MACROMEDIA CEO SPEAKS OUT Pg. 8

U.S. $8.99 (Canada $9.99)

ColdFusion Developer's Journal

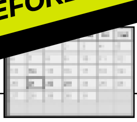
ColdFusionJournal.com

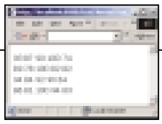
January 2003 Volume: 5 Issue: 1

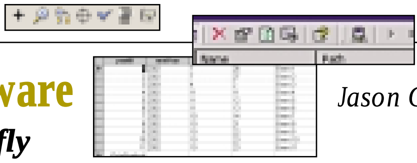
CFDJ: THE FIFTH YEAR

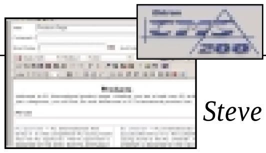
— FROM —
ColdFusion 4
— TO —
Contribute,
ColdFusion Developer's Journal Continues to Guide the CF Community

pg.36









RETAILERS PLEASE DISPLAY UNTIL MARCH 31, 2003

$8.99US $9.99CAN

02>

0 71486 02689 1

SYS-CON MEDIA

COLDFUSION Developer's Journal

editorial

editor-in-chief
Robert Diamond robert@sys-con.com
technical editors
Charles Arehart charlie@sys-con.com
Raymond Camden raymond@sys-con.com
editorial director
Jeremy Geelan jeremy@sys-con.com
executive editor
Jamie Matusow jamie@sys-con.com
editor
Nancy Valentine nancy@sys-con.com
associate editors
Gail Schultz gail@sys-con.com
Jean Cassidy jean@sys-con.com
assistant editor
Jennifer Stilley jennifer@sys-con.com

production

production consultant
Jim Morgan jim@sys-con.com
art director
Alex Botero alex@sys-con.com
associate art directors
Louis F. Cuffari louis@sys-con.com
Richard Silverberg richards@sys-con.com
Tami Beatty tami@sys-con.com

contributors to this issue

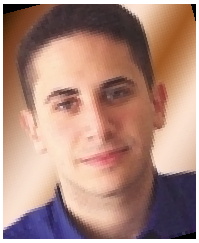
Charlie Arehart, Dennis Baldwin,
Mike Britton, Rob Burgess, Jason Clark,
Steve Drucker, Ben Forta, Hal Helms,
Simon Horwith, Alan McCollough

editorial offices

SYS-CON MEDIA
135 CHESTNUT RIDGE RD., MONTVALE, NJ 07645
TELEPHONE: 201 802-3000 **FAX:** 201 782-9600
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

Where Do We Go from Here?

In this month's issue, as we start the fifth year of ***ColdFusion Developer's Journal***, we've paused to look back at the past. Charlie Arehart wrote the retrospective cover story for this month, and I think you'll enjoy his trip down memory lane, starting back in version 1.0 of ColdFusion (1995), advancing to our first issue (1999), and stopping to look at all the major highlights that happened in between and since.



By Robert Diamond

How well do you know your CF trivia? Do you remember the fist logo? Do you remember when ColdFusion was the Cold Fusion Application Server (our editorial staff loved when they removed the space)? Would you like to know how ColdFusion moved on to the Java platform? For answers to these questions and lots of other facts, give it a read.

This issue isn't just about looking back, however, it's also about looking forward, and to that end, we've gotten our first (and hopefully not the last!) guest editorial straight from the top of Macromedia – CEO Rob Burgess. Many of us enjoyed his opening keynote this year at Macromedia DevCon, so we asked him if he would contribute a piece to the magazine, on that very same topic.

ColdFusion developers are, as he writes, "at the cutting edge of Internet development," and it's good to see and hear that from Macromedia directly. Their support is of course crucial as the company behind the technology, but what I personally find exciting about Macromedia as a whole is that there seems to be a genuine excitement for all that they do – in terms of innovation, integration, support, and playing a part in our daily development lives. I think that's well represented in Rob's article this month. Macromedia is always open to feedback, so let's keep telling them what they're doing right, and what they're doing wrong.

At ***CFDJ***, we're open to feedback as well, and I want to thank all of you who wrote to compliment the magazine's new, simpler, easy-to-read design. We're continuing to improve the layout, and the content, of course. This month, we've also improved the layout of the source code, another reader request. If there's anything else that we could be doing better, please don't hesitate to drop me a line.

We've always tried our best, since day 1, to represent the community as best we could, to help bring CFers together at industry events and at user groups, and to cover what's happening every month online and off. To that end, we've launched a new site this month that I'm quite proud of, having been involved in much of its development and construction (all running on ColdFusion MX of course!).

developer.sys-con.com is our hot new developer portal site. Check it out for its rich variety of technical meat and informed discussion threads, plus all the latest industry reporting, instant polls, and more....The whole thing is free and it's growing fast. There have already been a few popular ColdFusion discussions there, but it's not limited just to the world of CF – it's also got sections on all of the other major *i*-technologies from Java to WebSphere to WebLogic to Web services to Microsoft .NET, at the opposite side of the spectrum. The site aims to monitor the pulse of the developer community, so go check it out – I'll see you there.

About the Author

Robert Diamond is vice president of information systems for SYS-CON Media, and editor-in-chief of both **CFDJ** *and* **Wireless Business & Technology**. *Named one of the "Top thirty magazine industry executives under the age of 30" in* Folio *magazine's November 2000 issue, Robert holds a BS degree in information management and technology from the School of Information Studies at Syracuse University.*

robert@sys-con.com

SYS-CON MEDIA

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

business development
vp, business development
Grisha Davida grisha@sys-con.com
COO/CFO
Mark Harabedian mark@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 account manager
Megan Ring-Mussa megan@sys-con.com
associate sales managers
Carrie Gebert carrieg@sys-con.com
Kristin Kuhnle kristin@sys-con.com
Alisa Catalano alisa@sys-con.com

sys-con events
president, events
Grisha Davida grisha@sys-con.com
conference manager
Michael Lynch mike@sys-con.com
sales executives, exhibits
Michael Pesick michael@sys-con.com
Richard Anderson richarda@sys-con.com

customer relations
customer service representative
Margie Downs margie@sys-con.com
manager, jdj store
Rachel McGouran rachel@sys-con.com

sys-con.com
vp, information systems
Robert Diamond robert@sys-con.com
web designers
Stephen Kilmurray stephen@sys-con.com
Christopher Croce chris@sys-con.com

online editor
Lin Goetz lin@sys-con.com

accounting
financial analyst
Joan LaRose joan@sys-con.com
accounts receivable
Kerri Von Achen kerri@sys-con.com

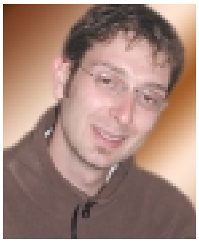
accounts payable
Betty White betty@sys-con.com

subscriptions
Subscribe@sys-con.com
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

Tales from the List

Flash discussions on the rise

There has been a growing trend in list discussions about Flash. This month, the list saw threads centered around the use of Flash as an alternative to writing DHTML Menus (hopefully cross-browser compatible), Flash on SSL, and resources for learning Flash – to name a few.



By Simon Horwith

As I mentioned in last month's DevCon summary (***CFDJ***, Vol. 4, issue 12), Flash MX/ColdFusion MX integration was a very hot topic at this year's conference. There's also been a noticeable growth in the number of Web-based business applications using Flash MX – based front-end GUIs. This month's installment of **Tales From the List** is all about the recent buzz surrounding Flash in the ColdFusion community.

Dave Sueltenfuss wrote to the CFDJList with his concerns about the use of Ben Forta's DHTML Menu tag in a site that uses frames. Frames, target support, and cross-browser compatibility were major concerns of Dave's, and he looked to the List for advice. Initial responses from Douglas Knudsen and Eugen Notter listed favorite sites of theirs (coolmenus.dhtmlcentral.com and greglanders.com to be specific).

James Gorman then added to the thread, noting that his company was currently exploring the use of Flash as a DHTML alternative, and wanted to get feedback from the list. Thus began a lengthy thread regarding Flash learning resources and the general benefits and drawbacks of using Flash to create site navigation and business logic front ends.

Dave replied immediately, noting that he believes there are issues with loading Flash SWFs in an SSL environment, which was immediately refuted by Tony Weeg who noted that Macromedia has addressed this issue (which simply means that the HTML code generated when publishing Flash movies is now able to reference a code base behind SSL).

Douglas Knudsen chimed in, stating that to make it work, "use HTTPS and the absolute path for CODEBASE and PLUGINSPAGE attributes, but use HTTP and the absolute path to your SWF for the EMBED SRC and the VALUE attribute of the PARAM tag," which is why Flash is not a choice solution for sites using SSL for all pages. As with most things, where there's a will, there's usually a way. Dave Deeds responded to the thread stating that his company is using Flash in an SSL/non-SSL environment – you simply need to put the SWF in both the http site and the https site to make this work.

Amit Patel then joined the discussion, inquiring about educational resources for learning Flash. Amit stated that he'd recently been on a job interview and in addition to ColdFusion, Flash was a strongly desired skill set. This is a growing trend among employers, and he's become interested in learning Flash. Aaron West recommended looking at some of the books that are out, as well as visiting the designer/developer center on Macromedia's site (www.macromedia.com/desdev/mx/flash/) and a few other sites he likes (www.flashkit.com and

—continued on page 26

About the Author

Simon Horwith, senior consultant at Fig Leaf Software in Washington, DC, has been using ColdFusion since version 1.5. He is a Macromedia-certified Advanced ColdFusion and Flash developer and is a Macromedia-certified instructor.In addition to administering the CFDJList List Serve and presenting at DC-area CFUGs, Simon is a contributing author to Professional ColdFusion 5.0 *(WROX) and to* ColdFusion MX - The Complete Reference *(McGraw-Hill), as well as technical editor of The ColdFusion 5.0 Certification Study Guide (Syngress).*

shorwith@figleaf.com

guest editorial

ColdFusion Developers Are in a Sweet Spot

At DevCon, in October, it was great to see the passion and commitment of the ColdFusion developer community. Since 2001, Macromedia has made a significant investment in ColdFusion, taking it to a whole new architecture with the MX release, partnering with IBM and Sun to support ColdFusion adoption, and integrating it with Macromedia Flash and Dreamweaver.

By Rob Burgess

We made the investment because we believe it matters. The ColdFusion development community has always been at the cutting edge of Internet development. As we move into 2003, ColdFusion developers are now in a unique position to help lead the development of the next wave of rich Internet applications.

When I speak with the CEOs and CIOs of our major customers, I consistently hear the same themes. They want to deliver Internet applications at a lower cost with faster time-to-market. They want to build on industry standards, and most important, they want the next generation of their Internet and intranet solutions to offer a radically better user experience for their employees, customers, and partners.

In the increasingly competitive global market, delivering great user experiences isn't a nice-to-have or simply a graphic design issue. Effective user experiences have become a business imperative because they are fundamental to how companies create and deliver value in today's economy, both online and off.

Take JetBlue Airways, for example. In December, they were named "Marketer of the Year" by *Advertising Age* because they led the industry in filling their planes in a very tough market for air travel. Their success is credited in large part to the unique in-flight experience they have created with innovations like TV for every passenger.

Online, the quality of the user experience goes far beyond graphic design or site usability. It is based on how well a site or application gives end users the ability to quickly, easily, and in a meaningful way, accomplish their goals.

Creating great online user experiences requires a deep understanding of the end users and brings together everything from user interface design to system architecture and application development. Great experiences are fundamentally the result of building systems from front end to back end so that they truly work for the user.

The experience imperative is critical for companies that want to maximize the impact of their software solutions on the bottom line. Forrester Research found that companies that redesign sites to support user goals can raise online revenues by 30% or more. They went on to estimate that investments in usability could deliver as much as a 245% return. In other words, bad user experiences cost businesses money in lost sales and lost productivity.

All this puts ColdFusion developers in a sweet spot. By taking advantage of the tremendous productivity of ColdFusion MX on standard J2EE servers and leveraging the unique capabilities of the entire Macromedia MX product family, I believe the ColdFusion developer community can be at the vanguard of creating the next generation of Internet applications.

Establishing ColdFusion as a key technology for businesses that want to accelerate their productivity on J2EE servers is critical. To continuously improve the quality of the user experience of an online solution, a development team needs to have the flexibility and productivity to respond rapidly to changing user requirements and needs. ColdFusion MX for J2EE finally gives development teams that flexibility, streamlining the development process, and increasing the velocity of innovation – ultimately driving long-term competitive advantage.

We are continually floored by the creativity and ingenuity of the Macromedia developer community. Already, we're seeing amazing examples of what can be created with the Macromedia MX product family, but as far as I'm concerned, we're just at the beginning. What's next? Well, that's in your hands.

About the Author

Rob Burgess is the chairman and CEO of Macromedia.

How May I Be of Web Service?

Is this one for real?

There is a phenomenon that appears to be unique to our industry – something I call *buzzword du jour*. On a regular basis a new buzzword (usually accompanied by a series of acronyms) appears on the front pages of the trade rags proclaiming to be "the future" – the message being "jump on board or become obsolete." This happens so frequently that many of us are becoming immune to it all, and in doing so are running the risk of missing something important when (or if) it happens. Which brings us to Web services.

By Ben Forta

What's the Deal?

If you are to believe the hype, Web services are more exciting than car chases, are the answer to world peace, can cure male-pattern baldness, will patch the ozone layer, can resurrect failing economies, will make you rich and desirable, may bring an end to the glut of reality TV shows… well, you get the idea.

So, the big question has to be, are Web services for real? Microsoft thinks so (but not everything Microsoft believes in is truly worthy of consideration – anyone remember Microsoft Bob?). Sun (thanks to applets) thinks so too, as does IBM (those folks who brought us OS/2, which did so well). The point is, just because the big guys say it's hot does not necessarily make it so. And yet everyone is talking and writing about Web services (and I myself must plead guilty here – I have mentioned Web services in passing in several columns since ColdFusion MX shipped). So, again, what's the deal?

It's Distributed Applications – Again

For starters, here's an explanation. Web services are simply a set of technologies that facilitate distributed computing; it is that simple. The idea is that an application can invoke another application (located anywhere, locally or remotely) which does some processing, possibly returning results when complete. That's all there is to it.

It's not a new idea. Distributed computing has been around in some way, shape, or form for a while now (heck, if you use ColdFusion's <CFHTTP> tag to invoke a remote page, you are using distributed computing).

So why the excitement this time around? There are a few reasons:

- ***Vendor support:*** It's been a while since everyone from Microsoft to Sun to IBM to Apple all talked the same language (for the most part). Web services are truly the first really viable solution to interoperability.
- ***Standards:*** Unlike other distributed computing technologies, Web services are built on existing standards and systems; no single vendor owns the space.
- ***Cost:*** The two prior bullets make this technology highly usable and accessible, and not at all cost prohibitive. Most developers can continue to use the platforms and languages they already use and don't need to throw out everything and start over.

