Worldwide
Developers
Conference

Uncommon Object Model: The Rhapsody Runtime

Object Runtimes Group

Blaine Garst

Toby Paterson

Uncommon Object Model

Reasonable features

- Dynamic dispatch
- Single inheritance of implementation
- Multiple inheritance of abstraction
- Introspection
- Exceptions
- Garbage collection

Uncommon Object Model

Mature frameworks

- Foundation classes
  - Collection classes
  - Unicode strings
  - OS abstractions
- Distribution Model
- Persistence Model

Where Do I Get One?

- Java

Where Do I Get One?

- Java
- Objective-C !!

Java in Developer Release

Mid-1997

- Sun JDK 1.1
- JIT
- Debugging

Java in Premier Release

Early 1998

- Better GC
- Better Performance
- Current JDK (1.2)

Java in Unified Release

Mid-1998

- Java 2.0, HotSpot
- Scalability
  - Reduced VM memory usage
  - Precompile commonly used classes
  - Scatter-load code and data to reduce memory footprint
  - Share classes between processes

Objective-C

Ten years of refinement

- Highly tuned, scalable runtime
- Mature (dynamic shared) libraries
- Development tools
- Performance tools
- Java style syntax!

Objective-C: Runtime

- Meta-data kept in executable
- Hash table based dispatching
- Freeze-dried selector caches
- Dynamic code loading (NSBundle)
- Dynamic class creation (!!)

Objective-C

- Ten years of refinement
- Highly tuned, scalable runtime
- Mature (dynamic shared) libraries
- Development tools
- Performance tools
- Java style syntax!

Objective-C: Modern syntax!

```
NSString description() {
    NSString here = @"Here: ";
    NSString tmp = list.description();
    return
    here.stringByAppendingString(tmp);
}
```

Objective-C

What's different

- Forwarding
- Categories (anybody can extend any class)
- Class objects
- C structs, unions, varargs, pointers…
- Swizzling

The Bridge

Packages map to Objective-C frameworks

- Java classes map to Objective-C classes
  - Subclass in Java
- Bridge handles messaging (this/self, super)
  - scalars -> scalars
  - exceptions -> exceptions
  - objects -> objects
  - protocols -> interfaces

What Should I Remember About This Talk?

- Closely working with Sun
- JDK, Beans, RMI, AWT, JFC, AFC…
- “Yellow Box” in Java on Mac OS, Rhapsody, Windows

Demo

Q&A

Worldwide
Developers
Conference