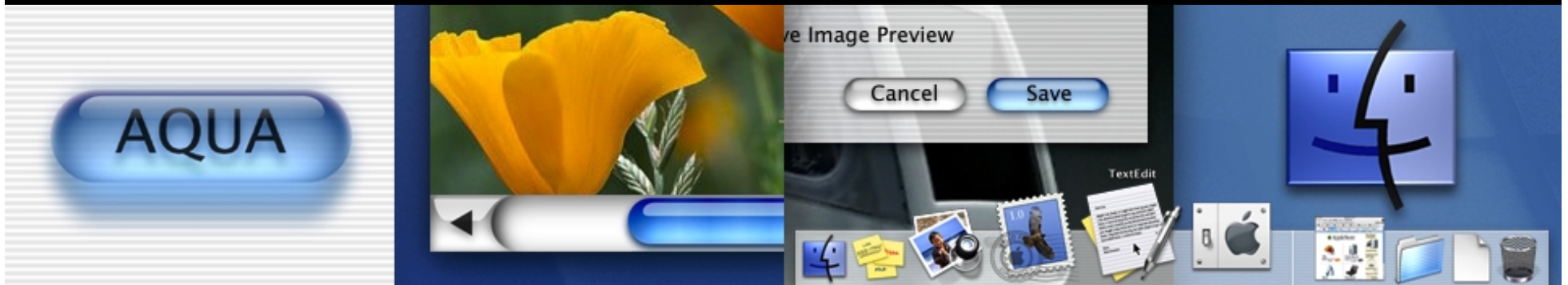




Session 401

WebObjects Technical Overview



Steve Hayman
Systems Engineer, Apple iServices

Introduction

- Understand the WO architecture
- See the tools, frameworks, and development process in action



What You'll Learn

- WO Tools
- WO Frameworks
- WO Deployment
- Understand the titles of the other WO sessions

**406—Rapid Development
with Direct to Java Client**

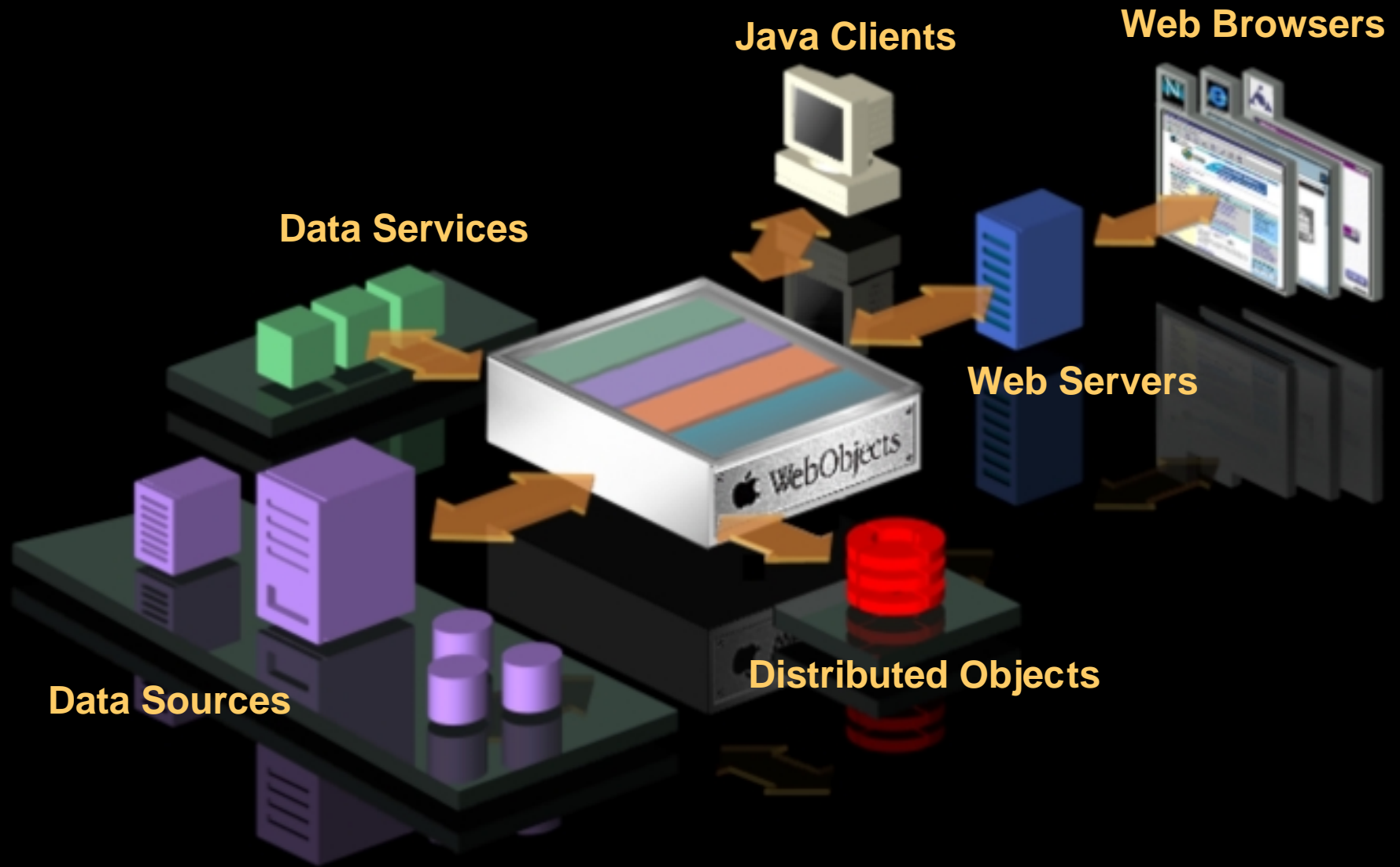


What WebObjects Does

- It's an *Application Server*
- Who invented that term anyway?
- It's more than just that
 - Developer Tools
 - Pre-written objects (Frameworks) ← The best part!
 - Runtime environment
 - Monitoring and control tools



For Instance



Who WebObjects Is For

- Developers
- Very powerful
- Steep learning curve
- But that's OK



WebObjects Works With . . .

- Java
- Databases—Oracle, Informix, Sybase, SQL Server, LDAP, OpenBase, FrontBase, . . .
- Mainframes—CORBA, 3270 screen-scraping
- Different UIs—HTML, WAP, PDF, XML
- Most web servers—Apache, Netscape, IIS
- Lots of 3rd-party objects and APIs
- Open open open open open open open open



Where You Can Use WebObjects

- Develop on Mac OS X, Mac OS X Server, or Windows NT
- Deploy on OS X, OS X Server, NT, Solaris, HP/UX, Linux
 - (Note: database support varies by platform)
- Mix and match as needed
- Start small and grow bigger

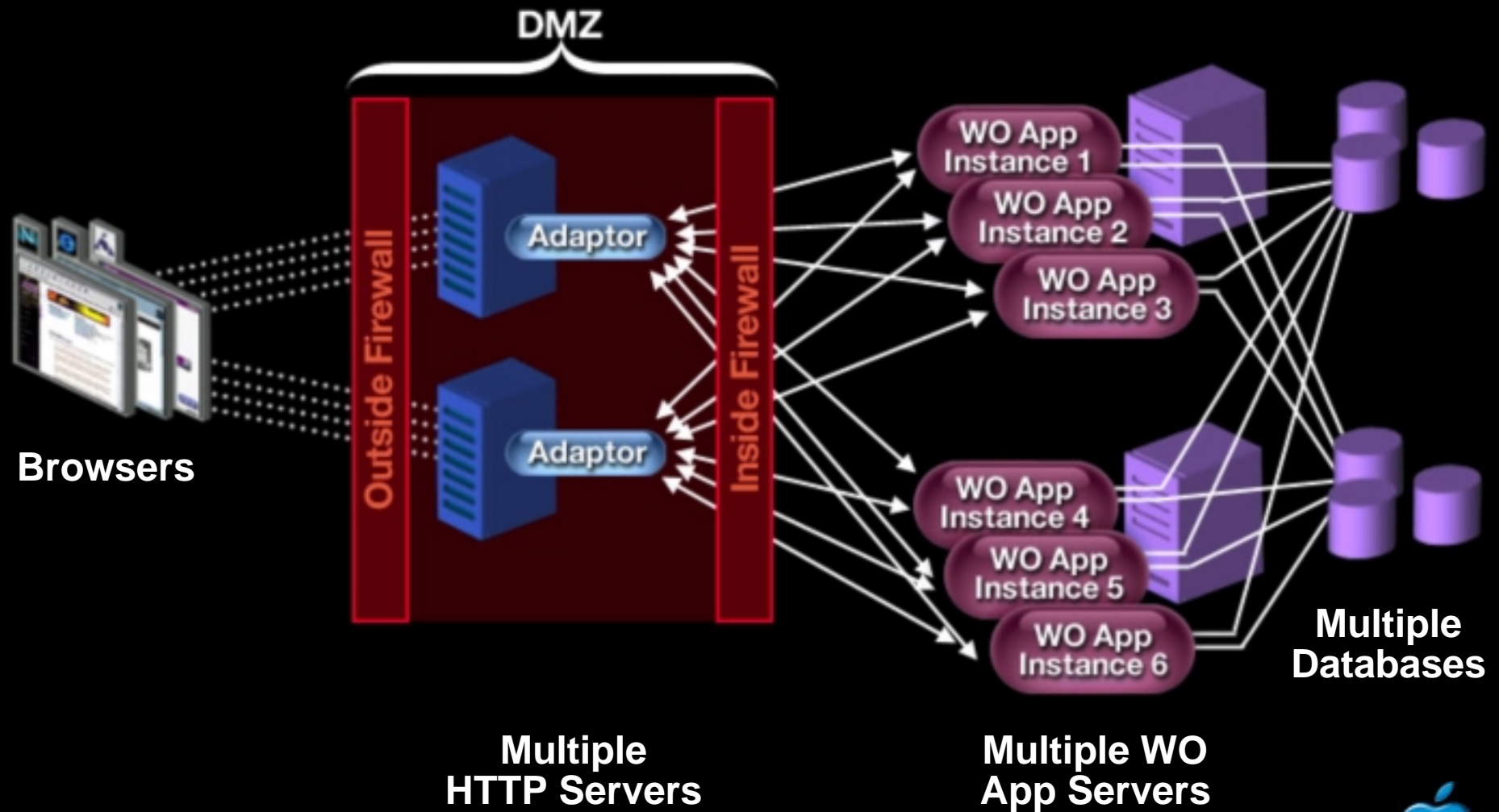


Start With This...



...And Scale Easily to This

417—Building
Large-Scale Apps



Key Ideas

- Web Components (front-end)
- Enterprise Objects (back-end)
- Kept completely separate
- Bound together at runtime



Web Components

- HTML templates
- +
- `<WEBOBJECT >` tags that indicate where something interesting and dynamic happens
- +
- Bindings that fill in the blanks
- +
- Lots of pre-written objects to reuse



What Comes Out of a Component?

- A string—of your choice
 - Plain text
 - HTML
 - Image data
 - JavaScript commands
 - AppleScript commands
 - ...



Non-HTML Components Too

- PDF
- WAP
- XML

Mix and match



Enterprise Objects

- Java objects that implement your business rules...
 - Customer, Order, Product, Vehicle, Package, Option
- Adaptors that speak to databases
 - Oracle, Sybase, Informix, LDAP...
- Models that connect objects to databases
 - Customer ↔ Oracle CUST table
 - Vehicle ↔ Informix VHCL table
- Relationships between objects
 - Customer has a list of Vehicles



EOF Is the Real Gem Here!

- Pure business objects
- Totally UI independent
- Totally database independent
- Persistence is completely managed for you

**403—Persistent Object
Modeling with EOModeler**



(And Coming Soon—EJB)

**407—WebObjects:
EJB**



... All Bound Together ...

- WO app
 - Loads components
 - Finds **<WEBOBJECT>** tags
 - Fetches and sends messages to Enterprise Objects
 - Mixes results into the component
- User sees plain HTML (PDF, WAP, XML)
- Customize the entire process at runtime if needed



...With Advanced Session Management...

