

Re: X-windows: changing cursor from and AST ?

Source: <http://unix.derkeiler.com/Newsgroups/comp.os.vms/2005-03/2146.html>

From: Bob Gezelter (gezelter_at_rlgsc.com)

Date: 03/31/05

Date: 31 Mar 2005 02:33:21 -0800

JF Mezei wrote:

- > *In an application which has a function which can either be instant or*
- > *take a long time depending on size and type of contents, I'd like to*
- > *be*
- > *able to trigger a busy X-windows cursor only if it is to take a*
- > *certain*
- > *amount of time.*
- >
- > *Since the loop is outside the X-events main loop, and since it*
- > *usually*
- > *executes rather quickly, would it be possible (knowing that the loop*
- > *itself does not generate any X-windows updates to the display) to*
- > *have*
- > *an AST triggered by a timer say one second after start of loop to*
- > *change*
- > *the cursor to a busy cursor ?*
- > *(and possibly update cursor every second).*
- >
- > *When the loop completes, it simply cancels the timer and then reverts*
- > *the cursor back to normal.*
- >
- > *If the loop performs some windows functions, would it be just a case*
- > *of*
- > *canceling AST delivery before you make a call to an X function (and*
- > *re-activating it later) ?*

JF,

Since you are running most of the loop in process level, this is one of the rare times when the old [generally bad] example of how to use an AST is appropriate.

Request a TIMER operation, with an AST and a AST Parameter that points to a data structure. When the AST fires, flip a flag that specifies the cursor style.

To maximize cleanliness, there are two approaches:

comp.os.vms: Re: X-windows: changing cursor from and AST ?

- as you mentioned, cancel the timer request, but you must be careful, you can create a race condition if you are not careful
- the approach I prefer, is to manage some form of event number, and a table as the AST parameter. Then you can arrange things to that the timer is not canceled, but does not produce any changes if it is no longer relevant. This avoids the possibility of a race condition.

I hope that the above is helpful.

- Bob Gezelter, <http://www.rlgsc.com>