NAME

XkbFreeCompatMap - Free an entire compatibility map or selected portions of one

SYNOPSIS

void XkbFreeCompatMap (XkbDescPtr *xkb***, unsigned int** *which***, Bool** *free_map***);**

ARGUMENTS

xkb Xkb description in which to free compatibility map

which
mask of compatibility map components to free

free_map
True => free XkbCompatMap structure itself

DESCRIPTION

which specifies the compatibility map components to be freed (see XkbGetCompatMap). *which* is an inclusive OR of the bits shown in Table 1.

Table 1 Compatibility Map Component Masks

Mask	Value	Affecting
XkbSymInterpMask	(1<<0)	Symbol interpretations
XkbGroupCompatMask	(1<<1)	Group maps
XkbAllCompatMask	(0x3)	All compatibility map components

free_map indicates whether the XkbCompatMap structure itself should be freed. If *free_map* is True, *which* is ignored, all non-NULL compatibility map components are freed, and the *compat* field in the XkbDescRec referenced by *xkb* is set to NULL.

STRUCTURES

```
typedef struct _XkbCompatMapRec {
    XkbSymInterpretPtr    sym_interpret;            /* symbol based key semantics*/
    XkbModsRec            groups[XkbNumKbdGroups];  /* group => modifier map */
```

```
     unsigned short      num_si;              /* # structures used in sym_interpret */
     unsigned short      size_si;             /* # structures allocated in sym_interpret */
   } XkbCompatMapRec, *XkbCompatMapPtr;
```

The complete description of an Xkb keyboard is given by an XkbDescRec. The component structures in the XkbDescRec represent the major Xkb components outlined in Figure 1.1.

```
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. The other fields specify structure components of the keyboard description and are described in detail in other sections of this document. Table 2 identifies the subsequent sections of this document that discuss the individual components of the XkbDescRec.

Table 2 XkbDescRec Component References

XkbDescRec Field	For more info
ctrls	Chapter 10
server	Chapter

	16
map	Chapter 15
indicators	Chapter 8
names	Chapter 18
compat	Chapter 17
geom	Chapter 13

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 3.

Table 3 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)

SEE ALSO

XkbGetCompatMap(3)