Worldwide

Developers

Conference

WebObjects: Frameworks

Allen Denison

Apple Enterprise Software

Agenda

- WebObjects as a Framework
- Building a Web Application
- Framework Specifics
- How it Works
- Summary
- Q&A

A Definition of WebObjects

WebObjects is a collection of frameworks that facilitate the rapid development of sophisticated Web-based applications

What Is a Framework?

A Framework is a collection of classes that work together to solve specific problems in a particular domain

Why Frameworks?

Why Frameworks?

- Apple concentrates on delivering a strong foundation

Why Frameworks?

- **Apple concentrates on delivering a strong foundation**
- **Partners can build on that foundation**
  - Tools (goLive's CyberStudio)
  - New Frameworks (Dover Pacific)

Why Frameworks?

- **Apple concentrates on delivering a strong foundation**
- **Partners can build on that foundation**
  - Tools (goLive's CyberStudio)
  - New Frameworks (Dover Pacific)
- **Run on multiple platforms**

Why Frameworks?

- **Apple concentrates on delivering a strong foundation**
- **Partners can build on that foundation**
  - Tools (goLive's CyberStudio)
  - New Frameworks (Dover Pacific)
- **Run on multiple platforms**
- **Shared between Web and traditional apps**

Why Frameworks?

- **Apple concentrates on delivering a strong foundation**
- **Partners can build on that foundation**
  - Tools (goLive's CyberStudio)
  - New Frameworks (Dover Pacific)
- **Run on multiple platforms**
- **Shared between Web and traditional apps**
- **Developers don't have to write as much code—assemble and deploy**

Apple Frameworks

Apple Frameworks

- **Foundation**
  - Data type objects, separation from OS, remote object support

Apple Frameworks

- **Foundation**
  - Data type objects, separation from OS, remote object support
- **Application Kit**
  - Traditional application UI objects

Apple Frameworks

- **Foundation**
  - Data type objects, separation from OS, remote object support
- **Application Kit**
  - Traditional application UI objects
- **Enterprise Objects**
  - Object persistence (database connectivity)

Apple Frameworks

- **Foundation**
  - Data type objects, separation from OS, remote object support
- **Application Kit**
  - Traditional application UI objects
- **Enterprise Objects**
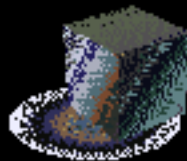 - Object persistence (database connectivity)
- **WebObjects**
  - Web application specific objects

Building a Web Application

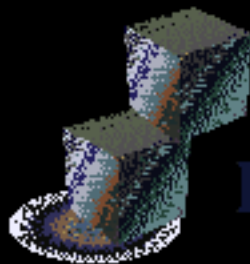# Building a Web Application

Learn HTTP/HTML

Building a Web Application

Understand Server API/Headers

Learn HTTP/HTML

Building a Web Application

Learn SQL/Build Queries

Understand Server API/Headers

Learn HTTP/HTML

Building a Web Application

Generate HTML

Learn SQL/Build Queries

Understand Server API/Headers

Learn HTTP/HTML

Building a Web Application

Manage User State

Generate HTML

Learn SQL/Build Queries

Understand Server API/Headers

Learn HTTP/HTML

Building a Web Application

Localize

Manage User State

Generate HTML

Learn SQL/Build Queries

Understand Server API/Headers

Learn HTTP/HTML

Building a Web Application

Write Application

Localize

Manage User State

Generate HTML

Learn SQL/Build Queries
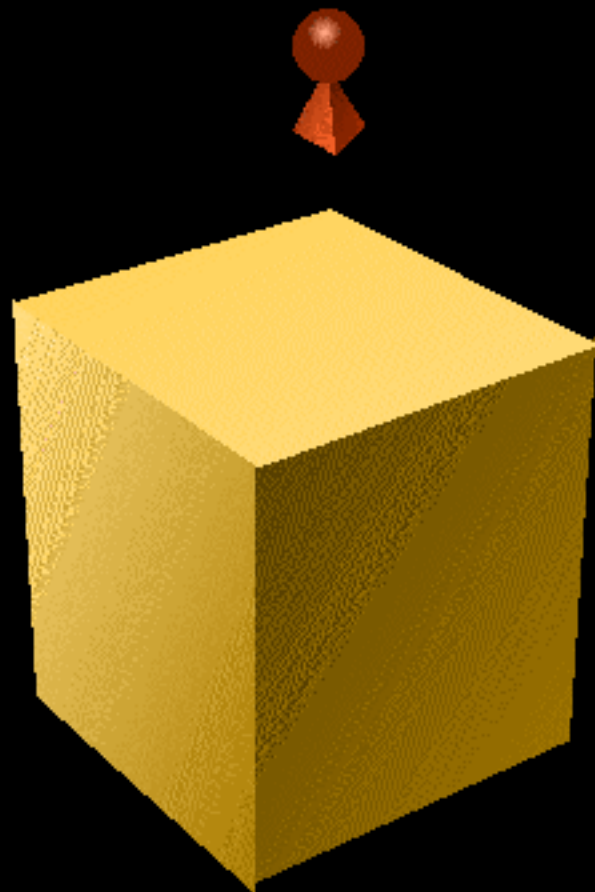
Understand Server API/Headers

