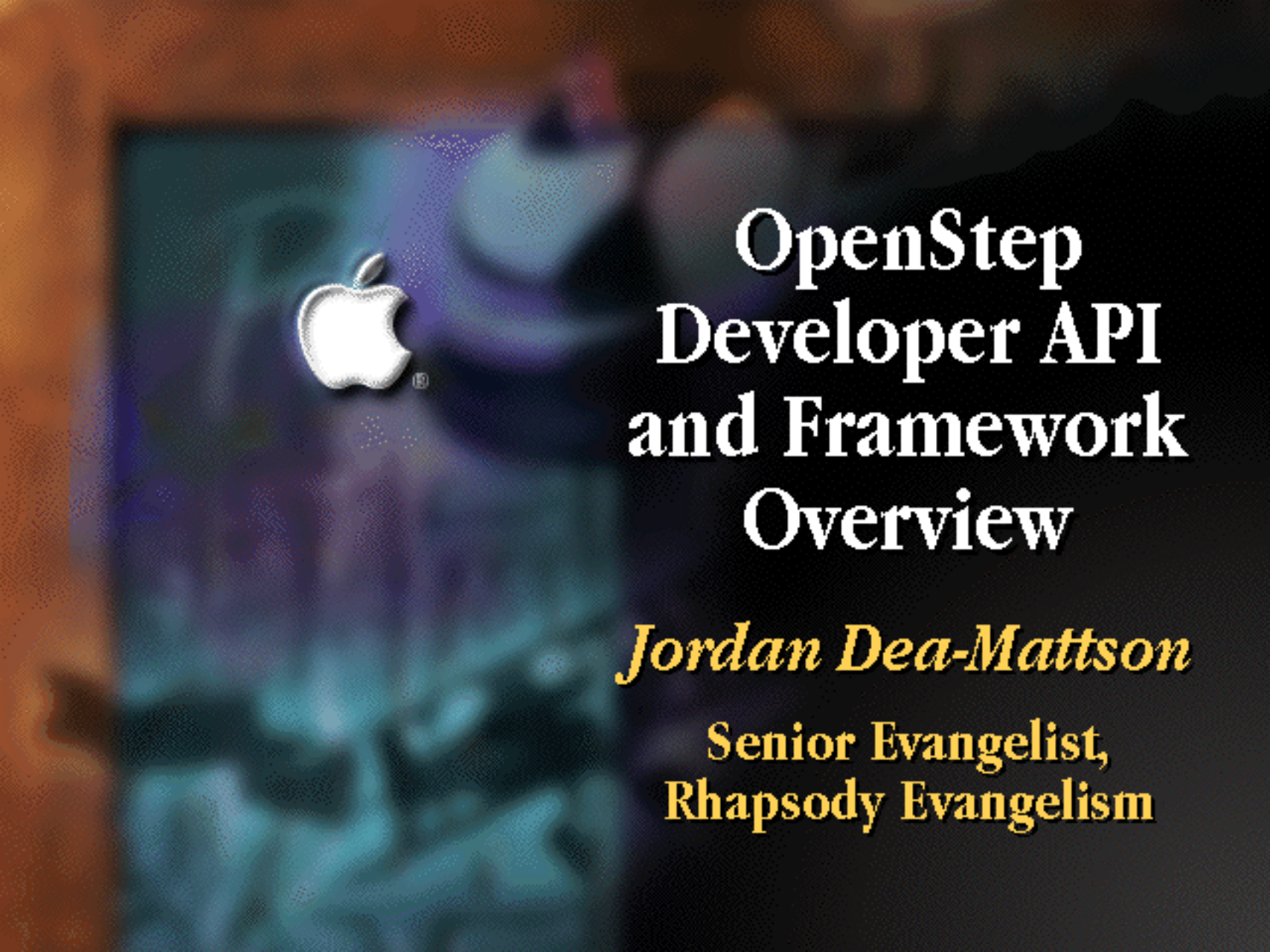

Worldwide
Developers
Conference


OpenStep
Developer API
and Framework
Overview
Jordan Dea-Mattson
Senior Evangelist,
Rhapsody Evangelism


Yellow Box
Technical
Overview
Peter Graffagnino
Director,
Yellow Box Engineering

What We'll Cover…

What We'll Cover…

- What is this Yellow Box thing?

What We'll Cover…

- What is this Yellow Box thing?
- Guiding Principles

What We'll Cover…

- What is this Yellow Box thing?
- Guiding Principles
- Functional Overview

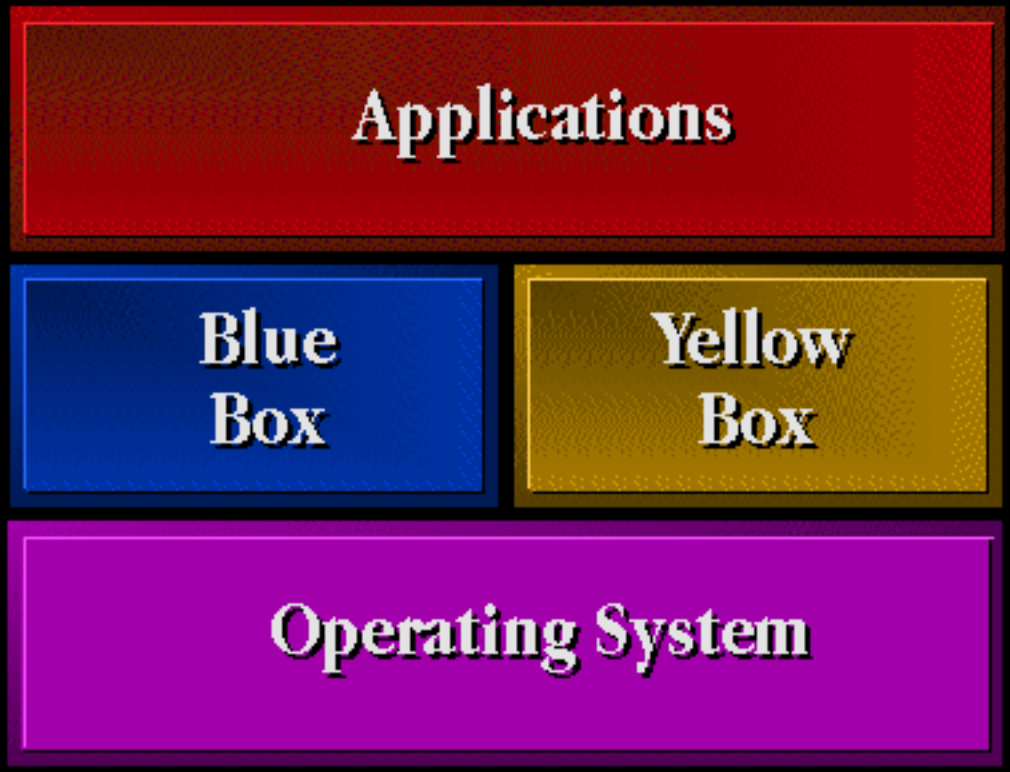
What We'll Cover…

- What is this Yellow Box thing?
- Guiding Principles
- Functional Overview
- Comparisons

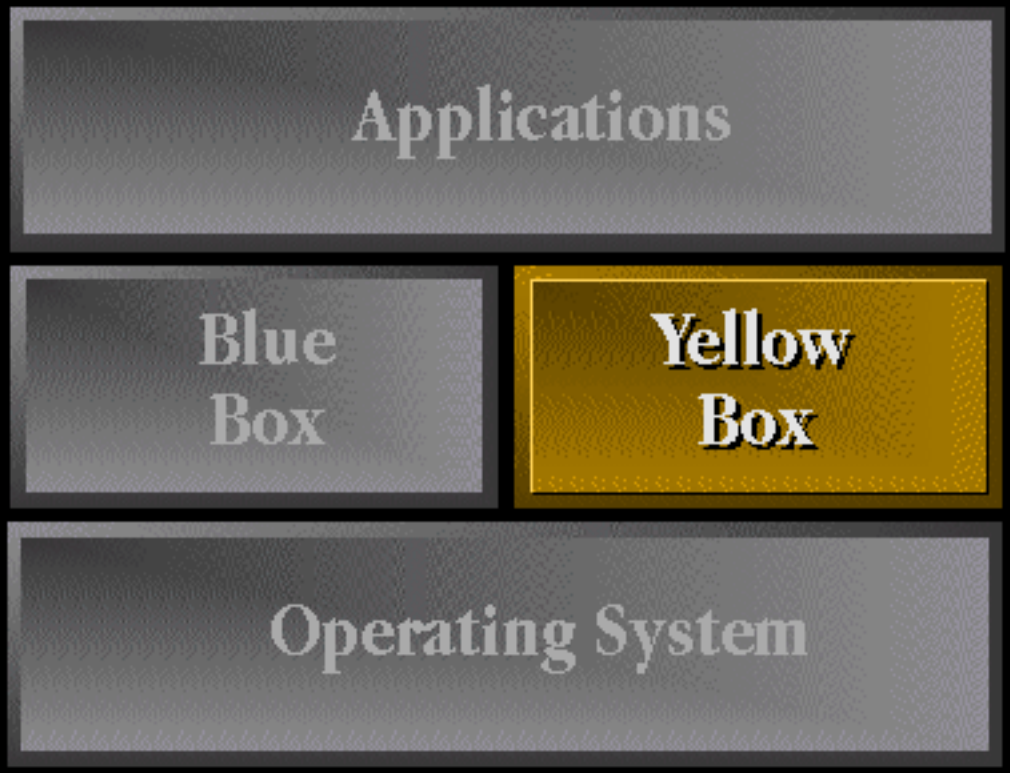
What We'll Cover…

- What is this Yellow Box thing?
- Guiding Principles
- Functional Overview
- Comparisons
- Opportunities

