

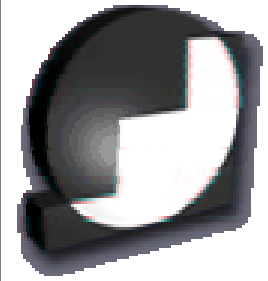


Step Into GNUstep

Adam Fedor

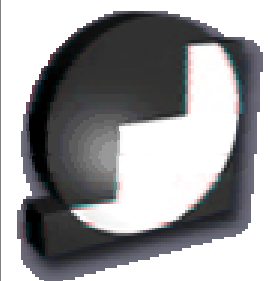
FOSDEM

Feb 16, 2002



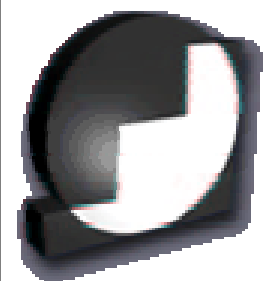
What Is GNUstep?

- An object-oriented development environment
- A well-designed, natural, intuitive graphical interface.
- Provides libraries that give a high-level interface to the GUI and the OS.
- Provides the tools to help create your App.




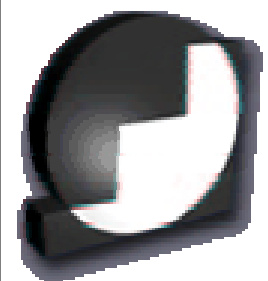
Why Use GNUstep?

- GNUstep is based on a well-established API that has been used and debugged in a commercial environment for over 10 years
- Gives you a high-level starting point upon which to base your application.
- Knowledgeable, responsive developer group.



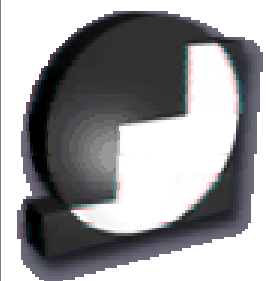
Outline

- 
- History
 - Description of the Components
 - Inner Workings
 - Is it Usable, Useful?
 - Future Work



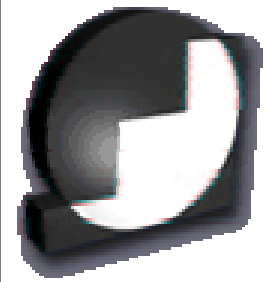
History

- 1993
 - ◆ LibObjCX (Paul Kunz), libobjects (Andrew McCallum)



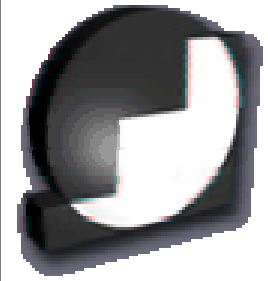
History

- 1993
 - ◆ LibObjCX (Paul Kunz), libobjects (Andrew McCallum)
- 1994
 - ◆ OpenStep Spec



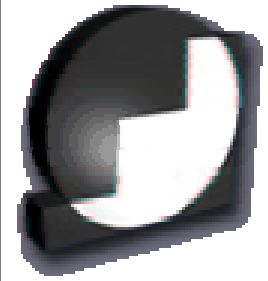
History

- 1993
 - ◆ LibObjCX (Paul Kunz), libobjects (Andrew McCallum)
- 1994
 - ◆ OpenStep Spec
- April 1995
 - ◆ GNUstep formed




History

- 1993
 - ◆ LibObjCX (Paul Kunz), libobjects (Andrew McCallum)
- 1994
 - ◆ OpenStep Spec
- April 1995
 - ◆ GNUstep formed
- April 2001
 - ◆ Base Library 1.0.0



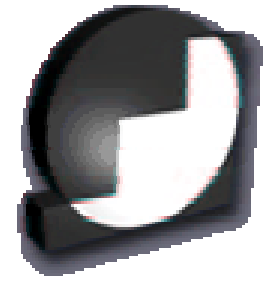
Components

- 
- Makefile Package
 - Base Library
 - GUI Library + Backend
 - IDE (ProjectCenter, Gorm)
 - Applications



Makefile Package

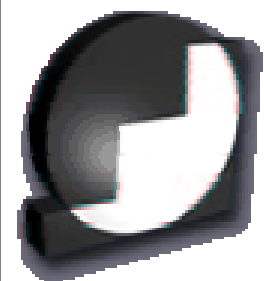
- Encapsulates common make rules
 - ◆ Applications, tools, documentation, libraries, bundles, java apps, frameworks, subprojects, packaging (rpm)
- Encapsulates system (cpu/os) specific information (like libtool).



