The GNOME Architecture

Federico Mena-Quintero
Red Hat Advanced Development Labs
federico@redhat.com

July 8, 1999

Overview

- What is in GNOME?
- Architecture in detail
- GNOME programming guidelines

What is in GNOME? (1)

- Graphical desktop environment
  - GTK+ as the GUI library and object system
  - Standard GUI
  - Consistent look and feel
  - Drag and drop
- Application framework
  - High-level widgets for applications
  - Internationalization (all 20 letters of it)
  - Session management
  - Configuration back-end
- Imaging architecture
  - The GNOME canvas
  - Libart imaging model
  - Printing system
  - Font system

What is in GNOME? (2)

- Component model
  - CORBA as the backend
  - Bonobo component model
  - Interfaces for (lack of) Unix services
- Miscellaneous backend libraries
  - libxml — XML loading/saving/parsing
  - gnome-db — Generic database API
  - libglade — GUI builder library
  - gnome-mailer — Mail API
  - libgtop — System information
  - zvt — Fast terminal emulation widget

GTK+ and Glib

- Glib
  - Data structures
  - Utility functions
  - Main loop abstraction
  - Portability functions
  - Threads and mutexes
- GTK+
  - Gdk wraps Xlib
  - Gtk provides objects and widgets

Glib (1)

- Data structures
  - Lists
  - Trees
  - Hash tables
  - Resizable arrays
  - Strings
  - Relations (N-way mappings)
  - Generic cache
- Utility functions
  - Memory allocation and memory chunks
  - Assertions for correctness
  - Logging and debugging
  - Datasets and hooks
  - Date functions
  - A lexical scanner

Glib (2)

- Main loop abstraction
  - Data sources for select(2)-like functionality
  - I/O channel abstraction
  - Timers
  - Idle functions
- Portability functions
  - All sorts of weird string functions
  - Byte-swapping functions
  - Useful macros
- Thread and mutex abstraction

Gdk

- Thin wrapper over Xlib
- Makes operations less hard (or easier)
- Ported to Win32 as well

GTK+

- Object and type system
  - Single inheritance
  - Signal/slot mechanism
  - Dataset mechanism
  - Dynamic type registration
- Widget system
  - Widgets and containers
  - Geometry negotiation
  - Signals for notification

GTK+ Widget Organization

- Base GtkWidget class
  - Signals for events
  - Signals for map/unmap/realize/unrealize/etc.
- Base GtkContainer class
  - Containers are widgets
  - They manage their children's geometry
  - Request/allocate size negotiation policy

GTK+ **Drag and Drop**

- Xdnd and Motif protocols
- Protocol is transparent to the programmer
- Based on MIME-types

Imlib, GdkPixbuf, and GdkRGB

- Imlib/gdk_imlib
  - Image loading with fallbacks
  - Color reduction and dithering
  - Image/pixmap Cache
- GdkRGB
  - High-quality dithering
  - Part of Gdk
- GdkPixbuf
  - Replacement for gdk_imlib
  - Separates loading, caching, and rendering
  - Uses GdkRGB

Core GNOME Libraries

- Libgnome for UI-independent stuff
  - Configuration file handling
  - Metadata for files
  - MIME-type handling
  - URL and Mozilla-remote handling
  - High score handling
  - Desktop entry files
- Libgnomeui for GUI framework
  - High-level widgets
  - Session management client
  - The GNOME canvas

Libgnome Features

- Configuration file handling
- Metadata information for files
  - Icons images, positions
  - Stored in a db2 database
  - Per-user and systemwide metadata
- MIME-types
  - Basic data typing system
  - Files are typified via a regex or file(1)-like match
  - Actions bound to MIME-types
- URLs, scores, desktop entries

Libgnomeui features

- High-level widgets
  - Toplevel application window
  - Menus and toolbars
  - Stock icons
  - Property dialogs
  - GnomeGuru (wizards)
  - Multiple document interface (MDI) framework
  - The GNOME canvas
- Session management client
  - Implemented as a GtkObject
  - Signals for SM requests

Internationalization

- GNU gettext for storing and translating message catalogs
- Unicode in GTK+ is the next step