Web Application Development with ColdFusion

Charles Arehart
Systemanage
Carehart@systemanage.com

ACM Seminar Series, November 1999

Audience

Who Are You?

- Web site developer
  - looking to add more interactivity to your site
- Web application developer or development manager who's heard of ColdFusion
  - wondering how it works, how easy it is to use, how scalable it is for enterprise apps

What We'll Cover

- We'll show:
  - how it can be used to add interactivity to your site
  - how easy CF is to use and how it works
- Not a sales seminar
  - definitely task oriented
  - discussions of basic topics and features
  - several live walkthroughs

Introduction

About ColdFusion

- Leading Web Application Development System
  - more than 430,000 CF developers worldwide
- Very easy to use, yet capable of enterprise-class applications
  - being used by many large organizations

Some Prominent CF Sites

- Netgrocer
- Autobytel
- Toysrus
- Smartmoney
- Reebok
- Casio
- Igolf/Golfonline
- Moen Faucets
- Kodak Intranet
- IRS Intranet
- DHL Australia
- Cotton Incorporated
- and many more

Why Consider CF?

- There are several key benefits to using CF
  - some are easily understood, others are somewhat more advanced
- We'll present a few of these, at a high level, before proceeding
  - not enough time to explain in detail, but may help frame it for those with some background in web app development

CF: Ready for Prime Time

- Robust
  - Industry leading development platform, at R4.5
- Scalable
  - Capable of enterprise-class applications
- Secure
  - Integrates with OS security

Broad Database Support

- CF can communicate with virtually any database, including:
  - SQL Server (Microsoft and Sybase)
  - Oracle
  - DB/2
  - Informix
  - and many more enterprise DBMS's, as well as desktop DBMS's such as MS Access

Broad OS/Web Server Support

- CF can run on Windows 95/98/NT, as well as Unix (Solaris, Linux, HP-UX)
- Can run on all web servers, with high-performance integration on Netscape, Microsoft, Apache, and other major servers

Great Portability

- Can generally change databases without impact on application
  - using ANSI standard SQL
- Can switch Web servers without impact
- Can even switch platforms without code change (Unix->Windows NT/vice versa)
- Almost unheard of portability!

Scalability

- ColdFusion now includes built-in clustering technologies
  - support for running multiple CF servers to enable high-volume transaction processing
- Many significant performance features
- Allaire also recently acquired Live Technologies, makers of JRUN
  - can now integrate CF apps and Java Servlets/Java Server Pages

Yet So Easy to Use!

- All those features are great
  - and should give comfort when discussing CF with those not familiar with it
- But what's better, is it's so easy to use!
- This presentation will focus on simpler aspects of using, developing in CF

Outline of Topics

Topics

- Basics of web page processing
- Making a static web page more dynamic
- Building web forms to accept user input
- Basics of SQL, the language of databases
- Creating database-driven web pages
- Adding search, data entry capabilities
- Incorporating Javascript and using wizards

Basics of Web Page Processing

You probably know this stuff, but...

- ACM seminar participants generally programmer oriented
  - but may not necessarily know HTML
  - need to start with most basic HTML processing
- Good news is that CF can work with very basic HTML, which is easy to learn
  - programmers can quickly learn and appreciate CF's capabilities

Simplest Web Pages

- Web pages are built using HTML
- A very simple layout description language

```
<HTML>
<BODY>
<H1>Employees</H1>
John Smith<br>
Bob Jones<br>
</BODY>
</HTML>
```

Browser Interprets HTML

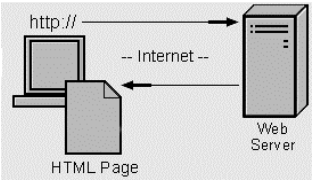
- Page containing HTML stored as file with .htm extension
- Can be stored anywhere on file system and viewed with any browser
- When a user opens that file in a browser
  - that HTML is rendered as follows

Web Page Storage

- Making them available for public view:
  - store file on a “web server” accessible to all users on internet (or intranet)
  - users browse to file with http://www.yourdomain.com/thefile.htm

Static Page Processing

Web browser makes request for a .htm page
Web server sends that page back to browser
Web browser interprets and renders the HTML



Demonstrations

- Browsing web pages, making changes, and viewing results

The Exploding Web

- This ease has made the web explode
  - nearly anyone can learn and apply this capability
- Many have reached limits of possibilities of simple HTML
  - may want to prompt users for input and process result (search pages, data entry applications)
  - may want to query or update databases
- CF makes that sort of thing very easy!