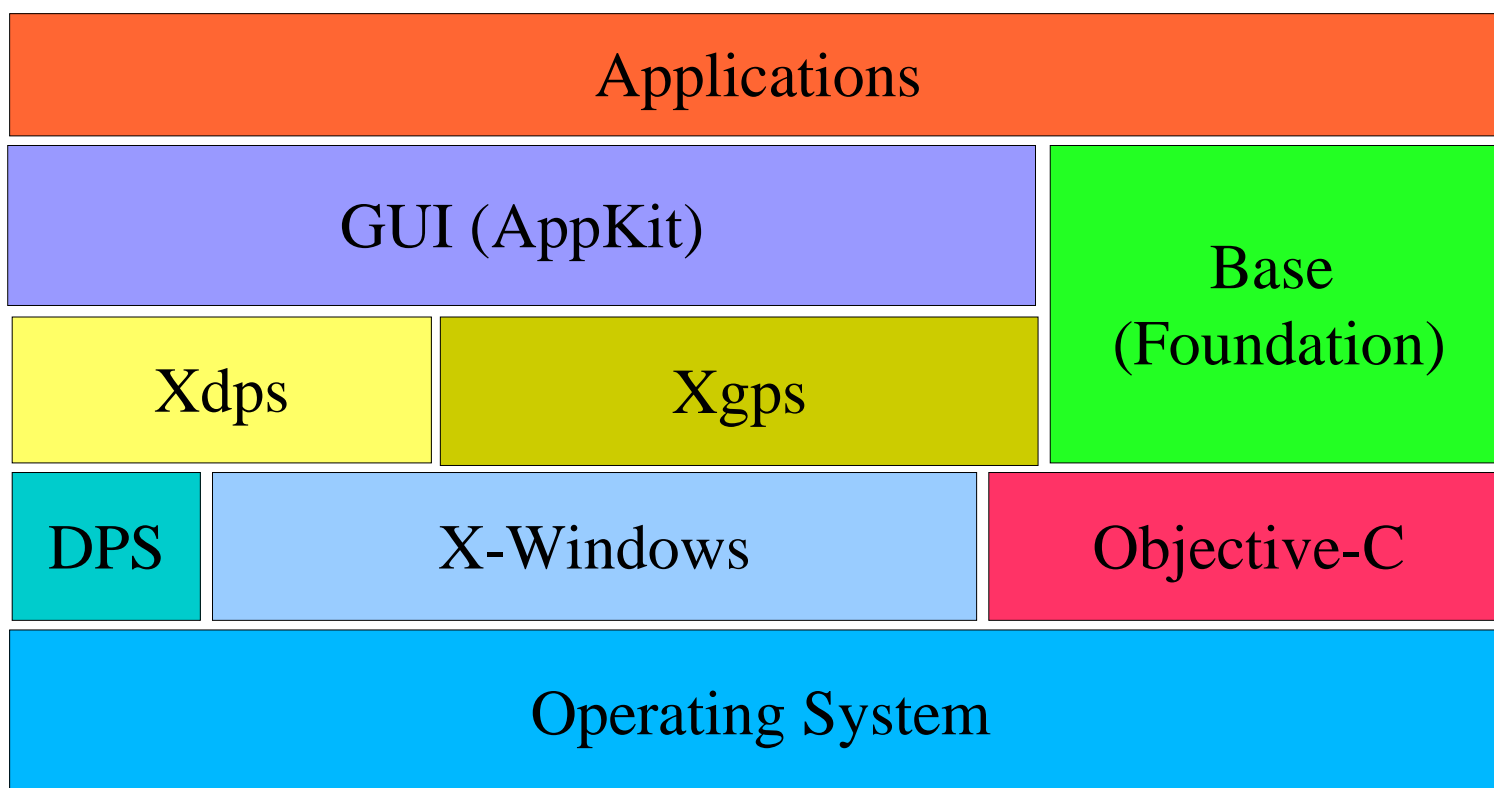
Makefile Package

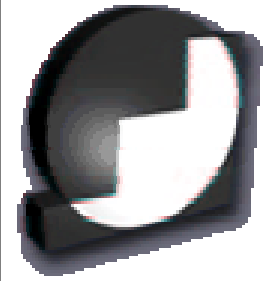
- All you need to compile (debug, cross-platform), install, and package as an rpm (in 8 lines).

```
include $(GNUSTEP_MAKEFILES)/common.make
PACKAGE_NAME=Bob
APP_NAME=Bob
GNUSTEP_INSTALLATION_DIR=$(GNUSTEP_LOCAL_ROOT)/
Bob_MAIN_MODEL_FILE=Bob.gorm
Bob_RESOURCE_FILES= Bob.gorm BobInfo.plist Graffiti.app_Tile.jpg
Bob_OBJC_FILES= main.m DissolveView.m
include $(GNUSTEP_MAKEFILES)/application.make
```



Structure



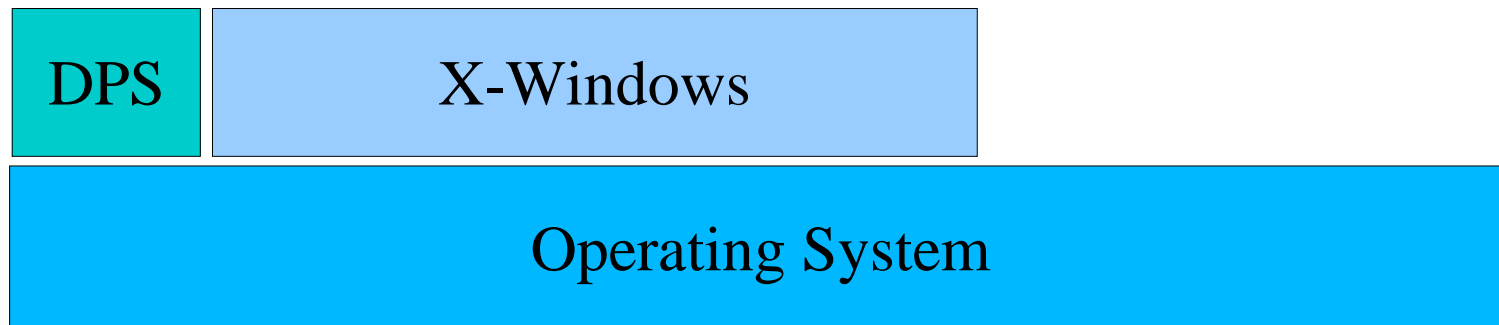


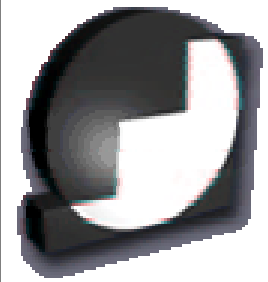
Structure

Choose from:

GNU/Linux, *BSD, Solaris, Debian, Unixware,
(Windows, Darwin).

Uses X-windows for display, uses DPS for drawing on
systems that support it (experimental extension on
Xfree86).





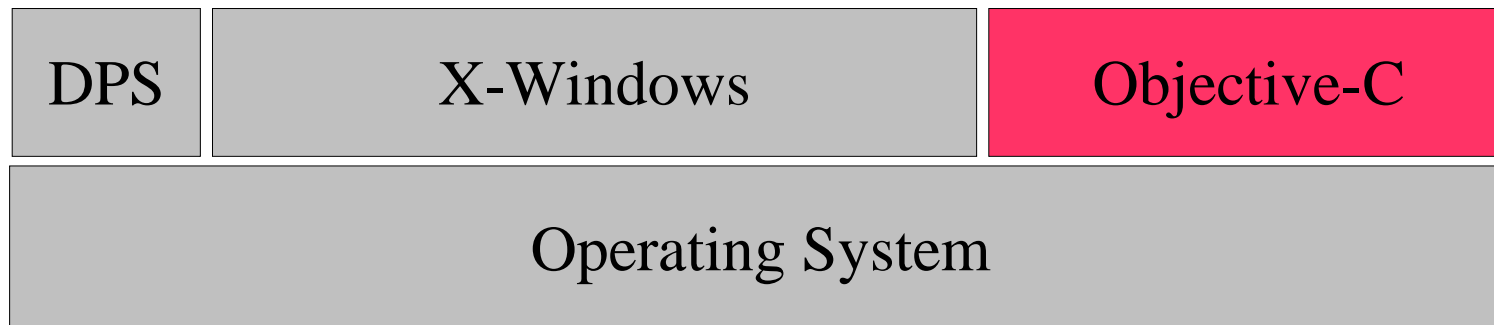
Structure

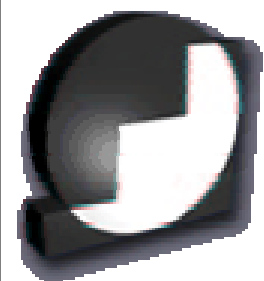
GCC Compiler comes standard with Objective-C support (on most systems).

Complete object-oriented language based on C.

Very simple to learn.

Powerful, but not as 'safe' as strict languages like C++ or Java.