Learn HTTP/HTML

Building a Web Application

Deploy

Write Application

Localize

Manage User State

Generate HTML

Learn SQL/Build Queries

Understand Server API/Headers
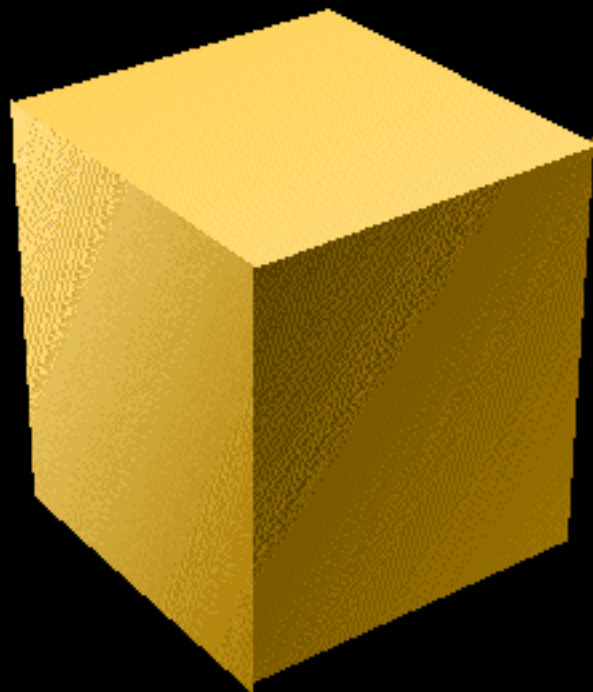
Learn HTTP/HTML
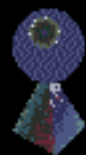
Building a WebObjects App

Building a WebObjects App
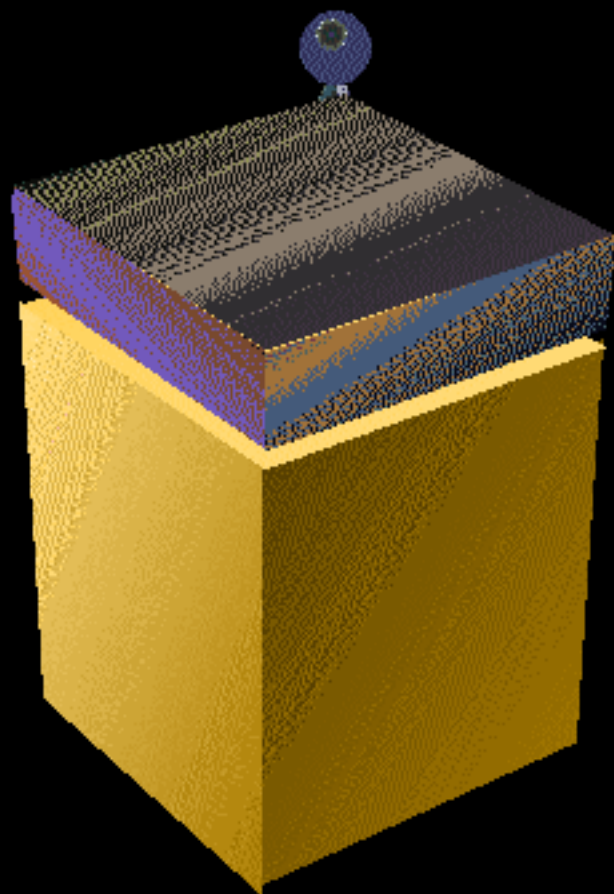
WebObjects Framework

