How GNOME Obsoleted its “Enable Accessibility” Setting

Alejandro Piñeiro (apinheiro@igalia.com)
Topics

• Accessibility vs. Accessibility Support
• Phase 1: Getting Started
• Phase 2: Rebuilding What We Had
• Phase 3: Achieving “Always On” Accessibility
• How Does This Affect Me?
Accessibility vs. Accessibility Support

An extremely quick introduction
Accessibility: removing barriers

- Themes and fonts
- Keyboard shortcuts
- Visual bells
- Assistive Technologies
Accessibility support: enabling access via ATs

- Screen Reader
- Screen Magnifier
- Navigation by on-screen keyboard
- Navigation by voice
Phase 1: Getting Started

Creating an Accessible, Free Desktop
Starting GNOME Accessibility

- Mar 1999: GNOME 1.0 release

- Oct 2000: “GNOME Accessibility Summit”
  - Sun, IBM, and others
  - How to make an accessible Free Desktop
  - Focus on GNOME 2.0

- Jun 2002: GNOME 2.0 released
Bolted on; not built in

- Allowed to evolve independently
- Implemented as gmodule plugins
  - gail
  - atk-bridge
- Loaded conditionally at run-time based on an Accessibility “on/off” user setting
Plugin pros

• Support won't interfere when it is not needed
• No additional memory/CPU overhead

(i.e. Hide the problems rather than fix them)
Plugin cons

- Tied to the session
- Loaded only at application start time
- Users need accessibility enabled in order to enable accessibility
Enabled by default in GNOME 2?

- Nov 2005: Accessibility enabled by default for development releases: **ACCEPTED**

- Jul 2008: Accessibility enabled by default for stable releases: **REJECTED**
“Still we have the problem that our current accessibility technology just sucks too much for being enabled by default.”

-- desktop-devel-list commenter
Wait a moment …

• Still “sucks too much” after eight years?
• Accessibility was seen as an optional add-on.
• The Accessibility Team was always small.
• And then …
Perfect storm

- Corporate accessibility development stops
  - Corporate “change of direction”
  - Layoffs followed by takeovers

- GNOME 3 development starts
Phase 2: Rebuilding What We Had

And taking the opportunity to make it better
2010: Rework existing support

- Bonobo deprecation
- AT-SPI over DBUS (AT-SPI2)
- GObject Introspection
- GSettings vs GConf
2010: Implement new support

- New toolkits like Clutter
- New widgets in existing toolkits like Gtk+ 3
- New desktops like GNOME Shell
2011: Make improvements

- Fix regressions in the new and reworked code
- Improve toolkit a11y support, e.g. GtkTreeView
- Improve performance and stability
2010-2011: Integrate modules

- Cally becomes part of Clutter
- Gail becomes part of GTK
- ATK implementations are now built-in
- But atk-bridge remained a plugin
Phase 3: Achieving “Always On” Accessibility

It takes a community to obsolete a setting.
2012: “Year of Accessibility”

- Hackfest to plan further work, including accessibility enabled by default
- Developers to focus on significantly improving the accessibility stack
- Friends of GNOME Campaign to support additional development
What changed?

- Stability improvements
- Performance improvements
- Accessibility framework not sending/getting messages unless an AT is listening
Enabled by default in GNOME 3?

• Apr 2012: proposed again

• tl;dr: ACCEPTED
The conservative approach

- Proposed by the Accessibility Team
- Just change the default value of the setting
- Approved by the GNOME Community
The no-turning-back approach

- Proposed by Benjamin Otte & Bastien Nocera
- Make atk-bridge a library which toolkits link to
- Embraced by the community (who pitched in)
- Implemented as the solution
What's the difference?

- Not an add-on but an add-in
- For users: It JustWorks™
- For developers:
  - App accessibility is tested by everyone
  - The atk-bridge is compiled by more developers
Is it working?

• Accessibility is always on as of GNOME 3.6.

• Distros are shipping it, users are using it.

• Developers have not suggested we undo it.

• We're finding and fixing bugs here and there.

• But...Yeah, it's working.
How Does This Affect Me?

As a developer who wants to make my application or environment accessible
Are you using GTK?

• If you are using GTK $\geq$ 3.6:
  – GTK depends on the new library
  – Your app should be accessible out of the box

• If you are using GTK2:
  – These changes were not backported
  – The setting was kept in GNOME 3 for such apps
Any other toolkit/application?

- Assuming that you already have an ATK implementation for your widgets ...

- You only need to use at-spi2-bridge library
  - Add the dependency
  - Call an init method

- GNOME Shell's patch:
  - 16 insertions, 199 deletions
Questions?