To Advance through Presentation
Use Page Up and Page Down Keys

99 | Worldwide Developers Conference

Apple's Development Tools for Mac OS X

Dave Payne

Manager, Mac OS X Developer Applications

Commitment

- Mac OS X is Apple's future

- ProjectBuilder and InterfaceBuilder are Apple's development tools for the future

Overall Goals

- Create a great Carbon development environment

- Extend leadership in Cocoa and WebObjects development

- Strong support for C, C++, ObjectiveC, and Java

What We'll Cover

- ProjectBuilder
- Compiler
- Debugger
- InterfaceBuilder

ProjectBuilder Directions

John Graziano

Lead Engineer
ProjectBuilder Core Internals

ProjectBuilder Directions

- Layout of projects
- Project navigation
- User interface polish
- Build system
- Graphical debugger

Layout of Projects

- Easy import of Macintosh source code

- Easy upgrade of current PB projects

- Move from "project types" to project templates

- User-definable grouping of files:
    - Independent of physical layout on disk
    - Independent of build system

- Multiple build targets per project

Project Navigation

- Navigation shortcuts in user interface

- Easier access to included files

- Language-sensitive searching in C, C++, ObjectiveC, and Java code

- Access to dependent framework and library projects

User Interface Polish

- More efficient use of screen real estate

- Less "context-switching"

- Simplified menu layout

- GUI for all option settings

- Strict adherence to Macintosh UI guidelines

Build System

- Faster incremental builds
- Fast, accurate dependency analysis
- Support new application package layout
- Spaces in path names

Demo

The egcs Compiler

- Generate PEF executables and symbol files for Mac OS 8

- Support for AltiVec

- Bring C++ libraries up to full standard

- Support Mac OS 8 programming conventions:
  - #pragma once
  - #pragma unused
  - Compiler defaults as expected for Mac OS
  - CR or LF line termination

ProjectBuilder's Graphical Debugger

- Data inspection
- Object inspection
- Persistent breakpoints
- "Just-in-time" debugging

The GDB Debugger

- Leverage open source:
  - Current with latest FSF sources
  - Signed agreement with FSF to send our changes back

- Support improved debugging GUI in ProjectBuilder

- Faster startup

- Independent control of thread execution

InterfaceBuilder

Henri Lamiraux

Manager,
User Interface Tools Group

Overview

- General improvements
- Beyond Cocoa
- New NIB format
- Demo

Improvements

- Better integration with ProjectBuilder
  - Outlets/Actions synchronization

- Streamline user interface

- Enhance IB Framework API
  - Palettes
  - Editors

Improvements

- Complete Undo support
- Generalize property inspector
- Continue improving Cocoa support
  - TabView support
  - OutlineView support

Leverage Key Features

- WYSIWYG layout
- Ease of use (Drag and Drop)
- Visual connections (Outlets/Actions)
- Custom Inspectors
- Extensibility (palettes)

Currently Supported

- Cocoa
- EOF
- WebObjects
- ActiveX (Windows only)
- JavaBeans

Coming Soon

- Carbon
- Swing

Carbon

- InterfaceBuilder is not a resource editor
- Will not replace ResEdit

Carbon

- Layout user interface elements:
    - Menus and MenuBar
    - Windows and Dialogs
    - Controls and Control Layouts
- Full support for Appearance Manager

Carbon

- Will not initially support:
    - Outlets and Actions
    - Autosizing
- Will evolve along with the Toolbox

Swing

- Build full fledged Swing Applications
    - All Swing widgets

    - Outlets

    - Actions (using Event Listeners)