Building a WebObjects App

WebObjects Framework
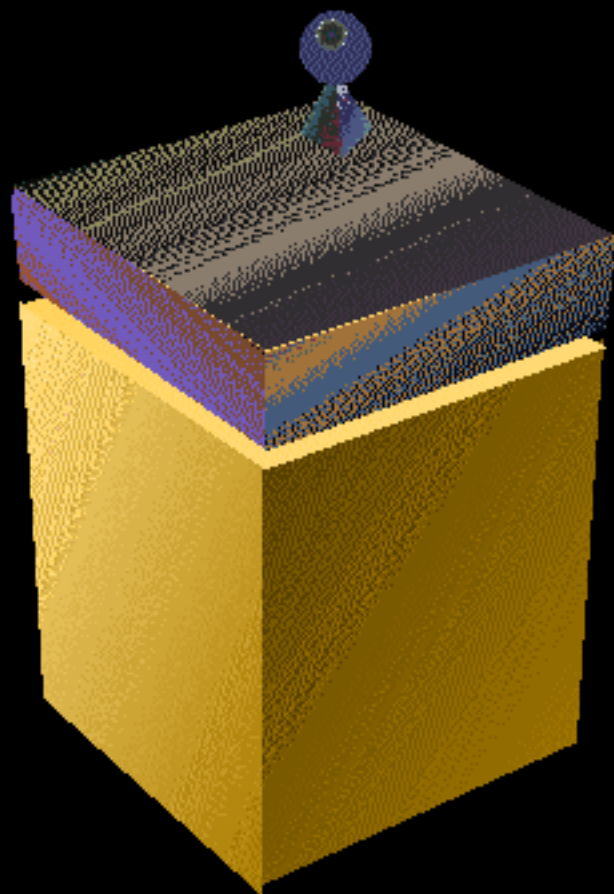
Building a WebObjects App

Write App

WebObjects Framework

Building a WebObjects App

Write App

WebObjects Framework

Framework Specifics

Learn HTTP/HTML

Framework Specifics

Learn HTTP/HTML

- **HTTP Request Class**
  - Get browser type
  - Get incoming data's MIME content type
  - Get incoming data
  - Get HTML input form values by name
  - Get browser's cookies
  - Localized

Framework Specifics

Learn HTTP/HTML

- HTTP Response Class
  - Set response's MIME content type
  - Append HTML to response
  - Append raw data to response
  - Set other HTTP response headers
    - Authentication challenges
    - URL redirection, etc.
  - Set content encoding for localization

Framework Specifics

Learn HTTP/HTML

- **HTML Classes (Dynamic Elements)**
  - Template is parsed into tree of objects
  - Each HTML tag has its own representation: tag name, attributes, etc.
  - Objects know how to render themselves in HTML

Framework Specifics

Understand Server API/Headers

Learn HTTP/HTML

Framework Specifics

Understand Server API/Headers

- HTTP Server Adaptor
  - Converts HTTP requests into HTTP Request objects
  - Converts HTTP Response objects into Server API specific format
  - Source code is shipped with releases
  - Supports CGI, NSAPI, ISAPI, Web*, etc.
  - Load distribution and load balancing

Framework Specifics

Learn SQL/Build Queries

Understand Server API/Headers

Learn HTTP/HTML

Framework Specifics

Learn SQL/Build Queries

- **Database Connectivity Classes (EOF)**
  - Database independent
  - Supports Oracle, Sybase, Informix, and ODBC
  - Multiple databases can be used to populate the attributes of an object
  - Doesn't tie SQL code directly to UI
  - Same EOF objects used in both OPENSTEP and WebObjects

Framework Specifics

Generate HTML

Learn SQL/Build Queries

Understand Server API/Headers

Learn HTTP/HTML

Framework Specifics

Generate HTML

- **Dynamic Elements**
  - Attributes of HTML elements determined by variables and method calls at runtime
    - e.g., populate an HTML table as a response to a query
  - Display HTML based on conditional logic
  - Repetition construct

Framework Specifics

- Manage User State
- Generate HTML
- Learn SQL/Build Queries
- Understand Server API/Headers
- Learn HTTP/HTML

Framework Specifics

Manage User State

- **Session State Management Classes**
  - The Session Class
    - Get/set session variables
    - Session knows how to serialize/ deserialize itself
  - Persistence: The SessionStore Class
    - Saves and retrieves Session objects (for application and individual pages)
  - Subclass (for flatfile or database storage)
  - Session timeout services

Framework Specifics

Localize

Manage User State

Generate HTML

Learn SQL/Build Queries

Understand Server API/Headers

Learn HTTP/HTML

Framework Specifics

Localize

- **Localization Support**
  - Hash table maps English keywords into localized values
  - Localized icons, images and HTML templates
  - User's language preferences are stored in Session object
  - API is localization aware

Framework Specifics

Write Application

Localize

Manage User State

Generate HTML

Learn SQL/Build Queries

Understand Server API/Headers

Learn HTTP/HTML

Framework Specifics

Write Application

- **Application Development Support**
  - Server side scripting
    - Easy to learn
    - Rapid prototyping and application development
    - No compilation required
  - Server side Java support
    - Write entire application in Java
  - Application Class

Framework Specifics

Deploy

Write Application

Localize

Manage User State

Generate HTML

Learn SQL/Build Queries

Understand Server API/Headers

Learn HTTP/HTML

Framework Specifics

Deploy

- **Deployment Support**
  - Load balancing and distribution
  - Deploy multiple instances of application on the same machine
  - Scale up by deploying on multiple machines
  - Deployment is configurable
    - Machines/instances can be added at run time without shutting down