- WO manages an extensible Session object for each individual user
- Also an Application object shared by all
- Complete control over session creation and archiving
- Extend Session to add security, shopping carts, user preferences, personalization and so on



... and Runtime Monitoring Tools

- Monitor
 - WO application that manages your big honkin' deployment environment

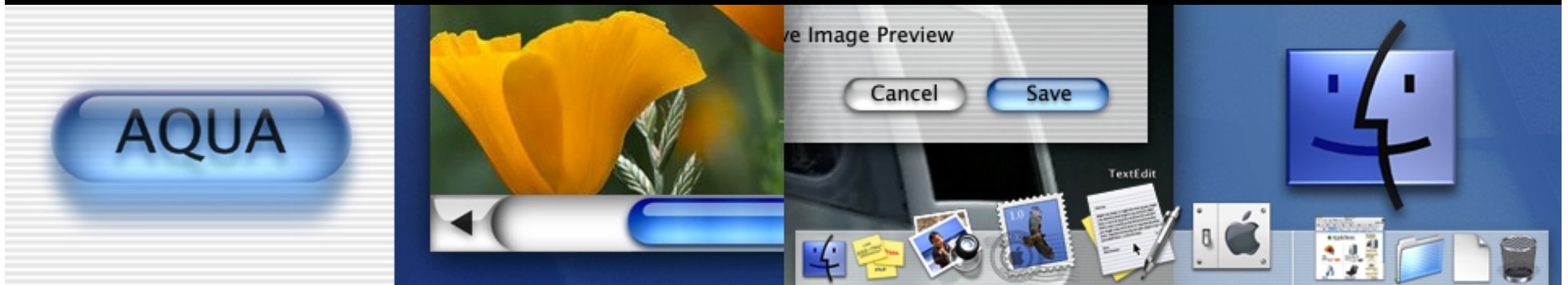
**404—Deploying
Applications**



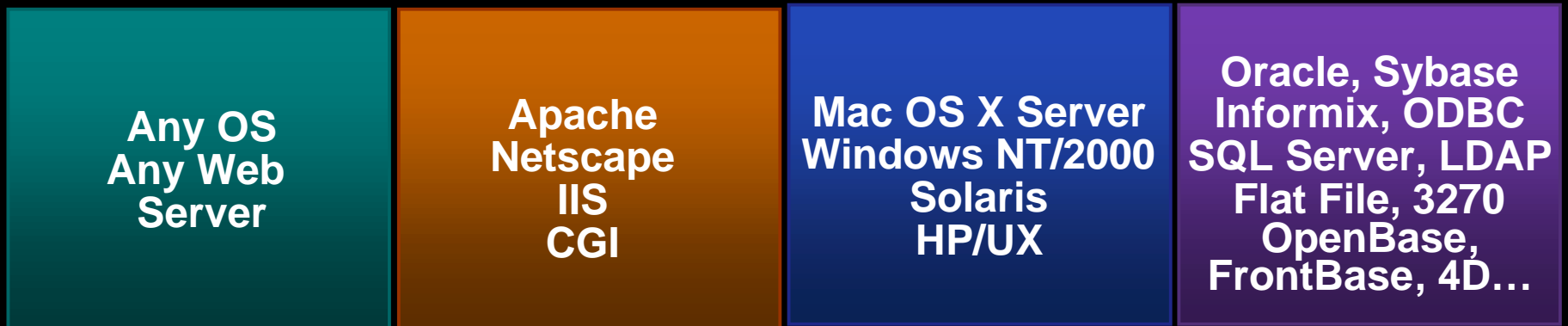
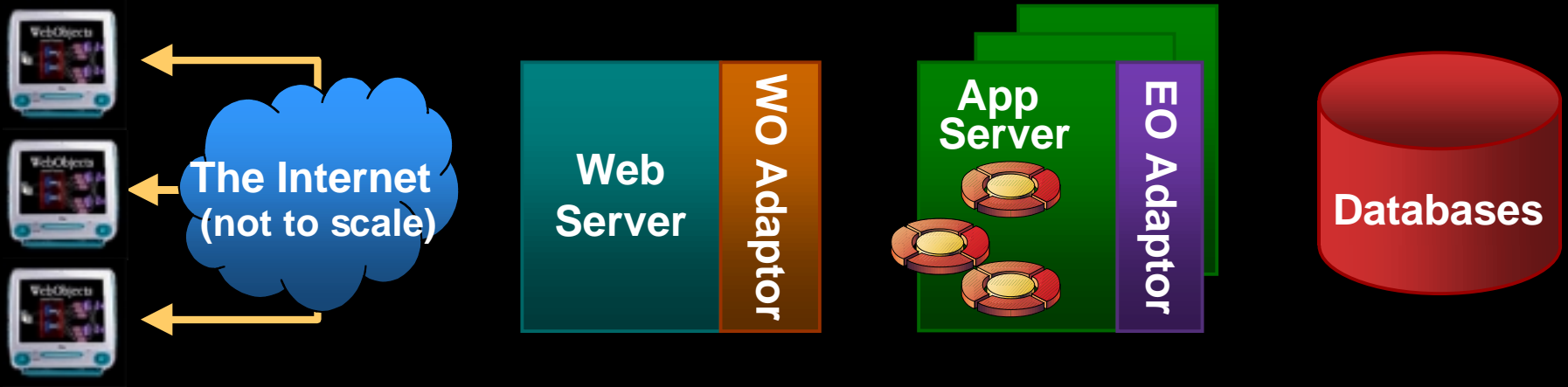