Rhapsody Architecture



You Are Here…



Some History and Terminology

Some History and Terminology

- "NEXTSTEP"—NeXT's Mach based OS and frameworks, last release NEXTSTEP 3.3

Some History and Terminology

- "NEXTSTEP"—NeXT's Mach based OS and frameworks, last release NEXTSTEP 3.3
- "OpenStep"—second generation of NEXTSTEP application framework

Some History and Terminology

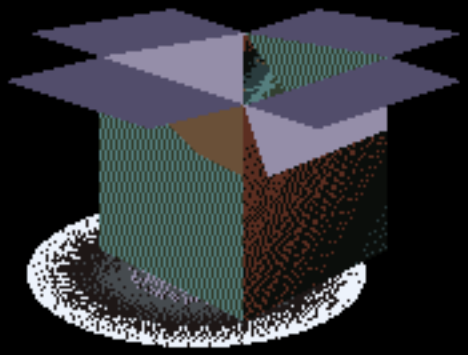
- "NEXTSTEP"—NeXT's Mach based OS and frameworks, last release NEXTSTEP 3.3
- "OpenStep"—second generation of NEXTSTEP application framework
- "OpenStep for Mach 4.X"—successor releases to NEXTSTEP 3.3

Some History and Terminology

- "NEXTSTEP"—NeXT's Mach based OS and frameworks, last release NEXTSTEP 3.3
- "OpenStep"—second generation of NEXTSTEP application framework
- "OpenStep for Mach 4.X"—successor releases to NEXTSTEP 3.3
- "OpenStep for Windows"—OpenStep frameworks on Windows NT/95

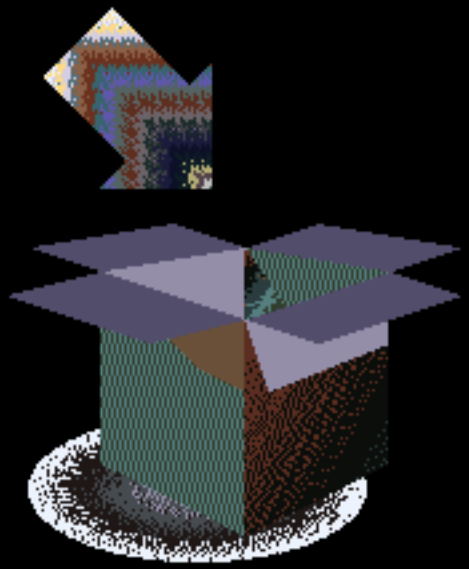
…And Now

…And Now



. . .And Now

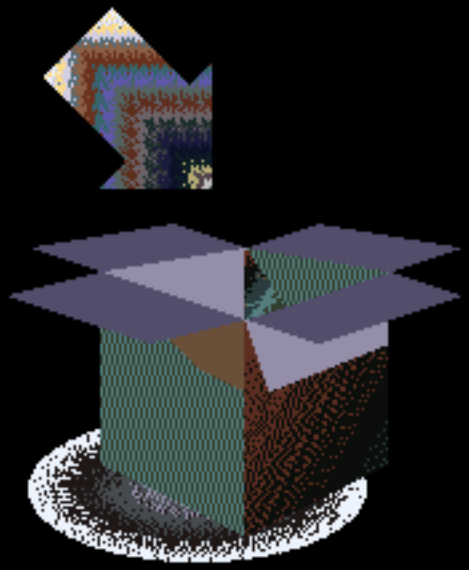
- NeXT Technology



…And Now

NeXT Technology

OpenStep APIs

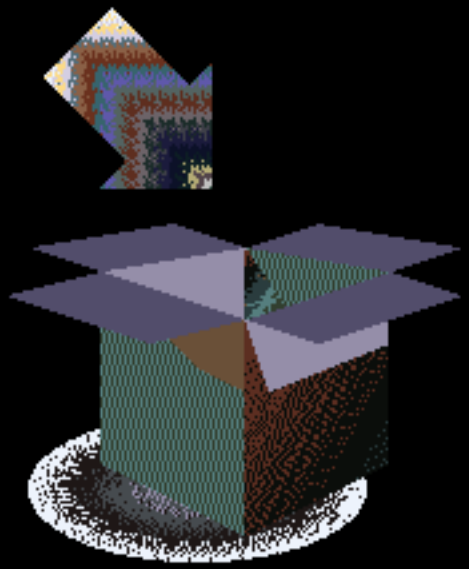


…And Now

NeXT Technology

OpenStep APIs

Distributed Objects



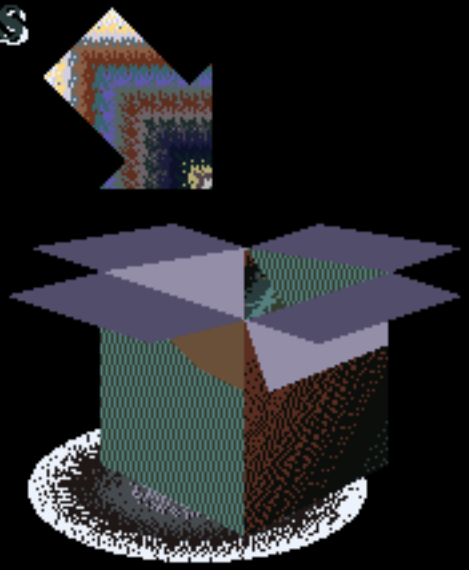
…And Now

NeXT Technology

OpenStep APIs

Distributed Objects

Enterprise Objects



…And Now

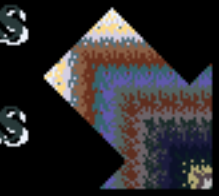
NeXT Technology

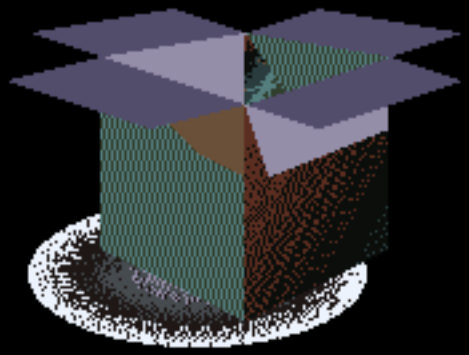
OpenStep APIs

Distributed Objects

Enterprise Objects

Web Objects





…And Now

NeXT Technology

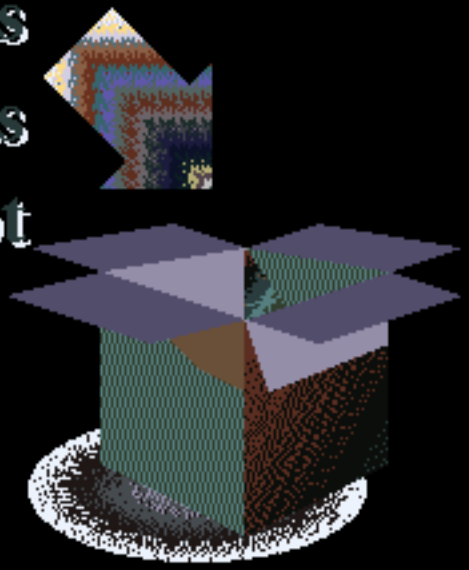
OpenStep APIs

Distributed Objects

Enterprise Objects

Web Objects

Display PostScript



…And Now

NeXT Technology

OpenStep APIs

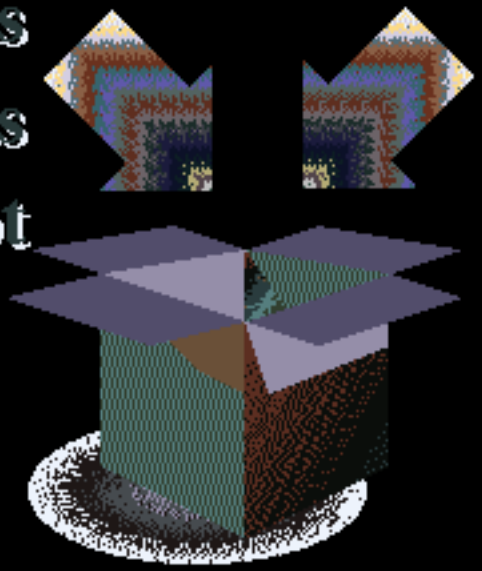
Distributed Objects

Enterprise Objects

Web Objects

Display PostScript

Apple Technology



…And Now

NeXT Technology

OpenStep APIs

Distributed Objects

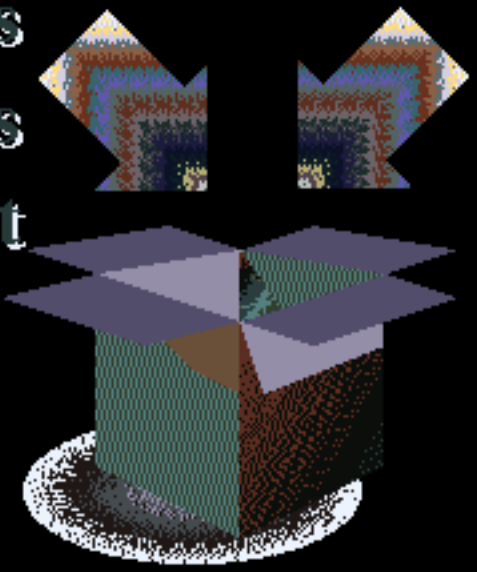
Enterprise Objects

Web Objects

Display PostScript

Apple Technology

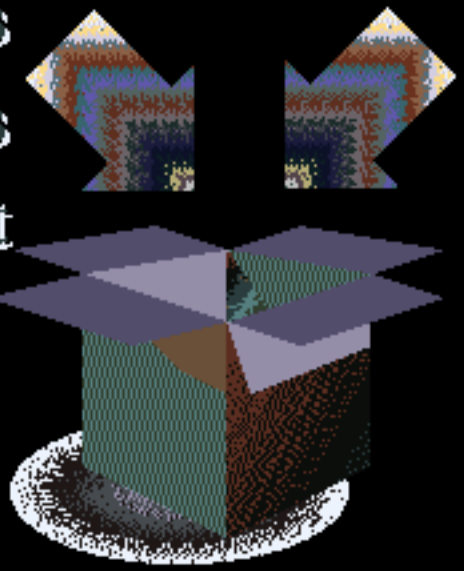
QTML



…And Now

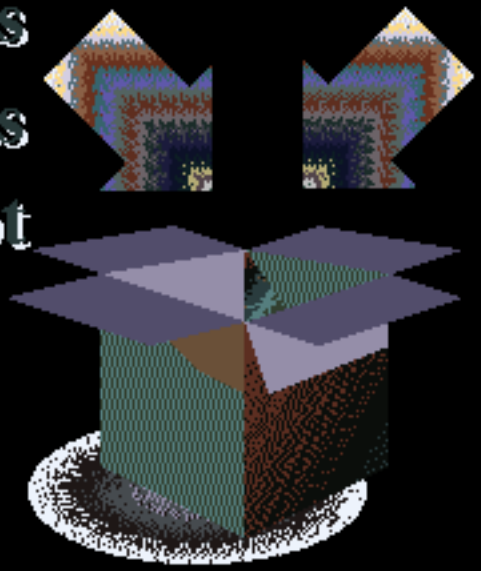
NeXT Technology
OpenStep APIs
Distributed Objects
Enterprise Objects
Web Objects
Display PostScript

