A New File Selector for GTK+

Owen Taylor
Red Hat, Inc
otaylor@redhat.com

FOSDEM, 9 February 2003

Outline

GTK+-2.4 overview
The old file selector
Other file selectors
Fitting into the environment
API considerations
Conclusions
API sketch
Other 2.4 features

GTK+ History

Development begins - Mid 1996

1.0 - April 1998

1.2 - February 1999

DND, Themes, GtkCTree, separate GLib

2.0 - March 2002

GtkTextView, GtkTreeView, Win32 port,

GObject, Pango, ATK

2.2 - December 2002

Multihead, Xft2

GTK+ 2.4

Schedule

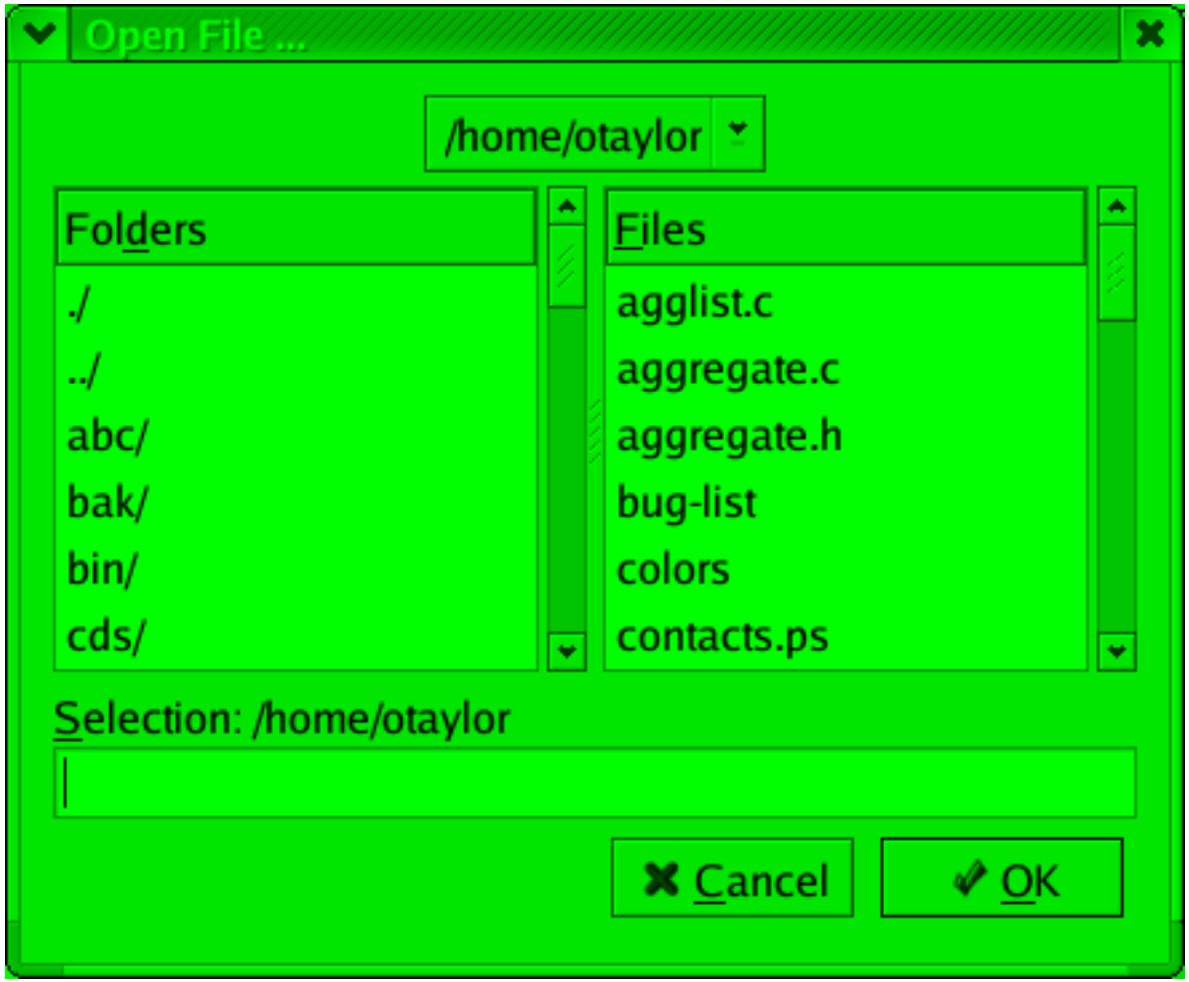
- 1 January - Start of development
- 1 July - Feature freeze
- 1 August - API/ABI freeze
- 1 September - Release

Possible features

- New file selector
- New Menu API
- Enhanced toolbar
- Replacement for GtkCombo and GtkOptionMenu
- Height-for-width geometry management
- Disclosure triangle widget