How it Works

Static Web Site

How it Works

Request

Static Web Site

How it Works

Response
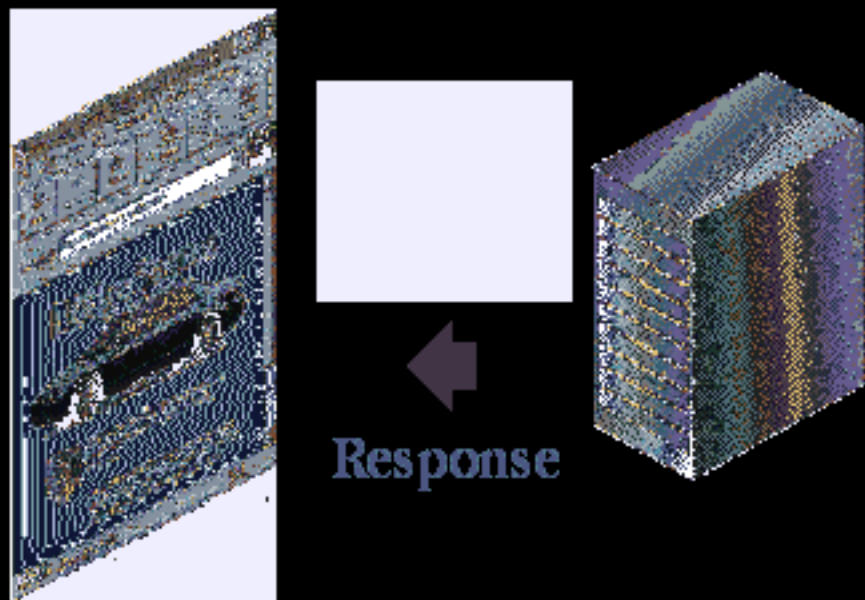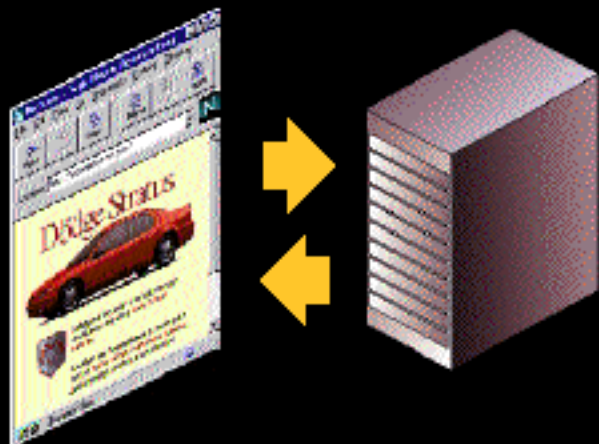
Static Web Site

How it Works

Dynamic WebObjects Site

How it Works

Server
Adaptor

Dynamic WebObjects Site

How it Works

Server
Adaptor

WebObjects
Application
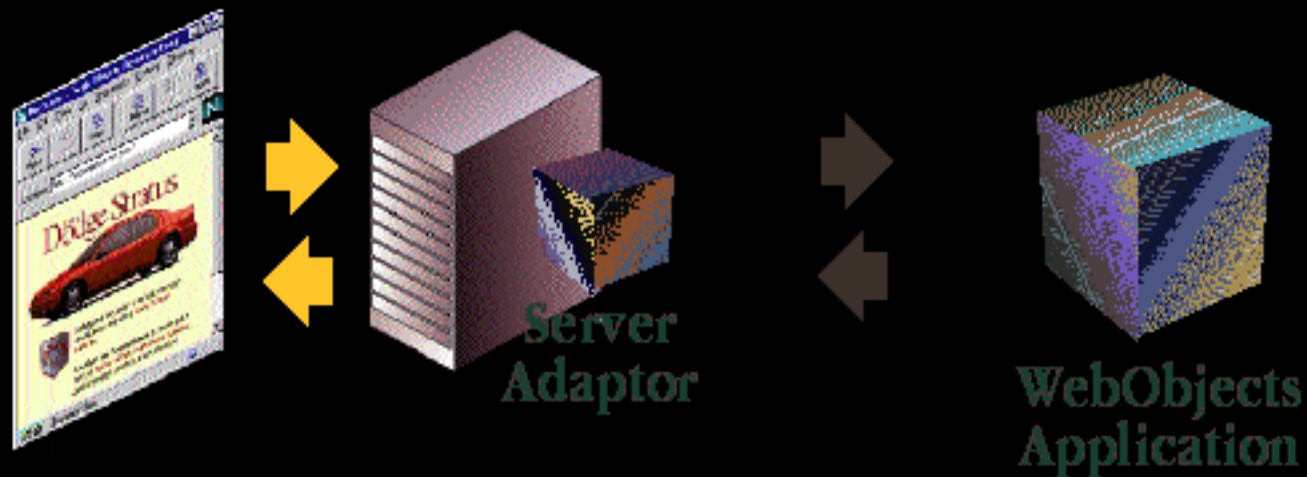
Dynamic WebObjects Site
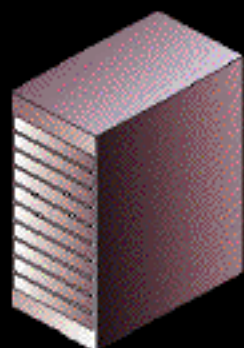
How it Works

