XtSetKeyTranslator, XtTranslateKeycode, XtRegisterCaseConverter, XtConvertCase – convert KeySym to KeyCodes

void XtSetKeyTranslator(*display***,** *proc***)**
  Display ****display*;**
  XtKeyProc *proc***;**

void XtTranslateKeycode(*display*, *keycode*, *modifiers*, *modifiers_return*, *keysym_return*)
  Display **display*;
  KeyCode *keycode*;
  Modifiers *modifiers*;
  Modifiers **modifiers_return*;
  KeySym **keysym_return*;

void XtRegisterCaseConverter(*display*, *proc*, *start*, *stop*)
  Display **display*;
  XtCaseProc *proc*;
  KeySym *start*;
  KeySym *stop*;

void XtConvertCase(*display*, *keysym*, *lower_return*, *upper_return*)
  Display **display*;
  KeySym *keysym*;
  KeySym **lower_return*;
  KeySym **upper_return*;

display Specifies the display.

keycode Specifies the KeyCode to translate.

keysym Specifies the KeySym to convert.

keysym_return Returns the resulting KeySym.

lower_return Returns the lowercase equivalent of the KeySym.

upper_return Returns the uppercase equivalent of the KeySym.

modifiers Specifies the modifiers to the KeyCode.

modifiers_return Returns a mask that indicates the modifiers actually used to generate the KeySym.

proc Specifies the procedure that is to perform key translations or conversions.

start Specifies the first KeySym for which this converter is valid.

stop Specifies the last KeySym for which this converter is valid.

The XtSetKeyTranslator function sets the specified procedure as the current key translator. The default translator is **XtTranslateKey**, an **XtKeyProc** that uses Shift and Lock modifiers with the interpretations defined by the core protocol. It is provided so that new translators can call it to get default KeyCode-to-KeySym translations and so that the default translator can be reinstalled.

The **XtTranslateKeycode** function passes the specified arguments directly to the currently registered KeyCode to KeySym translator.

The **XtRegisterCaseConverter** registers the specified case converter. The start and stop arguments provide the inclusive range of KeySyms for which this converter is to be called. The new converter overrides any previous converters for KeySyms in that range. No interface exists to remove converters; you need to register an identity converter. When a new converter is registered, the Intrinsics refreshes the keyboard state if necessary. The default converter understands case conversion for all KeySyms defined in the core protocol.

The **XtConvertCase** function calls the appropriate converter and returns the results. A user-supplied **XtKeyProc** may need to use this function.

X Toolkit Intrinsics – C Language Interface
Xlib – C Language X Interface