`http://gtk.org/plan/2.4`

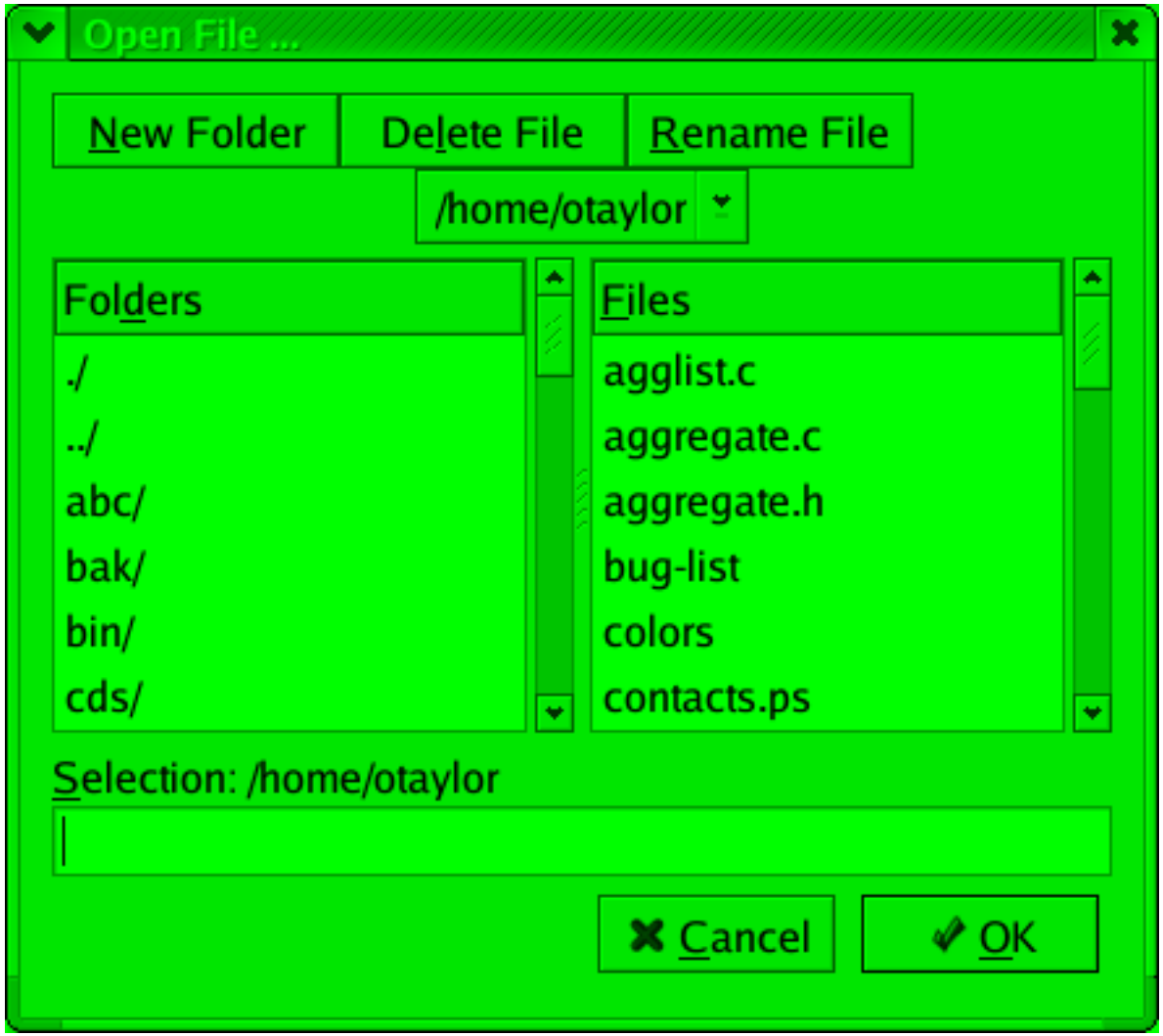
GtkFileSel



Features

- Dropdown for parent directories
- Directory list
- File list

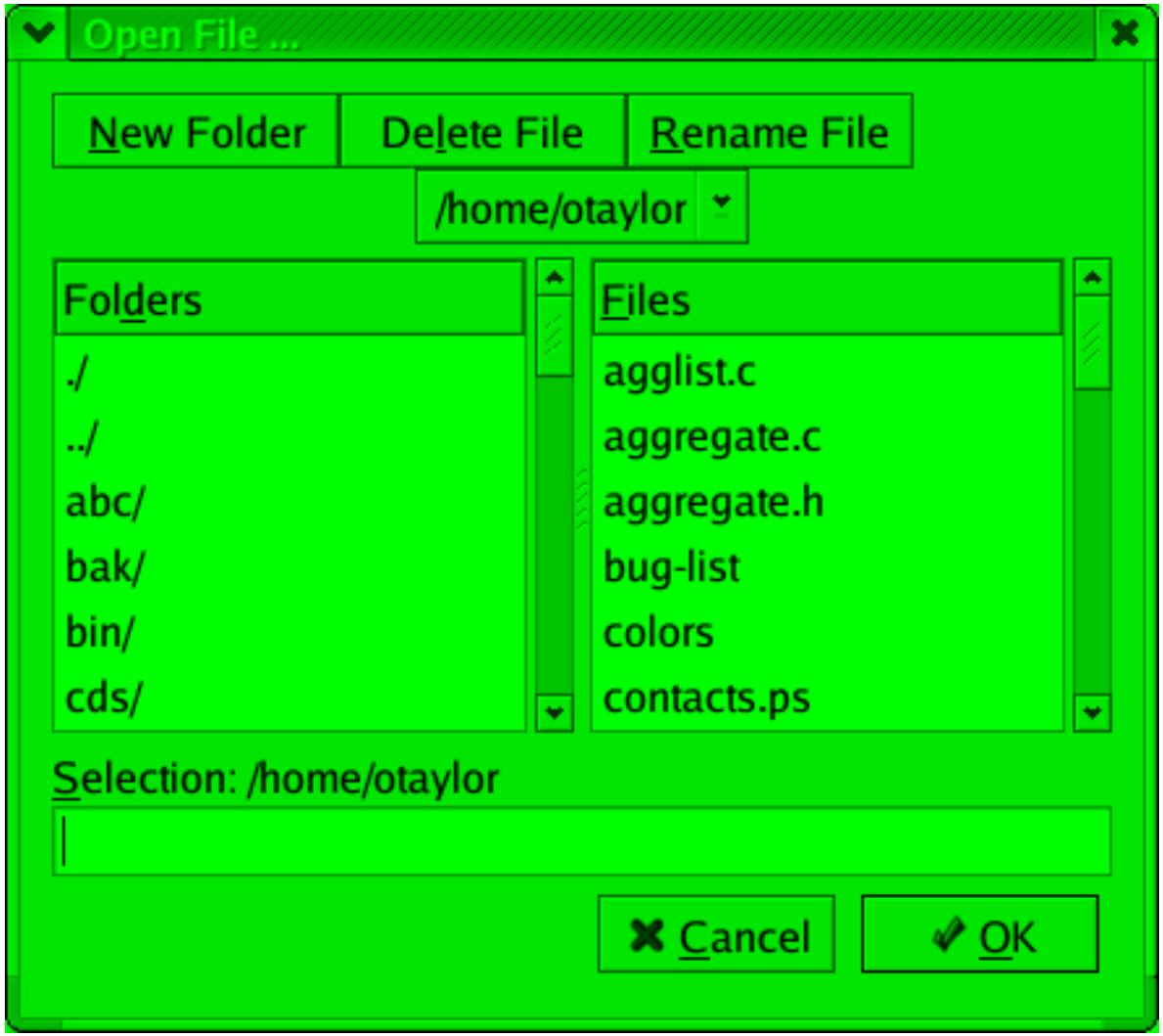
GtkFileSel



Features

- Dropdown for parent directories
- Directory list

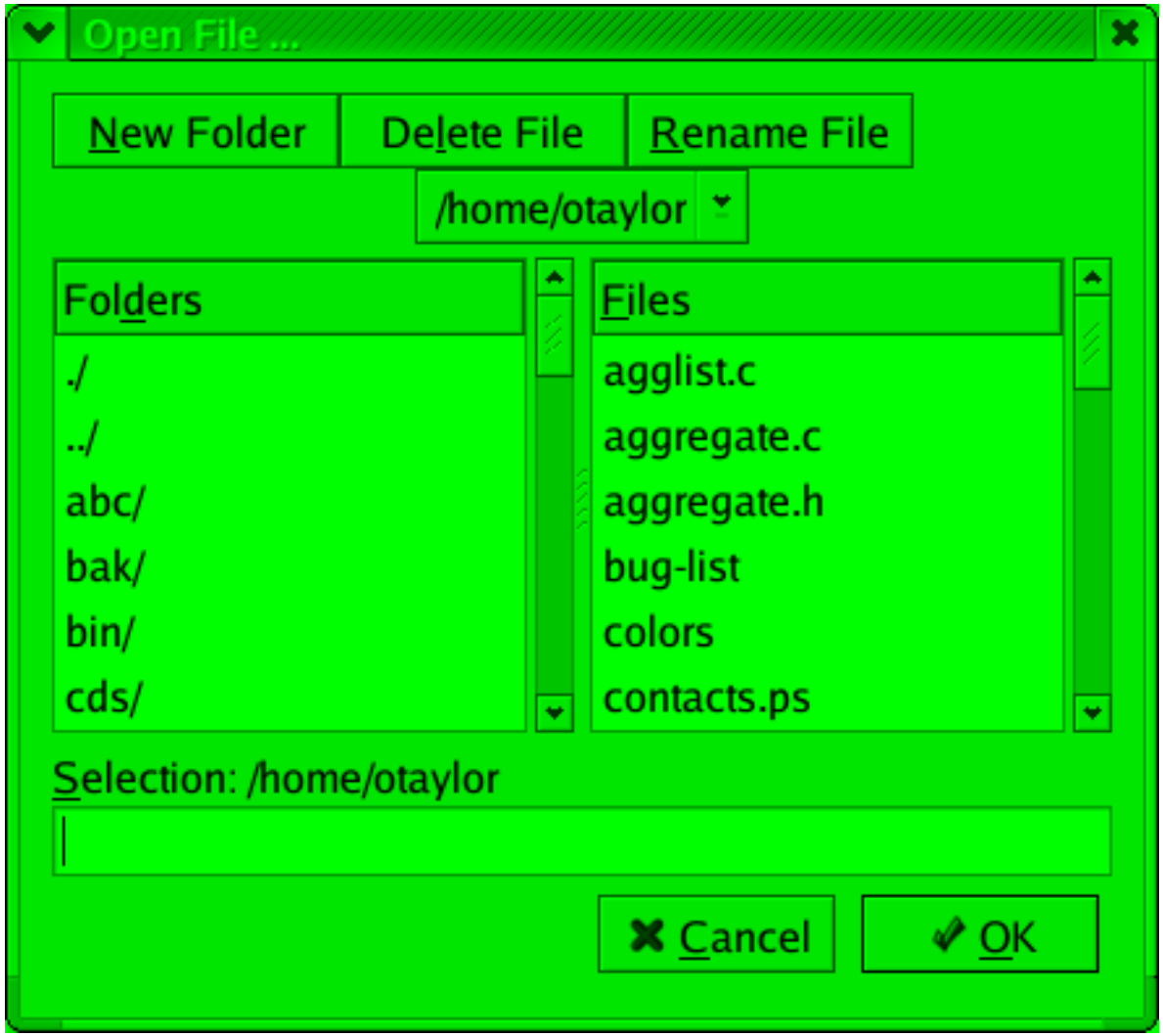
GtkFileSel



Bad

- No way to get to frequent used locations
- No obvious filtering features

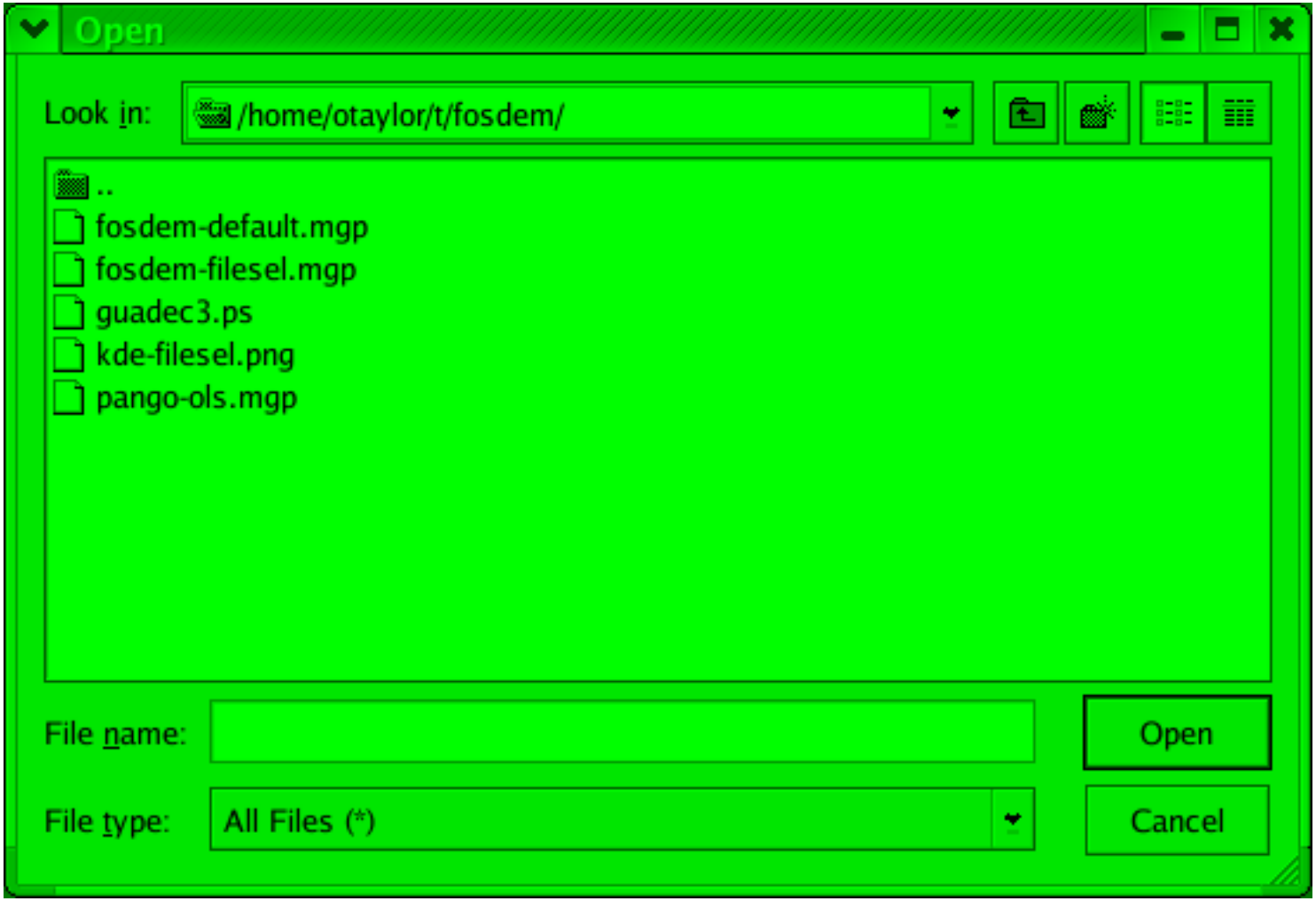
GtkFileSel



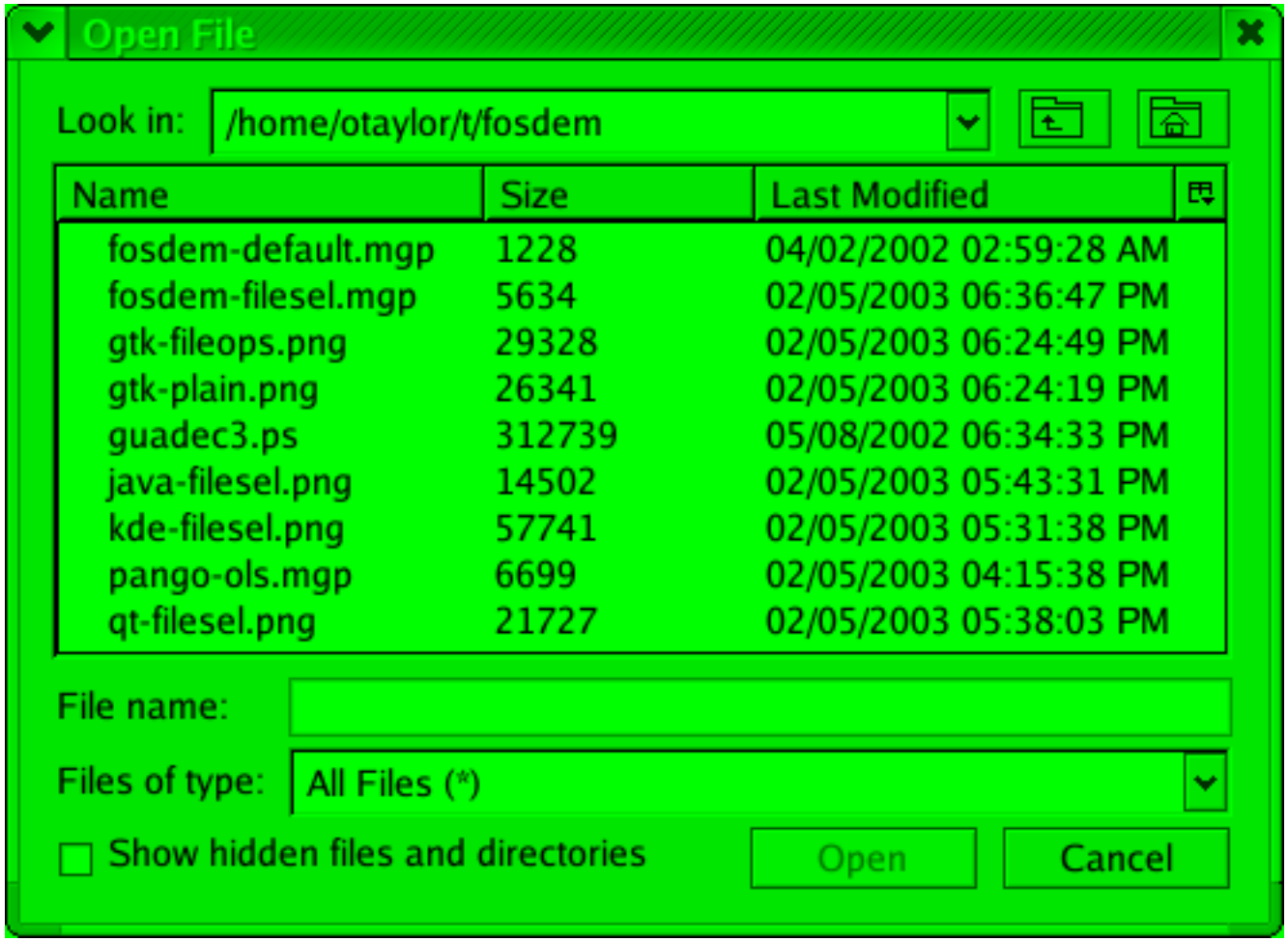
Good

- Simple