Passing the Request
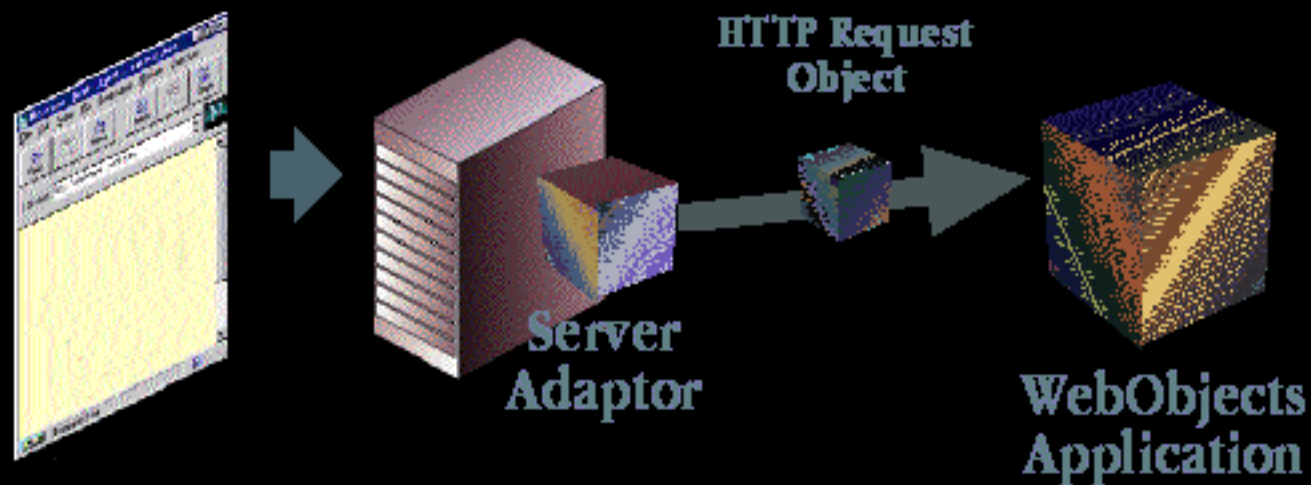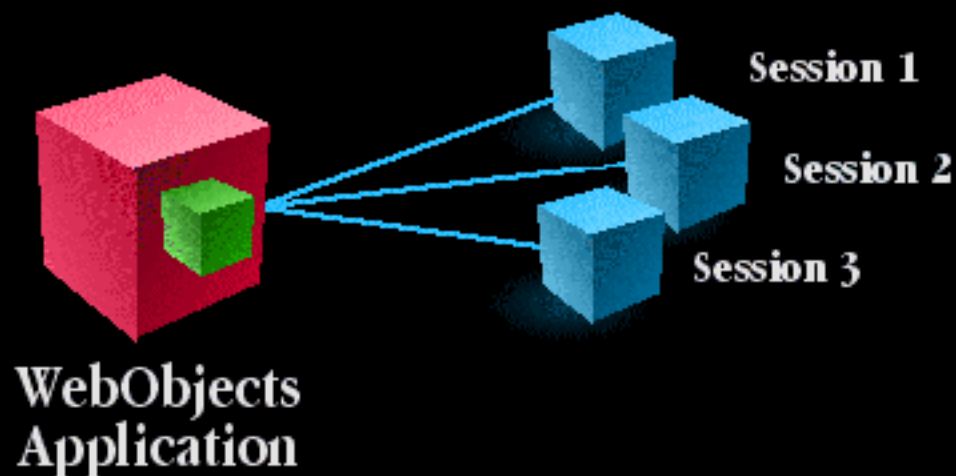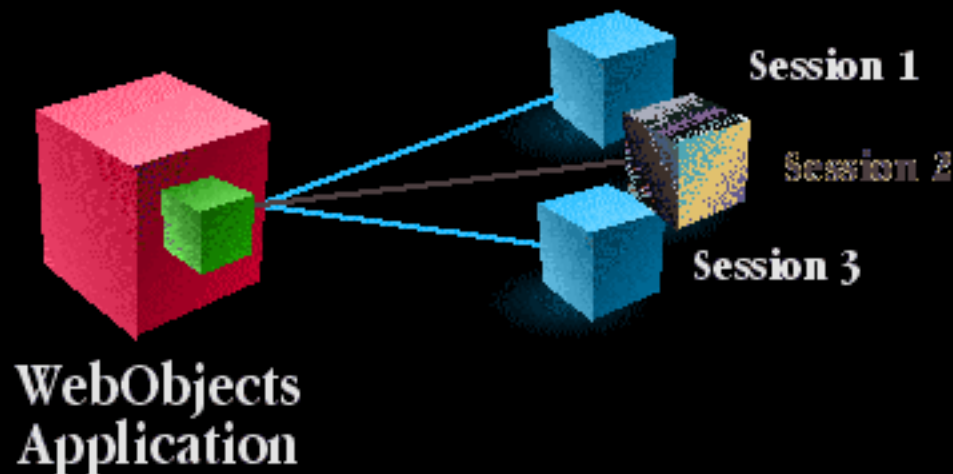
How it Works

