étoile

GNUstep-based Desktop
Goals

- Modularity
- Project and Document oriented
- Facilitate workflow
- Communication / Sharing
- Focus on ergonomy and usability
The Plan

- First versions: provides frameworks/programming utilities.
- Release desktop/environment applications (v0.2/0.3 +)
- Provides user applications
Frameworks (étoilé 0.1)

- LuceneKit
- PreferencesKit
- UnitKit (GNUstep)
- SQLiteClient
- Developers Examples
Frameworks (étoilé 0.2)

- improvements on existing frameworks
- Azalea 0.1
- Camaelon
- ServiceBarKit
- IconKit
- SystemPrefs (using GNOME system tool backend)
User Interface

- NeXT menus by default, optional horizontal menus
- Nesedah UI theme + new icons
- Tabbed shelf / Dock
- Projects integration + nesting (ala Squeak)
- Workspace annotations + folder spaciality
- Proxy Icons & first-class objects
```c
/*@ All Rights Reserved */

#import "Document.h"
#import <AppKit/AppKit.h>

@implementation Document

-(BOOL) loadDataRepresentation: (NSUndoRedoType)undoType {
    // Insert code here to read-in your document data
    return NO;
}

-(NSData *) dataRepresentationOfThisFile {
    // Insert code here to return a data representation of your document
    return nil;
}
@end
```
Project Management

- Projects as first class objects
- "special folders"
- similar to X11 virtual desktops
- ...but persistent state
Workspace + Annotations

- Close to OpenStep Workspace manager
- but more representations
- spatial view + annotation
- Indexing and metadata
Proxy icons

- Printer
- Contacts
- Locations
- etc.
Creating Documents

- created from the workspace
- list of documents types
- hierarchical (text>rtf) and user-defined, plus recently used

A DTP document lets you mix images and text in the same page
No more save panel!

Save/Export documents by using this proxy icon panel (ala RiscOS)

Accessory views still work
Questions