SYS-CON

SPECIAL FOCUS: DATA PRESENTATION

ColdFusion Developer's Journal

ColdFusionJournal.com

January 2005 Volume:7 Issue:1

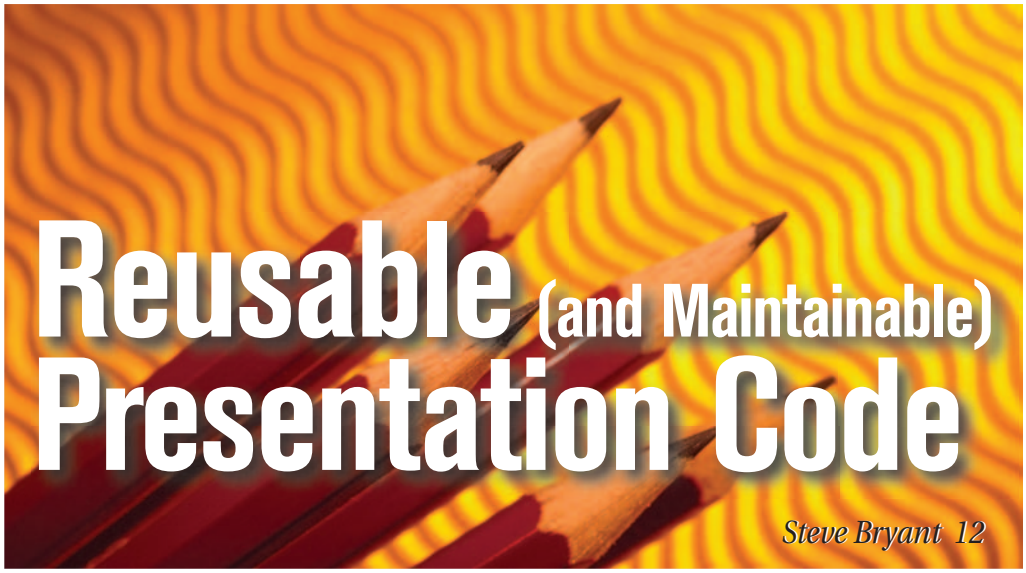
RETAILERS PLEASE DISPLAY UNTIL MARCH 31, 2005

$9.99US $9.99CAN

02

0 09281 02689 1

SYS-CON MEDIA



Reusable (and Maintainable) Presentation Code

Steve Bryant 12

macromedia

This is:

[ ] Video

[ ] Flash

[ ] Don't bother me. I'm staring.

The line between Flash and Video is forever gone.
Travel with us to Scandinavia for a behind the scenes tour
of the award-winning, genre-bending Volvo V50 website.

macromedia.com/go/volvo

CFDynamics

Dedicated Server Packages Starting at $189/mo.

All dedicated servers include:

- **FREE STATS SOFTWARE!**
- **FREE MAIL SOFTWARE!**
- **No long term commitments!**
- **Fair and simple pricing!**
- **FREE SQL server access!**
- **Optional server maintenance!**

As one of the premier ColdFusion hosting community leaders, CFDynamics is constantly looking for ways to provide **better service** to ensure your satisfaction. Not only do we offer the **finest in shared hosting plans**, but we now offer the **finest in 100% dedicated server plans**! Now you can afford the freedom of having your own dedicated server!

When your needs have outgrown shared hosting look to CFDynamics for **total freedom.** With dedicated server packages they're not offering an oxymoron; "virtually private" or "virtually dedicated" is NEITHER private nor dedicated. CFDynamics offers a solution that is **100% completely dedicated**. They don't play games with the fake stuff; CFDynamics only offers the real deal. Real Service. Real Satisfaction. Real Value.

Real Freedom.

macromedia ALLIANCE PARTNER

Paper Thin Partner

BlueDragon Certified V.I.P. Hosting Partner

Visit us online or call to order!

WWW.CFDYNAMICS.COM

1 - 866 - 233 - 9626
866 - CFDYNAMICS

ColdFusion Developer's Journal

editorial offices

SYS-CON MEDIA
135 Chestnut Ridge Rd., Montvale, NJ 07645
Telephone: 201 802-3000 Fax: 201 782-9638
COLDFUSION DEVELOPER'S JOURNAL (ISSN #1523-9101)
is published monthly (12 times a year)
by SYS-CON Publications, Inc.

postmaster: send address changes to:
COLDFUSION DEVELOPER'S JOURNAL
SYS-CON MEDIA
135 Chestnut Ridge Rd., Montvale, NJ 07645

New Year Brings Change

By Simon Horwith

For several months I've been promising there would be changes in *CFDJ* in the New Year. Change in the form of offering deep-focused issues. Change in the form of new regular and semi-regular columns. Well, the time has finally come. This month we are focusing on presenting data. A large number of our articles have to do with getting all of that raw data out of our databases and server memory, and onto the screen in a friendly format. This is as important, if not more important, than the functionality you code in your applications. I say this because if your user interface isn't pretty and/or easy to use, then your application most likely won't see much use. Clients and bosses like things that are pretty and they like interfaces that let them do what they want to do with your application as easily as possible. The good news is that you don't have to be a Picasso to create nice interfaces – hopefully our articles this month will shed new light on creating easy-to-use front ends.

This month we have an article by Steve Bryant about creating reusable and easy to maintain presentation layer code. Steve Nelson has written an article about using personas to visualize end-user interaction with the interfaces you create. Jeffry Houser, in his regular CF 101 column, examines grouping data for output in applications. The articles this month aren't just about creating simple HTML interfaces, either.

I have written an article that is a case study of a prototype SVG Gantt chart engine that allows designers and developers to create robust chart views of an application's underlying data. We also have an article from Matt Woodward that is a good introduction to creating Flash RIA front ends using Flex.

Presenting data is important, but so is making that data easy for people to find. Joe Cronin from Verity has written an article introducing you to the power of the Verity search engine and ways to extend it in order to make it more powerful.

In addition to articles focusing on presenting data, Phil Cruz has written part one of a series of articles examining using BlueDragon to create very portable CFML applications. How portable? How about installing and running them from a desktop icon or even running them off of a CD? Yes, THAT portable!

This month also introduces a new regular column, **CF At Work**. Each month, this column will focus on a case study of a company that is using ColdFusion to provide solutions, produce software, and/or just make life easier in general. Our first installment comes from Joe Zanter who works in a metallurgical lab. What is a metallurgical lab, and why would it need ColdFusion? Read his article to find out more!

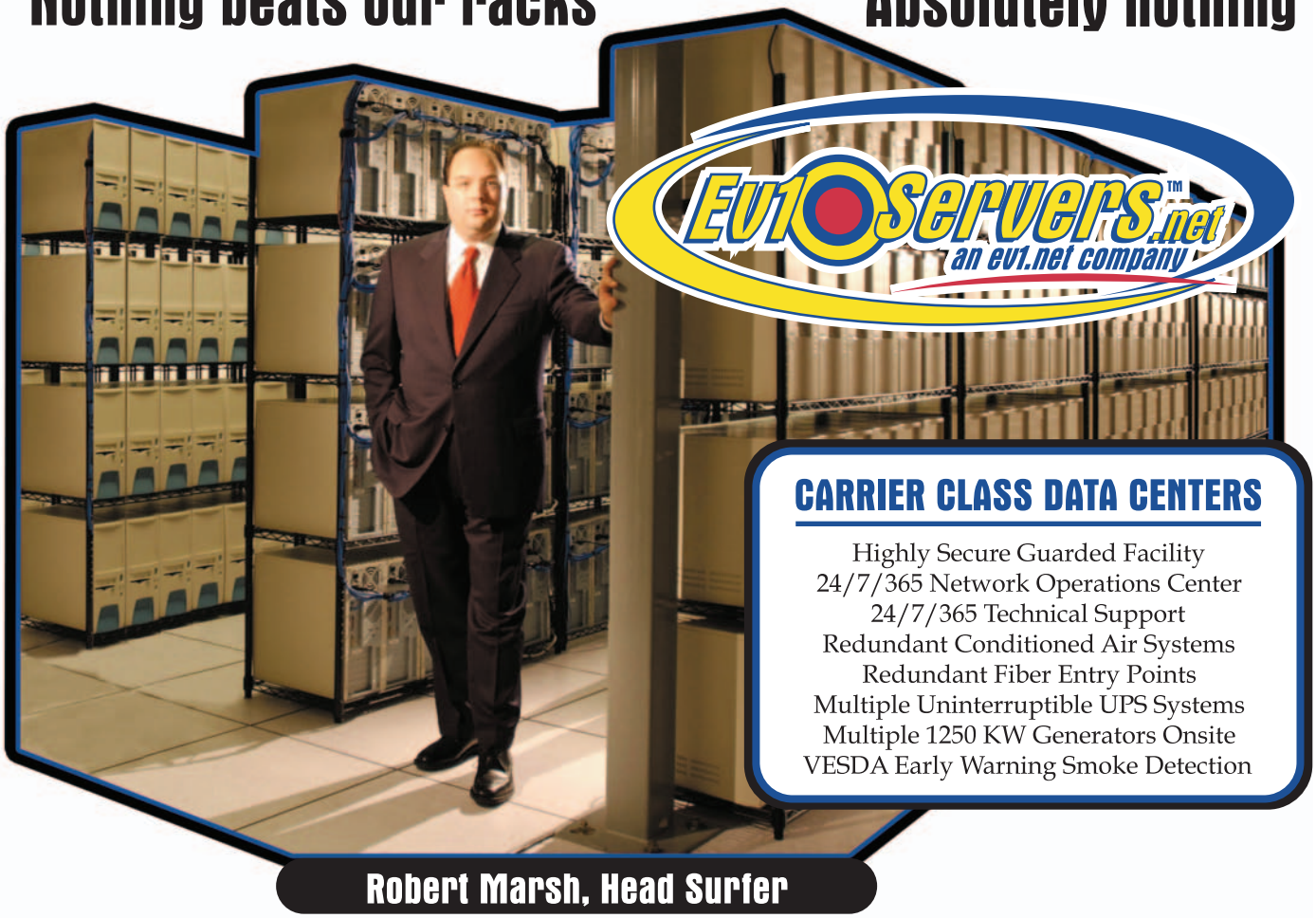
Next month we'll focus on IDEs. I've received many e-mails asking for a regular CF Administration column – so look for that very soon. All this and more await you in the pages that follow and in the months ahead!

About the Author

Simon Horwith is the editor-in-chief of ColdFusion Developer's Journal and is the CIO at AboutWeb, LLC, a Washington, DC based company specializing in staff augmentation, consulting, and training. Simon is a Macromedia Certified Master Instructor and is a member of Team Macromedia. He has been using ColdFusion since version 1.5 and specializes in ColdFusion application architecture, including architecting applications that integrate with Java, Flash, Flex, and a myriad of other technologies. In addition to presenting at CFUGs and conferences around the world, he has also been a contributing author of several books and technical papers. You can read his blog at www.horwith.com.
simon@horwith.com

Nothing beats our racks

Absolutely nothing



Robert Marsh, Head Surfer

CARRIER CLASS DATA CENTERS

Highly Secure Guarded Facility
24/7/365 Network Operations Center
24/7/365 Technical Support
Redundant Conditioned Air Systems
Redundant Fiber Entry Points
Multiple Uninterruptible UPS Systems
Multiple 1250 KW Generators Onsite
VESDA Early Warning Smoke Detection

START YOUR OWN WEB HOSTING BUSINESS TODAY!

from **$299*** *Instant Activation!*

Dedicated Server

Dual Xeon 2.4 GHz
2 GB RAM • 2 x 73 GB SCSI HD
Remote Console • Remote Reboot
2000 GB Monthly Transfer Included

Over 20,000 Servers!

1-800-504-SURF | ev1servers.net

PLESK7 RELOADED
Preferred Control Panel

IP Compliant. Price subject to change. Quantities Limited.
*Per month. Set-Up fees apply. See web site for complete details.

SYS-CON MEDIA

president & ceo
Fuat Kircaali fuat@sys-con.com

vp, business development
Grisha Davida grisha@sys-con.com

group publisher
Jeremy Geelan jeremy@sys-con.com

advertising

senior vp, sales & marketing
Carmen Gonzalez carmen@sys-con.com

vp, sales & marketing
Miles Silverman miles@sys-con.com

advertising director
Robyn Forma robyn@sys-con.com

advertising manager
Megan Mussa megan@sys-con.com

sales & marketing director
Dennis Leavey dennis@sys-con.com

associate sales managers
Kristin Kuhnle kristin@sys-con.com
Dorothy Gil dorothy@sys-con.com
Kim Hughes kim@sys-con.com

sys-con events

president, events
Grisha Davida grisha@sys-con.com

national sales manager
Jim Hanchrow jimh@sys-con.com

customer relations

circulation service coordinators
Edna Earle Russell edna@sys-con.com
Linda Lipton linda@sys-con.com
Monique Floyd monique@sys-con.com

manager, jdj store
Brunilda Staropoli bruni@sys-con.com

sys-con.com

vp, information systems
Robert Diamond robert@sys-con.com

web designers
Stephen Kilmurray stephen@sys-con.com
Matthew Pollotta matthew@sys-con.com

online editor
Martin Wezdecki martin@sys-con.com

accounting

financial analyst
Joan LaRose joan@sys-con.com

accounts payable
Betty White betty@sys-con.com

credits & accounts receivable
Stephen Michelin smichelin@sys-con.com

subscriptions

Subscribe@sys-con.com
Call 1-888-303-5282

For subscriptions and requests for bulk orders, please send your letters to Subscription Department
Cover Price: $8.99/issue
Domestic: $89.99/yr (12 issues)
Canada/Mexico: $99.99/yr
All other countries $129.99/yr
(U.S. Banks or Money Orders)
Back issues: $12 U.S. $15 all others

Playing with Numbers

It's all in the math

By Simon Horwith

For this month's installment, I thought I'd return the **Community** column to its original roots ("Tales From the List") and focus on a recent thread in the *CFDJ List*. Brian Simpson wrote to the List posing the following problem: "How do I round to the nearest 'x'?" Specifically, Brian has a client that wants to be able to use its e-commerce application to reduce all of their product prices by 10% and then round up to the nearest number evenly divisible by 50. Before you read on, ask yourself how you would achieve this task. It's not as easy as you'd think…or is it?

The first response Brian received was from Evik. He suggested removing any decimal values (cents) and then treating the number as a string. If the last two characters have a numeric value in the range from 1–50, then make the last two characters 50. If the last two characters have a numeric value in the range 51–99, then make the last two numbers 00 and then add 100 to the number. The code, written in a <CFSCRIPT> function, to do this would look like the following:

```
<cfscript>
   function roundToNext50(theNum){
      var retVal = 0;
      var someNum = 0;

      if (len(theNum) is 1){
         retVal = 50;
      } else {
         someNum = val(right(theNum, 2));
      }

      if (not retVal){
         if (someNum gt 50){
            if (len(theNum) is 2){
               retVal = 100;
            } else {
               retVal = 100 + val(left(theNum,
len(theNum) - 2) & "00");
            }
         } else {
            if (right(theNum, 2)){
               retVal = val(left(theNum,
len(theNum) - 2) & "50");
            } else {
               retVal = theNum;
            }
         }
      }
      return retVal;
   }
</cfscript>
```

This is not terribly pretty code – I certainly wouldn't want to have to make this function flexible enough to support rounding up to a dynamic increment. That said, it will round any number not already divisible by 50 to the next number that is divisible by 50. There are plenty of other ways to write the code to treat a number as a string and round it to the next 50 – this is just one of them.

A regular on the *CFDJ List*, Dave (from TheaterMania), then responded with an elegant solution that surprised everyone. He simply states, "What I do is divide by the rounding increment, round up to the nearest integer, then multiply by the rounding increment." So you divide your number by 50, use ceiling to round it up, then multiply by 50. Presto! You have your next number divisible by 50. The code for this, written in a <CFSCRIPT> function, would look like this:

```
<cfscript>
   function roundToNext50(theNum){
      return ceiling(theNum/50)*50;
   }
</cfscript>
```

continued on page 28

About the Author

Simon Horwith is the editor-in-chief of ColdFusion Developer's Journal. Simon is a Macromedia Certified Master Instructor and a member of Team Macromedia. He has also been a contributing author of several books and technical papers. You can read his blog at www.horwith.com
simon@horwith.com

Optimize, Extend and Enhance the Search Functionality in ColdFusion MX

Adding value quickly and easily

By Joe Cronin

Do your end users complain about the quality of your ColdFusion application's search functionality?

Have you exceeded the 250,000-document limit of the search functionality that comes embedded in ColdFusion MX? Have the requirements of your application changed to include searching for content both inside and outside of the ColdFusion environment?

If you answered yes to any of the above questions, it sounds like your ColdFusion MX application could use some help. In surveys by industry analysts, such as Gartner, IDC, and Forrester, end users, executives, and developers alike consistently rank the ability to search as one of the most important features of all online applications. Yet search continues to be one of the most maligned utilities of Web sites, business applications, and you name it. The bottom line is that if your end users can't find what they're looking for with your application's search tools, you're not realizing the full value of your ColdFusion investment.

Early on, Allaire, and then Macromedia, understood the need to provide ColdFusion developers with the ability to integrate advanced search features into their applications. Since 1997, the search functionality embedded in ColdFusion has been provided by Verity, Inc. Verity was selected for a number of reasons: ease of integration into ColdFusion, advanced functionality, and Verity's recognized position as the market leader in the enterprise search space.

But even the best search tool is only as good as its implementation. This article includes tips on optimizing the Verity search included in ColdFusion MX 6.1. With this search, you can build applications with advanced, enterprise-class full-text search of up to 250,000 documents and/or database records within the ColdFusion environment (if you need a larger search, go directly to the end of this article for more information). In addition, this article describes how to quickly and easily add value to applications by enhancing the search within ColdFusion, and by extending search to content outside ColdFusion.

Search Within ColdFusion MX

To ensure that users of ColdFusion applications can find the specific information they need in databases and hundreds of file formats, Macromedia integrated Verity full-text search. This includes:

- Full-text search of all ColdFusion content
- The ability to search a wide range of document types, including HTML, binary documents, and database records
- The ability to limit search to specific groups, or collections of documents, in order to enable subject-specific searches
- Support for multiple languages, including most European and Asian languages
- Fielded search against index metadata

There is more to the Verity search embedded in ColdFusion than just a box into which users type a query. To deliver exceptional performance, advanced functionality, and high relevancy, Verity performs searches against Verity Collections, not against the actual documents and database records within the ColdFusion application. A Verity Collection is a special index created by the Verity "spider." The spider locates all the documents and databases that are to be made searchable and extracts the text and metadata within each document or record. It also extracts other information, such as document zone and field data, word proximity, and the physical file system address or URL. All of this information is then gathered together in the Verity Collection. Bringing all of this information together in one index and running searches against it, rather than having to locate and access the actual documents and databases each time a user searches for information, dramatically increases the speed and relevancy of ColdFusion's search capabilities. It also enables advanced features such as document summaries in results lists and the ability to limit searches to specific groups of documents.

One of the strengths of Verity search solutions is that they can be configured to meet specific business and technical objectives. To optimize search with the ColdFusion environment,

Macromedia implemented Verity to support content of the following basic data types:

- Text files such as HTML pages and CFML pages
- Binary documents
- Record sets returned from cfquery, cfldap, and cfpop queries

As a developer, you can build Verity Collections from individual documents or from an entire directory tree. Collections can be stored anywhere, so you have greater flexibility in accessing indexed data when building applications.

Typical Applications of Verity Search

By taking advantage of Verity's flexibility you can add significant value to your ColdFusion applications. Typical uses of Verity search within ColdFusion include:

- Indexing the content of a Web site and providing a generalized search mechanism, such as the familiar search box
- Indexing specific directories that contain documents on a specific topic in order to provide subject-based searching, or to limit the focus of searches to specific groups of documents
- Indexing cfquery record sets into a single Verity Collection and letting users search against the collection with a single query rather than requiring them to perform multiple database queries to return the same data
- Indexing cfldap and cfpop query results
- Indexing e-mail generated by ColdFusion application pages and making the resulting Verity Collection available for searching from your ColdFusion application pages
- Building Verity Collections with inventory data and making those collections available for searching from your ColdFusion application pages
- Supporting international users in a range of languages, using the cfindex, cfcollection, and cfsearch tags

Advantages of Using Verity Search

One of the most obvious advantages of Verity search embedded in ColdFusion is its performance. For example, using Verity to index the output from database queries and then to perform searches against the indexed record sets is much faster than using SQL to search databases directly. Additional advantages of Verity over other search methods include:

- Superior relevancy of search results lists
- The display of document titles and summaries in search results lists
- Elimination of the need to programmatically create query constructs by allowing novice and expert users alike to use the same type of full-text queries they're used to using on the Web
- Indexing of database text fields, such as notes and product descriptions, that cannot be effectively indexed by native database tools
- The indexing and display of document URLs in results lists, a valuable document management feature

Implementing Verity Search in ColdFusion Applications

The good news is that Verity's advanced search features are straightforward to deploy within ColdFusion MX. In general, adding optimized osearch to your application involves three basic tasks:

1. Creating a Verity Collection
2. Indexing the content within your ColdFusion application
3. Designing a search interface

Each task can be performed programmatically – that is, by writing CFML code. Alternatively, you can use the ColdFusion MX Administrator to create a Verity Collection and index the content within your application. Also, Macromedia HomeSite+ has a Verity wizard, which generates ColdFusion pages that index content and design search interfaces. Table 1 summarizes the methods available for all three tasks.

There are pros and cons to using either the ColdFusion MX Administrator or CFML for deploying Verity search within ColdFusion applications. Refer to Table 2 to determine which is appropriate for your application and information environment.

Just as there is more than one method for deploying Verity search, you can configure your search implementation to meet specific business objectives. Primarily, you do this by running cfsearch or cfquery. Table 3 lists the advantages and uses of each.