Need a practical example? Try this one: you create an online store (I know that is so '90s, but bear with me). Customers visit your site and buy goods that you ship, using a carrier of your choice. Great. That is until a week goes by and no package shows up, so the customer revisits your site to track the order. You display the order information, complete with the shipper information and tracking number, allowing the customer to check on the delivery (and maybe, as you are so customer-service driven, you even provide a direct link to the shipper's site). Problem solved. Right?

Wrong! If you have an online store, the last thing you want is customers wandering off to other sites; you want them to stay and part with their hard-earned money. You never want to send them off to fedex.com or ups.com or any other site. Amazon.com doesn't do this – they show shipping information provided by those companies right in their own pages, the customer goes nowhere.

I have no idea what systems the shipping companies use, I don't know what software they use, and I have no access to it anyway (I am guessing the same is true for you). Amazon.com is a very significant player in the e-commerce space, the shipping companies are likely to work with them to provide them with access to this kind of data. But what about the rest of us? How could we do the same?

The answer is Web services (predictable, I know, sorry). If the shipping companies created a Web service that exposed tracking information, any application could submit a real-time request for the data, and then process or display it as needed right within their own pages. In other words, the client browser talks to your application; your application talks to the shipper via their Web service, retrieving the necessary information; and then your application displays it as needed.

Fantasy? Well, that depends on who you ship your products with. But some shipping companies have indeed made Web services available. It's not fantasy at all.

Web Services Lingo

Okay, so now that you know why you need Web services, let's review some of the lingo you'll need to use (drop these in conversations to sound like a seasoned pro):

- ***Web service:*** This is the application itself.
- ***XML:*** Web service requests are made in XML and returned data is also in XML.
- ***SOAP*** (Simple Object Access Protocol): The XML-based protocol used to send and receive Web service requests.
- ***WSDL*** (Web Services Description Language): An XML language used to document Web services; every Web service has an associated WSDL URL which lists the methods (functions) that may be invoked as well as what parameters they expect and what data is returned.
- ***UDDI*** (Universal Description, Discovery and Integration [of Web services]): A directory technology used to locate Web services (UDDI is not supported by ColdFusion in CFMX).

When you use a Web service you consume it, and when you create a Web service you publish it.

Consuming Web Services

Enough talk, let's try this thing. In ColdFusion, Web services are consumed using the <CFINVOKE> tag. <CFINVOKE> requires the URL to a Web service's WSDL, the name of the method to invoke (a single Web service may expose multiple methods), and any required parameters.

The following is a simple example. It invokes a Web service on my own Web site – this one returns my ColdFusion Tip-of-the-Day:

```
<!--- Get tip for today --->
<CFINVOKE
WEBSERVICE="http://www.forta.com/cf/tips/syndi-
cate.cfc?wsdl"
          METHOD="Get"
          DATE="#Now()#"
          RETURNVARIABLE="DailyTip">
```

As you can see, the Web service itself is specified using the URL to its WSDL file. ColdFusion retrieves and reads this file so as to know how to interact with the Web service, and how to call the method specified in the METHOD attribute. RETURNVARIABLE is the name of a ColdFusion variable to be populated with whatever results are returned. In this example my Web service returns a ColdFusion structure, and so DailyTip will be a structure containing the tip.

If you want to try this example you can use <CFDUMP> to display the results, like this:

```
<CFDUMP VAR="#DailyTip#">
```

You'll notice that there is no XML here, no SOAP, nothing beyond a simple tag call. ColdFusion hides all the complexity from you – it does not matter what is on the other end, what language it was written in, what data type it returns, what platform it runs on – none of that matters. All that is important is that the returned data is ready to use just like any other ColdFusion variable.

Here is another example, the same Web service call, but this time using a slightly different calling convention:

```
<!--- Get tip for today --->
<CFINVOKE
WEBSERVICE="http://www.forta.com/cf/tips/syndi-
cate.cfc?wsdl"
          METHOD="Get"
          RETURNVARIABLE="DailyTip">
   <CFINVOKEARGUMENT NAME="date"
                     VALUE="#Now()#"/>
</CFINVOKE>
```

As you can see, arguments may be specified as name=value pairs in <CFINVOKE>, or passed to nested <CFINVOKEARGUMENT> tags – the choice is yours, the end result is the same.

Let's look at a complete example. Alta Vista has a translation engine named Babel Fish – pass it text and a destination language and it returns the translated text to you. The Babel Fish service requires two pieces of information, the text to be translated, and the translation mode (from what to what, for example en_de to translate English to German). See Listing 1 for a complete working application. It prompts for text and a destination language and displays the translation.

As you can see, using Web services in ColdFusion is easy. I have no idea what the Babel Fish Web service is written in or what platform it runs on, nor do I care – the important thing is that I can invoke it and it just works. And that is the beauty of Web services.

Note: Looking for a Web service to experiment with? Try www.xmethods.net.

If you are using Dreamweaver MX you'll find that Web services are even easier to use. In Dreamweaver, open the Application window, click on the Component tab, and then change the dropdown to Web services. Dreamweaver will list any known Web services (the list will be empty if this is the first time you are doing this). Click the + button and you'll be prompted for the WSDL URL of a Web service – simply enter it, and click OK. Dreamweaver MX will retrieve the WSDL, process and parse it, and will then display a tree of Web service methods and arguments. Then all you need to do is drag the desired method into your ColdFusion code, and Dreamweaver will create the <CFINVOKE> call for you automatically.

Publishing Web Services

Now that you know how to consume Web services, let's look at how to create and publish them. ColdFusion MX makes creating Web services incredibly easy – they are created as ColdFusion Components (explained in detail in ***CFDJ,*** Volume 4, issues 6, 7, and 10).

Here is a simple ColdFusion Component. The file is named random.cfc, and it contains two methods: (1) Get returns a random number and (2) GetRange returns a random number within the specific range. (Okay, not the most glamorous examples, but they'll have to do). See Listing 2 for the code.

The CFC code itself is rather simple. Get accepts no arguments and returns whatever is generated by Rand(); GetRange() accepts two required numeric arguments and passes them to RandRange() returning the results. This CFC can be used within your ColdFusion code using <CFINVOKE>, <CFOBJECT>, or CreateObject().

So what makes this a Web service? The answer is the ACCESS level. Each of the <CFFUNCTION> tags has ACCESS="remote". Any ColdFusion Component methods with an ACCESS level of remote can be accessed remotely – as Web services; it's as simple as that.

What about the WSDL file you need? ColdFusion does that too. The WSDL URL is the URL to the CFC with ?wsdl appended to it. I saved my random.cfc in my test directory, so the WSDL URL is http://localhost:8500/test/random.cfc?wsdl.

Listing 3 is sample code showing this CFC being used as a Web service.

Of course this particular CFC could be invoked as a CFC rather than as a Web service (and probably should be as it is ColdFusion and it is local). But with no extra work, the ColdFusion Component can also be used by:

- Other ColdFusion servers
- .NET
- Java

- PHP
- Client-side Flash

ColdFusion, and CFCs in particular, makes publishing Web services painless and even fun. You don't have to worry about XML, SOAP, data-type conversions – ColdFusion does all that for you. In fact, as I demonstrated at DevCon in Orlando, you'll find it easier to create Web services for Java and .NET in ColdFusion than in Java or any .NET language!

Conclusion

Web services are all the rage right now. While the hype will die down a bit, all indications are that this is an important technology and one that will play an important role in application development now and in the foreseeable future. And best of all, as a ColdFusion developer, you have nothing to lose and lots to gain – it does not get much better than that.

About the Author

Ben Forta is Macromedia's senior product evangelist and the author of numerous books, including ColdFusion MX Web Application Construction Kit *and its sequel,* Advanced ColdFusion MX Application Development, *and is the series editor for the new "Reality ColdFusion" series. For more information visit www.forta.com.*

ben@forta.com

Listing 1

```
<!---
BabelFish.cfm
Description: ColdFusion BabelFish client.
Requires: ColdFusion MX
Usage notes: Just run it.
ben@forta.com
--->

<!--- Was a form submitted? --->
<CFIF IsDefined("FORM.string")>

   <!--- Yes, invoke service --->
   <CFINVOKE
WEBSERVICE="http://www.xmethods.net/sd/2001/BabelFishService.wsdl"
              METHOD="babelFish"
              RETURNVARIABLE="aString">
      <CFINVOKEARGUMENT NAME="translationmode"
                          VALUE="#FORM.lang#"/>
      <CFINVOKEARGUMENT NAME="sourcedata"
                          VALUE="#FORM.string#"/>
   </CFINVOKE>

   <!--- Display results --->
   <CFOUTPUT>
   <STRONG>Text:</STRONG> #FORM.string#<BR>
   <STRONG>Translation:</STRONG> #astring#<P>
   </CFOUTPUT>

</CFIF>

<!--- Form --->
<CFOUTPUT>
Enter some text and select the language
you'd like it translated into.<P>
<FORM ACTION="#CGI.SCRIPT_NAME#" METHOD="post">
Text: <INPUT TYPE="text" NAME="string">
<SELECT NAME="lang">
   <OPTION VALUE="en_fr">French</OPTION>
   <OPTION VALUE="en_es">Spanish</OPTION>
   <OPTION VALUE="en_de">German</OPTION>
   <OPTION VALUE="en_it">Italian</OPTION>
   <OPTION VALUE="en_pt">Portuguese</OPTION>
</SELECT>
<BR>
<INPUT TYPE="submit" VALUE="Translate">
</FORM>
</CFOUTPUT>
```

Listing 2

```
<!--- Random number component --->
<CFCOMPONENT>

   <!--- Get a random number --->
   <CFFUNCTION NAME="Get"
                RETURNTYPE="numeric"
                OUTPUT="no"
                ACCESS="remote">
      <CFSET Randomize(GetTickCount())>
      <CFRETURN Rand()>
   </CFFUNCTION>

   <!--- Get a random number in a specified range --->
   <CFFUNCTION NAME="GetRange"
                RETURNTYPE="numeric"
                OUTPUT="no"
                ACCESS="remote">
      <CFARGUMENT NAME="randlo"
                   TYPE="numeric"
                   REQUIRED="yes">
      <CFARGUMENT NAME="randhi"
                   TYPE="numeric"
                   REQUIRED="yes">
      <CFSET Randomize(GetTickCount())>
      <CFRETURN RandRange(ARGUMENTS.randlo, ARGUMENTS.randhi)>
   </CFFUNCTION>

</CFCOMPONENT>
```

Listing 3

```
<CFINVOKE WEBSERVICE="http://localhost:8500/test/random.cfc?wsdl"
           METHOD="get"
           RETURNVARIABLE="result1">

<CFINVOKE WEBSERVICE="http://localhost:8500/test/random.cfc?wsdl"
           METHOD="getRange"
           RETURNVARIABLE="result2">
   <CFINVOKEARGUMENT NAME="randlo" VALUE="1"/>
   <CFINVOKEARGUMENT NAME="randhi" VALUE="100"/>
</CFINVOKE>

<CFOUTPUT>
Random: #result1#<BR>
Random Range: #result2#<BR>
</CFOUTPUT>
```

The Flash MX Calendar Component

By Mike Britton

If you've spent time dabbling in Flash lately, you've probably encountered Flash MX components. Components are predefined Movie Clip objects designed to add functionality to Flash applications, and they're being applauded for encouraging standardization and consistency in Flash GUI design.

Combined with Flash Remoting and ColdFusion, components represent an exciting chance for ColdFusion developers to connect their applications to sophisticated Flash user interfaces. For the purpose of this article, I'll assume you have ColdFusion MX running, with Flash Remoting Services installed. For instructions on Remoting Services, see Macromedia's Web site.

As you can see, the Flash calendar (see Figure 1) is a compact version of the traditional wall calendar. It's a sophisticated component with a lot under the hood, but for the purpose of this article I won't be discussing the component code. Let's just say it "represents a major step toward making rich applications a popular Web front-end solution," and leave it at that. Despite its unassuming appearance, the calendar's straightforward design is perfect for integration into all kinds of interfaces.

We'll be using the calendar component (called FCalendar in Flash-ese) to call methods in a server-side ColdFusion Component (CFC) and return values back to Flash. We'll do this with Flash Remoting, a great new way to connect Flash to ColdFusion MX.

The Event Calendar

We're building a very basic "Event Calendar" application. Our application will use a database to look up scheduled events. When an event is returned, the text for that event (a journal or Web log entry, for instance) will be displayed in a scroll box.

Our Event Calendar is visually very simple (see Figure 2). On the left side is the calendar. This instance of FCalendar has been given the name "cal" on the Flash main stage. On the right side of the screen is a text field equipped with a scroll bar component. This scrolling text field displays the event for a selected day.

When the user selects a day on the calendar, Flash tells ColdFusion MX to call a method on the server. This method returns a value and displays it to the user in the Event Display side of our Event Calendar application.

In a real-world scenario and with very little modification (maybe some custom skinning), this could serve as an interface for Web logs and online scheduling front ends. At the very least, giving this component a good workout could become a useful addition to your toolbox.

What You'll Need

- ColdFusion MX Server
- Flash Remoting Services
- Flash MX
- Flash MX UI Components Set 2
- The source files for this project

The Data Source

The data source for this application, *fcal*, contains just one table, *dates* (see Figure 3). In dates are five fields: postID, textYear, textMonth, textDay, and textEvent. Field postID is set to AutoNumber, while textEvent is a memo field. All other fields are text, with default Access settings for text fields.

Note the strange, seemingly nondescriptive field names. In ActionScript we'll be working with a handful of variables, so it would be to our benefit to name our fields in a way that's easily distinguishable from the ActionScript. This is why I've prefixed my fields with *text* – I just find it more intuitive for demonstration purposes to do this. Your future table fields will, of course, be descriptive of the information they contain.

fcal.dsn: Field Names

- ***postID:*** A number to use for sorting and record identification purposes inside the application.
- ***TextYear:*** The year an event takes place.
- ***TextMonth:*** The month an event takes place.
- ***TextDay:*** The day an event takes place.
- ***TextEvent:*** The event itself; this is a memo field and can be as long as you want.

