Building Dynamic WAP Applications with ColdFusion

Charlie Arehart
Chief Technology Officer
SysteManage
carehart@systemanage.com
Web: www.systemanage.com

conferences

Why Should You Stay?

In this session you'll learn:
- What dynamic content is, and why it's useful
- What ColdFusion (CF) is, and why it's powerful
- How to build dynamic WAP applications with CF
- Solving Typical Problems Coding WAP in CF

Audience:
- WAP developers new to CF
- Experienced CF developers looking at WAP

What Will We Cover, Specifically?

Preaching the good news:
- What is a dynamically generated WAP application?
- What makes them so great?
- Introduction to ColdFusion, a stellar tool for building them

Preaching to the choir:
- Serving WAP content via ColdFusion
- Studio features for creating WAP applications
- Tricks and traps

What is a Dynamically Generated WAP Application?

Content generated on the fly
- Perhaps generated from a database
- Or based on user input
  - Creating 2-way WAP applications

Requires a web application server
- Such as ColdFusion, ASP, Perl, Java, etc.

What Makes Them So Great?

Dynamic Sites--Sign of a mature site:
- Site content generated from databases
- Query as well as update of databases
- Content customized to visitor
- Securing of site sections and features
- Generation of e-mail to clients, site mgrs
- Integrating content from other applications or even other sites
- Scheduled content (push, notifications)

About ColdFusion

Leading cross-platform web application server
- Runs on Windows, UNIX (including Linux)
- Runs with nearly all web servers (including Apache)

Recognized for:
- Security, scalability, integration, ease of use

Inexpensive:
- Ranging from just $1,295-5,000 US, single server
- Free version with some limitations of functionality

CF's Ease of Use

```
A Quick WML Example

<card>
    <CFQUERY NAME='GetDepts' DATASOURCE='Test' >
        Select DeptName from Depts
    </CFQUERY>
    <p>Depts Found: <br/>
        <CFOUTPUT QUERY='GetDepts' >
            #DeptName#<br/>
        </CFOUTPUT>
    </p>
</card>
```

CF's Flexibility

Many operating systems and web servers supported

Databases supported:
- Any ODBC (Access, FoxPro, etc.)
- SQL Server, Sybase, Oracle, Informix, DB2

Integration:
- COM/DCOM, CORBA, EJB, etc.

CF's Scalability

Can handle substantial loads:
- Integrated load balancing
- Clustering
- Failover

High performance architecture
- Multi-threaded service, template caching, etc.
- State management

Serving WAP Content via ColdFusion

- CF-generated pages are .cfm files rather than .wml files

- Must set MIME type for the page:
  ```
  <CFCONTENT TYPE="text/vnd.wap.wml">
  ```

- This tells the WML browser to expect WML
  - tag not normally needed in HTML pages

- This is where things get different for experienced CF developers

---

Complete Skeletal CF/WML File

```
<CFCONTENT TYPE="text/vnd.wap.wml"><?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//PHONE.COM//DTD WML 1.1//EN"
                     "http://www.phone.com/dtd/wml11.dtd">
<wml>
  <card>
  </card>
</wml>
```

- Note <?xml> tag on first line
- Is placed next to CFCONTENT
  - Prevents extra carriage return before <?xml>
- Note Phone.com DTD: could use WAPForum's

Where to Go From Here?

Learn CF and learn WML:
- Place any valid WML on page
  - WML is different from HTML, do need to learn it
- Use any ColdFusion tags to create WML code
  - Can generate WML from database, or based on data gathered from other objects, applications, or sites
- Perform actions in support of WML processing
  - Can search/store data in databases, generate e-mail, pass data to other objects, applications, or sites

Some CF Examples

- Form processing
- Form-driven query processing
- Dynamic select list

Form Processing Example

- Goal:
  - Create mechanism for user to enter text input
  - Process that input on the server
    - For now, simply display result to user

  - Will soon do more with the input
    - As well as change process to limit user data entry via typing

Form Processing Code - Part 1

```
<CFCONTENT TYPE="text/vnd.wap.wml"><?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
<wml>
<card>
    <do type="accept">
        <go href="wml_action.cfm" method="post">
            <postfield name="symbol" value="$(symbol)"/>
        </go>
    </do>
  <p>
    Enter stock symbol:
    <input name="symbol" format="4A"/>
  </p>
</card>
</wml>
```
wml_form.cfm

