

:max-fonts — The maximum number of fonts returned. Default is 65535.

:result-type — The type of sequence to return. Default is **'list**.

Font Attributes

8.4 The following paragraphs describe the CLX functions used to access font attributes.

font-all-chars-exist-p *font* Function

Returns:
exists-p — Type **boolean**.

Returns true if glyphs exist for all indexes in the range returned by **font-min-char** and **font-max-char**. Returns **nil** if an index in the range corresponds to empty glyph.

font — A **font** object.

font-ascent *font* Function

Returns:
ascent — Type **int16**.

Returns the vertical *ascent* of the *font* used for interline spacing. The *ascent* defines the nominal distance in pixels from the baseline to the bottom of the previous line of text. Some font glyphs may actually extend beyond the font *ascent*.

font — A **font** object.

font-default-char *font* Function

Returns:
index — Type **card16**.

Returns the *index* of the glyph drawn when an invalid or empty glyph index is specified. If the default index specifies an invalid or empty glyph, an invalid or empty index has no effect.

open-font *display name* Function

Returns:
font — Type **font**.

Opens the font with the given *name* and returns a **font** object. The name string should contain only ISO Latin-1 characters; case is not significant.

display — A **display** object.

name — A font name string.

close-font *font* Function

Deletes the association between the resource ID and the *font*. The *font* is freed when no other server resource references it. The *font* can be unloaded by the X server if this is the last reference to the *font* by any client. In any case, the *font* should never again be referenced because its resource ID is destroyed. This might not generate a protocol request if the *font* is reference-counted locally or if it is a pseudo-font.

font — A **font** object.

discard-font-info *fonts* Function

Discards any state that can be re-obtained with **open-font**. This is simply a performance hint for memory-limited systems.

font — A **font** object.

Listing Fonts

8.3 The following paragraphs describe CLX functions that return fonts or font names that match a given pattern string. Such pattern strings should contain only ISO Latin-1 characters; case is not significant. The following pattern characters can be used for *wildcard* matching:

#* — Matches any sequence of zero or more characters.

#\? — Matches any single character.

For example, the pattern "T?mes Roman" matches the name "Times Roman" but not the name "Thames Roman". However, the pattern "T*mes Roman" matches both names.

font-path *display* &key (**:result-type 'list**) Function

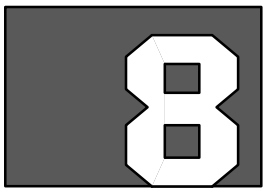
Returns:
paths — Type **sequence** of either **string** or **pathname**.

Returns a **list** (by default) of names containing the current search path for fonts. With **setf**, this function sets the search path for font lookup. There is only one search path per server, not one per client. The interpretation of the names is server-dependent, but they are intended to specify directories to be searched in the order listed.

Setting the path to the empty list restores the default path defined for the server. Note that as a side-effect of executing this request, the server is guaranteed to flush all cached information about fonts for which there are currently no explicit resource IDs allocated.

display — A **display** object.

:result-type — Specifies the type of resulting sequence.



FONTS AND CHARACTERS

Introduction

8.1 An X server maintains a set of fonts used in the text operations requested by client programs. An X font is an array of character bit maps (or *glyphs*) indexed by integer codes. In fact, font glyphs can also represent cursor shapes or other images and are not limited to character images. X supports both linear and matrix encoding of font indexes. With linear encoding, a font index is interpreted as a single 16-bit integer index into a one-dimensional array of glyphs. With matrix encoding, a font index is interpreted as a pair of 8-bit integer indexes into a two-dimensional array of glyphs. The type of index encoding used is font-dependent.

In order to access or use a font, a client program must first open it using the **open-font** function, sending a font name string as an identifier. **open-font** creates a CLX **font** object used to refer to the font in subsequent functions. Afterward, calling **open-font** with the same font name returns the same **font** object. When a font is no longer in use, a client program can call **close-font** to destroy the **font** object.

A font has several attributes which describe its geometry and its glyphs. CLX provides functions to return the attributes of a font, as well functions for accessing the attributes of individual font glyphs. Glyph attributes are referred to as *character attributes*, since characters are the most common type of font glyphs. A font also has a property list of values recorded by the X server. However, the set of possible font properties and their values are not standardized and are implementation-dependent. Typically, CLX maintains a cache of font and character attributes, in order to minimize server requests. However, the font cache mechanism is implementation-dependent and cannot be controlled by the client. In some cases, CLX may create a *pseudo-font* object solely for the purpose of accessing font attributes. A pseudo-font is represented by a special type of **font** object that cannot be used in a **gcontext**. If necessary, CLX can automatically convert a pseudo-font into a true font, if the name of the pseudo-font is known.

The set of available fonts is server-dependent; that is, font names are not guaranteed to be portable from one server to the next. However, the public X implementation from MIT includes a set of fonts that are typically available with most X servers.

The following paragraphs describe CLX functions to:

- Open and close fonts.
- List available fonts.
- Access font attributes.
- Access character attributes.
- Return the size of a text string.

Opening Fonts

8.2 The following paragraphs discuss the CLX functions for opening and closing fonts.