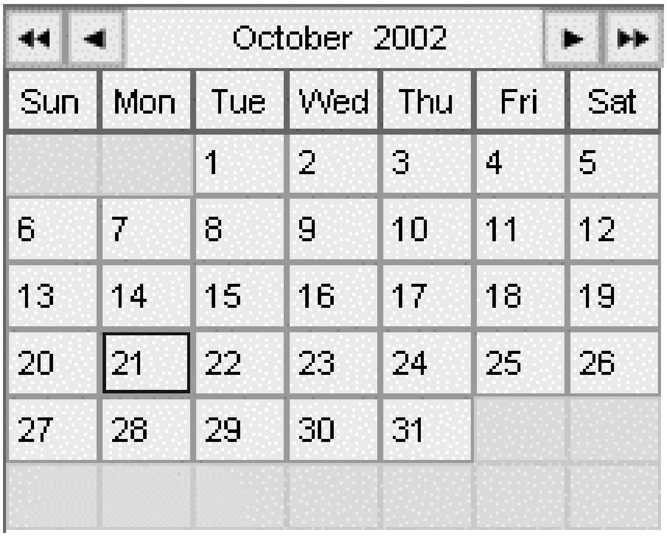
Obviously, if you were building something for use in a large Web application, you'd want to make sure your database design is well planned. The database we're using was created for demonstration purposes, and shouldn't be considered ideal. In fact, one of the great things about following along with an article like this is the virtual guarantee that you'll find a different way to do what I'm going to demonstrate here. Regardless, you will be able to use code from this article to spin off your own ideas for your clients.

The CFC

Let's open the CFC we'll be using to generate our queries to the Access data source (see Listing 1).

Method 1: getAllDates

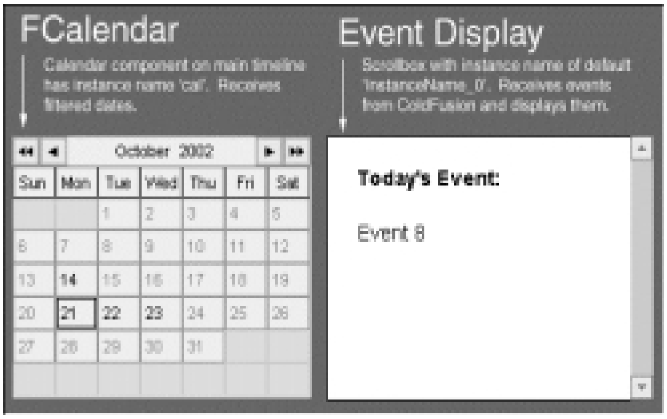
```
<cffunction name="getAllDates" access="remote" returntype="query">
   <cfquery name="getDates" data source="fcal" dbtype="ODBC">
   select postID, textYear, textMonth, textDay, textEvent
   from dates
   </cfquery>
   <cfreturn getdates>
</cffunction>
```



October 2002

Sun	Mon	Tue	Wed	Thu	Fri	Sat
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

Figure 1: The Flash calendar is a compact version of the traditional wall calendar



FCalendar

Calendar component on main timeline has instance name 'cal'. Receives filtered dates.

Event Display

Scrollbox with instance name of default 'instanceName_0'. Receives events from ColdFusion and displays them.

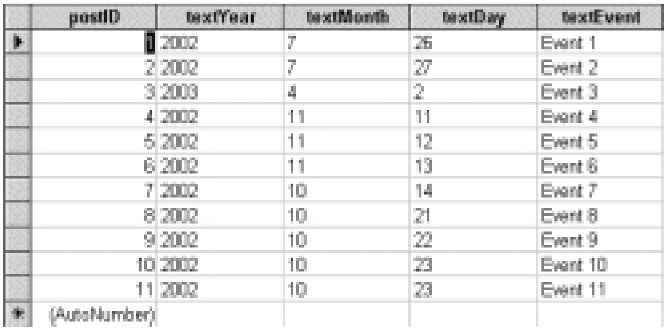
October 2002

Sun	Mon	Tue	Wed	Thu	Fri	Sat
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

Today's Event:

Event 8

Figure 2: The GUI for our "Event Calendar" Flash Remoting Application



postID	textYear	textMonth	textDay	textEvent
1	2002	7	26	Event 1
2	2002	7	27	Event 2
3	2003	4	2	Event 3
4	2002	11	11	Event 4
5	2002	11	12	Event 5
6	2002	11	13	Event 6
7	2002	10	14	Event 7
8	2002	10	21	Event 8
9	2002	10	22	Event 9
10	2002	10	23	Event 10
11	2002	10	23	Event 11
(AutoNumber)				

Figure 3: The dates table, the data source for fcal

This is the method we'll be calling to populate the calendar with dates. We'll be doing this using Flash Remoting, but let's focus on the CFC for a little longer.

Method 2: getTodaysEvent

```
<cffunction name="getTodaysEvent" access="remote" returntype="query">
   <cfargument name="postYear" type="string">
   <cfargument name="postMonth" type="string">
   <cfargument name="postDay" type="string">
   <cfquery name="q_allTodaysEvents" data source="fcal"
```

```
      dbtype="ODBC">
            SELECT textEvent
            FROM dates
            where textYear = '#trim(postYear)#' and textMonth =
              '#trim(postMonth)#' and textDay = '#trim(postDay)#'
    </cfquery>
    <cfreturn q_allTodaysEvents>
</cffunction>
```

Three arguments will be passed to this method from Flash: year, month, and day. These values will have initially been loaded into Flash (and the calendar) via function getAllDates. These arguments will be used to find a record in table dates to match these values.

Method 3: getEvent

```
<cffunction name="getEvent" access="remote" returntype="query">
    <cfargument name="postYear" type="string">
    <cfargument name="postMonth" type="string">
    <cfargument name="postDay" type="string">
    <cfquery name="q_allEvents" data source="fcal" dbtype="ODBC">
            SELECT textEvent
            FROM dates
            where textYear = '#trim(postYear)#' and textMonth =
    '#trim(postMonth)#' and textDay = '#trim(postDay)#'
    </cfquery>
    <cfreturn q_allEvents>
</cffunction>
```

This method will be called to retrieve a date based on three arguments passed to it by Flash: year, month, and day. It may seem weird to have another query for the same purpose (that being to return a single textEvent value), but Flash Remoting allows you to call a CFC directly from ActionScript, then receive its value with another ActionScript function bearing the calling function's name, only suffixed with '_result'. You will understand this better as we ease into the ActionScript, where the real "action" begins...

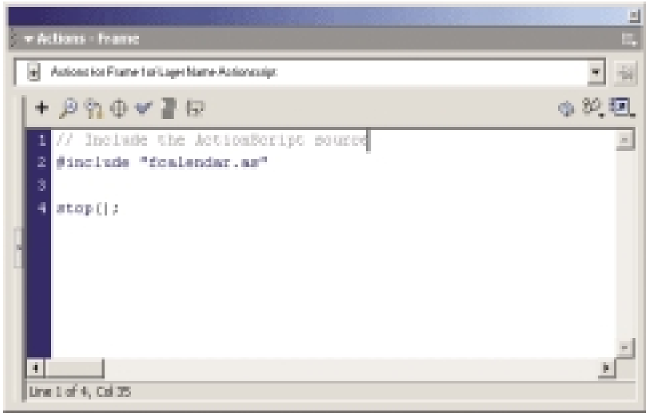
fcalendar.as

Once again, I recommend following along with the ActionScript source files provided with this project. Things will make more sense if you can test the application a few times in the Flash testing environment to see what's going on.

To make our ActionScript work in the Flash movie, we simply include fcalendar.as on the root timeline (see Figure 4). Any ActionScript in the file will behave like it was coded into the Flash ActionScript Panel. Besides making us feel clever, this can be an easier way to manage ActionScript files because it allows you to code ActionScript without Flash running.

If you're not familiar with ActionScript, it looks a lot like JavaScript and is used in Flash much the same way JavaScript is used to access browser-specific objects like Document, Window, and Anchor. Some familiarity with JavaScript will take some of the pain out of the learning curve.

Now let's take a look at the code. To ease reading of the ActionScript, I've included comments. You will see my instructions duplicated there.



```
// Include the ActionScript source
#include "fcalendar.as"

stop();
```

Figure 4: The ActionScript

Step 1: Connect to Flash Remoting
Create a new date object. We're going to use this date:

```
myDate = new Date();

trace("Today's date, set in Flash: " + (myDate.getMonth() + 1) + "/" +
myDate.getDate() + "/" + myDate.getFullYear());
```

As you may know, *trace()* outputs values to the Flash MX output panel. This is useful in understanding how variables are output in Flash.

```
// Include the Required NetService class files
// For debugging purposes only - comment when not in use:
#include "NetDebug.as"
// Call Flash Remoting's Netservices.as, required to establish gateway
connection to the server
#include "NetServices.as"

// Connect to the Flash Remoting service
if (isGatewayOpen == null) {
    isGatewayOpen = true;

    // Make the Gateway connection to CF MX Server
    NetServices.setDefaultGatewayUrl("http://localhost:8500/flash
    services/gateway");
    gatewayConnnection = NetServices.createGatewayConnection();

    // Initialize a variable to contain the data passed back
through fcalendar.cfc
    flash_dates = gatewayConnnection.getService("cfc.fcalendar",
    this);

    // Access method getgetAllDates() in fcalendar.cfc
    flash_dates.getAllDates();
}
```

Step 2: Initialize variables and populate the calendar with dates from the query.

getAllDates_Result() is called automatically by Remoting after the initial call to getAllDates.

A number of things happen here (see Listing 2):

- A recordcount is established called theLength. Not very original, but it'll do.
- A new object, dateFilter, is created to contain the values passed back from fcalendar.cfc.
- The query data is loaded into datefilter's isSelectable property.
- The datefilter object is passed to the calendar component's setDateFilter method in Flash.
- Method getTodaysEvent in fcalendar.cfc is called (through the putTodaysEvent function at the end of getAllDates_ Result), and returns the corresponding event for the date the user selects.

Step 3: Write functions to receive user input and retrieve "events" from table date's textEvent field.

Functions must exist to accomplish the tasks shown in Listing 3:

- Execute method getEvent, which returns the textEvent value from our database table dates, corresponding to the date variables sent to it.
- Populate the scrollbox (_root.InstanceName_0) first with a "Loading…" message, then with a selected date's textEvent field (from table dates) after a recordset has been returned from the CFC's method getEvent.
- Populate the scrollbox first with a "Loading…" message, then with today's textEvent field after today's date is passed to the getTodaysEvent method .

Conclusion

If you've followed along carefully, you should be able to test the demo application either inside the Flash MX authoring environment, or from a browser. I recommend testing inside Flash MX initially – Remoting works in the Flash testing environment, too. The debugging information in the Flash Output Panel will tell you if anything's wrong, or display output (trace() functions) in the ActionScript that will show you what's going on in the code.

I hope this article has shown you how the Flash MX Calendar component can enhance your Web applications. The techniques I've described here can be leveraged to build functionality into your own sites. With a little experience and some familiarity with OOP concepts, it's likely that more Web developers will begin using Flash to enhance the functionality of their apps.

About the Author

Mike Britton is a Web developer for INDUS Corporation in Research Triangle Park, North Carolina.

mbritton72@yahoo.com

Listing 1

```
<cfcomponent>
    <cffunction name="getAllDates" access="remote" returntype="query">
           <cfquery name="getDates" data source="fcal" dbtype="ODBC">
           select postID, textYear, textMonth, textDay, textEvent
           from dates
           </cfquery>
           <cfreturn getdates>
    </cffunction>
    <cffunction name="getTodaysEvent" access="remote"
      returntype="query">
           <cfargument name="postYear" type="string">
           <cfargument name="postMonth" type="string">
           <cfargument name="postDay" type="string">
           <cfquery name="q_allTodaysEvents" data source="fcal"
             dbtype="ODBC">
                      SELECT textEvent
                      FROM dates
                      where textYear = '#trim(postYear)#' and textMonth
                 = '#trim(postMonth)#' and textDay = '#trim(postDay)#'
           </cfquery>
           <cfreturn q_allTodaysEvents>
    </cffunction>
    <cffunction name="getEvent" access="remote" returntype="query">
           <cfargument name="postYear" type="string">
           <cfargument name="postMonth" type="string">
           <cfargument name="postDay" type="string">
           <cfquery name="q_allEvents" data source="fcal"
              dbtype="ODBC">
                      SELECT textEvent
                      FROM dates
                      where textYear = '#trim(postYear)#' and textMonth
                 = '#trim(postMonth)#' and textDay = '#trim(postDay)#'
           </cfquery>
           <cfreturn q_allEvents>
    </cffunction>
</cfcomponent>
```

Listing 2

```
function getAllDates_Result(result) {
    // The total number of records returned
    theLength = result.items.length;
    // The dateFilter object will be passed to the calendar and filtered
    dateFilter = new Object();
    // Load the result set from the ColdFusion query into the date
        filter object's isSelectable property
    dateFilter.isSelectable = function(comp, dt) {
           // getDate method - returns the day of the month (an inte
          ger from 1 to 31) of the specified Date object according
          to local time
           var day = dt.getDate();
           // getMonth method - returns the month (0 for January, 1
              for February, and so on) of the specified Date object
           var month = dt.getMonth();
           // Include the events as selectable
           var year = dt.getFullYear();
           for (i=0;  i<theLength;  i++) {
                      if (day == result.items[i].textDay && year ==
                      result.items[i].textYear && (month ==
                      result.items[i].textMonth - 1)) return true;
           } return false;
    }

    // Load the resulting dateFilter object into FCalendar instance
           *cal* by calling its setDateFilter method
           cal.setDateFilter(dateFilter);
    length=result.items.length;
    trace("Length: " + length);
    // Sets scrollbox to display html-formatted text
    _root.InstanceName_0.html = true;
    // Initialize variables for today's date to pass them to
     getTodaysEvent
    tday = myDate.getDate();
    tMonth = myDate.getMonth() + 1;
    tYear = myDate.getFullYear();

    // Calls flash_dates.getTodaysEvent
    _root.putTodaysEvent(tYear, tMonth, tDay);
}
```

Listing 3

```
// Executed by calendar when user clicks on a selectable day
function putDate() {
    // Set the getSelectedItem property of calendar instance *cal* to
    variable
    aDate = cal.getSelectedItem();
    // create a valid date object
    aDate = new Date(aDate);
    // set the year
    aYear = aDate.getFullYear();
    // set the month, add 1 since the month is zero indexed (non-CF array)
    aMonth = aDate.getMonth() + 1;
    // set the day
    aDay = aDate.getDate();
    // Debug information
    trace("Year: " + aYear + " Month: " + aMonth + " Day: " + aDay);
    trace(cal.getSelectedItem());
    // Calls function that calls cfc method getEvent
    _root.putEvent(aYear, aMonth, aDay);
    // Displays *Loading...* message for user in scrollbox
    _root.content_feed_display = "<br>Loading...";
}
// Passes values to getEvent method in CFC
function putEvent(year, month, day) {
    flash_dates.getEvent(year, month, day);
}
// Retrieves content from getEvent method; called automatically by
Remoting after initial call to getEvent
function getEvent_Result(result) {
    trace("Year: " + aYear + " Month: " + aMonth + " Day: " + aDay);
    // Make sure scrollbox is set to display html-formatted text
    _root.InstanceName_0.html = true;
    // Output the textEvent to scrollbox
    _root.InstanceName_0.htmlText = "<br><b>Event for " + aMonth + "/"
+ aDay + "/" + aYear + "</b><br><br>" + result.items[0].textEvent;
}
// Passes values to getTodaysEvent method in fcalendar.cfc
function putTodaysEvent(year, month, day) {
    flash_dates.getTodaysEvent(year, month, day);
}
/* Retrieves content from getTodaysEvent method; called automatically by
Remoting after initial call to getTodaysEvent */
function getTodaysEvent_Result(result) {
    trace("Year: " + tYear + " Month: " + tMonth + " Day: " + tDay);

    // Make sure scrollbox is set to display html-formatted text
    _root.InstanceName_0.html = true;
    // Ouput selected day's event in the scrollbox
    _root.InstanceName_0.htmlText = "<br><b>Today's Event: </b><br><br>" +
result.items[0].textEvent;
}
```

Web Services in a Flash

Access distributed data using Flash Remoting

Of course the latest buzzword in the world of Internet applications is Web services. So we're assuming that you already know what a Web service (WS) is and we're not going to worry about creating one. This article will focus on accessing a Web service using Flash Remoting, and displaying the information in a rich Flash UI.