Demonstrations

- Coffee Valley Sales Application
  - simple search interface
  - advanced search interface
  - simple data entry interface
  - advanced data entry interface

Making a Static Web Page More Dynamic

Why Dynamic Pages?

- Often web pages become stale for lack of changing content
  - someone responsible for “updating” pages, job often goes undone
- More important, data may already be in databases (or available from system)
  - would be much easier to generate web pages from that database (or with that system info)
  - with no manual intervention

A More Dynamic Page

- Simple example: display today's date/time on web page:

```
<H1>Employees</H1>
John Smith<br>
Bob Jones<br>
<p>
<CFOUTPUT>
Date: #date#
</CFOUTPUT>
```

Employees

John Smith
Bob Jones

Date: Nov 14, 1999

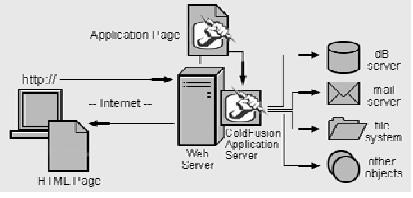
(For simplicity, presumes "date" variable was created and formatted previously on page)

CF Tag Processing

- Notice CFOUTPUT tag on previous page
  - this is not an HTML tag, instead is CF tag
  - called CFML, or ColdFusion Markup Language
  - CFML looks like HTML, but is not understood by the browser
- Instead, CF tags are processed on web server first:
  - CF tags often generate HTML

Server Interprets CFML

- Page containing CFML (and HTML) stored as file with .cfm extension
  - web server passes file to CF Server to process



Demonstrations

- Viewing CFML source
- Browsing that page to see the conversion of CFML to HTML
- Observing dynamic change of date/time without page modification
- Including a single navigational component on several pages

Creating, Using Variables

- CFSET tag creates variables:

```
<CFSET FirstName = "Teddy">
<CFSET LastName = "Bear">
<CFSET FullName = FirstName & " " &
  LastName>
<CFOUTPUT>
Name is: #FullName#
</CFOUTPUT>
```

Using Functions

- Functions expand range of data manipulation, access to system information
- Types of functions include:
  - String Processing
  - System-Information
  - Date/Time
  - Display/Formatting
  - List, Array, and Structure
  - Mathematical and Trigonometric

Demonstrations

- Various Function Examples
  - date
  - date formatting
  - string formatting
  - number formatting

Server vs Client Processing

- ColdFusion page can only have CFML, no other server-side processing
  - such as ASP, PERL, Java
- But it can send to browser any valid client-side code
  - such as Javascript, VBScript, Java applets, Activex controls, DHTML

Technology Integration

- ActiveX
- COM/DCOM
- CORBA
- JavaBeans
- JSP, Servlets

- Cybercash
- ICVerify
- OpenMarket

- Verity SEARCH'97

- Macromedia Flash
- Macromedia Generator

- Macromedia Dreamweaver
- NetObjects Fusion

- XML
- SMIL
- HDML SDK

Site Updating: Old Way

- Manual maintenance
  - many sites maintain lists of data on web pages
  - changes are made manually
  - updates are e-mailed in to person responsible
- Bottleneck
  - maintainer must know HTML
  - maintainer must make time to read and process emails
  - some changes fall through cracks, rarely timely

Site Updating: The CF Way

- Fully automated, database-generated approach
  - static data is moved to a database (quite easy)
  - ColdFusion used to read data from database and display on web page. No change to user.
  - new administrator interface used to update data
    - or let the users enter the data themselves!