Session 401

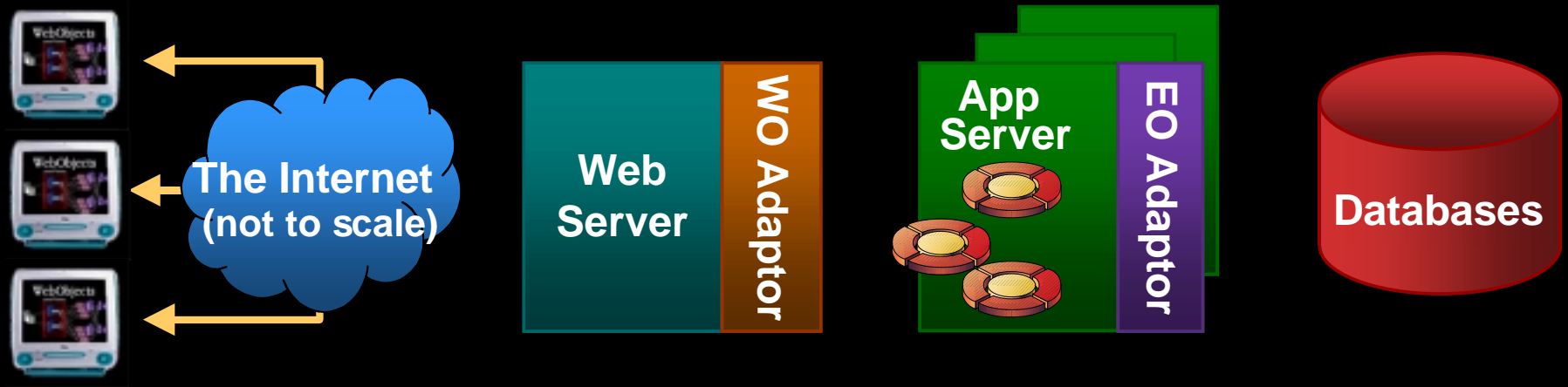
WebObjects in Action



WebObjects in Action



A Really Simple App



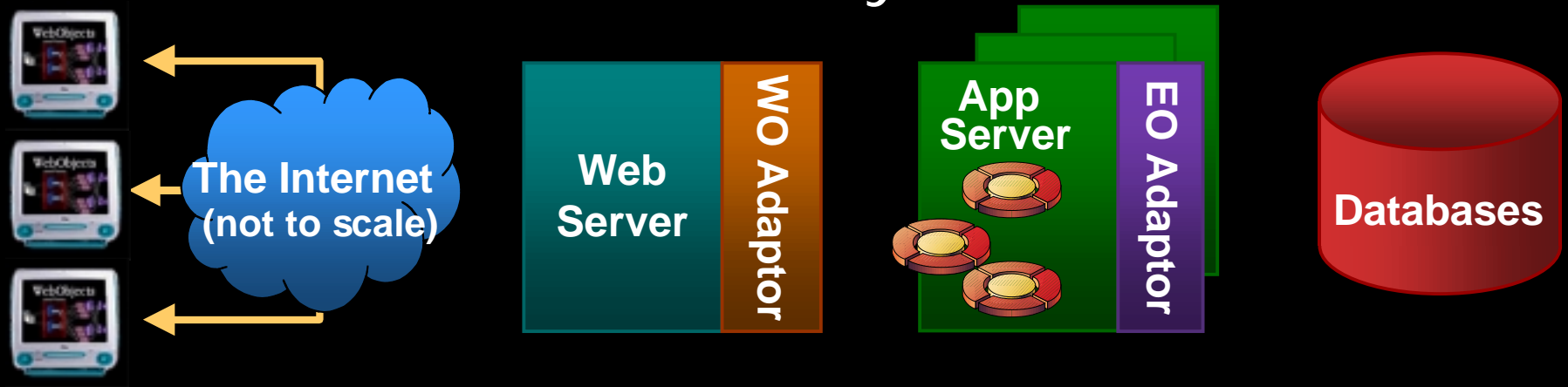
1. Process a Request



2. Generate a Response



Three Steps WO Does for You Automatically



1. Take Values from a Request

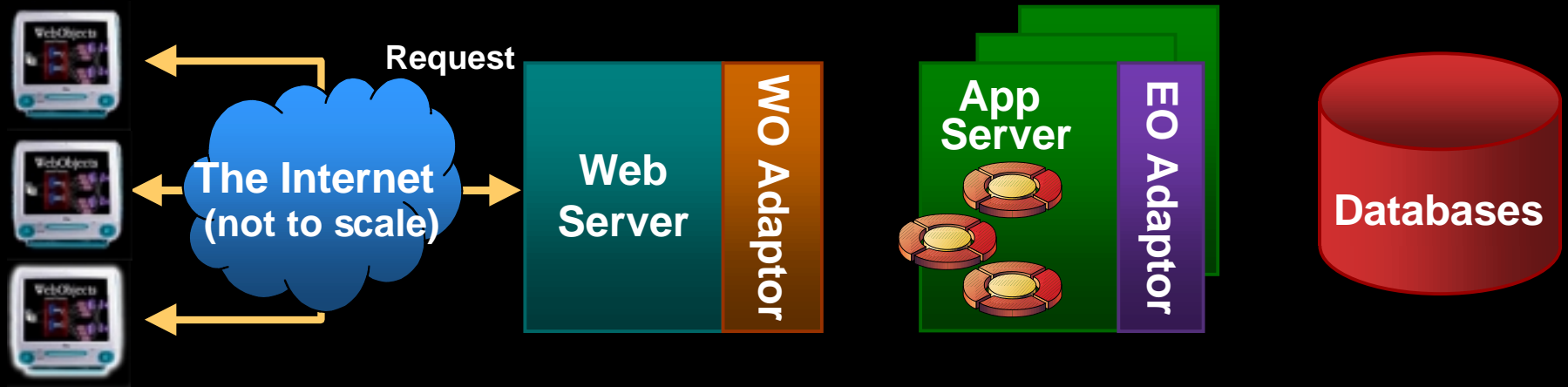


3. Generate a Response

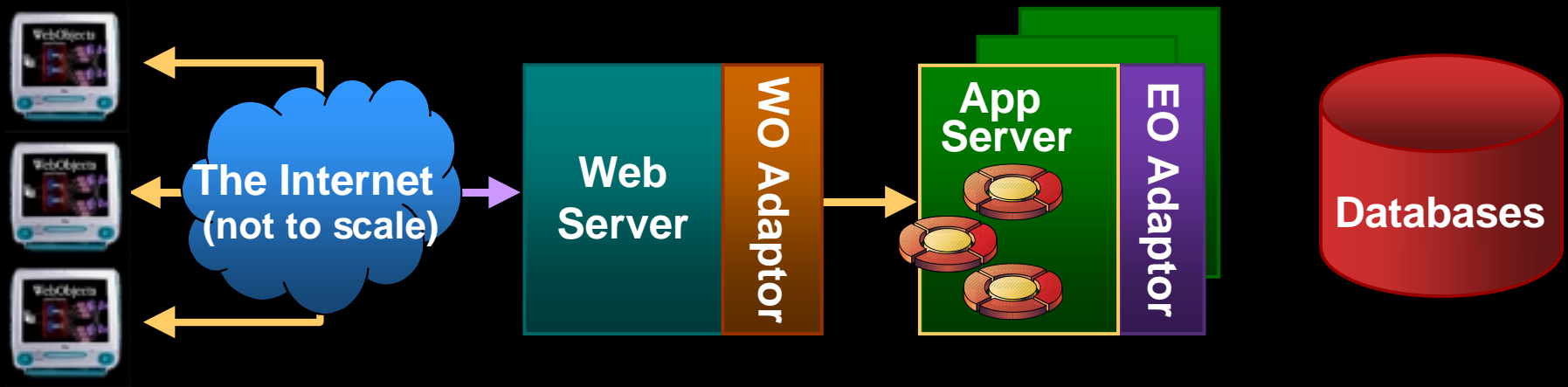
2. Invoke an Action



User Submits Request to Web Server



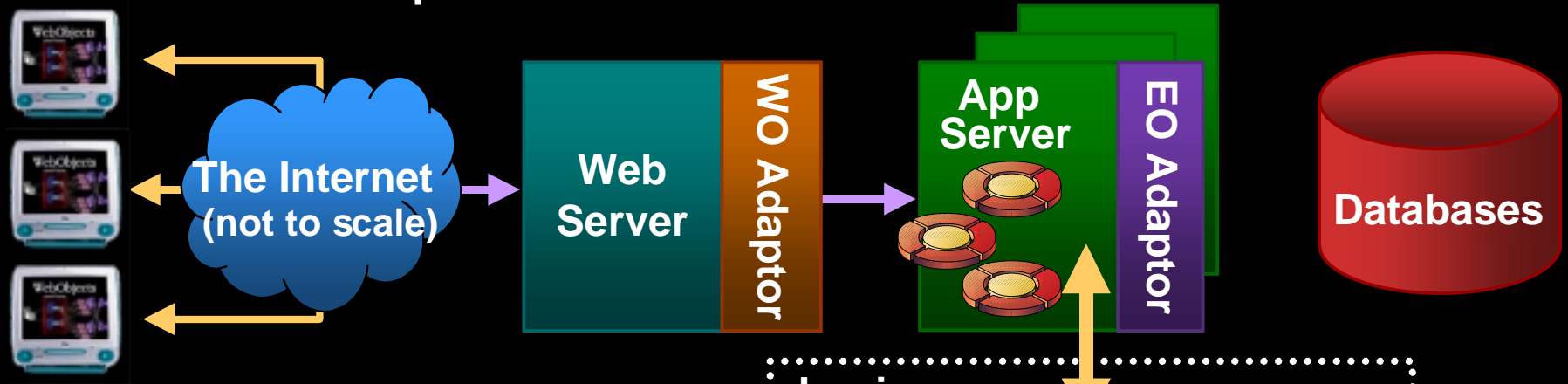
User Submits Request to Web Server



- Web server adaptor finds appropriate application server and instance, and forwards the request to it
 - First request? A random instance is chosen, and a new Session created
 - Subsequently? Return to the original one



Session Finds the Requesting WOComponent

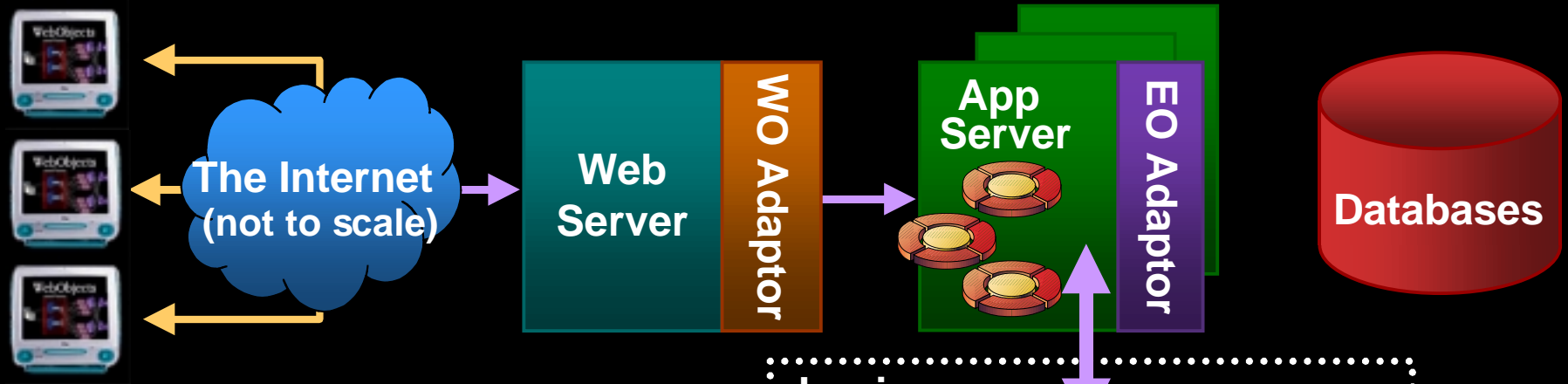


Login.wo

```
<HTML>
What's your user name?
<WO NAME=FORM>
<WO NAME=INPUT>
<WO NAME=SUBMIT>
```

```
INPUT: WOString
{ value = userName; }
SUBMIT: WOSubmitButton {
action = handleLogin; }
```





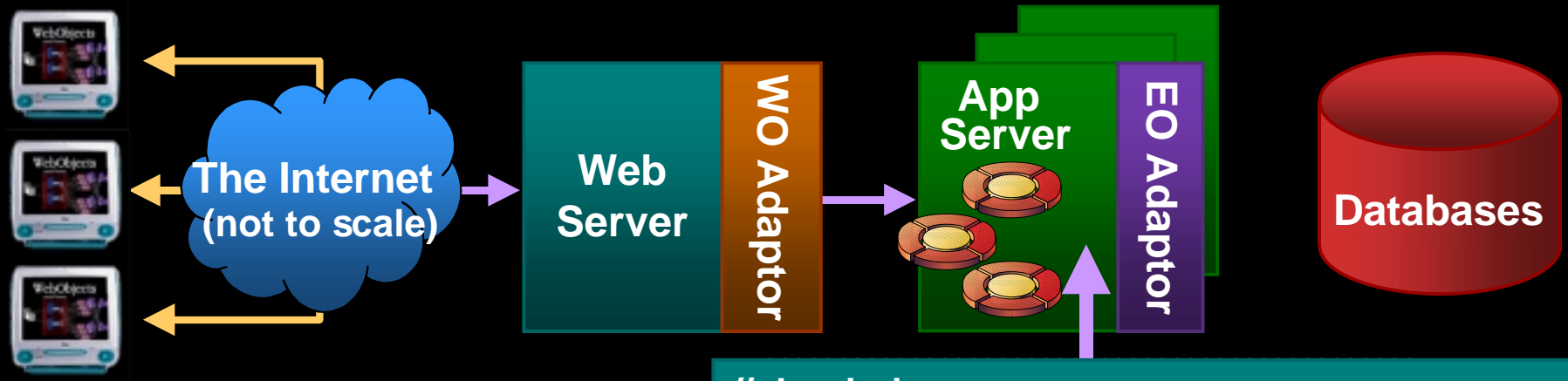
- The requesting component studies any form values that have been submitted

Login.wo

```
<HTML>
What's your user name?
<WO NAME=FORM>
<WO NAME=INPUT>
<WO NAME=SUBMIT>
```

```
INPUT: WOString
{ value = userName; }
SUBMIT: WOSubmitButton {
  action = handleLogin; }
```



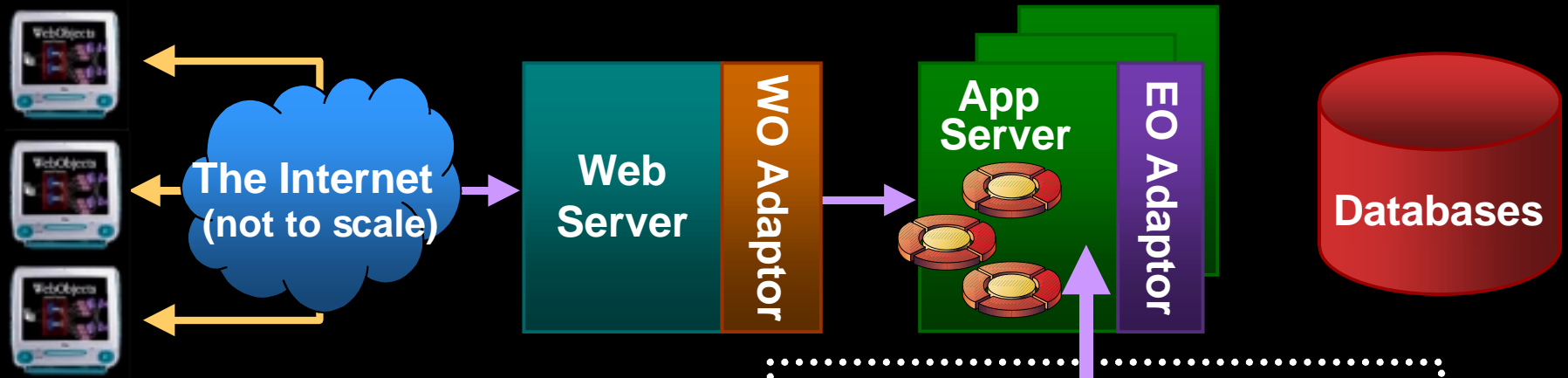


- Requesting component then figures out what action to invoke
 - Action returns another component

```
// Login.java
public class Login extends WOComponent {
    public String userName;
    public WOComponent handleLogin() {
        session().fetchCustomer(userName);
        return pageWithName("Welcome");
    }
}
```

```
{ value = userName; }
SUBMIT: WOSubmitButton {
    action = handleLogin; }
```





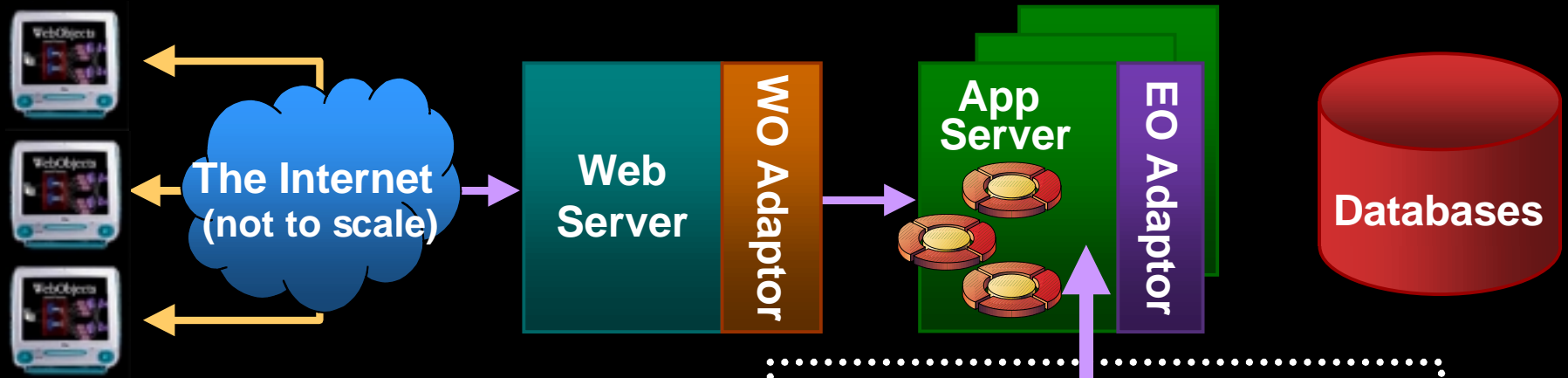
- Action is invoked, and session loads the resulting response Component

Welcome.wo

```
<HTML>
Welcome back,
<WEBOBJECT
NAME=UserName>
```

```
UserName: WOString
{ value =
  session.customer.firstName;
}
```