Apple Technology
QTML
ColorSync



…And Now

NeXT Technology
OpenStep APIs
Distributed Objects
Enterprise Objects
Web Objects
Display PostScript



Apple Technology
QTML
ColorSync
GX Typography

…And Now

NeXT Technology

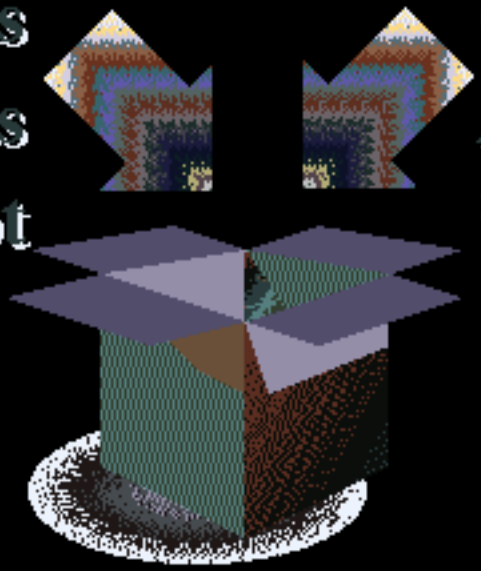
OpenStep APIs

Distributed Objects

Enterprise Objects

Web Objects

Display PostScript



Apple Technology

QTML

ColorSync

GX Typography

Advanced Look and Feel

Yellow Box

Cross-Platform Application Substrate

Yellow Box

Cross-Platform Application Substrate

- 100% “Buzzword” compliant

Yellow Box

Cross-Platform Application Substrate

- 100% "Buzzword" compliant
  - Cross platform

Yellow Box

Cross-Platform Application Substrate

- 100% "Buzzword" compliant
  - Cross platform
  - Object oriented

Yellow Box

Cross-Platform Application Substrate

- 100% "Buzzword" compliant
  - Cross platform
  - Object oriented
  - Scalable

Yellow Box

Cross-Platform Application Substrate

- 100% "Buzzword" compliant
  - Cross platform
  - Object oriented
  - Scalable
  - Enterprise ready

Yellow Box

Cross-Platform Application Substrate

- 100% "Buzzword" compliant
  - Cross platform
  - Object oriented
  - Scalable
  - Enterprise ready
  - Media-rich

Yellow Box

Cross-Platform Application Substrate

- 100% "Buzzword" compliant
  - Cross platform
  - Object oriented
  - Scalable
  - Enterprise ready
  - Media-rich
  - Client/Server

Yellow Box

Cross-Platform Application Substrate

- 100% "Buzzword" compliant
  - Cross platform
  - Object oriented
  - Scalable
  - Enterprise ready
  - Media-rich
  - Client/Server
  - World ready

Yellow Box

Cross-Platform Application Substrate

- 100% "Buzzword" compliant
  - Cross platform
  - Object oriented
  - Scalable
  - Enterprise ready
  - Media-rich
  - Client/Server
  - World ready
  - Java-enabled

Yellow Box Platforms

Yellow Box as a "Meta Platform" is highly portable

Yellow Box Platforms

Yellow Box as a "Meta Platform" is highly portable

- Rhapsody (Mach + BSD Unix)

Yellow Box Platforms

Yellow Box as a "Meta Platform" is highly portable

- Rhapsody (Mach + BSD Unix)
- Windows NT

Yellow Box Platforms

Yellow Box as a "Meta Platform" is highly portable

- Rhapsody (Mach + BSD Unix)
- Windows NT
- Windows 95…

Yellow Box Platforms

Yellow Box as a "Meta Platform" is highly portable

- Rhapsody (Mach + BSD Unix)
- Windows NT
- Windows 95…
- Mac OS

Yellow Box Platforms

Yellow Box as a "Meta Platform" is highly portable

- Rhapsody (Mach + BSD Unix)
- Windows NT
- Windows 95…
- Mac OS
- ????

Guiding Principles

Make developers more productive

Guiding Principles

Make developers more productive

- Visual development tools

Guiding Principles

Make developers more productive

- Visual development tools
- Powerful and consistent APIs

Guiding Principles

Make developers more productive

- Visual development tools
- Powerful and consistent APIs
- Factor out common code from Apps

Guiding Principles

Make developers more productive

- Visual development tools
- Powerful and consistent APIs
- Factor out common code from Apps
- Integrate industry standard technologies

Guiding Principles

Make developers more productive

- Visual development tools
- Powerful and consistent APIs
- Factor out common code from Apps
- Integrate industry standard technologies
- Enable “ComponentWare”

Guiding Principles

Make developers more productive

- Visual development tools
- Powerful and consistent APIs
- Factor out common code from Apps
- Integrate industry standard technologies
- Enable “ComponentWare”
- Make Apps work well together

Visual Development Tools

Visual Development Tools

- Interface Builder

Visual Development Tools

- Interface Builder
- Project Builder

Visual Development Tools

- Interface Builder
- Project Builder
- Enterprise Objects Modeler

Visual Development Tools

- Interface Builder
- Project Builder
- Enterprise Objects Modeler
- 3rd party tools (i.e., Metrowerks, Symantec, Roaster Technologies, etc.)

Visual Development Tools

- More Info:
  "Developing with Interface Builder and Project Builder"
  Friday, 10:50, Hall 1

Powerful and Consistent APIs

Keep paradigms simple and synergistic

Powerful and Consistent APIs

Keep paradigms simple and synergistic

- Keep simple things simple

Powerful and Consistent APIs

Keep paradigms simple and synergistic

- Keep simple things simple
- Make complex things possible

Powerful and Consistent APIs

Keep paradigms simple and synergistic

- Keep simple things simple
- Make complex things possible
  - No “walls” or “cliffs”

Powerful and Consistent APIs

Keep paradigms simple and synergistic

- Keep simple things simple
- Make complex things possible
  - No “walls” or “cliffs”
- Consistent, meaning ful naming:

Powerful and Consistent APIs

Keep paradigms simple and synergistic

- Keep simple things simple
- Make complex things possible
  - No “walls” or “cliffs”
- Consistent, meaning ful naming:
  - WIN32: ERROR_SUCCESS, IUnknown

Powerful and Consistent APIs

Keep paradigms simple and synergistic

- Keep simple things simple
- Make complex things possible
  - No “walls” or “cliffs”
- Consistent, meaning ful naming:
  - WIN32: ERROR_SUCCESS, IUnknown
  - Java: 30 methods named add( )

Powerful and Consistent APIs

Keep paradigms simple and synergistic

- Keep simple things simple
- Make complex things possible
  - No “walls” or “cliffs”
- Consistent, meaning ful naming:
  - WIN32: ERROR_SUCCESS, IUnknown
  - Java: 30 methods named add( )
- “Reusability” of paradigms

Powerful and Consistent APIs

Keep paradigms simple and synergistic

- Keep simple things simple
- Make complex things possible
  - No “walls” or “cliffs”
- Consistent, meaning ful naming:
  - WIN32: ERROR_SUCCESS, IUnknown
  - Java: 30 methods named add( )
- “Reusability” of paradigms
  - Autorelease strategy

Factor Common Code from Apps

Applications should leverage platform for common services

Factor Common Code from Apps

Applications should leverage platform for common services

- Copy/Paste of sample code is not reuse

Factor Common Code from Apps

Applications should leverage platform for common services

- Copy/Paste of sample code is not reuse
- Faster development cycles

Factor Common Code from Apps

Applications should leverage platform for common services

- Copy/Paste of sample code is not reuse
- Faster development cycles
- More consistent user experience across applications

Factor Common Code from Apps

Applications should leverage platform for common services

- Copy/Paste of sample code is not reuse
- Faster development cycles
- More consistent user experience across applications
- Reduce maintenance burden on developers

Factor Common Code from Apps

Applications should leverage platform for common services

- Copy/Paste of sample code is not reuse
- Faster development cycles
- More consistent user experience across applications
- Reduce maintenance burden on developers
- Platform can evolve to add value to applications

Factor Common Code from Apps

Applications should leverage platform for common services

- Copy/Paste of sample code is not reuse
- Faster development cycles
- More consistent user experience across applications
- Reduce maintenance burden on developers
- Platform can evolve to add value to applications
  - Systemwide fax support

Factor Common Code from Apps

- More Info:
  "Designing Objects for Reuse and Extensibility"
  Wednesday, 5:50, A-1

