burning cpu and battery on the gnome desktop

ryan lortie (desrt)

january 19, 2007

why am i here?

why am i here?

- two problems

but first, some motivation...

but first, some motivation...

- sleeping chips

but first, some motivation...

- sleeping chips
- tickless kernel

but first, some motivation...

- ► sleeping chips
- ► tickless kernel
- ► olpc-style microsleeps...

polling is becoming increasingly expensive...

polling is becoming increasingly expensive...

- ▶ when devices use very low power,
  waking them up is relatively costly

polling is becoming increasingly expensive...

- ▶ when devices use very low power,
  waking them up is relatively costly
- ▶ even on your own laptop

the problem is polling

the problem is polling

- please hold your questions...

the problem is polling

- please hold your questions...
- every 10 seconds, i'll just ask you if you have any.

why do applications want to poll?

why do applications want to poll?

- for no good reason.

why do applications think they want to poll?

why do applications think they want to poll?

- monitor files for changes

why do applications think they want to poll?

- ▶ monitor files for changes
- ▶ poorly written mainloop

why do applications think they want to poll?

- ▶ monitor files for changes
- ▶ poorly written mainloop
- ▶ update their display

why do applications think they want to poll?

- ▶ monitor files for changes
- ▶ poorly written mainloop
- ▶ update their display
- ▶ check hardware for real world events

why do applications think they want to poll?

- ▶ monitor files for changes
- ▶ poorly written mainloop
- ▶ update their display
- ▶ check hardware for real world events
- ▶ reasons i can't even begin to guess at

some examples (stuff that got fixed)

some examples (stuff that got fixed)

- gnome-vfs

some examples (stuff that got fixed)

- gnome-vfs
- vte

some examples (stuff that got fixed)

- ▶ gnome-vfs
- ▶ vte
- ▶ gnome-system-tools backend

funny story about gnome-system-tools backend

funny story about gnome-system-tools backend

- woke up 20 times per second

funny story about gnome-system-tools backend

- ▶ woke up 20 times per second
- ▶ to check for file changes in /etc...

funny story about gnome-system-tools backend

- ▶ woke up 20 times per second
- ▶ to check for file changes in /etc...
- ▶ ...using fam...

funny story about gnome-system-tools backend

- woke up 20 times per second
- to check for file changes in /etc...
- ...using fam...
- fam isn't even installed.

funny story about gnome-system-tools backend

- woke up 20 times per second
- to check for file changes in /etc...
- ...using fam...
- fam isn't even installed.
- was doing absolutely nothing, 20 times per second.

some examples (stuff that is hard to fix)

some examples (stuff that is hard to fix)

- gnome-vfs

some examples (stuff that is hard to fix)

- gnome-vfs
- clock applet

some examples (stuff that is hard to fix)

- ▶ gnome-vfs
- ▶ clock applet
- ▶ i810 driver

some examples (stuff that is hard to fix)

- ▶ gnome-vfs
- ▶ clock applet
- ▶ i810 driver
- ▶ gtkrecentmanager

some examples (stuff that is hard to fix)

- gnome-vfs
- clock applet
- i810 driver
- gtkrecentmanager
- gnome-screensaver

some examples (stuff that is hard to fix)

- gnome-vfs
- clock applet
- i810 driver
- gtkrecentmanager
- gnome-screensaver
- hal

some timer use that is good

some timer use that is good

- vte

some timer use that is good

- vte
- in general, the use of one-off timers to defer work

how do you avoid polling?

how do you avoid polling?

- inotify

how do you avoid polling?

- inotify
- poll() on sysfs

how do you avoid polling?

- inotify
- poll() on sysfs
- mask timers when not needed

how do you avoid polling?

- inotify
- poll() on sysfs
- mask timers when not needed
- things like gtk-cursor-blink-timeout

how do you avoid polling?

- ▶ inotify
- ▶ poll() on sysfs
- ▶ mask timers when not needed
- ▶ things like gtk-cursor-blink-timeout
- ▶ for the love of god, please just use glib.

and if you absolutely must poll...

and if you absolutely must poll...

- obviously, less is better.

and if you absolutely must poll...

- ▶ obviously, less is better.
- ▶ use g_timeout_add_seconds, if possible.

and if you absolutely must poll...

- obviously, less is better.
- use g_timeout_add_seconds, if possible.
- mask or decrease timer frequency when unneeded.

and the other problem?

and the other problem?

- excessive notification

how is it possible to have too much notification?

how is it possible to have too much notification?

▶ events waking up too many processes

how is it possible to have too much notification?

- events waking up too many processes
- processes being woken up by too many events

how is it possible to have too much notification?

- ▶ events waking up too many processes
- ▶ processes being woken up by too many events
- ▶ lots of context switching

examples of this

examples of this

- every single gtk+ app on the desktop

examples of this

- every single gtk+ app on the desktop
- propertynotify on toplevel x windows

examples of this

- every single gtk+ app on the desktop
- propertynotify on toplevel x windows
- inotify for missing files

which is worse?

which is worse?

- it is far worse to poll more often

which is worse?

- it is far worse to poll more often
- with notification, the system was already awake

finding broken applications

finding broken applications

▶ use timeouttop

finding broken applications

- ► use timeouttop
- ► use strace

finding broken applications

- use timeouttop
- use strace
- use gdb

timeouttop

timeouttop

- kernel patch to report wakeups per-process (via /proc)

timeouttop

- kernel patch to report wakeups per-process (via /proc)

timeouttop

- kernel patch to report wakeups per-process (via /proc)
- "top"-like frontend to catch offenders

timeouttop

- kernel patch to report wakeups per-process (via /proc)
- "top"-like frontend to catch offenders
- http://desrt.mcmaster.ca/code/pstimeouts/

strace

strace

▶ nice tool to see what programs are up to

strace

- nice tool to see what programs are up to
- can attach to running processes with -p

strace

- ▶ nice tool to see what programs are up to
- ▶ can attach to running processes with -p
- ▶ instant visual feedback about wakeups

gdb

gdb

- find out exactly why processes are waking up

gdb

- ▶ find out exactly why processes are waking up
- ▶ for glib programs, breakpoint g_timeout_dispatch

gdb

- ▶ find out exactly why processes are waking up
- ▶ for glib programs, breakpoint g_timeout_dispatch
- ▶ will work best with debugging packages installed

that's it.
any questions?