- Powerful tab completion

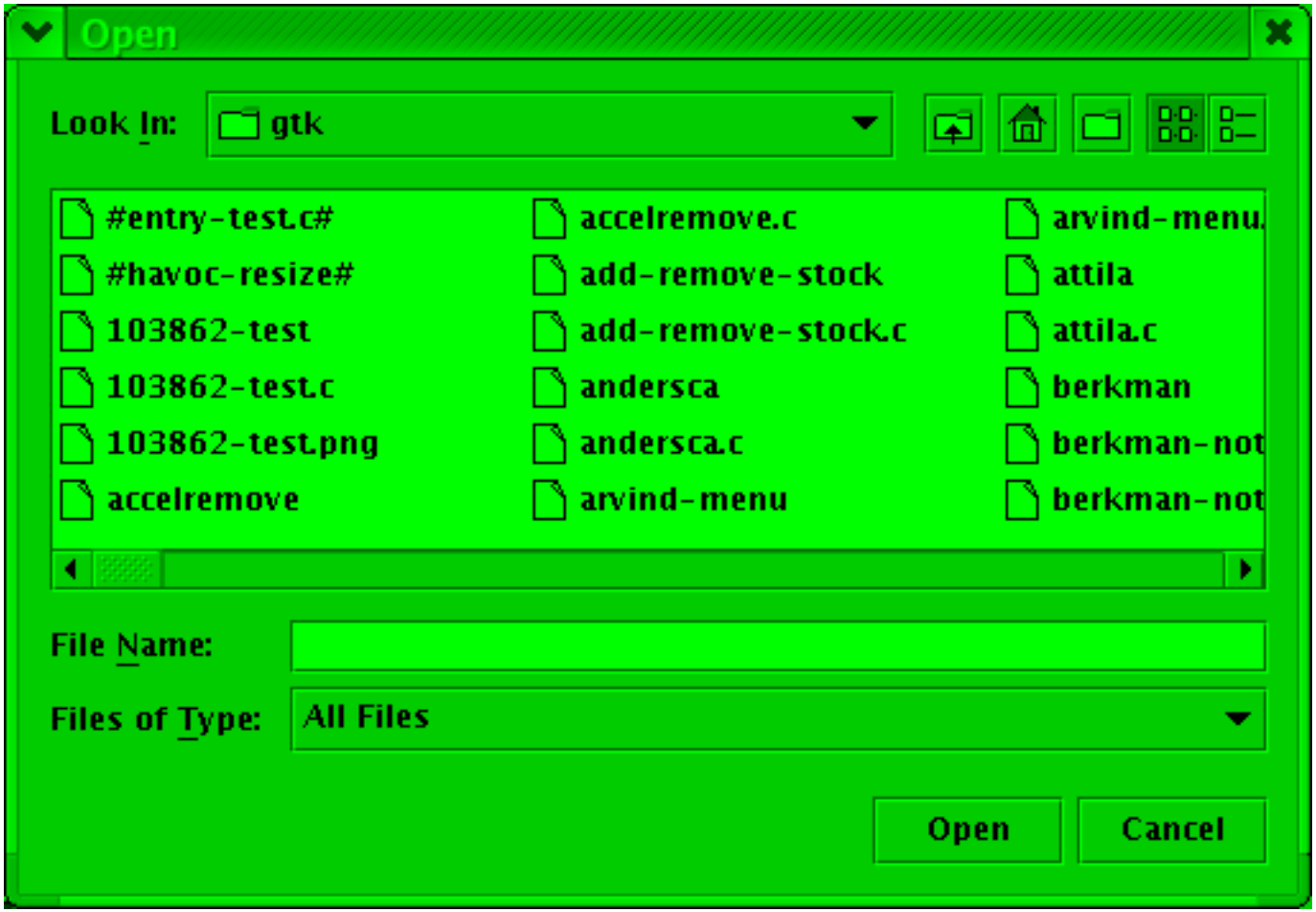
Qt File Selector



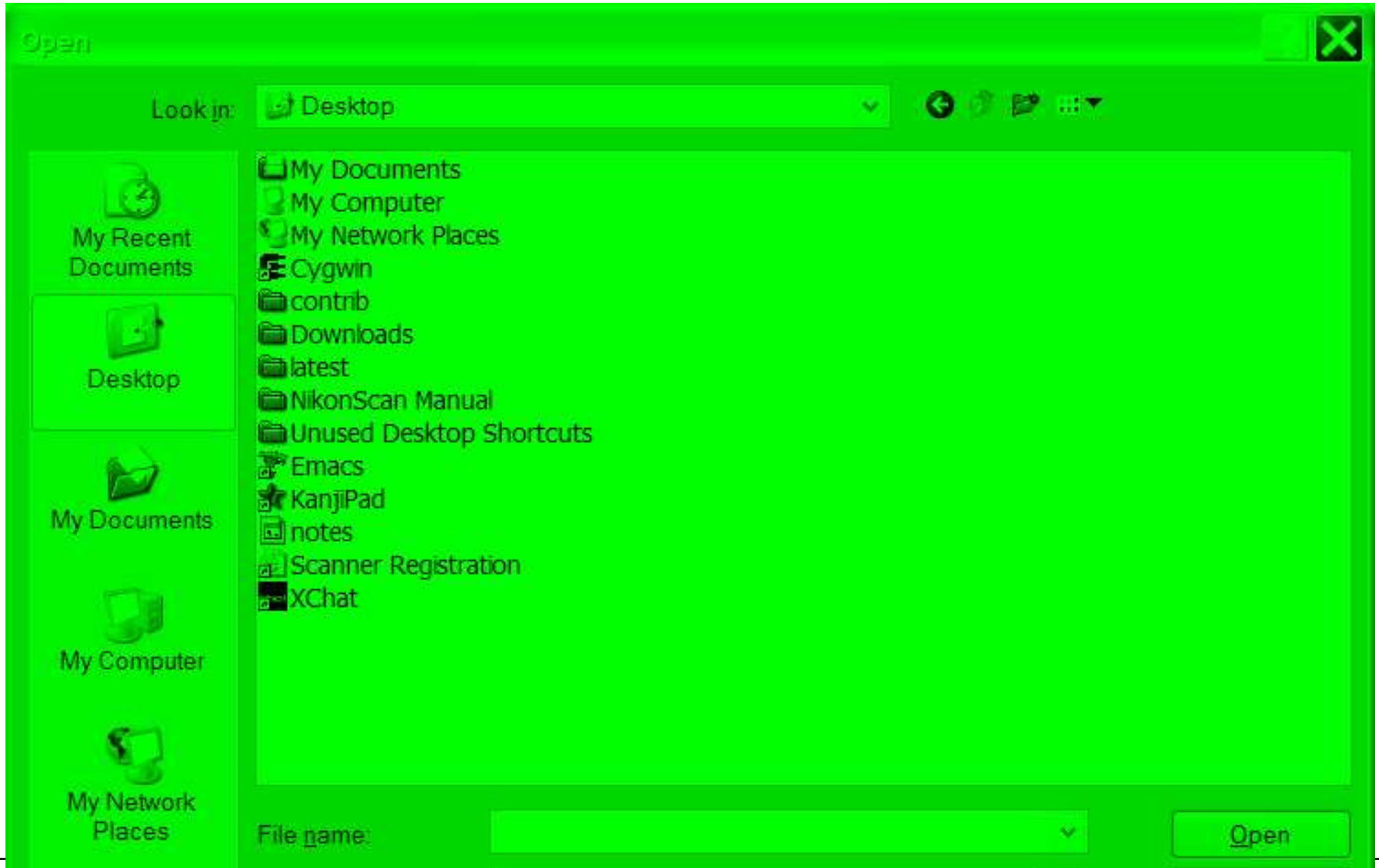
Mozilla File Selector



Java File Selector



Windows XP File Selector

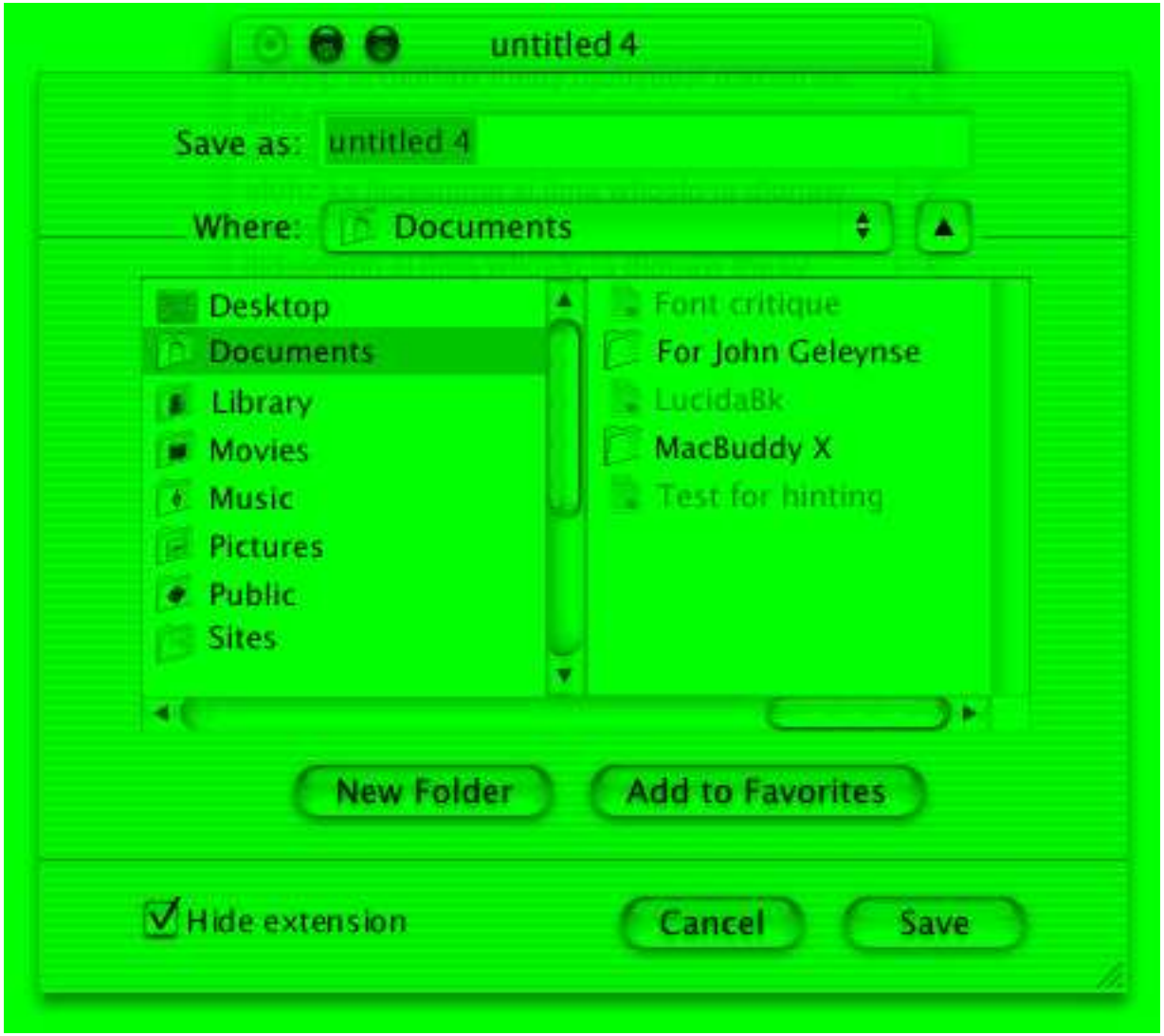


KDE File Selector

OS X file selector

OS X file selector

OS X file selector



Loading versus Saving

Loading

- Previews

Saving

- Create new folders
  Type in new file name
- Often don’t need to navigate filesystem

Allow using platform native widgets?

Disadvantages