Optimizing Search Relevancy

Once you've deployed the Verity search, you can also optimize its relevancy for your specific information environment. The ColdFusion implementation of Verity Query Language (VQL) uses operators and modifiers. These can either be used directly by advanced users, or implemented transparently so that they are applied automatically to all queries. The following are some of the more commonly used VQL operators:

Step	CFML	ColdFusion MX Administrator	Verity Wizard
Creating a collection	Yes	Yes	Yes
Indexing a collection	Yes	Yes	Yes
Designing a search interface	Yes	No	Yes

Table 1: Methods for deploying Verity search within ColdFusion applications

ColdFusion MX Administrator	CFML
To index documents	To index ColdFusion query results
When content changes frequently	When indices must be updated frequently
When writing CFML code is undesirable	To dynamically update a Verity Collection from a ColdFusion application page
To create a Verity Collection once	When updating the Verity Collection is done by others
Servers are accessible	To create an index on an inaccessible server
When index optimization is not necessary	When optimization is required

Table 2: ColdFusion MX Administrator versus CFML

Cfquery	cfsearch
Searches a data source	Searches a collection
Requires name attribute	Requires name attribute
Uses SQL statements to specify search criteria	Uses a criteria attribute to specify search criteria
Returns variables keyed to database table field names	Returns a unique set of variables
Uses cfoutput to display query results	Uses cfoutput to display search results

Table 3: cfquery vs. cfsearch

search functionality

- ***Evidence operators:*** Evidence operators specify either a basic word search or an intelligent word search. A basic word search finds documents that contain only the word or words specified in the query. An intelligent word search expands the query terms to create an expanded word list so that the search returns documents that contain variations of the query terms. Documents retrieved using evidence operators are not relevance-ranked unless you include the MANY modifier.
- ***Soundex:*** Expands the search to include the specified word and one or more words that sound like, or whose letter pattern is similar to, the word. Example: <SOUNDEX> sale returns documents that include words such as "sell," "seal," "shell," and "scale."
- ***Stem:*** Expands the search to include the word entered plus its linguistic variations. Example: <STEM> film returns documents that include words such as "films," "filmed," and "filming."
- ***Thesaurus:*** Expands the search to include the word entered plus similar words. Example: <THESAURUS> altitude returns documents that include words such as "height" and "elevation."
- ***Typo/n:*** Expands the search to include the specified word plus words that are similar. The optional n variable specifies the maximum number of errors between the query term and matched terms. Example: <TYPO> mouse returns documents that include words such as "house," "louse," and "moose."
- ***Wildcard:*** Matches wildcard characters included in the search string. Example: <WILDCARD> corp* returns documents that include words such as "corporation," "corporal," and "corpulent."
- ***Word:*** Performs a basic word search, selecting documents that contain one or more occurrences of the word. Example: <WORD> rhetoric will not match "rhetorical" or "rhetorician."
- ***Concept operators:*** Concept operators combine the meaning of search elements to identify a concept in a document. Documents retrieved using concept operators are relevance ranked.
- ***And:*** Selects documents that contain all of the search words specified. Example: german shepherd <AND> irish wolfhound returns only documents that contain the phrases "german shepherd" and "irish wolfhound".
- ***Or:*** Selects documents that include at least one of the search elements specified. Example: computers <OR> laptops returns documents that contain either "computers" or "laptops," or both "computers" and "laptops," but does not necessarily give a document that contains both terms a higher rank.
- ***Proximity operators:*** Proximity operators specify the relative location of specific words in the document. In the case of the Near/N operator, retrieved documents are relevance-ranked based on the proximity of the specified words. When proximity operators are nested, use the ones with the broadest scope first.
- ***In:*** Selects documents that contain specified values in one or more document zones. Example: "environmental regulation" <IN> summary returns documents that contain the phrase "environmental regulation" in the document summary.
- ***Near/n:*** Selects documents that contain two or more words within n number of words of each other. N is optional. Example: apple <NEAR/1> computer returns documents that contain the phrases "apple computer" or "computer apple."
- ***Paragraph:*** Selects documents that include all of the search elements you specify within a paragraph. Example: drug <PARAGRAPH> "cancer treatment" returns documents that contain "drug" and "cancer treatment" in the same paragraph.
- ***Phrase:*** Selects documents that contain the specified phrase. Example: <PHRASE> (twenty, years, ago, today) returns documents that contain the phrase "twenty years ago today."
- ***Sentence:*** Selects documents that include all of the specified words within a single sentence. Example: american <SENTENCE> innovation returns documents that contain "american" and "innovation" within the same sentence.
- ***Score operators:*** Score operators affect how scores are calculated for retrieved documents. When a score operator is used, the search engine first calculates a separate score for each search element found in a document, then performs a mathematical operation on the individual element scores to arrive at the final score for each document. The YESNO operator has wide application, whereas the PRODUCT, SUM, and COMPLEMENT operators are intended for use mainly by application developers who want to generate queries programmatically.
- ***Complement:*** Calculates scores for documents matching a query by taking the complement (subtracting from 1) of the scores for the query's search elements. Example: <Word> computers. If the search scores 0.80, then <Complement> <Word> computers scores 0.20.
- ***Product:*** Calculates scores for documents matching a query by multiplying the scores for the query's search elements together. Example: <PRODUCT> ("computers," "laptops"). If a search on "computers" generated a score of 0.5 and a search on "laptops" generated a score of 0.75, the preceding search would produce a score of 0.375.
- ***Sum:*** Calculates scores for documents matching a query by adding together the scores for the query's search elements. Example: <SUM> ("computers," "laptops"). If a search on "computers" generated a score of 0.5 and a search on "laptops" generated a score of 0.2, the search would produce a score of 0.7. If a search on "computers" generated a score of 0.5 and a search on "laptops" generated a score of 0.75, the search would produce a score of 1.00 (the maximum).
- ***YesNo:*** Enables you to limit a search to only those documents matching a query, without the score of that query affecting the final scores of the documents. Example: <YesNo> ("Chloe"). If the retrieval result of the search on "Chloe" was 0.75, with the YesNo operator, the result would be 1; if the retrieval result is 0, it remains 0.
- ***Modifiers:*** Modifiers are used in conjunction with operators to change the behavior of the operator.
- ***Case:*** Performs a case-sensitive search. The search engine attempts to match the exact use of uppercase and lowercase letters provided in the query expression when a mixed case query is used. Example: <CASE> NeXt returns only the precise string "NeXt", and not

"next" or "Next."

- ***Many:*** Considers the density of search words when calculating relevance-ranked scores. This means that shorter documents with multiple occurrences of the search terms are ranked higher than larger documents with the same number of occurrences, because the relative density of the occurrences is greater in the shorter document.

COLDFUSION TAGS

ColdFusion Search–Specific Tags

Creating a Collection with the cfcollection tag

When using the cfcollection tag, you can specify the same attributes as in the ColdFusion MX Administrator:

- Action: (Optional) The action to perform on the collection (create, delete, repair, or optimize). The default value for the action attribute is list. For more information, see cfcollection in CFML Reference.
- Collection: The name of the new collection, or the name of a collection on which you will perform an action.
- Path: The location for the Verity collection.
- Language: (Optional) The language used to create the collection (English, by default).

You can create a collection by directly assigning a value to the name attribute of the cfcollection tag, as shown in the following code:

```
<cfcollection action = "create"
collection = "a_new_collection"
path = "c:\CFusionMX\verity\collections\">
```

Indexing a Collection Using the cfindex tag

You can index a collection in CFML using the cfindex tag, which eliminates the need to use the ColdFusion MX Administrator.

- Collection: The name of the collection. If you are indexing an external collection (external = "Yes"), you must also specify the fully qualified path for the collection.
- Action: (Optional) Can be update (the default action), delete, purge, or refresh.
- Extensions: (Optional) The delimited list of file extensions that ColdFusion uses to index files if type="Path".
- Key: (Optional) The path containing the files you are indexing if type="path".
- URLpath: (Optional) The URL path for files if type="file" and type="path". When the collection is searched with cfsearch, the pathname is automatically prefixed to filenames and returned as the URL attribute.
- Recurse: (Optional) Yes or No. Yes specifies, if type = "Path", that directories below the path specified in the key attribute are included in the indexing operation.
- Language: (Optional) The language of the collection. English is the default.

Using the cfsearch tag

You use the cfsearch tag to search an indexed collection. Searching a Verity Collection is similar to a standard ColdFusion query: both use a dedicated ColdFusion tag that requires a name attribute for their searches.

The following are important attributes for the cfsearch tag:

- Name: The name of the search query.
- Collection: The name of the collection(s) being searched. Use a fully qualified path for an external collection. Separate multiple collections with a comma; for example, collection = "sprocket_docs,CodeColl".
- Criteria: The search target (can be dynamic).

Each cfsearch returns variables that provide the following information about the search:

- RecordCount: The total number of records returned by the search.
- CurrentRow: The current row of the record set being processed by cfoutput.
- RecordsSearched: The total number of records in the index that were searched. If no records were returned in the search, this property returns a null value.

Note: To use cfsearch to search a Verity K2 Server collection, the collection attribute must be the collection's unique alias name as defined in the k2server.ini and the external attribute must be "No" (the default). For more detail, see Configuring and Administering ColdFusion MX.

- ***Not:*** Excludes documents that contain the words or phrases indicated. Example: apple <AND> Mac <AND> <NOT> Washington returns information about Apple computers, but not Washington apples.
- ***Order:*** Specifies the order in which search elements must occur in the document. Example: <ORDER> <PARAGRAPH> (cat, chases, dog) is more likely to return documents that refer to cats chasing dogs, than dogs chasing cats.

Additional Resources

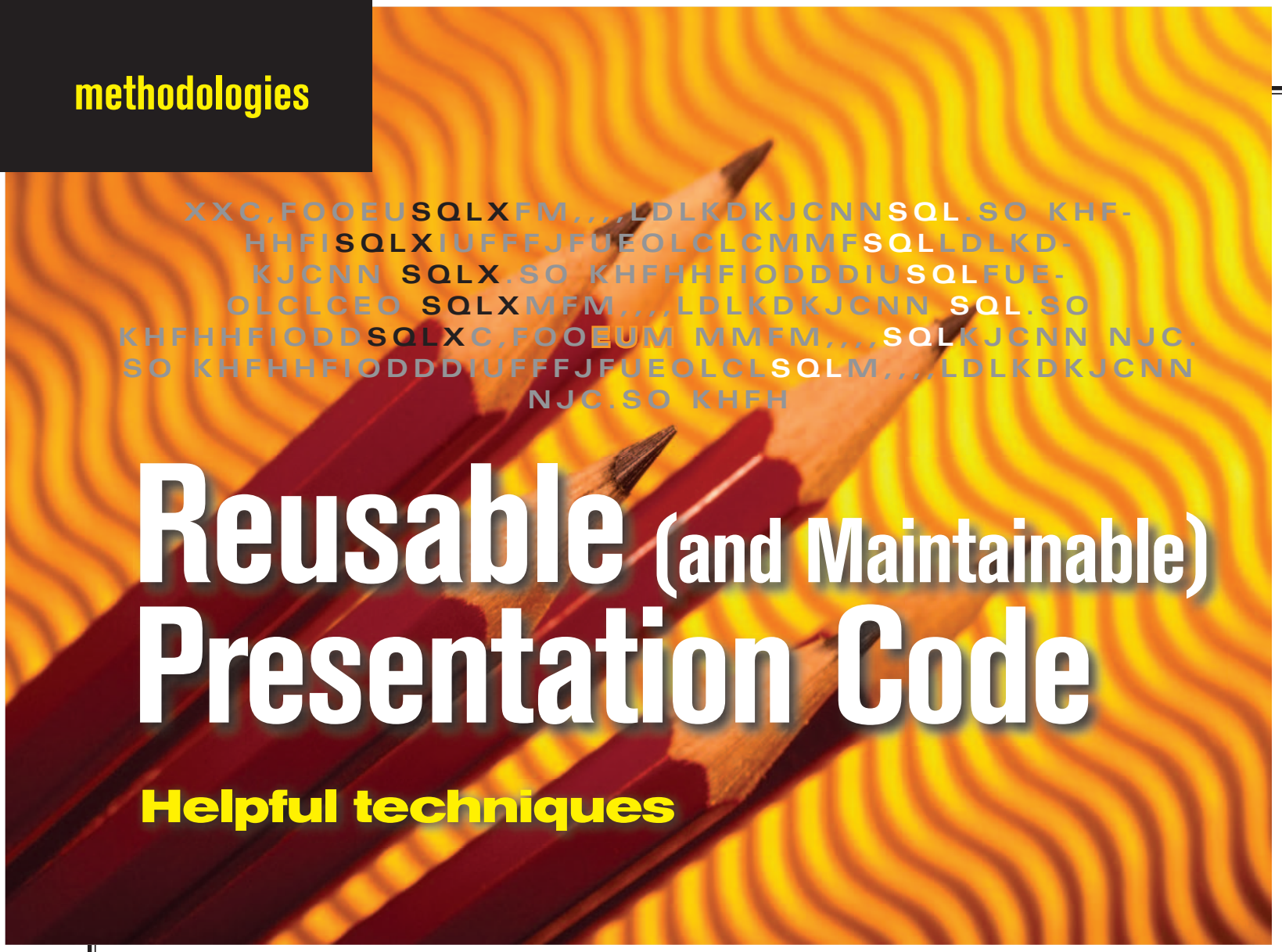
For additional resources, see the ColdFusion MX documentation available at www.macromedia.com. All product questions and support for ColdFusion, including the Verity search integrated into ColdFusion MX, are provided by Macromedia.

Extending and Enhancing the Search in Macromedia ColdFusion MX

As advanced as the Verity search is within ColdFusion, you may eventually want to deploy ColdFusion applications with enhanced search capabilities that are not possible using Macromedia's implementation of Verity. This is not due to any inherent limitations on Verity's part as much as it is a result of the robust, advanced capabilities of ColdFusion that enable you to develop ever more powerful applications. Verity has found that the need for additional search features is largely driven by three key requirements, along with a less common fourth requirement:

1. *Searching content outside of ColdFusion*. The Verity search functionality within ColdFusion is limited to searching content and database records within your ColdFusion application. Many ColdFusion developers need to extend the search functionality to areas outside of ColdFusion, such as intranets, external Web sites, file servers, external databases, Microsoft Exchange, and third-party document management systems.
2. *Additional Administrative Tools*. To meet their specific application requirements, some developers require additional administrative tools.

– continued on page 28



Reusable (and Maintainable) Presentation Code

Helpful techniques

By Steve Bryant

Everyone knows that you should reuse your code so that you don't have to repeatedly write the same functionality. You put widgets in custom tags and encapsulate logic in CFCs. Even so, your application's user interface may frequently end up changing. Sometimes the changes are so substantial that it hardly seems worth the effort to try to reuse code at all. Your efforts to reuse code and separate business logic from presentation aren't worthwhile unless you can write your presentation code so that it will be easy to change to fit changes to your clients' needs or other business requirement changes.

In this article, I will try to show a few techniques that I have found helpful in writing reusable code for application user interfaces. I'll use an example from a fictional search-reporting tool that was created in Dreamweaver and modified in HomeSite+.

Separate Logic and Presentation Code

If you look at Listing 1, the first thing that you might notice is that within the output loop I am querying the database for more information about each search term. This presents a number of problems, including the potential for very poor performance. The more pertinent problem, however, is that it clutters up the presentation code; making changes to this code is more confusing and tiresome (especially in cases where there is more logic code inside the output loop than is shown here).

The query logic (executed on each loop pass) can be pulled into the original query (see the SQL sidebar for more details). Pulling the logic into the original query improves performance (due to fewer connections to the database) and improves readability and maintainability of the code.

SQL

I would highly recommend that you learn and practice using

some of the basic SQL aggregate functions such as GROUP BY and OUTER JOIN if you aren't already comfortable with them. The GROUP BY keyword is used in Listing 2 because it will return the results of aggregate functions across multiple rows by the columns in the GROUP BY statement.

All of this is standard SQL and therefore should work on most databases. Ben Forta's *Teach Yourself SQL in 10 Minutes* (Sams Publishing) covers this well, as does *SQL for Dummies* (Wiley Publishing). You may not want to learn from a *Dummies* book, but I first learned SQL from this book (after being introduced to it in Ben Forta's *ColdFusion Web Application Construction Kit*) and I think it is pretty good. Either book is a good choice – just remember that while the aforementioned CFWACK includes a good introduction to SQL, your life will be made easier by learning more.

Function	On Numeric Columns	On Date Columns	On Character Columns
COUNT(*)	Counts the number of rows in the table	(same)	(same)
COUNT(column)	Counts the number of values found in the column	(same)	(same)
SUM(column)	Adds the total of the values found in a column	(not allowed)	(not allowed)
AVG(column)	Computes the average of values found in a column	(not allowed)	(not allowed)
MIN(column)	Finds the smallest value	Finds the earliest date	Finds the first value in a column (alphabetically)
MAX(column)	Finds the largest value	Finds the latest date	Finds the last value in a column (alphabetically)
From ColdFusion Web Application Construction Kit, Fifth Edition			

Table 1: Standard aggregate functions

Table 1 shows some standard aggregate functions. Since you are moving logic out of the output loop, look at this line:

```
<cfset PercentTotal = (numSearches / sumSearches) * 100>
```

This line also represents logic performed in the output loop. It is only one line so it probably isn't a major problem to leave it in the output loop. Even so, your life will be made easier in the long run if all of your logic is performed outside of the output. In this case, you can perform this calculation directly within the query itself – yes that's right, let the database do the work for you. (What if you couldn't? – see the Query Functions sidebar.)

As you can see from looking at Listing 2 (see sys-con.com/coldfusion/sourcec.cfm), the changes made the code much easier to read and, therefore, easier to maintain and modify.

Query Functions

In this case, the percent calculation was easy to perform in SQL. What if it weren't? Let's pretend that you couldn't perform that calculation in SQL. If that were the case, you would use the code in Listing 3 (see sys-con.com/coldfusion/sourcec.cfm).

As you can see, I created the column with a blank value in the SQL code itself. Immediately after the query, I looped through the results and set the value of that column to the result of the equation. This can be a very handy approach in situations where you have logic that cannot be performed in your SQL code but want to represent it within recordset rows.

Table 2 shows three built-in CFML query functions that I frequently find useful:

Function	Result (This is a table.)
QueryNew(columnlist)	Creates a query recordset with the given columns.
QueryAddRow(query[, number])	Adds a new row to the given query
QuerySetCell(query, column_name, value[, row_number])	Sets the value of one cell in the query

Table 2: Built in CFML query functions

Separate Style from Output

Since the separation of logic from presentation is done, future listings won't include the logic portion of the code (as that won't be changing from previous listings).

Now the code is easier to read and maintain. The gray and teal color scheme fits in with the look of the (fictional) site for which this application is designed. However, it won't fit with the color scheme of every site on which you work (and may not work on the original site if and when it undergoes an eventual redesign).

You want to minimize the effort to import this application into another site and adopt that site's look and feel. If this were the only page of the application, it would be easy to change, but in all likelihood the application has more pages than this one. What can you do to make the transition easier?

Right now, all of the styling is intermixed with output code. If you want to change the look of anything, you have to go through every page of output and make the changes throughout all of the output code. To avoid such tedium, use CSS (cascading style sheets) to move the styling away from the display content itself.

Looking at Listing 4 (see sys-con.com/coldfusion/sourcec.cfm), you will see that CSS is used to set the styles outside of the output code (although not shown in the example, you could use an external style sheet – see the External Style Sheets sidebar). This centralizes a place to make changes (especially if multiple pages use the same styles) and eliminates the need for some extraneous HTML, thereby making the code easier to read and maintain. Another thing that I like about moving styles outside of output code is that it removes the need to escape your "#" in your color definitions – which used to be the cause of one of the most common errors that I would get during development. Sometimes you can also achieve display and layout traits that cannot be achieved without the use of CSS.

External Style Sheets

You can make your life easier and your load times faster if you move all of your styles to external style sheets. In order to do so, just move all of the code between the <style> tags (not including any HTML comments) into a new file and save it with a .css extension.

You have two major options for referencing your external style sheet from your page. Both options involve code that should be placed in the <head> section of your page.

```
<link rel="stylesheet" type="text/css" href="/all.css" />
```

This is the option that I use most frequently. It is widely sup-

ported even by older browsers and I find it easy to read.

```
<style type="text/css">@import url(/all.css);</style>
```

This option is also very good and widely supported by most (if not all) modern browsers. You can also use this same syntax (@import url(filename);) at the top of an external style sheet to import another style sheet.

This approach can be very handy for code reuse. Each application within a site can have its own style sheet. Each page of the site can import a site style sheet, which in turn imports style sheets for each application.

Using Descendant Selectors

My code still has a few remaining problems that aren't – perhaps – as easy to spot. One problem is that the background color of the row (not the cell) is set and not all browsers will apply that background color to each cell in the row like you want. Another is that if something else on the page is using one of the classes or tags that you are styling, it will affect those elements as well – with possibly unpleasant effects.

Fortunately, descendant selectors come to the rescue on both counts. Descendant selectors are not the same as the CSS cascade, rather they allow you to specify which elements to style based on how tags are nested within each other. As you can see in Listing 5 (see sys-con.com/coldfusion/sourcec.cfm), the table is surrounded with a div tag and id of searchapp. Then the styles are changed to take advantage of that change.

```
#searchapp th
```

This CSS means that the browser will apply the given styling to any th tag inside of any tag with an id of searchapp. Anything in this format will work the same way. Descendant selectors keep styles from being inadvertently applied where you don't want them. They also keep you from having to add lots of code to your page in order to apply styling correctly. In my opinion, descendant selectors are really what make CSS great.

```
#searchapp tr.odd td
```

As you can see, you can also nest your descendant selectors. This line applies the background color to any td tag below any tr tag that has a class of "odd." If you notice, a class of "even" is used as well, but apply styling isn't applied to it. This will leave it the default color. The nice thing about this approach is that you can easily style even rows later on if you want to do so.

Other Thoughts

When you are done with the modifications you can put all of your styles into one CSS file. I would recommend adding notes to that file to indicate styles that you have in your code, but are not explicitly styling in the CSS (so that you don't have to go searching through your code when you are ready to make a change). You could also use a variable for your mask for the DateFormat() function (and one for NumberFormat() and other formatting masks as well).