Technology Integration

Technology Integration

Technology Integration

Technology Integration

ColorSync

RTF, HTML, TIFF, EPS, JPG, GIF

PostScript

Technology Integration

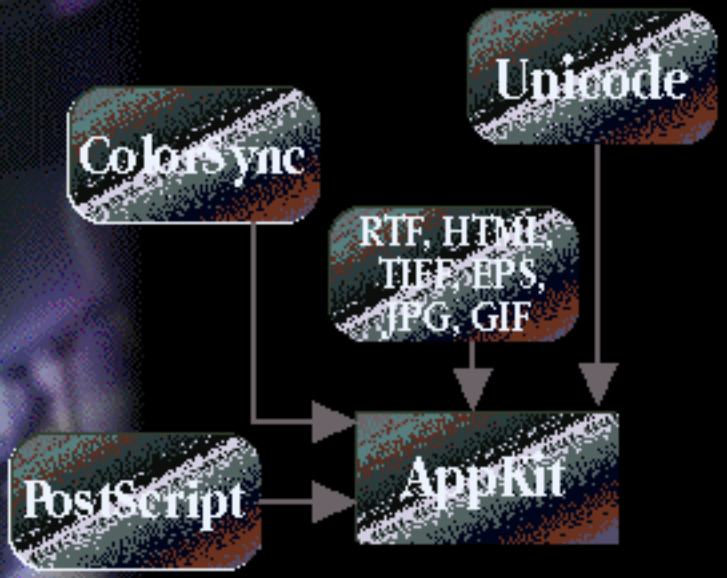
Unicode

ColorSync

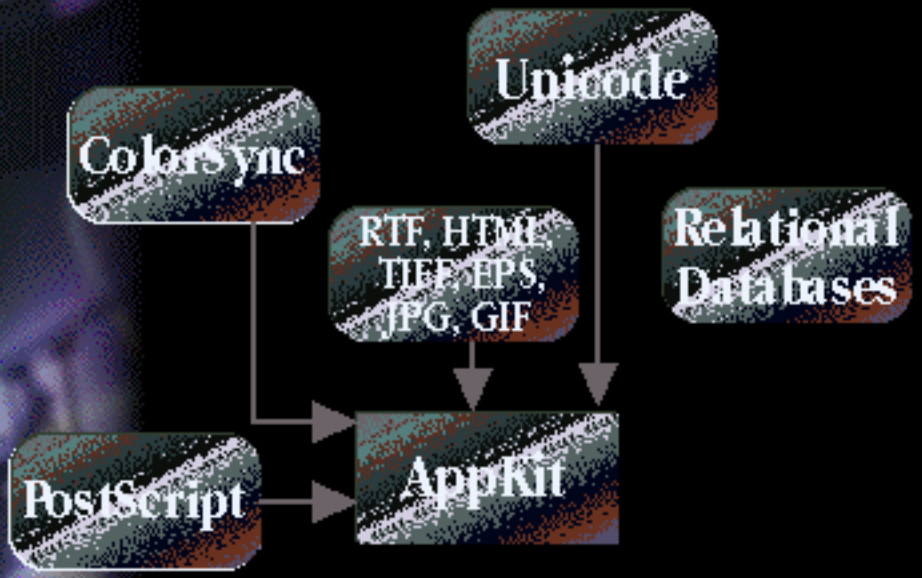
RTF, HTML,
TIFF, EPS,
JPG, GIF

PostScript

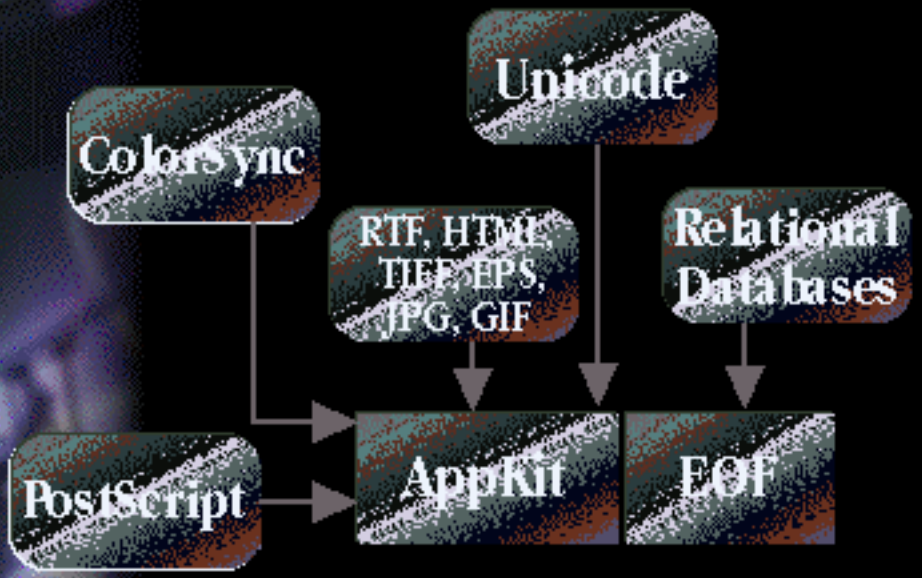
