Gnome 3.0

gnome-shell, clutter and the future

Tomáš Bžatek <tbzatek@redhat.com>
GNOME Roadmap

- GNOME 2.0 released in 2002
- strong roadmap, 6 month release cycle
- GLib, GTK+ roadmap partially independent on Gnome

- "community wants change"
- "Because of lack of excitement. Because of lack of vision."
 GNOME 3.0 Roadmap

• GNOME 2.30 == GNOME 3.0

• "if the QA Team feels a release should be delayed, then it will be delayed."

• radical changes will start with GNOME 3.2
GNOME 3.0 Roadmap

- two new major features:
  - gnome-shell
  - Zeitgeist
- „both have already been in development for a while, GNOME 3.0 will be less about "starting", and more about "finishing". “
- possible GStreamer 1.0 and geolocalization inclusion (with GeoClue and libchamplain)
- the goal is to let developers know the preference so they start using new libraries
Library deprecation *(codename Project Ridley)*

- LibBonobo
- LibGnome
- LibGnomeUi
- LibGnomeCanvas
- LibArtLGPL
- LibGnomeVfs
- LibGnomePrint
- ESPSound
- Orbit
- LibGlade
- LibSexy
- LibEgg
- LibEel
- GnkGIExt
GLib 3, GTK 3 API changes

- 3.0 API mostly compatible with latest 2.x API
- no use of deprecated symbols - `G_DISABLE_DEPRECATED`
- top-level header includes (e.g. `glib.h`, `gobject.h`, `gio.h`) - `G_DISABLE_SINGLE_INCLUDES`
- use accessor functions instead direct access - `GSEAL_ENABLE`
- possible fundamental changes in GObject
- lots of planned theming changes
- recently landed client-side windows, offscreen rendering
GSettings

- configuration system, GConf replacement (feature-wise)
- possibly different backends, dconf by default
- no more request serialization
- caching, in memory db, safe fsync
- fast reading, writing provided by separate daemon
- still experimental
GObject-Introspection

- "will deeply change the way GNOME development can be done"
- "Two level applications - C and <your favorite runtime>"
  - C for the core and performance-critical stuff
  - higher level language for convenience (garbage collector, memory allocations)
- automatically generating bindings in runtime
GObject-Introspection

• other nice features:
  – API verification
  – automatic documentation generator

• the plan is to let gobject-introspection generate C bindings from D-Bus interface, creating convenient C interface to D-Bus services
Clutter

- graphics library (scene-graph) for creating UI
- "for creation of fast, animated, visually bespoke and impressive graphical interfaces"
- uses OpenGL to accelerate transforms etc.
- extensively used in Moblin GUI
Clutter

- ~70k lines codebase, 12+ k lines test suite
- 1600+ symbols in Clutter (99% documented)
- ~300 symbols in COGL (90% documented)
- 1200+ single commits
- 29 single authors, 14 non-Intel

- stable 1.0.2 release
Clutter

- written around GObject, can be integrated with GLib mainloop, GTK+ - like API (no knowledge of OpenGL required)
- actors (widgets), stage (window)
- actors can be:
  - positioned in 3D space
  - rotated around 3D axis
  - sized
  - scaled
  - have opacity set
  - clipped
gnome-shell

• panel and window manager replacement
• flexible workspaces
• overlay (dashboard) replaces „Start menu“

• gnome-panel will become obsolete, shipped few releases for compatibility, then dropped
• work in progress, technology preview in Gnome 2.28
gnome-shell

• no taskbar
• application based:
  – preventing multiple instances, but allowing multiple windows from one process
  – no persistency, ability to close apps normal way (WM decoration or exit() )
  – switching based on applications, not on windows (probably two different keystrokes like in OS X...)
gnome-shell

- Mutter - moblin origin, compositing WM, metacity codebase
- C Core
- UI written in JavaScript (GJs = the Mozilla SpiderMonkey binding), bridged using gobject-introspection
Gnome-shell

- planned media and event notifications (IM, X urgent hint)
- developer tools - Looking Glass
- integrated recording tool
- future Zeitgeist integration
Zeitgeist

- "Changing the way we access documents"
- "a tool for easy browsing and finding files on your computer"
- some kind of "journal"
- logging your day-to-day activity
- navigation: using timelines, tags, bookmarks, etc.
- daemon + GUI architecture
Zeitgeist use cases

- gnome-shell integration (Recent documents)
- parental control, killing apps (browsing porn pages)
- real-time team workflow monitoring by forwarding personal activities to others
- overview for monthly reports (for your boss)
- restoring a work state of a certain time. All documents, which have been open
Thanks for attention