

**NAME**

XkbAddGeomRow - Add a row to a section

**SYNOPSIS**

**XkbRowPtr XkbAddGeomRow (XkbSectionPtr *section*, int *sz\_keys*);**

**ARGUMENTS**

*section*

section to be updated

*sz\_keys*

number of keys to be reserved

**DESCRIPTION**

Xkb provides functions to add a single new element to the top-level keyboard geometry. In each case the *num\_\** fields of the corresponding structure is incremented by 1. These functions do not change *sz\_\** unless there is no more room in the array. Some of these functions fill in the values of the element's structure from the arguments. For other functions, you must explicitly write code to fill the structure's elements.

The top-level geometry description includes a list of *geometry properties*. A geometry property associates an arbitrary string with an equally arbitrary name. Programs that display images of keyboards can use geometry properties as hints, but they are not interpreted by Xkb. No other geometry structures refer to geometry properties.

One of the components of a keyboard geometry section is one or more rows of keys.

*XkbAddGeomRow* adds one row to the specified *section*. The newly created row contains space for the number of keys specified in *sz\_keys*. They are allocated and zeroed, but otherwise uninitialized. *XkbAddGeomRow* returns NULL if any of the parameters is empty or if it was not able to allocate space for the row. To allocate space for an arbitrary number of rows to a section, use the *XkbAllocGeomRows* function.

**STRUCTURES**

```
typedef struct _XkbRow {
    short      top;
    short      left;
    unsigned short num_keys;
    unsigned short sz_keys;
    int        vertical;
    XkbKeyPtr   keys;
```

```
    XkbBoundsRec  bounds;  
} XkbRowRec, *XkbRowPtr;
```

**SEE ALSO****XkbAllocGeomRows(3)**