Gtk Marries Ada:
The GUI Technology Revolution

charlet@gnat.com

charlet@gnat.com
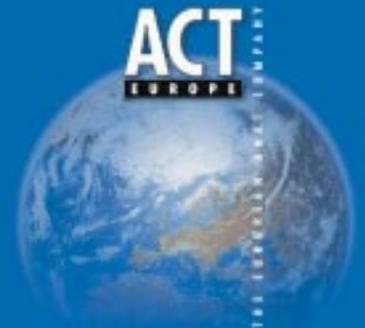
Overview

- **History of GtkAda**

- **GtkAda Features**

- **Why Gtk Rather Than Other Toolkits?**

- **Why GtkAda rather than Gtk+?**

GtkAda - History

- ✓ **The GIMP**

  - – **GNU Photoshop clone**

- ✓ **The Gtk+ library**

  - – **Becomes independent**

- ✓ **GtkGlade: a Gtk+ GUI builder**

- ✓ **Gnome: a desktop manager**

- ✓ **GVD: the GNU Visual Debugger**

GtkAda Features

✓ **High-level binding to the Gtk+ library**
  - **object-oriented**
  - **type safety**
  - **small and efficient**

✓ **Highly portable**
  - **Unixes: Linux, Solaris, …**
 - **Windows NT**

✓ **Native**

GtkAda Features (2)

✓ **Event handling**

✓ **Drawing services**

    – **Lines, rectangles, …**

    – **OpenGL (3D graphics)**

✓ **Style support**

✓ **Large set of widgets**

GtkAda - Widget set

- ✓ **Two types of widgets**

  - – **containers and visual objects**

- ✓ **About 100 widgets are provided**

- ✓ **High-level widgets**

  - – **notebook, text widget, tree, spin button, table, toolbar, ...**

GtkAda - Screenshots

✓ **Ctree**
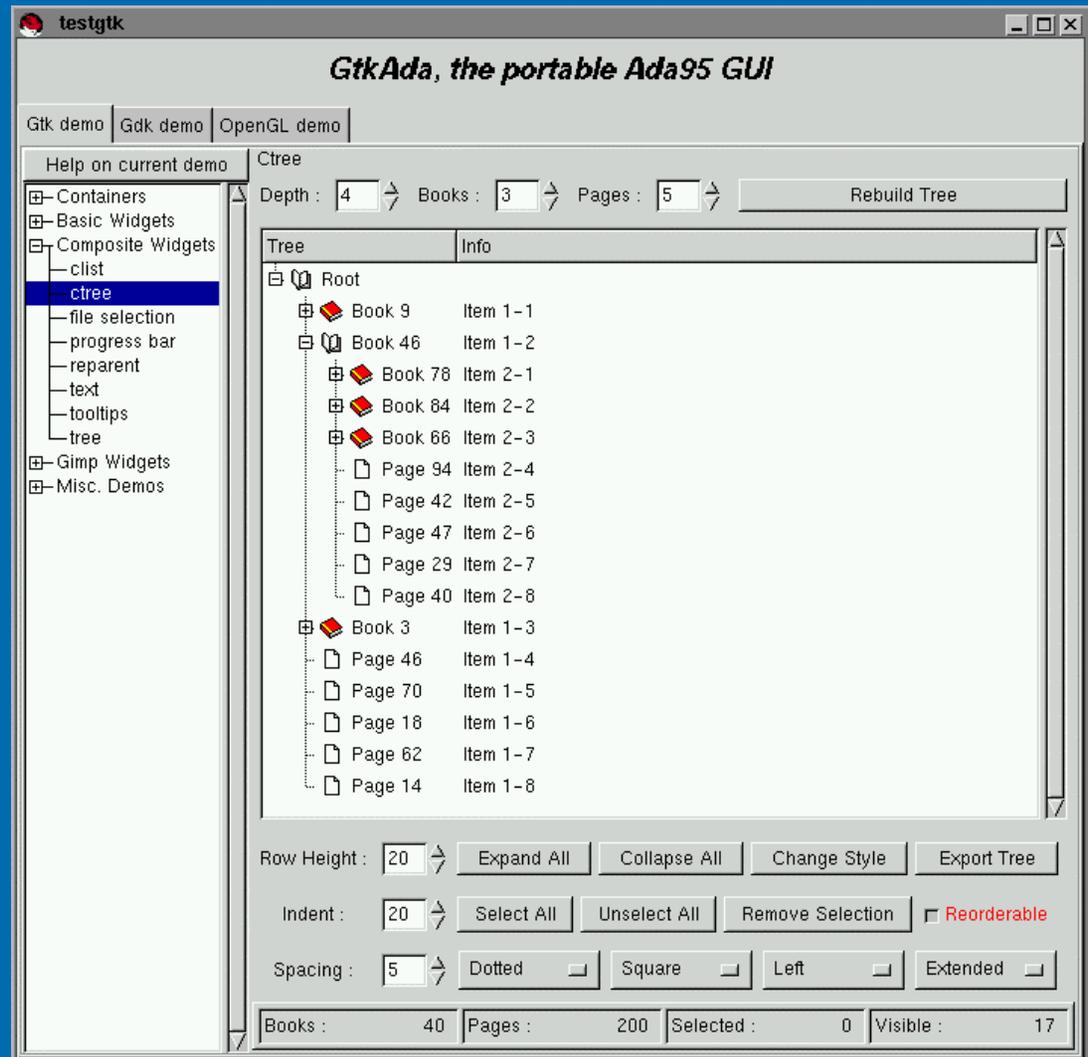
GtkAda - Screenshots (2)