Technology Integration



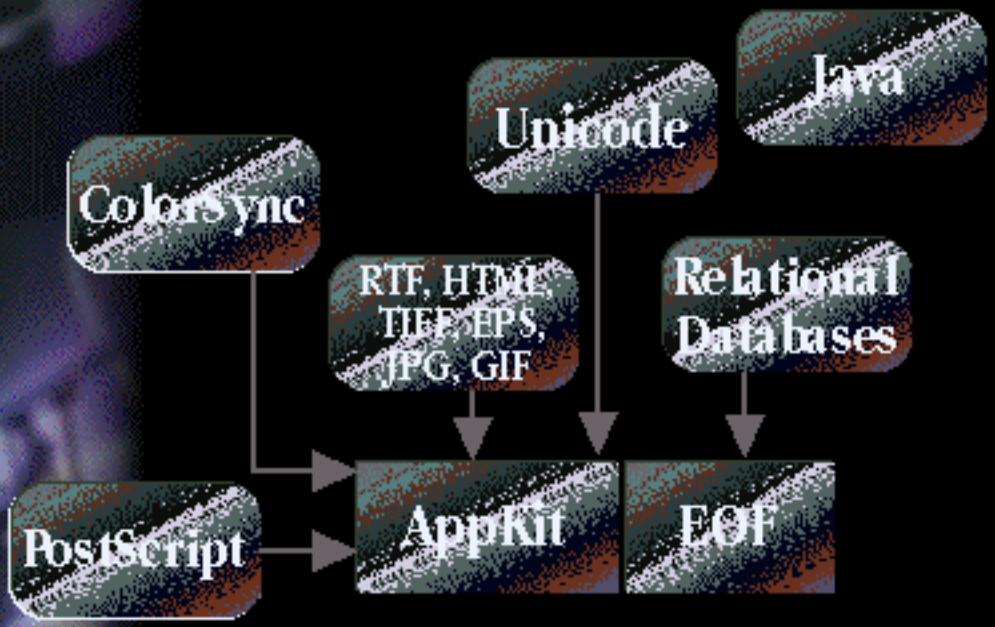
Technology Integration



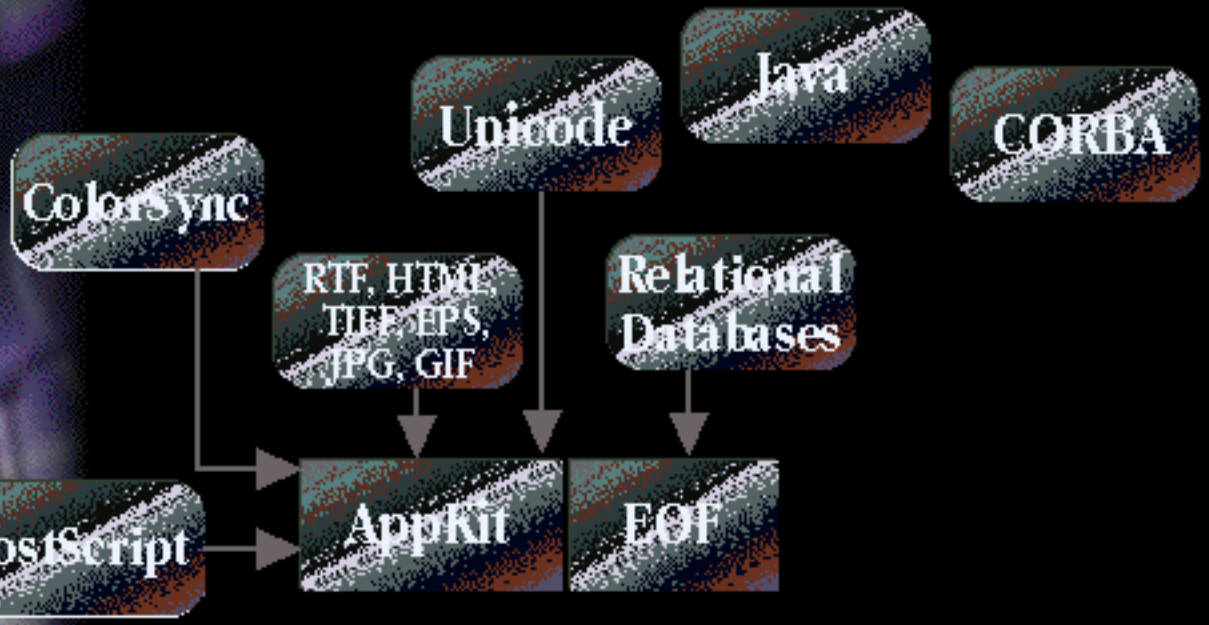
Technology Integration



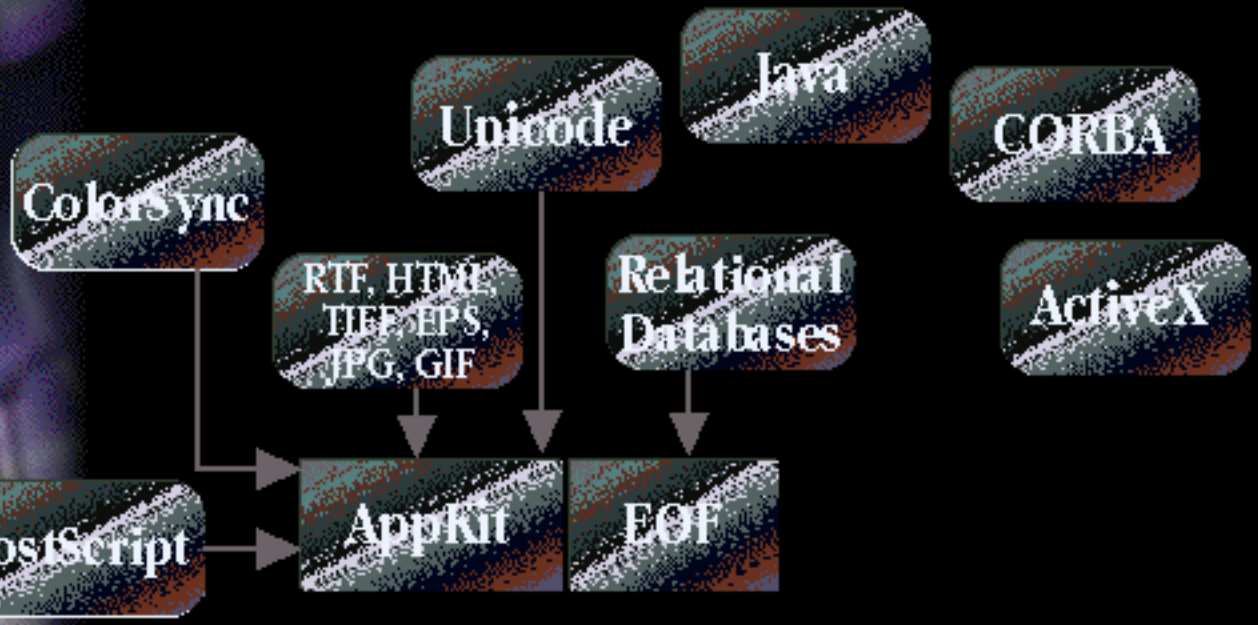
Technology Integration



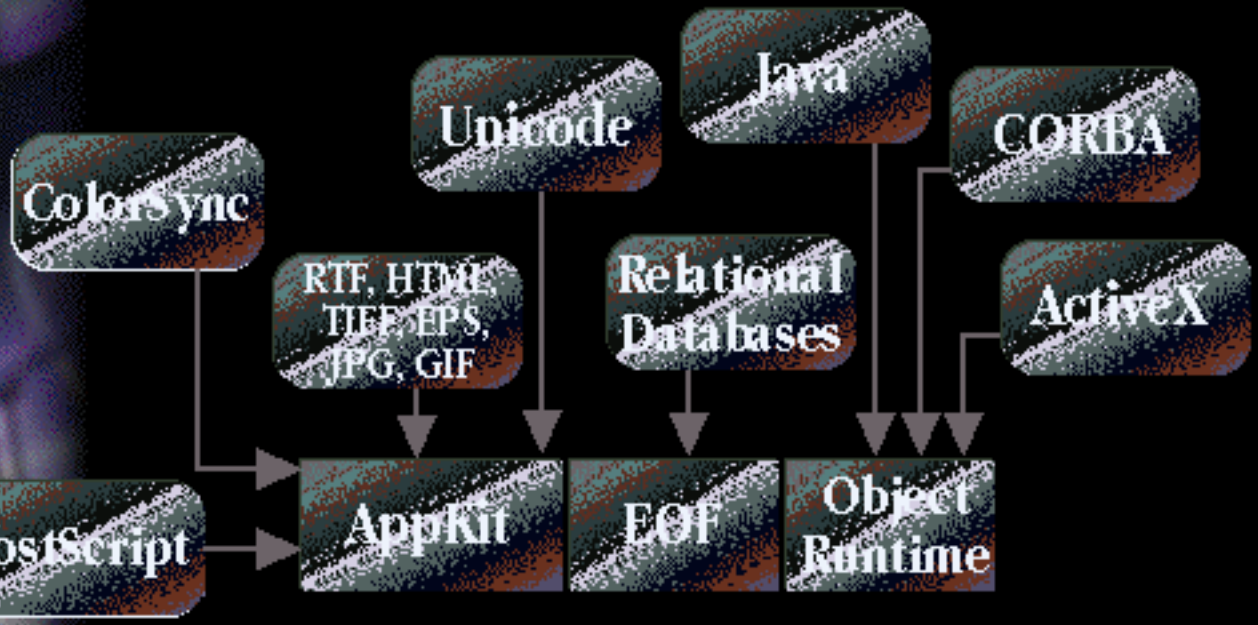
Technology Integration



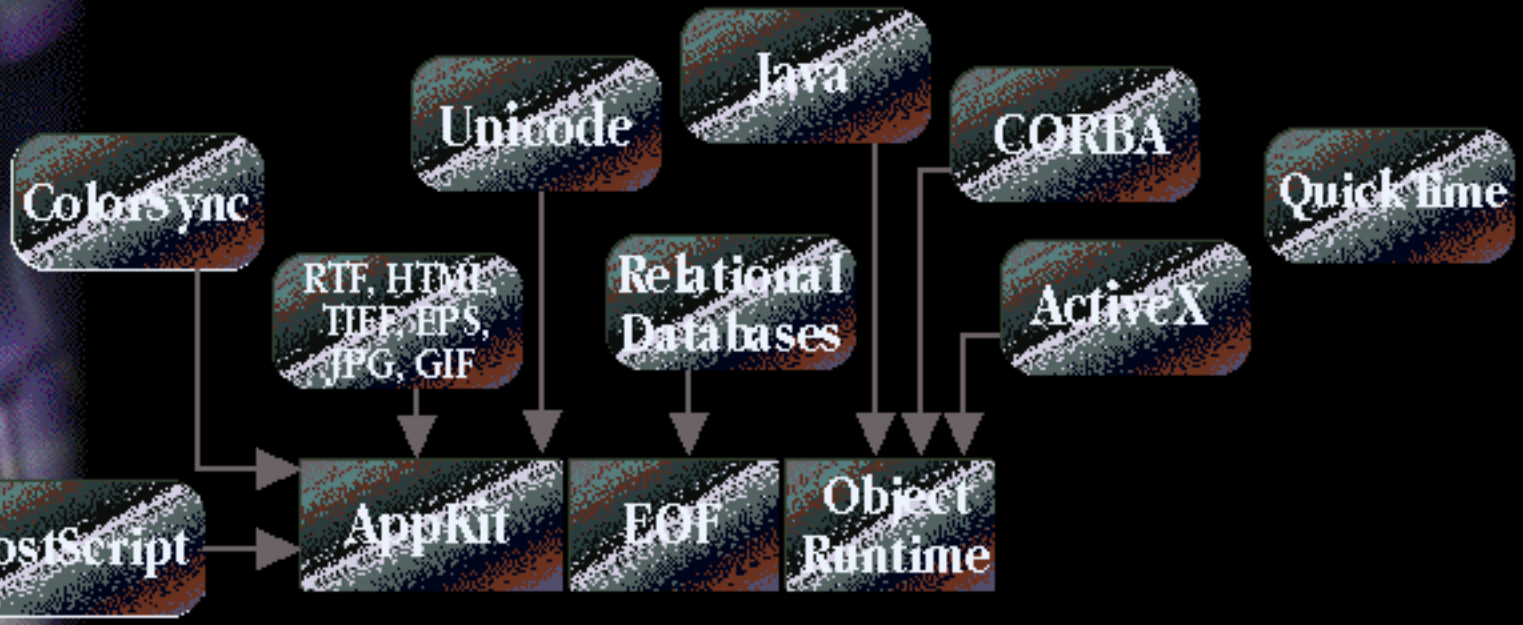
Technology Integration