By Dennis Baldwin

We'll be looking into the emfusion.com Web service which supplies the latest news and information to many organizations. The emfusion engine gathers up-to-the-minute content and makes it accessible to developers and Web administrators through a WS. Although there are many methods available through the emfusion WS, we'll be focusing on two basic ones that allow us to access the necessary data for display in the Flash movie.

The emfusion WS we'll be accessing is a scaled-down version for developers to access and test with the application that will be discussed in this article. The Web Services Description Language (WSDL) file is located at www.emfusion.com/ws.cfc?wsdl. Feel free to access this file through the browser to see the basic structure of the file and the methods we'll be accessing. Let's get started!

Overview of the Application

This article assumes that you have the Flash Remoting components installed. If you don't, then you can download them for free from Macromedia's Web site: www.macromedia.com/software/flashremoting/downloads/components. This will install the necessary documentation, ActionScript files, and even the NetConnection debugger, which can be useful for troubleshooting your Flash applications. We won't cover the NetConnection debugger but if you're interested in learning more about it, then feel free to check out my article, "Get Connected with Flash Debugging" in the October issue (***CFDJ***, Vol. 4, issue 10).

The Flash UI will be responsible for displaying data supplied from the WS. We're going to display a list of news categories and titles supplied by the WS. The initial Flash Remoting call will be made to the getCategories method. This will return a recordset of category IDs and category names, which will be used to dynamically display a list of PushButton components across the top of the Flash movie.

For the purpose of this article the news categories will be limited to U.S., Sports, and Religion. Depending on which category button is pressed, the corresponding category ID will be passed to the getNews method of the WS. The response will again be sent as a recordset and parsed inside of Flash. The news recordset will contain lots of data such as title, date, author, source, and URL fields. For our application we will be accessing only the title, date, and URL of each article. The values will be used to duplicate a movie clip for each record and then display it within the Flash ScrollPane component.

Now that we have a basic understanding of how the application will work, let's take a look at the communication process. Everything will start from the Flash movie where the user will interact and perform actions which, in turn, will make the necessary method calls to the WS. Figure 1 illustrates the communication process between Flash and the WS.

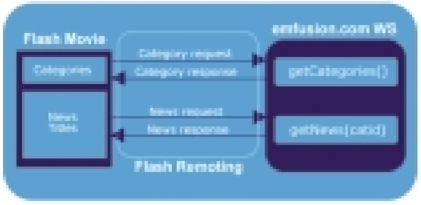
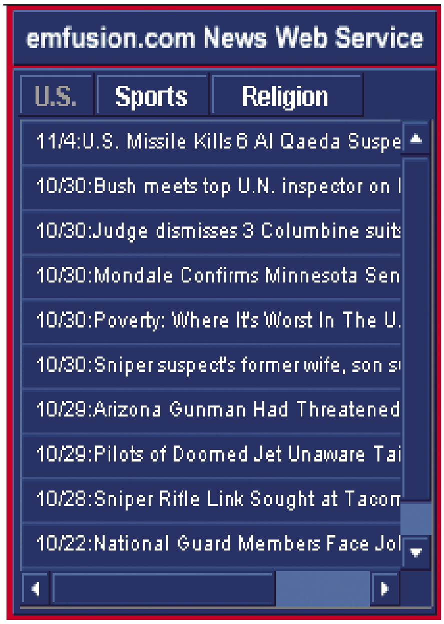


Figure 1: Communication between Flash movie and Web service via Flash Remoting


emfusion.com News Web Service

11/4:U.S. Missile Kills 6 Al Qaeda Suspe
10/30:Bush meets top U.N. inspector on I
10/30:Judge dismisses 3 Columbine suit
10/30:Mondale Confirms Minnesota Sen
10/30:Poverty: Where It's Worst In The U.
10/30:Sniper suspect's former wife, son s
10/29:Arizona Gunman Had Threatened
10/29:Pilots of Doomed Jet Unaware Tai
10/28:Sniper Rifle Link Sought at Tacom
10/22:National Guard Members Face Jol

Figure 2: The finished application in its initial state

Before we actually dig into the code, let's take a quick look at the completed product to get a better idea of how it functions. If you have access to a Web browser check out the following URL: www.emfusion.com/cfdj/emfusion_ws.html. You will also be able to download all the source code from this page. When the URL is first accessed, the Flash movie makes two Remoting calls: one to the getCategories method and one to the getNews method of the WS. This is what initializes our application and sets it to its default state. We initially pass an ID of 1 to the getNews method and grab the U.S. news. This will happen each time the application is accessed or the browser is refreshed. Figure 2 is a screen shot of the finished application.

Digging into the Code

Due to space constraints we're not going to cover all of the Flash ActionScript code line by line. For more code detail feel free to download the source files at www.emfusion.com/cfdj/emfusion_ws.zip.

When the application is initialized the init code runs:

```
If(inited==null) {
inited = true;
NetServices.setDefaultGatewayURL("http://www.emfusion.com/flashservices/ga
teway");
    gw = NetServices.createGatewayConnection();
    // create a reference to our Web Service cfc
    newsService = gw.getService("ws", this);
    // call the getCategories method of the WS
    newsService.getCategories();
    // grab the initial news headlines
    newsService.getNews(1);
    // more code down here
}
```

This happens only once per each instance of the application. If the browser is refreshed or the user leaves and comes back to the page, the init code will run again. This code segment handles establishing the connection with the Flash Gateway creating an object reference to the emfusion WS. This object reference is known as a service object and will then be used to make method calls to the WS.

The first parameter passed to this method is the name of the service, which is "ws". The second parameter of the getService method is the default responder. This is an object that will handle the response sent back from the WS. The value "this" used for the second parameter tells Flash Remoting that objects in _level0 or _root of the movie will handle responses sent from the WS.

In the code listing above, you can see that once a connection with the gateway is established, the getCategories and getNews methods of the WS are called. In Flash Remoting, each time a method of the WS is called, a default responder must exist to handle the response from the service. By default, Macromedia recommends using the format of methodName_Result. So the default responders for the getCategories and getNews methods would be getCategories_Result and getNews_Result respectively.

When the getCategories method is called, there are no parameters required by the Web service. So a basic call can be made to the method and then a list of category IDs and names will be sent to the caller, in our case Flash. The category information is passed to Flash as a RecordSet object. Therefore, we should be able to access the field names and values of each record using the Flash Remoting RecordSet methods. A truncated version of the getCategories responder is listed below:

```
function getCategories_Result(result) {
    // get the number of categories
    var len = result.getLength();
    // initial x position where we'll place the first cat clip
    var currx = 5;
    // loop and place the clips next to each other
    for(var i=0; i<len; i++) {
        var cat_id = result.getItemAt(i).cat_id;
        var cat_name = result.getItemAt(i).cat_name;
        var clip_width = parseInt(cat_name.length) * 10;
        var clip = "cat_" + cat_id;
        // duplicate the master button to create the cat buttons
        duplicateMovieClip("cat_btn", clip, i);
    }
    // more code down here
}
```

The RecordSet result is passed into the responder and the information is accessed and handled accordingly. We can access the number of records returned from the query and then duplicate the PushButton component as many times as necessary. The cat_id and cat_name fields are also accessed and used to populate the PushButtons. For demonstration purposes this query will return only three records, so we will have three buttons that display the values U.S., Sports, and Religion.

Immediately after the getCategories method is called, we make a call to getNews. Multiple requests/responses can happen at the same time, which makes Flash Remoting extremely powerful. By default, we pass a category ID of 1 as a parameter in the getNews method. This tells the WS to give us all news content that corresponds with U.S. news. A different responder exists to handle the news RecordSet object passed from the WS to Flash:

```
function getNews_Result(result) {
    // more code up here
    // get the number of news records
    var len = result.getLength();
    // loop through the news records and grab
    // the title, link, and date of each arti-
cle
    for(var i=0; i<len; i++) {
            var title =
result.getItemAt(i).news_title;
            var link =
result.getItemAt(i).news_link;
            // create a date object so we can
access the necessary values
            var news_date = new
Date(result.getItemAt(i).publish_dt);
            var display_date =
news_date.getMonth() + 1 +"/" +
news_date.getDate();
            // call the method to duplicate the
title clips and populate the data
            setTitleText(title, link,
display_date, i);
    }
    // refresh the scroll pane with the new
contents
    news_sp.refreshPane();
}
```

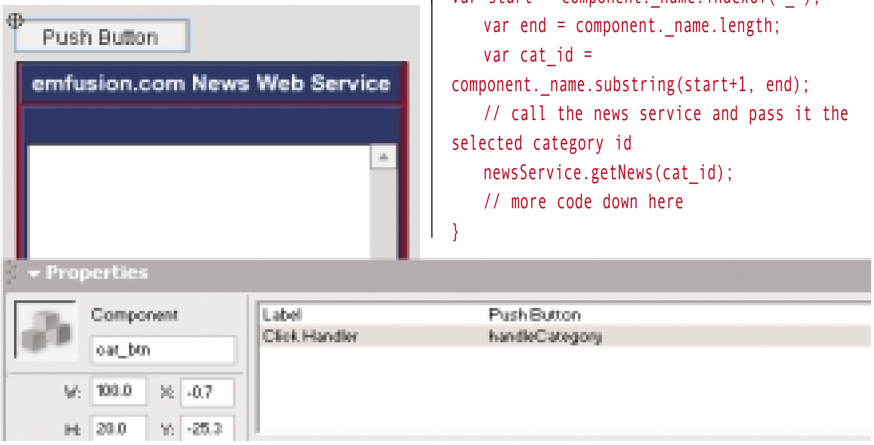
The news RecordSet contains certain fields such as title, link, and date. These fields are accessed from the RecordSet and then used to duplicate a movie clip for each record. The movie clips are then tiled vertically within the ScrollPane component. The actual duplication, text setting, and placement of clips is done within the setTitleText method which is called in each iteration of the RecordSet loop. Once the loop has completed, the refreshPane method of the ScrollPane is called to update the component.

Now we have an almost completely functioning application. The category buttons are displaying as well as the initial news titles. You can now click on a news title and you'll be directed to the corresponding article. The last thing that needs to be implemented is the user's interaction with the category buttons. By default, the first category button is disabled. We don't want the user to select the same category that he or she is already viewing. But if a user clicks another button, for example "Religion", then we should tell Flash to call the getNews method and send the corresponding cat_id to the WS.

Every PushButton component has a special Click Handler parameter that tells Flash what to do when the button is clicked. In our application we set the Click Handler during author time even though it can be specified at run time (via ActionScript code). Since we'll be using the same Click Handler for all category buttons, it makes sense to specify this within the authoring environment. The Click Handler we'll be accessing is called handleCategory. If you're viewing the emfusion_ws.fla Flash file you'll see a PushButton component sitting directly above the stage. The properties panel will allow a Click Handler to be specified for this component (see Figure 3).

The handleCategory code is:

```
function handleCategory(component) {
var start = component._name.indexOf("_");
    var end = component._name.length;
    var cat_id =
component._name.substring(start+1, end);
    // call the news service and pass it the
selected category id
    newsService.getNews(cat_id);
    // more code down here
}
```



cat_btn

W: 100.0 X: -0.7

H: 20.0 Y: -25.3

Label PushButton

Click Handler handleCategory

Figure 3: The PushButton component with Click Handler handleCategory

A nifty parameter that can be passed into the Click Handler method is a reference to the movie clip object that calls the method. This makes it simple to handle the actions of many buttons from a single method. We grab the cat_id from the button's name and then pass it to the getNews method. This allows us to grab the news for the selected category. Once again the getNews_Result responder will handle the news RecordSet and display the necessary news titles. That's it! You now have a fully functioning Flash UI that interfaces with the emfusion WS!

Conclusion

All in all, creating Flash applications that access Web services is very straightforward. Flash Remoting and the Flash components (PushButton, ScrollPane) have simplified a lot of the tasks mentioned in this article. To develop these components alone would take much longer than creating this entire application 10 times over! You're probably thinking that's an exaggeration, but I can't tell you how thorough the components are and how much time they saved me in the past.

We concentrated on two main methods of the emfusion WS in this article. Consider exploring more of the functionality built into the WS. For example, it might be a good idea to let the user pass a sort field to the WS and have the results sorted in alphabetical order or by date. You could even access the searchArticles method of the WS and let the user search for specific content, all from within the Flash UI. As you can see, there are vast amounts of possibilities and using Flash Remoting with Web services is a great way to access distributed data. Until next time, enjoy!

About the Author

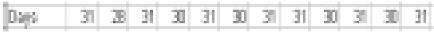
Dennis Baldwin is the lead developer for Eternal Media, a company that provides information and technology solutions for nonprofit organizations and ministries. He's also the coauthor of Reality ColdFusion: Flash MX Integration, *a book that focuses on applying Flash Remoting to real-world applications. Dennis also helps maintain a couple of online resources for Flash and ColdFusion developers at www.devmx.com and www.flashcfm.com.*

dennis@devmx.com

All About Arrays: Part 2

Helpful constructs that simplify code

In the last issue of ***CFDJ*** (December 2002), we looked at arrays, those enormously useful creatures that are underused by some developers. We confined ourselves to one-dimensional arrays – those arrays that look like a single row from a spreadsheet.

