NAME

XkbKeyAction - Returns the key action

SYNOPSIS

XkbAction XkbKeyAction (XkbDescPtr *xkb***, KeyCode** *keycode***, int** *idx***);**

ARGUMENTS

xkb Xkb description of interest

keycode
      keycode of interest

idx index for group and shift level

DESCRIPTION

A key action defines the effect key presses and releases have on the internal state of the server. For example, the expected key action associated with pressing the Shift key is to set the Shift modifier. There is zero or one key action associated with each keysym bound to each key.

Just as the entire list of key symbols for the keyboard mapping is held in the *syms* field of the client map, the entire list of key actions for the keyboard mapping is held in the *acts* array of the server map. The total size of *acts* is specified by *size_acts,* and the number of entries is specified by *num_acts.*

The *key_acts* array, indexed by keycode, describes the actions associated with a key. The *key_acts* array has *min_key_code* unused entries at the start to allow direct indexing using a keycode. If a *key_acts* entry is zero, it means the key does not have any actions associated with it. If an entry is not zero, the entry represents an index into the *acts* field of the server map, much as the *offset* field of a KeySymMapRec structure is an index into the *syms* field of the client map.

The reason the *acts* field is a linear list of XkbActions is to reduce the memory consumption associated with a keymap. Because Xkb allows individual keys to have multiple shift levels and a different number of groups per key, a single two-dimensional array of KeySyms would potentially be very large and sparse. Instead, Xkb provides a small two-dimensional array of XkbActions for each key. To store all of these individual arrays, Xkb concatenates each array together in the *acts* field of the server map.

The key action structures consist only of fields of type char or unsigned char. This is done to optimize data transfer when the server sends bytes over the wire. If the fields are anything but bytes, the server has to sift through all of the actions and swap any nonbyte fields. Because they consist of nothing but bytes, it can just copy them out.

XkbKeyAction returns the key action indexed by *idx* in the two-dimensional array of key actions associated with the key corresponding to *keycode.idx* may be computed from the group and shift level of interest as follows:

    idx = group_index * key_width + shift_level

STRUCTURES

The KeySymMapRec structure is defined as follows:

```
#define XkbNumKbdGroups        4
#define XkbMaxKbdGroup         (XkbNumKbdGroups-1)

typedef struct {                /* map to keysyms for a single keycode */
    unsigned char    kt_index[XkbNumKbdGroups];  /* key type index for each group */
    unsigned char    group_info; /* # of groups and out of range group handling */
    unsigned char    width;    /* max # of shift levels for key */
    unsigned short   offset;   /* index to keysym table in syms array */
} XkbSymMapRec, *XkbSymMapPtr;
```