# Worldwide

# Developers

# Conference

# Developing with Interface Builder and Project Builder

## Jordan Dea-Mattson

### Senior Evangelist
### Rhapsody Evangelism

# Project Builder Overview

*Mike Monegan*

**Project Builder
Technical Lead
Rhapsody Engineering**

# Project Builder Overview

- Browse, edit and search your source code
- Manage the content and structure of your "project document"
- Build your project and fix errors/warnings
- Debug your project via a graphical debugger integration
- Launch other apps used to edit nib documents, images, sounds, etc.
- Integrates with your source control system

# The Project Model

- A project is:
  - A set of sources, resources, and supporting files
  - A set of attributes for the project
  - A set of subprojects—each a project in its own right
  - A project type provides structure, attribute definitions, and a built-in build capability
- Each project has a type that provides a template and some built-in behavior

# Project Types

- **Application—a standard Yellow Box application**
  - Template contains default nib file, frameworks, and main()
  - Code in any combination of Objective-C, Objective-C++, or Java
- **Component—a generic subproject useful for logically segregating the parts of a large project**

# Project Types (cont.)

- Framework—a bundle containing a dynamic shared library plus resources such as header files, nib files, images, and documentation

- JavaPackage—a collection of Java classes

- Tool—a project for a server or a command-line tool

- Loadable Bundle—generates executable code and related resources that can be dynamically loaded at run time

# Project Types (cont.)

- Palette—a bundle suitable for starting an Interface Builder palette
- Library—generates a static library or a dynamic shared library (no resources)
- Aggregate—a collection of loosely related subprojects
  - Entire build is under the control of a common project
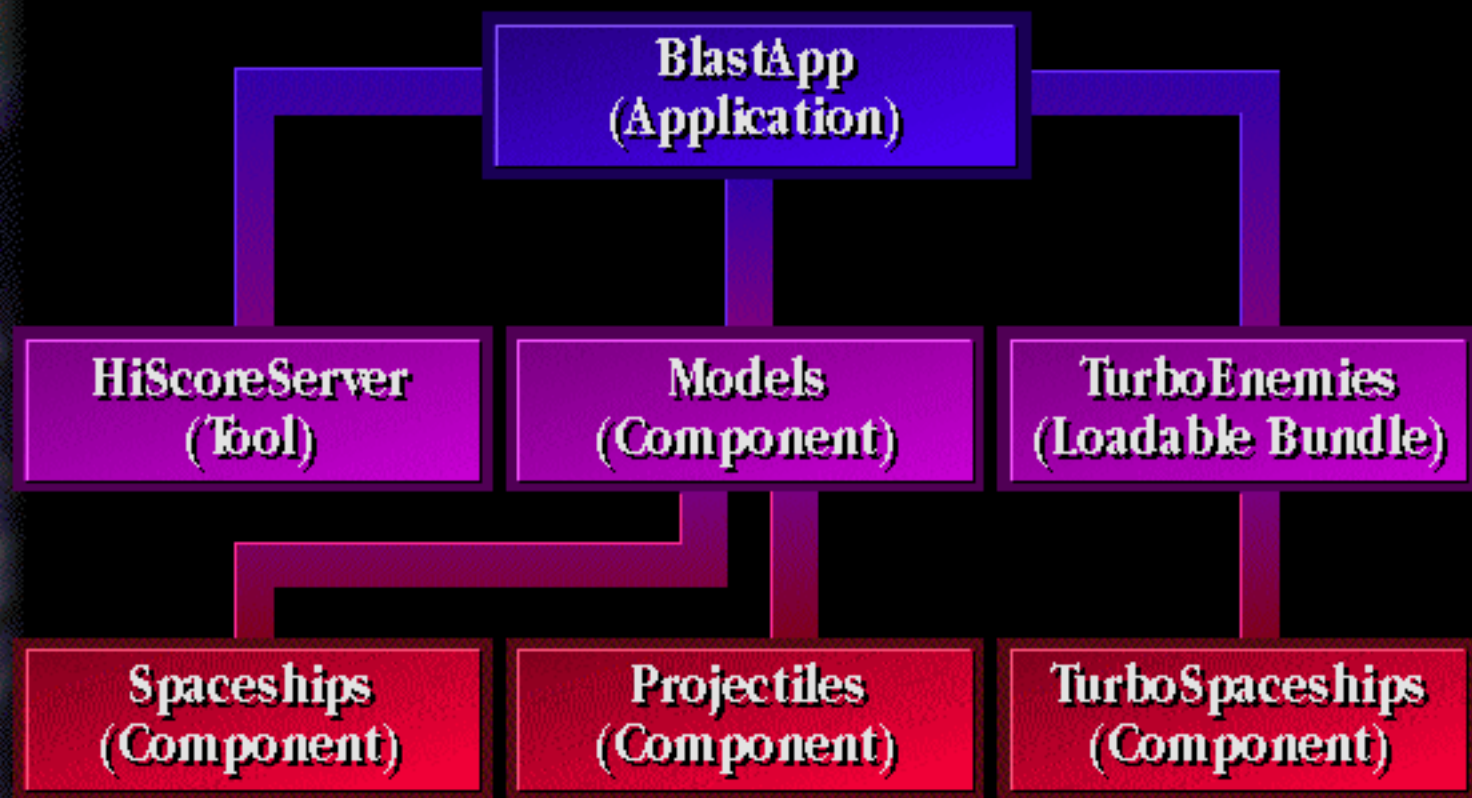- Legacy—a project for which PB does not maintain the Makefile