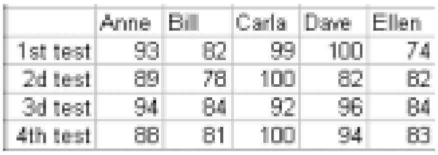


Days	31	28	31	30	31	30	31	31	30	31	30	31

Figure 1: A one-dimensional array

When Data Won't Fit into a One-Dimensional Array

But, of course, all data doesn't fit neatly into a single row. Consider, for instance, the case in which we want to keep track of five students who have each taken four tests. We might create a spreadsheet like this to keep track of their test scores:



	Anne	Bill	Carla	Dave	Ellen
1st test	93	82	99	100	74
2d test	89	78	100	82	82
3d test	94	84	92	96	84
4th test	88	81	100	94	83

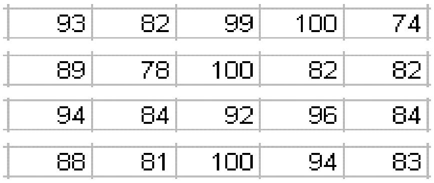
Figure 2: Data that doesn't fit into a one-dimensional array

How, though, do we get that into an array? We can fit those scores into an array only if we can expand our definition of array to mean something more than a single row of values. Luckily we can. We have just such a definition when we explore multidimensional arrays in ColdFusion.

By Hal Helms

Arrays of Arrays

You can best think of a multidimensional array as an "array of arrays." In the test scores example above, changing the image a bit can be helpful in understanding this concept. For the sake of clarity, I omit the descriptive information that isn't part of the array itself:



93	82	99	100	74
89	78	100	82	82
94	84	92	96	84
88	81	100	94	83

Figure 3: Representing a multidimensional array as a series of stacked one-dimensional arrays

Now we have a series of one-dimensional arrays stacked on top of one another. That idea, as we'll see, is key to understanding how to use ColdFusion array functions with multidimensional arrays. But first, let's create this array.

Creating a Two-Dimensional Array

If we only had the first "row" to deal with (the test scores for the first test), creating the array would be easy:

```
<cfscript>
   onedee = ArrayNew(1);
   onedee[1] = 93;
   onedee[2] = 82;
   onedee[3] = 99;
   onedee[4] = 100;
   onedee[5] = 74;
</cfscript>
```

Each number in square brackets (the *index*) is the number of an associated column. To work with arrays in two dimensions (width and height), we need to use two indices to point at a specific array element. The first index will indicate the row number and the second will indicate its column number. The code for writing the "array of arrays" to keep track of all test scores is just a matter of keeping the row and column numbers straight.

```
<cfscript>
   twodee = ArrayNew(2);

   twodee[1][1] = 93;
   twodee[1][2] = 87;
   twodee[1][3] = 99;
   twodee[1][4] = 100;
   twodee[1][5] = 74;

   twodee[2][1] = 89;
   twodee[2][2] = 78;
   twodee[2][3] = 100;
   twodee[2][4] = 82;
   twodee[2][5] = 82;

   twodee[3][1] = 94;
   twodee[3][2] = 84;
   twodee[3][3] = 92;
   twodee[3][4] = 96;
   twodee[3][5] = 84;

   twodee[4][1] = 88;
   twodee[4][2] = 81;
   twodee[4][3] = 100;
   twodee[4][4] = 94;
   twodee[4][5] = 83;
</cfscript>
```

Accessing Two-Dimensional Array Elements

How did Carla (who's represented by the third column) do on her second test (represented by the second row)? Let's find out:

```
<cfoutput>
   Carla got a #twodee[2][3]# on her second
test.
</cfoutput>
```

...and that snippet produces this:
Carla got a 100 on her second test.
How did Ellen do on her first test?

```
<cfoutput>
   Ellen got a #twodee[1][5]# on her first
test.
</cfoutput>
```

...which produces: Ellen got a 74 on her first test. Whoops, I guess we shouldn't have asked.

Looping over Two-Dimensional Arrays

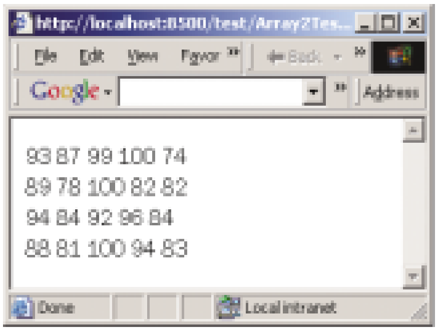
Looping over one of these "stacked-up" arrays requires us to use two loops, one nested within the other. Start by looping over the rows and then within this loop, loop over the columns for each row. Here's the code:

```
<cfoutput>
   <cfloop
      from="1"
      to="#ArrayLen(twodee)#"
      index="rowNumber">
      <cfloop
         from="1"

to="#ArrayLen(twodee[rowNumber])#"
                    index="columnNumber">

#twodee[rowNumber][columnNumber]#
      </cfloop><br>
   </cfloop>
</cfoutput>
```

That code will produce this screen (see Figure 4):

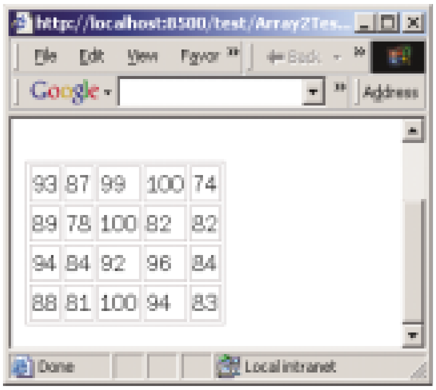


```
93 87 99 100 74
89 78 100 82 82
94 84 92 96 84
88 81 100 94 83
```

If you'd like it to look a bit more like a spreadsheet, you can adapt the code so that it places the information in a table:

```
<table border="thin" bordercolor="silver">
<cfoutput>
   <cfloop
      from="1"
      to="#ArrayLen(twodee)#"
      index="rowNumber">
      <tr>
      <cfloop
         from="1"
         to="#ArrayLen(twodee[rowNumber])#"
         index="columnNumber">
         <td>
            #twodee[rowNumber][columnNumber]#
         </td>
      </cfloop><br>
   </tr></cfloop>
</cfoutput>
</table>
```

That code produces this screen (see Figure 5):



93	87	99	100	74
89	78	100	82	82
94	84	92	96	84
88	81	100	94	83

Working with ColdFusion Array Functions

How do ColdFusion's built-in array functions work on multidimensional arrays? Unfortunately, the ColdFusion documentation does not offer us much guidance on this, so we'll have to put on our lab coats and become scientists. Our first observation should be of the code above that loops over the twodee array. There, we make use of the ArrayLen() function.

The first time we use ArrayLen() is in the outer loop:

```
<cfloop from="1" to="#ArrayLen(twodee)#"
index="rowNumber">
```

When we use the formulation ArrayLen(*array_name*), where *array_name* is a *multidimensional* array, ColdFusion returns the number of *rows* in the array, so ArrayLen(twodee) returns "4". But if *array_name* represents a *one-dimensional* array, ColdFusion returns the number of *columns*.

We need to get both rows and columns. Rows, we see, is quite simple – just provide a two-dimensional array to the ArrayLen() function. But to get columns, we must pass ArrayLen() a one-dimensional array – something we don't have. Or do we? Look again at Figure 3. I said that a multidimensional array can be thought of as a series of one-dimensional arrays and one-dimensional arrays are just what we need.

Looked at this way, twodee is a set of four stacked, one-dimensional arrays. We tell ColdFusion that we want to access one of these by providing the array name and an index representing the row number, like this: twodee[1]. That particular formulation represents the first row in our multidimensional array. What number will ArrayLen(twodee[1]) display?

If you said, "five", give yourself full marks. If you said, "four", you might not have seen that, by providing the index along with the array name, we are pointing at the first row, which is a one-dimensional array.

Most of the ColdFusion array functions are designed for one-dimensional arrays and can therefore be used by pointing to a one-dimensional array within a multidimensional array. I've taken the following material from the "ColdFusion Foundations" class I teach, as it illustrates the differences in operation of an array function depending on the type of array. Each of the illustrations assumes the existence of A, a one-dimensional array, and B, a two-dimensional array.

```
A: 1-D array [ 1 2 3 4  ]
             [ 2 4 6 8  ] B: 2-D array
             [ 3 6 9 12 ]

ArrayAppend(A, 'x')  →  1 2 3 4 X
ArrayAppend(B, 'x')  →  Error

                           1 2 3 4
                           2 4 6 8 X
ArrayAppend(B[2], 'x')  →  3 6 9 12

ArrayAvg(A)  →  2.5
ArrayAvg(B)  →  Error
ArrayAvg(B[2])  →  5

ArrayClear(A)  →  Yes
ArrayClear(B)  →  Yes
ArrayClear(B[2])  →  Yes

ArrayDeleteAt(A, 2)  →  1 3 4

                        1 2 3 4
ArrayDeleteAt(B, 2)  →  3 6 9 12

                           1 2 3 4
                           2 6 8
ArrayDeleteAt(B[2], 2)  →  3 6 9 12

ArrayInsertAt(A, 3, 'x')  →  1 2 X 3 4
ArrayInsertAt(B, 2, 'x')  →  Error

                                1 2 3 4
                                2 X 4 6 8
ArrayInsertAt(B[2], 2, 'x')  →  3 6 9 12

ArrayMax(A)  →  4
ArrayMax(B)  →  Error
ArrayMax(B[2])  →  8

ArrayMin(A)  →  1
ArrayMin(B)  →  Error
ArrayMin(B[2])  →  2

ArrayPrepend(A, 'x')  →  X 1 2 3 4
ArrayPrepend(B, 'x')  →  Error

                            1 2 3 4
                            X 2 4 6 8
ArrayPrepend(B[2], 'x')  →  3 6 9 12

ArraySet(A, 1, 4, 2)  →  2 2 2 2
ArraySet(B, 1, 4, 2)  →  Error

                            1 2 3 4
                            2 2 2 2
ArraySet(B[2], 1, 4, 2)  →  3 6 9 12

ArraySort(A,'numeric','desc')→  4 3 2 1
```

“Once arrays are explained, their mystery vanishes”

```
ArraySort (B, 'numeric', 'desc')  →  Error

                                   1 2 3 4
ArraySort                          8 6 4 2
 (B[2], 'numeric', 'desc')  →      3 6 9 12

ArraySum(A)  →  10
ArraySum(B)  →  Error
ArraySum(B[2])  →  20

ArraySwap(A, 3, 2)  →  1 3 2 4

                          1 2 3 4
                          3 6 9 12
ArraySwap(B, 3, 2)  →     2 4 6 8

                          1 2 3 4
                          2 4 6 8
ArraySwap(B[2], 3, 2)  →  3 6 9 12

ArrayToList(A)  →  1,2,3,4
ArrayToList(B)  →  Error
ArrayToList(B[2], '|')  →  2|4|6|8
```

Once arrays are explained, their mystery vanishes and they become just what they were invented to be – extraordinarily helpful constructs that can actually make our code simpler. For more on arrays – including arrays of more than two dimensions – and for some exercises in arrays, go to halhelms.com and test your knowledge of arrays with this month’s test/exercise.

About the Author

Hal Helms (www.halhelms.com) is a Team Macromedia member who provides both on-site and remote training in ColdFusion, Java, and Fusebox.

hal@fusebox.org

cf community

—continued from page 7

www.flashcfm.com). Devendra Shrikhande chimed in and also recommended www.echoecho.com/flash.htm.

The discussion spawned another thread on official Macromedia training with Flash. I had just returned from a trip to Macromedia’s east coast headquarters in Boston, where Steve Drucker and I spent five days sitting through and becoming certified to teach Macromedia’s new class “Developing Rich Internet Applications with Flash and ColdFusion.” This course teaches ColdFusion developers (with little to no experience with Flash) everything they need to know about Flash and ActionScript to begin building robust applications.

You do not need to be a designer (I can’t even draw stick figures), as the course focuses on the coding side of Flash, as well as key concepts in the Flash authoring environment. Of course, in addition to Flash ActionScript, the course also covers Flash Remoting and how to use Flash and ColdFusion together to create dynamic, data-driven Flash interfaces to ColdFusion business logic back ends. I brought it up on the List because it is exactly the type of training that so many ColdFusion developers are now looking for to expand their skill sets. On a personal note, it’s probably the best class produced by Macromedia training, to date.

Lisa Haas and Evik James both wrote to the List inquiring whether or not the DRIA course has something to offer already experienced Flash designers and whether or not a CF 5 developer not planning on migrating to CFMX would benefit from the class. Flash designers who know ColdFusion and are ready to learn ActionScript certainly will gain from the class, and CF 5 developers would benefit from all of the material in the class, though they wouldn’t be able to implement the remoting technology using CF 5.

Raymond Camden chimed in about the class (for the second or third time that month) to let everyone know that he has also reviewed the materials and feels that it is the best class currently offered by Macromedia, and the best resource for CF developers looking to learn how to use Flash to build front ends to their business applications. More information about the “Developing Rich Internet Applications” class can be found on the Macromedia Web site, or by visiting the DRIA synopsis page on the Fig Leaf Training site at http://training.figleaf.com/figleaf-training/Courses/DRIA.cfm.

ColdFusion MX Hardware Performance

What makes CFMX fly

Quite frequently, the question has been posed: "What kind of server should I set up to run ColdFusion MX?" Of course, any answer given by a software vendor, like Macromedia, would include a brief set of system requirements that detail what the minimum and recommended platforms are for the software.

By Jason Clark

While this serves its purpose to inform you of your platform requirements, it doesn't really provide insight into what hardware would truly run the software at its optimum. To offer a response to this elusive question, a series of tests were performed on several platforms to illustrate just what makes ColdFusion MX fly.

The CPU and How It Relates to ColdFusion

During the tests, it was obvious that MX scales well with multiple processors, clock speed, and L2 Cache. Okay – hold on; you might say, "What the heck are clock speed and L2 Cache?" Before we get into the test results, here's a brief CPU tutorial as it relates to MX.

When you request a template in ColdFusion, the CPU will perform all associated operations for that template. To accomplish this as quickly as possible, the CPU needs to keep the maximum number of instructions or amount of data relating to that operation available in its internal cache. There are generally between two and three "levels" of cache on the CPU. The idea behind splitting cache into multiple levels of a hierarchy is to continue to provide even faster memory.

The L1 Cache is relatively small and holds any data relevant to the operation. The L2 Cache holds excess data that couldn't be stored in the L1 Cache. It's relevant to the current operation, and perhaps the consecutive operation as well. However, the L2 Cache is significantly larger.

When the CPU has maximized its internal cache, and therefore must use external resources, such as RAM or memory, it slows the operation down considerably. In comparison, some of today's memory runs at 166MHz, while the CPU's internal cache usually runs at the same speed as the CPU. For example, if your CPU is a 2.0GHz processor, then the internal cache is also running at 2.0GHz – that's over 12 times faster than your memory.