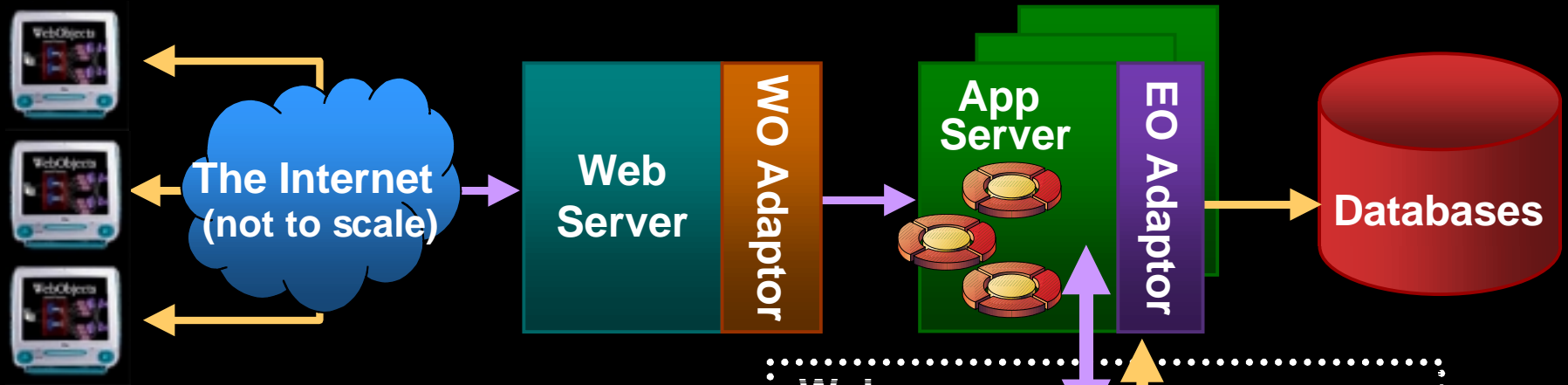
- WEBOBJECT tags and corresponding messages are identified

Welcome.wo

```
<HTML>
Welcome back,
<WEBOBJECT
NAME=UserName>
```

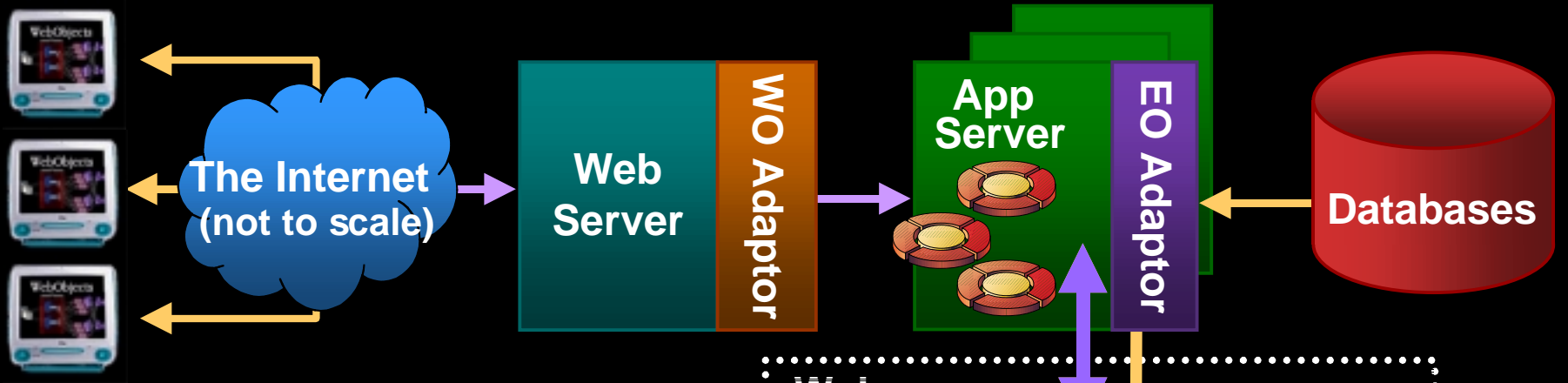
```
UserName: WOString
{ value =
  session.customer.firstName;
}
```



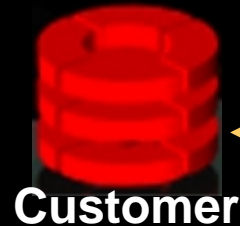


- Any needed objects are fetched from the database
- SQL is generated automatically





- Rows are returned from the database and converted into Enterprise Objects



Welcome.wo
 <HT
 Wel
 <W
 NAME=UserName>

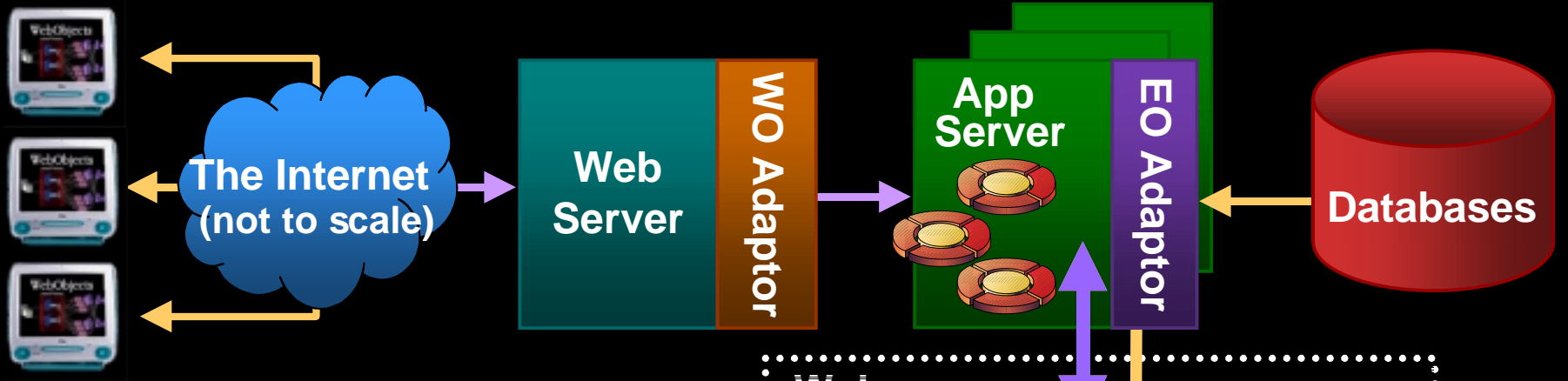
"Steve", "Hayman",
 "shayman"

EOModel

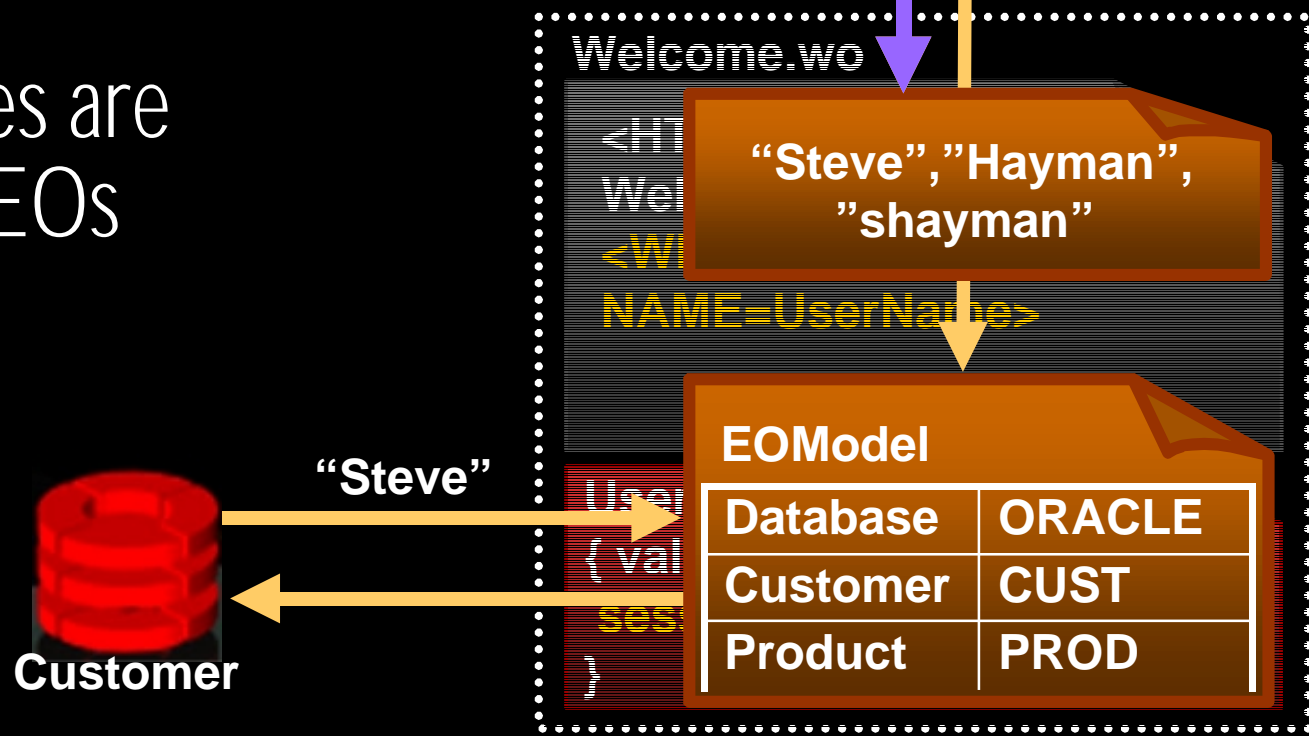
Database	ORACLE
Customer	CUST
Product	PROD