# Project Types (cont.)

- Also available are:
  - EOF
    - EOApplication—complete with fancy project creation wizard
  - WOF
    - WebObjectsApplications—PB does apps for the web too!
- Developers may create custom project types
  - PB is open and extensible
  - Full range of project types

# Example Project Hierarchy

# Demo

## Creating the project

- Create a new project
- Import legacy code
- Add Objective-C class
- Editing modes
  - "project window"
  - "window-per-file"

# Demo

*Navigating the project and documentation*

- Browse into a framework
- Navigate the symbols in a header file
- Click-for-help when you want to know more
- Find results provide access to documentation

# Demo

## *Building the project*

- **Will support builds for Rhapsody for PowerPC, Rhapsody for Intel, Yellow Box for Windows, Yellow Box for Mac OS**

- **Compilers**
  - Default C/Objective-C/C++ compiler based on GNU's gcc 2.7.2.1
  - Default Java compiler is a port of Sun's JDK 1.1 javac
  - Can plug other compilers and linkers on per-project basis

# Demo

**Debugging the project**

- Launch panel
  - Runs and debugs executables
  - Controls execution environment
  - Displays data, pointers, and objects

# Beyond Project Builder

- **FileMerge**
  - Compare and merge files and folders
- **ObjectAlloc**
  - Track and analyze Object-C object allocations
- **MallocDebug**
  - Track and analyze all heap allocations

# Beyond Project Builder (cont.)

- **Sampler**
  - Sample and analyze activity in an app at regular intervals
- **Interface Builder**
  - Directly edit and connect user interface objects

# Interface Builder Overview

## Derek Clegg

**Interface Builder
Technical Lead
Rhapsody Engineering**

# Interface Builder Overview

- Your interface development tool for Rhapsody

- Every Rhapsody application—including Interface Builder—is developed with Interface Builder

- We eat our own dog food!

Interface Builder
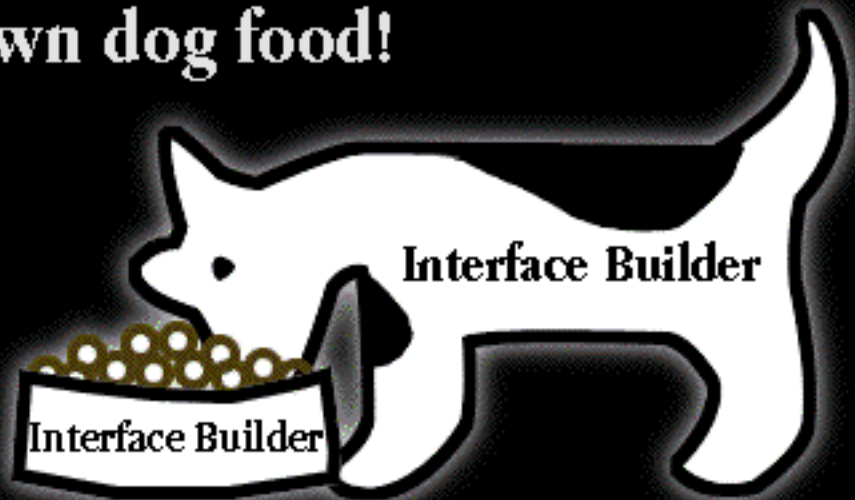
Interface Builder

# Interface Builder Isn't…

- A picture painter
- A property list editor
- A code generator
- A widget creator

# Interface Builder

*No waterfalls!*

- You don't create a UI and then fall off a cliff
- Documents can be edited and modified at each stage of the development process
- All objects in IB are "live"
- Objects in IB are bug-for-bug compatible with objects in your application
- This is a Good Thing™

# Demo

- In-place editing
- In-place attribute modifications
  - Font
  - Color
  - Images

# Demo

*Testing your interface*

- Quick prototyping
- Last-minute UI changes
- Easy to test new widget behavior

# Demo

## *Interface Builder is fully extensible*

- Developers create palettes of their own objects
- Each palette provides inspectors for its objects
- Palette can even provide new inspectors for objects
- Palettes are self-contained

# Look Out, Here Comes Java!

*If you're not part of the steamroller, you're part of the road!*

- Interface Builder will support 100% Java development

- Palettes will contain JavaBeans™ and widgets

- Full Objective-C and Java interoperability

# Q&A