Application Servers
G22.3033-011

Session 2 - Main Theme
Page-Based Application Servers

Dr. Jean-Claude Franchitti

New York University
Computer Science Department
Courant Institute of Mathematical Sciences

Agenda

- ColdFusion 5.0- MX 6.1 Environment
- PHP 4.3.3 Environment
- XML-Based Application Servers
- Summary
- Class Project Overview
- Readings
- Assignment #2
Application Servers for Enhanced HTML
(a.k.a., Page-Based Application Servers)

- Examples
  - Macromedia ColdFusion 5.0 – MX 6.1 Server
  - Microsoft IIS with ASP
  - WithEnterprise Pty Ltd Tango 2000/WiTango
    - http://www.witango.com
  - etc.
- Typically less expensive than Servers for standalone use, and servers with IDEs
- Technology stays within the familiar HTML confines
  - See Session 2 Sub-Topic 1 Slides on “HTML Review”

The Web Application Platform
Rapid Web Application Development

System Differentiators

- Visual InterDev (ASP)
  - Management of site development process
  - Scripting
  - Macromedia’s Drumbeat or NetObjects’ Fusion can be used as alternative IDEs
- ColdFusion Studio 4.5 – MX 2004
  - HTML coding
  - Basic database integration
  - UltraDev 4 or Fusion can be used as alternative IDE (4.5)
  - Dreamweaver, Flash, Fireworks, FreeHand (MX 2004)
Technology

- IDE + Application Server
- IDE
  - Creates pages with mixture of HTML and proprietary tags or script code
  - Visual page creation (textual creation possible as well)
- Application Server
  - Evaluates the code upon user requests and provides HTML pages

Tagging v.s. Scripting

- ColdFusion
  - Easy tag-oriented dynamic pages for simple tasks
  - Script use when more complex coding is required
    - arrays, case & switch statements, and error handling
- Example
  - Simple phone directory application: 2 custom tags + 1 SQL statement
  - Same would take 100 lines of ASP code ...
Part I

ColdFusion 5.0-6.1 & MX Environment

Also See Session 2 Handouts on:
“ColdFusion CFML Tags, Functions, and Variables”
“The ColdFusion Integrated Web Development Environment”

The ColdFusion Development Platform
ColdFusion Web Applications

How ColdFusion Works

How ColdFusion Works

1. Any Web browser requests a dynamic page
2. The Web server notifies ColdFusion
3. ColdFusion processes the dynamic page based on the CFML tags
4. ColdFusion generates an HTML page
5. The Web server sends the HTML page to the browser
The ColdFusion Development Process

- Write some code
- Save it as a page (use .cfm extension)
- View it in a browser
- Write some more code
- Save the page again
- View it in a browser
- etc.

Sample ColdFusion Application

<HTML>
<HEAD>
<TITLE>My First Page</TITLE>
</HEAD>
<BODY>
<STRONG>ColdFusion</STRONG>
<CFSET ProductName = "ColdFusion">
</BODY>
</HTML>
Outputting a Variable Value

```html
<HTML>
<HEAD>
<TITLE>My First Page</TITLE>
</HEAD>
<BODY>
<STRONG>ColdFusion</STRONG>
<CFSET ProductName = "ColdFusion">
<CFOUTPUT>
#ProductName#
</CFOUTPUT>
</BODY>
</HTML>
```

Querying a Data Source

```html
<HTML>
<HEAD>
<TITLE>Course List</TITLE>
</HEAD>
<BODY>
<H1>Course List</H1>
<CFQUERY NAME="CourseList" DATASOURCE="cfsnippets">
SELECT CORNUMBER, CORNAME
FROM CourseList
</CFQUERY>
<CFOUTPUT QUERY="CourseList" >
#CORNUMBER# #CORNAME#<BR>
</CFOUTPUT>
</BODY>
</HTML>
```
Coldfusión Features
Rapid Development

- Powerful and intuitive tag-based server scripting language
- Two-way visual programming and database tools
- Remote interactive debugging
- Web application wizards & tag-based component architecture
- Source control integration
- Secure file and database access via HTTP

ColdFusion Rapid Development
ColdFusion Server Architecture (5.0)

ColdFusion Features
Scalable Deployment

- Multi-threaded service architecture
- Database connection pooling
- JIT page compilation and caching
- Dynamic load balancing
- Automatic server recovery and fail-over
ColdFusion Features
Open Integration

- Database connectivity (ODBC, OLE-DB, native database drivers)
- Embedded support for full text indexing and searching
- Standards-based integration (directory, mail, etc.)
- CORBA and COM+ connectivity
- Open extensibility with C/C++

ColdFusion Features
Complete Security

- Integration with existing authentication systems (NT/Win 2000 domains, LDAP directory servers)
- Advanced access control to files and data sources
- Support for existing database security
- Server sandbox security
- Support for Web server authentication, security, and encryption
ColdFusion Studio’s IDE

ColdFusion Studio’s WorkSpace
ColdFusion Studio

- Supports other languages than HTML
  - Handled Device Markup Language
  - Synchronized Multimedia Integration Language
- Visual Tool Markup Language
  - Support the inclusion of tag editing dialogs
  - Support the addition of XML capabilities
- CSS integration is clumsy (separate editor)
- Link management utility limited to page by page (no site diagramming)