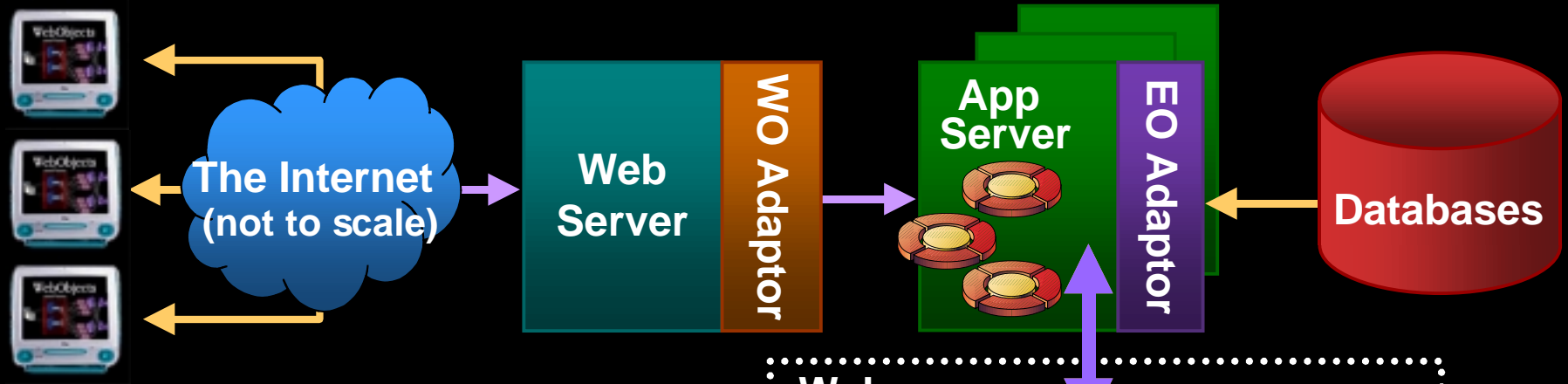
User
 { val
 ses



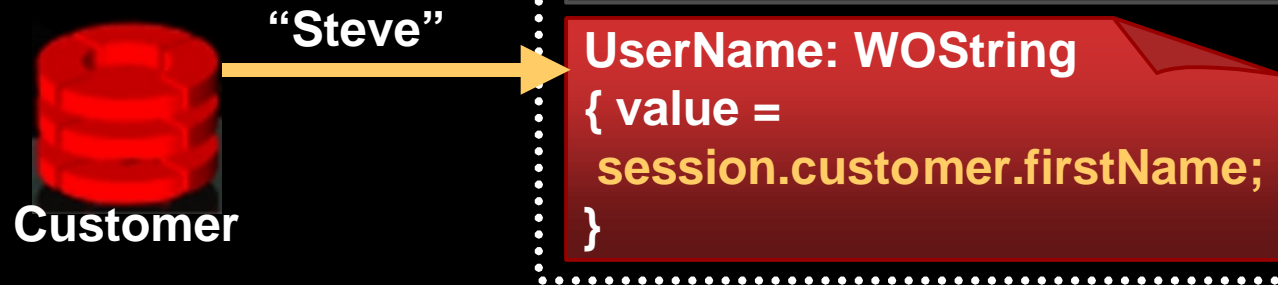


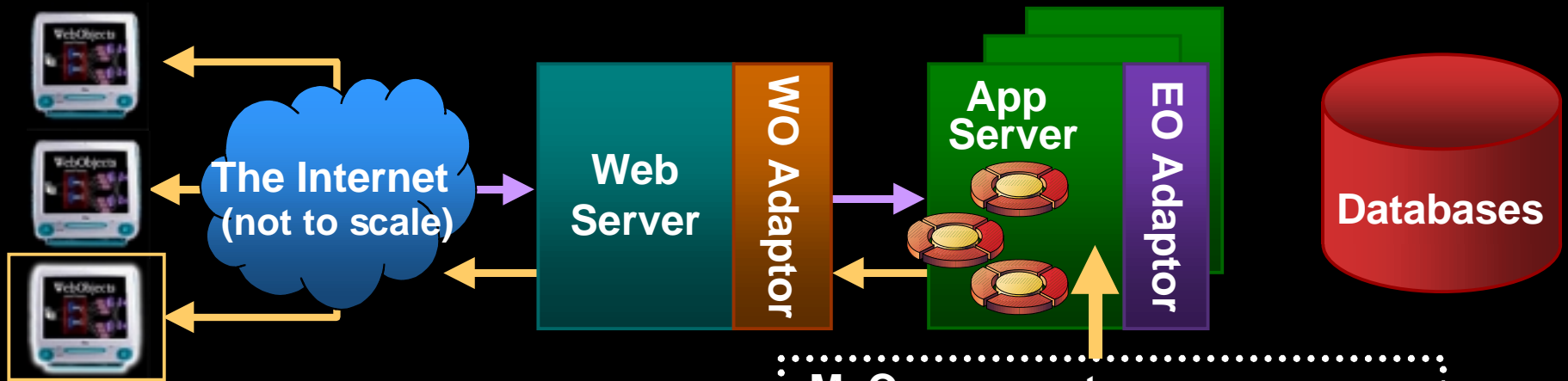
- Messages are sent to EOs





- Response is substituted into HTML template





- Session sends plain HTML back to user



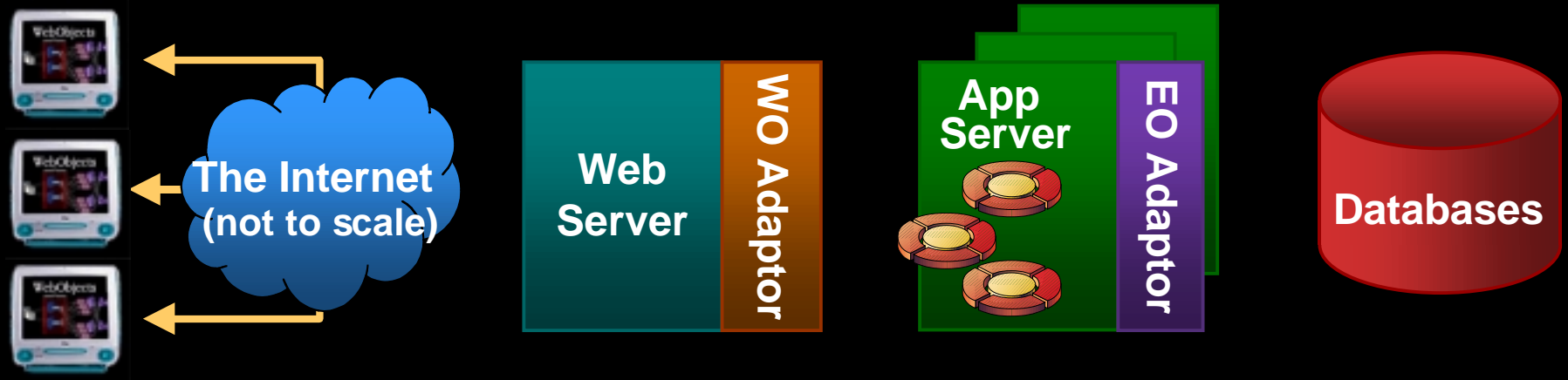
MyComponent.wo

```
<HTML>
Welcome back,
Steve
```

```
UserName: WOString
{ value =
  session.customer.firstName;
}
```



That's the Request/Response Loop



```
public void takeValuesFromRequest
```

```
public WOEElement invokeAction
```

```
public void appendToResponse  
aResponse,
```

```
( WORequest aRequest,  
  WOContext aContext )
```

```
( WORequest aRequest,  
  WOContext aContext )
```

```
( WOResponse
```

```
  WOContext aContext )
```

- Application, Session, and all Components participate (If they want to)



Please Note!

- No mixing of SQL and HTML
- Database, interface, business logic are all completely separate and independent
- No need to write SQL or HTML at all (unless you want to)



The Tools



The Project Builder software box features a white Apple logo at the top center. Below the logo, the text "Project Builder" is displayed in a large, white, serif font. Underneath the product name, the URL "http://www.apple.com" is written in a smaller white font. At the bottom of the box, the copyright notice "© 1992-1999 Apple Computer, Inc., All Rights Reserved." is printed in a small white font. The background of the box is a dark, textured grey with a faint, abstract pattern.

Apple

Project Builder

<http://www.apple.com>

© 1992-1999 Apple Computer, Inc., All Rights Reserved.



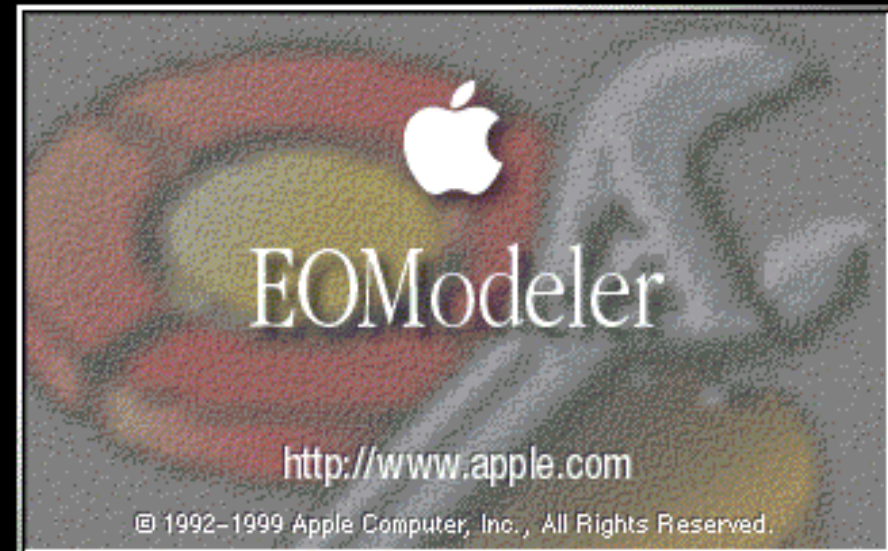
The WebObjects Builder software box features a white Apple logo at the top center. Below the logo, the text "WebObjects Builder" is displayed in a large, white, serif font. Underneath the product name, the URL "http://www.apple.com" is written in a smaller white font. At the bottom of the box, the copyright notice "© 1992-1999 Apple Computer, Inc., All Rights Reserved." is printed in a small white font. The background of the box is a dark, textured grey with a faint, abstract pattern.

Apple

WebObjects Builder

<http://www.apple.com>

© 1992-1999 Apple Computer, Inc., All Rights Reserved.



The EOModeler software box features a white Apple logo at the top center. Below the logo, the text "EOModeler" is displayed in a large, white, serif font. Underneath the product name, the URL "http://www.apple.com" is written in a smaller white font. At the bottom of the box, the copyright notice "© 1992-1999 Apple Computer, Inc., All Rights Reserved." is printed in a small white font. The background of the box is a dark, textured grey with a faint, abstract pattern.

Apple

EOModeler

<http://www.apple.com>

© 1992-1999 Apple Computer, Inc., All Rights Reserved.



**403—Designing Dynamic
HTML Components**



You'll Create WO Components

- Main.wo/

- Main.html
- Main.wod

```
<HTML>
```

```
This product
```

```
<WEBOBJECT NAME=Image4>
```

```
costs
```

```
<WEBOBJECT NAME=String1>
```

```
<HR>
```

```
<WEBOBJECT NAME=Link1>
```

```
Click here
```

```
</WEBOBJECT> to order.
```

```
Image4: WOImage {
```

```
src = "pic.gif"; }
```

```
String1: WOString {
```

```
value = product.cost
```

```
numberFormat =
```

```
"$#,###"; }
```

```
Link1: WOHyperlink {
```

```
action = placeOrder; }
```

- Main.java

```
public class Order extends WOComponent {  
    private ShoppingItem product;  
    public WOComponent placeOrder( ) {  
        session().addToShoppingCart(product);  
        return pageWithName("Checkout");  
    }  
}
```



You'll Create Lots of WO Components

- One for each "page"
- Subcomponents for headers, footers, navigation bars
- Reusable components for common UI elements
- JavaScript client-side tricks
- Components that contain other components



So How Do You Make Those Components ?

- They're just text files
- Use whatever you like
 - emacs
 - vi
 - stickies
 - cat



Oh, Come On

- OK. You can use any WYSIWIG HTML editor
- DreamWeaver
- GoLive
- Or even WebObjects Builder



Inspector

WOHyperlink Binding Inspector

Attribute	Binding
action	placeOrder
actionClass	
directActionName	
disabled	
fragmentIdentifier	
href	

Dynamic Inspector

Hyperlink1

Make Static

HTML

Main.wd - ~/Projects/Ex1

This product costs

[Click here](#)

- action
- actionClass
- directActionName
- disabled
- fragmentIdentifier
- href
- otherTagString
- pageName
- queryDictionary
- string
- target
- Connect to new binding...

