Looking back at GTK+-2.0

Owen Taylor
Red Hat, Inc
otaylor@redhat.com
Introduction

What did we do?
How did we do it?
How did it go?
How do we do it better next time?
Scope

GObject

GDK Improvements
  Ports (Win32, linux-fb)
  32-bit coordinates
  Double buffering

Pango, Unicode

New Widgets
  GtkTextView
  GtkTreeView

gdk-pixbuf

Accessibility
General ideas

Make GTK+ a world class toolkit

Do large scale cleanups

  GObject
  Port infrastructure

Fix problem widgets

  GtkText => GtkTextView
  GtkWidget, GtkWidget, GtkWidget, GtkWidget => GtkWidget

Add support for the hard stuff

  Internationalization
  Accessibility
Timeline

Feb 1999 - Branched 1.3
Mar 1999 - Start integrating in Win32
Jul 1999 - Coding starts in GScript (later Pango)
Nov 1999 - Add port structure (gdk/x11, win32)

Apr 2000 - GtkTextView
Apr 2000 - Pango integrated
May 2000 - GObject integrated
Jun 2000 - gdk-pixbuf integrated
Oct 2000 - GtkTreeView integrated
Dec 2000 - Xft support added

Jan 2001 - Started using Bugzilla
May 2001 - Dependency on ATK added

Mar 2002 - Released 2.0.0
Stats

ChangeLog
   45,000+ lines
   1.5+ megabytes

Lines of code added
   Over 300,000 lines of code added
   More than 1 line every 5 minutes over 3 years

Bugzilla bugs resolved
   More than 1600

Contributors
   288 separate people credited in release announcements
   (Core team of 10-15)
Small core team (Tim and Owen)

Give people authority over areas they work on
  Threading - Sebastian Wilhelmi
  GtkTextView - Havoc
  GtkTreeView - Jonathan, Kristian Rietveld
  Documentation - Matthias Clasen
  ATK - Sun accessibility team

Ideal is to have two eyes look at any change
Planning

Rough scope decided early 2000 (1 year in)
    Plan was GOBject, Pango, GDK enhancements, GtkTextView

GtkTreeView added later that year

Atk added in early 2001

No public roadmap; informal discussions
## Tools

### CVS

### Bugzilla

### Mailing lists

- gtk-devel-list@gnome.org
Bugzilla

Defect tracking
  Bugs

Issue tracking
  Things that need to be done

Enhancement tracking
  Things to do
Project documentation

CVS history + ChangeLog + Bugzilla + Lists

Cross-references give coherency
  ChangeLog references bug #
  Bug report references mailing lists

Design Documents
  Go to mailing list
  Do they need to be separately maintained, updated?
What went well

Lots of new functionality

High quality
  Careful review of code going in
  Always was in a working state

Community contributions

Lots of testing, people using it in last year or so
  Time to correct API mistakes
What went badly

Schedule
  Timescale not ridiculous for amount of work
  Prediction poor (about a year longer than expected)

Design work somewhat private

Division of labor
  Most of work still done by a few people

Users got no significant enhancements for 3 years
Process improvements

Shorter release cycles
  Better predictability
  Shorter "time to market"

Public design documents

Separation of roles
  Release mechanics
Future roadmap

2.2
  Multihead support (Erwann Chenede)
  Official Win32 support (Tor and Hans)

2.4
  File selection
  "Combo" widget(s) (Kristian Rietveld)
  New Menu API (James Henstridge)

2.x
  Icon list
  Sheet widget
  Help API
  Enhanced geometry management

3.x
  "Java2D" like rendering
  New theme system