Server
Adaptor

Passing the Request

How it Works

HTTP Request
Object

Server
Adaptor

Passing the Request

How it Works

Server
Adaptor

HTTP Request
Object

WebObjects
Application

Passing the Request

How it Works

WebObjects
Application

Session 1

Session 2

Session 3

Determining the Session

How it Works

WebObjects
Application

Session 1

Session 2

Session 3

Determining the Session

How it Works

Session 2

WebObjects
Application

Determining the Page

How it Works

Session 2
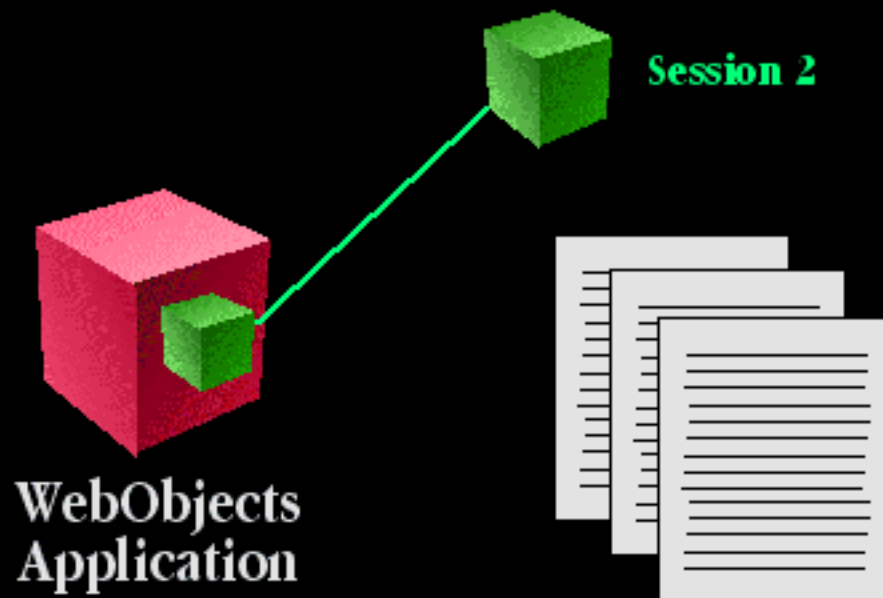
WebObjects
Application
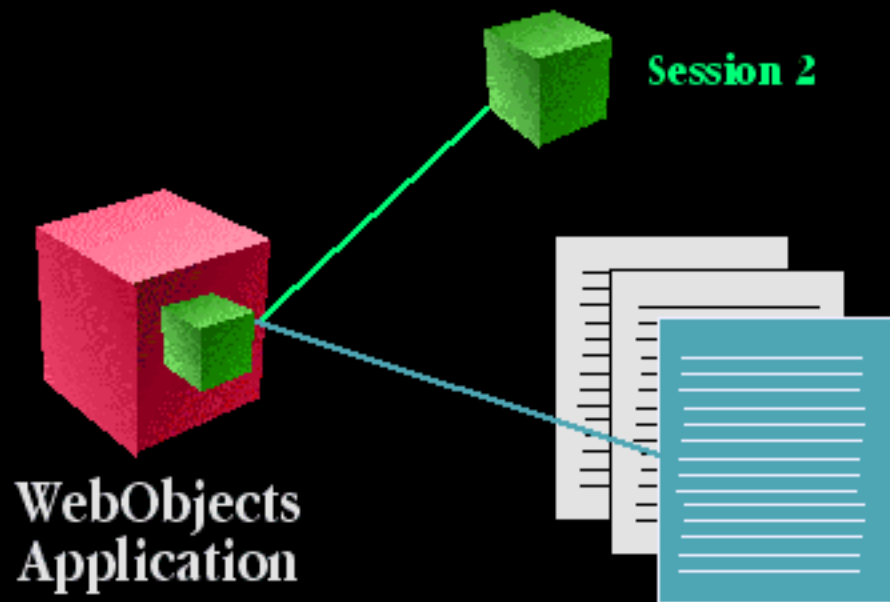
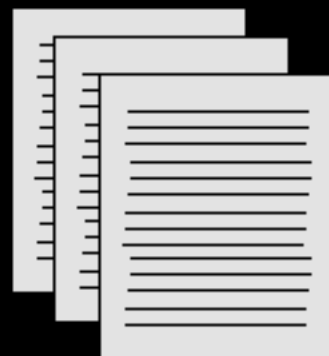## Determining the Page

