



X Application Performance Profiling

Adam Jackson <ajax@redhat.com>
linux.conf.au 2008

oprofile is hard

- Let's go shopping!

X-Resource

- Adds client introspection to X
- XIDs uniquely identify a client
- Resource accounting
- Pixmap memory usage accounting

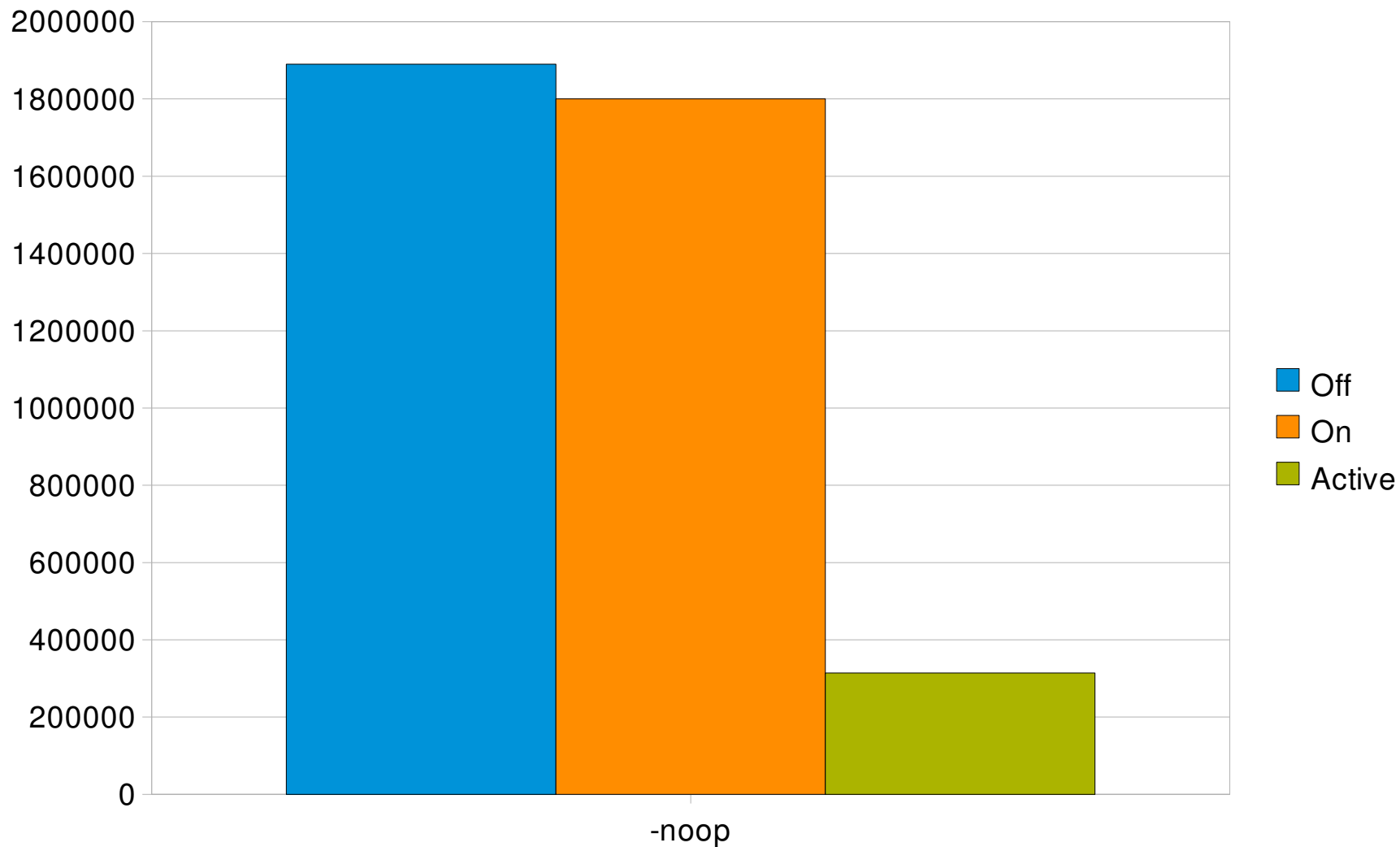
XAccessControlExtension

- Framework for building security models
- X-SELinux, X-TSOL, etc.
- Hooks
- Lots of hooks
- Really, wow, lots of hooks