This approach has worked very well for me – I would love to hear how it works for you, as well as any ideas you might have. Hopefully, you will find this approach beneficial. Good luck!

About the Author

Steve Bryant is the owner of Bryant Web Consulting LLC (www.bryantwebconsulting.com) and teaches ColdFusion at Breakaway Interactive (www.breakawayinteractive.com). He earned his BA in philosophy at Oklahoma State University and still has no idea how that led to a career in Web development. He can be reached at <mailto:steve@bryantwebconsulting.com> steve@bryantwebconsulting.com.

Listing 1

```
<cfquery name="qAll" datasource="#request.maindsn#">
SELECT     Count(searchid) AS Total
FROM       Searches
</cfquery>
<cfquery name="qSearches" datasource="#request.maindsn#">
SELECT     DISTINCT Term
FROM       Searches
</cfquery>
<cfset sumSearches = qAll.Total>
<style type="text/css">
.style1 {font-family: Arial, Helvetica, sans-serif}
</style>

<table width="600" border="0" cellspacing="0" cellpadding="2">
  <tr bgcolor="#CCCCCC">
    <th scope="col"><span class="style1">Search Term</span></th>
    <th scope="col"><span class="style1">Number of Searches</span></th>
    <th scope="col"><span class="style1">Most Recent Search </span></th>
    <th scope="col"><span class="style1">Percent of Total </span></th>
  </tr>
<cfoutput query="qSearches">
  <cfquery name="qDetail" datasource="#request.maindsn#">
  SELECT Count(searchid) AS Count FROM Searches WHERE Term = '#Term#'
  </cfquery>
  <cfset numSearches = qDetail.Count>
  <cfquery name="qDetail2" datasource="#request.maindsn#">
  SELECT Max(WhenRan) AS Last FROM Searches WHERE Term = '#Term#'
  </cfquery>
  <cfset LastSearch = qDetail2.Last>
  <cfset PercentTotal = (numSearches / sumSearches) * 100>
  <tr<cfif CurrentRow MOD 2> bgcolor="##669999"</cfif>>
    <td><strong>#Term#</strong></td>
    <td><div align="right">#NumberFormat(numSearches)#</div></td>
    <td><div align="right">#DateFormat(LastSearch,"m/dd/yyyy")#</div></td>
    <td><div align="right">#NumberFormat(PercentTotal)#%</div></td>
  </tr>
</cfoutput>
</table>
```

Download the Code...
Go to www.coldfusionjournal.com

Seapine Software™
changing the world of software development

Complete source code and asset management in Dreamweaver MX—now possible with Surround SCM.

Dreamweaver users know a beautiful Web-based product is only skin deep. Underneath, it's a tangle of hundreds or thousands of ever changing source files. Without a good development process and strong tools, bad things happen. Surround SCM can help.

Surround SCM lets you...

Track multiple versions of your source files and easily compare and merge source code changes.

Check out files for exclusive use or work in private workspaces when collaborating on a team.

Automatically notify team members of changes to source files—push changes through your organization.

View complete audit trails of which files changed, why, and by whom.

Associate source code changes with feature requests, defects or change requests (requires additional purchase of TestTrack Pro).

Remotely access your source code repository from Dreamweaver MX.

Surround SCM adds flexible source code and digital asset control, powerful version control, and secure remote file access to Dreamweaver MX. Whether you are a team of one or one hundred, Surround SCM makes it easier to manage your source files, letting you focus on creating beautiful Web-based products.

Features:

Complete source code and digital asset control with private workspaces, automatic merging, role-based security and more.

IDE integration with Dreamweaver MX, JBuilder, Visual Studio, and other leading Web development tools.

Fast and secure remote access to your source files—work from anywhere.

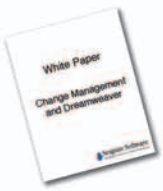
Advanced branching and email notifications put you in complete control of your process.

External application triggers let you integrate Surround SCM into your Web site and product development processes.

Support for comprehensive issue management with TestTrack Pro—link changes to change requests, bug reports, feature requests and more.

Scalable and reliable cross-platform, client/server solution supports Windows, Linux, Solaris, and Mac OS X.

Want to learn more?



Download Seapine's just-released white paper, **Change Management and Dreamweaver**, to discover tips, tricks and best practices for achieving full software configuration functionality. Visit www.seapine.com/whitepaper.php and enter code WM0604 to receive your copy today.

www.seapine.com
1-888-683-6456

Surround SCM Source Control
TestTrack Pro Defect Management
QA Wizard Automated Testing

macromedia ALLIANCE PARTNER

software development 13th annual product excellence award

Data Presentation with Macromedia Flex

Break free of the limits of HTML

By Matt Woodward

Last month I introduced you to the wonderful world of building Rich Internet Applications with Macromedia Flex ("Flex Your ColdFusion Muscles," CFDJ, Vol. 6, issue 12) and discussed some of the distinct advantages RIAs have over traditional HTML applications.

This month I'd like to go a bit deeper and discuss how Flex can give users a much better experience through its powerful interactive data presentation capabilities. Data presentation isn't always given a great deal of attention because we've become accustomed to doing things "the HTML way" and assume that's our only option. The reality is that Flex opens up exciting new possibilities for data presentation that are definitely worth a look.

Flex includes a vast range of building blocks that can be leveraged to solve even the most demanding data presentation problems. At a high level, Flex's UI toolkit consists of two basic elements: containers and components (some of which are also referred to as controls). Containers are used to create the layout and navigation of the application. Components include simple elements such as buttons as well as more complex elements such as data grids, date choosers, and charts. Let's take a moment to look at containers and components in a bit more detail, then we'll see how these elements are used with one another to solve three different data presentation challenges.

Containers

If you have a background in traditional desktop application development with technologies like Visual Basic, or if you've done something along the lines of Java Swing development, Flex containers will seem very familiar to you. If you have a more traditional Web development/design background you may not have had exposure to containers, but not to worry; containers are easy to understand and even easier to use.

In simple terms, containers are rectangular layout elements that contain other UI elements, hence the name "container." Containers have a great deal of built-in functionality and default behaviors to control aspects of the UI such as element positioning, layout, and resizing. You simply place elements within the container – these become the container's children – and Flex's layout manager monitors things behind the scenes, applying layout rules and behaviors as users interact with the application. Containers can even be nested within each other as needed to create more complex layouts.

The two types of containers available in Flex are layout con-

tainers and navigator containers. I won't address them all here, but some of the layout containers available in Flex are as follows:

- ***Application:*** Top-level container that contains the Flex application.
- ***Canvas:*** Elements within the canvas container are positioned using specific x,y coordinates. The Canvas' children are not governed by layout manager rules for positioning and resizing.
- ***HBox and VBox:*** Containers that display children horizontally or vertically respectively.
- ***HDividedBox and VDividedBox:*** Containers that display children either horizontally or vertically respectively, but also include a bar between each child so the width and height of each child may be adjusted by the user (similar to HTML frames).
- ***Form:*** Container used to automatically control the layout of form input elements. By default the form elements are left-aligned and arranged vertically. The form container also controls the size and spacing of the form elements.
- ***Tile:*** Container that arranges its children in a grid format either horizontally or vertically (direction is an attribute of the tile tag). Each cell within the grid is uniform in size and the number of total cells is calculated by the layout manager based on the number of children within the tile container.
- ***Grid:*** Similar to the tile container but the number of columns and rows may be specified, as opposed to being controlled by the layout manager. The grid container is very similar to an HTML table, complete with row and column span capabilities. Unlike the tile container, rows and columns may have varying widths and heights rather than each cell being identical in size.

For additional information on the types of layout containers available in Flex, please visit Macromedia's Flex documentation site at www.macromedia.com/support/documentation/en/flex/.

Navigator containers serve a more specific purpose than the basic layout containers and allow for the creation of extremely rich UIs. The basic concept behind navigator containers is that they are used to contain related groups of elements of which only one is displayed at any given time, and a control of some sort allows for selection of the various elements within the group. Element selection may also be accomplished programmatically using ActionScript.

The navigator containers in Flex are as follows:

- ***TabNavigator:*** A horizontal tabbed interface that is visually similar to file folders. Each tab contains title text and users simply click on the various tabs to navigate between elements in the group.
- ***MenuBar:*** A menu bar similar to that found in desktop applications. Can contain nested fly-out menus.
- ***Accordion:*** A collection of vertically-stacked panels, each with a clickable title bar that expands that particular section of the accordion interface. Often used for multistep forms.
- ***ViewStack:*** A collection of separate independent "screens" or views within the application. The ViewStack does not have built-in means of switching between views, so developers are free to use ActionScript to display different views or provide some type of menu to allow users to switch between views.
- ***LinkBar:*** A simple horizontal box containing links with a horizontal divider separating each link. The LinkBar contains a built-in control for the ViewStack navigator so it is often used to control a ViewStack.
- ***TabBar:*** A horizontal collection of tabs, but unlike the TabNavigator, the TabBar does not have built-in content areas. As with the LinkBar, the TabBar has a built-in control for the ViewStack navigator.

As you might imagine, using these containers in conjunction with one another offers developers the ability to create incredibly rich, user-friendly UIs that can handle even the most complex data presentation challenges.

Components

Components are the low-level building blocks of Flex applications and are part of what is placed inside Flex containers. As with containers, Flex provides an extremely broad, rich set of components from simple controls such as the Button, Link, ComboBox, and TextInput to more complex, full-featured components such as the DateChooser, DataGrid, or one of the numerous charting and graphing components that are new to Flex 1.5. As you'll see in a moment, when you start filling containers with Flex components the data presentation possibilities become extremely powerful.

Flex contains far too many components to list here, but following are some of the more commonly used components:

- ***Form Input Controls:*** Familiar form elements such as the TextInput, TextArea, Checkbox, and RadioButton are all available in Flex. Flex also adds form elements such as the NumericStepper, DateChooser, and DateField that are not available in HTML.
- ***List:*** Similar to an HTML select that allows multiple selections, but graphically richer and allows for data binding, runtime editing of options within the list, and the ability to include images within the List.
- ***ComboBox:*** A single-select version of the list component.
- ***HorizontalList:*** Displays a horizontal list of items with a scrollbar as needed; often used in conjunction with a custom cell

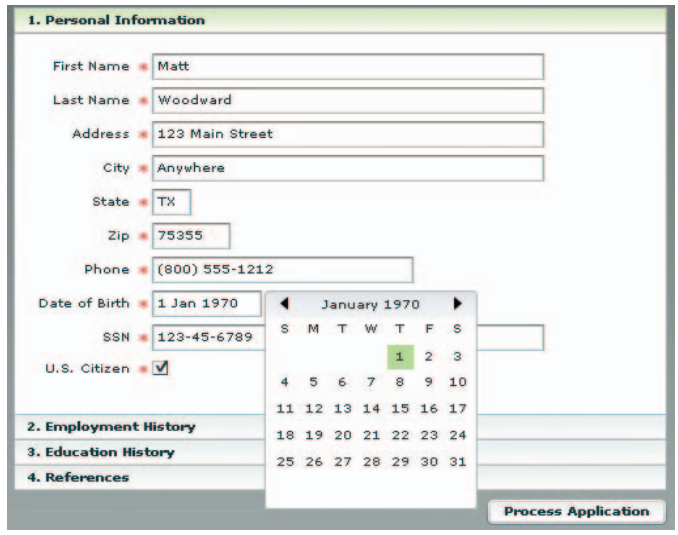


Figure 1: Multistep form using the Accordion Navigator

renderer to display images in combination with other data.

- ***TileList:*** Similar to the HorizontalList, but displays elements in columns and rows of cells that are uniform in size.
- ***Tree:*** Provides a means of displaying hierarchical data in the form of an expandable tree.
- ***DateChooser:*** Graphical calendar control that allows for navigation across months and years as well as the selection of a single date.
- ***DateField:*** Text field that displays a date and also has a calendar icon to the right of the text field. Clicking the calendar icon launches a DateChooser, and selection of a date from the DateChooser populates the text field with the selected date.
- ***DataGrid:*** An extremely rich component for presenting rows and columns of data. The Flex DataGrid has built-in functionality for sorting, highlighting, column resizing, cell editing, multiple selection modes, data paging, and with Flex 1.5 the DataGrid allows for variable row height, word wrap, and custom cell renderers for the inclusion of images and other components within the DataGrid cells.
- ***Charting Components:*** New to version 1.5, the types of charts available in Flex are the AreaChart, BarChart, BubbleChart, ColumnChart, LineChart, PieChart, and PlotChart, all of which allow for dynamic data drill-down and animated transitions.

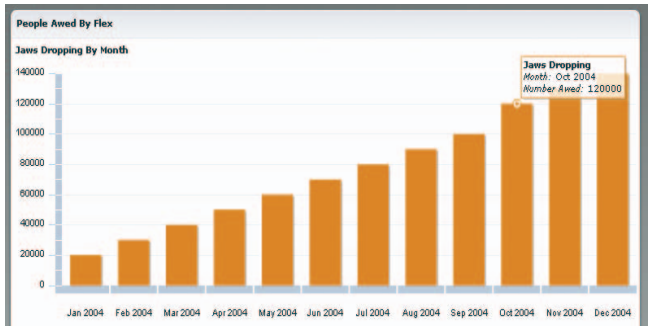


Figure 2: Flex column chart

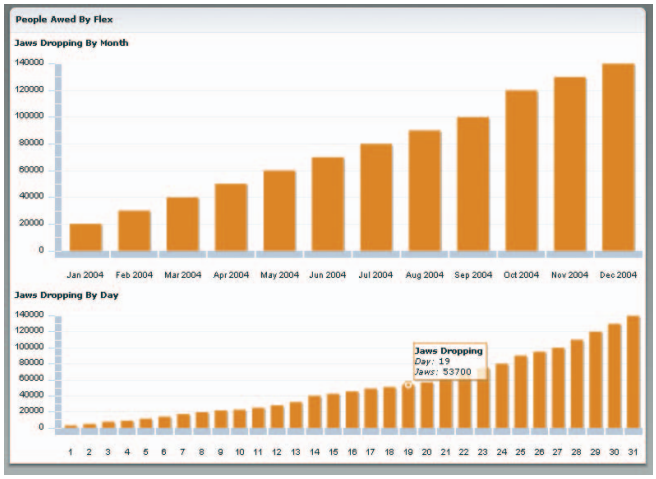


Figure 3: Data drill-down in column chart

Believe it or not, the possibilities don't end there! Many of these components (such as the DataGrid) also support drag-and-drop behavior and other advanced functionality that allows for the creation of extremely intuitive, rich UIs that users simply aren't used to seeing in their Web browsers.

Now that you know some of what's available in Flex, let's get down to business and start combining containers and components to show off Flex's data presentation capabilities.

Data Presentation Challenge #1: Multi step Data Entry Forms

I'm sure we've all had the frustrating experience of completing lengthy, multistep forms in Web applications, and building these types of forms in HTML isn't any more fun than using them. By combining the navigator containers and form elements in Flex, however, even the most complex forms become simple, clear, and almost fun to complete. (OK, completing a mortgage application is no one's idea of fun, but at least with Flex it'll look nice and be easy to use!)

Reviewing the navigator containers discussed previously, the two obvious choices for creating multistep forms are the TabNavigator and the accordion. Both choices work quite well, but for the purposes of this example we're going to take a look at the accordion navigator and how it can be used to present a typical job application form (see Figure 1).

By including all of the form steps within a single accordion navigator we are able to present a great deal of information in a limited amount of space, and we also remove the jarring page-refresh model of multistep HTML forms, something which causes many users to abandon form completion altogether. We can also use the accordion title bars to clearly label each step with a description and number so users know exactly where they are in the process.

This example also serves as an excellent illustration of the form container and some of Flex's form controls. A discussion of data validation is beyond the scope of this article, but as you can see in Listing 1 (can be downloaded from sys-con/coldfusion/sourcec.cfm), form elements can easily be flagged as required and Flex takes care of the visual indication to the user. Flex also includes advanced data validation for input such as e-mail addresses, credit card numbers, dates, phone numbers, and zip codes so the forms don't just look great, they're highly functional as well.

This is a relatively simple example (I left the forms beyond the first step for you to complete!), but it illustrates the vast improvement Flex offers for presenting complex forms and collecting data in a more user-friendly way. When the improved visual and data entry experience is combined with Flex's powerful validation capabilities and features (such as data binding which allows for cross-field relationships, auto-completion, and more), Flex is clearly adept at handling forms.

Data Presentation Challenge #2: Charting and Graphing

As ColdFusion users we've bragged for quite some time about ColdFusion's built-in charting and graphing capabilities, but as the saying goes, "you ain't seen nothing yet." One of the most requested enhancement features after the release of Flex

THE GRASS REALLY IS GREENER ON THE SERVERSIDE.

SUPERIOR MANAGED HOSTING

- Intelligent Routing
- Redundancy
- Network Security
- Service Level Agreement
- Environmental Controls
- Network Uptime Guarantee
- Scalability
- OC3/SONET Backbone
- Backup Power
- Physical Security
- Fire Protection
- Server Hardware Guarantee

IF YOU'VE BEEN SEARCHING for a reliable managed hosting partner, your search for greener pastures is over. There is no longer a need to settle for inferior support and lack of accountability. ServerSide takes the guess work, and the associated hassles, out of working with a technology partner.

ServerSide provides managed web hosting solutions for a wide range of customers including; Fortune 500 corporations, small and medium sized businesses and non-profit organizations located across the United States and around the World.

We provide a single point of accountability for all your web hosting infrastructure needs — while adding value, not cost.

>> SPECIAL OFFER

Sign up online for a new shared hosting account now at www.serverside.net and ServerSide will waive the set-up fee.

ENTER CODE: mxd2004

The grass is greener at
www.serverside.net
888.682.2544
hosting@serverside.net

serverside™
Dynamic Web Development. Superior Managed Hosting.

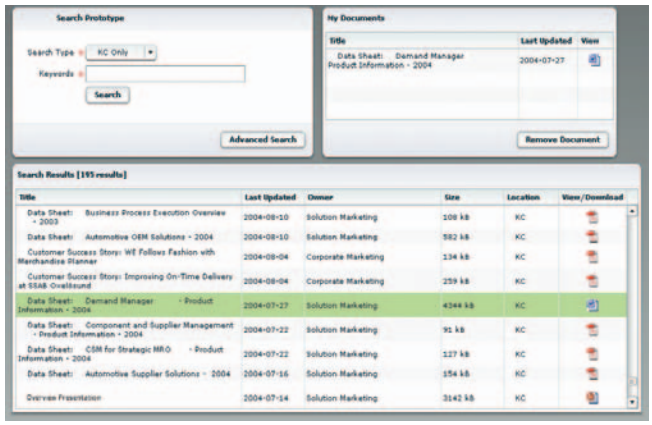


Figure 4: Search tool built with Flex

1.0 was for the addition of charting components. True to form Macromedia has created a solution that is as visually stunning as it is functional.

Using charts for data presentation is a topic that could fill an entire article on its own, so I'll take just a moment to at least introduce you to what's possible with Flex's new charting and graphing components (see Listing 2 at sys-con/coldfusion/sourcec.cfm). For simplicity's sake this example will use hard-coded data, but of course in a real-world application your data would likely be coming from a database or some other dynamic data source.

This is a nice-looking column chart and the mouse-over tool tips are definitely slick (see Figure 2). While it's true that beauty is only skin deep, Flex charts have both style and substance. I'm sure you've had as many requests as I have from users who want to be able to drill down into data and view it in different ways. With a few lines of code we can enable this functionality in Flex, and with a simple click on one of the bars in the chart, users can see a detailed view of that particular segment of the data.

Presenting data visually is always extremely compelling, and Flex takes this powerful method of data presentation to the next level by providing a tremendous amount of out-of-the-box functionality so we can not only present data visually to our users, but allow them to interact with it as well (see Figure 3). Charting functionality in Flex is incredibly rich so bear in mind that I am barely scratching the surface with this brief example.

Data Presentation Challenge #3: Search Applications

For our final data presentation challenge, let's consider a more complex scenario that combines several of the containers and components we've outlined into a single application. Web users have become quite accustomed to the behavior of search applications and the way in which search results are displayed in traditional HTML, but this isn't necessarily an ideal way to present this type of data. The page-based model is more or less a necessity with HTML, but by using Flex's DataGrid along with a few other components, not only can we create a search tool that is more intuitive and requires no page reloads as we retrieve and navigate through search results, but we can also allow users to interact with search result data in completely new ways.

When I first started working with Flex I built a relatively simple search tool prototype. While still in its early stages, this single-screen model for searching a knowledge management system will give users a much better experience compared with what is typically a rather complex HTML-based search tool. Best of all, it was surprisingly easy to build with Flex and leveraged existing ColdFusion code to tap into data collections. (Note that in search result images company names have been removed, which is the reason for the gaps in the DataGrid content.)

The use of multiple containers clearly defines the search form, search results, and "My Documents" sections of the UI (see Figure 4). A pop-up window is used to perform more advanced searches, which keeps the screen clutter free. In the search results panel, the DataGrid component allows users to sort on columns, which is possible to build into HTML-based applications but requires a page refresh each time a column is sorted, sing the Flex DataGrid users can also change the widths of the columns as desired, something that simply isn't possible with HTML. Even performing a new search repopulates the search results grid without requiring a page refresh.

The use of a custom cell renderer in the View/Download column makes it easy for users to see at a glance what type of document they will be viewing or downloading by clicking on the icon. Also, by using Tool Tips (new in Flex 1.5), a summary of the document can be presented to users when they mouse over the document title. This is an extremely powerful data presentation technique that allows for the maximum number of search results to be displayed in the available vertical space while not sacrificing the user's ability to read more about the document before download (see Figure 5).

Through Extremely compelling so far, this little prototype has one last trick up its sleeve. I haven't addressed the "My Documents" panel in the upper-right corner yet. You may recall my earlier mention of the DataGrid's built-in drag-and-drop capabilities. By tapping into this powerful feature of Flex, users have the ability to drag documents from the search results grid and drop them into the "My Documents" grid for future use (see Figure 6).