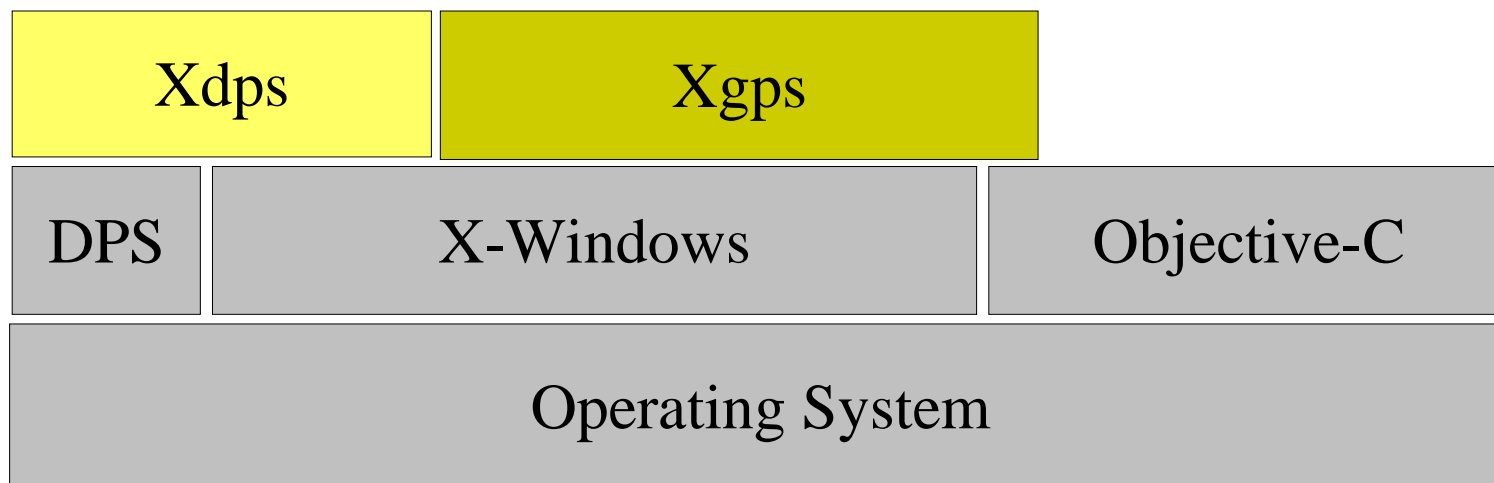


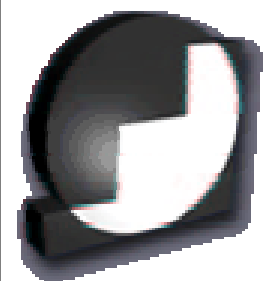


Structure

Backend Library

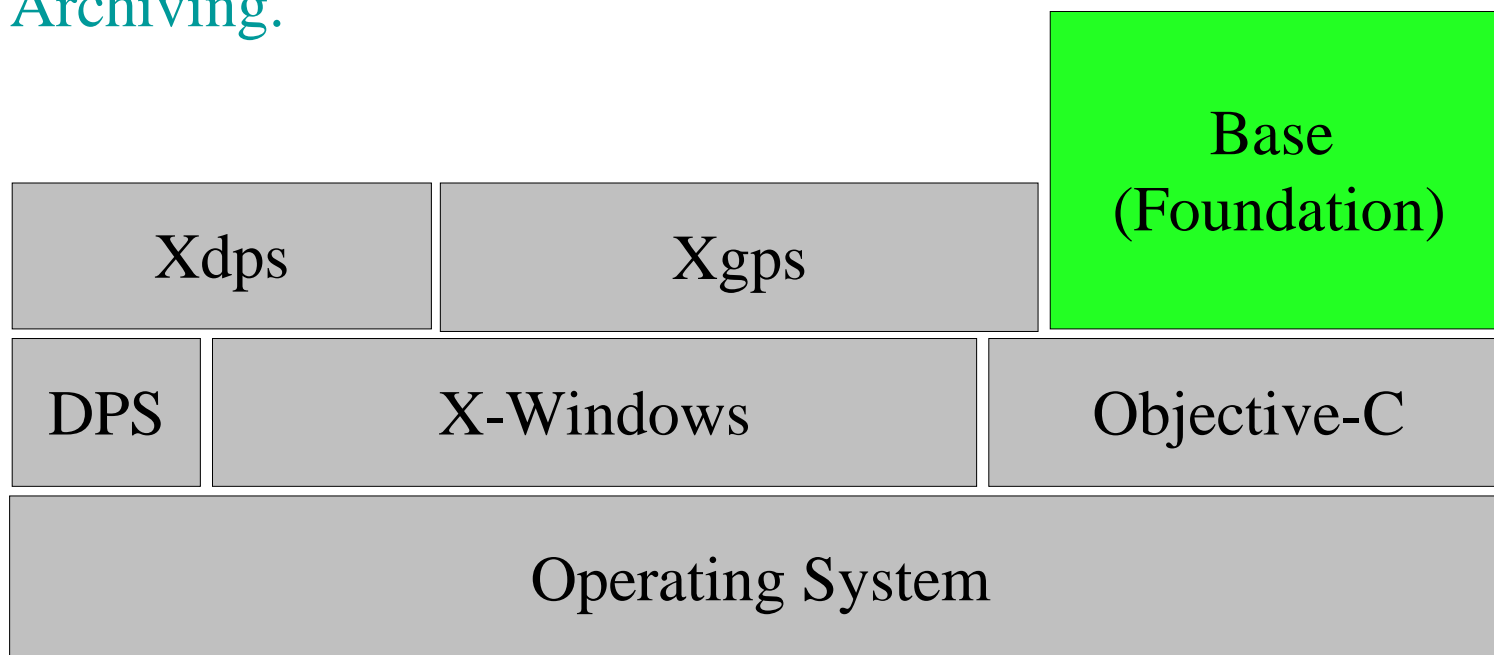
Provides basic event handling, window handling, and drawing specific to a windowing system. Xgps emulates DPS drawing, while Xdps really uses DPS.

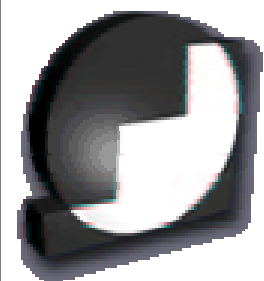




Structure

Abstraction from operating system (file handling, etc).
Exceptions, Remote object calling.
All the basic objects (strings, numbers, dates, arrays).
Archiving.