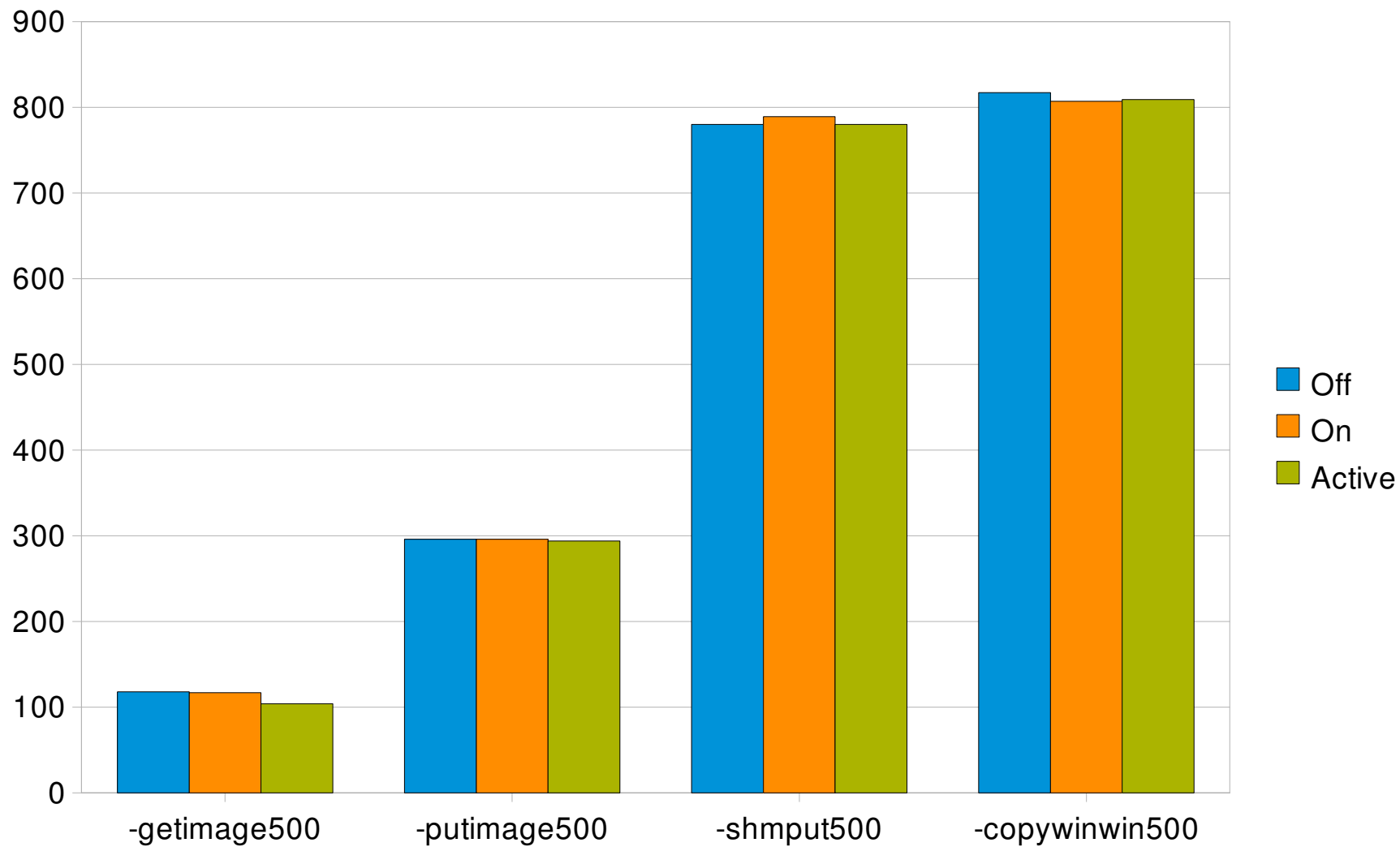
You got chocolate in my peanut butter!