Because this is a relatively extensive application behind the scenes, I don't have space to show code for this particular example, but if I've inspired you to investigate further (and I hope I have!) you'll be surprised how easy it is to implement even advanced functionality such as drag-and-drop with Flex.

This is very powerful stuff, and we can take it a lot further. By extending these basic concepts with additional components like a Tree or

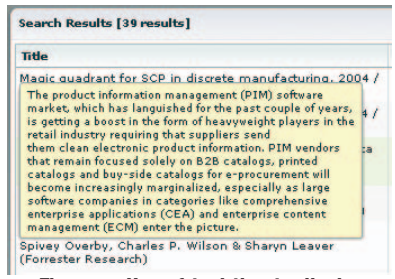


Figure 5: Use of tool tips to display document summaries

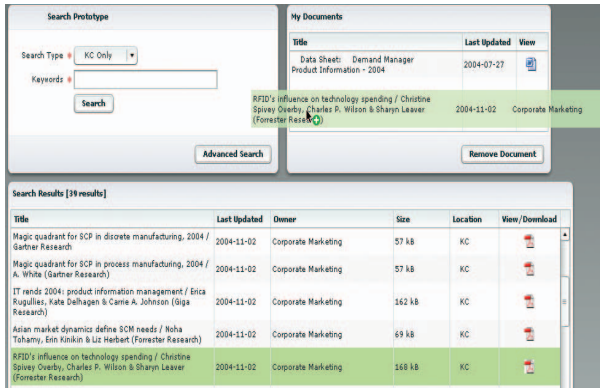


Figure 6: Drag-and-Drop from DataGrid

TabNavigator we can provide users with things such as multiple, tabbed search results and file folders for storing and sorting documents. This makes for an incredibly full-featured search tool that breaks the HTML model, adds a great deal of functionality, and presents data in better, more useful ways.

Conclusion

This was admittedly a bit of a whirlwind tour, but I hope it has opened your eyes to the fantastic data presentation possibilities offered by Flex. When the extremely capable containers and components available in Flex are used in combination with one another the data presentation capabilities are truly endless.

Resources

- Greenfield, Ely (2004). "Beyond the DataGrid Control: Better Data Visualization with Flex Charting Components." www.macromedia.com/devnet/flex/articles/charting.html
- Lally, Susan (2004). "Introducing Flex 1.5." www.macromedia.com/devnet/flex/articles/flex15_preview.html
- Macromedia, Inc. (2004). Flex 1.5 Documentation. www.macromedia.com/support/documentation/en/flex/
- Meeker, David and Carlos, Ryan (2004). "Delivering Rich Business Intelligence." www.macromedia.com/devnet/flex/articles/rich_intelligence.html
- Sundermeyer, Mike (2004). "Understanding the Macromedia Flex Experience Model." www.macromedia.com/devnet/flex/articles/halo.html

About the Author

Matt Woodward is a Web application developer for i2 Technologies in Dallas, Texas, and also works as a consultant for his company, Sixth Floor Software. He is a Macromedia Certified ColdFusion Developer, a member of Team Macromedia, and has been using ColdFusion since 1996. In addition to his ColdFusion work, Matt also develops in Java and PHP.

mpwoodward@mac.com

Personas Can Improve User Interface

Basing software flow on human flow

By Steve Nelson

For a long time I've been stuck in the mindset that creating a fabulous user interface requires a top-notch artist. I thought someone who is a master of colors and graphics is bound to make the ideal user interface. Trial and error has proved this incorrect.

I've done projects with and without fabulous artists. A good artist can improve the details of the interface to make it easier on the eye, but rarely does he or she make suggestions that dramatically improve how the software works. That's okay; that's not their job. But if they're not changing how the software works, then maybe colors and graphics aren't what make a user interface easy to use.

Maybe it's the programmers who make it easy to use. The programmers know the database, therefore they must be masters of data input fields and data display, right? Without the right data, the software is going to be useless. As most of us already know, programmers are generally pretty bad at making easy-to-use interfaces. It's probably because we're all left-brained creatures. Or it may be because most programmers are quick to try and save time by reusing prebuilt interface widgets that may not be ideal for the people using the software.

It's not the people on our team who make a good interface, it's the design process. As much as I don't like McDonald's, they've created a business that can be operated by uneducated teenagers. They have created an operation that focuses on ideal steps that just about anyone can do. While most of the people reading this article fall into the "programmer" category, it is possible for us to improve our interfaces by improving our design process.

> "Personas help us decide why one solution is better than another"

People Have a Name

The first step is to paint a picture of who is going to use your software. If you are building a business application, it's likely numerous people will use the software. We can't account for everyone so it's important to group people together. Most of us already do this, but the most important thing we can do is to stop referring to them as "users." They are people. Instead give them a name, gender, age, picture, and a short story. Paint their picture.

This is not a new concept; it is commonly used in the marketing world, and known as a *persona*. With software, a persona is a fictitious person using our system. Don't confuse personas with *UML actors*. The concepts are similar, but different on a broader level. *Use case actors* help us understand how a person fits into the flow of the software, whereas personas help us decide why one solution is better than another.

For example, Sally is 16 years old and a high-school dropout. She left school because she just couldn't seem to pass her math classes. Now she works for McDonald's as a cashier, earning minimum wage.

Even with only a few sentences we can paint a fairly dramatic picture of Sally, enough to help us rethink how the cash register software needs to work. McDonald's has a good understanding of Sally. For example, their cash registers do not require her to type in how much an item costs. Instead they have pictures of hamburgers, French fries, etc. But could they improve the cash register so Sally doesn't have to count the correct change? That may seem ludicrous to a programmer who does highly complex math every day, but remember Sally is not a mathematical genius.

Distinguish Tasks from Goals

When we're using software we're frequently completing tasks. Improving tasks does not immediately make software easy to use. For example, we don't go to Amazon.com just to enter our credit card information into their database. That's just a means to an end. Joe, an Amazon customer, uses Amazon.com to obtain a book to read. Reading the book is his goal, it is not a task. Reducing the tasks to achieve a goal immediately improves an interface. Amazon has done a good job by remembering Joe's information when he returns. Their one-click order system is the ultimate attempt at reducing tasks to achieve a goal.

Identify the Human Flow

After identifying the goals our personas are trying to reach, we want to define the human flow they will go through to get there. By human flow I mean the steps they take in day-to-day life. Some of the steps may involve your software, others may not. Each of these steps should have a date/time associated with it. Let's look at using a cellphone interface.

5:00 p.m.: Todd leaves his office to go home.
5:05 ***p.m.:*** While driving, Todd decides to call his wife to find out if he should pick up groceries for dinner.
5:08 ***p.m.:*** Todd hangs up.
5:15 ***p.m.:*** Todd drives to the store.
5:25 ***p.m.:*** Todd forgets all the items his wife asked for. He has to call her back to ask again.

Not all of the steps in the human flow involve the software. The human flow may leave out critical steps necessary for the software to work. Even so, the software flow should be based on the human flow. In the steps above, there are a few events that involved the cellphone. Understanding the human flow provides our user interface with the boundaries in which the interface needs to work. For example, can we improve making a phone call while driving a car?

Since Todd's eyes should be on the road and not on his phone, can we improve the audible interface for his phone? Since it's more likely he'll want to make a call while driving versus playing Tetris, there is little reason to include the games in the audible menu. How about helping him remember what was said during a conversation? Instead of calling his wife back and bothering her, couldn't the phone just record the conversation and play it back?

These boundaries may appear artificial at first, but if the personas defined accurately represent the real users they are based on "probable" issues, not "possible" issues. While it's possible to put a Ferrari engine in a station wagon and add some tractor wheels, it's probably not a good idea. How many soccer moms need to pick up their kids from soccer practice and plow a field, while doing 160 miles an hour? Our software should not attempt to solve every problem in the world. Instead we should focus on the probable issues and solve those. Personas help us identify these probable issues.

Only when we have identified why the people using our software are trying to accomplish certain goals can we effectively make decisions about how the user interface should be built. Personas provide us with a structure to improve our chances of making the correct decision about our software and its interface.

About the Author

Steve Nelson is a ColdFusion freelance consultant for SecretAgents, Inc. Steve invented the original Fusebox framework, which has become wildly successful. His company specializes in managing large teams of remote ColdFusion developers. Steve's Web site is

www.secretagents.com

ColdFusion Everywhere PART 1

Create applications that run on the Web, from CD, or on the desktop

By Phil Cruz

By Dick Applebaum

Think about the last time you bought a car. Did you walk around the vehicle, look under the hood and sit in the driver's seat to get a feel for it? You probably took it for a test-drive before making the final decision. If you're a developer producing the next "killer app," it's only reasonable for your customers to want to test-drive your application as well. Have you ever wanted to demo your application but thought…

- It's too difficult to put together a general-purpose demo.
- Even if you did publish a "live" demo site, it would be open to wrong-headed visitors entering offensive data, corrupting the demo for others.
- If you don't allow data entry, you limit the effectiveness of the demo.
- You could allow users to download your application (in some protected time-limited format), but they would need to install it and all the prerequisite programs (CFML server, database server, application server, Web server). They would also have to configure the application, populate the database, etc., etc., etc. This could take hours (or days).
- This is a daunting proposition even for the technically capable – who are likely not the people you want to see your demo. (If potential customers had to put together a Lexus before taking it for a test drive, they'd buy a BMW...)

Ideally, potential customers can experiment with your application at the time and place they choose, using their own data – with no unnecessary activities to deter or distract them, and no need to have access to the Internet.

What if you could deliver your application into the hands of customers in a package they could just double-click and run?

Well, you can! With readily available building blocks you can put together a self-contained package that holds your application and everything needed to run it on any platform you support. Your application appears as an icon on the desktop. You double-click and run… that's it! Nothing gets in the way!

In this article we'll show you a way to package and deploy your CF applications that is different from anything you've done before – making your application available as a single download (zip) file, or burning it onto a CD or DVD. If your application doesn't need to write files or update a database, it can be run directly from the CD. The objective is to enable you to run your CF application everywhere (the Web, a desktop, or CD) – we call the concept "CFEverywhere."

Examples of CFEverywhere Applications

CF has been used (almost) exclusively to develop applications for Web (and intranet) sites. With CFEverywhere, you are no longer limited to the Web dynamic. You can take advantage of the power of CF to write desktop applications.

Let's look at some examples of how you can use CFEverywhere today with current CF products.

A Downloadable Trial Application

This is the scenario discussed earlier – you have a Web application that you want to give to potential customers to "try before you buy." Here customers run your Web application with full read/write database and file capabilities on their desktops.

A Web Site On a CD

A company maintains a Web site containing a catalog of thousands of items in hundreds of categories. To promote their site the company makes CDs of the entire site, which it hands out to potential customers at an annual trade show.

The Web site exists as a database-driven series of CF templates. Without CFEverywhere, it takes days of tedious, painstaking (and expensive) effort to convert the site to thousands of interlinked static HTML pages that can be accessed from a CD. Some Web site capabilities, such as searching and sorting, are not possible in the static version. With CFEverywhere, the Web site (CF templates and

database) is copied intact to the CD along with the other components necessary to run CFEverywhere.

The time and expense of the conversion to static pages is eliminated. The user experience is enhanced because the CFEverywhere version contains all of the searching and sorting capabilities that exist on the Web site – in fact, we have duplicated the actual Web site onto the CD.

A Combination Offline/Online Application

There is a class of applications that can be used in a combination of ways. Some sections of the application do not require access to the Internet and run quite happily on the desktop using local files and data. Other sections require connection to the Web site and synchronization with the corporate database.

Consider this scenario: a manager updates a local copy of employee appraisals while off on retreat at a mountain cabin (with no access to the Internet). Later, upon returning to the real world, the manager connects to the corporate Web site and applies all the local updates to the corporate database.

There are many corporate activities that could follow the same pattern of offline data entry batched for online data update/cutover: price changes, new product catalogs, inventory counts, etc.

For example, a mail-order company with a large seasonal catalog sends a promotional CD to potential customers. Customers browse the catalog at their convenience selecting items to order: gift cards, shipping addresses, wrapping paper, etc. This is done over several days or weeks. The program is run directly from the CD, but the information is saved in a file on the customer's hard disk. When customers are ready to order, they click a button and the application connects to the vendor's Web site, uploads the order, and displays any "specials" not in the catalog. Truly, customers shop at their convenience – not governed by whether the Internet is available (or performing poorly because of traffic congestion).

Portfolio Piece

The Web is ever-changing. You may have written Web sites or applications that have been taken in-house and/or have been extended by other developers. Or, your implementation of the site was so dazzling that the company was purchased by another and for whatever reason, you can't point to these Web sites as examples of your work.

With CFEverywhere, you can maintain an archive of all the CF sites and applications you have developed, and create a "portfolio" CD for potential customers.

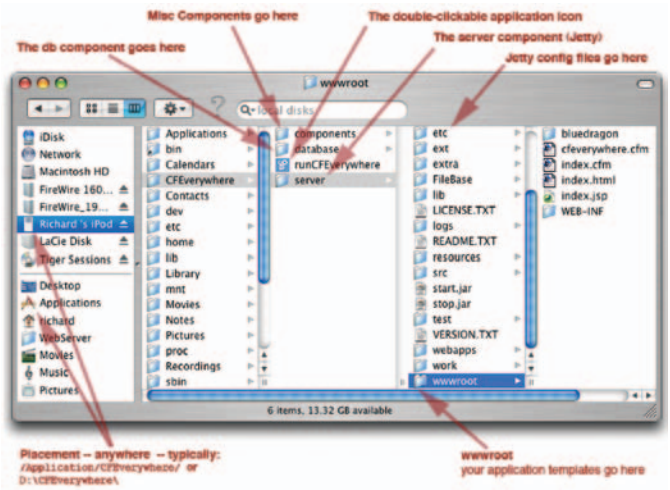


Figure 1: CFEverywhere directory structure

CF Desktop (non-Web) Applications

Traditionally, CF has been used to develop Web applications. However, many developers use CF as a quick, powerful solution for doing some non Web-related tasks such as reformatting and populating a database, statistical/design analysis of a database; application prototyping, and ad hoc report generation.

With CFEverywhere, you can actually develop applications specifically for the desktop. Use your imagination. Once you realize that you don't (necessarily) need a Web site to take advantage of the power of CF, a world of possibilities opens:

- A CF application can be a low-cost alternative to writing a desktop application in Java or C. This is especially true when the programming resources are limited and there is a large backlog of high(er) priority applications. A CF solution could be implemented to provide immediate results (and benefits) until a more rigorous solution can be implemented in the other language.
- Or, maybe the requirement is for a one-time report or analysis that is due yesterday – where CF is the only way to get it done. While a browser implementation of a user interface may not be ideal for a desktop application, it is likely to be easier and less expensive to implement than a custom UI written by other means. Arguably, the browser UI will be more familiar (and comfortable) for most users than a custom UI.
- You could write a control panel/monitor that displays the status of all components of the system.
- You can write desktop applications that invoke Web-related facilities (e-mail, messaging, Web services) to provide desktop functions.
  - For example, the control panel/monitor application could use e-mail to send an SMS text message to a technician's cellphone, alerting him that the system needs attention.
  - Or, a desktop application could invoke a Web service to obtain stock prices to update your portfolio.
  - Or, send an e-mail to a CF desktop application that reboots the database server. Since the CF application runs on the desktop, it can (with authentication) have complete access to the OS facilities, file system, scripting languages, etc.

For example, you could write a CF application that watches a certain folder in a user's system. Whenever an image file is added to the folder, the CF application executes a shell script to invoke Photoshop to copy and resize the image into a database or photo album (or product catalog).

These are only a few ideas of how CFEverywhere can deliver applications – on the desktop! Certainly, these are quite different from "typical" CF Web applications! Hopefully, we've whet your appetite for CFEverywhere. In the next section we will begin to assemble the CFEverywhere system to deliver your application.

CFEverywhere Components

To achieve CFEverywhere, the following components are

required: a J2EE server, a Web server, a database server, a CFML runtime and, of course, our application code. The J2EE server acts as a "container" for the runtime that executes the application code. Let's go through the process of choosing the appropriate components and performing their configuration. In this section, we will configure and install the Web/J2EE servers and the CFML runtime. (*Note:* This process is cross-platform and works for Windows and Mac OS X/Linux.) Figure 1 shows the CFEverywhere directory structure.

There are several J2EE servers available on the market today. Some are open source while others are commercial offerings. You may have heard of these J2EE server products: JBoss AS, BEA WebLogic, and Macromedia JRun. For our purposes we will use a server called Jetty from Mort Bay Consulting. Jetty was selected for the following reasons:

- It's free
- It includes a Web server
- The license allows for commercial use and distribution
- It's easy to install and configure
- It has a small footprint and is embeddable

We also need to select a CFML runtime. In addition to Macromedia CFMX/J2EE, there is a product called BlueDragon 6.1/J2EE from New Atlanta. Of these products, only BlueDragon can be used with CFEverywhere due to licensing restrictions. New Atlanta has a very friendly OEM/partner program that allows for distribution of BlueDragon with your CF application. Contact them and they'll be happy to discuss the details of their program. Macromedia currently has no such program but we understand the next release of CFMX (code name Blackstone) may address this issue and make it suitable for a CFEverywhere application. We'll have to wait and see.

Before we proceed further, make sure your system has a Java runtime installed. At a command prompt (or terminal on Mac OS X) type:

```
java -version
```

"The objective is to enable you to run your CF application everywhere (the Web, a desktop, or CD) – we call the concept "CFEverywhere"

You should see something like "java version 1.4.2_05." If you get an error, you don't have Java installed/configured on your system. Go to http://java.sun.com/downloads/index.html to download the Java runtime. Note that although the latest version of Java is 1.5 (a k a 5.0) most products are not yet supported for that version. Your best bet is to install a Java runtime version 1.4.x. Also, create a folder for the project and call it "CFEverywhere." For example:

- D:\CFEverywhere (for Windows)
- /Applications/CFEverywhere (for Mac OS X)

Download Jetty from the Jetty Web site, http://jetty.mortbay.org. The latest stable version is 5.1.0 so you should download jetty-5.1.0-all.zip, which includes demo files and other extras. Extract the zip file into your CFEverywhere folder. It will expand as jetty-5.1.0, but to save some typing at the command line, rename the folder *server*. Create a new folder underneath the CFEverywhere/server directory called wwwroot. This folder will serve as the Web root for our CFML application. To test your Jetty installation, open a command line and change to the server directory

```
cd d:\CFEverywhere\server
```

Then start the server

```
java -jar start.jar etc/demo.xml
```

This will start Jetty; you can test it by pointing your Web browser to http://localhost:8080/. You should see a version of the Jetty Web site. If you don't, you may have a port conflict with port 8080. Check for any other servers listening on that port and shut them down.

By launching the Jetty demo, you noticed that Jetty is configured with XML files. We need to create a configuration file for our CFEverywhere application. Copy the demo.xml to a file called CFEverywhere.xml. Open the file in any text editor and change the root context to point to the /CFEverywhere/server/wwwroot folder. It should look like the following:

```
  <!-- Add root context web applications. -->
  <Call name="addWebApplication">
    <Arg>/</Arg>
    <Arg><SystemProperty name="jetty.home" default="."/>/wwwroot</Arg>
    <Set name="defaultsDescriptor">org/mortbay/jetty/servlet/webdefault.
xml</Set>
  </Call>
```

(The complete file can be downloaded from sys-con.com/coldfusion/sourcec.cfm). Let's create a couple of test pages. Create a simple index.html page that displays "Hello World" and put it in the wwwroot directory. Next, create a file CFEverywhere.cfm with this code:

```
<cfset message = "Hello World! -- CFEverywhere" />
<cfoutput>#message#</cfoutput>
<br/>
<br/>
```

```
<br/>
<cfdump var=#cgi#>
```

Now launch Jetty with:

```
java -jar start.jar etc/CFEverywhere.xml
```

This time when you point your browser to http://localhost:8080/, you should see "Hello World." Great! We're able to serve up HTML files but we really want to run our CFML page. Enter BlueDragon…

Go to the New Atlanta Web site, www.newatlanta.com, and download the latest BlueDragon 6.1 for J2EE (BlueDragon_J2EE_61.zip). Extract the archive to a convenient location like your desktop but don't put it in the CFEverywhere folder. Copy the contents of the BlueDragon_webapp_61 folder to your CFEverywhere/server/wwwroot folder (just the contents, not the entire folder). I hope you didn't blink because you might have missed it. You just deployed BlueDragon/J2EE on Jetty! Now launch Jetty again like we did previously. Browse to http://localhost:8080/CFEverywhere.cfm and you should see "Hello World! -- CFEverywhere" as well as a dump of the CGI scope variables.

That's it! You've just completed your first CFEverywhere application. Try transferring the CFEverywhere files to another computer (even on a different platform) and see if you can get it to run. If the "Hello World! -- CFEverywhere" application doesn't inspire you, come back for the other articles in this series where we'll:

- Show you how to deploy a typical CF application that includes a database
- Make a clickable application that can run on Mac OS X and Windows
- Burn it to CD-ROM

About the Authors

Phil Cruz is a Macromedia Certified Advanced ColdFusion developer with over 12 years of experience in the computing industry. He is responsible for www.mach-ii.info, a community site for the Mach-II framework. As a micro-ISV, he created Tracking Tools, an easy-to-use bug tracking application built with Mach-II (www.tracking-tools.com).

Dick Applebaum has been involved with computers (and their predecessors) since 1956, including 16 years with IBM. Dick and two partners opened the fifth retail computer store in Silicon Valley in 1978. Eleven years later, he sold the stores and didn't touch a computer for seven years (when he discovered the Web). Since 1997, Dick has been developing applications using ColdFusion.

phil@philcruz.com
dicklacara@mac.com

Optimize, Extend and Enhance the Search Functionality in ColdFusion MX

– continued from page 11

3. *Adding more advanced end-user interface options*. Another common request among ColdFusion developers is the ability to add advanced search features not included with the ColdFusion implementation of Verity, such as spell checker/recommendations, and search-term highlighting.
4. *Searching over 250,000 documents and/or records*. Occasionally, developers will build applications that exceed the 250,000 documents and/or records that the ColdFusion implementation of Verity is limited to searching. For applications that must search large databases or repositories, developers need to extend this limit. This topic is not covered in this article. For information on overcoming the document number limitations of ColdFusion, contact sales@verity.com.