Demonstrations

- Manually updated site (ultm204)
- User-updated site (hpcareer.net)

Other Possibilities

- Auto-generate email to registrants, as well as to company insiders upon registration
- “Approval” process for submissions, if needed
- many, many more

- Let’s discuss HTML form processing

Building Web Forms to Accept User Input

Gathering User Data

- Previous example showed registration form
- HTML “Forms” are the key to gathering data from web visitors
  - registration forms
  - search interfaces
  - data entry interfaces
  - and more

Form Elements

- Forms composed of following elements:
  - Text entry (single- and multiple-line)
  - Choice selection (checkbox, radio, drop-down)
  - Buttons (submit, clear)
  - Filename for upload
- HTML syntax covered in basic HTML books

Demonstrations

- Creating a simple form
- Looking at search and data entry forms
- Using Studio tools to simplify building forms

Processing Forms

- While forms are easy to create, the challenge is in processing them
  - can't do in just HTML
  - need server side process to interpret, act on form
  - traditionally PERL and CGI scripts
    - some pre-canned scripts available for common tasks
    - difficult to create for custom purposes
- ColdFusion makes form processing EASY!

ColdFusion Form Processing

- Simply name a ColdFusion template as the form's "action" page
- On that page, all form fields are available as variables
- If form had fields named firstname, lastname:
  - refer to them on action page as:
    - Form.firstname
    - Form.lastname

Demonstration

- Displaying form data on form action page

Sending Form Result as Email

- Can use form data in many ways
  - will show database interaction next
- Simple example might be sending email based on form data
  - can be used to send email of a registrants interest in a subject
  - or for sending tech support request, etc.
- ColdFusion offers simple CFMAIL tag

Demonstrations

- Sending email with CFMAIL tag
- Using CFMAIL on a form action page

Database Integration

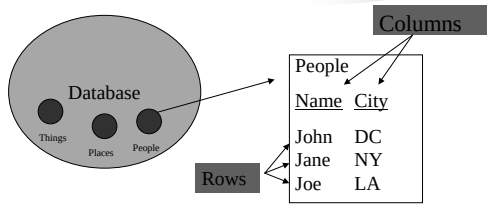
- Natural next step is to use form submission for querying or updating a database
  - ColdFusion offers easy integration with databases
  - Can easily use form variables to query/update DB
- First need to understand databases and how to query and update them with SQL

Basics of SQL, the Language of Databases

Again, Some May Already Know...

- Before explaining CF database integration, let's review basics of databases and query processing
- Databases are composed of tables
  - tables are composed of records and columns
- SQL, or Structured Query Language, is a standard language for database processing
  - ColdFusion leverages SQL processing
  - you must understand SQL and db processing

Tables, Rows & Columns



Selecting Data From Table

- Most basic database processing is querying a table for data
- SQL SELECT statement is simple:

  SELECT NAME, CITY FROM PEOPLE
- Retrieves all records from PEOPLE table, returning all values for NAME and CITY columns
  - can list as many or as few columns as needed

Query Results

- In traditional database systems, this SQL is entered in some query tool, and the result is displayed to user:

Name	City
John	DC
Jane	NY
Joe	LA

ColdFusion Query Processing

- In ColdFusion, that result is not “displayed” to user:
  - instead is made available to CF program as a query result set
  - up to CF program to determine what to show, and how to format
- Will show how to do this later
- For now, let’s see how to execute and display queries for testing purposes

CF Studio Query Builder

- CF Studio is the “Integrated Development Environment”, or IDE, for CF development
- Primarily an advanced HTML/CFML editor, as has been demonstrated briefly
- Valuable component is Query Builder
  - allows easy drag and drop creation of SQL
  - can test SQL without creating CF code to process
  - can build queries against all databases on server

Demonstration

- CF Query Builder
  - viewing data in various server database tables
  - using query building features against a single table

Limiting Rows Selected

- To limit which rows are returned, use WHERE:

```
SELECT NAME FROM PEOPLE
WHERE CITY='DC'
```

- Result is:

Name
Joe

Demonstration

- Building WHERE criteria in query builder

SQL for Updating Data

- Equally simple SQL for insert/update/delete
- INSERT INTO PEOPLE (name, age)
  VALUES ('Charlie',36)
- UPDATE PEOPLE
  SET age = 37
  WHERE personid=1
- DELETE FROM PEOPLE
  WHERE personid=1