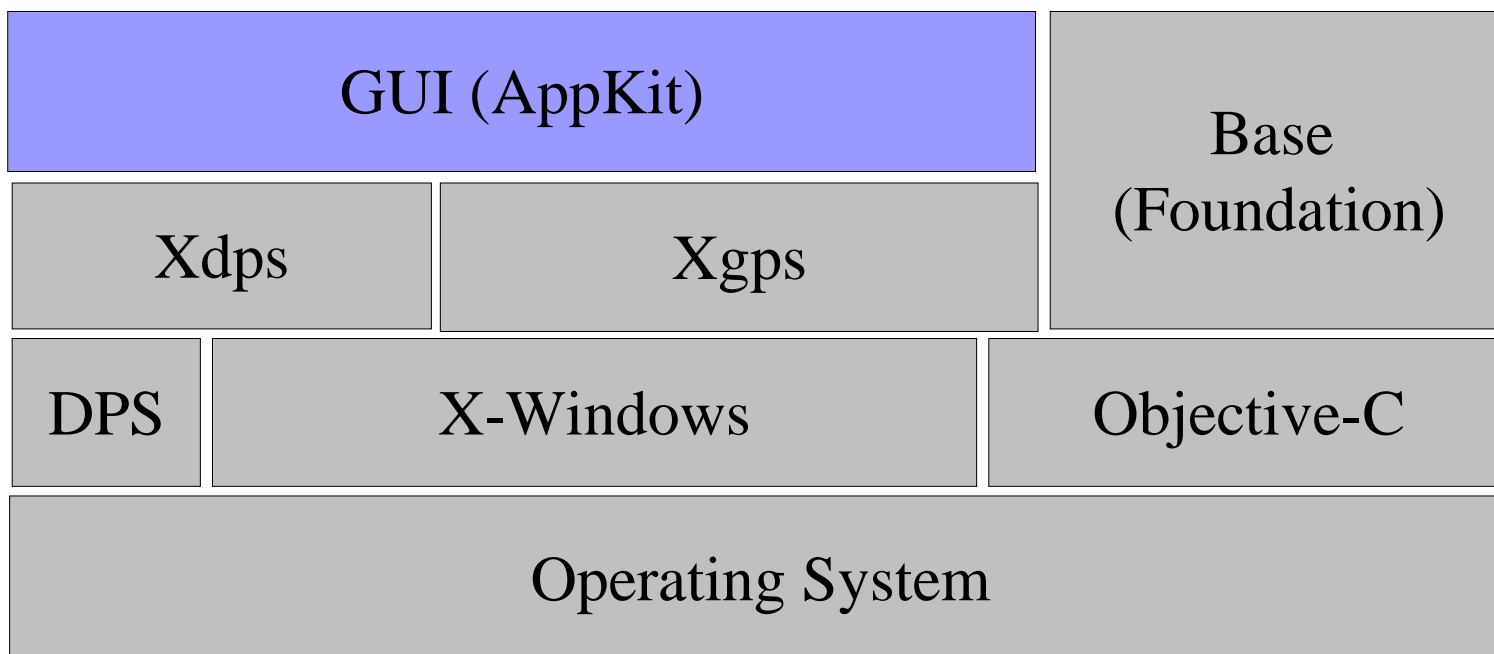


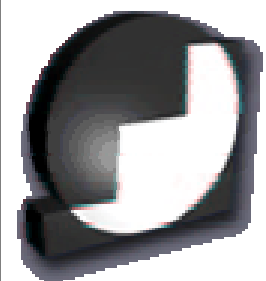


Structure

Display-independent GUI 'widgets'

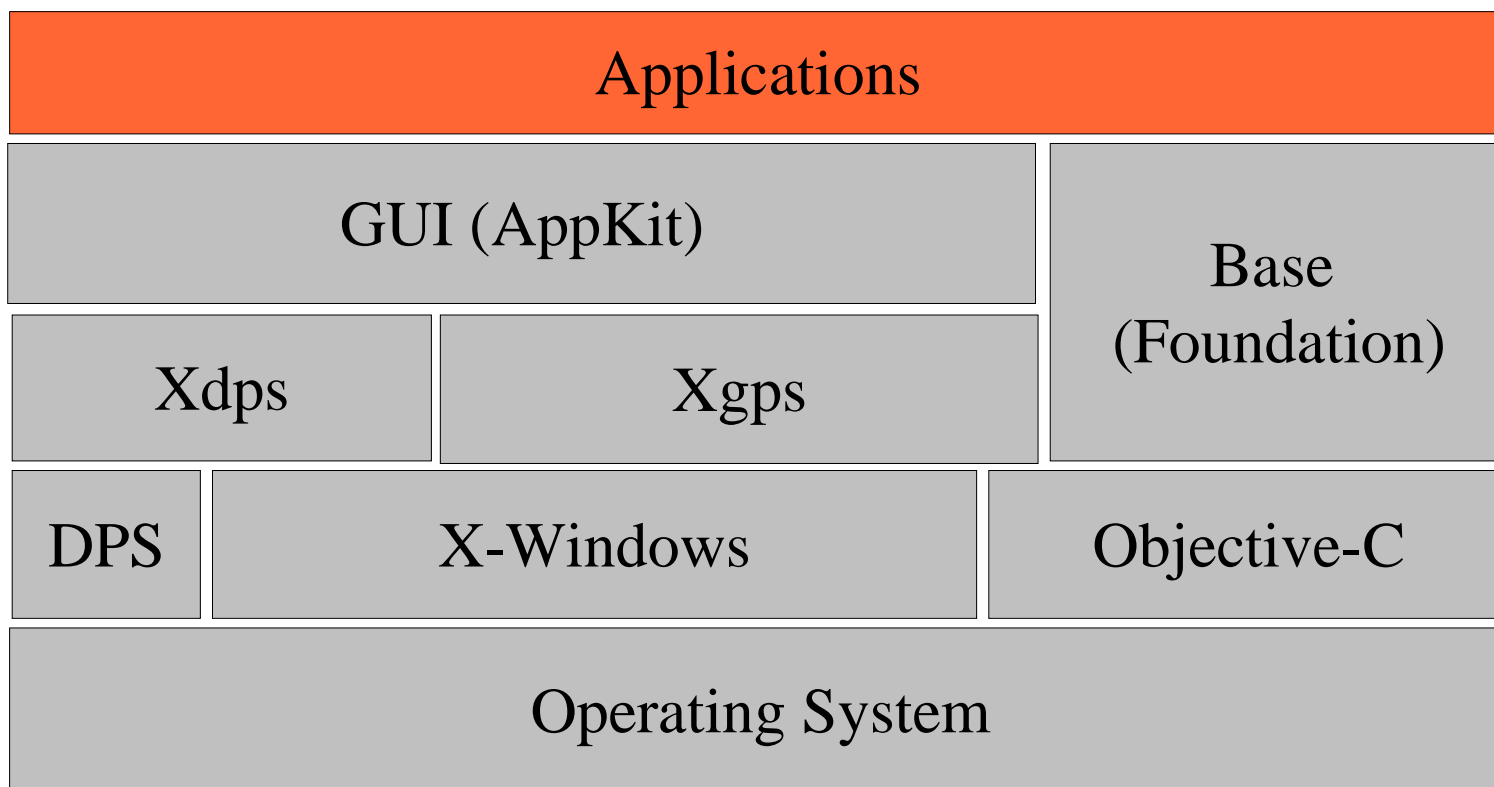
Provides true MVC abstraction (most classes never need to be subclassed).

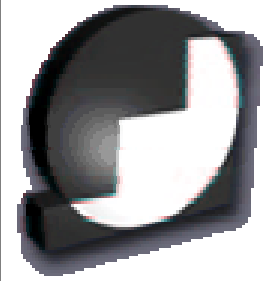




Structure

Whatever you want!





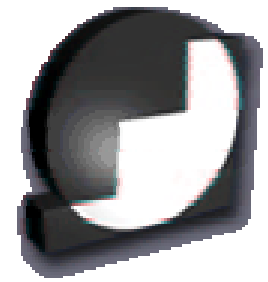
Inner Workings

- Distributed Objects
- Distributed Notifications
- Project Building



Distributed Objects

- Built-in Multiplatform Communication
 - ◆ Across threads
 - ◆ Across applications
 - ◆ Across machines (endiness, type sizes)
- Objective-C naturally handles remote calls
 - ◆ No different than calling a normal object
- Exception handling



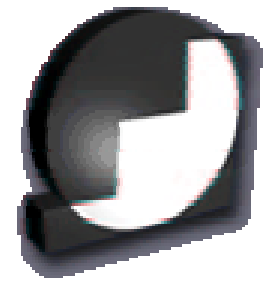
Distributed Objects

Server Setup:

```
NSConnection *connect;  
connect = [NSConnection defaultConnection];  
[connect setRootObject: self];  
[connect registerName: @"GeorgeServer"];
```

Client Setup:

```
server = (id <George>)[NSConnection  
    rootProxyForConnectionWithRegisteredName: @"GeorgeServer"  
    host: @"*"];  
RETAIN(server);
```



Distributed Objects

Server Setup:

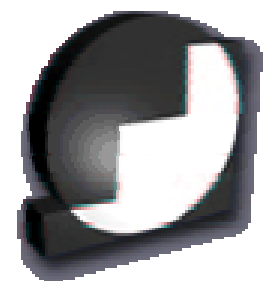
```
NSConnection *connect;  
connect = [NSConnection defaultConnection];  
[connect setRootObject: self];  
[connect registerName: @"GeorgeServer"];
```

Register server name
with DO manager and
opens a socket to listen
to a port.