- Three new requests in X-Resource
- WatchClientTime, UnwatchClientTime
- ReportClientTime
- XACE dispatch audit hooks
- Report is just `struct timespec`
 - ... times about 32896
 - ... and truncated to `uint32_t`

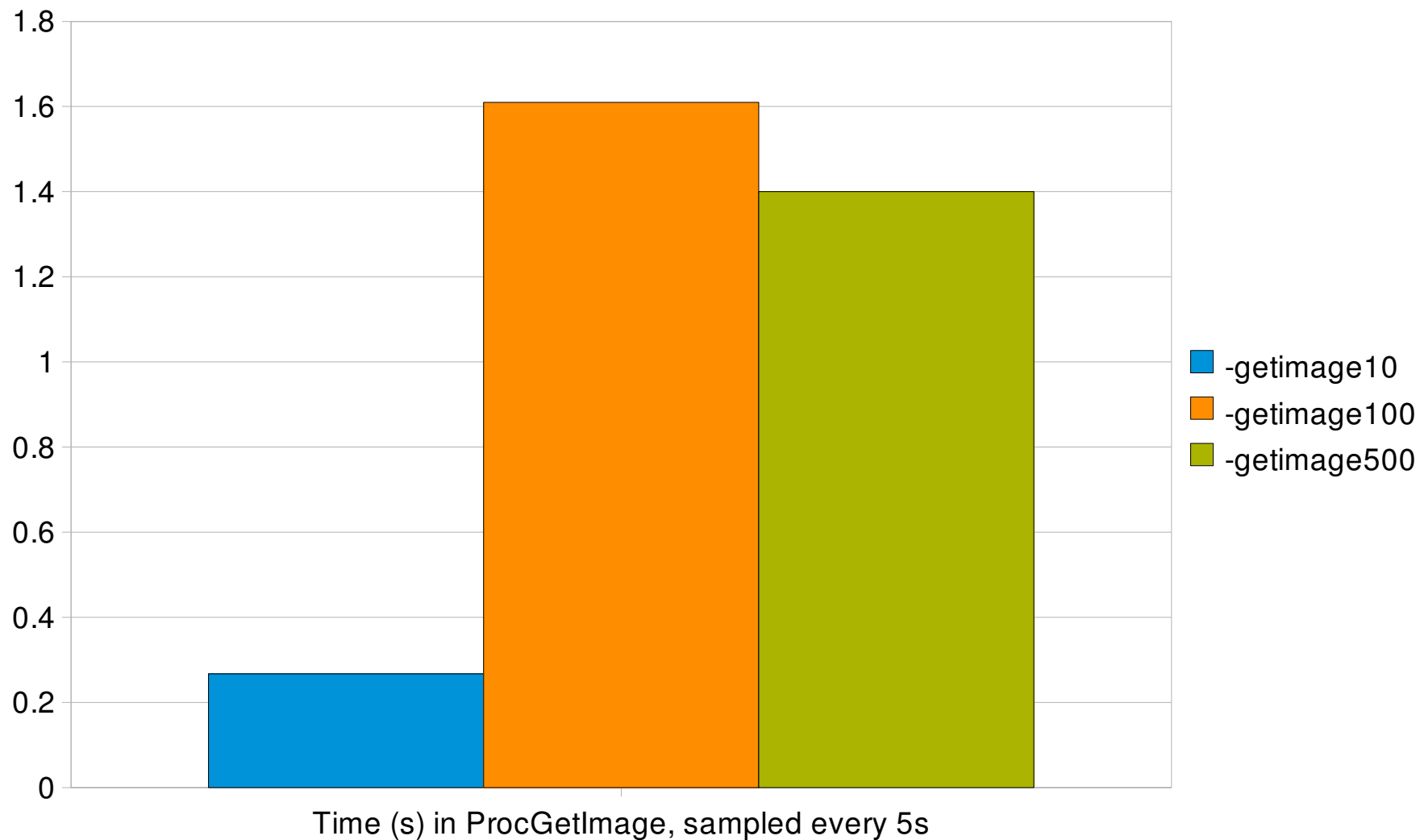
Xvfb no-op performance



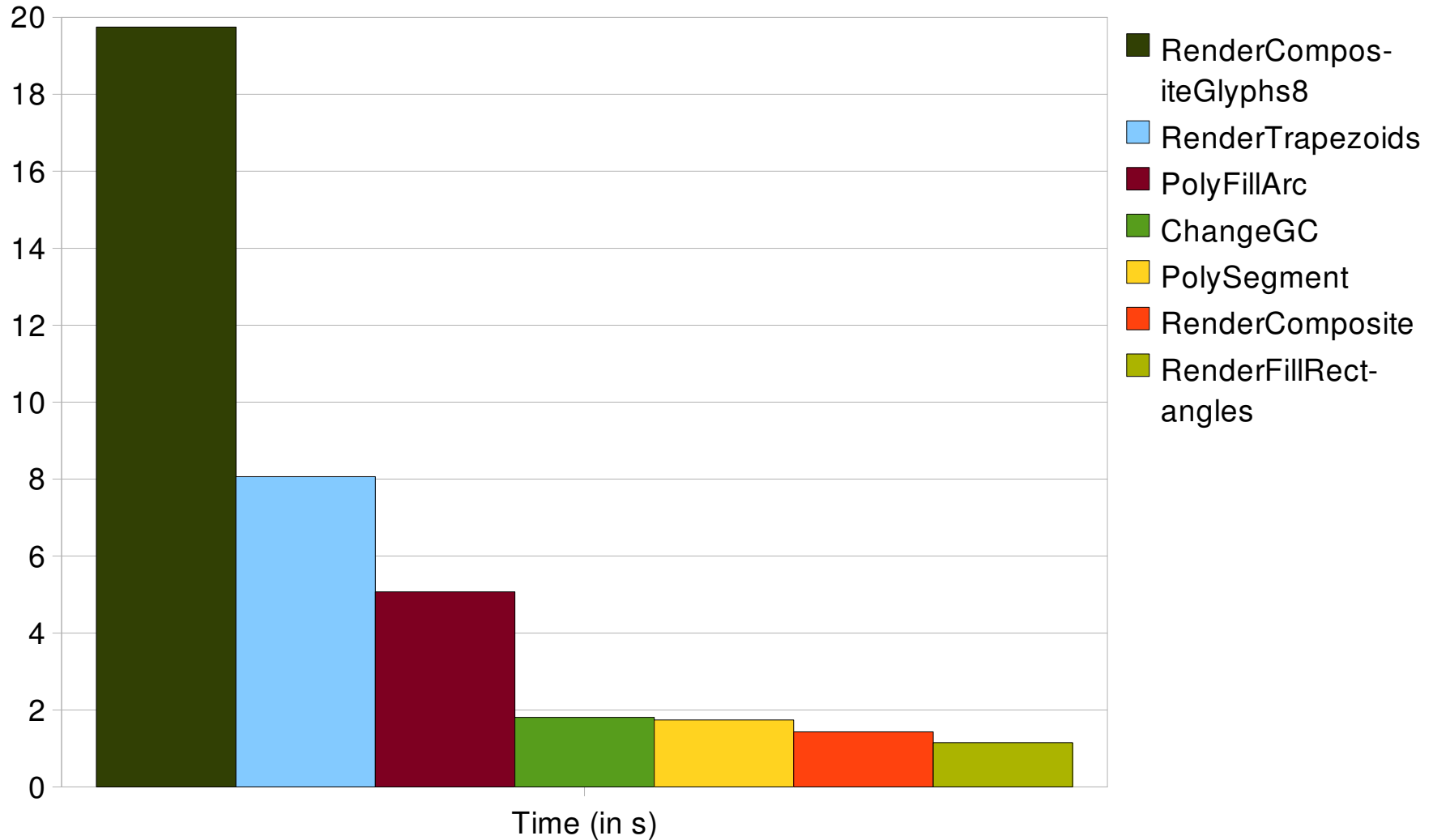
Xvfb blit performance



Xvfb GetImage weirdness



Xvfb gtkperf -c 500



Client TODO

- Python XRes bindings
- Pretty top-like GUI
- Instrument xlib/xcv for better backtrace
- mshare(2)
- Tackle the DRI profiling problem

Server TODO

- Attack dispatch overhead
 - Zero-copy image transfer
 - Reduce in-server request buffering
 - epoll(2) like it's 1999
- Break (and fix) internal API dumbness

So why isn't this in git yet?

- API needs a second look
- No accounting for GPU or sleep time
- No accounting for dispatch overhead
- Leaks watches
- Swapped dispatch is buggy
- Extension request reporting is hacky