- Are frequently modal only
- Won’t derive from GtkDialog
- Don’t allow customization with embedded GTK+ widgets
- Will cause inconsistency between when we can use them and when we can’t

No platform widgets

Allow platform customized file selectors

- Use a similar layout to native selector
- Pick up native icons, thumbnails, filetypes

API Issue - Virtual filesystems

Virtual file system

ftp:, http:

Lack of standardization of URI schemes a problem

KDE - fish:
GNOME - ssh:

Can implement gnome-vfs support as a plugin

But plain GTK+ apps won’t under gnome-vfs URIs

Need:

gtk_file_chooser_set_understands_uris()

API Issue - Desktop integration

Do we want to allow replacement of the GUI

- Implement filesel in Nautilus process?
- Filesel customized to look like Win32, OS X

Do we need integration?

- File selector is modal
- File manager allows full set of operations

Need to get:

- Favorite / recent locations
- Icons
- Mime types
- VFS integration

API Issue - Application customization

Custom preview widget

Custom thumbnails

File type selection for saving

Random controls
- Open-readonly

Things that reduce customizability
- Allowing total desktop integration
- Using native file selector (completely)

API Issue - static methods?

```
response = gtk_file_chooser_open_file ("Title");
```

vs:

```
GtkWidget *chooser = gtk_file_chooser_dialog_new (GTK_FILE_CHOOSER_OPEN);
if (gtk_dialog_run (GTK_DIALOG (chooser) == GTK_RESPONSE_OK))
  {
    name = gtk_file_chooser_dialog_get_name (GTK_FILE_CHOOSER (chooser));
  }
gtk_widget_destroy (chooser);
```