ColdFusion Homesite Editor

- HomeSite editor
  - Supports on-the-fly typing validation
  - DTD conformance
  - Basic syntax checking
  - Can categorize tag attributes by version and types
  - Can add custom tags and attributes
ColdFusion App Server

- Supports clustering
- Addresses performance and scalability issues at most levels
- Supports ODBC, OLE, and native drivers for Oracle and Sybase
- Also supports stored procedures
- Supports server load balancing (Bright Tiger Technologies’ ClusterATS) and failover

Server Platforms

- ColdFusion
  - Windows
  - Solaris
  - Linux
- ASP
  - Windows
  - Use ChiliSoft for other servers
ColdFusion Features Summary

- Advanced Editor
- Visual Database Tools
- Two-way Visual Programming
- Web Application Wizards
- Code Re-Use
- Interactive Debugging
- Dynamic Page Quality Assurance
- Tag Property Inspection

ColdFusion Features Summary
(continued)

- Code Sweeper
- Extensible Tag Editors
- Custom Wizards
- Visual Tool Object Model
- Customizable Workspace
- Server-Side Source Control
- Shared Project Management
- One-Step Deployment
ColdFusion 5.0 vs. MX 6.1

- See

- MX Features
  - See: http://www.macromedia.com/software/coldfusion/productinfo/overview/
  - Server Scripting (CFML, XML, JSPs/custom tags)
  - Integrated Application Services (Flash, Web services)
  - Flexible Application Deployment
  - High Performance Architecture
  - Advanced Development Capabilities (CFCs)
  - Enterprise Systems Integration
  - Advanced Server Management

Using CFCs As A Façade
(keep code that invokes Java objects out of CFML)
See http://www.macromedia.com/devnet/mx/coldfusion/articles/intro_cfc.html

Hybrid Application
Part II

*PHP 4 Environment*

---

**PHP Technology**

- Server-side, cross-platform HTML embedded scripting language
- PHP is an open source project of the Apache Software Foundation
- Example (hello.php):
  ```html
  <html><head><title>PHP Test</title></head>
  <body>
  <?php echo "Hello World<p"; ?>
  </body></html>
  ```
PHP Examples

- Showing variables
  `<?php echo $HTTP_USER_AGENT; ?>`

- Getting a list of web server variables
  `<?php phpinfo(); ?>`

- Checking for Internet Explorer
  ```php
  if(strpos($HTTP_USER_AGENT,"MSIE")) {
    echo "You are using Internet Explorer<br>";
  }
  ?>
  ```

PHP Examples (continued)

- Jumping in and out of PHP mode
  ```php
  if(strpos($HTTP_USER_AGENT,"MSIE")) {
    ?><center><b>You are using Internet Explorer</b></center><?
  } else {
    ?><center><b>You are not using Internet Explorer</b></center><?
  }
  ?>
  ```
PHP Examples
(continued)

- Flexible HTML Forms Handling
  Typical HTML form:
  
  ```html
  <form action="action.php" method="post">
    Your name: <input type="text" name="name">
    You age: <input type="text" name="age">
    <input type="submit">
  </form>
  
  Action.php is as follows:
  
  Hi <?php echo $name; ?>.
  You are <?php echo $age; ?> years old.
  ```

PHP Software

- Source and binaries downloadable from:
  - Includes
    - CGI binary plus server API versions for Apache, AOLserver, ISAPI and NSAPI
    - MySQL support built-in
    - Many other extensions
Part III

XML-Based Application Servers

XML Application Server Architecture

(HP Bluestone XML Server 1.0/Visual-XML)
XML Application Server At Work
(HP Bluestone XML Server 1.0/Visual-XML)

- See Session 2 handout on “XML MOM Application Server Frameworks”

XML Application Server At Work
(Binary Evolution Velocigen)
Part IV

Conclusion

Summary

- Page-Based Application Servers are either based on HTML tagging or scripting
- Page-Based Application Servers are less expensive to use and simpler than Servers for standalone use, and servers with IDEs
- ColdFusion is based on HTML extensions, and supports the development of tag-oriented dynamic pages for simple tasks
- PHP is a server-side cross-platform HTML embedded scripting language
- XML Application Servers are either MOM- or POP-oriented and rely on server-side processing of XML documents
Readings

- Readings
  - Handouts posted on the course web site
  - Explore the ColdFusion 5.0-6.1/MX and PHP 4.3.3 Environments
  - Read white papers under technical resources at
    - ColdFusion, and PHP related whitepapers on vendors’ sites
  - Review Web/network programming, and HTML

- Project Frameworks Setup (ongoing)
  - Apache Web Server (version 1.3.28/2.0.47, www.apache.org)
  - Perl (version 5.8.0, www.perl.org)
  - ColdFusion 5.0-MX 6.1 (www.macromedia.com)
  - PHP 4.3.3

Assignment

- Assignment:
  - Explore references to Application Server technology (cont.)
  - #2a: Investigate page-based application server development environments. Write a short report that documents your findings and recommendations with respect to selection criteria in support of page-based development environments for application server technology
  - #2b: See homework #2 specification
Next Session:
Page-Based Application Servers (Part II)

- ASP Environment (i.e., IIS with COM+/.Net & ASP)
- Servlets and Servlet Engines
- JSPs and JSP Engines
- TomCat/JRun
- Cocoon 2/XSPs