Asks DO manager to
look for a server with
the given name, returns
a proxy object which
will forward messages
to the server

Client Setup:

```
server = (id <George>)[NSConnection  
    rootProxyForConnectionWithRegisteredName: @"GeorgeServer"  
    host: @"*"];  
RETAIN(server);
```



Distributed Objects

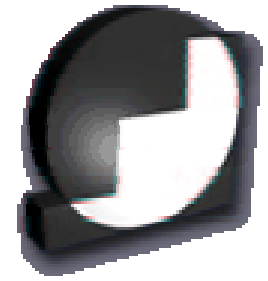


```
@protocol George <NSObject>  
- (void) hereIAm: (NSDictionary *)dict;  
@end
```

Protocol tells client what
● messages server responds
to.

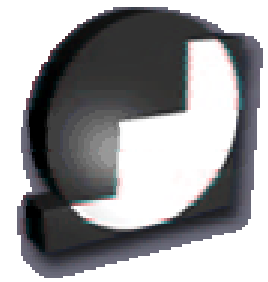
```
dict = [NSDictionary dictionaryWithObjectsAndKeys: name, @"Name",  
                                                [NSValue valueWithPoint: location], @"Location", nil];  
[server hereIAm: dict];
```

● Client just sends a message normally.



Distributed Notification

- Send messages to any/all applications that request it.
- Managed through a notification server.



Distributed Notification

Request a Notification

```
not = [NSDistributedNotificationCenter defaultCenter];  
[not addObserver: self  
    selector: @selector(hereIAm:)  
    name: @"HereIAmNotification"  
    object: nil];
```

Send it.

```
dict = [NSDictionary dictionaryWithObjectsAndKeys: name, @"Name",  
    [NSValue valueWithPoint: location], @"Location", nil];  
  
not = [NSDistributedNotificationCenter defaultCenter];  
[not postNotificationName: @"HereIAmNotification"  
    object: name  
    userInfo: dict];
```



Project Building

■ ProjectCenter

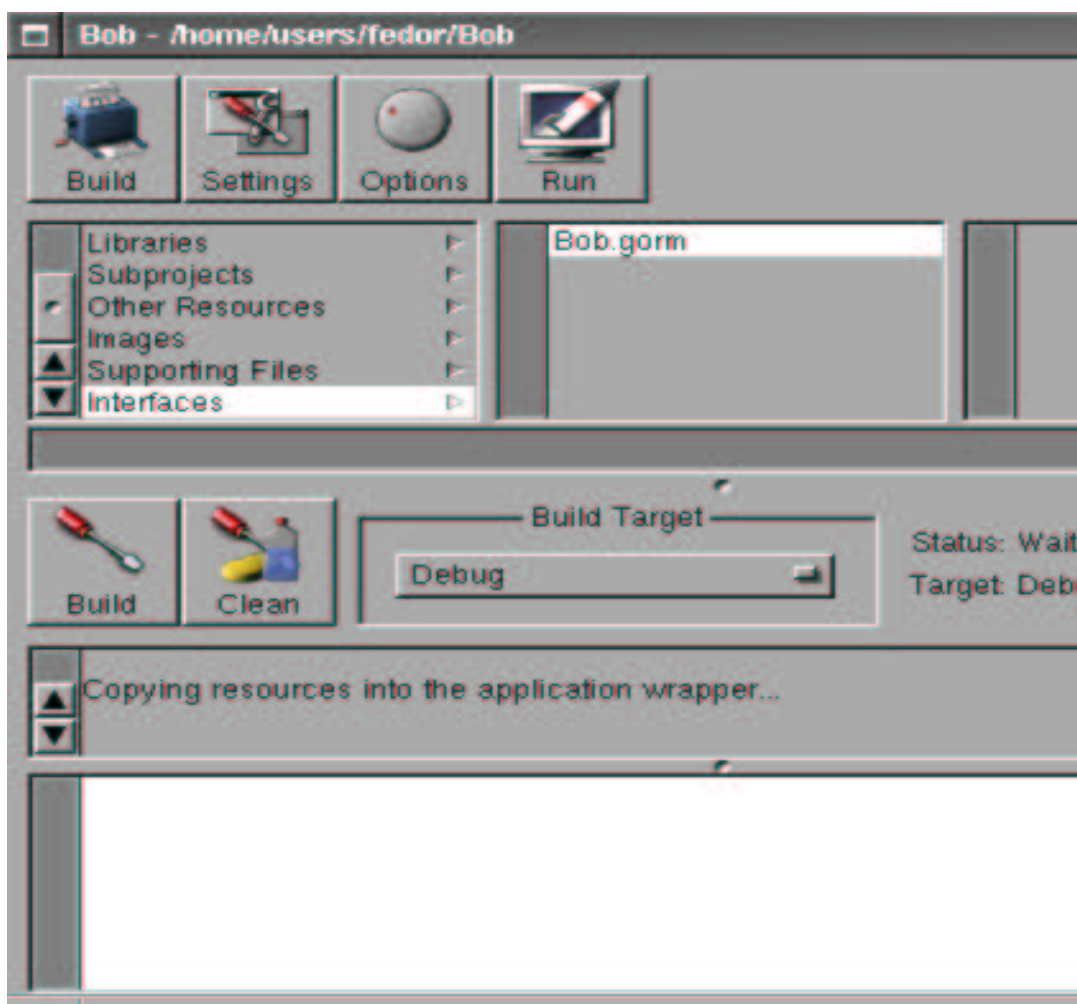
- ◆ Manage applications, bundles, tools, libraries
- ◆ Automatic makefile creation
- ◆ Build, Test, Install, Package

■ Gorm

- ◆ Graphical Interface Builder
- ◆ Drag and drop all graphical elements
- ◆ Connect actions to targets (messages to objects).



Project Center





In other OO GUI systems, when you want a window, you might subclass the window class and add ivars to hold elements you want to display. Adding code to the subclass to control the elements. Thus the subclass is also the controller.