    - Layout Managers support
        - At least Spring and Strut layout
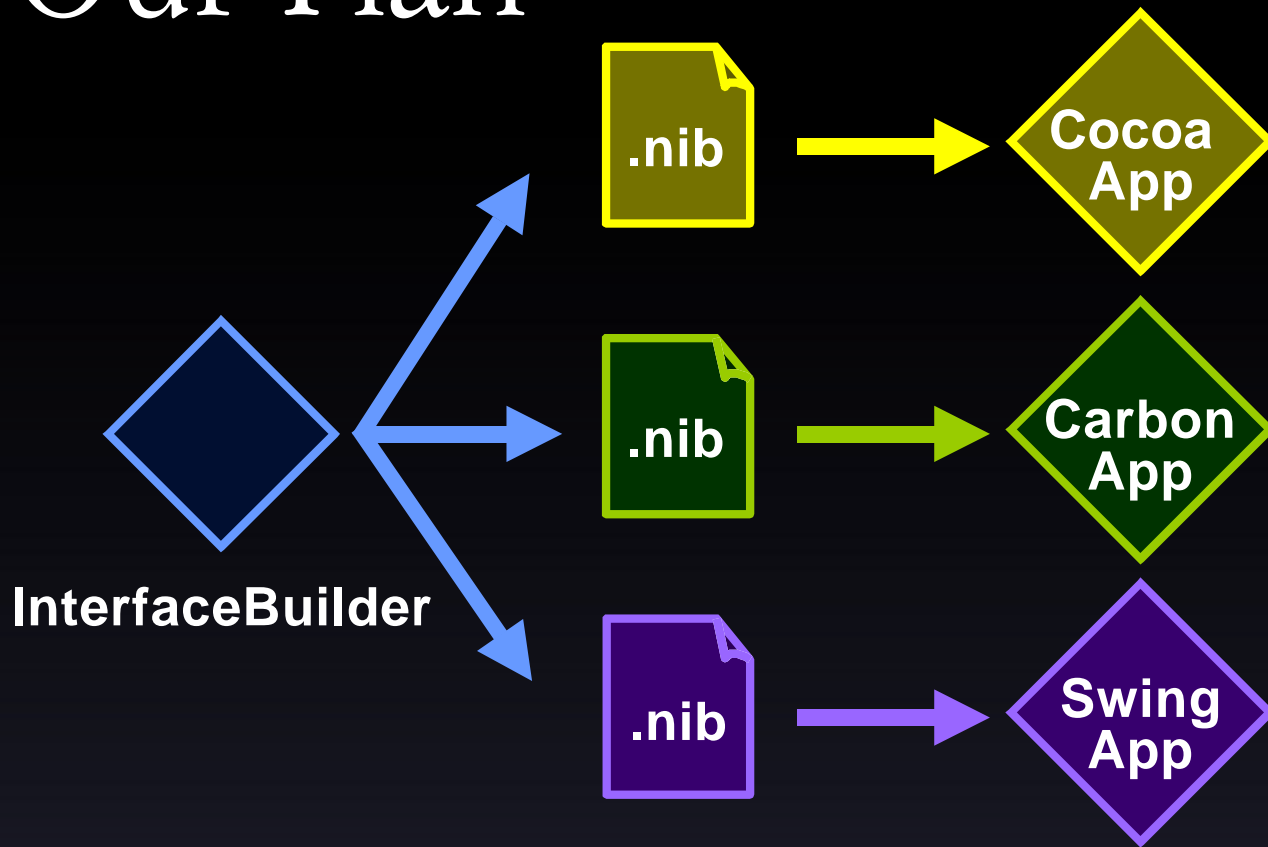
Today's UI File Formats

- Cocoa
    - NIB (InterfaceBuilder)

- Carbon
    - Resources
    - Limited to Dialogs and MenuBar

- Swing
    - No real standard
    - Source Code Generation

Our Plan

Current NIB Format

- Cocoa centric
- Binary format
- Persistent state of Cocoa objects

New XML Based NIB

- Human readable format
- Industry standard
- XML parsers are fast

Key-Value XML Format

- Describes Objects with named properties

- Only non-default property values are written

- Backward and forward compatibility
  - Unknown keys are ignored
  - Default values used for missing keys

IB Runtime for Cocoa

- No changes

IB Runtime for Carbon

- New Application packaging provides the containment facility

- Use new set of APIs to create windows, menubar…

  Continue…

IB Runtime for Carbon

(preliminary APIs)

```
OSErr IBCreateNibReference(CFStringRef inNibName,
                           IBNibRef *outNibRef);


OSErr IBGetNewWindow(IBNibRef inNibRef,
                     CFStringRef inName,
                     void *inWindowStorage,
                     WindowRef inBehind,
                     WindowPtr* outWindow);


OSErr IBGetNewMenuBar(IBNibRef inNibRef,
                      CFStringRef inName,
                      Handle* menuBar);


....
```

IB Runtime for Swing

- Add NIB to the application Jar

- Use provided package to instantiate UI elements from the NIB file

  (preliminary APIs)

```
public class IBNib {
        public IBNib(String nibName);
        public JFrame instantiateFrame(String name, ….);
        ...
}
```

Summary

- Apple is committed to creating great development tools

- Please send us your feedback— **macosx-tools-feedback@group.apple.com**

Roadmap

Carbon Overview	Hall C **Fri., 9:00am**	
Carbon on Mac OS 8 (Porting)	Hall C **Fri., 1:00pm**	
Carbon on Mac OS X	Hall C **Fri., 2:30pm**	
Core Foundation: Plug-ins	Hall A1 **Fri., 4:00pm**	
Intro to the Cocoa (Yellow) Framework	Hall B **Fri., 9:00am**	
What's New in Cocoa (Yellow) Framework	Hall B **Fri., 10:15am**	

**To Advance through Presentation
Use Page Up and Page Down Keys**