Extending and Enhancing Search with Verity Ultraseek

The three most common ColdFusion search enhancements listed above can be accomplished with the addition of Verity Ultraseek. This downloadable search engine is easily integrated into applications, using its available Java API. It can also be readily deployed into mixed application environments, using its Web services interface, which supports both the .NET and J2EE platforms. A number of factors make Ultraseek the best choice for extending search outside of the ColdFusion environment and providing additional administration tools:

- Ultraseek's easy-to-implement, set-and-forget design requires extremely low ongoing maintenance and overhead
- Ultraseek's end-user interfaces (i.e., search boxes and results lists) are similar to those of the Verity search embedded in ColdFusion
- Ultraseek provides enterprise-class search at a price point in line with ColdFusion

The best thing about Ultraseek is that you can deploy it at no cost on a 30-day trial basis. To download a free, 30-day trial version of Ultraseek, go to www.verity.com/cfsearch.

About the Author

Joe Cronin is director of Technical Services in Verity, Inc.'s Channel Partners group. He has a BS in computer engineering technology from Wentworth Institute of Technology. Verity is recognized by industry analyst groups such as Gartner, IDC, and the Delphi Group as the market leader in intellectual capital management software, including enterprise search, classification, recommendation, monitoring, and concept extraction solutions.

cfsearch@verity.com

Playing with Numbers

– continued from page 7

Quite a difference in the amount of code between the two, isn't there? What's more, this last technique is extremely flexible – add a second argument and you can now round to the next highest number evenly divisible by any increment you want, like so:

```
<cfscript>
   function roundToNextNum(theNum,theIncrement){
      return ceiling(theNum/theIncrement)*theIncrement;
   }
</cfscript>
```

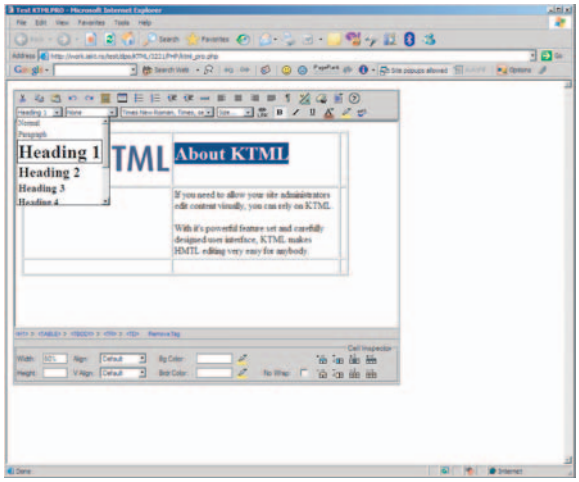
I chose to discuss this thread because it illustrates a very important concept. Not just an algebraic concept, but also a development concept. Anytime your immediate solution to a business logic problem involves treating a number as a string, stop and ask yourself whether or not there's a way to achieve your objective using math functions. There almost always is – and it's almost always more elegant. Note that I said *business logic problem*. Treating a number as a string in presentation-tier code is certainly a safe (and sometimes necessary) practice. If your math skills aren't up to snuff or you're unsure how to write an algorithm to achieve the results you want, send an e-mail to one of the list servers – you're almost always sure to get decent help.

Web (r)evolution:

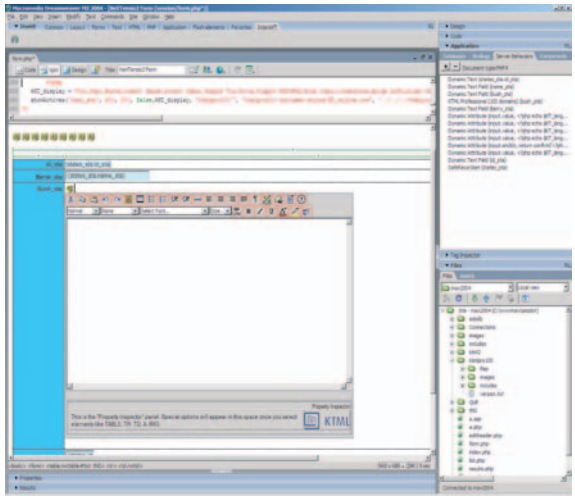
KTML

Let your clients edit their Web sites content through the browser

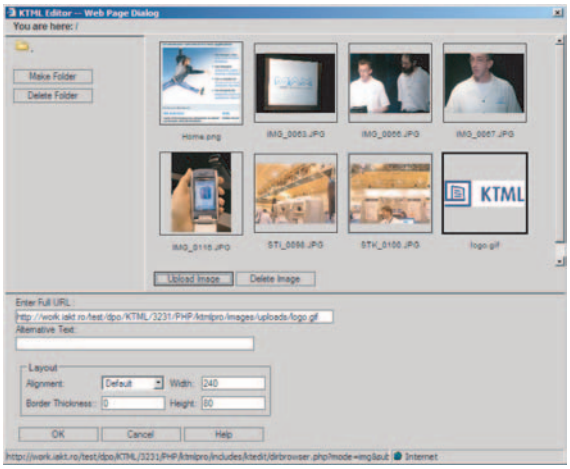
Visually use CSS styles (Word like)



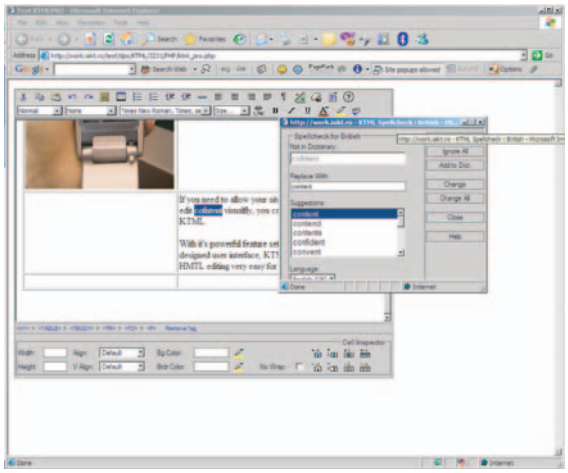
Native Dreamweaver MX integration



Easy-to-use image explorer



Multi-language spell-checker



KTML 3.5 - HTML online editor

See all the features, benefits and a live demo at:
http://ktml.interaktonline.com/

Interakt
macromedia ALLIANCE PARTNER

Supported Technologies:

macromedia DREAMWEAVER MX 2004 · php · Microsoft ASP · macromedia COLDFUSION MX · Designed for Microsoft Windows XP · Built for Mac OS X

No Experience Required

Don't be afraid to build your own Web-based data management tools

By Joe Zanter

This is not a piece full of technical anything. It's a story about a boy and his dog…ok, not that. It is a story about how a Web-dabbling engineer created some highly useful apps for his co-workers and himself.

We work in a metallurgical lab, which means, among other things, we study metals by looking at finely polished samples (let's call them "mounts," as they are often mounted/embedded in plastic) under microscopes. We like to number and keep track of these samples and their descriptions for reference, and to comply with the record retention requirements of our customers. However, a few years ago we became acutely aware of the limitations of our hard copy log (the mount book) for these samples. Even at only about 1,000 samples per year, the volume of data was too much to search through by turning pages. In other words, "It would be nice to have an application that we could all access to store data, assign new numbers, and do other neat things automatically." Web- (intranet) based seemed the obvious choice, as there were enough computers and an already established intranet presence. Other solutions were available of course, such as a Lab Information Management System (LIMS), but we weren't likely to get approval for a large software purchase. Our IT department had adopted a trouble ticket system, which was fairly simple, but very expensive – more than $20,000. We knew that wasn't going to happen for us; this data was important, but making it convenient and more useful wasn't mission critical.

We had MS Access so we started there. It worked, though it was only really meant for a single user. Knowing that much (or that little) about Access told me that this was not the solution we were looking for. The company has production databases it uses to track the manufacturing of everything we make, and for many other functions besides. Using this system was an option, but it would mean turning over development and any changes to someone else, and, consequently, someone else's timeframe. Our hard copy logs weren't rotting away, so the priority of this work probably wouldn't be very high for them, though it would be for us. That wouldn't work, as we're a rather impatient group when it comes to having the right tools—especially when the right tools seem to be within our reach. Second, changes were typically slow, given the workload of that coding group, so we'd have to live with limitations for longer than we cared to. Third, we weren't talking about one application; we were talking one application to *start* with, and several others to follow in quick succession. Fourth, if we needed another reason, the work and the output data aren't just text. Images are integral. Though we could probably adapt the production system to deal with images (storing the paths to image files), it seemed more natural to pick some kind of Web-based solution. Finally, the choice between using Access and other established company resources was the choice between doing it ourselves or giving control over to someone else. I knew we could do this, so we decided to keep control. Thus, we tried Access-generated Web pages. Retaining control was the right choice. Using Access Web pages was just a step, however.

At first, the data entry Web pages worked, but they were limited. I thought that maybe I just didn't understand all of the features, but after asking our internal guru it was clear that Access Web pages would not do what we wanted. In essence, we needed something that supported at least a handful of simultaneous users and maintained data integrity without anyone overwriting other data. Access' pages had already proven unfaithful with these modest requirements, much to my dismay. Consequently, some data had to be re-entered from memory. In this case it was caused by simultaneous users. One record was started and before it was finished another person started a record, but it was the same record, so the race was on. One set of data was written and the next overwrote the first. Data erasure is unforgivable by our customers and we don't enjoy it so much either. So – once bitten, twice shy – I went back to talk to someone in IT. Due to the requirements of traceability and data integrity in so much of what we do, the solution had to at least perform these functions well. Flexibility would hopefully be part of the solution.

Our IT department had recently started using ColdFusion on a limited basis and recommended it to me. They were running a server so it was just a matter of learning this new software (Studio 4.5). Looking back, it appears that in it's simplest form CF is just another set of HTML tags allowing the server to build pages from data in databases. At the time, it seemed a little more daunting. I didn't know SQL either. I found enough resources on the Internet to get started. Between several now-forgotten tutorials and some resources at Webmonkey, I started coding. Not having to run the server took much of the steepness out of the learning curve. After a few days of learning some of the ropes and testing code, I started building the interface to enter data for our samples. SQL syntax gave me more fits than CF. If I'd read a little bit more, it may have helped, but it wasn't too bad and there were people to talk to here if I got stuck.

It wasn't long before I had the beginnings of a usable Web interface for our mount book. It's come a long way since those first days. Figure 1 is a screen shot of the latest version of the entry page.

Walking through the major steps of the code, a security template is applied, the last mount number is queried, and a new mount number is generated. The new number, along with the user's ID, is inserted into the mount book so that it is reserved for as long as it takes to fill out the form. The number itself took a little bit of coding effort as it's alphanumeric: a letter and three digits. For each letter, mounts 0–999 are used, then the next letter comes up. That was our original numbering scheme. I think I'd do it differently if I had to do it over. The two little forms at the top are aids, one if users accidentally hit a link to start a mount and the other for inserting mount data from another existing record to save keystrokes for similar samples. Next comes the main body of the form.

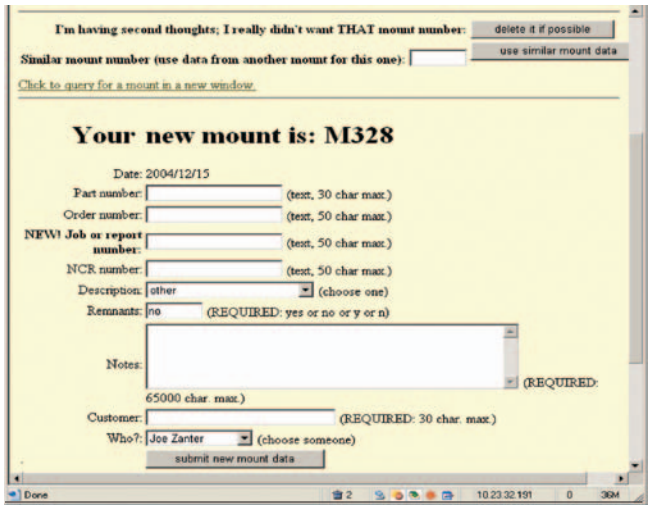


Figure 1: A screen shot of the mount book data entry page

Figure 2: A screen shot of the query results page

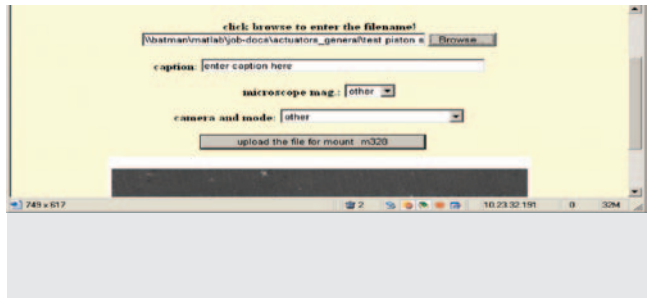


Figure 3: A screen shot of the image upload form

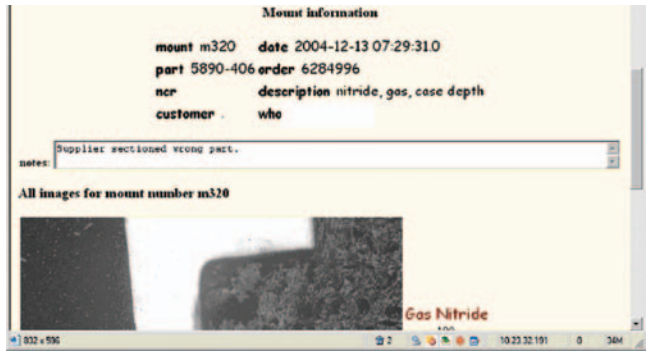


Figure 4: A screen shot of the mount book data entry page

Submitting the form gets the data inserted properly and displayed (from a query) for users to check. This application has a query associated with it, of course. The form page is plain. Figure 2 shows a sample of a mount book query result.

The part link, 2177-204, is a link to display an engineering drawing of that part. The update button allows a user to go back in and edit the record data. If that user is not the one who entered the mount, he/she can only add text to the notes field and his/her comments are labeled in the field with his/her ID. Another notable feature is the Add image link. Clicking this link takes users to yet another form where any image can be uploaded. Figure 3 shows this image upload form.

Users browse for an image and, when selected, the page displays the image with a nifty little JavaScript. The JavaScript displays the browsed image shown in Figure 3. Thanks to the HouseOfFusion lists. (Credit to Jon Hall for this script.)

```
<script type="text/javascript">
function showImage(img) {
        var imgObj = document.images['header'];
        imgObj.src = 'file:///' + img;
}
</script>

<input type="File" name="header_image" id="header_image" size="45"
onchange="showImage(this.value)">
<img src="bg_head.gif" name="header" id="header">
```

For a certain description of mount, a copy of the image is copied to another network location for processing. An image analysis application is run using a macro to control annotation and measurements and saving of those results. Later in the day, a scheduled task takes the processed image and copies it back to the images tied to that mount, inserting an appropriate record. For any mount with images associated with it, the mount number

m328 (as shown in Figure 2), becomes a link to display all images for this mount number. Figure 4 shows the display of all images for this mount page. The data from the mount book is shown, the image is displayed, and all entered image information is shown to the right of the image, including the option to inactivate the image.

I coded apps to eliminate all of our logbooks and I was asked to code a few utilities for other departments as well. One of those departments takes input from a barcode reader and calculates parameters needed to hard-anodize aluminum. I've used Verity in simple ways, such as to give us full text search options over thousands of new and legacy documents. Scheduled tasks monitors some data, reminds us of work that's due, reminds us of equipment maintenance, and supports some QA functions.

The most attractive thing about this setup is its customization and adaptability. Every one of our modest whims was satisfied and we began to have some not-so-modest whims. Some applications have undergone dozens of minor revisions and are now customized to the nth degree by end users. Adaptations emerged from suggestions like uploading images of the metal samples and tying the file to the sample record. That led to another application for uploading and indexing documents for a different department. At the end of the day, we get what we want and need and it becomes not just usable, but also useful. Our customers have access to much of our live data. Presenting analyses to remote locations on our intranet is simplified by adding a few links to an e-mail.

It didn't take long for these applications to become critical to our daily jobs. Now we rely on them, not just for the ability to query our growing body of data, but to satisfy audit requirements and customer requirements as well.

I did have the benefit of not having responsibility for the care and feeding of a Web server. This eased any learning curve that was present and allowed me to take hours rather than days to code something useful. That and the fact that that many CF tags are very straightforward made the self-teaching route practical for a part-time coder. A Ben Forta book and a copy of Studio was enough.

Looking back, I know this was a good path to follow. I knew some HTML and had built a few sets of static pages prior to learning CF; I didn't realize until I started learning ColdFusion tags that the step from static to dynamic was not large. Granted, many of my applications might be considered bicycles compared to others' SUVs. My page designs may not be pretty, but developing for yourself gets you exactly what you want. It's been great for us!

About the Author

Joe Zanter is a metallurgical engineer at Woodward Aircraft Engine Systems, a fuel systems supplier for military and commercial jet aircraft. He contributes ColdFusion applications to make his lab's data and specifications easily available to all Woodward plants.

jzante@woodward.com

Charting New Territory

Creating Dynamic Gantt Charts With ColdFusion MX

By Simon Horwith

Recently, a client approached me about adding a new requirement to their software – a hybrid content-management application written in ColdFusion, Java, and Flash. The desired new requirement was the ability to create dynamic Gantt charts on various pages in the Web application.

As you may already know, ColdFusion comes out of the box with a very powerful charting and graphing engine, and your first instinct would likely be to use those features (the CFCHART engine). Unfortunately, the complexity of Gantt charts is beyond the scope of what can be achieved easily with the charting engine built into ColdFusion. What's more, the charts had to be dynamic, not dependent on any one OS or browser version, not require having to purchase any additional software, load fast for the clients, and be easy for any designer or developer to drop into the application and have control over every aspect of the look and feel of the charts. To make matters worse, I had three days to create a functional prototype. What would you do?

Before we continue, I'll give you a brief explanation of Gantt charts. In simple terms a Gantt chart is a chart that plots things that are measurable by units of time. Often times, a single "subject" being plotted will have several entries in the timeline. For example, a Gantt chart would be a perfect way to represent time spent out of the office for all of the employees at a company. The chart would span some period of time – let's say an entire year. Measurement labels would go across the top (months, quarters, days, or weeks, for example) and down one side would be a label for each employee. Alongside each employee label, plotted, would be one or more colored shapes representative of vacation time and sick time taken by that employee. The width of the shape shows how long that particular vacation or sick leave lasted, and its start and end positions represent the actual start and end dates. You may have already seen and used Gantt charts without realizing it. Now that we all know what a Gantt chart is, let's get back to the original question – given all of the requirements and restrictions I describe above, what would you do?

Once I had ruled out using the ColdFusion charting engine (yes, that was the first thing I looked into), I immediately looked around on the Web for free third-party tools like applets and ActiveX controls. Unfortunately, nothing I found was as flexible and lightweight as my client required…not for free and/or not that could be implemented in three days and in a way that would be easy for designers and developers to use. Ideally, I'd have enough time to create a generic Gantt Charting Flash movie. A Flash movie could easily be embedded in pages, could be parameterized in order to give developers/designers control over the UI settings, and could use Flash Remoting and/or Web services to talk to the application server and get data and other information required to draw a chart. Unfortunately, creating and testing something as generic and flexible as a Flash movie was not an option given the three-day time limit I faced. I turned to SVG.

SVG stands for scalable vector graphics. It's an official recommendation of the World Wide Web Consortium (W3C) for defining graphics in XML format. SVG is similar to Flash in that it requires the client to have a plug-in (an SVG plug-in in this case) and in that it allows you to create and manipulate images. Because SVG is simply a rendering of what's defined in an XML packet, there's no scripting language built into it per se, but that doesn't mean that you can't make things extremely dynamic and user interactive. SVG exposes itself to the browser as a DOM, and as such it can be manipulated with JavaScript. User interaction, animation, and many other useful bits of functionality are usually relatively easy to achieve. I chose SVG because it's browser independent, free, has very robust graphical capabilities, and because SVG graphics are defined using XML.

ColdFusion has very good support for XML and I have a lot of experience creating APIs using custom tags and CFCs, so I knew it was a good option. I also like the fact that in the future, if time allows for a generic Flash component to be created in order to render charts, that Flash component need only know how to parse SVG XML and render it visually. The API I wrote and all of its accompanying business logic would remain the same – only the component that the XML is handed off to would need to change. All of the ins and outs of SVG are beyond the scope of this article. For more on SVG visit the W3C SVG homepage at www.w3.org/Graphics/SVG/. There's also an excellent tutorial at www.w3schools.com/svg/default.asp, and there's quite a bit of useful information on the Adobe Web site as well. In this article we'll

explore how the API prototype works.

To understand how the API works you have to be familiar with ColdFusion custom tags and be vaguely familiar with the basics of XML – you can learn about XML at www.w3schools.com/xml/default.asp.

Custom tags were used to create the API because custom tags are invoked using tag syntax, so they're easy for designers and anyone else who knows HTML or CFML to use. They encapsulate all of the logic required to do what they do and run it in a "protected" environment (variables in the tag don't overwrite variables in the caller page), which allows the API to be safely integrated with any application, and the chart API must write to the screen. I am a firm believer that custom tags, rather than ColdFusion Components, are the ideal place to encapsulate code that is part of the display tier in a CF application. Tags can also be nested within each other with nice clean syntax, which is ideal for an API like this. Creating a chart is easier when you place one <CF_CHART> tag in a file and then nest <CF_CHARTITEM> tags for each of the data sets to plot.