Creating Database-Driven Web Pages

Three-tiered approach

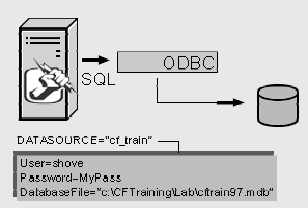
- In client server (two-tier) approaches
  - connection to database was from client directly to database
- ColdFusion is a three-tiered approach
  - user connects to ColdFusion page using browser
  - ColdFusion Server connects to database on user's behalf, builds resulting HTML page, sends to user

2 Steps to Database Connection

- Using databases with ColdFusion is easy
- Administrator
  - describe to ColdFusion Server the connection to the database
- Programmer
  - code the CF template providing the SQL to be performed
    - use ColdFusion variables, functions, and more to dynamically build the SQL as needed

Describing the Connection

- Databases in CF are accessed through "datasources"



Datasources?

- A Datasource is simply a name that points to a complete description of the database:
  - type of connection
  - location
  - username/password, if needed
  - much more, if needed

Types of Connection

- "Type of connection", or "driver", can vary based on database management system
  - ODBC is a generic driver supported by nearly all databases, and even simple text or spreadsheet files
  - OLEDB is a more recent MS-specific driver
  - large-scale databases like Oracle, Sybase, DB2, and Informix offer native drivers

ODBC Commonly Used

- ODBC is a standard of connection that's been around for years, now ANSI standard
- Connecting to ODBC database is a standard feature of CF
  - even in the ColdFusion Express product
- May not be the best choice for performance and scalability
  - Great news: CF code doesn't vary if datasource is changed

Executing SQL in CF

- SQL is sent to database in CF using CFQUERY:

```
<CFQUERY DATASOURCE="xxx"
NAME="yyy">
    SQL statement
</CFQUERY>
```

- NAME attribute used for later reference

Demonstration

- Defining a datasource
- Placing SQL in a CFQUERY

Result of SQL Statement

- While most SQL processing tools simply automatically display the results
  - CF holds results in memory to be displayed at your control
  - we refer back to the query by its NAME attribute
- All column names from resulting SQL become variables:
  #queryname.columname#

Demonstration

- Viewing CFQUERY results

Looping Through Results

- Often a query will result in many records, all held in memory awaiting processing
- CFOUTPUT QUERY="xxx" loops through all records in query named "xxx"
  - can optionally indicate startrow and maxrows

```
<CFOUTPUT QUERY="xxx">
  statements that are looped over
</CFOUTPUT>
```

Demonstration

- Looping over CFQUERY results

Using Results to Build HTML

- Result of CFQUERY can be simply displayed, or used to format HTML elements:
  - Tables, Lists, Form elements
  - and much more
- Simply need to understand how those HTML elements are built
  - and substitute CF variables for data

Demonstration

- Building an HTML list as a result of a query
- Building an HTML Table
- Building form elements

Adding a Search, Data Entry Capabilities

Using Form Data in Queries

- Simple combination of forms, SQL, and CF can create powerful applications, easily
- We saw use of form data for display or even sending email
  - can also use to build SQL statements on-the-fly
- Obvious applications are search, data-entry interfaces

Dynamic SQL

- SQL within a CFQUERY can certainly refer to CF variables and functions:

```
<CFQUERY …>
        SELECT Name From People
        WHERE AGE = #form.age#
</CFQUERY>
```

Still More Dynamic SQL

- Can even use CF tags to conditionally perform SQL:

```
<CFQUERY …>
        SELECT Name From People
        <CFIF isnumeric(form.age)>
                WHERE AGE = #form.age#
        </CFIF>
</CFQUERY>
```

Building a Search Interface

- Can use this to build search interfaces
- Form presents prompts for user to describe expected results
  - action page builds SQL and search criteria using form data to add criteria

Demonstration

- Simple search interfaces
- More advanced search interfaces

Building Data Entry Interfaces

- Just a natural extension, using SQL that performs updates rather than queries
- ColdFusion also offers simple means to validate form data:
  - required, integer/float, date, time, range of values
  - placed as “hidden” fields on form
  - ColdFusion action page will analyze form on submission and report errors if validation fails

Demonstration

- Simple data entry interface
- Data validation
- More advanced data entry interfaces

Incorporating Javascript

What is Javascript?