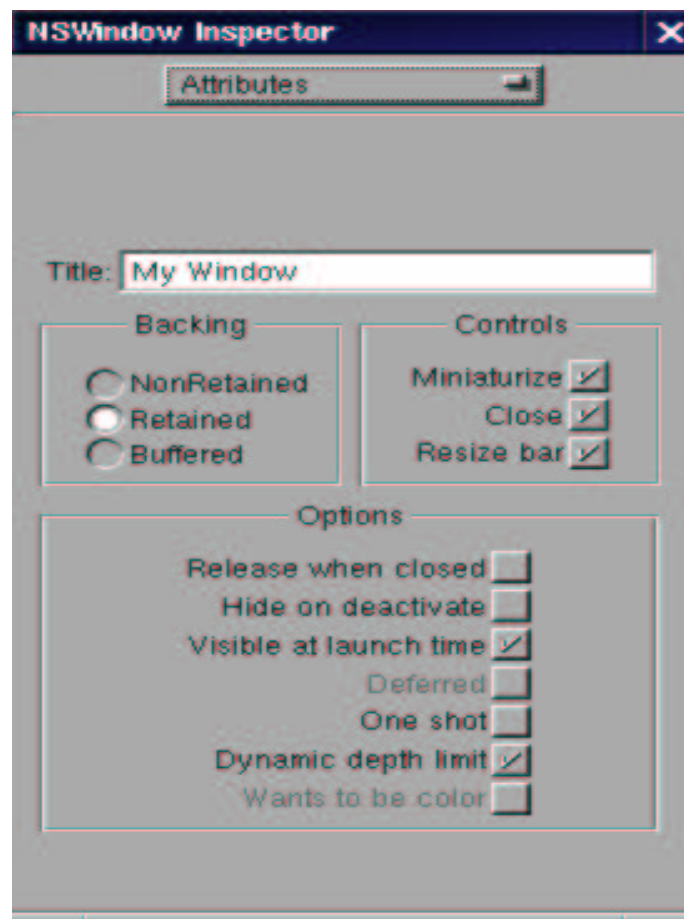
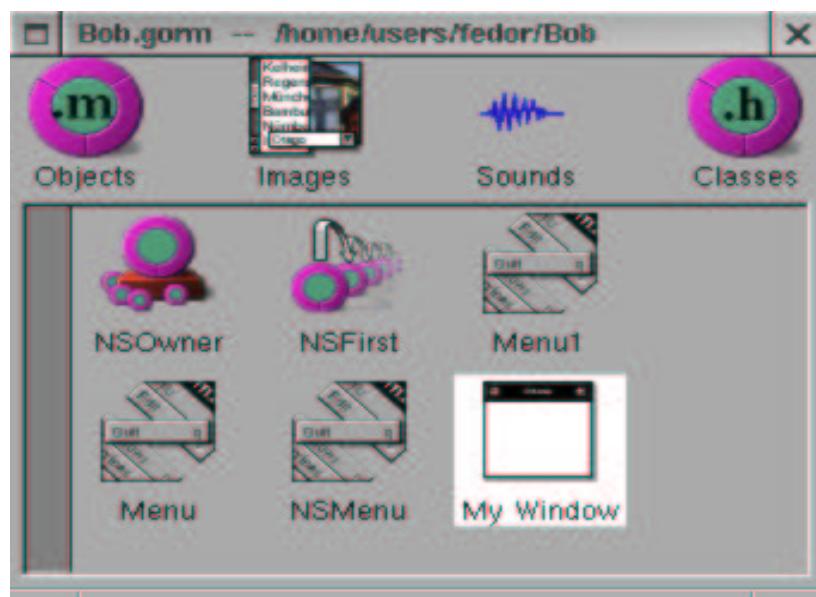
```
Class MyDialog : Dialog
{
    double age;
    string name;

    constructor()
    {
        // initialization code;
    }
    // more controller code....
}
```



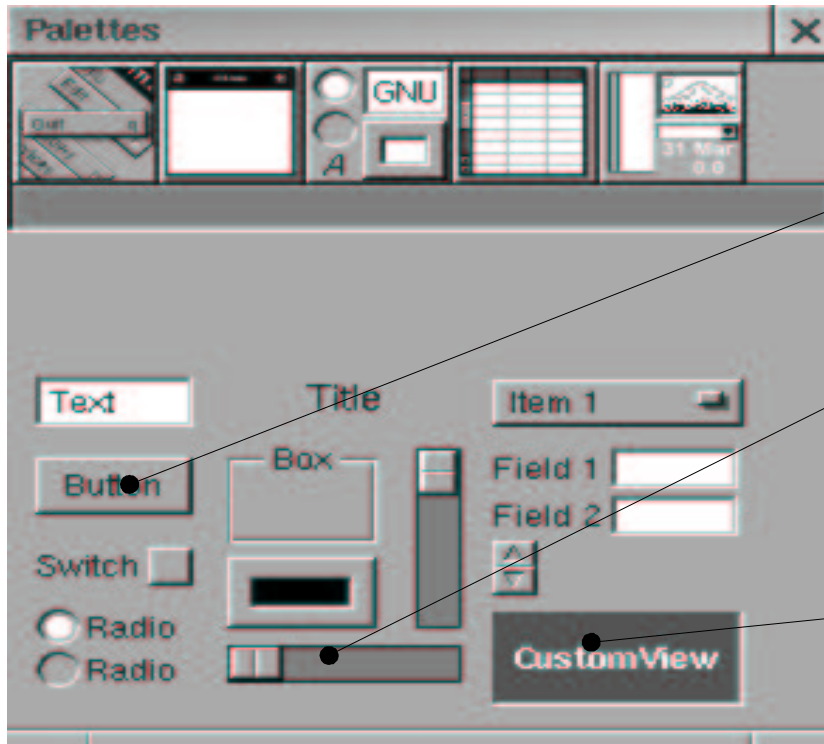
Gorm

In GNUstep, you never subclass a window. The controller is separate and Gorm automatically handles the connections.

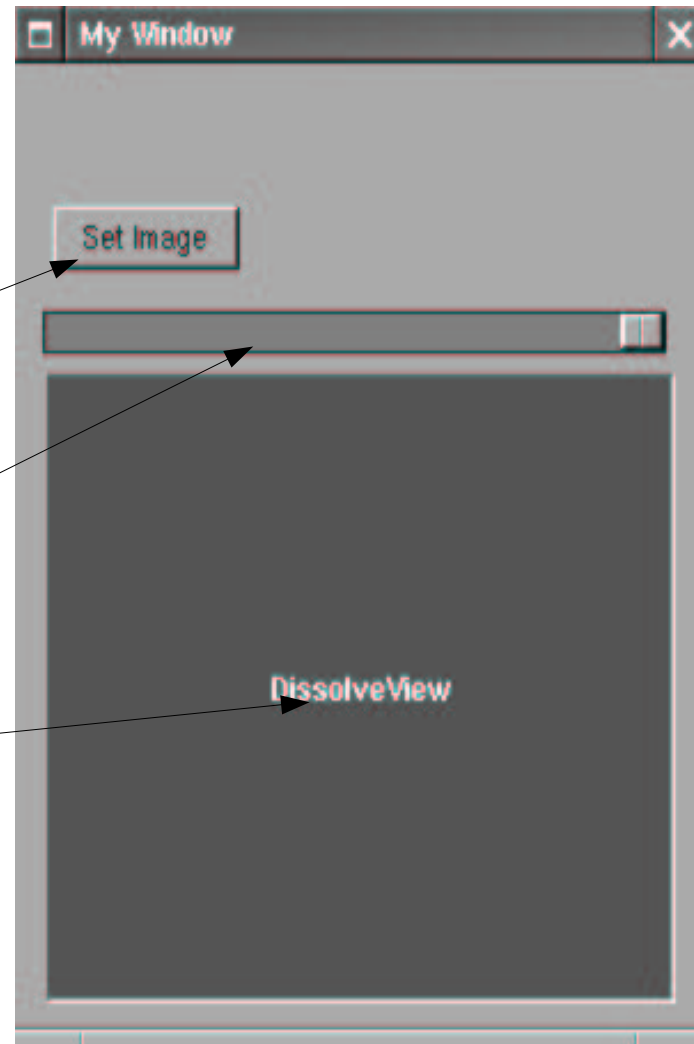




Gorm



Drag and drop elements onto a window.