The only thing the tags need to do is output the XML text that represents the data you want to represent. According to the SVG specification, SVG XML consists of tags that define shapes (rectangles, circles, polygons, etc.) all nested inside of a base "SVG" tag. The API itself only needs two tags – one to create the container and one to create a shape within that container. (These shapes represent the data being plotted in the chart and this tag is called repeatedly to represent multiple data items on the chart.)

There are three ways SVG can be used in a browser:

1. A client can browse directly to an SVG file
2. SVG XML can be embedded inline within an XHTML document
3. An external SVG file can be embedded within a non- XHTML-compliant HTML document

Because the charts need to be embedded in pages containing other content, Option 1 wasn't an option for me. Unfortunately, the application for which I initially created the chart API isn't XHTML compliant, so Option 2 was out also. This left me with Option 3 – embedding a separate SVG file. Of course, the chart needs to be dynamic, so rather than linking to an actual SVG (XML) file, pages will link to a CFML file (using the same syntax as if it were SVG text), and that file will dynamically output SVG XML for the plug-in to render.

One hurdle I hit came in the form of blank pages constantly being rendered during testing. As with any good developer, I not only encapsulated business logic but I also used structured exception handling and validate all data. Typically this should be performed in the tag you pass attributes to. If you recall, in order to generate the chart I passed chart information to the "wrapper" tag to create the chart container and to a "chart item" tag in order to render individual entries on the chart. Unfortunately, these tags render XML which is then passed to the SVG plug-in… and the SVG plug-in won't render anything unless it's valid SVG XML text. So, in order to be able to catch errors thrown from the API, I implemented the "front controller" design pattern and created one tag that sits in front of the API. This tag validates all of the data for the API, calculates new height/width values for the entire chart if it needs to (the outer container must be high and wide enough to hold all of the shapes being plotted, for example), writes all that data to the session scope, and then makes the <embed> call to another CFM file (that calls the SVG XML rendering tags based on what's in the session scope).

The data is written to the session scope because of a second hurdle I hit: you can't pass complex data to a file using the <embed> tag. This is because the embed tag makes a separate request – meaning the variables in the local "caller" CFM aren't accessible. To get around this limitation I temporarily put all of the data required for a chart in ColdFusion's session scope, thus making it accessible to the embedded page and keeping it user specific, so it doesn't conflict with data that other users or charts are concurrently trying to view.

The data validation in this tag represents the bulk of the actual programming I had to do. In order to make the API robust and easy to use, I made the chart height and width larger than what the user specified. If too much data had been passed for representation, I gave options for turning on and off "subject" and chart unit measurement labels; the ability to control the background color and transparency, and to control the same traits for a unit of measurement grid; and the ability to control fonts and font sizes, etc. I also had to calculate the absolute pixel placement for whatever types of shapes would be plotted.

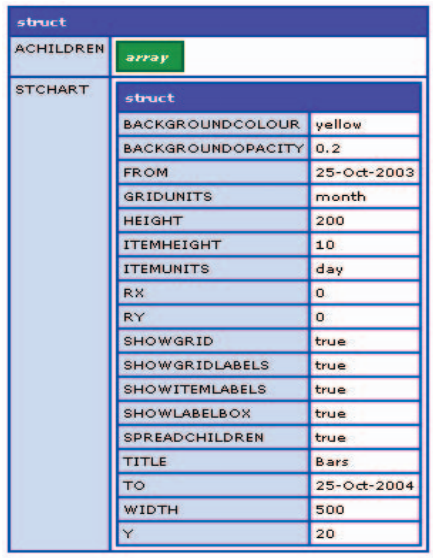


Figure 1: Parent chart data

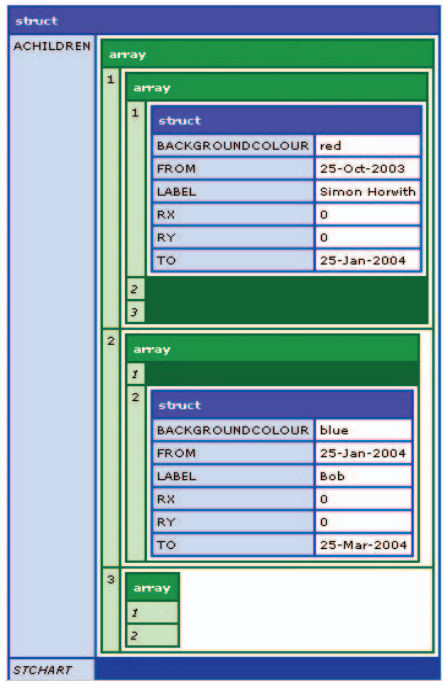


Figure 2: Child data to plot in the chart

This is because graphics in SVG are drawn very similarly to the way the ActionScript Drawing API works. You must pass shape point coordinates, do "line-to"s, etc., based on pixel coordinates. For the end user it's as simple as nesting start and end date information for each shape, but the tag calculates the scale and the pixel coordinates (offset from the master container) for all of the items in the background. The embedded CFM file calls a tag that outputs

the container XML, and has a nested (within the container tag) loop over the data to plot inside the chart. As it loops over that data, it calls a tag that plots an individual chart entry on each pass in order to spit out the XML for individual items. In this case all validation had to happen in the first tag call – before the EMBED tag. This is because exception and/or any other output that's not SVG

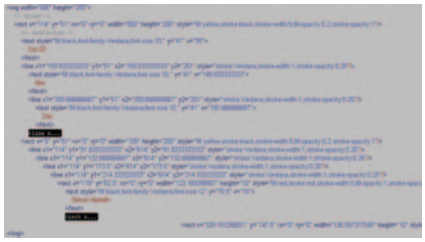


Figure 3: XML returned to the plug-in from the API

```
<gant:ganttchart
itemUnits="#session.stChartData.stChart.itemUnits#"
gridUnits="#session.stChartData.stChart.gridUnits#"
width="#session.stChartData.stChart.width#"
height="#session.stChartData.stChart.height#"
from="#session.stChartData.stChart.from#"
to="#session.stChartData.stChart.to#"
itemHeight="#session.stChartData.stChart.itemHeight#"
y="#session.stChartData.stChart.y#"
backgroundColour = "#session.stChartData.stChart.backgroundColour#"
backgroundOpacity = "#session.stChartData.stChart.backgroundOpacity#"
borderColour = "#session.stChartData.stChart.borderColour#"
borderOpacity = "#session.stChartData.stChart.borderOpacity#"
borderWidth = "#session.stChartData.stChart.borderWidth#"
rx = "#session.stChartData.stChart.rx#"
ry = "#session.stChartData.stChart.ry#"
showGrid = "#session.stChartData.stChart.showGrid#"
spacing = "#session.stChartData.stChart.spacing#"
spreadChildren = "#session.stChartData.stChart.spreadChildren#"
x = "#session.stChartData.stChart.x#"
unitLength = "#session.stChartData.stChart.gridUnitLength#"
showGridLabels = "#session.stChartData.stChart.showGridLabels#"
showItemLabels = "#session.stChartData.stChart.showItemLabels#"
gridLabelFont = "#session.stChartData.stChart.gridLabelFont#"
gridLabelColour = "#session.stChartData.stChart.gridLabelColour#"
gridWidth = "#session.stChartData.stChart.gridWidth#"
gridColour = "#session.stChartData.stChart.gridLabelFont#"
gridOpacity = "#session.stChartData.stChart.gridOpacity#"
labelX = "#session.stChartData.stChart.labelX#"
showLabelBox = "#session.stChartData.stChart.showLabelBox#"
gridUnitLength = "#session.stChartData.stChart.gridUnitLength#"
gridLabelFontSize = "#session.stChartData.stChart.gridLabelFontSize#"
title = "#session.stChartData.stChart.title#">
```

Figure 4: Calling the parent tag (the chart container)

Figure 5: Calling the children tags (the data to plot) in a loop

cannot take place within the files that generate SVG – by then the plug-in is doing all rendering and if it's not valid SVG XML, it doesn't get displayed. Figure 1 and Figure 2 show how the data for the parent (chart) and individual plotted data items look, respectively. Figure 3 is an example of the XML generated by the API.

The result is a base tag that, based on its parameters, will show multiple subjects with multiple data that span time within a nice chart interface…and nearly every aspect of individual items and the chart itself is configurable. Examples of configurable values for chart data or the chart itself include colors, transparency, fill, line width, corner sharpness (round or square corners), and much more. Most important is the fact that all of this is implemented simply by passing data to a simple tag call. Designers and developers can implement the API with no knowledge of SVG or the inner workings of the code, and can tweak the data they send and see the changes in real time in order to make it look exactly as they want. Figure 4 and Figure 5 show what typical calls to the API look like under the hood (in a page, the API call is trivial – you simply pass one complex variable to a single tag in a single attribute).

Adding further functionality to the API is also very simple for someone with a little CF experience without having to look at most of the API code complexities. The entire API proof of concept version took me approximately two days to write, test, and document. Figures 6 through 9 show simple charts created with the API.

More than any other reason, ColdFusion is chosen as a solution because of its rapid development environment. It is also robust enough to create APIs that expose complex functionality to developers and designers who would otherwise be unable to achieve similar results on their own. In only two days, ColdFusion's XML functionality and custom tag architecture gave me the ability to allow my client to easily create the XML required to generate impressive Gantt charts in their pages using the open source definition (SVG) for representing graphics with XML. Future implementations with Flash, applets, or any other technology will be easy to implement using the groundwork laid down with the prototype API, which in the meantime could also be left in place as-is for as long as the client requires. Though Flash and other Macromedia technologies are often the ideal solution, sometimes they aren't an option due to business requirements, or time or cost constraints. Leveraging open source and third-party solutions from within Macromedia technology solutions can offer developers the best of both worlds.

All of the code for the Gantt Chart API is available for free download at www.horwith.com/downloads/gantt_proto.zip. You are welcome to use and modify it any way you see fit. Also, Shlomy Gantz has taken the code base and created a simple HTML Gantt Chart API, which he has also made freely available at www.shlomygantz.com/customtags/cf_gantt/gantt_readme.cfm.

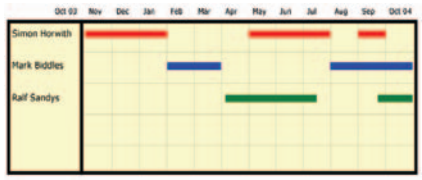


Figure 6: A "straight corner" chart with entries that span time

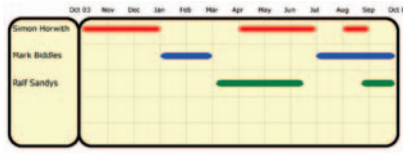


Figure 7: The same chart as Figure 6 after changing one value for the corners

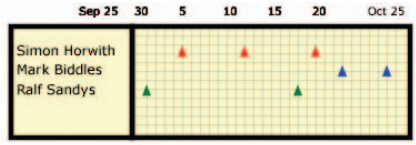


Figure 8: Showing individual events (span one day) with triangles

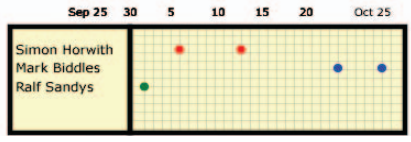


Figure 9: The same data passed as Figure 8, but "circle"shape now specified

CFDJ Advertiser Index

ADVERTISER	URL	PHONE	PAGE
ACTIVEPDF	WWW.ACTIVEPDF.COM	866-468-6733	21
CFDYNAMICS	WWW.CFDYNAMICS.COM	866-233-9626	4
COLDFUSION DEVELOPER'S JOURNAL	WWW.SYS-CON.COM/COLDFUSION	888-303-5282	38
EKTRON	WWW.EKTRON.COM/CFDJ	866-4-EKTRON	23
EV1 SERVERS	WWW.EV1SERVERS.NET	800-504-SURF	6
HAL HELMS, INC	WWW.HALHELMS.COM		37
HOSTMYSITE.COM	WWW.HOSTMYSITE.COM/BLACKSTONE	877-215-4678	27
INTERAKT ONLINE	HTTP://KTML.INTERAKTONLINE.COM/	4031 401.68.19	29
INTERMEDIA.NET	WWW.INTERMEDIA.NET	800-379-7729	COVER IV
MACROMEDIA	WWW.MACROMEDIA.COM/GO/VOLVO	415.252.2000	COVER II & PAGE 3
MACROMEDIA WEB PUBLISHING SYSTEM	WWW.WEBPUBLISHINGSYSTEM.COM	415.252.2000	COVER III
PAPERTHIN	WWW.PAPERTHIN.COM	800-940-3087	33
SEAPINE SOFTWARE	WWW.SEAPINE.COM/WHITEPAPER.PHP	888-683-6456	15
SERVERSIDE	WWW.SERVERSIDE.NET	888-682-2544	19
WEB SERVICES EDGE 2005 EAST	WWW.SYS-CON.COM/EDGE	201-802-3066	39 - 48
SYS-CON PUBLICATIONS	WWW.SYS-CON.COM/2001/SUB.CFM	888-303-5282	31
SYS-CON NEWSLETTERS	WWW.SYS-CON.COM	888-303-5282	37

See what everyone is talking about… ***The Application Server Shoot-Out!***

International Web Services Conference & Expo

Keynotes

Matt Ackley
eBay

Ari Bixhorn
Microsoft

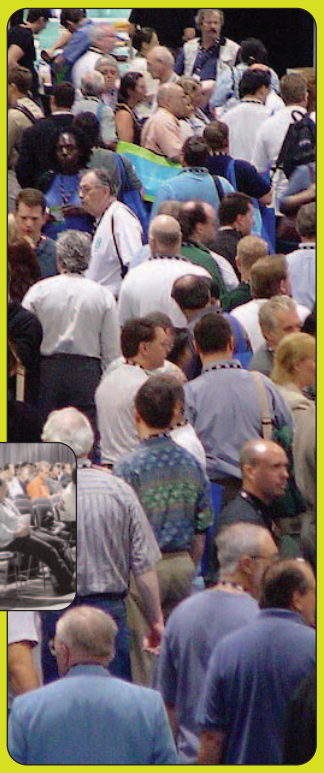
Mike Milinkovich
Eclipse.org

web services conference EDGE & expo

Register by JANUARY 21, 2005 SAVE Up To $300

WEB SERVICES EDGE 2005 EAST

Product Teams Delivering SOA, Web Services, Java, .NET, & XML Solutions



New for 2005…

- Cabana Night Developer Exchange
- Application Server Shootout
  facilitated by Burton Group
- Hands On VS and .NET Labs
- Software Giveaway

Free Tutorials presented by critical sites · Novell · Microsoft · Java Community Process

For more information:
visit: www.sys-con.com/edge
call: 201-802-3066
e-mail: jimh@sys-con.com

February 15-17, 2005
John B. Hynes Convention Center
Boston, MA

PRODUCED BY: SYS-CON MEDIA
PRESENTED BY: SYS-CON EVENTS

DIAMOND SPONSOR: Microsoft
GOLD SPONSORS: Novell, SAP
ASSOCIATION SPONSORS: Integration Consortium, OASIS, ineta, eclipse
CORPORATE SPONSOR: Java Community Process
APP.-SERVER SHOOTOUT FACILITATOR: Burton Group
MEDIA SPONSORS: Information Storage+Security Journal, JDJ, WebServices Journal, .NET Journal, Queue, CoDe, XML Journal, wldj, asp.netPRO, WebSphere Journal, ColdFusion Journal, SD Times, MX developer's journal

SPECIAL INSERT: WEB SERVICES EDGE 2005
WWW.SYS-CON.COM/EDGE
FEBRUARY 15-17, 2005
JOHN B. HYNES CONVENTION CENTER • BOSTON, MA

web services conference&expo EDGE

Web Services Edge 2005 East

February 15–17, 2005
John B. Hynes Convention Center
Boston, MA

The Largest *i*-Technology Event of the Year!

Who Should Attend

- Software Developers
- Software Engineers
- Application Developers
- Development Managers
- Technical Directors
- Analysts/Programmers
- IT Managers
- CEOs
- CTOs
- CIOs
- Technical Architects
- Team Leaders
- Software Consultants
- IT Directors
- Project Managers

Features & Attractions

- 3 Days Packed with Education and Training
- Keynotes & Panel Discussions from Industry Leaders
- 50+ Hard-hitting and Informative Seminars
- FREE Web Services Tutorial Presented by Novell
- FREE .NET Tutorial Presented by Microsoft
- FREE Web Services Security Tutorial Presented by Critical Sites
- Cabana Night Developer Exchange Presented by INETA
- Panel Discussions Presented by JCP
- Opening Night Welcome Reception
- Compelling Case Studies & Best Practices
- Hands-On Labs Featuring .NET and Visual Studio Presented by Microsoft Partners
- Featured Product Demonstrations
- Exhibit Floor featuring hundreds of products
- Real-time SYS-CON Radio Interviews

VISIT **www.sys-con.com/edge** FOR TIMES & SCHEDULES

The program, including topics and times, is subject to change. Please refer to www.sys-con.com for all updates.

SPECIAL INSERT: WEB SERVICES EDGE 2005
WWW.SYS-CON.COM/EDGE
FEBRUARY 15-17, 2005
JOHN B. HYNES CONVENTION CENTER • BOSTON, MA

ices Conference & Expo

Keynote Speakers

Tuesday, February 15, 11 a.m.

Matt Ackley
Senior Director, eBay Developers Program

Web Services for eCommerce

eBay, The World's Online Marketplace, has more than 114 million registered users, 10,000 developers, and over 700 live, third-party applications. Four years ago, eBay began allowing third parties to build applications that tap into eBay, and today eBay hosts one of the leading Web services platforms. Through its developer program, eBay enables third parties to create cutting-edge Web services applications that benefit the buyers and sellers on eBay. At present, 40% of eBay's listings come through its API, which handles more than a billion Web services calls a month. Ackley will discuss the rewards and challenges of building and maintaining one of the world's leading Web services platforms, and share insights and practical guidelines for others.

Matt Ackley is senior director of the eBay Developers Program. He supports eBay's vision to be the leading platform for global online commerce, and is chartered with creating a thriving ecosystem between eBay, its community of users, and third-party developers and solution providers. Ackley joined eBay in 2003 as part of eBay's acquisition of FairMarket, which provided technology solutions and services to online marketplaces.

Wednesday, February 16, 11 a.m.

Ari Bixhorn
Director, Web Services Strategies, Microsoft Corporation

Introducing Indigo: The Unified Programming Model for Building Service-Oriented Applications

Indigo is Microsoft's unified programming model for building service-oriented applications on the Windows platform. It enables developers to build secure, reliable, transacted solutions that integrate across platforms and interoperate with existing investments. Indigo combines and extends the capabilities of existing distributed application technologies, including .NET Enterprise Services, System.Messaging, Remoting, ASMX, and WSE to deliver a unified development experience spanning distance, topologies, hosting models, protocols, and security models. This keynote will provide an inside look at Indigo and show you how Indigo will radically simplify the development of distributed, service-oriented applications.

Ari Bixhorn is the director of Web Services Strategy in the Developer and Platform Division at Microsoft Corp. He is responsible for product planning and technical evangelism for Microsoft's Web services offerings, including "Indigo," the code name for a component of the next version of the Windows operating system, code-named Windows "Longhorn." Bixhorn has spent the past five years at Microsoft, driving product management efforts for the Visual Basic and Visual Studio development systems.

Thursday, February 17, 11 a.m.

Mike Milinkovich
Eclipse.org

An Open Development Platform for Web Services

Open source technology runs the Internet. Linux, Apache, PHP and Eclipse are highly successful open source communities that provide the backbone for today's Web applications. All indications point to a continued value proposition for organizations for leveraging open source when developing and deploying SOA-based applications. This keynote will examine the benefits of using open source technologies, the decision-making process used when adopting these solutions and the potential for contributing back to the open source community.

Mike Milinkovich has held key management positions at Oracle, WebGain, The Object People, and Object Technology International Inc. (which subsequently became a wholly owned subsidiary of IBM), assuming responsibility for development, product management, marketing, strategic planning, finance, and business development. Mike earned his MS degree in information and systems sciences and a bachelor of commerce degree from Carleton University in Ottawa, Canada.

VISIT **www.sys-con.com/edge** FOR TIMES & SCHEDULES

The program, including topics and times, is subject to change. Please refer to www.sys-con.com for all updates.