How it Works

- Has three parts
  - HTML Template
  - Server side logic
  - Declarations (bindings)
- Receives and processes HTTP Request objects
- Generates HTTP Response objects

A WebObjects Page

How it Works

HTML Template	Declarations	Logic

```
<HTML>

<BODY>

<FORM>

<WEBOBJECT NAME=A>

<WEBOBJECT NAME=B>

</FORM>

</BODY>

</HTML>
```

```
A: TextField {
       value=data;
   }


B: SubmitButton {
       action=setData;
   }
```

```
id data;

function setData {
    object=doIt();
    return object;
}
```
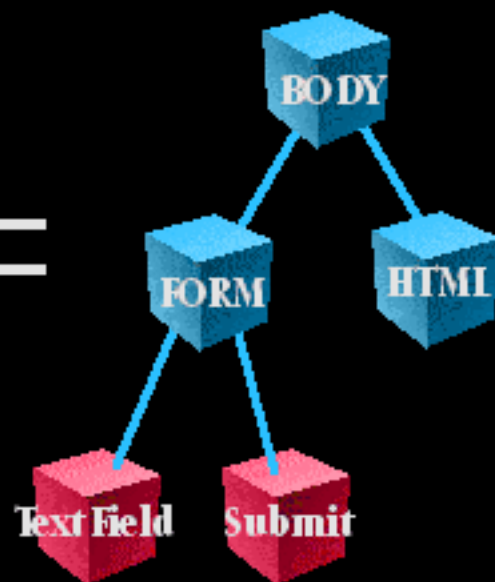
A WebObjects Page

How it Works

```
<HTML>
<BODY>
<FORM>
...
</FORM>
</BODY>
</HTML>
```
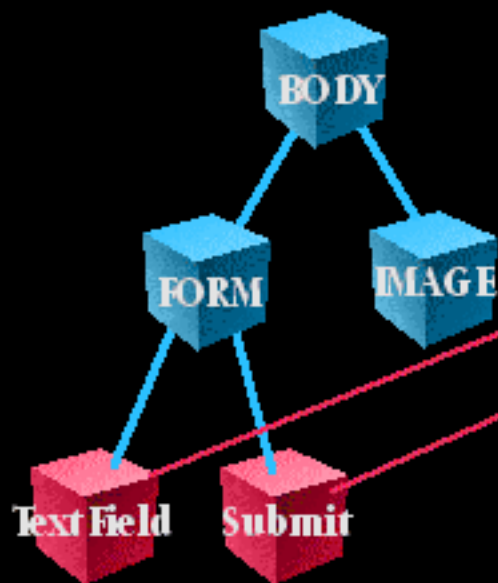
+

```
A: TextField {
    ...
}
```

=

An Object Tree is Built
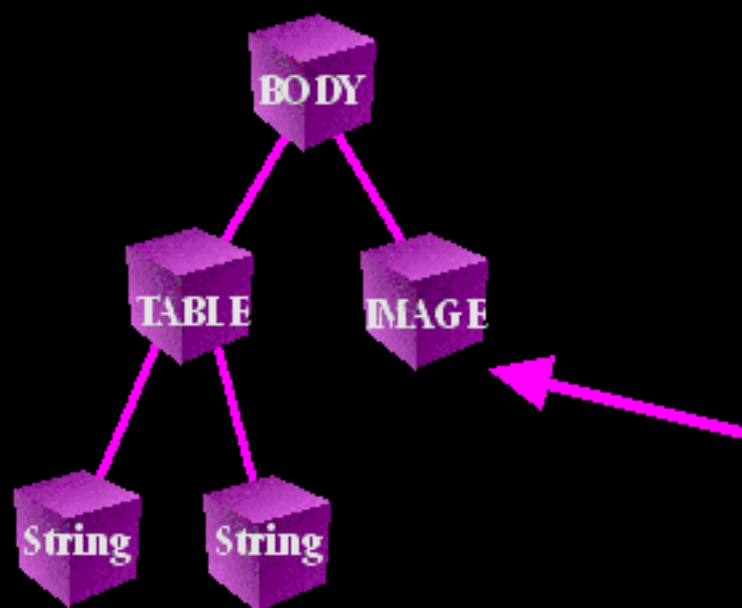
How it Works


```
id data;

function setData {
    object=doIt();
    return object;
}
```
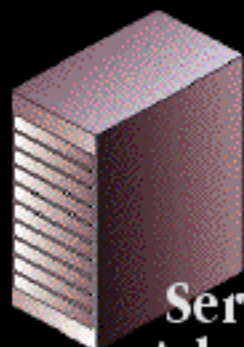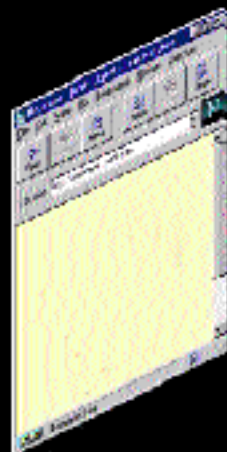
Code is Executed