This shows just how important it is that the L1/L2 Cache be as large as possible. Let's say you have a CFMX Web site consisting of one template. The first time the template is requested, it is pulled from main memory, which is slow (remember: over 12 times slower than the CPU), but then the CPU stores it into its cache. The next time the template is requested, it is served directly in the cache, and we now have a cache "hit."

Performance is improved tremendously, as the CPU doesn't have to go out to memory to perform its operations. It is for this reason that a fast cache is necessary for good performance in CFMX. But what happens when you have, say, 200 templates?

In this scenario, it is obvious that all of the templates won't fit into the small cache (usually 512KB or less for L2, and 128KB or less for L1). Thus, the CPU must go to main memory more frequently, which will reduce performance. If you can reduce the number of instances that the CPU must go to main memory in order to fetch data it needs, performance will improve dramatically – this is where a larger L2 Cache to store more useful data will help.

Now that you have an understanding of what the L1 and L2 Caches are in a CPU, we can discuss the other component with which ColdFusion MX seems to scale well – clock speed. To put it simply, clock speed is the speed in hertz at which the processor runs. I'm sure you're all aware of the different speeds of available processors, like 500MHz, 1GHz, etc. ColdFusion MX scales well with faster clocked CPUs. So if you upgrade from say, a P4 1.6GHz to a P4 2.0GHz, you'll see an increase in performance.

The Tests

Creating a Web-based test that will constantly work a CPU to the point of showing results between different clock speeds isn't as easy as you might think. This holds true because Web templates are relatively small, and it takes many clients to create a situation where the CPU is working as hard as possible without saturating the machine.

To construct the test, a load test was run on a rather small database, on a real-world application. The application used was FuseTalk Community Edition (made by FuseTalk Inc.), which is architected for ColdFusion MX. As a side note, one might assume that this application was chosen for this test because it's my employer's product. That's not entirely true. The main reason is that it's a real production CFMX application leveraging one of the major new features in CFMX – CFCs.

Have you heard enough about these lately? While they are fairly simple to write, they have drastically changed the way most ColdFusion applications are written. Since ColdFusion MX recently came into the scene, there are very few commercial applications designed and written for MX. FuseTalk Community Edition was designed and written with CFCs, and its entire backbone is based on a CFC API of sorts. Obviously, this design will be used by many people within their own applications, since it's going to provide the most scalability with ColdFusion MX. So it makes a useful test case.

The load test itself was recorded with the Microsoft Web Application Stress tool. It's a free tool that works quite well for loading up an application. You can use it to record a browser session with your application, loading any part of the application you want to load test. The major parts of the application were hit to construct this test recording, such as loading categories, threads, messages, and logging in. When recording the test patterns, the Web Stress tool was instructed not to record delay, so that we could stress the application by playing back the results as fast as possible.

The Web Stress tool was set to a stress level of 20 threads with a socket multiplier of 1. During the 30 minutes, the Web Stress tool replayed the test pattern, and simulated anywhere from 200–700 users depending on how frequently the user would refresh a page. The operating system used for the tests was Windows 2000 Server SP3. Apache 2.0.39 was used as the Web server. No additional tuning was done to the default install.

ColdFusion MX Tuning

To get the most out of the platform, you have to tune the software, of course. ColdFusion MX runs well as an out-of-the-box product, but it runs even better with a bit of tuning here and there. The adjusted settings are:

Trusted cache: On
Template cache size: 200
Limit the number of cached queries: 300

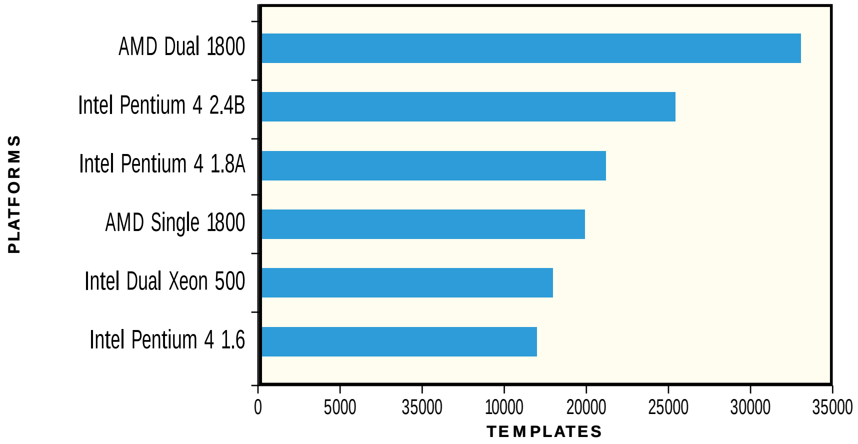
For more information on tuning the ColdFusion Administrator, check out Charlie Arehart's article "Toward Better CF Server Administration – Part 1: Performance," (***CFDJ***, Vol. 4, issue 1).

Next up is the JRE. The stock JRE in ColdFusion MX is Sun 1.3.1. The JRE was changed to BEA JRockit 7.0 for 1.3.1 J2SE. This is a free JRE, distributed by BEA (www.bea.com). The JRockit JRE is noticeably faster than the Sun JRE, by a good 20% from what I have seen so far. The JRE memory settings were set to 128MB minimum and 400MB maximum in the ColdFusion Administrator under the Java and JVM tab. The BEA JRockit JRE is free, distributed by BEA.

Processed Templates

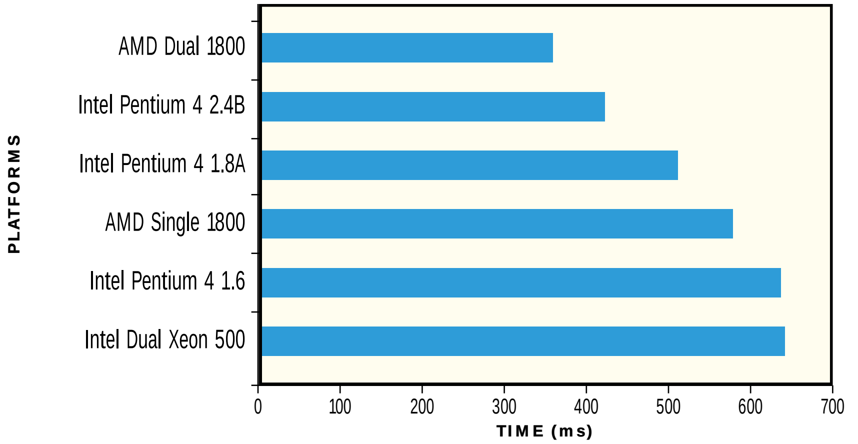
This test will indicate the number of templates each platform had processed in the 30-minute test time (see Figure 1).

The results may surprise some people, especially those still running the outdated P3 Xeon platform. While the P3 Xeon was a solid performer in its day, it's now outperformed by single CPU desktop processors. As you can see, the Dual AMD Athlon 1800 surpasses everything here, and does so with a small hit to the wallet. At approximately $700 for the CPUs, 512MB of memory, and motherboard, you really can't go wrong.



	Intel Pentium 4	Inel Dual Xeon 500	AMD Single	Intel Pentium 4	Intel Pentium 4	AMD Dual 1800
🟦 Processed Templates	16748	17639	19679	20983	25166	32832

Figure 1: Processed templates



	Inel Dual Xeon 500	Intel Pentium 4	AMD Single	Intel Pentium 4	Intel Pentium 4	AMD Dual 1800
🟦 Average Request Time	643	640	583	510	430	345

Figure 2: Average request time

What we're seeing in these results is the advantage of pure clock speed and L2 Cache scaling. The Intel Pentium 4 2.4B and 1.8A surpassed the Dual Xeon, and the Intel Pentium 4 1.6. Both the 2.4B and 1.8A have a 512K L2 Cache, giving them a decent performance increase over the P4 1.6 with it having only 256K of L2 Cache. Clock speed also plays a part; as you can see, ColdFusion MX scales well with the clock speed advantage that the 2.4B has over the 1.8A.

Another important point to note is how well ColdFusion MX scales with multiple CPUs. The Dual 1800 performed 66% better than the single 1800.

One thing to consider: you could just upgrade to a more current dual CPU platform, and run only one CPU. As your need for more power increases, just add another CPU. Besides the advantage of the faster CPUs available today, you'll benefit from the most current memory and motherboard architectures.

Average Request Time

This test outlines the average request time for the test period. All request times for each request are recorded, and then averaged at the end of the 30-minute test period. Again, the raw clock speed and Cache scaling play a significant part in the single processor differences. The Dual AMD setup shows its strength again here (see Figure 2).

Conclusions

There are a number of conclusions to draw from these tests:

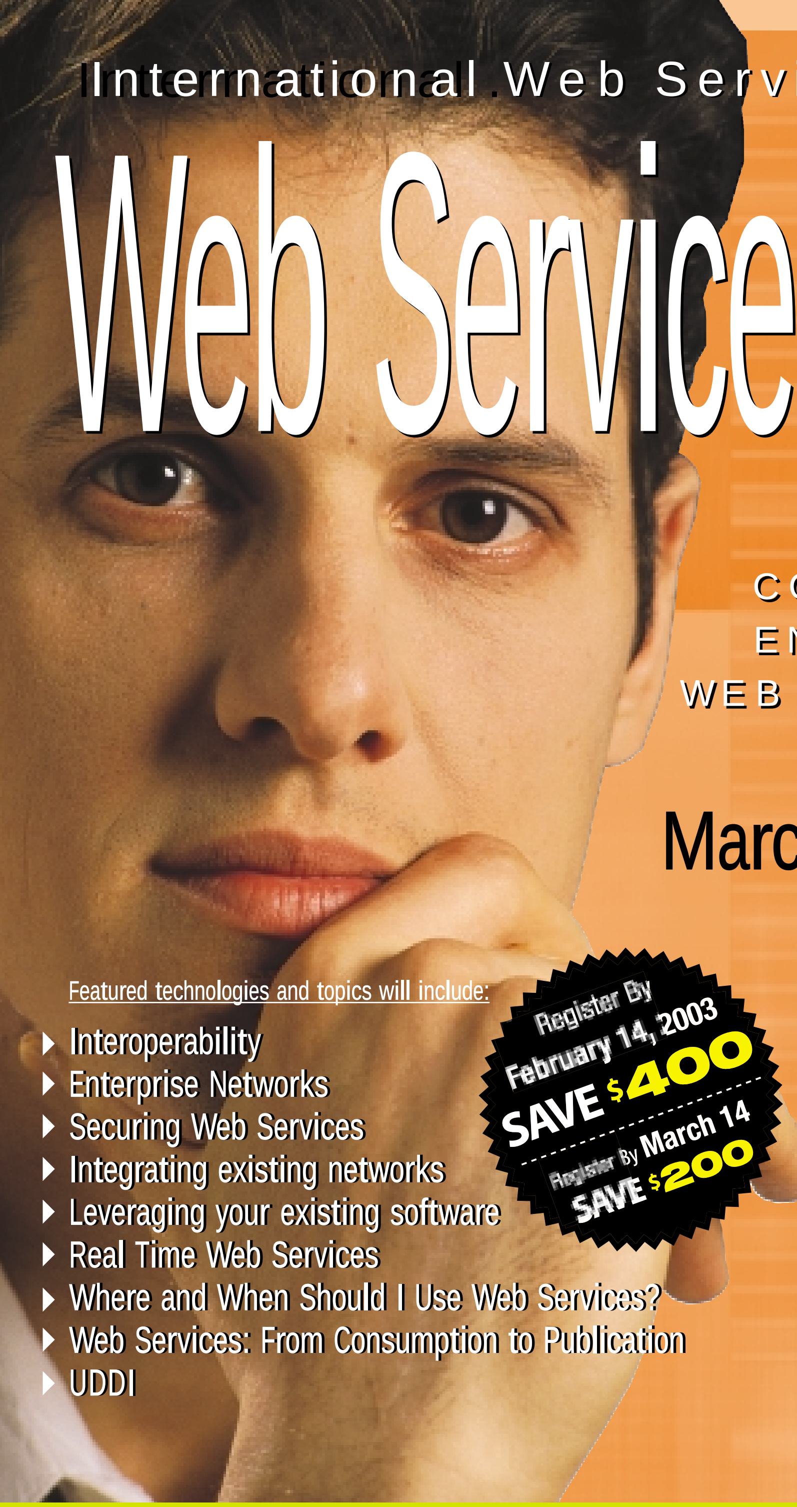
- ***L2 Cache plays a factor in the single CPU tests:*** If you noticed, the Intel Pentium 4 1.8A pulls ahead of the AMD 1800+. This is because of L2 Cache. The 1.8A has 512K, while the AMD 1800+ has 256K. As soon as the CPU has to release L2 Cache memory to fetch the data needed for the next operation, performance degrades. The Intel P4 1.8A will have to do this less often than the AMD 1800, hence the performance advantage.
- ***We have clock speed, which is shown by the differences between the Intel Pentium 4 2.4B and the 1.8A:*** The CPUs are virtually identical in architecture, so this is the advantage of higher clock speed. The Dual Xeon 500 platform is outmuscled by clock speed. It has plenty of L2 cache, but it's only running at 500MHz, which is slow by today's standards.
- ***Last, but certainly not least, is the advantage of a second CPU, illustrated by the AMD 1800+ tests:*** Adding another CPU is most certainly going to give the highest gains, especially the fact that you're getting something modern. Adding a second CPU will not only add a CPU to process another operation simultaneously, but it will add another 512K of L2 cache. If AMD isn't your game, a dual P4 Xeon will more than likely perform as well, if not better, than the Dual AMD 1800+.

For a fairly good price, you can feed ColdFusion MX what it needs – a faster CPU and a decent amount of L2 Cache. If you're looking for the highest performing hardware for your ColdFusion MX Server, it may be time to turn that old Xeon or Dual P3 into a high-performance development machine and bring those product servers up to date.