✓ **OpenGL**

Why Gtk+?

- ✓ **Portable**

- ✓ **Native**

- ✓ **Extensible**

- ✓ **Open Source**

- ✓ **Actively developped**

- ✓ **Thread-safe**

Why Gtk+? (2)

- ✓ **High level widgets**

- ✓ **Easy to use scrolling capabilities**

- ✓ **Wide range of users**

- ✓ **Very powerful layout capabilities**

  – **Complete set of containers**

- ✓ **Powerful and easy to use GUI builder**

- ✓ **Easy to bind**

The GUI Builder

✓ **Multi-language GUI builder**

– **Language-independent save file (XML).**

– **Code-generation specific to each language.**

– **Dynamic loading of XML files**

Why GtkAda?

- ✓ **Compile time checks**

- ✓ **Real Object Oriented hierarchy**

- ✓ **Code readability**

- ✓ **Group most popular Gtk widgets**

- ✓ **Can create new types and extend widgets**

- ✓ **More extensive documentation**

GtkAda vs Gtk+: Example

```
#include <gtk/gtk.h>


void main (int    argc,
            char *argv[])
{
  GtkWidget *window;
  GtkWidget *button;

  gtk_init (&argc, &argv);
  window = gtk_window_new (GTK_WINDOW_TOPLEVEL);
  gtk_container_set_border_width
    (GTK_CONTAINER (window), 10);
  button = gtk_button_new_with_label ("Hello World");
  gtk_container_add (GTK_CONTAINER (window), button);
  gtk_widget_show_all (window);
  gtk_main ();
}
```

```
with Gtk.Main;    use Gtk.Main;
with Gtk.Button;  use Gtk.Button;
with Gtk.Window;  use Gtk.Window;
with Gtk.Enums;   use Gtk.Enums;

procedure Hello is
    Window : Gtk_Window;
    Button : Gtk_Button;
begin
    Init;
    Gtk_New (Window, Window_Toplevel);
    Set_Border_Width (Window, 10);

    Gtk_New (Button, "Hello World");
    Add (Window, Button);
    Show_All (Window);
    Main;
end Hello;
```

Generic widget

Unique namespace

No dispatching

Typed Widget

Use clauses

Built-in OO

Ada Core
TECHNOLOGIES,INC

ACT

GUI Builder Support with GtkAda

✓ **Support built-in**

✓ **Takes advantage of object-oriented features.**
  – **Each widget generates its own code**

✓ **Generated sources**
  – **easy to read sources (generates new composite widgets)**
  – **sources can be modified and regenerated**

GtkAda: a complete GUI Solution

✓ **Portable**

✓ **Native**

✓ **Easy to use**

✓ **Safe**

✓ **Object Oriented**

✓ **Comes with a GUI builder**

✓ **Same look-and-feel**

Ada Core
TECHNOLOGIES.INC

ACT