Shorter, but API more complex

Conclusions

No native dialogs

Allow plugin file systems

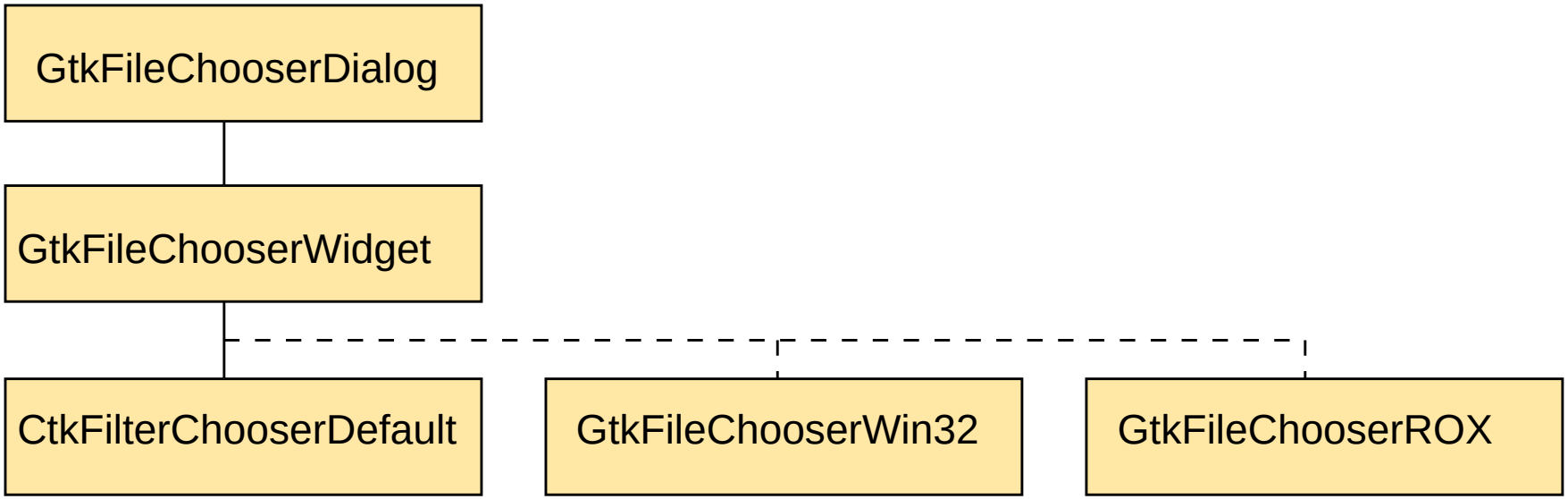
Think about replacement widgets for future

No static methods

Application customization

- Custom preview widget
- Pack extra widgets

GtkFileChooser



GtkFileChooser

Same set of methods for

- Dialog
- Embeddable widget
- Replacement widgets
- "Lasagna API"

Use an interface - GtkFileChooser

Methods:

- select_uri
- deselect_uri
- add_filter
- remove_filter
- add_accessory_widget
- ...