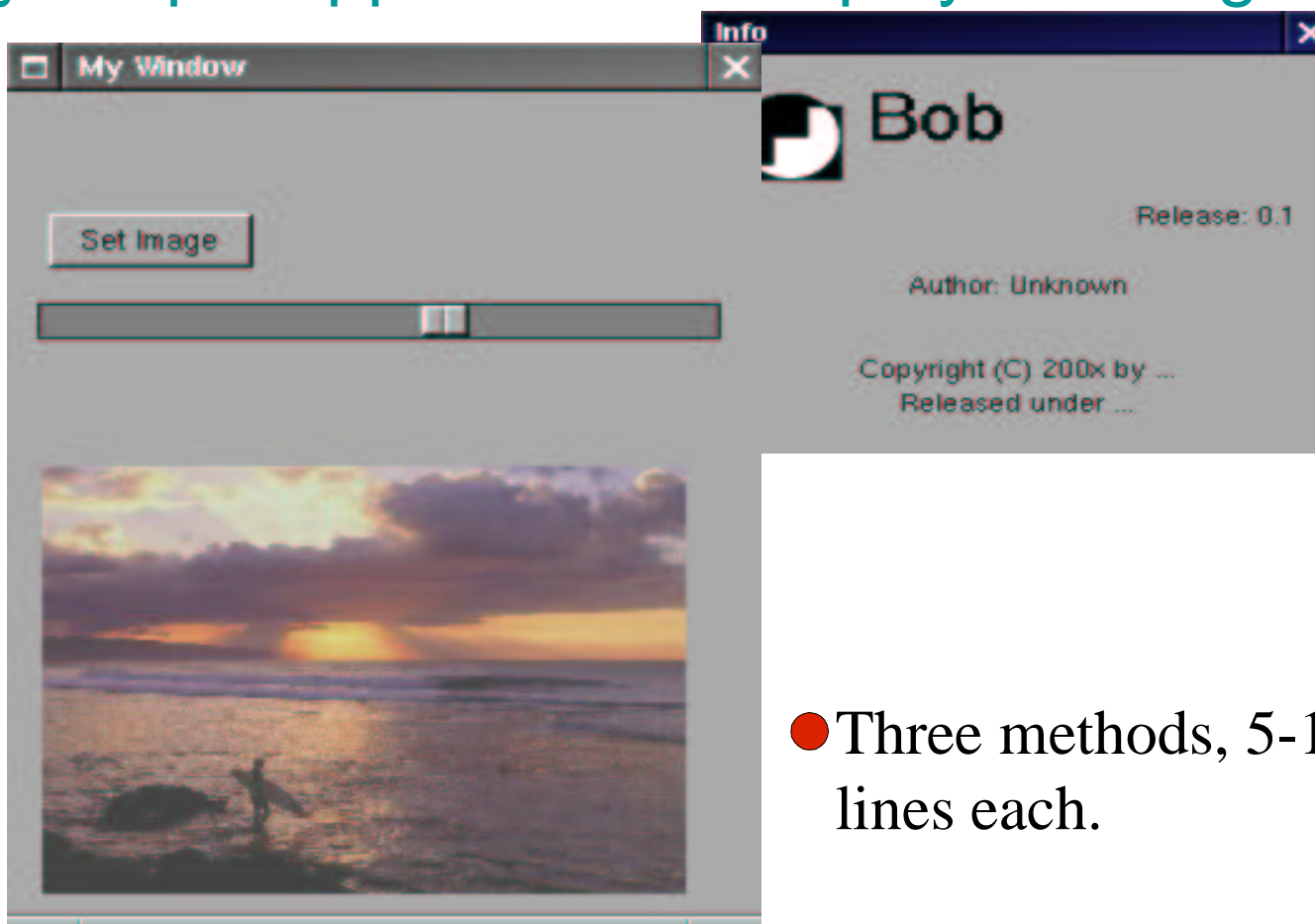




Project Example

- Very simple application to display an image

Bob	
Info	i
Hide	h
Services	s
Quit	q

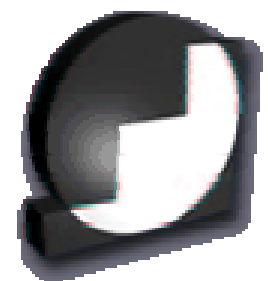


- Three methods, 5-10 lines each.



Real World Uses

- GNUstep is being used commercially now in intensive computing environments.
 - ◆ TCP/IP Servers (Multiplatform)
 - ◆ Database access
 - ◆ Network Management software



Applications

- Mail
- Workspace/File Browser
- Lots of others



Mail

IMAP on exchange1.doc.com - Inbox

131 messages (1968KB) - 75 unread (1093KB) - 1 selected (2KB)

	Id	Status	Date	From	Subject
	4024C		Feb 11	Ginger Shaw	Phonotronics Spectra magazine subscribers ONLY...
	4025E		Feb 11	Helge Hess	Re: Proposition for a Gorm feature. Was: Gorm too complex
	4025E		Feb 11	Nicola Pero	Re: =?iso-8859-1?Q?Re: Proposition for a G
	4026E		Feb 11	Michael Hanni	orm feature. Was: Gorm too complex ??=
	40267	A	Feb 11	Eden Chen	NSPopUpButton, the actual patch
	4026E		Feb 11	Raphael Bauduin	RE: Mask Layout for SIOB
	4027E		Feb 11	Philippe C.D. Robert	FOSDEM activity on saturday evening
	4028E		Feb 11	Yinbao Yang	Re: classes file in PCGormProj
	4030C	A	Feb 12	enrico donelli	process definition for metro waveleocker mask layout a
	40312		Feb 11	william85@usa.net	problem installing gnustep-ruby
	40313N		Feb 11	Gregory Casamento	Jump Start Your Business.....ADVERTISE on the INTERNET

On saturday evening, we have reserved a place where we all can come together with a little buffet (very simple). We invite all speakers, all developers from meetings (KDE, Gnome, GNustep, PHP, Mozilla), the FSF people, ...

We need an estimation of the number of people who will be present. Can you tell me if you want to be part of this? I really think it will be a good occasion to meet with each other in a relaxed atmosphere.

Raph

Free Software and Open Source Developers Meeting

See you at the 2002 edition. Check the 2001 sessions on www.opensource-tv.com

Visit <http://www.fosdem.org> and become member of the mailing list!