- Javascript is a language used primarily for extending web browser interfaces
  - it is not Java, nor a subset nor really related in any way
  - some dismiss it as “too hard”
    - while others dismiss it as “nothing important”
  - it’s neither: it’s important and not hard to learn

Browser Support

- One challenge to using it is that older browsers may not support it (or not fully)
- Also, Netscape Navigator and IE have their own slightly different variations on the language
- ECMAScript is a standard which tries to unify a common core of the language
  - IE generally regarded as better implementation (more complete and closer to standard)

Three Views of Javascript

- Core Javascript
  - parts of the language for straightforward programming (assignments, object creation, flow control, etc)
- Browser Javascript
  - parts of the language for manipulating browser interfaces, including the document object model
- Server Javascript
  - use of the language in server-side processing, without connection with the browser

CF Can Build Javascript for You!

- CF can build simple Javascript-driven data-entry validation for you
  - You don't need to learn javascript
- Simply change FORM to CFFORM, INPUT to CFINPUT, and add some validation attributes
  - REQUIRED="yes", VALIDATE="integer"
- CF converts this back to a form but also sends along complex cross-browser javascript

Demonstration

- CFFORM code
- Result of CFFORM

Additional Client-side Validation

- Besides the basic validations available already, also adds:
  - credit card
  - social security number
  - telephone number
  - us zip code

But You Should Learn More!

- CF builds simple javascript validation routines
- You can build very powerful web interfaces with more knowledge of Javascript

What Javascript Can Do for You

- Dynamic HTML
- Browser event handling
- Dynamically populated form elements
- Frame-based processing
- much much more

Demonstrations

- Advanced Javascript-driven interfaces

Great Resources for More

- O'Rielly's *JavaScript: The Definitive Guide* by David Flanagan
  - excellent resource, both as tutorial and reference, with good coverage of both core and cross-browser js
- IDG's *JavaScript Bible* by Danny Goodman
  - more tutorial than reference, but quite a large tutorial, with a slight netscape bias

Using Wizards to Create Applications for You

ColdFusion Studio Wizards

- CF Studio wizards can build simple, but complete, applications:
  - Data Entry
  - Data Drill Down
  - Record Viewer
- The results are rather simplistic, but can solve a problem quickly
  - and serve as useful examples for study

Wizard Walkthrough

- Wizards prompt for all info needed for application:
  - Choose among available CF Server (remote, too!)
  - Choose among available datasources
  - Choose among available databases
  - Choose from available tables, columns, etc.

Demonstration

- Data entry, data drill down, and record viewer wizards

Conclusion

ColdFusion: A Complete App. Dev. Environment

- We've seen how easy CF is
- Discussed that it's also:
  - scalable (clustering, caching, etc.)
  - secure (advanced security, integrated with OS)
  - robust (fail-over, load balancing, multi-threaded)
  - integrated with other tools and resources (CORBA, COM, EJB, and more)
- CF Studio provides integrated dev env (IDE)

Comparison to ASP, Java

- Microsoft Active Server Pages
  - Free, heavily leverages VBScript
  - Very similar in framework and capabilities
  - More cumbersome to code, maintain
- Java Server Pages, Servlets
  - Powerful new approach leveraging Java for server-side web application development
  - Again, more cumbersome to code but does bring all the power of Java

Demonstration

- ASP and Java Code samples, compared to equivalent CF code

Where to Learn More

- *ColdFusion Web Application Construction Kit*, by Ben Forta, et al
  - THE bible in many people's eyes
- www.allaire.com
  - an excellent informational and support web site
- ColdFusion Developer's Journal
  - www.coldfusionjournal.com
- CFAdvisor.com, Defusion.com

Good Luck!

- And enjoy ColdFusion!