About the Author

Jason Clark is CTO of FuseTalk.

jclark@fusetalk.com



International Web Services Conference & Expo

Web Services Edge 2003

web services EDGE conference&expo

JDJ EDGE conference&expo

XML EDGE conference&expo

.NET EDGE conference&expo

CONNECTING THE ENTERPRISE WITH WEB SERVICES, JAVA, XML, AND .NET

March 18-20, 2003

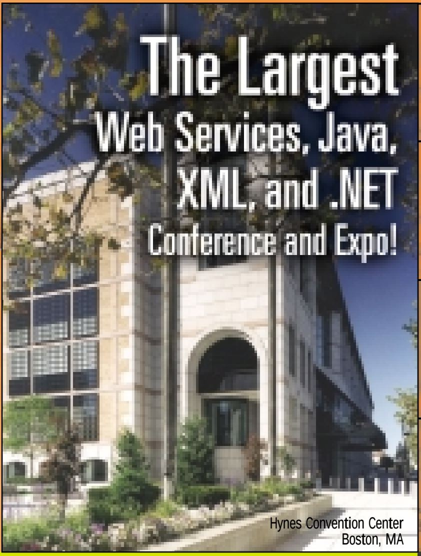
Boston, MA

Register By February 14, 2003 SAVE $400

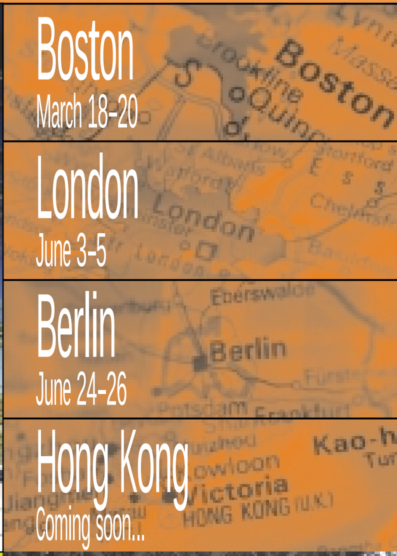
Register By March 14 SAVE $200

Featured technologies and topics will include:

- Interoperability
- Enterprise Networks
- Securing Web Services
- Integrating existing networks
- Leveraging your existing software
- Real Time Web Services
- Where and When Should I Use Web Services?
- Web Services: From Consumption to Publication
- UDDI



Hynes Convention Center
Boston, MA



Boston
March 18-20

London
June 3-5

Berlin
June 24-26

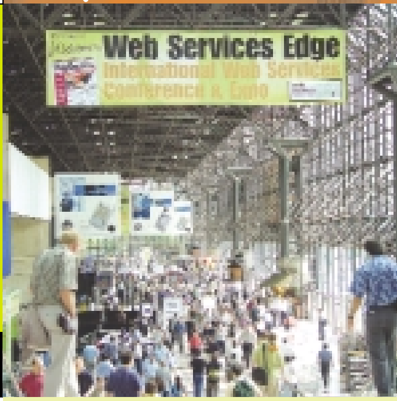
Hong Kong
Coming soon...

For more information visit
www.sys-con.com

Over 200 participating companies will display and demonstrate over 500 developer products and solutions.

Over 3,000 Systems Integrators, System Architects, Developers and Project Managers will attend the conference expo.

Over 100 of the latest sessions on training, certifications, seminars, case-studies, and panel discussions promise to deliver REAL Web Services Benefits, the industry pulse and proven strategies.



Contact information: U.S. Events: 201 802-3069 or e-mail grisha@sys-con.com • European & Asian Events: 011 44 208 232 1600

OWNED BY SYS-CON MEDIA

PRODUCED BY SYS-CON EVENTS

JAVA DEVELOPER'S JOURNAL • Java World • WebServices JOURNAL • LINUX BUSINESS WEEK • XML JOURNAL • SAMS • BASIS • SD Times • .NET DEVELOPER'S JOURNAL • asp.netPRO • WebSphere DEVELOPER'S JOURNAL • BEA WebLogic DEVELOPER'S JOURNAL • wireless BUSINESS&TECHNOLOGY • CF Advisor • COLDFUSION Developer's Journal • PerfectXML • PowerBuilder DEVELOPER Journal

Conference program available online!
www.sys-con.com/webservices2003east

Understanding Anonymous and Windows Authentication, and Applying It to Fusebox

Thinking outside the box

This article applies to developers and site administrators working in an environment using Macromedia ColdFusion running on a Windows server using Internet Information Server 4.0 or better. Those who work in an intranet environment will be especially interested. It will conclude with information on specific Fusebox applications, though non-Fusebox users may find that it's applicable to them as well.

By Alan McCollough

Anonymous Access vs Integrated Windows Authentication

With Internet Information Server 5.0, the built-in Web server provided with Windows 2000 Server, there are several user authentication schemes available. The ones we are interested in are Anonymous Access and Integrated Windows Authentication. These controls can be set at the Web site, virtual directory, and file levels, which can be useful for controlling access to any of these resources.

With Anonymous Access, all incoming requests to the Web site in question are mapped to a specific Windows user account designated for anonymous Web access. All interactions with the Web server then inherit whatever permissions are assigned to that anonymous account. Typically, this account is named "IUSR_{webserver name}," and is set with a limited set of permissions. Anonymous Access is typically used in a public Web server environment.

With Integrated Windows Authentication, when an Internet Explorer user browses a Web site that uses Integrated Windows Authentication, their current logon credentials are passed to the Web server, and all subsequent interactions with the Web server use those credentials. Thus, it is possible to make use of existing permissions within a Windows domain. Typically, Integrated Windows Authentication is used in an intranet environment, where an organization knows that all internal Web browsers will be Internet Explorer, and users log on to their Windows workstation with a specific username.

Why Have Both?

You may have a Web application in which the main portion of the site is for everybody, but certain sections require user authentication. Many applications fall into this category, where anybody can read, but only registered users can post messages, update records, etc. As we'll see later, you can have parts of your site set for anonymous authentication and others secured.

ColdFusion and User Authentication

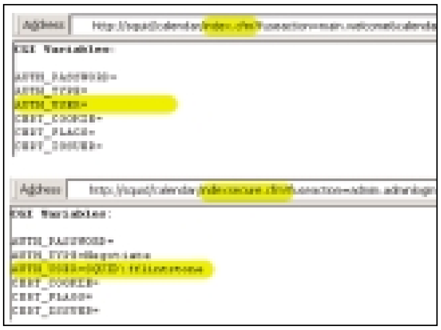
You probably know that ColdFusion has had the capability to do user authentication since version 4.0 or so, using the <CFAU THENTICATE> tag. With the release of CFMX, <CFAUTHENTICATE> has been eliminated in favor of a new family of tags that provide somewhat similar functionality.

For whatever reason, many developers find that the built-in user authentication functions in ColdFusion do not meet their needs. The method described in this article does not make use of any of the built-in user authentication tags in ColdFusion.

ColdFusion, IIS and User Authentication

Many corporate intranets make use of the Microsoft family of products, and have an existing domain user authentication model in place. Typically, in the Microsoft-based intranet environment, user authentication is handled by setting IIS to use Integrated Windows Authentication instead of Allow Anonymous Access. All Web page requests by the current user then use that user's credentials.

When a Web browser requests a ColdFusion template, and that template (or directory or entire site) is marked as Allow Anonymous Access in IIS, the value of CGI.AUTH_USER is null. When that same CF template is called with Integrated Windows Authentication active, the value of CGI.AUTH_USER will be set to the DOMAIN\username of the current user (see Figure 1).



```
http://squid/calendar/index.cfm?fuseaction=main.welcome&calendar

CGI Variables:

AUTH_PASSWORD=
AUTH_TYPE=
AUTH_USER=
CERT_COOKIE=
CERT_FLAGS=
CERT_ISSUER=
```

```
http://squid/calendar/indexsecure.cfm?fuseaction=admin.adminlogin

CGI Variables:

AUTH_PASSWORD=
AUTH_TYPE=Negotiate
AUTH_USER=[illegible]
CERT_COOKIE=
CERT_FLAGS=
CERT_ISSUER=
```

Figure 1: CGI.AUTH_USER is null until "Allow Anonymous" is turned off in IIS

Fusebox – A Framework, Not a House

Newcomers and veterans to Fusebox find that much online discussion occurs over just what Fusebox is, or what is expected of it. As the official Fusebox Web site (www.fusebox.org) states, "Fusebox is a standard framework for building Web-based applications."

Because it's a framework, Fusebox provides the developer with a fantastic way of organizing a Web application, and that's it. Folks often get disappointed that Fusebox does not handle forms validation, wash the dishes, or milk the cow, but remember, Fusebox is a framework, nothing more. Many developers out there have come up with a cornucopia of components that do expand on the Fusebox framework, including components for handling user authentication. The technique described here doesn't require any specific components, just a rearranging of what already exists.

Fusebox Makes It Easy

One of the "rules" of Fusebox is that all browser requests point to index.cfm. You could have a thousand templates in a Fusebox application, with dozens of nested subdirectories, but all the end user will ever see is a URL pointing to "index.cfm" at the root as the target template. Many Fusebox developers wisely prevent users from accessing any other template in the application by adding code to Application.cfm to test for any .cfm template call other than index.cfm (see Listing 1).

In order to get what we want, an application that honors both Anonymous Access and Authenticated Access, we need to think outside the box – the Fusebox, that is. Instead of having all requests point to a single template, index.cfm, we will create a new template, indexsecure.cfm, for all requests requiring Authenticated Access:

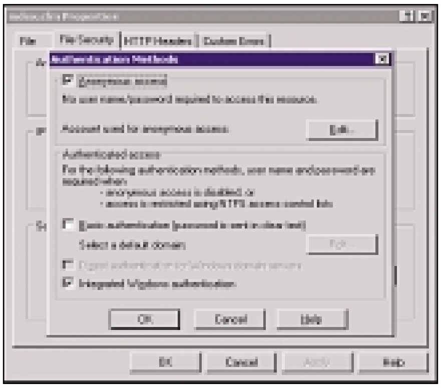
- All requests where Anonymous Access is desired will point to index.cfm
- All requests where user authentication is desired will point to indexsecure.cfm

Putting It All Together – Modifying an Existing Fusebox Application to Use Both Authentication Modes

The following work with both older Fusebox 2.x and Fusebox 3 apps.

Changes in Your Fusebox Application

- First and foremost, make a backup copy of your existing Fusebox application before proceeding!
- Make a copy of index.cfm and save it as indexsecure.cfm. There should now be index.cfm and indexsecure.cfm at the root level of your Fusebox application (see Figure 2).



☑ Anonymous access

No user name/password required to access this resource.

Account used for anonymous access: Edit...

Authenticated access

For the following authentication methods, user name and password are required when
- anonymous access is disabled, or
- access is restricted using NTFS access control lists

☐ Basic authentication (password is sent in clear text)

Select a default domain: Edit...

☐ Digest authentication for Windows domain servers

☑ Integrated Windows authentication

Figure 2: By turning both options on, IIS will use either the anonymous or authenticated user credentials

- Throughout the entire Fusebox application, change any index.cfm referrals to indexsecure.cfm where user authentication is desired.
- Leave any referrals to index.cfm alone where anonymous access is desired.
- Modify application.cfm so it will allow both index.cfm and indexsecure.cfm (see Listing 2).

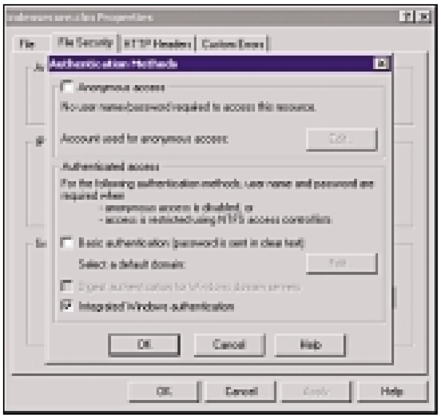
To prevent sneaky users from changing a call to indexsecure.cfm back to index.cfm in order to execute an unauthorized fuseaction, it is necessary to modify your Fusebox application so that only the appropriate fuseaction may be called. In a Fusebox application, actions – called *fuseactions* – are called by traversing a CFSWITCH which determines what action to take.

With Fusebox 2.x applications, securing which fuseactions get called is easy. Simply delete the fuseactions requiring user authentication from index.cfm and add them to indexsecure.cfm.

With Fusebox 3.x applications, this is a bit trickier, since index.cfm no longer houses the big CFSWITCH, which resides in FBX_Switch.cfm.We instead modify that file to control which fuseactions are secured and which are not (see Listing 3).

Changes in File Level Permissions

You may need to make some changes to permissions at the file level. In order to allow both the anonymous account and authenticated domain users access to your application, it is necessary to ensure that both the anonymous account and authenticated users have Read access. The images here show a "Before" and "After" of file-level permissions settings on a folder. This example works if your Anonymous Access account is a member of the "Domain Users" group. Consult with your domain security specialist before making changes to file-level permissions (see Figures 3 and 4).



☐ Anonymous access

No user name/password required to access this resource.

Account used for anonymous access: Edit...

Authenticated access

For the following authentication methods, user name and password are required when
- anonymous access is disabled, or
- access is restricted using NTFS access control lists

☐ Basic authentication (password is sent in clear text)

Select a default domain: Edit...

☐ Digest authentication for Windows domain servers

☑ Integrated Windows authentication

Figure 3: In IIS, highlight the indexsecure.cfm file, turn Allow Anonymous Access OFF, and turn Integrated Windows Authentication ON

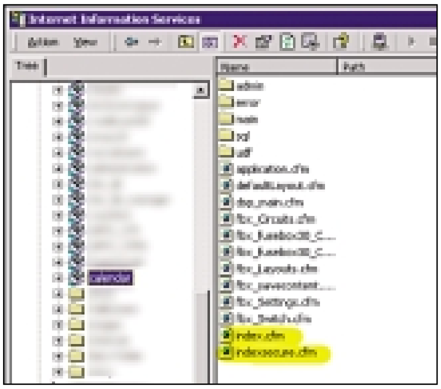
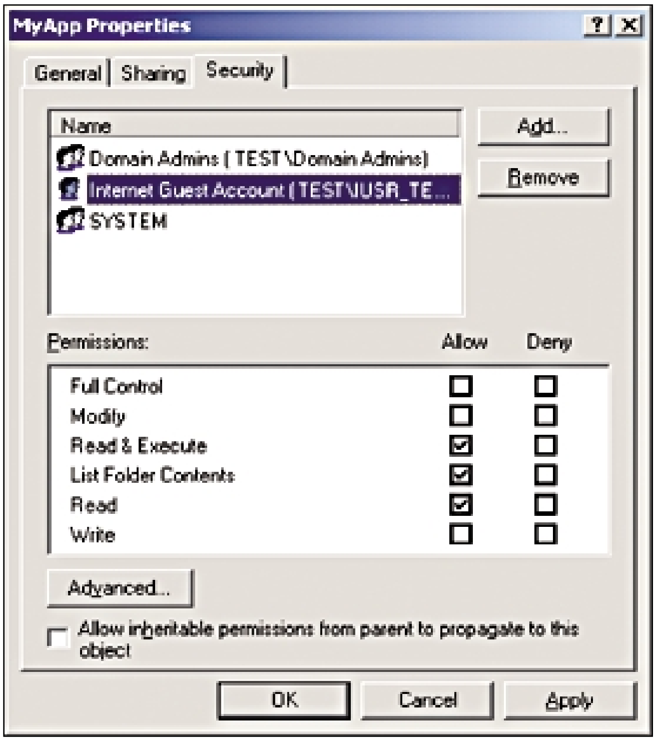


Figure 4: Index.cfm and indexsecure.cfm at the root level of your Fusebox application

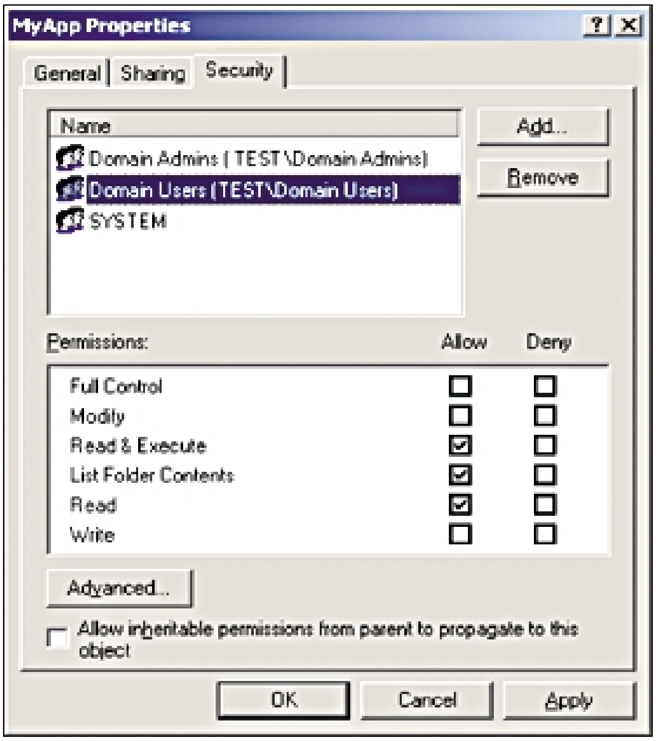


Domain Admins (TEST \Domain Admins)
Internet Guest Account (TEST\IUSR_TE...
SYSTEM

Permissions:	Allow	Deny
Full Control	□	□
Modify	□	□
Read & Execute	☑	□
List Folder Contents	☑	□
Read	☑	□
Write	□	□

□ Allow inheritable permissions from parent to propagate to this object

Figure 5: Remove the Anonymous Internet account...