Preferences Panel

Preferences Panel

Personal

Viewing

Sending

Receiving

Compose

Quoting

MIME

Delivery methods

Method	Value
SMTP	smtp.doc.com

Add Edit Remove

Additional Outgoing Headers

Key	Value
-----	-------

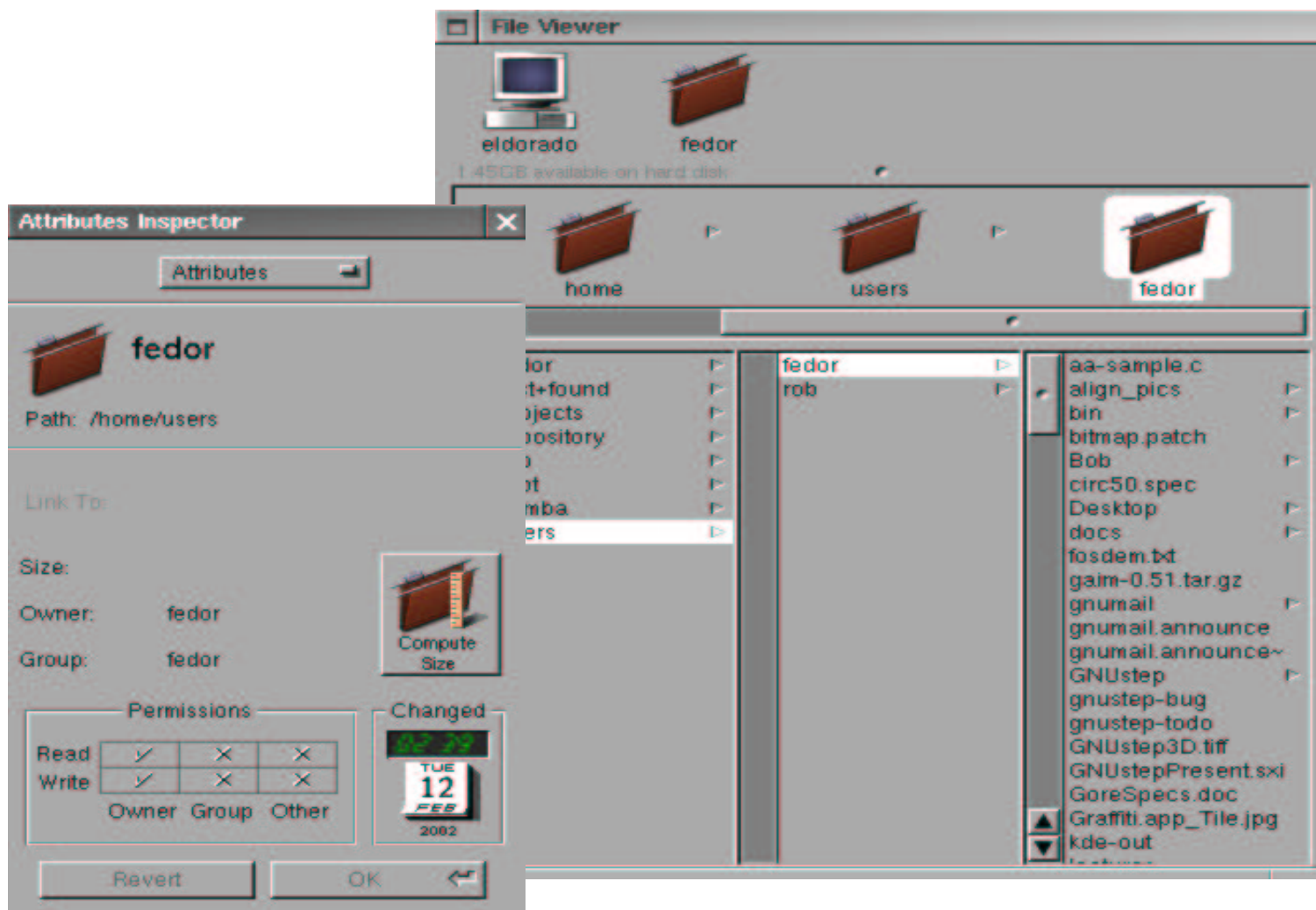
Key: Value:

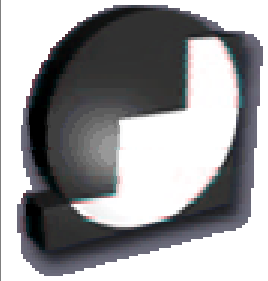
Add Remove

Apply Cancel OK ↩



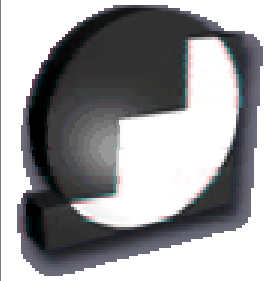
Gworkspace





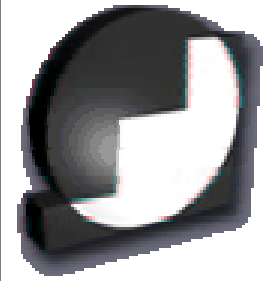
Is It Usable? Useful?

- Hackers
- Users
- 'Commercial' Developers
- MacOSX



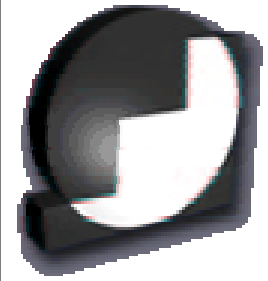
Is It Usable?

- Hackers? - Of course!
 - Stable makefile package and Foundation
 - Usable IDE Tools
 - Lots of things to fix 😊



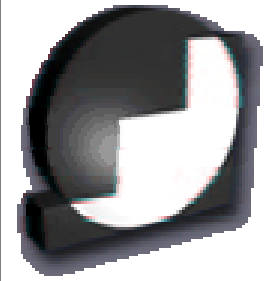
Is It Usable?

- Hackers? - Of course!
- Users? - Useful Apps
 - Mail
 - Gworkspace
 - Preferences
 - Simple games, utilities, etc.
 - (Not completely polished...)



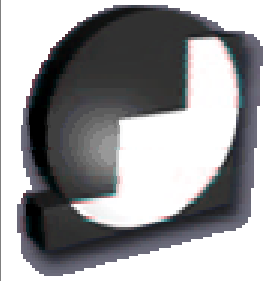
Is It Usable?

- Hackers? - Of course!
- Users? - Useful Apps
- 'Commercial' Developers— - Potential
 - Fully debugged API
 - Most of the basic work done.
 - Needs more testing, optimization, bug fixes



Is It Usable?

- Hackers? - Of course!
- Users? - Useful Apps
- 'Commercial' Developers? - Potential
- **MacOSX? - Not very close**
 - ◆ Need to perfect every aspect of the system.
 - ◆ Always behind Apple (adding features).
 - ◆ Can take advantage of other Free projects!



Next Steps

- More robust drawing (libart?, OpenGL?)
- Text and glyph handling
- Optimization,
- Documentation
- Windows backend
- Integration with more popular systems (GNOME, KDE).
- CORBA



Conclusion

- GNUstep has a complete framework, we just need to fill in the details.
- Libraries and Tools are usable now for Application development
- User applications exist and very well developed, considering most are written by one person in a short time span. **This shows the power of GNUstep!!**