Form Processing Code - Part 2

```
<CFCONTENT TYPE="text/vnd.wap.wml"><?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
<wml>
<card>
<p>
The stock symbol selected was:
<CFOUTPUT>#form.symbol#</CFOUTPUT>
</p>
</card>
</wml>
```

wml_action.cfm

Potential Problems with Forms

➢ Some phone/gateway combinations will not send
  form field values sent via "method=post"
  – Not really a ColdFusion issue
➢ Form will submit, but action page will have no
  form fields to process

Solving Form Submission Problems

➢ Simple solution is to change method to "get"
  – Form fields then passed as CGI query string
➢ For ColdFusion, value of variable changes from
  "form.*variablename*" to "url.*variablename*"
➢ Change previous code to:
  – <CFOUTPUT>#url.symbol#</CFOUTPUT>

Query Processing Example

> Goal:
> – Extend example to use user input to lookup stock symbol in server-side database

> Notes:
> – Will change to using method="get"
> – Also changed from using POSTFIELD to using QUERY String to pass variables

> – Will drop tags at start and end of code
> – CFCONTENT, wml, card
> – Can presume it's always there

Query Processing Code – Part 1

```
<do type="accept">
        <go href="getholdings_action.cfm?symbol=$(symbol)"
method="get"></go>
</do>
<p>
Enter stock symbol:
<input name="symbol" format="4A"/>
</p>
```

Getholdings_form.cfm

Query Processing Example

> Action page will perform the search
> Sample "Portfolios" Database Schema
> – ClientHoldings Table to track stock holdings
> – HoldingsId – primary key, auto generated
> – ClientId – foreign key, related to Clients table
> – Symbol – stock symbol owned
> – NumShares – number of shares purchased
> – PurchaseDate – Date purchased
> – PurchasePrice – Price at which shares were purchased
> – Related Clients table with client details
> Search page presumes session variable named "clientid", tracking current user

Query Processing Code – Part 2a

```
<CFQUERY DATASOURCE="Portfolios" NAME="GetHoldings">
        SELECT * FROM ClientHoldings
        WHERE ClientId=#session.clientId#
        AND Symbol='#url.symbol#'
</CFQUERY>
<CFIF getholdings.recordcount IS 0>
        <CFOUTPUT>
        <p>You have no holdings for #url.symbol#</p>
        </CFOUTPUT>


<!--- continued on next page -->
```

Getholdings_action.cfm

Query Processing Code – Part 2b

```
<!--- continued from previous page -->
<CFELSE>
        <CFOUTPUT>
        <p>Holdings for #url.symbol#:</p>
        </CFOUTPUT>
        <p mode="nowrap">
        <CFOUTPUT QUERY="GetHoldings">
                #NumShares# on #PurchaseDate#  at
#PurchasePrice#<br/>
        </CFOUTPUT>
        </p>
</CFIF>
```

Getholdings_action.cfm

Dynamic Select List Example

- Goal:
  - Extend example to present user's current Stock Holdings symbols
    - Much easier for user than typing data

  - Will perform query of database and use results to build form's Select control

Dynamic Select List Code

```
<CFQUERY datasource="portfolios" name="getstocks">
        SELECT distinct symbol FROM clientholdings
        WHERE Clientid=#session.clientid#
</CFQUERY>
<CFIF getstocks.recordcount IS 0>
        <p>You have no stocks to view</p>
<CFELSE>
        <do type="accept">
                <go href="getholdings_action.cfm?symbol=$(symbol)"
method="get"></go>
        </do>
        <p>
        Choose one of your stocks to view:
        <select name="symbol">
                <CFOUTPUT query="getstocks">
                <option value="#symbol#">#symbol#</option>
                </CFOUTPUT>
        </select>
        </p>
</CFIF>           Getholdings_list_form.cfm
```

Dynamic Select List Example

➢ Goal:
  – Extend example to present user's current Stock Holdings symbols
    – Much easier for user than typing data