<BODY> <WOHy

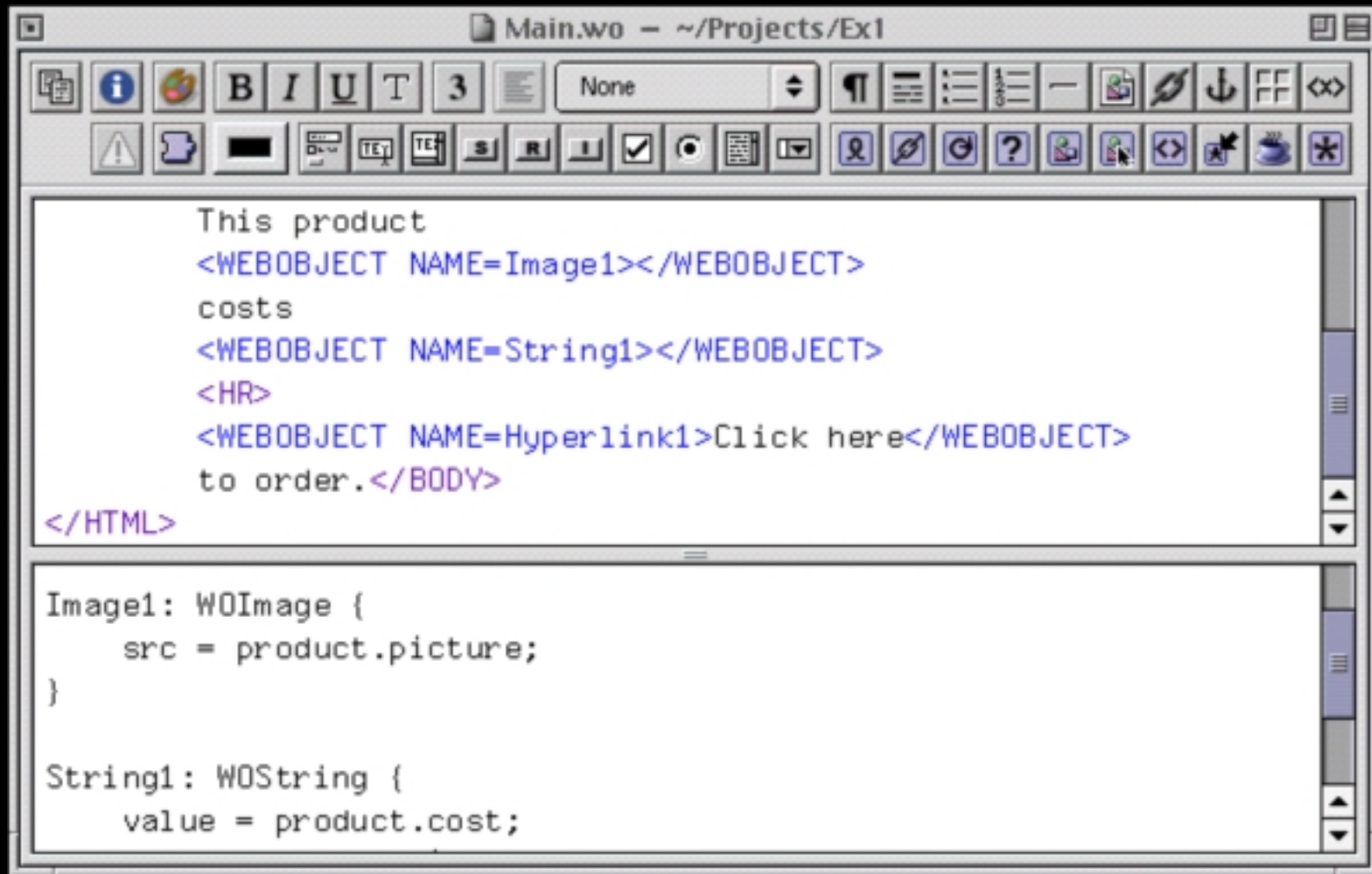
application
session
product

placeOrder

Edit Source

Objects and Actions

Use WOB in Raw Mode, if You Like



The screenshot shows a window titled "Main.wo - ~/Projects/Ex1" with a rich toolbar. The main text area contains the following HTML code:

```
This product
<WEBOBJECT NAME=Image1></WEBOBJECT>
costs
<WEBOBJECT NAME=String1></WEBOBJECT>
<HR>
<WEBOBJECT NAME=Hyperlink1>Click here</WEBOBJECT>
to order.</BODY>
</HTML>
```

Below the HTML code, two WOB objects are defined:

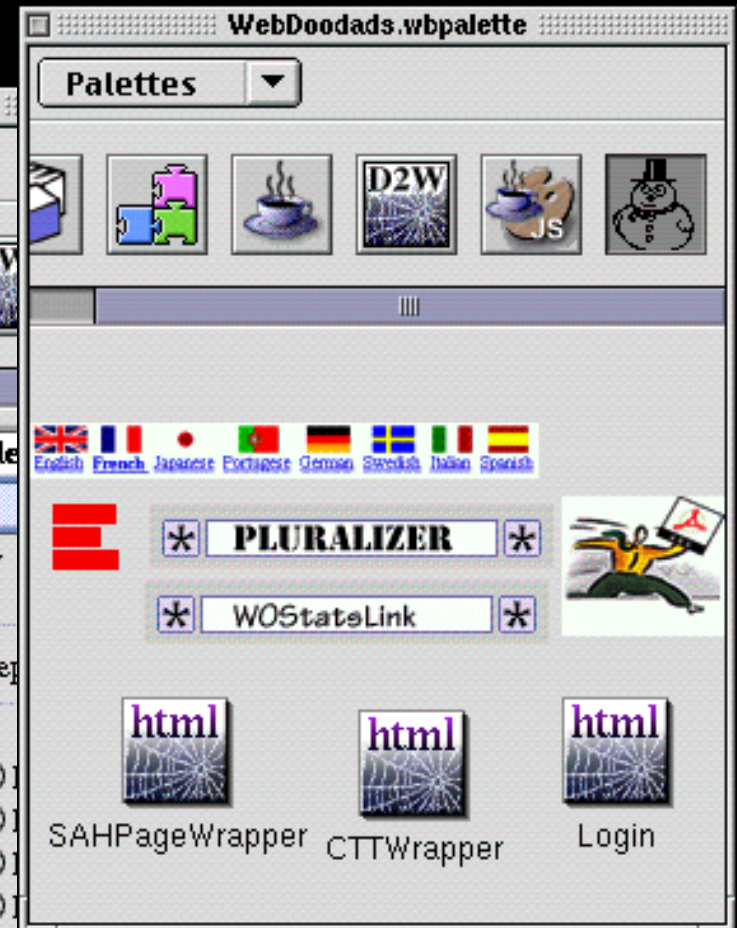
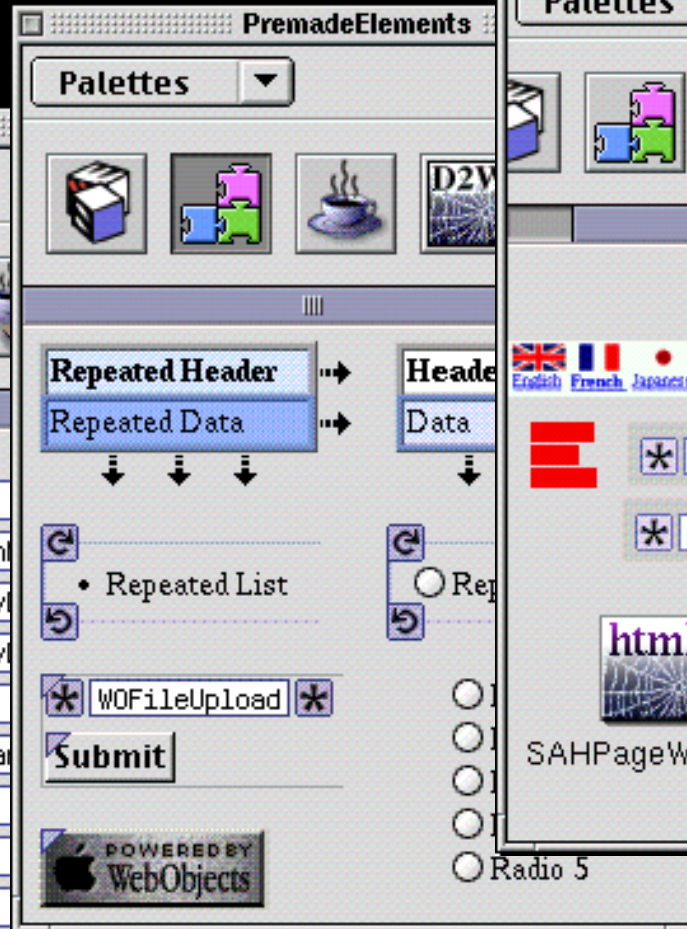
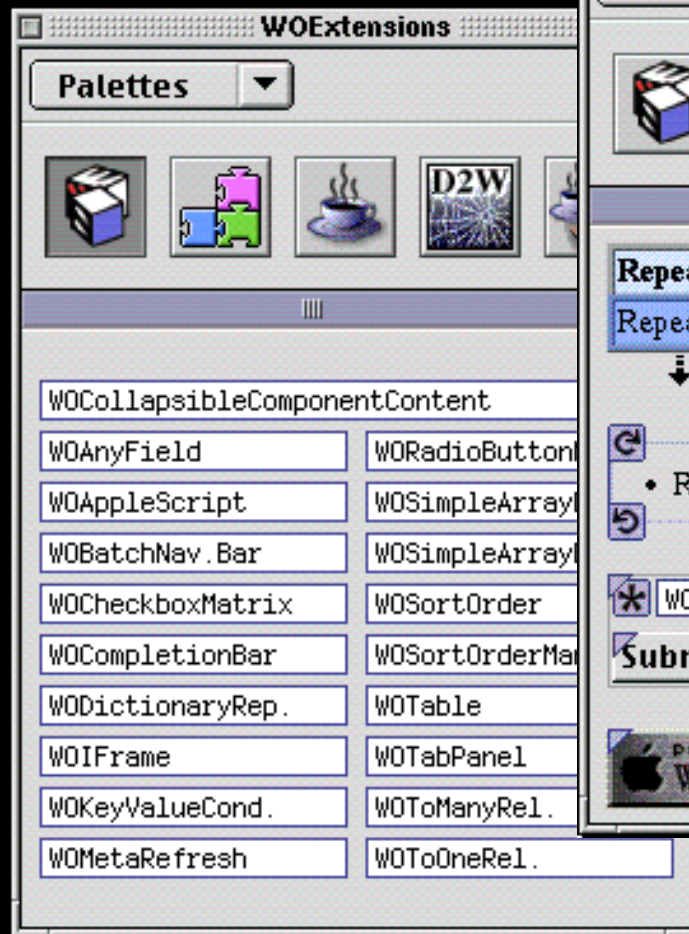
```
Image1: WOImage {
    src = product.picture;
}

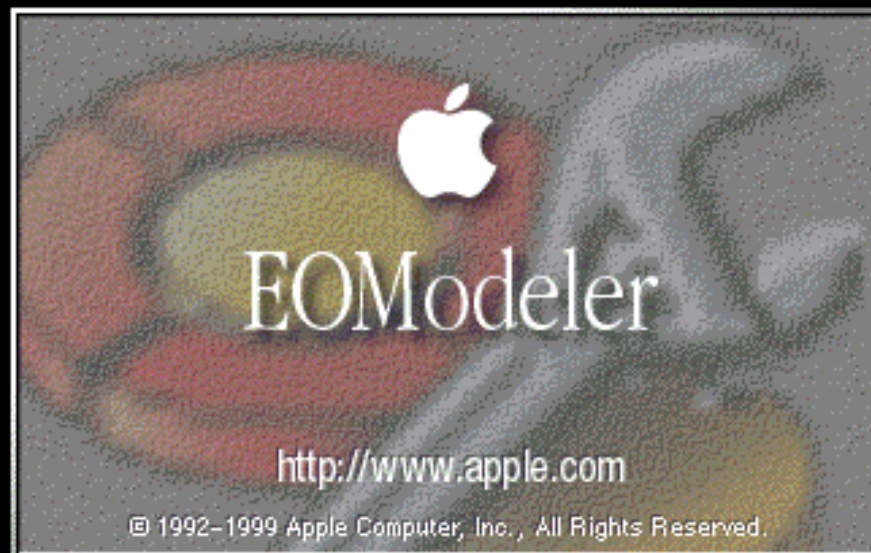
String1: WOString {
    value = product.cost;
```



Use Palettes to Organize Your Stuff

416—Designing Reusable Components





**403—Persistent Object
Modeling with EOModeler**



EOModeler and EOF

- Define a recipe for constructing objects from databases

Vehicle objects == the MDL table
Vehicle.basePrice() == the PRMIN column

Name	Column	External Type
basePrice	PRMIN	NUMBER
blurb	ADTAG	VARCHAR2
carid	CARID	NUMBER
categoryId	CATID	NUMBER
loadedPrice	PRMAX	NUMBER
manuid	MANUID	NUMBER
picture	IMG_JPG_DATA	LONG RAW
vehicleName	MDLNAME	VARCHAR2