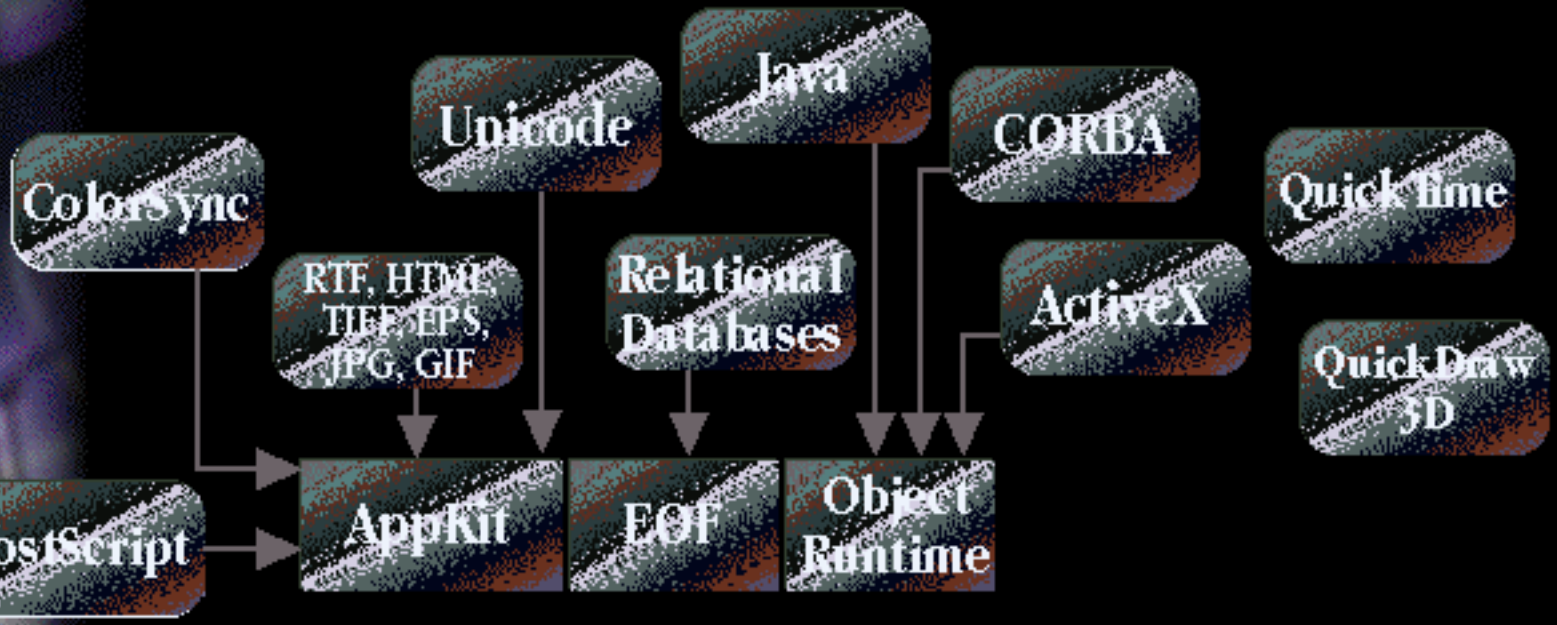
Technology Integration



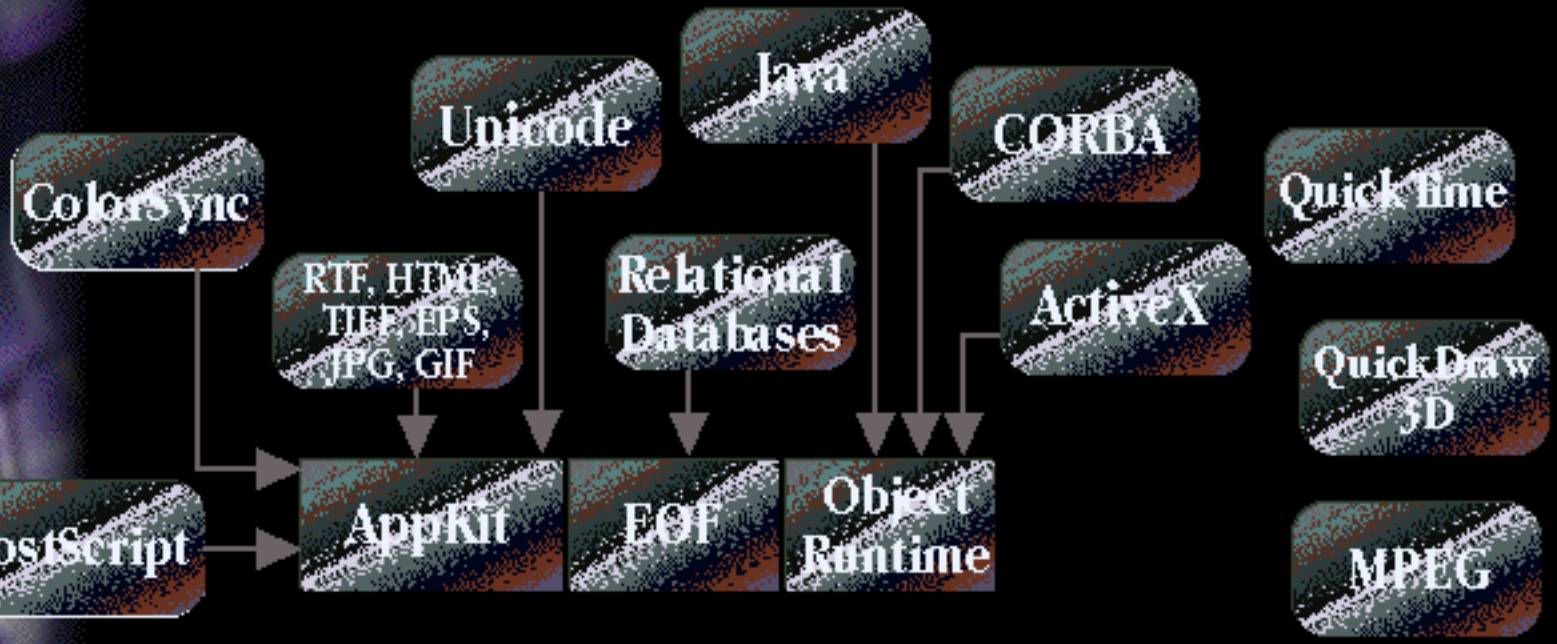
Technology Integration



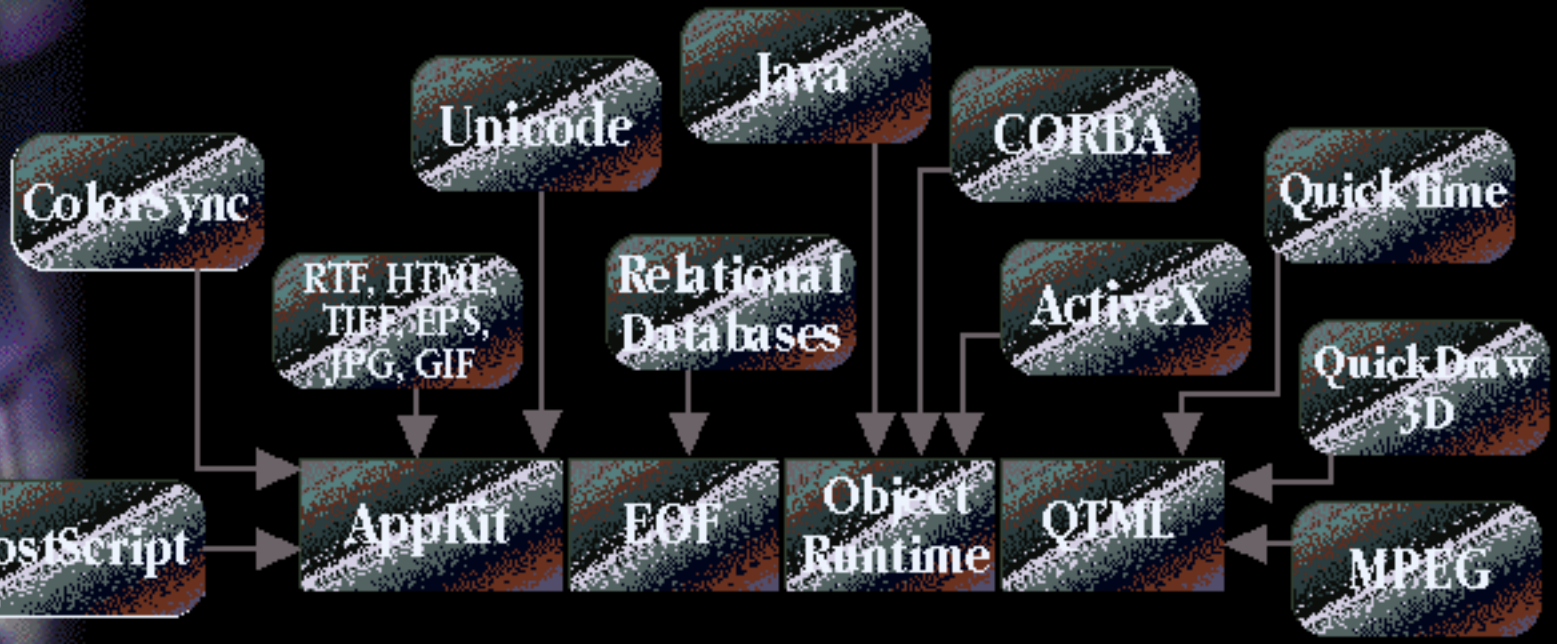
Technology Integration



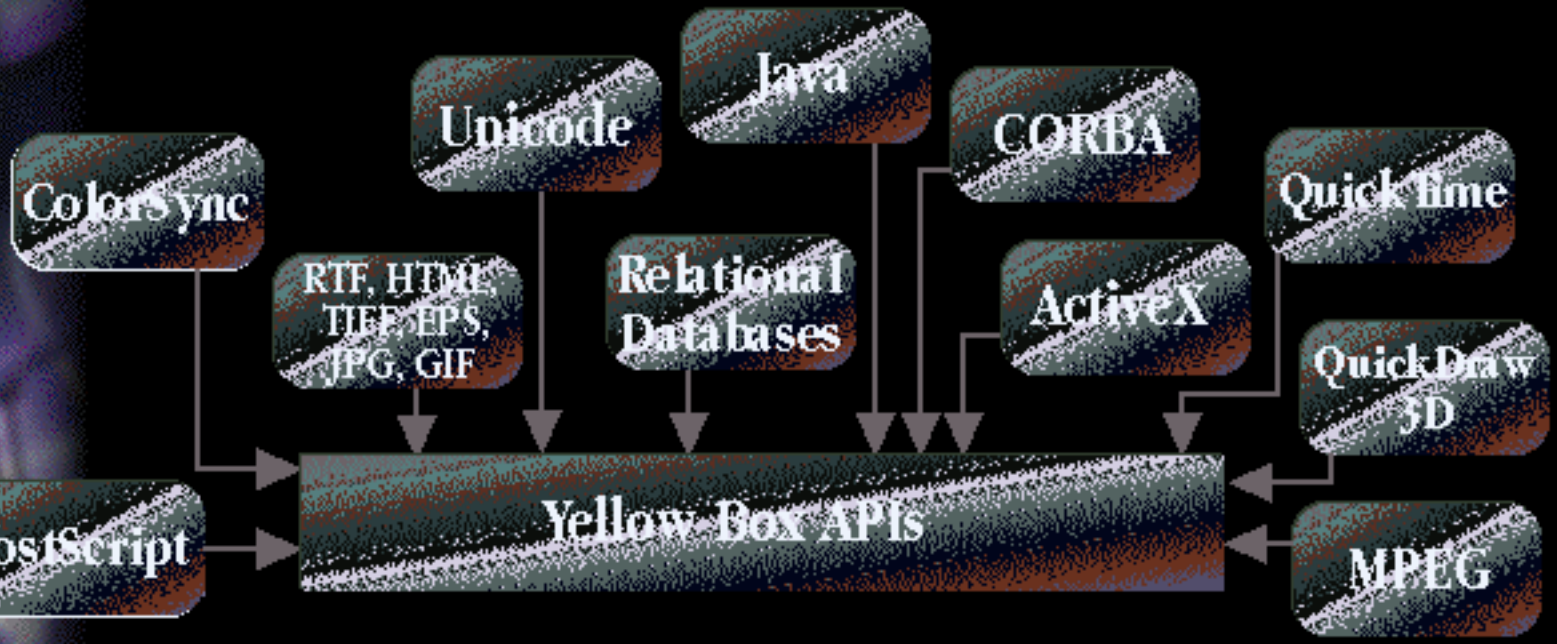
Technology Integration



Technology Integration



Technology Integration



Enable "ComponentWare"