Filter objects

Standard types:

- Filename
- Mime type
- Callback

Associated human readable name

```
GtkFileFilter *filter = gtk_file_filter_new ();
gtk_file_filter_add_mimetype (filter, "image/jpeg");
gtk_file_filter_add_mimetype (filter, "image/png");
gtk_file_filter_add_mimetype (filter, "image/x-xcf");
gtk_file_filter_set_label (filter, "Images (JPEG,PNG,XCF)");

gtk_file_chooser_add_filter (chooser, filter);
```

File system object

Loaded from dynamic module

Works entirely in terms of URIs

Operations

- List roots
  Iterator for walking heirarchy
- Create folder
  Get details (mime type, file size, modification date)
- Get icon

Preview API

To set custom preview widget:

gtk_file_chooser_set_preview_widget()

When get ::update-preview signal, either:

Have preview for file

```
gtk_file_chooser_set_has_preview (chooser, TRUE);
[ Update chooser]
```

No preview for file

```
gtk_file_chooser_set_has_preview (chooser, FALSE);
```

Where from here

Convert API sketch into header files

Get first implementation into GTK+

- ? Reuse of code from EggFileSelector (Bastien Nocera) ?

Get UI design (GNOME UI team)

Implement UI design

Iterate until everybody is minimally unhappy

Release

New ComboBox - Kristian Rietveld

Get rid of the GtkList usage

(GtkList => GtkCList => GtkTreeView)

Unify GtkCombo, GtkOptionMenu

Appearance should be theme issue

Handle things like color, fill style combos

Grids of choices

(Involves adding grid menus)

Completion

New menu/toolbar API (jhmenu) - James Henstridge

Action based ("Print", "Save")

Organization independent from action definition

- Allows rearrangement

Do merging

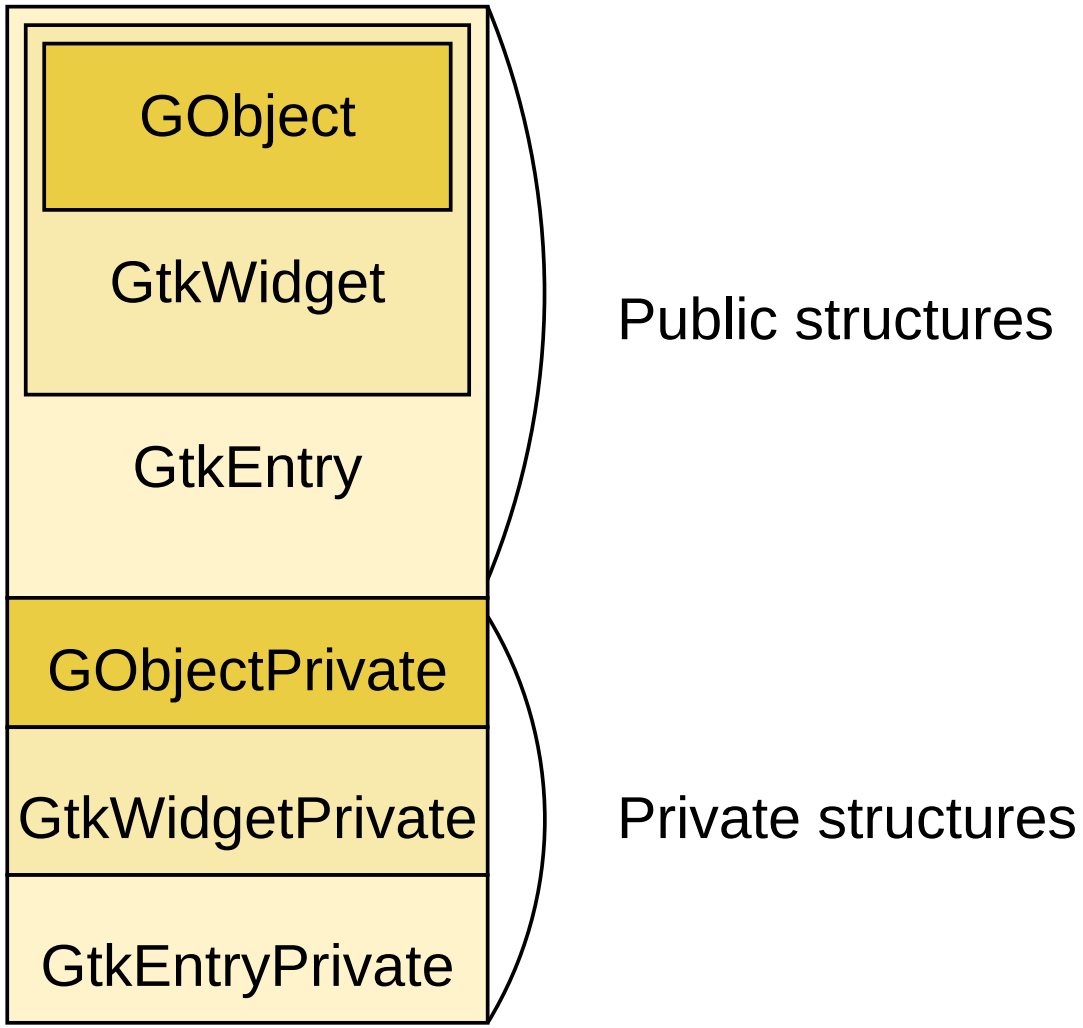
- Merge entry entries into main menu
- Inter-process merging

Improved toolbar (EggToolbar) - Anders Carlsson

Clean up GtkToolbar API

Handle overflows

Miscellaneous improvements

- Non-homogeneous buttons
- Priority text?

Object private data