X11−big−cursor MINIHOWTOHow to use enlarged mouse cursors with the X window system

Table of Contents

X11−big−cursor MINIHOWTO

How to use enlarged mouse cursors with the X window system

Jörg Schneider

v2, 11 August 1997

This document describes how to use enlarged mouse cursors with the X window system.

7.Related info

1.Introduction

There are several reasons why the standard X mouse cursors are hard to track for some people:

- when running X on a notebook with low contrast LCD
- on normal screens when using high resolution, 1600x1280 e. g.
- for visually impaired persons even on normal hardware

In all cases it might help to use enlarged mouse cursors. Ideally this job should be done by a single X program that automatically enlarges every mouse cursor.

To my knowledge there is no simple way to write a utility like this, because the X protocol has no provision to query mouse cursors. For more details see section Technical discussion below.

If we aim for a less general goal, though, something can be done:

There is a set of standard mouse cursors that can be found in the cursor font (try `xfd −fn cursor` to look at it). Most programs use these mouse cursors and the key idea is to replace the standard cursor font with an enlarged version.

2.About this document

The motivation for this MINIHOWTO was a visually impaired co−student who asked me how to enlarge the mouse cursor under X. After I found out how this can be done, I wrote an initial version of this document. The knowledge about the method described here does not seem to be common, so I decided to share it and submitted this document as a Linux MINIHOWTO, despite the fact that it is *not specific to Linux at all*. As all other MINIHOWTOs it can be found in the home of of the Linux Documentation Project (LDP).

The master of this document is maintained in the SGML/linuxdoc format. This makes it possible to automatically provide versions in the following formats (which can be found in the same place as the master): html, text, LaTeX, DVI, PostScript, GNU info.

Shinobu Miyata has done a Japanese translation of this MINIHOWTO. It can be found in http://i11www.ira.uka.de/~schneid/jp/X11−big−cursor/.

3.How to do it

Follow the steps detailed below. If you don't want to get and compile the bdfresize package yourself, you can skip to step 3 and download a magnified font instead of creating it.

1. get cursor.bdf, the source of the cursor font, from some X distribution, e. g. from ftp://ftp.x.org/pub/R6.3/xc/fonts/bdf/misc/cursor.bdf (if you don't find it there try an archie search or get it from my copy).
2. get, compile and install the bdfresize package from ftp://ftp.cs.titech.ac.jp/X11/contrib/Local/bdfresize−1.4.tar.Z (or from my copy):

   ```
   zcat bdfresize−1.4.tar.Z  | tar xf −
   cd bdfresize−1.4
   xmkmf
   make
   ```

   On Linux you probably have to use:

   ```
   make CCOPTIONS='−include /usr/include/bsd/bsd.h' clean all
   ```

3. create a directory and install a magnified cursor font in it (magnification factor 2 in this example):

   ```
   mkdir $HOME/fonts
   bdfresize −f 2 cursor.bdf | bdftopcf >$HOME/fonts/cursor2.pcf
   mkfontdir $HOME/fonts
   ```

   I have prepared some cursor fonts with the following magnification factors: 1.5, 2, 2.5, 3, 4, 5, 6, 7, 8 and 16. You can download one of them an copy it to `$HOME/fonts` if you don't want to use bdfresize.
4. modify your `.xinitrc` or `.xsession` file: before any X client (that uses cursors) is started the following commands must be executed:

   ```
   xset +fp $HOME/fonts
   xsetroot −cursor_name X_cursor
   ```

5. leave your X session and restart.

That's itnow all mouse cursors should have doubled in size.

4.Notes and limitations

- X servers may have a limit for the maximum cursor size, especially if they use a hardware implementation for the mouse cursor. Others do not have such a limit. E. g. XF86_S3 3.3 works even with a 512x512 mouse cursor (rather slowly).
- The magnified cursor font must have the same name as the original font (the *font name* must be `cursor`, *file name* does not matter)that is no problem as `bdfresize` does not change the font name.
- The directory with the new cursor font must be placed before the directory with the standard cursor

font in the the font paththis is accomplished with `xset +fp` (as opposed to `xset fp+`).

- Changes in `$HOME/fonts/` will be visible only after the command `mkfontdir $HOME/fonts; xset fp rehash` and only in newly started X clients (more exactly: for newly created cursors).
- `xset +fp path` may not work on a X−Terminal. In this case a font server (see the section How to use a fontserver) can be used if supported by the X−Terminal or some other method to install the font on the X−Terminal (this can generally only be done by your system administrator).
- The same approach can be used for `olcursor` and `decw$cursor` fonts and any other cursor font you may encounter.
- Cursor fonts produced by bdfresize don't look smooth, especially at larger magnification factors. It would be nice if someone could create better looking handcrafted version at some common sizes.

5.Technical discussion

Is it possible to write a X program that enlarges cursors automatically?

(Partial) solution 1

Use the XTestCompareCursor from the XTEST extension. For all windows that the mouse pointer enters compare the cursor of this window with a set of `known' cursors (e. g. from the cursor font). If the cursor is found, replace it with an enlarged version, otherwise either leave it alone or substitute a standard cursor. This will only work where the XTest extension is available.

Solution 2

Write a proxy X server that relays all client requests unchanged to the real X server, except that it intercepts all requests corresponing to the `XCreate*Cursor` Xlib functions. `XCreate*Cursor` requests should be modified to use an enlarged cursor.
This proxy server simulates a new display, e. g. `:1`. All clients that connect to this display (e. g. `xterm -display :1`) are displayed on the real server (normally `:0`) and their mouse cursors are enlarged automatically. The mouse cursors of clients that connect to `:0` will remain *unchanged*.

6.Other ideas how to make the mouse cursor more visible

Here are some ideas for rather simple X programs that might make mouse cursors easier to track.

- When a hot key is pressed display something (big cursor, small window, shaped window) at pointer position for 0.5s.
- use XRecolorCursor to change the mouse cursor color every 0.1s

A more demanding project would be **mouse trails** à la windoze, i. e. when the mouse is moved and the mouse cursor needs to be drawn in a different position, then the old mouse cursor does not disappear at once, but after a short delay. Mouse trails would be probably best implemented in a X server, but it might be feasible to do it as a X client, or better as a proxy server (see section Technical discussion for details).

7.Related info

7.1 How to use a font server

A font server is a net service that provides a set of X11 fonts with a simple protocol. It can be queried which fonts it provides and will supply the font bitmap data on request.

You might want to use a font server to provide the X server with a modified cursor font, instead of telling it where to find the font on the file system.

This method is especially handy if you use several machines that don't share a common file system or if you use X terminals that support the font server protocol.

A font server program and associated tools comes with the X11R5+ distribution (AFAIK).

Setting up a font server

Read the manual pages *fs(1)*, *fslsfonts(1)* (or *xfs(1)*, *xfslsfonts(1)* under X11R6) and try itit isn't hard. Say, you are running the server on host `some.host.edu` on port 7100. You can test the setup with the command

```
fslsfonts −server some.host.edu:7100
```

To actually use the server issue the command

```
xset +fp tcp/some.host.edu:7100
```

which should return without an error message.

7.2 How to get the bdf source for some font

If you have set up a font server simply use `fstobdf` which comes with the font server.

Alternatively you may try `getbdf` which can dump any installed X11 font to a bdf file.
