

XLookupKeysym, XRefreshKeyboardMapping, XLookupString, XRebindKeySym – handle keyboard input events in Latin-1

KeySym XLookupKeysym(*key_event*, *index*)

XKeyEvent **key_event*;
int *index*;

XRefreshKeyboardMapping(*event_map*)

XMappingEvent **event_map*;

int XLookupString(*event_struct*, *buffer_return*, *bytes_buffer*, *keysym_return*, *status_in_out*)

XKeyEvent **event_struct*;
char **buffer_return*;
int *bytes_buffer*;
KeySym **keysym_return*;
XComposeStatus **status_in_out*;

XRebindKeysym(*display*, *keysym*, *list*, *mod_count*, *string*, *num_bytes*)

Display **display*;
KeySym *keysym*;
KeySym *list*[];
int *mod_count*;
unsigned char **string*;
int *num_bytes*;

<i>buffer_return</i>	Returns the translated characters.
<i>bytes_buffer</i>	Specifies the length of the buffer. No more than <i>bytes_buffer</i> of translation are returned.
<i>num_bytes</i>	Specifies the number of bytes in the string argument.
<i>display</i>	Specifies the connection to the X server.
<i>event_map</i>	Specifies the mapping event that is to be used.
<i>event_struct</i>	Specifies the key event structure to be used. You can pass XKeyPressedEvent or XKeyReleasedEvent .
<i>index</i>	Specifies the index into the KeySyms list for the event's KeyCode.
<i>key_event</i>	Specifies the KeyPress or KeyRelease event.
<i>keysym</i>	Specifies the KeySym that is to be .
<i>keysym_return</i>	Returns the KeySym computed from the event if this argument is not NULL.
<i>list</i>	Specifies the KeySyms to be used as modifiers.
<i>mod_count</i>	Specifies the number of modifiers in the modifier list.
<i>status_in_out</i>	Specifies or returns the XComposeStatus structure or NULL.
<i>string</i>	Specifies the string that is copied and will be returned by XLookupString .

The **XLookupKeysym** function uses a given keyboard event and the index you specified to return the KeySym from the list that corresponds to the KeyCode member in the **XKeyPressedEvent** or **XKeyReleasedEvent** structure. If no KeySym is defined for the KeyCode of the event, **XLookupKeysym** returns **NoSymbol**.

The **XRefreshKeyboardMapping** function refreshes the stored modifier and keymap information. You usually call this function when a **MappingNotify** event with a request member of **MappingKeyboard** or **MappingModifier** occurs. The result is to update Xlib's knowledge of the keyboard.

The **XLookupString** function translates a key event to a KeySym and a string. The KeySym is obtained by using the standard interpretation of the **Shift**, **Lock**, group, and numlock modifiers as defined in the X Protocol specification. If the KeySym has been rebound (see **XRebindKeysym**), the bound string will be stored in the buffer. Otherwise, the KeySym is mapped, if possible, to an ISO Latin-1 character or (if the

Control modifier is on) to an ASCII control character, and that character is stored in the buffer. **XLookupString** returns the number of characters that are stored in the buffer.

If present (non-NULL), the **XComposeStatus** structure records the state, which is private to Xlib, that needs preservation across calls to **XLookupString** to implement compose processing. The creation of **XComposeStatus** structures is implementation-dependent; a portable program must pass NULL for this argument.

The **XRebindKeysym** function can be used to rebind the meaning of a KeySym for the client. It does not redefine any key in the X server but merely provides an easy way for long strings to be attached to keys. **XLookupString** returns this string when the appropriate set of modifier keys are pressed and when the KeySym would have been used for the translation. No text conversions are performed; the client is responsible for supplying appropriately encoded strings. Note that you can rebind a KeySym that may not exist.

XButtonEvent(3X11), XMapEvent(3X11), XStringToKeysym(3X11)

Xlib – C Language X Interface