Name	Destination	Source Att
category	Category	categoryId
maker	Maker	manuid

Entity Inspector

Name: **Vehicle**

Table Name: MDL

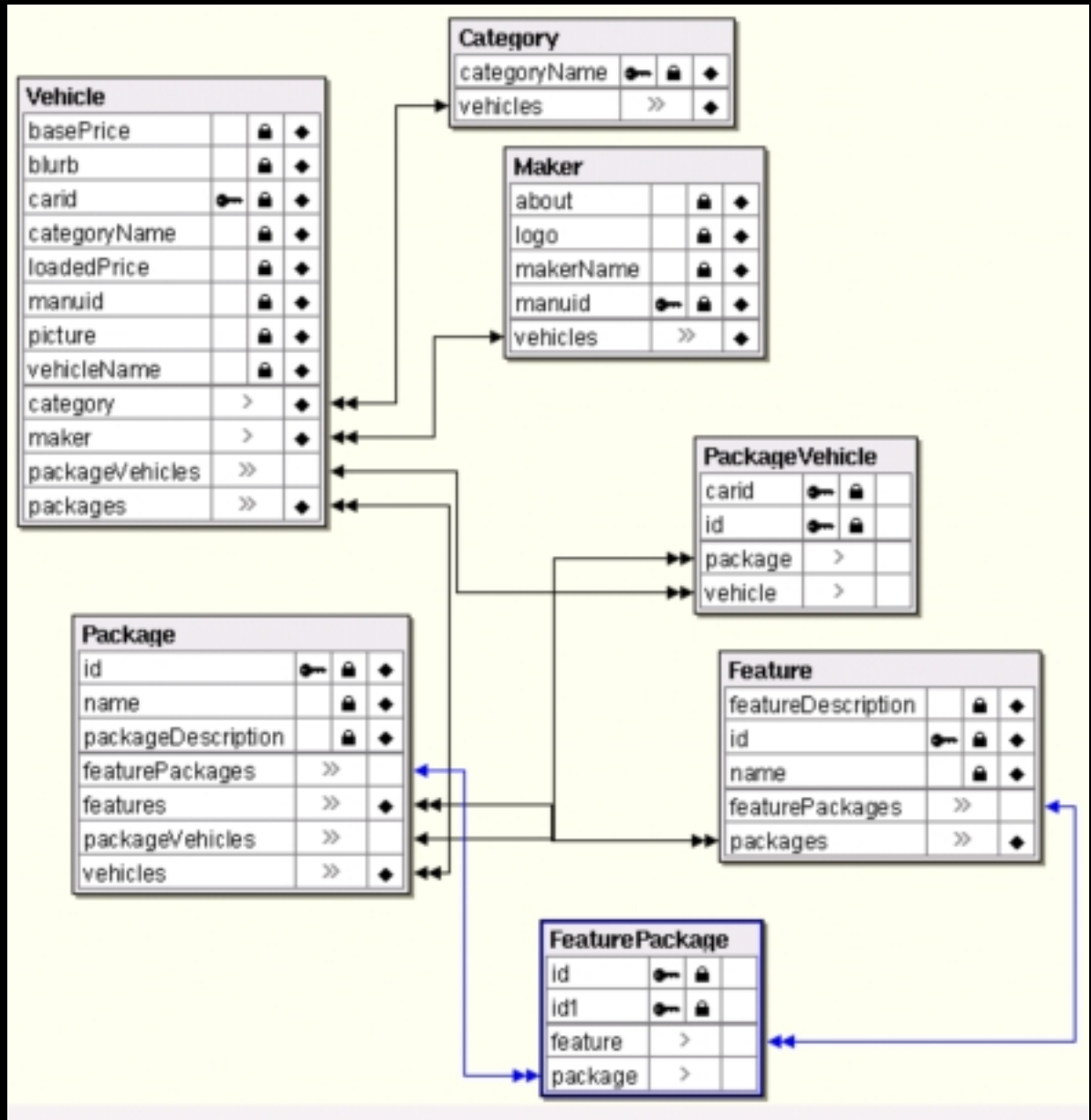
Class: EOGenericRecord

Properties

- basePrice
- blurb
- carid
- categoryId
- loadedPrice

EOModeler

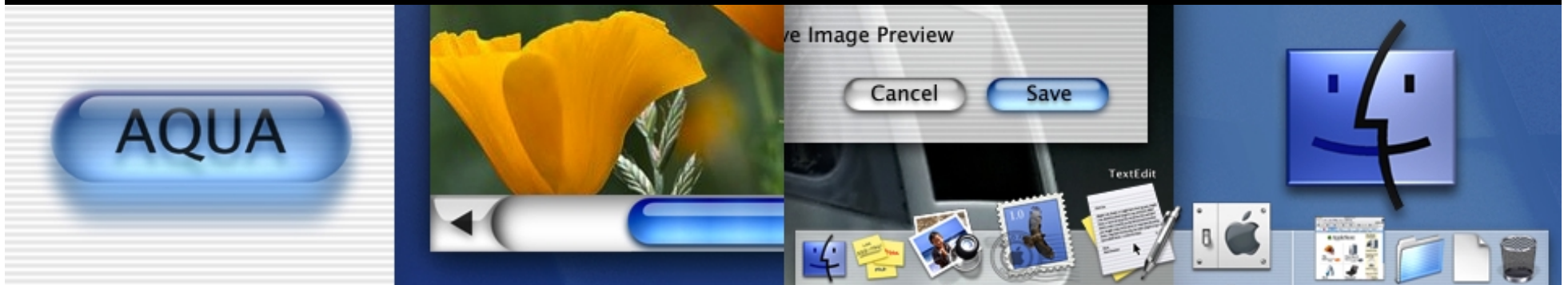
Model complex relationships between objects too.
vehicle.maker.
makerName
vehicle.packages





Session 401

A Really Brief EOModeler Demo





ProjectBuilder

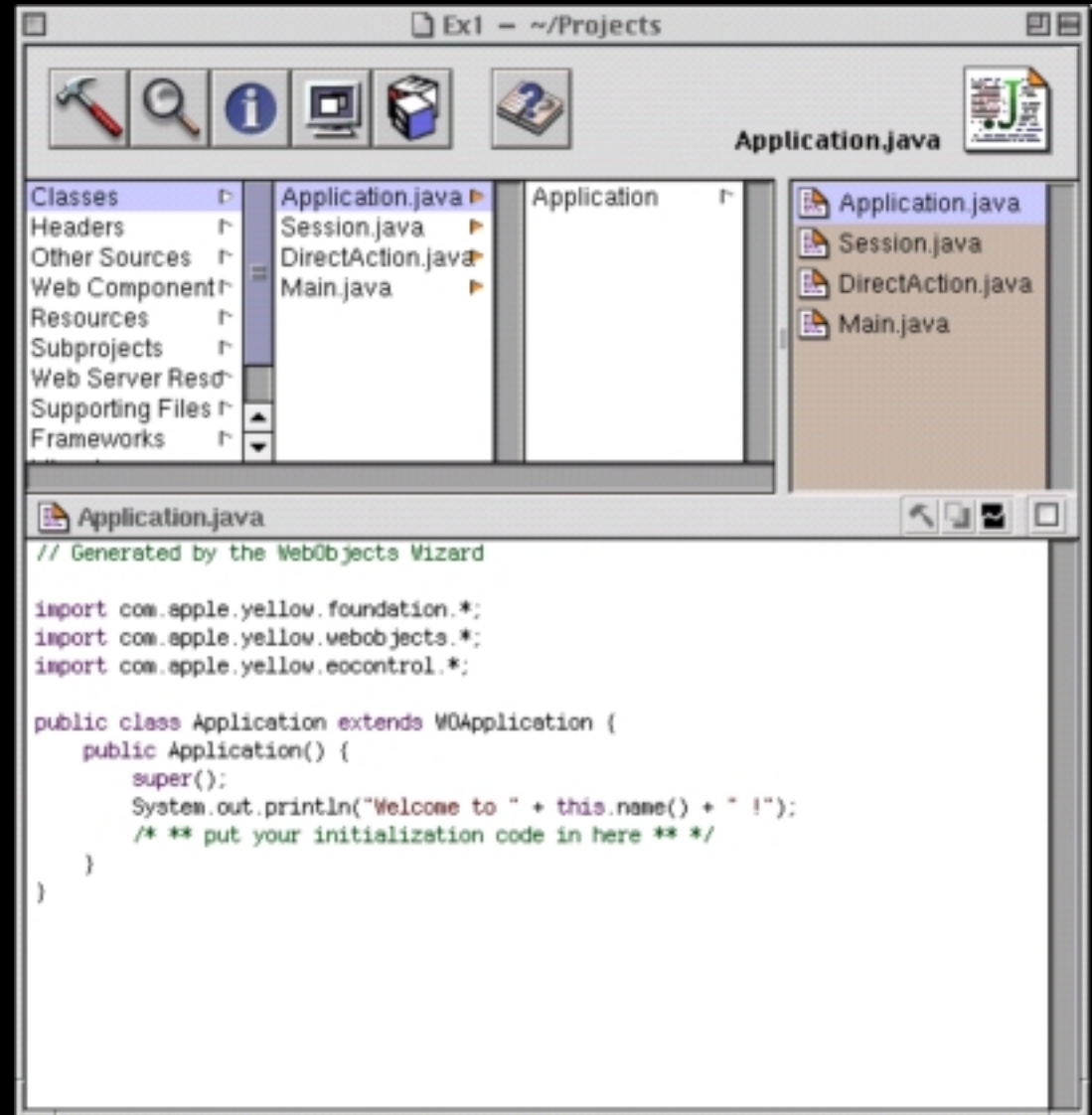
<http://www.apple.com>

© 1992-1999 Apple Computer, Inc., All Rights Reserved.



Project Builder WO

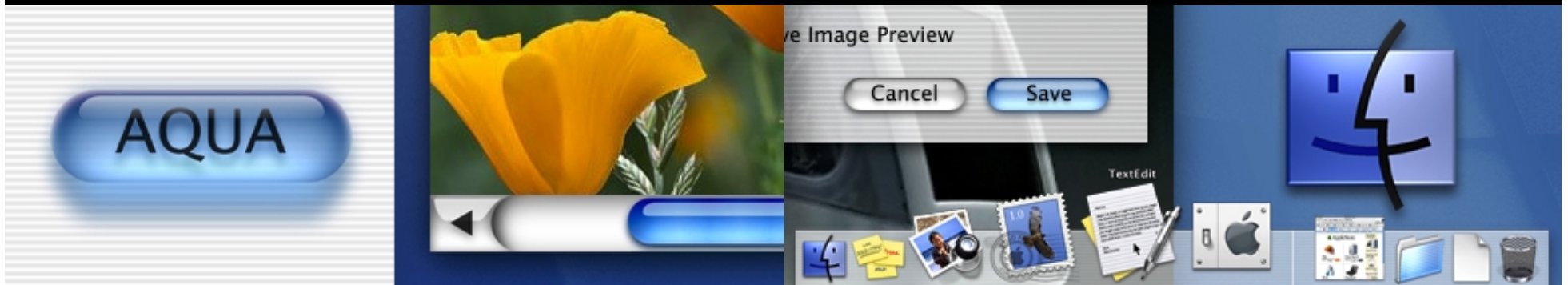
- Coordinates resources
- Code editing and debugging
- Localization





Session 401

Delivering Other User Interfaces



Nobody Said You Had to Use HTML

Other UIs

- Deliver your business objects through other front-end technologies
- Java Client
- PDF
- XML
- WAP

**411—Integrating
Client-Side
Technologies**



Java Client

406—Rapid Development
with Direct to Java Client

- A more “traditional” client-server application
- Pure Java application on the client
- WO app on the server
- Enterprise Objects migrate between the two



PDF—High-Quality Printable Documents

- www.reportmill.com
- Object-based PDF generation and reporting
- Subtotaling, grouping, sorting, slicing, dicing



XML

- Turn one or more objects into an XML string with only a line or two of code

```
WOXMLCoder aCoder = WOXMLCoder.coder();  
aString = aCoder.encodeRootObjectForKey( currentCar, "Vehicle" );
```

```
<Vehicle type="com.apple.yellow.eocontrol.EOGenericRecord" >  
  <vehicleName type="java.lang.String">Hummer</vehicleName>  
  <basePrice type="java.math.BigDecimal">71000</basePrice>  
  ...  
</Vehicle>
```



WAP—Wireless Applications

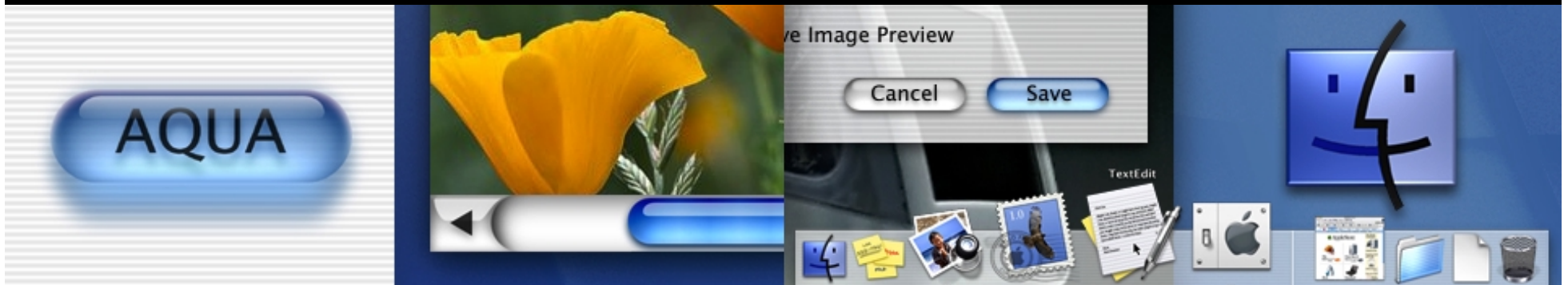
- Special objects that emit WML instead of HTML
- www.wapobjects.com





