NAME

XkbGetUpdatedMap – Update the client or server map information in an existing keyboard description

SYNOPSIS

Status XkbGetUpdatedMap (Display ****display*, unsigned int** *which***, XkbDescPtr** *xkb***);**

ARGUMENTS

display connection to X server

which mask selecting subcomponents to populate

xkb keyboard description to be updated

DESCRIPTION

The *which* parameter is a bitwise inclusive OR of the masks in Table 1. If the needed components of the *xkb* structure are not already allocated, *XkbGetUpdatedMap* allocates them. *XkbGetUpdatedMap* fetches the requested information for the device specified in the XkbDescRec passed in the *xkb* parameter.

Table 1 Xkb Mapping Component Masks
and Convenience Functions

Mask	Value	Map	Fields	Convenience Functions
XkbKeyTypesMask	(1<<0)	client	types	XkbGetKeyTypes
			size_types	XkbResizeKeyType
			num_types	XkbCopyKeyType
				XkbCopyKeyTypes
XkbKeySymsMask	(1<<1)	client	syms	XkbGetKeySyms
			size_syms	XkbResizeKeySyms
			num_syms	XkbChangeTypesOfKey
			key_sym_map	
XkbModifierMapMask	(1<<2)	client	modmap	XkbGetKeyModifierMap
XkbExplicitComponentsMask	(1<<3)	server	explicit	XkbGetKeyExplicitComp
XkbKeyActionsMask	(1<<4)	server	key_acts	XkbGetKeyActions
			acts	XkbResizeKeyActions
			num_acts	
			size_acts	
XkbKeyBehaviorsMask	(1<<5)	server	behaviors	XkbGetKeyBehaviors
XkbVirtualModsMask	(1<<6)	server	vmods	XkbGetVirtualMods
XkbVirtualModMapMask	(1<<7)	server	vmodmap	XkbGetVirtualModMap

XkbGetUpdatedMap is synchronous; it queries the server for the desired information, waits for a reply, and then returns. If successful, *XkbGetUpdatedMap* returns Success. If unsuccessful, *XkbGetUpdatedMap* returns one of the following: BadAlloc (unable to allocate a component in the XkbDescRec structure), BadValue (some mask bits in *which* are undefined), BadImplementation (a compatible version of the Xkb extension is not available in the server or the reply from the server was invalid).

RETURN VALUES

Success The *XkbGetUpdatedMap* function returns Success if a reply is received to the server query for the desired information.

STRUCTURES

The complete description of an Xkb keyboard is given by an XkbDescRec. The component structures in the XkbDescRec represent the major Xkb components.

```
typedef struct {
  struct _XDisplay * display;      /* connection to X server */
  unsigned short     flags;        /* private to Xkb, do not modify */
```

```
   unsigned short     device_spec;  /* device of interest */
   KeyCode            min_key_code; /* minimum keycode for device */
   KeyCode            max_key_code; /* maximum keycode for device */
   XkbControlsPtr     ctrls;        /* controls */
   XkbServerMapPtr    server;       /* server keymap */
   XkbClientMapPtr    map;          /* client keymap */
   XkbIndicatorPtr    indicators;   /* indicator map */
   XkbNamesPtr        names;        /* names for all components */
   XkbCompatMapPtr    compat;       /* compatibility map */
   XkbGeometryPtr     geom;         /* physical geometry of keyboard */
} XkbDescRec, *XkbDescPtr;
```

The *display* field points to an X display structure. The *flags* field is private to the library: modifying *flags* may yield unpredictable results. The *device_spec* field specifies the device identifier of the keyboard input device, or XkbUseCoreKeyboard, which specifies the core keyboard device. The *min_key_code* and *max_key_code* fields specify the least and greatest keycode that can be returned by the keyboard.

Each structure component has a corresponding mask bit that is used in function calls to indicate that the structure should be manipulated in some manner, such as allocating it or freeing it. These masks and their relationships to the fields in the XkbDescRec are shown in Table 2.

Table 2 Mask Bits for XkbDescRec

Mask Bit	XkbDescRec Field	Value
XkbControlsMask	ctrls	(1L<<0)
XkbServerMapMask	server	(1L<<1)
XkbIClientMapMask	map	(1L<<2)
XkbIndicatorMapMask	indicators	(1L<<3)
XkbNamesMask	names	(1L<<4)
XkbCompatMapMask	compat	(1L<<5)
XkbGeometryMask	geom	(1L<<6)
XkbAllComponentsMask	All Fields	(0x7f)

DIAGNOSTICS

BadAlloc Unable to allocate storage

BadImplementation
Invalid reply from server

BadValue An argument is out of range

SEE ALSO

XkbChangeTypesOfKey(3), **XkbCopyKeyType**(3), **XkbCopyKeyTypes**(3), **XkbGetKeyActions**(3), **XkbGetKeyBehaviors**(3), **XkbGetKeyExplicitComponents**(3), **XkbGetKeyModifierMap**(3), **XkbGetKeySyms**(3), **XkbGetKeyTypes**(3), **XkbResizeKeyActions**(3), **XkbResizeKeySyms**(3), **XkbResizeKeyType**(3), **XkbGetVirtualModMap**(3), **XkbGetVirtualMods**(3)