

**NAME**

XShmQueryExtension, XShmQueryVersion, XShmPixmapFormat, XShmAttach, XShmDetach  
XShmCreateImage, XShmPutImage, XShmGetImage, XShmCreatePixmap, XShmGetEventBase - X  
Shared Memory extension functions

**SYNTAX**

```
#include <X11/Xlib.h>
#include <sys/ipc.h>
#include <sys/shm.h>
#include <X11/extensions/XShm.h>
```

```
Bool XShmQueryExtension(
    Display *display);
```

```
Bool XShmQueryVersion(
    Display *display;
    int *major, *minor;
    Bool *pixmap);
```

```
int XShmPixmapFormat(
    Display *display);
```

```
Bool XShmAttach(
    Display *display;
    XShmSegmentInfo *shminfo);
```

```
Bool XShmDetach(
    Display *display;
    XShmSegmentInfo *shminfo);
```

```
XImage *XShmCreateImage (
    Display *display;
    Visual *visual;
    unsigned int depth;
    int format;
    char *data;
    XShmSegmentInfo *shminfo;
    unsigned int width, height);
```

```

Bool XShmPutImage(
    Display *display;
    Drawable d;
    GC gc;
    XImage *image;
    int src_x, src_y, dest_x, dest_y;
    unsigned int width, height;
    bool send_event);

```

```

Bool XShmGetImage (
    Display *display;
    Drawable d;
    XImage *image;
    int x, y;
    unsigned long plane_mask);

```

```

Pixmap XShmCreatePixmap(
    Display *display;
    Drawable d;
    char *data;
    XShmSegmentInfo *shminfo;
    unsigned int width, height, depth);

```

```

Status XShmGetEventBase(
    Display *display);

```

## STRUCTURES

*Events:*

```

typedef struct {
    int type;          /* of event */
    unsigned long serial; /* # of last request processed by server*/
    Bool send_event;   /* true if this came from a SendEvent request*/
    Display *display;  /* Display the event was read from */
    Drawable drawable; /* drawable of request */
    int major_code;    /* ShmReqCode */
    int minor_code;    /* X_ShmPutImage */
    ShmSeg shmseg;     /* the ShmSeg used in the request*/
    unsigned long offset; /* the offset into ShmSeg used in the request*/

```

```
} XShmCompletionEvent;
```

a structure of type *XShmSegmentInfo* :

```
typedef struct {
    ShmSeg shmseg; /* resource id */
    int shmId; /* kernel id */
    char *shmaddr; /* address in client */
    Bool readOnly; /* how the server should attach it */
} XShmSegmentInfo;
```

## DESCRIPTION

*XShmQueryExtension* checks to see if the shared memory extensions are available for the specified display.

*XShmQueryVersion* returns the version numbers of the extension implementation. Shared memory pixmaps are supported if the *pixmaps* argument returns true.

*XShmAttach* tells the server to attach to your shared memory segment. If all goes well, you will get a non-zero status, back and your *XImage* is ready for use.

*XShmDetach* tells the server to detach from your shared memory segment.

*XShmPutImage* combines an image in memory with a shape of the specified drawable. If *XYBitmap* format is used, the depth must be one, or a ‘BadMatch’ error results. The foreground pixel in the GC defines the source for the one bits in the image, and the background pixel defines the source for the zero bits. For *XYPixmap* and *ZPixmap*, the depth must match the depth of the drawable, or a ‘BadMatch’ error results.

*XShmGetImage* reads image data into a shared memory *XImage* where *display* is the display of interest, *drawable* is the source drawable, *image* is the destination *XImage*, *x* and *y* are offsets within the drawable, and *plane\_mask* defines which planes are to be read.

*XShmCreateImage* allocates the memory needed for an *XImage* structure for the specified display but does not allocate space for the image itself.

*XShmPixmapFormat* gets the format for the server. If your application can deal with the server pixmap data format, a shared memory segment and *shminfo* structure are created.

*XShmCreatePixmap* points to a pixmap which you can manipulate in all of the usual ways, with the added bonus of being able to edit its contents directly through the shared memory segment.

*XShmGetEventBase* gets the completion event value.

**SEE ALSO**

*MIT-SHM - The MIT Shared Memory Extension*