Allow developers to leverage each other

Enable "ComponentWare"

Allow developers to leverage each other

- NSBundle API allows cross platform code packaging, including localized resources

Enable "ComponentWare"

Allow developers to leverage each other

- NSBundle API allows cross platform code packaging, including localized resources
- Interface Builder palettes allows drag and drop application "parts"

Enable "ComponentWare"

Allow developers to leverage each other

- NSBundle API allows cross platform code packaging, including localized resources
- Interface Builder palettes allows drag and drop application "parts"
  - Java Beans

Enable "ComponentWare"

Allow developers to leverage each other

- NSBundle API allows cross platform code packaging, including localized resources
- Interface Builder palettes allows drag and drop application "parts"
  - Java Beans
  - Active X (Windows only)

Enable "ComponentWare"

Allow developers to leverage each other

- NSBundle API allows cross platform code packaging, including localized resources
- Interface Builder palettes allows drag and drop application "parts"
  - Java Beans
  - Active X (Windows only)
- Enable custom "plug-in" architectures for applications (e.g., JavaBundle)

Make Apps Work Well Together

- Rich set of industry standard datatypes

Make Apps Work Well Together

- Rich set of industry standard datatypes
  - EPS, TIFF, RTF, HTML, GIF, Unicode, MPEG

Make Apps Work Well Together

- Rich set of industry standard datatypes
  - EPS, TIFF, RTF, HTML, GIF, Unicode, MPEG
- Rich set of data transfer services

Make Apps Work Well Together

- **Rich set of industry standard datatypes**
  - EPS, TIFF, RTF, HTML, GIF, Unicode, MPEG
- **Rich set of data transfer services**
  - Multiple pasteboards

Make Apps Work Well Together

- **Rich set of industry standard datatypes**
  - EPS, TIFF, RTF, HTML, GIF, Unicode, MPEG
- **Rich set of data transfer services**
  - Multiple pasteboards
  - Cross-platform drag and drop

Make Apps Work Well Together

- **Rich set of industry standard datatypes**
  - EPS, TIFF, RTF, HTML, GIF, Unicode, MPEG
- **Rich set of data transfer services**
  - Multiple pasteboards
  - Cross-platform drag and drop
- **Services architecture**

Make Apps Work Well Together

- **Rich set of industry standard datatypes**
  - EPS, TIFF, RTF, HTML, GIF, Unicode, MPEG
- **Rich set of data transfer services**
  - Multiple pasteboards
  - Cross-platform drag and drop
- **Services architecture**
  - Menu for standard operations on data

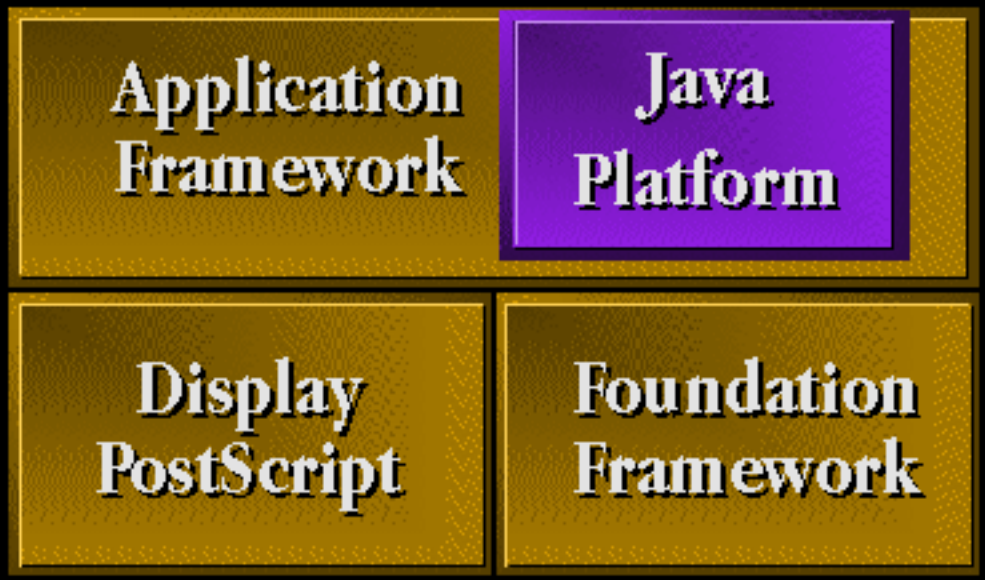
Make Apps Work Well Together

- **Rich set of industry standard datatypes**
  - EPS, TIFF, RTF, HTML, GIF, Unicode, MPEG
- **Rich set of data transfer services**
  - Multiple pasteboards
  - Cross-platform drag and drop
- **Services architecture**
  - Menu for standard operations on data
  - Filter services can extend datatypes through translation

Make Apps Work Well Together

- **Rich set of industry standard datatypes**
  - EPS, TIFF, RTF, HTML, GIF, Unicode, MPEG
- **Rich set of data transfer services**
  - Multiple pasteboards
  - Cross-platform drag and drop
- **Services architecture**
  - Menu for standard operations on data
  - Filter services can extend datatypes through translation
- **Scriptability "built-in"**

Yellow Box Functional Overview



Application Kit

- Presentation layer widgets

Application Kit

- **Presentation layer widgets**
  - "Pluggable" and native look and feel

Application Kit

- **Presentation layer widgets**
  - "Pluggable" and native look and feel
- **International text system**

Application Kit

- **Presentation layer widgets**
  - "Pluggable" and native look and feel
- **International text system**
- **Rich set of fundamental datatypes**

Application Kit

- **Presentation layer widgets**
  - "Pluggable" and native look and feel
- **International text system**
- **Rich set of fundamental datatypes**
- **Data transfer services**

Application Kit

- **Presentation layer widgets**
  - "Pluggable" and native look and feel
- **International text system**
- **Rich set of fundamental datatypes**
- **Data transfer services**
  - Drag and drop, pasteboard

Application Kit

Application Kit

- Application Framework

Application Kit

- **Application Framework**
  - More info:
    "Intro to OpenStep Application Framework", Thursday, 1:50, Hall 1

Application Kit

- **Application Framework**
  - More info:
    "Intro to OpenStep Application Framework", Thursday, 1:50, Hall 1
- **International text system**

Application Kit

- **Application Framework**
  - More info:
    "Intro to OpenStep Application Framework", Thursday, 1:50, Hall 1
- **International text system**
  - More info:
    "Rhapsody Text System and Localization"
    Wednesday, 4:30, A-1

Foundation Kit

- Collection classes (strings, arrays, dictionaries, etc.)

Foundation Kit

- Collection classes (strings, arrays, dictionaries, etc.)
- Automatic memory management

Foundation Kit

- Collection classes (strings, arrays, dictionaries, etc.)
- Automatic memory management
- Operating system insulation

Foundation Kit

- Collection classes (strings, arrays, dictionaries, etc.)
- Automatic memory management
- Operating system insulation
  - File I/O, threads, tasks, etc.

Foundation Kit

- Collection classes (strings, arrays, dictionaries, etc.)
- Automatic memory management
- Operating system insulation
  - File I/O, threads, tasks, etc.
- RunLoop and notification services

Foundation Kit

- Collection classes (strings, arrays, dictionaries, etc.)
- Automatic memory management
- Operating system insulation
  - File I/O, threads, tasks, etc.
- RunLoop and notification services
- Loadable component packaging

Foundation Kit

- Collection classes (strings, arrays, dictionaries, etc.)
- Automatic memory management
- Operating system insulation
  - File I/O, threads, tasks, etc.
- RunLoop and notification services
- Loadable component packaging
- Distributed objects

Foundation Kit

- More info:
  "Intro to OpenStep Foundation Framework"
  Wednesday, 4:30, Hall 1

Display PostScript WindowServer

- Client/Server graphics server

Display PostScript WindowServer

- Client/Server graphics server
- PostScript imaging model

Display PostScript WindowServer

- Client/Server graphics server
- PostScript imaging model
- Compositing used as generalized Blit()

Display PostScript WindowServer

- Client/Server graphics server
- PostScript imaging model
- Compositing used as generalized Blit()
- EPS is structured graphics standard

Display PostScript WindowServer

- Client/Server graphics server
- PostScript imaging model
- Compositing used as generalized Blit()
- EPS is structured graphics standard
- Any NSView can print, fax, or generate EPS

Graphics and Imaging

- **More info:**
  - "Imaging Under Rhapsody"
    Wednesday, 5:50, Hall 1
  - "Printing Under Rhapsody"
    Thursday, 3:10, Hall 1

Java Application Integration

- Fully support 100% Pure Java code

Java Application Integration

- Fully support 100% Pure Java code
- High performance VM and AWT/JFC implementation

Java Application Integration

- Fully support 100% Pure Java code
- High performance VM and AWT/JFC implementation
- Native Yellow Box APIs will also be made available in Java

Java Application Integration

- Fully support 100% Pure Java code
- High performance VM and AWT/JFC implementation
- Native Yellow Box APIs will also be made available in Java
  - AppKit, Foundation, …

Java Application Integration

- Fully support 100% Pure Java code
- High performance VM and AWT/JFC implementation
- Native Yellow Box APIs will also be made available in Java
  - AppKit, Foundation, …
  - Write once, run on any Yellow platform (Mac OS, Windows NT/95, Rhapsody, ???)