Session 401

Superduper Assistants



Got a Model? You're 95% Done

Direct to Web

**405—Rapid Development
with Direct to Web**

Your EOModel

+ Direct to Web

A complete, full featured,
and good-looking web app



Direct to Java Client

406—Rapid Development
with Direct to Java Client

Your EOModel

+ Direct to Java Client

A complete, full-featured,
and good-looking
Java-based application

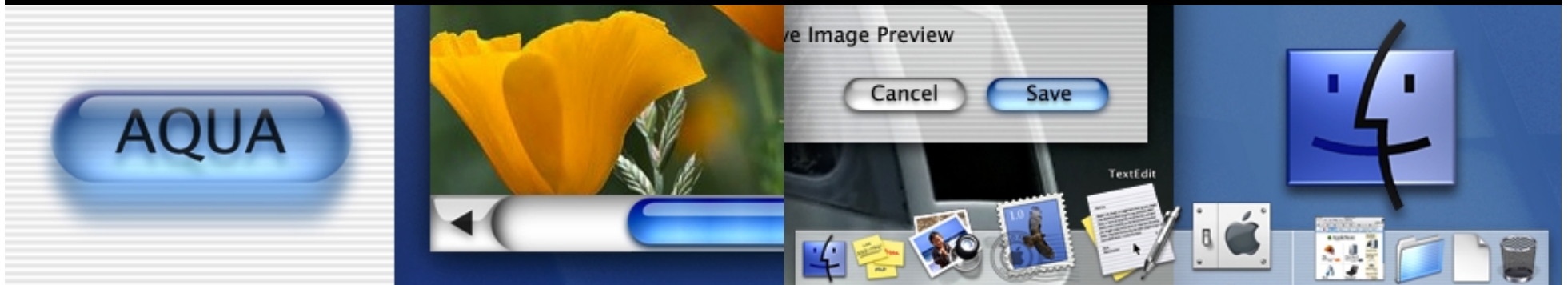




Demo

Direct to Web

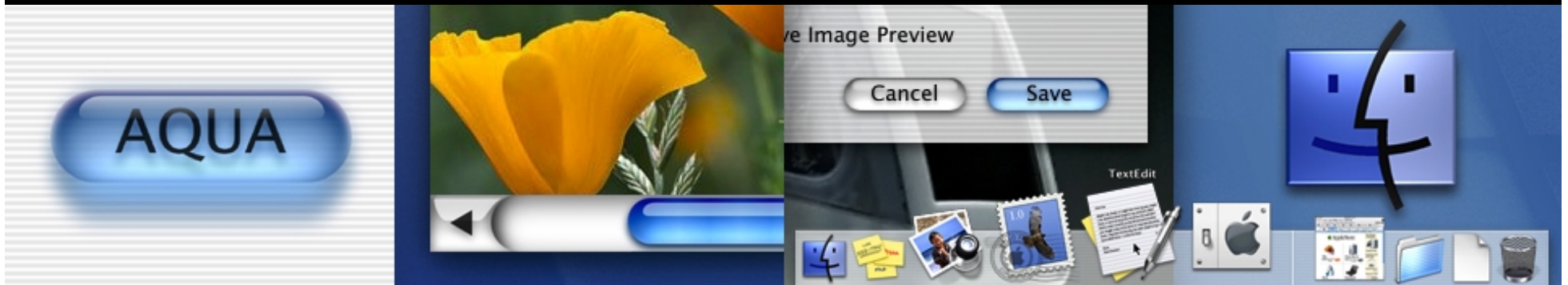
Direct to Java Client





Session 401

Deploying Your Application



How Do You Launch Your App and Keep It Running?

Monitor

- A WebObjects application that manages other WebObjects applications
- Tell it what WO Application Server machines you have, and what applications you'd like to run, and how many instances of each
- A service called wotaskd launches them and keeps them running








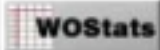
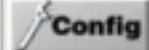


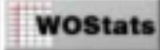
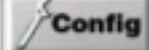


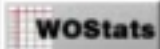
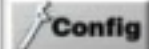


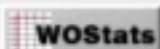
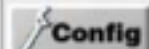


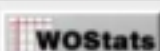
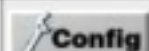
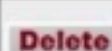
Monitor

[Applications](#) [Hosts](#) [Configure](#)



AutoBySteve

 Refresh  Add Instance  Start All  Stop All

Host - Port	Status	Schedule	Auto Recover	Refuse New Sessions	Statistics						WOStats	Configure	
					Transactions	Active Sessions	Average Transaction	Average Idle Period	Deaths	Exceptions			
1 yyz 2002		OFF	OFF	OFF	-	-	-	-	0	0	 WOStats	 Config	 Delete
2 yyz 2004		OFF	OFF	OFF	-	-	-	-	1	0	 WOStats	 Config	 Delete
3 yyz 2005		OFF	ON	OFF	1	0	0.000	16.09	0	0	 WOStats	 Config	 Delete
4 yyz 2006		OFF	ON	OFF	1	0	0.000	16.38	0	0	 WOStats	 Config	 Delete
5 yyz 2007		OFF	ON	OFF	1	0	0.000	16.56	0	0	 WOStats	 Config	 Delete
Totals					3	0	0.000	16.347					

	Per Second	Per Minute
Transaction Rate	0.11	6.5

[Exception and Death History](#)



More on Big Application Deployment

**405—Deploying
Applications**

**414—Performance
Metrics**

410—Optimization

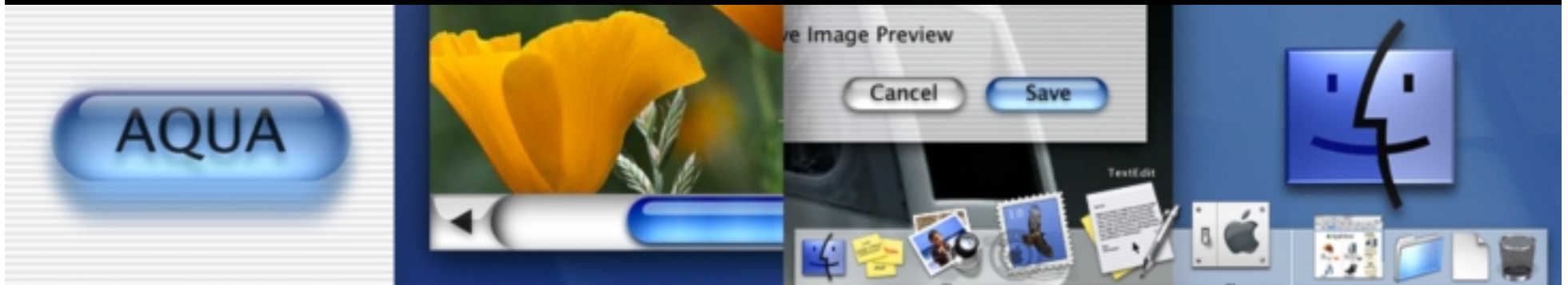
**418—Building
Large-Scale Applications**





Session 401

Learning More



WOInfoCenter

- Documentation
- Books
- Examples
- Tutorials
- All browseable and searchable via a WebObjects application



- Main Page
- Search
- Apple

Info Center



Your search yielded 271 hits.

◀ Page 1 of 19 ▶

EOEditingContext Search

RELEVANCE DOCUMENT

- 100% [What are EODisplayGroups and EOEditingContexts?](#) [eoeditingcontext]
in [Books/WebObjects/Java Client Tutorial](#)
- 94% [Invalidating Objects](#) [eoeditingcontext]
in [Books/WebObjects/Topics in WebObjects Programming](#)
- 91% [Managing State](#) [eoeditingcontext]
in [Books/WebObjects/WebObjects Developer's Guide](#)
- 87% [Using Nested Editing Contexts](#) [eoeditingcontext]
in [Books/WebObjects/Topics in WebObjects Programming](#)
- 86% [EOEditingContext Delegate](#) [eoeditingcontext]
in [Reference/Enterprise Objects Framework/EOControl Reference \(Objc\)](#)
- 84% [EOEditingContext Editor](#) [eoeditingcontext]
in [Reference/Enterprise Objects Framework/EOControl Reference \(Java\)](#)
- 83% [Application Configurations](#) [eoeditingcontext]
in [Books/Enterprise Objects Framework/Enterprise Objects Framework Developer's Guide](#)
- 82% [Application Configurations](#) [eoeditingcontext]
in [Books/Enterprise Objects Framework/Enterprise Objects Framework Developer's Guide](#)
- 80% [EOEditingContext Delegate](#) [eoeditingcontext]
in [Reference/Enterprise Objects Framework/EOControl Reference \(Java\)](#)
- [eoeditingcontext]



WebObjects Info Center - Microsoft Internet Explorer

File Edit View Favorites Tools Help

● Main Page
● Search
● Apple

Info Center

[Click here for help](#) Search

RESOURCES

- ▶ [Books](#)
- ▶ [Reference](#)
- ▼ [Examples](#)
 - ▼ [WebObjects](#)
 - ▼ [Java](#)
 - BugTracker
 - [See the README](#) [Open the Project](#) [Open in Workspace](#) [Launch the application](#)
 - This simple bug tracking application demonstrates how to efficiently use DirectToWeb to quickly build database applications using embedded components (D2WList...), named configurations, D2WCustomComponent...
 - Element Tour
 - [See the README](#) [Open the Project](#) [Open in Workspace](#) [Launch the application](#)
 - WebObjects application which exercises virtually all dynamic elements in the system.
 - FDF 1040EZ
 - [See the README](#) [Open the Project](#) [Open in Workspace](#)
 - Illustrates how to use the FDF Framework to integrate FDF in your WebObjects applications. Due to a limitation in Acrobat, this application will not behave correctly in direct connect mode; you have to go



WO Training Classes

- Programming WebObjects I
- Programming WebObjects II
- WebObjects Deployment

- www.apple.com/webobjects/schedule.html



Roadmap

402 WebObjects Builder:

Designing dynamic HTML components

Room J2
Tues., 1:30 p.m.

403 Persistent Object Modeling with Enterprise Objects Framework

Room J2
Tues., 3:00 p.m.

404 WO: Deploying Applications

Room J2
Tues., 4:30 p.m.

WO: Rapid Development

405 Direct To Web

Room J2
Wed., 9:00 a.m.

406 Direct to Java Client

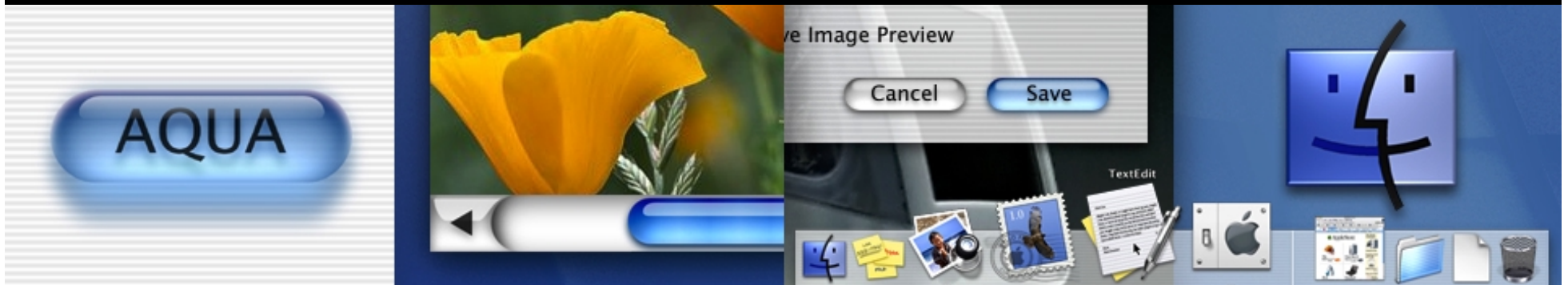
Wed., 10:30 a.m.





Session 401

Q&A





WWDC

Worldwide Developers Conference 2000



Think different.