

**NAME**

XkbCopyKeyTypes - Copy more than one XkbKeyTypeRec structure

**SYNOPSIS**

**Status** XkbCopyKeyTypes (**XkbKeyTypePtr** *from*, **XkbKeyTypePtr** *into*, **int** *num\_types*);

**ARGUMENTS**

*from*

pointer to array of XkbKeyTypeRecs to copy

*into*

pointer to array of XkbKeyTypeRecs to change

*num\_types*

number of types to copy

**DESCRIPTION**

*XkbCopyKeyTypes* copies *num\_types* XkbKeyTypeRec structures from the array specified by *from* into the array specified by *into*. It is intended for copying between, rather than within, keyboard descriptions, so it doesn't check for overlaps. The same rules that apply to the *from* and *into* parameters in *XkbCopyKeyType* apply to each entry of the *from* and *into* arrays of *XkbCopyKeyTypes*. If any allocation errors occur while copying *from* to *into*, *XkbCopyKeyTypes* returns BadAlloc. Otherwise, *XkbCopyKeyTypes* copies *from* to *into* and returns Success.

**RETURN VALUES**

Success            The XkbCopyKeyTypes function returns Success when there are no allocation errors.

**STRUCTURES**

Key types are used to determine the shift level of a key given the current state of the keyboard. The set of all possible key types for the Xkb keyboard description are held in the *types* field of the client map, whose total size is stored in *size\_types*, and whose total number of valid entries is stored in *num\_types*. Key types are defined using the following structure:

```
typedef struct {
    /* Key Type */
    XkbModsRec    mods;    /* modifiers used to compute shift level */
    unsigned char num_levels; /* total # shift levels, do not modify directly */
    unsigned char map_count; /* # entries in map, preserve (if non-NULL) */
    XkbKTMapEntryPtr map;    /* vector of modifiers for each shift level */
    XkbModsPtr    preserve; /* mods to preserve for corresponding map entry */
}
```

```
Atom      name;      /* name of key type */  
Atom *    level_names; /* array of names of each shift level */  
} XkbKeyTypeRec, *XkbKeyTypePtr;
```

**DIAGNOSTICS**

**BadAlloc**        Unable to allocate storage

**SEE ALSO**

**XkbCopyKeyType(3)**