Java Application Integration

- More info:
  “Building Java-based Applications for Rhapsody”
  Friday, 3:10, Hall A-2

Java and Object Runtimes

- **More info:**
  - "Uncommon Object Model: The Rhapsody Runtime" Thursday, 9:50, Room C
  - "Object-Oriented Programming and Languages" Thursday, 11:10, Room C

Enterprise Objects Framework

- Relational database to Object mapping

Enterprise Objects Framework

- Relational database to Object mapping
- Database independent

Enterprise Objects Framework

- Relational database to Object mapping
- Database independent
- Scales to multi-tier client/server model

Enterprise Objects Framework

- Relational database to Object mapping
- Database independent
- Scales to multi-tier client/server model
- Flexible transaction management

Enterprise Objects Framework

- More info:
  "Enterprise Object Frameworks Overview"
  Friday, 10:50, Hall A-1

Comparisons

Comparisons

Four major application platforms

Comparisons

Four major application platforms

- Mac Toolbox

Comparisons

Four major application platforms

- Mac Toolbox
- WIN32/MFC

Comparisons

Four major application platforms

- Mac Toolbox
- WIN32/MFC
- JavaSoft's "Java Platform"

Comparisons

Four major application platforms

- Mac Toolbox
- WIN32/MFC
- JavaSoft's "Java Platform"
- Apple's "Yellow Box Platform"

Microsoft's WIN32/MFC

Microsoft's WIN32/MFC

- Lots of shrinkwrap

Microsoft's WIN32/MFC

- Lots of shrinkwrap
- Not cross-platform

Microsoft's WIN32/MFC

- Lots of shrinkwrap
- Not cross-platform
- Procedural, low-level APIs

Microsoft's WIN32/MFC

- Lots of shrinkwrap
- Not cross-platform
- Procedural, low-level APIs
- Objects are an afterthought

Microsoft's WIN32/MFC

- Lots of shrinkwrap
- Not cross-platform
- Procedural, low-level APIs
- Objects are an afterthought
- Java viewed as a language only

Microsoft's WIN32/MFC

- Lots of shrinkwrap
- Not cross-platform
- Procedural, low-level APIs
- Objects are an afterthought
- Java viewed as a language only
- Built-in datatypes not cross platform (BMP, GDI Metafiles, character encoding, etc.)

JavaSoft's Java Platform

JavaSoft's Java Platform

- Object APIs only

JavaSoft's Java Platform

- Object APIs only
- Cross-platform, processor independent

JavaSoft's Java Platform

- Object APIs only
- Cross-platform, processor independent
- Procedural C-based APIs "impure"

JavaSoft's Java Platform

- Object APIs only
- Cross-platform, processor independent
- Procedural C-based APIs "impure"
- Not mature technology

JavaSoft's Java Platform

- Object APIs only
- Cross-platform, processor independent
- Procedural C-based APIs "impure"
- Not mature technology
- Performance and scalability concerns

JavaSoft's Java Platform

- Object APIs only
- Cross-platform, processor independent
- Procedural C-based APIs "impure"
- Not mature technology
- Performance and scalability concerns
- Fundamental datatypes lacking in certain areas (Rich Text, Structured Graphics,...)

JavaSoft's Java Platform

- Object APIs only
- Cross-platform, processor independent
- Procedural C-based APIs "impure"
- Not mature technology
- Performance and scalability concerns
- Fundamental datatypes lacking in certain areas (Rich Text, Structured Graphics,...)
- Not ready for shrinkwrap

Apple's Yellow Box Platform

Apple's Yellow Box Platform

- Superset of Java platform

Apple's Yellow Box Platform

- Superset of Java platform
- Highly portable implementation

Apple's Yellow Box Platform

- Superset of Java platform
- Highly portable implementation
- Mature, proven technology

Apple's Yellow Box Platform

- Superset of Java platform
- Highly portable implementation
- Mature, proven technology
- Object APIs primarily

Apple's Yellow Box Platform

- Superset of Java platform
- Highly portable implementation
- Mature, proven technology
- Object APIs primarily
- Procedural APIs OK, too!

Apple's Yellow Box Platform

- Superset of Java platform
- Highly portable implementation
- Mature, proven technology
- Object APIs primarily
- Procedural APIs OK, too!
- Wide variety of cross-platform datatypes

Apple's Yellow Box Platform

- Superset of Java platform
- Highly portable implementation
- Mature, proven technology
- Object APIs primarily
- Procedural APIs OK, too!
- Wide variety of cross-platform datatypes
- Ready for shrinkwrap today

Apple's Yellow Box Platform

- Superset of Java platform
- Highly portable implementation
- Mature, proven technology
- Object APIs primarily
- Procedural APIs OK, too!
- Wide variety of cross-platform datatypes
- Ready for shrinkwrap today
- Great basis for future innovation

Opportunities

Use the best technology for the job:

Opportunities

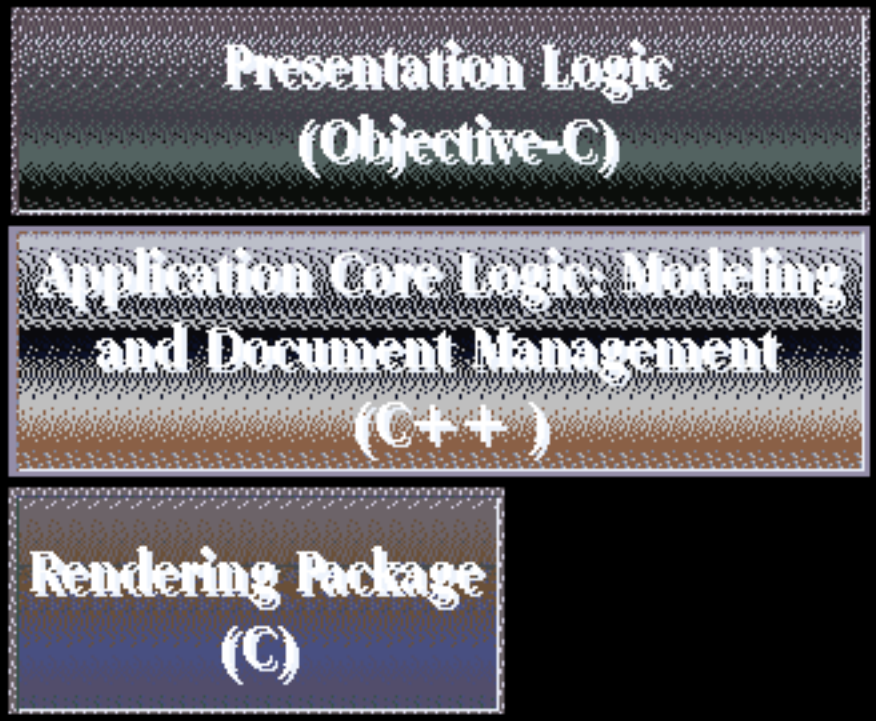
Use the best technology for the job:

Opportunities

Use the best technology for the job:

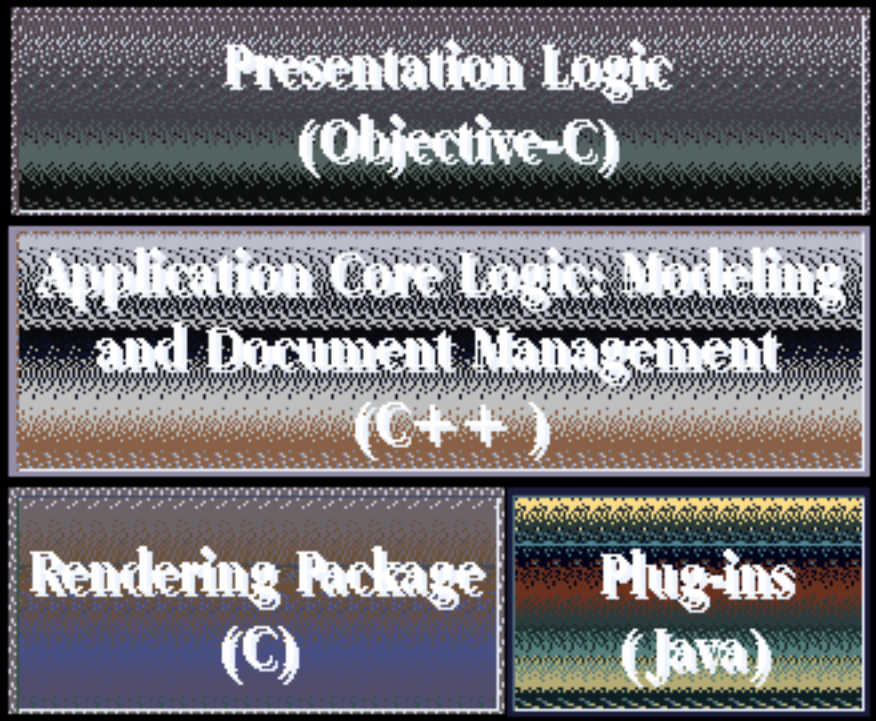
Opportunities

Use the best technology for the job:



Opportunities

Use the best technology for the job:



Summary…

- Yellow Box is a robust, cross-platform, application substrate
- Ready for shrinkwrap today


Q&A


Worldwide
Developers
Conference