How it Works


```
id data;

function setData {
    object=doIt();
    return object;
}
```

New Tree is Built
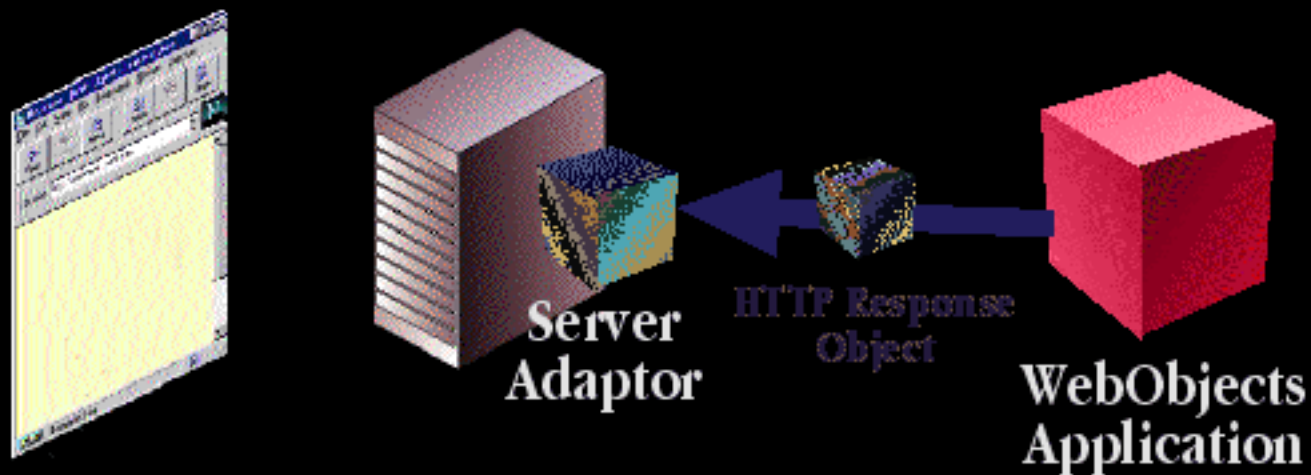## Based on Code

How it Works

Server
Adaptor

WebObjects
Application

Passing the Response

How it Works

Server
Adaptor

HTTP Response
Object

WebObjects
Application

Passing the Response

How it Works

Server
Adaptor

HTTP Response
Object

WebObjects
Application

Passing the Response

Summary

WebObjects Framework Features

- **Handles interaction with HTTP server**

Summary

WebObjects Framework Features

- Handles interaction with HTTP server
- Generates HTML dynamically

Summary

WebObjects Framework Features

- **Handles interaction with HTTP server**
- Generates HTML dynamically
- Database connectivity

Summary

WebObjects Framework Features

- Handles interaction with HTTP server
- Generates HTML dynamically
- Database connectivity
- Manages user state

Summary

WebObjects Framework Features

- **Handles interaction with HTTP server**
- Generates HTML dynamically
- Database connectivity
- Manages user state
- Scalable application server

Summary

WebObjects Framework Features

- **Handles interaction with HTTP server**
  - Generates HTML dynamically
  - Database connectivity
  - Manages user state
  - Scalable application server
  - Technology neutral

Summary

WebObjects Framework Benefits

Summary

WebObjects Framework Benefits

- Separate custom code and data from the Web delivery mechanism

Summary

WebObjects Framework Benefits

- Separate custom code and data from the Web delivery mechanism
- Concentrate on dynamic content, instead of fighting with technology

Summary

WebObjects Framework Benefits

- Separate custom code and data from the Web delivery mechanism

- Concentrate on dynamic content, instead of fighting with technology

- Rapidly develop sophisticated Web applications

Q&A