  – Will perform query of database and use results to build form's Select control

---

Other Useful CF Features

➢ Database Updates, via SQL INSERT/UPDATE
➢ CFMAIL
➢ CFSCHEDULE
➢ CFSEARCH
➢ Session, Client, Application, and Server variables
➢ Query caching

➢ Can learn more about these in CF manuals
  ➢ Or any of several books on the subject

Studio/HomeSite Features for Creating WAP Applications

CF Studio is the IDE for CF developers
  - HomeSite HTML editor shares most features

Both are advanced Text Editors
➢ Offer many useful features for web developers

As of 4.5, both now support WML coding
➢ Creating new WAP templates
➢ Creating/editing WML tags
➢ Documentation of WML tags

A Quick Tour of Studio/HomeSite

➢ Creating a WML page
➢ Adding WML tags
  – Tag insight, validation
  – Tag chooser
➢ Editing WML tags
  – Tag editors
➢ Getting help
  – F1, help within tag editors

Some CF/WML Traps

➢ CFCONTENT problems
  – Can be turned off by administrator
  – Not supported in CF Express
➢ Studio/HomeSite support
  – Based on Phone.com WML
  – Option may enter tags in upper case
➢ CF's HTML heritage
  – CF error messages and debugging output are both in HTML format
    – Can alleviate problems with properly configured application.cfm file

An Ounce of Prevention

- Application.cfm is a special file run before every other .cfm file
  - It can turn on features to solve some "HTML heritage" problems

- Place it in the same directory as WML code
  - Preferably a WML-only directory

Sample Application.cfm

```
<CFSETTING ENABLECFOUTPUTONLY="Yes">
<CFERROR type="exception" template="wml_error.cfm">

<CFSETTING SHOWDEBUGOUTPUT="no">
<CFSETTING ENABLECFOUTPUTONLY="no">
```

Sample wml_error.cfm - Part 1

```
<CFCONTENT TYPE="text/vnd.wap.wml"><?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML
   1.1//EN" "http://www.wapforum.org/DTD/wml_1.1.xml">
<wml>
<card>
<CFOUTPUT>
<p><b>An error has occurred in this application. Please
   share this diagnostic information with the system
   administrator<CFIF CFERROR.mailto IS NOT ""> at
   #CFERROR.mailto#</CFIF>.</b></p>
<p><b>Error Text:</b>
   #htmleditformat(CFERROR.diagnostics)#</p>
<p><b>DateTime:</b> #CFERROR.datetime#</p>

<!--- continued on next page --->
```

Sample Error.cfm - Part 2

```
<!--- continued from previous page -->
<p><b>Error Template:</b>
<CFIF CFERROR.Template IS
  "">(unavailable)<CFELSE>#CFERROR.Template#</CFIF></p>
<p><b>Referrer: </b>
<CFIF CFERROR.httpreferer IS
  "">(unavailable)<CFELSE>#CFERROR.httpreferer#</CFIF></p>
<p><b>Query String:</b>
<CFIF CFERROR.querystring IS
  "">(unavailable)<CFELSE>#CFERROR.querystring#</CFIF></p>
<p><b>Browser: </b>
<CFIF CFERROR.browser IS
  "">(unavailable)<CFELSE>#CFERROR.browser#</CFIF></p>
</CFOUTPUT>
</card>
</wml>
```

Some CF Tricks

- Circumventing CFCONTENT restriction
- Browser detection
- Other items:
  - Converting semicolon-delimited "multiple SELECT" values
  - Escaping # and $

- Covered in detail in Professional WAP book, chapter on CF/WML programming

Other CF/WML Resources

- Some sites:
  `www.wapuseek.co.uk/wap_cf.cfm`
  `www.systemanage.com/wml/`
  `www.allaire.com/developer/TechnologyReference/wap.cfm`

- CF-WAP mailing list:
  Sign up at `www.houseoffusion.com`

Crossing the Finish Line

> Dynamically generated WML is a powerful tool
> ColdFusion is especially well-suited
> Taking your sites to this next step will add tremendous value

Building Dynamic WAP Applications with ColdFusion

Charlie Arehart
Chief Technology Officer
SysteManage
carehart@systemanage.com
Web: www.systemanage.com

wroxconferences