

Re: Fast memcpy in kernel space

Source: <http://linux.derkeiler.com/Newsgroups/comp.os.linux.development.system/2003-08/0237.html>

phil-news-nospam_at_ipal.net

Date: 08/12/03

Date: 12 Aug 2003 16:17:47 GMT

On Mon, 11 Aug 2003 22:29:07 -0700 Tim Roberts <timr@probo.com> wrote:
| Jayr Al-Dyn <jayraldyn@jayraldyn.net> wrote:

|>

|>I think the best way is to have two processes: one of them in user mode,
|>capturing from the X server (with XGetImage, I guess) and the other as a
|>kernel module receiving data from the first one and delivering to a
|>/dev/video device.

|

| I doubt it. What are you actually trying to do? Are you trying to
| implement a screen-capture program? You should probably investigate some
| of the XFree86 extensions to get access to the frame buffer directly.
| XGetImage is not the most efficient method. Xshm might be a good choice.

I would like to see a VC managed method of mapping direct access to the video hardware frame buffer. Processes would have a specific VC memory mapped. For the processes which have mapped the VC that is currently active, the segment tables would have the virtual addresses going directly into the video buffer in real memory. When a VC switch takes place (and this is the slower part) the video memory will be copied out of the hardware buffer to RAM, and the RAM copy of the VC being switched to would be copied in. As long as the video is operating in a consistent mode, so that mode changes do not have to take place between VC switching, this should work. The catch is to have a LOT of RAM for holding all the VCs that are mapped in graphical mode (those which are mapped in text mode would operate as usual, drawing characters to the video hardware buffer from the text buffer ... slower but with less demand on RAM).

Programs that operate on a variety of text VCs don't need to re-draw their contents. But with graphical FBs, that's not been the case (unless there are recent changes I missed). The chief argument against true VFBS is the large amount of RAM involved. But isn't that getting to be less and less of a valid argument? And one would not have to put every VC into graphical mode.

There are things I would like to do from userland with direct hardware access to the video buffer (once the mode is set up).

--

| Phil Howard KA9WGN | <http://linuxhomepage.com/> | <http://ham.org/> |

comp.os.linux.development.system: Re: Fast memcpy in kernel space

| (first name) at ipal.net | <http://phil.ipal.org/> <http://ka9wcn.ham.org/> |