SPECIAL INSERT: WEB SERVICES EDGE 2005
WWW.SYS-CON.COM/EDGE
FEBRUARY 15-17, 2005
JOHN B. HYNES CONVENTION CENTER • BOSTON, MA

SPECIAL INSERT: WEB SERVICES EDGE 2005

WWW.SYS-CON.COM/EDGE

FEBRUARY 15-17, 2005

JOHN B. HYNES CONVENTION CENTER • BOSTON, MA

DAY 1 FEBRUARY 15

	Java	.NET	Web Services
7:30	Registration		
8:00	**FREE Tutorial** – Ashish Larivee, Novell, **Using a Web Services Framework to Build SOA Applications**		
9:00	(J-1) What's New In JDO 2.0	(.NET-1) Intro to SPOT	(WS-1) Ensuring Web Services Interoperability
10:00	(J-2) Using Java Messaging in Real-Time Trading Systems	(.NET-2) An Introduction to SQL Server Reporting Services	(WS-2) Web Services Standards: Going Behind the Mask
11:00	**Opening Keynote** – Matt Ackley, Senior Director, eBay Developer Program, eBay		
12:00	**EXPO OPEN** (12 P.M.–5 P.M.)		
3:00	**Keynote Panel** Presented by JCP – **Web Services and Security** Moderator: Onno Kluyt, Sr Director & Chair, JCP Program, Sun Microsystems		(WS-2B) Solving Complex Business Problems Though SOA
4:00	(J-3) The ROI of a Java-Rich Client	(.NET-3) Go With The Flow – Human Workflow Services in BizTalk 2004 / (.NET-3B) Techniques with Visual Basic.NET	(WS-3) The XML Data Challenge
5:00	Opening Night Reception		

DAY 2 FEBRUARY 16

	Java	.NET	Web Services
7:30	Registration		
8:00	**FREE Tutorial –** Thom Robbins, Microsoft – **The Next Generation of Visual Studio** (free with VIP preregistration)		
9:00	(J-4) Web Services End-to-End Security on J2EE: Gaps and Proposed Solutions	(.NET-4) The Microsoft Framework: An Agile Software Development Process for Building Web Service Applications	(WS-4) How To Bulletproof Your Web Services
10:00	(J-5) J2ME and Eclipse	(.NET-5) Securing Service-Oriented Architecture with Microsoft's WSE 2.0	(WS-5) The Role of Policy in Web Services Integration – It's More Than Just Security
11:00	**Keynote** – Ari Bixhorn, Director, Web Services Stategies, Microsoft Corporation		
12:00	**EXPO OPEN** (12 P.M.–4 P.M.)		
3:00	**Application Server Shootout**		
4:00	(J-6) The Impact of JBoss and Mono on the Application Server Market	(.NET-6) Web Services Security for Dummies with WSE2 / (.NET-6B) J2EE to .NET Interoperability and App. Integration	(WS-6) B2B Policy Enforcement: The Third Rail of SOA Implementation
5:00	(J-7) Migrating Enterprise Applications Between J2EE Application Servers	(.NET-7) So You THINK You Know What an Object Is… / (.NET-7B) Building and Using Advanced ASP.NET Web Controls	(WS-7) Driving SOA Governance
6:00	**Cabana Night –** Hosted by INETA		

DAY 3 FEBRUARY 17

	Java	.NET	Web Services
7:30	Registration		
8:00	**FREE Tutorial** – Patrick Hynds and Duane Laflotte, Critical Sites – **Security, The New Reality** (free with VIP preregistration)		
9:00	(J-8) Design Patterns and Project Organizational Techniques for "Write Once, Debug Everywhere"	(.NET-8) Migrating ASP to ASP.NET	(WS-8) SOA: From Pattern to Production
10:00	(J-9) Using Grid Computing with Web Services and J2EE to Create Internet-based SOAs	(.NET-9) Smart Client Development with the Offline Application Block	(WS-9) High Performance Web Services – Tackling Scalability and Speed
11:00	**Keynote** – Mike Milinkovich, Executive Director, Eclipse Foundation		
12:00	**EXPO OPEN** (12 P.M.–4 P.M.)		
3:00	(J-10) Java Web Services Programming Tips & Tricks	(.NET-10) CLR Internals	(WS-10) So You Want an SOA: Best Practices for Migrating Toward Service Orientation in the Enterprise
4:00	(J-11) JCP Program: How the Java Technology Binary Software Standard is Managed and Evolves	Visit Web site for update	(WS-11) Four Abilities SOA Will Lack Without a Registry

The program, including topics and times, is subject to change. Please refer to www.sys-con.com for all updates.

	WS Security	Case Study
	FREE Tutorial –Aaron Williams, JCP, **Developing Web Services Using Java Technology**	
	(WSS-1) Identity in SOA	(CS-1) Developing E-Commerce Applications with Web Services
	(WSS-2) Securing Web Services with WS-Security	(CS-2) Developing Enterprise Class Web Services
		(CS-3) Service-Oriented Development on NetKernel – Patterns, processes and product to reduce the complexity of IT systems
	(WSS-3) Anatomy of a Web Services Attack	

Novell

FREE Web Services Security Tutorial Presented by Novell

Using a Web Services Framework to Build SOA Applications

Tuesday, Feb. 15
8:00 A.M. – 11:00 A.M.

FREE Tutorial Presented by Java Community Process

Developing Web Services Using Java Technologies

Tuesday, Feb. 15
8:00 A.M. – 11:00 A.M.

Java Community Process

Free Tutorials with VIP Preregistration ONLY!

	WS Security	Case Study
	(WSS-4) Using Mobile Phones as an SSO Authentication Device in SOA Solutions	(CS-4) Orchestrating FORCEnet Engagement Packs with BPEL for Web Services
	(WSS-5) Building Intelligent Enterprises with Novell's Identity-Driven Computing	(CS-5) CPI: A Globally Integrated Problem-Tracking and Resolution System Using Java Web Services
	(WSS-6) XML Content Attacks	(CS-6) The Transformation of SiteRefresh into a Web Services
	(WSS-7) The Interoperability Challenge of Web Services Security Standards	

Microsoft

FREE
.NET Tutorial Presented by Microsoft

The Next Generation of Visual Studio

Thom Robbins

Wednesday, Feb. 16
8:00 A.M. – 11:00 A.M.

Free Tutorials with VIP Preregistration ONLY!

	WS Security	Case Study
	on)	
	(WSS-8) Transitioning Successfully to SOA and Web Services: Building the Infrastructure for SOA Growth	(CS-8) Using SOA and Web Services to Issue Business Licenses in the District of Columbia
	Visit Web site for update	(CS-9) Developing Web Services with Eclipse

critical sites

FREE
.NET Tutorial Presented by Critical Sites

Security, The New Reality

Patrick Hynds

Thursday, Feb. 17
8:00 A.M. – 11:00 A.M.

Free Tutorials with VIP Preregistration ONLY!

The program, including topics and times, is subject to change. Please refer to www.sys-con.com for all updates.

SPECIAL INSERT: WEB SERVICES EDGE 2005
WWW.SYS-CON.COM/EDGE
FEBRUARY 15-17, 2005
JOHN B. HYNES CONVENTION CENTER • BOSTON, MA

WEB SERVICES TRACK

WS-1 Ensuring Web Services Interoperability

Chris Ferris, IBM

Tuesday, Febuary 15, 2005 9:00 A.M. – 9:50 A.M.

Despite the open industry standards that underlie Web services, interoperability has been a key challenge for vendors and customers implementing Web services. One reason for this is that the relevant industry standards often permit multiple acceptable implementation alternatives. This presentation will discuss in detail the challenge of Web services interoperability and the role played by the premier industry organization formed to address it, the Web Services Interoperability Organization. In particular, the presentation will cover the critical importance of WS-I profiles to an organization's Web services initiatives, including the manner in which companies can put WS-I profiles immediately to work.

BIO: Chris Ferris is chair of the WS-I Basic Profile Working Group and a senior technical staff member with IBM's Emerging Technology Group. He has been actively engaged in open standards development for XML and Web services since 1999 and is an elected member of the OASIS Technical Advisory Board. Chris is also a coauthor and editor of the WS-Reliable Messaging specification.

WS-2 Web Services Standards: Going Behind the Mask

Glen Daniels, Sonic Software

Tuesday, Febuary 15, 2005 10:00 A.M. – 10:50 A.M.

Web services and service-oriented architectures (SOAs) are emerging as an integral part of the enterprise IT strategy. According to a recent IDC study, Web services – related revenue is expected to triple from $1.1 billion worldwide in 2003, to $3.4 billion in 2004, and $16.6 billion by 2008. As SOAs proliferate and the number of Web services added to them increases, standards will play an increasingly significant role. This session will look at the state of key Web services standards such as WS-Choreography, WS-Reliability and WS-ReliableMessaging, SOAP/MTOM/XOP, WSDL, XPath, XQuery, and WS-Notification as well as related Java standards and open source efforts. It will also look at the organizational impact of standards adoption in the industry.

BIO: Glen Daniels is manager of standards and consortia at Sonic Software and coauthor of *Building Web Services with Java*. He has been working with Web services technologies since their inception in the late '90s, and in addition to developing products and helping to found Apache's Axis project, he has been an active participant in standards bodies such as the W3C, and a member of the SOAPBuilders interoperability group.

WS-2B Solving Complex Business Problems Though SOA

John Daly, netNumina

Tuesday, Febuary 15, 2005 3:00 P.M. – 3:50 P.M.

WS-3 The XML Data Challenge

Nancy Vodicka, DataDirect Technologies

Tuesday, Febuary 15, 2005 4:00 P.M. – 4:50 P.M.

Most businesses store and query data with relational databases but need to use Extensible Markup Language (XML) to exchange and display data on the Web and with vendors and partners. As a result, programmers need to deal with both relational and XML data, often at the same time. Emerging standards such as XQuery, XQJ, and SQL/XML, promise to revolutionize data exchange and the ways applications are developed, deployed, and utilized. Learn the key facts about these standards, including what they mean, when they will be available, and what you, the developer, can do to prepare.

BIO: As the XML Product Manager at DataDirect Technologies, Nancy Vodicka is responsible for DataDirect Connect for SQL/XML, a database-independent SQL/XML implementation that is currently shipping, and DataDirect XQuery, a database-independent XQuery implementation that is currently in development. Nancy has more than 15 years experience in the software industry working with technologies such as XML, Web services, relational databases, and SQL.

WS-4 How To Bulletproof Your Web Services

David McCaws, Parasoft

Wednesday, Febuary 16, 2005 9:00 A.M. – 9:50 A.M.

Web services are gaining industry-wide acceptance and usage and are moving from proof-of-concept deployments to actual usage in mission-critical enterprise applications. Web services range from major services such as storage management and customer relationship management to much more limited services such as furnishing stock quotes or providing weather information. As companies and consumers begin to rely more and more on Web services, the need for developing reliable, high-quality Web services is even stronger. This session will explain issues specific to Web services and will illustrate solid engineering and testing practices required to ensure complete Web service functionality, interoperability, and security. Whether creating Web services from scratch or integrating a legacy back-end server via Web services, the practices and principles outlined in this session will be of great benefit.

BIO: David McCaw has over eight years of experience in helping software development teams improve quality throughout the development process. Over the last three years, he has led the Parasoft Web Services Solutions team, which has developed an industry-leading approach for Web services testing. He has implemented Web service quality solutions for development groups in organizations such as Sabre-Holdings, Yahoo! Overture, and McGraw-Hill. McCaw has an extensive background in the areas of Java and Web service reliability, performance, and security. He is involved with OASIS and WS-I, and is a frequent speaker at industry events.

WS-5 The Role of Policy in Web Services Integration – It's More Than Just Security

Toufic Boubez, Layer 7 Technologies

Wedneday, Febuary 16, 2005 10:00 A.M. – 10:50 A.M.

Too often today the preferences, terms, and conditions describing how a Web service behaves when discovered and invoked is programmed right into the business logic. Hard-coding this behavior logic however introduces cost, complexity, and rigidity into a Web services architecture. A better approach is to abstract a Web services usage "policy" out of code where this metadata can be managed as need be. This session introduces the concept of Web Services Policy and describes how the construct can be used to implement a more customized and versatile Web service infrastructure.

BIO: Toufic Boubez is a well-respected and renowned Web services visionary. Prior to cofounding Layer 7 Technologies, Toufic was the chief Web services architect for IBM's Software Group and drove their early XML and Web services strategies. He is a sought-after presenter and has chaired many XML and Web services conferences. He is an author of many publications and his most recent book is the top-selling *Building Web Services with Java: Making Sense of XML, SOAP, WSDL ,and UDDI.*

WS-6 B2B Policy Enforcement: The Third Rail of SOA Implementation

Alistair Farquharson, Digital Evolution

Wednesday, Febuary 16, 2005 4:00 P.M. – 4:50 P.M.

One of the great benefits of a service-oriented architecture is the ability it gives you to extend programmatic, integration capabilities to business part-

ners. Going beyond simple sharing of data with partners, SOA enables true B2B application integration. At the same time, this capability creates a vexing security policy enforcement dilemma. How can you be sure that a user from a partner organization is actually authorized to integrate with your applications? How can you authenticate that user? Do you even want that headache in the first place? This session will discuss the issues that arise in B2B security policy enforcement and explore several proven approaches to solving the problem. In particular, it will focus on the emerging technology of XML Virtual Private Networks (XML-VPNs) and their potential to mitigate security policy enforcement issues in B2B SOA implementations.

BIO: Alistair Farquharson is the CTO of Digital Evolution, where he spearheads product development and provides thought leadership to enterprise customers implementing Web services. His skills span many industries and include designing and implementing system architectures, as well as spearheading initiatives such as development/team lead. He is an expert in custom-application development, distributed environments, architecting scalable hardware and software applications and systems, and Web services application development.

WS-7 Driving SOA Governance

Brent Carlson, LogicLibrary

Wednesday, Febuary 16, 2005 5:00 P.M. – 5:50 P.M.

In the past year, Web services and service-oriented architectures (SOAs) have become mainstream because of their ability to provide business agility and flexibility through integration, productivity, and reuse. With SOA enablement on the rise, IT groups must address SOA governance as a means of controlling what and how services located within an SOA are deployed. This session will discuss SOA governance, specifically how an organization can manage and control assets and artifacts located within an enterprise, while ensuring that deployed assets meet an organization's business and technical architectural standards. It will also outline governance best practices such as monitoring the UDDI publish process in order to seamlessly tie together the development and operational views of Web services within the enterprise.

BIO: Brent Carlson drives the development and delivery of LogicLibrary's products. He is a 17-year veteran of IBM, where he served as lead architect for the WebSphere Business Components project and held numerous leadership roles on the "IBM San Francisco Project." He is a member of the Eclipse Board of Stewards and a BEA Regional Director.

WS-8 SOA: From Pattern to Production

David Chappell, Sonic Software

Thursday, Febuary 17, 2005 9:00 A.M. – 9:50 A.M.

Service-oriented architecture (SOA) represents the opportunity to achieve broad-scale interoperability, while providing the flexibility required to continually adapt technology to business requirements. No small feat, particularly when one considers the extent and complexity of today's IT environments. As both a technology concept and IT discipline, the challenge inherent in SOAs is maintaining the right architectural approach. If all services in an SOA are treated as interdependent point-to-point interfaces, then the complexity of implementing and maintaining them in this spaghetti-like architecture becomes enormous. The enterprise service bus (ESB) has emerged as one of the first true SOA product offerings, bringing SOA from pattern to production. ESBs provide a framework for building and deploying an event-driven, enterprise SOA and accommodates the configuration, hosting, and management of integration components as services across the business.

BIO: VP and chief technology evangelist for Sonic Software, Dave Chappell has over 18 years of experience in the software industry covering a broad range of roles including R&D, codeslinger, sales, support, and marketing. He also has extensive experience in distributed computing, including message-oriented middleware, CORBA, COM, and Web application server infrastructure.

WS-9 High Performance Web Services – Tackling Scalability and Speed

Sameer Tyagi, Sun Microsystems

Thursday, Febuary 17, 2005 10:00 A.M. – 10:50 A.M.

Web services facilitate application-to-application integration and interoperability across different platforms. However, critics usually point to an inefficient processing model and bandwidth requirements for developing Web services. This is often cited as a reason why Web services cannot perform and scale well in production environments. This session takes a detailed look at performance and scalability issues around Web services in the real world, as well as strategies that architects and developers can adopt to mitigate such risks in these applications. Some analytical and modeling strategies that enable acceptable application performance will also be covered.

BIO: Sameer Tyagi works as a senior Java architect with Sun Microsystems. He remains focused on architecture, design, and implementation of large-scale enterprise applications with Java technology. His publications include industry periodicals and books on Java and J2EE technologies including *Java Web Services Architecture.*

WS-10 So You Want an SOA: Best Practices for Migrating Toward Service Orientation in the Enterprise

Eric Newcomer, IONA

Thursday, Febuary 17, 2005 3:00 P.M. – 3:50 P.M.

Replacing complex, monolithic applications with nimble applications built from exposed services promises increased developer productivity, greater flexibility, and ultimately reduced cost. The adoption of Web services and SOA can also remove a significant level of complexity and integration problems from enterprise application development projects. But, as with any large-scale project, IT departments must have the right plan and the right resources in place to ensure a successful transformation of their computing infrastructure. This article will explore what IT organizations need to know to be successful in their attempts to migrate the enterprise to a service-oriented architecture.

BIO: In the role of chief technology officer at IONA, Eric Newcomer is responsible for IONA's technology roadmap and the direction of IONA's e-business platforms as relates to standards adoption, architecture, and product design.

WS-11 Four Abilities SOA Will Lack Without a Registry

Luc Clement, Systinet

Thursday, Febuary 17, 2005 4:00 P.M. – 4:50 P.M.

A service-oriented architecture (SOA) is the design blueprint for seamless connectivity between business processes and IT infrastructure, enabling innovation and improving productivity. SOA provides the most efficient, standard way to dynamically interoperate with any customer, supplier, product or employee. SOA makes integration intrinsic. Web services are the foundation building blocks of an SOA, and they are already proliferating inside most enterprises. In an SOA, Web services become business services with the ability to perform a particular function or access data dynamically. This presentation will discuss the four abilities that a registry provides for an SOA.

BIO: Luc Clement is director of product marketing, SOA Registry for Systinet. He is also cochair for the UDDI Specification Technical Committee. Formerly Microsoft UDDI Program Manager, Luc is well known in the UDDI community and has been heavily involved with the UDDI specification for several years.

WEB SERVICES SECURITY TRACK

WSS-1 Identity in SOA

Sekhar Sarukkai, Oblix

Tuesday, Febuary 15, 2005 9:00 A.M. – 9:50 A.M.

The mainstreaming of SOAs requires a more general approach to the notion of identities – beyond simply central management of people identities and into the realm of managing applications, devices, and other identities that represent entities that are first-class participants in this application network while also providing this as a pluggable service into the larger enterprise SOA. Enterprises should view identity as a service that is ubiquitously available and is a shared infrastructure service necessary for application networking, rather than as being managed by a server, such as an Authentication or Access server. While it makes architectural sense to consider an Identity service, there are business and related drivers that may force the need to deploy such an architecture.

BIO: Sekhar Sarukkai is currently a technical architect at Oblix. He was the original founder and CTO of Confluent Software, a leading Web services management company, which was acquired by Oblix in 2004. He holds a PhD in computer science from Indiana University.

Attend a FREE* Tutorial

Thursday, February 17, 2005, 8 – 11 a.m.

Using a Web Services Framework to Build SOA Applications

Ashish Larivee

Ashish Larivee, Novell

Service-oriented architecture (SOA) has quickly taken center stage as the primary development style of the next decade and beyond. Businesses of all types are preparing for the SOA revolution that promises consistency of process, reduction in duplicate work, ease of maintenance, service reusability and broad interoperability. The Web Services Framework (WSF) is the foundation that can deliver on the promise of SOA. Come learn about the components of an SOA including the core WSF standards. Attend this free Novell tutorial and learn about the future of SOA-style development, including legacy system enablement, platform interoperability, open source in SOA and building composite applications that leverage SOA services using Novell exteNd. In this session, we will create SOA application logic that orchestrates legacy services, JBoss4 Web services and MS.Net Web services. We will create Web services in Novell exteNd, Eclipse and Visual Studio respectively. We will then orchestrate these Web services and expose a single course, process level interface to public Web service consumers. Finally, we will cre-

Novell

* Free Tutorials with VIP Preregistration ONLY!

WSS-2 Securing Web Services with WS-Security

Dr. Jothy Rosenberg, Service Integrity

Tuesday, Febuary 15, 2005 10:00 A.M. – 10:50 A.M.

An up-to-date, comprehensive, and practical discussion of Web services security, and the first to cover the final release of new standards SAML 1.1 and WS-Security. Comprehensive coverage and practical examples of the industry standards XML Signature and XML Encryption will be presented.

BIO: Dr. Jothy Rosenberg is a serial entrepreneur. He is a founder and CTO of Service Integrity, a company that helps Web service operators see, measure, understand and fully leverage operational and business information flowing across their Web service networks. Prior to this venture, Jothy cofounded GeoTrust, the world's second largest certificate authority.

WSS-3 Anatomy of a Web Services Attack

Mamoon Yunus, Forum Systems

Tuesday, Febuary 15, 2005 4:00 P.M. – 4:50 P.M.

A broad range of new security threats is facing enterprises implementing XML Web services, leaving the enterprises open to financial risks, loss of property, and tarnished reputations. The basic rules of security – authentication, authorization, and auditing – no longer provide adequate security in the new world of straight-through processing paths into mission-critical systems. What's worse, WSDL documents provide a guide book to security exposure. Most attacks on traditional Web-based applications exploit weaknesses in HTML-enabled custom, or packaged, applications. However, hackers and other malicious users are quickly uncovering new techniques at the SOAP/XML data level that bypass HTML and target weaknesses in Web services programming, technology, and architecture. This session will outline the innovative techniques that hackers use to map out the vulnerabilities of an organization's network, and how Web server security must now complement Web services security in order to provide an adequate defense.

BIO: Mamoon Yunus, CTO of Forum Systems, was previously a global systems engineer for webMethods, where he developed business integration strategy and architecture for Global 2000 companies. He is an industry-honored CTO in advanced technological solutions for enterprise customers.

WSS-4 Using Mobile Phones as an SSO Authentication Device in SOA Solutions

Dr. Michael Juntao Yuan, University of Texas

Wednesday, Febuary 16, 2005 9:00 A.M. – 9:50 A.M.

Federated identity management across multiple single-sign-on domains is a major challenge for SOA-based solutions to fully realize its business potential. The traditional username/password combination is often too weak to protect the extremely sensitive single-sign-on credentials. The new-generation mobile phones could be used to identify and authorize users for SOA services. The device-based authentication scheme depends on not only "what you know" but also "what you own." This session will discuss new advances in Java-based mobile devices to interoperate with Sun's Liberty Alliance Services.

BIO: Dr. Michael Juntao Yuan is an author, developer, and software architect for end-to-end mobile software. He is a contributing editor to *JDJ* and a frequent contributor to many developer forums and publications. He is the author of two books. Michael has a PhD from the University of Texas at Austin and teaches information systems at the college level.

WSS-5 Building Intelligent Enterprises with Novell's Identity-Driven Computing

Ashish Larivee, Novell

Wednesday, Febuary 16, 2005 10:00 A.M. – 10:50 A.M.

Companies are now facing complexities dealing with issues such as regulatory compliance and security while still providing for company-wide collaboration between employees, partners, and suppliers. Identity systems are becoming a crucial component of applications, enabling developers to take advantage of a new set of services that know who you are, where you are, what you are trying to do, and can adapt to your changing business needs. Identity-driven computing addresses these problems by applying best practices learned from Novell's leadership in identity management for the management of people to all aspects of an enterprise, including servers, PCs, devices, applications, and even Web services.This presentation will outline identity-driven computing, describe the attributes of an identity-driven application, and discuss steps enterprises can take to make the transition to an identity-driven computing environment.

BIO: With more than nine years of experience in the software industry, Ashish Larivee has designed and developed many enterprise applications across a variety of platforms, including Microsoft, Lotus Notes/Domino, and J2EE. In 1999, Ashish joined SilverStream Software, acquired by Novell in July 2002, and has served in various roles in consulting, development, and technical marketing. In her current role, she helps define the strategy and product direction across Novell's Web Application Development Products.

WSS-6 XML Content Attacks

Girish Juneja, Sarvega, Inc.

Wednesday, Febuary 16, 2005 4:00 P.M. – 4:50 P.M.

This talk defines a new class of threats, XML Content Attacks, and differentiates these threats from more general Web services attacks and XML security-based attacks. These three related but distinct threat areas are explained. The talk covers XML Content Attacks with regard to tree-based parsing exploits related to coercive parsing, node-depth attacks, and DOM. XML grammar validation exploits such as schema poisoning and lax-content models are discussed, and why traditional schema validation cannot ensure content-model consistency. Web services attacks like WSDL scanning and parameter tampering (SQL Injection, SOAP array attack) are discussed – highlighting common mistakes made when applying message-level security (WS-Security).

BIO: Girish Juneja has more than 15 years' experience in the high technology industry with extensive product management, product strategy, engineering management, and technology marketing expertise. He is the cofounder of Sarvega. Since Sarvega's inception, Girish has led the Sarvega engineering and customer services organizations to develop Sarvega's industry-leading core XESOS technology and XML Networking products.

WSS-7 The Interoperability Challenge of Web Services Security Standards

Eve Maler, Sun Microsystems

Wednesday, Febuary 16, 2005 5:00 P.M. – 5:50 P.M.

The Web Services Interoperability Organization chartered its Basic Security Profile Working Group to develop an interoperability profile involving transport layer security, SOAP message layer security, encryption, signatures, and other security considerations. This session will discuss the interoperability challenges presented by current Web services security standards and the work of the WS-I Basic Security Profile. The session will highlight typical Web services security threats and countermeasures and the related design goals, usage conventions, and conformance testing of the soon-to-be-released Basic Security Profile.

BIO: Eve Maler is an XML standards architect at Sun Microsystems, where she coordinates Sun's involvement with Web services security standards such as SAML and the WS-I Basic Security Profile.

WSS-8 Transitioning Successfully to SOA and Web Services: Building the Infrastructure for SOA Growth

Dan Foody, Actional

Thursday, Febuary 17, 2005 9:00 A.M. – 9:50 A.M.

This session will address how to approach service-oriented architecture (SOA) management from a project-based level while still allowing room for future expansion and incremental growth to an enterprise-wide SOA. The session will provide valuable insight into how SOA management can help organizations ease the complexity of moving toward a loosely coupled environment.

BIO: As CTO at Actional, Dan Foody leverages his extensive experience in enterprise systems software toward designing robust and manageable service-oriented architectures. He is an active participant in the Web services standards community, including WS-I and OASIS, where he spearheads Actional's contributions on the OASIS Web Services Distributed Management Committee (WSDM).

RESERVE YOUR HOTEL ROOM NOW! AT THE SHERATON BOSTON HOTEL

THE OFFICIAL CONFERENCE HOTEL OF EDGE 2005 EAST

THE SHERATON BOSTON HOTEL
PRUDENTIAL CENTER
39 DALTON STREET
BOSTON, MA 02199
PHONE (617) 236-2000
FAX (617) 236-1702

Hotel Arrangements Are Easier Than Ever! Web Services Edge 2005 East has secured special group rates, well below regular rates, at one of the finest hotels in Copley Square, The Sheraton Boston Hotel. The hotel is directly next door to the Hynes Convention Center, and is connected via The Prudential Mall. To learn more about hotel savings, or to make your individual or group reservations please contact Meeting & Travel Consultants:
Toll free (866) 549-3985
Direct (732) 549-3985
Fax reservations to (732) 549-3987.
Make online reservations by Jan. 17, 2005
visit: http://www.mtcreservations.com

SPECIAL CONFERENCE RATES

Single	Double
$196	$196

Credit card information is required to guarantee reservations and expedite confirmation. Confirmations will be mailed directly from the hotel, time permitting. All changes and cancellations should be made directly through MTI.

DRIVING DIRECTIONS TO THE SHERATON BOSTON FROM LOGAN INTERNATIONAL AIRPORT

• Follow direction sign from terminals to Sumner Tunnel/Rt. 93 North • Take the Rt. 93 North ramp as you exit the Sumner Tunnel • Exit at Storrow Drive (26 B) • Follow Storrow Drive approximately 2 miles to the Fenway/Kenmore Exit (1st exit after Massachusetts Ave., on left) • Stay left as exiting Storrow Drive going towards Fenway • Continue to 1st set of lights, staying left, going onto Boylston Street • Go through 4 sets of lights on Boylston Street • The Hynes Convention Center's main entrance driveway will be on the right immediately after going through the 4th set of lights.

WEB SERVICES EDGE CONFERENCE PAST SPONSORS & EXHIBITORS

VISIT **www.sys-con.com/edge** FOR TIMES & SCHEDULES

The program, including topics and times, is subject to change. Please refer to www.sys-con.com for all updates.

SPECIAL INSERT: WEB SERVICES EDGE 2005
WWW.SYS-CON.COM/EDGE
FEBRUARY 15-17, 2005
JOHN B. HYNES CONVENTION CENTER • BOSTON, MA

International Web Services Conference & Expo

Register by JANUARY 21, 2005 SAVE Up To $300

For Assistance Tel: 201-802-3066 Fax: 201-782-9601

CONFERENCE & EXPO: February 15 – 17, 2005

John B. Hynes Veteran Memorial Convention Center • Boston, MA

THREE WAYS TO REGISTER FOR CONFERENCE

1) On the Web: Credit Cards or "Bill Me" Please make checks payable to SYS-CON Events
2) By Fax: Credit Cards or "Bill Me" 201-782-9601
3) By Mail: 135 Chestnut Ridge Road, Montvale, New Jersey 07645, Attention: Registration

Please note: Registrations are not confirmed until payment is received.

Please complete sections 1, 2, 3 and 4

1 YOUR INFORMATION (Please Print) ☐ Mr. ☐ Ms. ☐ Mrs.

First Name _____ Last Name _____

Title _____

Company _____

Street _____

Mail Stop _____

City _____

State _____ Zip _____ Country _____

Phone _____

Fax _____ E-Mail _____

2 PAYMENT METHOD: (Payment in full due with registration)

☐ Check or Money Order Enclosed (Registration confirmed upon receipt of payment)

Check # _____ Amount of Check $ _____

Charge my ☐ Visa ☐ MasterCard ☐ American Express ☐ Discover

Name on card _____

Card # _____ Exp. Date _____

Signature _____

Billing Address (if different from mailing address)

_____

3 PLEASE INDICATE YOUR CONFERENCE CHOICE:

Total Registration fee $_____

On Site	Before 12/31/04	Before 1/21/05	Before 2/11/05
☐ **3D: Three Day Conference** Good for all three days of the conference program, including keynotes, all conference sessions.	$1,195.00	$1,395.00	$1,495.00	$1,695.00
☐ **2D: Two Day Conference** Good for two days of the conference program, including keynotes, all conference sessions.	$995.00	$1,195.00	$1,295.00	$1,395.00
☐ **1D: One Day Conference** Good for one day of the conference program, including keynotes, all conference sessions.	$595.00	$695.00	$695.00	$795.00
☐ **VIP PASS** FREE Tutorials with VIP preregistration ☐ FREE - Novell Tutorial (Feb. 15) ☐ FREE - Developing Web Services Using Java Technology (Feb. 15) ☐ FREE - The Next Generation of Visual Studio (Feb. 16) ☐ FREE - Security, The New Reality (Feb. 17)	FREE	FREE	FREE	$75.00
☐ **EO: Expo Hall Only** Includes keynotes, expo presentations, and workshops	FREE	FREE	FREE	$50.00

SPECIAL GROUP RATES

Registering a group of four or more? Please call Jim Hanchrow at 201-802-3066, and receive an additional discount.

4

A. Your Job Title

- ☐ Senior Software Developer/Engineer/Architect/Analyst
- ☐ Director of IT/Development/Engineering
- ☐ Software Developer/Engineer/Architect
- ☐ Systems Architect/Engineer/Analyst
- ☐ Software Consultant/Analyst
- ☐ Project Manager/Team Leader
- ☐ Business Development
- ☐ General Manager/Division or Department Head
- ☐ CTO,CIO, Chief Architect
- ☐ CEO, President, Owner, Partner/Principal
- ☐ VP/Technical Director
- ☐ Network Administration/Manager/Consultant
- ☐ Web Developer/Programmer
- ☐ Database Administrator
- ☐ VP/Sales & Marketing Director /Product Management
- ☐ Other (please specify)_____

B. Business/Industry

- ☐ Agriculture/Mining/Oil/Gas
- ☐ Computer Hardware and Electronics
- ☐ Construction/Architecture/Engineering
- ☐ Consulting (non-computer)
- ☐ Education
- ☐ Banking/Insurance/Accounting
- ☐ Government (including Military)
- ☐ Healthcare/Pharmaceuticals/Biotech/ Biomedical
- ☐ Internet/Web/e-Commerce
- ☐ Manufacturing and Process (other than computer related)
- ☐ Media/Marketing/Advertising
- ☐ Non-Profit/Trade Association
- ☐ Real Estate/Legal
- ☐ Telecommunications
- ☐ Transportation/Rail/Sea/ Air/Ground
- ☐ Travel/Hospitality/Entertainment
- ☐ Utilities
- ☐ Wholesale/Trade/Distribution/ Retail
- ☐ Other (please specify) _____

C. Total Number of Employees at Your Location and Entire Organization (check all users that apply):

	Location	Company
10,000 or more	01 ☐	01 ☐
5,000 - 9,999	02 ☐	02 ☐
1,000 - 4,999	03 ☐	03 ☐
500 - 999	04 ☐	04 ☐
100-499	05 ☐	05 ☐
100 or less	06 ☐	06 ☐

D. Please indicate the value of software products and services that you recommend, buy, specify or approve over the course of one year:

- ☐ $10 million or more
- ☐ $1 million - $9.9 million
- ☐ $500,000 - $999,999
- ☐ $100,000 - $499,999
- ☐ $10,000 - $99,999
- ☐ Less than $10,000
- ☐ Don't know

E. What is your company's gross annual revenue?

- ☐ $10 billion or more
- ☐ $1 billion - $9.9 billion
- ☐ $100 million - $999 million
- ☐ $10 million - $99.9 million
- ☐ $1 million - $9.9 million
- ☐ Less than $1 million
- ☐ Don't know

F. Do you recommend, specify, evaluate, approve or purchase mainframe products or services for your organization?
01 ☐ Yes 02 ☐ No

G. Which of the following products, services, and/or technologies do you currently approve, specify or recommend the purchase of?

- ☐ Application Servers
- ☐ Web Servers
- ☐ Server Side Hardware
- ☐ Client Side Hardware
- ☐ Wireless Device Hardware
- ☐ Databases
- ☐ Java IDEs
- ☐ Class Libraries
- ☐ Software Testing Tools
- ☐ Web Testing Tools
- ☐ Modeling Tools
- ☐ Team Development Tools
- ☐ Installation Tools
- ☐ Frameworks
- ☐ Database Access Tools / JDBC Devices
- ☐ Application Integration Tools
- ☐ Enterprise Development Tool Suites
- ☐ Messaging Tools
- ☐ Reporting Tools
- ☐ Debugging Tools
- ☐ Virtual Machines
- ☐ Wireless Development Tools
- ☐ XML Tools
- ☐ Web Services Development Toolkits
- ☐ Professional Training Services
- ☐ Other [Please Specify]

_____

If you require special assistance covered under the Americans with Disabilities Act, please call 201-802-3066 by February 4, 2005.

CANCELLATIONS, SUBSTITUTIONS, REFUNDS

Fax written request to SYS-CON Registration 201-782-9601. Requests for refunds received prior to January 7, 2005 will be honored, less a 10% handling charge; requests received after January 7, 2005, and before February 4, 2005, will be honored less a 20% handling charge. No requests for refunds will be honored after February 4, 2005. Requests for substitutions must be made in writing prior to February 11, 2005. No one under 18 is permitted to attend. No warranties are made regarding the content of sessions or materials.

Speakers, sessions and schedule are subject to change without prior notice.

No solicitation by anyone other than official exhibitors, sponsors or marketing partners is permitted. Such behavior is cause for expulsion without refund.

SPECIAL INSERT: WEB SERVICES EDGE 2005
WWW.SYS-CON.COM/EDGE
FEBRUARY 15-17, 2005
JOHN B. HYNES CONVENTION CENTER • BOSTON, MA

Displaying Grouped Output with ColdFusion

A great tool for your arsenal

By Jeffry Houser

Knock! Knock! Hey [insert your name here], can you create a report of all the users who have registered on our site? Geography is really important to the sales folk, so make sure you categorize it by state. Oh, and can I have it done today?"

"Sure, I think we can do that."

The Problem

So, you've been asked to create a report on all the users who have registered on your site. The report has to be categorized by state, for the sales folk. Our result listing will probably look something like this:

The name of the state is listed first, and all the users from a certain state are listed underneath their state category. You won't be able to do much without reviewing the data in the database. This is what you're dealing with:

The two relevant tables are a user table and a state table. The user table contains all the relevant user information, such as the user's name and some address information. The state information is stored in the state table. The two tables are linked using a relationship on the StateID. By looking at the user's StateID, you can find the state that user exists in. Using a StateID, you can find all the users in a specified state.

This Works, But It Isn't Perfect

How do you parse the data to generate the listing? If you are new to the SQL language, you might take an approach like this:

```
<cfquery name="GetStates" datasource="january05">
 select * from states
</cfquery>


<cfoutput query="GetStates">

 <cfquery name="GetUsers" datasource="january05">
  select * from users where users.StateID = #GetStates.StateID#
 </cfquery>

 <cfif GetUsers.RecordCount NEQ 0>
  <b>#GetStates.State#</b><br>
  <cfoutput query="GetUsers">
  #GetUsers.FirstName# #GetUSers.LastName#<br>
  </cfoutput>
  <br>
 </cfif>

</cfoutput>
```

First, the code runs a single query to get the list of all states. The GetStates query is looped over, using cfoutput. The code inside the loop will retrieve all the users. If there are users for that given state, the state information is displayed along with the name of the user. This works and makes the boss happy, at least for now.

It Just Keeps Getting Better...

A couple weeks later, the boss gives a knock on your cubicle door. "Hey, you know that state/user report you created for me? I love the way that it works – that I can see all the users. Our membership is really increasing, but the page is running really slow. Can you fix it?

Let's start by analyzing the original code a bit more. How many queries will run on the page? One query will execute to retrieve the list of states. For each record returned from that query, another query will run. For U.S. states, you'll get at least 50 additional queries, more if you include areas such as the District of Columbia and Puerto Rico. With a small amount of data, this approach may not be a problem. However, if you have a lot of registrations on your Web site, this could start to become an inefficient approach. Fifty or more queries on a page are undesirable. I bet we can cut down the number of queries drastically.

If you join the user table and the state table in a single query, you can return all the data from the previous template in a single query:

```
<cfquery name="GetUsers" datasource="january05">
 select *
 from users, states
 where users.StateID = states.stateID
 order by state
</cfquery>
```

This query will return all the data that we need. It will return all the states that have users in them, along with all the same user data. You'll just need a way to loop over the data to create

the display. Here's one method:

```
<cfoutput query="GetUsers">
 <b>#GetUsers.State#</b><br>
 #GetUsers.FirstName# #GetUSers.LastName#<br>
</cfoutput>
```

Try this code in the browser. Unfortunately, with this approach the state is displayed for every user. The page is more cluttered and the functionality has changed. This won't do.

The state data should be displayed only when the state changes. You'll need to know the value of the state in the previous loops execution. You can store that in a variable. If the previous state value is different than the current state value, then the code has moved onto a new state; thus its name should be displayed. A variable, PreviousState, can hold the value of the previous state.

```
<cfset PreviousState = "">

<cfoutput query="GetUsers">
 <cfif PreviousState NEQ GetUsers.State >
  <br><b>#GetUsers.State#</b><br>
  <cfset PreviousState = GetUsers.State>
 </cfif>
 #GetUsers.FirstName# #GetUSers.LastName#<br>

</cfoutput>
```

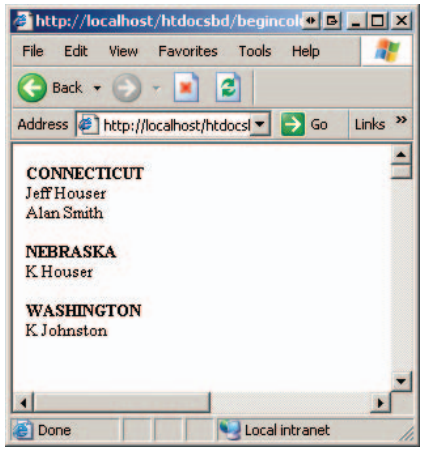


Figure 1: Result listing

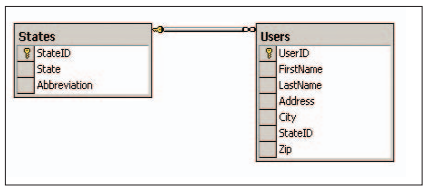


Figure 2: A user table and a state table

PreviousState will have to be initialized, which I do to an empty string. That makes sure that the first state is displayed. The rest of the code should be straightforward.

...And Better

The report code is starting to get pretty tight, but I know it can be even better. In the previous iteration, we had to write some code so we can tell when the state information has changed. We can avoid having to write this additional code by using the group attribute of the cfoutput tag and a nested cfoutput block. The group attribute is used to something called query grouping. Portions of the query are split up into groups, based on a column of the query. The outer loop executes once for each entry in the column. The inner loop will execute for each individual row. This is better explained with an example. In our problem, we want to group the query by state.

A PreviousState variable is no longer needed, since the grouping handles this functionality for us. When outputting over the query, the group attribute is used. It specifies the field that the code will use to loop over the group:

```
<cfoutput query="GetUsers" group="State">
  <b>#GetUsers.State#</b><br>
  <cfoutput>
   #GetUsers.FirstName# #GetUSers.LastName#<br>
  </cfoutput>
  <br>
</cfoutput>
```

The first thing inside the loop is a display of the state data. Then we have another cfoutput block. In versions of ColdFusion prior to CFMX, this was the only place where you could use nested cfoutputs. The nested cfoutput will loop over all the rows until the state changes. Conceptually this is very similar to the second method, except we are not writing additional code to keep track of when the code moves onto a new state.

Many of you may have heard about the group by clause in SQL. I want to specify that using the group attribute of cfoutput is not identical to using a group attribute in SQL. They are similar, but different. In SQL, the group by clause is used to run aggregate functions against subsets of data. For example, if we wanted to get the number of users in a particular state, we could use the grouped output in CFML with a loop counter to find the number of users. However it would be better to calculate that information in the query, like this:

```
<cfquery name="GetUsers"
datasource="january05">
 select states.state, count(users.lastname) as
UserCount
 from users left outer join states on (users.
StateID = states.stateID)
 group by state
 order by state
</cfquery>
```

The aggregate function, count, is used to return the number of users. Instead of returning the total number of records in the query, we get the state, and the number of users in that state. A great article on the group by clause can be found at www.4guysfromrolla.com/webtech/011202-1.shtml. More information about aggregate functions in SQL Server can be found at http://msdn.microsoft.com/library/default.asp?url=/library/en-us/tsqlref/ts_fa-fz_9yuk.asp.

Conclusion

Grouped queries can be a powerful tool in your arsenal. It will work especially well combined with the use of aggregate functions. For example, you could use the count function to find the total number of users in any state. ColdFusion will also support multiple levels of grouping with cfoutput. You may want to use this if you are working in an international application and want to separate the output by country, then state.

About the Author

Jeffry Houser has been working with computers for over 20 years and in Web development for over 8 years. He owns a consulting company and has authored three separate books on CF, most recently ColdFusion MX: The Complete Reference *(McGraw-Hill Osborne Media).*

jeff@instantcoldfusion.com

When was the last time your Intranet was updated?

Announcing the Macromedia® Web Publishing System. You build and manage the site. Users keep content fresh.

With Macromedia® Studio MX 2004, Contribute™ 3 and Contribute Publishing Services, you can affordably build, manage, and publish enterprise sites. And unlike the average content management system, this works. You determine who can edit and who can publish. Then users simply point and click to update any page. In no time, dated content will be a thing of the past. Learn more at **www.webpublishingsystem.com**

macromedia®
WEB PUBLISHING
SYSTEM

be the pilot!

FREE SETUP on Shared Hosting Accounts With ColdFusion MX Support
Use Promo Code CFDJ2004

INTERMEDIA.NET

WE DARE YOU TO TAKE A FREE TEST FLIGHT!

Managing technology that runs your business is a matter of trust and control. INTERMEDIA.NET gives you both.

TRUST. Since 1995 we have been providing outstanding hosting service and technology to our clients. Don't take our word for it... take theirs.

"The support and service that you offer are nothing short of golden. The high quality of your system and service for CF customers is something one could only ever dream of." – Claude Raiola, Director, AustralianAccommodation.com Pty. Ltd.

CONTROL. We give you instant control over your site, server and account configuration changes. No more submitting requests and waiting for someone else to take action. You are in control to pilot your business through its daily needs.

BE THE PILOT. Take a free test flight and see what our HostPilot™ Control Panel offers you beyond all others. Check out our SLA guarantees. To see more testimonials and to find out about our competitive advantages, visit our Web site at www.Intermedia.NET.

Managed Hosting • Shared Hosting • Microsoft Exchange Hosting

Call us at: 1.800.379.7729 • Visit us at: WWW.INTERMEDIA.NET | HostPilot™