Domain Admins (TEST \Domain Admins)
Domain Users (TEST\Domain Users)
SYSTEM

Permissions:	Allow	Deny
Full Control	□	□
Modify	□	□
Read & Execute	☑	□
List Folder Contents	☑	□
Read	☑	□
Write	□	□

□ Allow inheritable permissions from parent to propagate to this object

Figure 6: ...and add the Domain Users group instead

Changes in IIS

For the directory holding the Fusebox application, select both Allow Anonymous Access *and* Integrated Windows Authentication. Why turn them both on? When a user passes from a user-authenticated template back to an Anonymous-Access template, they will be denied, since the browser session is now mapped to the current user's credentials instead of the anonymous account. By turning both options on, IIS will use either the anonymous user credentials, or authenticated user credentials (see Figures 5 and 6).

Those familiar with Fusebox will see that this technique can be modified to work in a variety of ways, but the basic idea remains:

- All requests where Anonymous Access is desired will point to index.cfm
- All requests where user authentication is desired will point to indexsecure.cfm

Note: The technique outlined here represents a deviation from accepted standard Fusebox technique. For more details on Fusebox, please visit www.fusebox.org.

About the Author

Alan McCollough is a certified Macromedia ColdFusion Developer who works at Alaska Native Medical Center in Anchorage, Alaska.

amccollough@anmc.org

Listing 1:

```
<!--- application.cfm --->
<!--- Allow only index.cfm to be executed --->
<cfif ListLast(GetTemplatePath(),'\') neq "index.cfm">
   <!--- The user is calling a template OTHER than index.cfm -
  You should redirect the user or halt further execution.-->
<cfelse>
```

Listing 2:

```
<!--- application.cfm --->
<!--- Allow only an authorized list of templates to be executed --->
<cfset variables.safetemplatelist = "index.cfm,indexsecure.cfm ">
<cfset variables.templatename = ListLast(GetTemplatePath(),'\')>
<cfif ListFind(variables.safetemplatelist, variables.templatename,",") eq
0>
  <!--- call is being made to an unauthorized template.
  You should redirect the user or halt further execution.-->
</cfif>
```

Listing 3:

```
Within FBX_Switch.cfm, split the one CFSWITCH in two.

<cfif ListLast(GetTemplatePath(),'\') eq "indexsecure.cfm">
  <CFSWITCH expression="#fusebox.fuseaction#">
    …. Put user authenticated fuseactions here
  </CFSWITCH>
<cfelse>
  <CFSWITCH expression="#fusebox.fuseaction#">
    …. Put anonymous access fuseactions here
  </CFSWITCH>
</cfif>
```

2 *Ultimate* COLDFUSION CDs

ONE LOW PRICE!

HURRY! LIMITED TIME OFFER WHILE SUPPLIES LAST!

CF Advisor
THE COMPLETE WORKS
CFA
EVE
ColdFusion Developer's Journal
PRESENTS
ColdFusion
Resource CD
Secrets of the ColdFusion Masters
$198 VALUE
FROM
Inside: SPECIAL
Robert Diamond
Every Issue of CFDJ Ever Published on One CD!

SOLD SEPARATELY ONLY $71.99

SOLD SEPARATELY ONLY $89.99

The most complete library of exclusive ColdFusion articles on CD!

CDs are edited by ***ColdFusion Developer's Journal*** Editor-in-Chief Robert Diamond and organized into 44 chapters containing more than 650 exclusive ColdFusion articles.

Applications
CF & Java
CF Basics
CF Server
CFDJ List
CFMX
CF Tips & Techniques
Custom Tags
Database
E-Commerce
Editorial
Enterprise CF
Error Handling
Flash
Forms
Fusebox
Hosting
Interviews
News
Object-Oriented CF
Programming Tips
Q&A
Reviews
Scalability
Security
Source Code
CFSpectra
Upgrading CF
User-Defined Functions
Verity
WDDX
Web Services/XML
Wireless

2 CD OFFER

"CFA: THE COMPLETE WORKS"

PLUS

"CFDJ: RESOURCE CD"

ONLY $99.99!

Sold Exclusively through... JDJStore.com

SYS-CON MEDIA

Order Online at
www.JDJSTORE.com

OFFER SUBJECT TO CHANGE WITHOUT NOTICE

ALL BRAND AND PRODUCT NAMES USED ON THESE PAGES ARE TRADE NAMES, SERVICE MARKS, OR TRADEMARKS OF THEIR RESPECTIVE COMPANIES.

the 5th year ColdFusion Developer's Journal
1999 • 2000 • 2001 • 2002 • 2003

CFDJ: THE FIFTH YEAR

FROM COLDFUSION 4 TO CONTRIBUTE, CFDJ CONTINUES TO GUIDE THE CF COMMUNITY

Four years is a generation in Internet time; in the life of ***ColdFusion Developer's Journal*** and ColdFusion itself, it marks an epoch. As we celebrate the new year and the beginning of ***CFDJ***'s fifth, here's a look at where we've been and where we may be going.

By Charles Arehart

We'll focus on some of the CF milestones in that time, and recall some instances perhaps best forgotten. We'll also highlight key resources that have matured. Overall, we hope this retrospective can serve as a checkpoint to put the past in perspective and bring the present into focus.

This marks my 33rd article in *CFDJ*; looking back through the previous 32 provides a ColdFusion timeline of sorts. My first article appeared in the second issue, and I've been a "regular" ever since. *CFDJ* has attracted many fine contributors over the years, among them, Ben Forta and Hal Helms, who have written even more articles than I have.

Four Years Ago This Month

What were you doing four years ago? Some of us had been using ColdFusion for a couple of years or more, while others were just getting started with HTML. Some hadn't even used the Web yet, while others were still in school. Perhaps some were programming in other languages like Visual Basic, PowerBuilder, Java, or even on the mainframe.

ColdFusion was already going strong by January 1999. Indeed, we were celebrating the recent release of ColdFusion 4 in the first issue, with Richard Schulze's "What's Different About ColdFusion's Version 4.0" (www.sys-con.com/coldfusion/article.cfm?id=4).

Perhaps the most significant aspect of version 4 was its focus on enterprise performance and scalability, with the introduction of native database drivers, load balancing and failover, server sandbox security, CORBA support, and more. These were all elements of the new "Enterprise Edition" of ColdFusion.

Gone was the "Workgroup Edition," though users of the "Professional Edition" would also gain scalability features (trusted cache, query caching, the ability to store client variables in data sources or cookies rather than the registry, enhanced multithreading, and more) as well as enhanced security in the form of the new Basic and Advanced Security models.

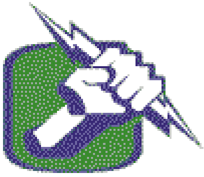
At the time, Jeremy Allaire, in an interview with SYS-CON Radio, said, "With this offering we were able to support dozens of large dot-com customers handling millions or tens of millions per day, and had a couple of the top 10 holiday e-commerce sites deployed."

I continued the discussion of the new release in my follow-up to Richard's article in the second issue, "ColdFusion 4.0: More on Developer Enhancements" (www.sys-con.com/coldfusion/article.cfm?id=22). Other new features that debuted were structures, exception handling tags, WDDX, switch/case processing, CFSCRIPTing, a syntax checker, and lots more.

ColdFusion Studio also received a major upgrade in its version 4 released about the same time. Some of us lauded the new interactive debugger and benefited from it for years, while others never seemed able to get it to work for them. It also introduced one-step deployment, the codesweeper tool, style editor, and more.

Before you conclude that Versions 4.5 and 5 were the only changes of import until MX, note that 4.01 of CF and Studio soon followed and introduced additional features that many may have missed. See my "Hidden Gems in 4.0.1 – What You Might Have Missed," at www.sys-con.com/coldfusion/article.cfm?id=81.

Version 4 marked another turning point: it was no longer to be referred to as "Cold Fusion Application Server," but simply ColdFusion. No space, no "application server." Many today still seem not to have gotten the memo (there really was a document explaining this) and continue to slip into calling it CFAS. Sadly, it wouldn't be long before we'd also say goodbye to the classic ColdFusion "fist" logo that many of us loved so much.



The classic CF "fist" logo

All this was part of another momentous occasion that took place in January 1999: Allaire completed its initial public offering. Of course, Macromedia would acquire the company exactly two years later. If you're curious about what sort of things happened in Allaire's history leading up to that point, you can see a timeline of Allaire milestones at www.macromedia.com/v1/company/milestones.cfm.

Reaction to the merger has been mixed and varied in the two years since. Like so many things, some see it as a win while others decry it as having compromised some essence of their memory of "the Allaire days." Frankly, I prefer to just move forward and appreciate the many good things that have come about since the merger.

Speaking of mergers and acquisitions, another one took place that year that may have barely registered a blip on the radar screens of most CF developers. It was in June of 1999 that Allaire acquired Live Software and "some product called JRun" that could "run JSPs and servlets." Little did we realize that this acquisition would mark the beginning of the transition of ColdFusion to a Java platform.

With ColdFusion releases 4.5 and 5 in the meantime, we benefited from continued refinement and extension to the platform, tag set, and functionality. But clearly no release was as momentous as CFMX.

The Techniques, the Articles

CFDJ has been there through it all, with articles explaining seemingly every facet and technique of ColdFusion development. Some of the many frequently discussed topics have included (see Table1):

THE FUSEBOX DEVELOPMENT APPROACH	LOAD SESSION BALANCING	SESSION MANAGEMENT
CHOOSING AND WORKING WITH DATABASES	USER-DEFINED FUNCTIONS	CUSTOM TAGS
HOSTING	WDDX	ARCHITECTURE
USABILITY	VERITY INDEXING	PERFORMANCE TUNING
ERROR HANDLING	CFSCRIPTING	STUDIO TOOLS AND TIPS
APPLICATION TESTING	SERVER ADMINISTRATION	REGULAR EXPRESSIONS

And much more. Of course, all the current hot topics in CFMX have been covered this year, including:

- Web services
- Components
- Flash integration
- XML processing
- Java integration

Interestingly, those last three topics (and some others) that are now tightly integrated in CFMX have been available for some time, and there are articles going back to Volume 1.

This is a good point for us to recall some of the highlights of the four years of articles. Of course, no such retrospective could do justice to all the fine contributors and great topics that have been covered.

We can't start with anyone other than Ben Forta, Macromedia's senior evangelist, who's written more articles in the magazine than anyone else (46), sometimes more than one per issue. Indeed, he's missed only a single issue, in April of this year, which was probably at the height of publication of his three *Reality ColdFusion* books, or the update to his CF Web App Construction Kit, or the Advanced CFMX App Dev book. Clearly, Ben's a busy guy, and we're glad to have him on the team.

Hal Helms comes in a very close second, with 40 articles. As a leader in the Fusebox community, most of his articles have covered facets of that, but many readers will also recall his adroit use of classic quotes and erudite references to Mark Twain, Aristotle, Zen Buddhism, the *Screwtape Letters*, his father's ministerial influence, his past in carpentry, *Tipping Points*, and so much more. Whether you care to learn about Fusebox, you'll almost always learn something about life from Hal.

I'd like to single out a couple of other writers who've put together some multipart series that delved into valuable details for readers who stuck it out over the months! Guy Rish's "Cold Cup O' Joe," an eight-part series on CF/Java integration served as a great foundation, especially coming as it did before MX and showing the possibilities then that are as useful now. David Gassner's recent three-part series on XML integration in CFMX was a real tour de force in helping developers get started and go the extra mile at the same time.

Similarly, Simon Horwith has now written nearly a dozen **Tales from the List** columns, highlighting the best of the ***CFDJ*** mailing list (more on that later), and Bruce Van Horn has logged more than two dozen **Ask the Training Staff** columns, where he provides answers to common questions asked of CF instructors.

This seems like a good time to point out that if you would like to explore all these past articles, there are a few ways to do so. First, you can view the titles of all past articles at the archives (www.sys-con.com/coldfusion/archives.cfm), and can even read many of the feature articles for free (www.sys-con.com/coldfu sion/ features.cfm). If you want to search for some text or a name, see the search form on the left nav bar of each page on the *CFDJ* site. At the bottom of the search result page from that, you can also perform a search by author.

Finally, and perhaps best of all, you can now get all four years on a single CD, 450 articles organized into 23 "chapters," searchable as well. You can order the CD online at www.jdjstore.com/col rescd.html. Makes a great gift!

Gone But Not Forgotten?

Not every topic we've covered (or that CF developers have explored) has led a long life. A trip down memory lane would be remiss if it didn't present a few topics that have come, and mostly gone, in the life of ColdFusion:

- Allaire Alive
- ColdFusion Express
- Java graphlets
- Kawa
- Netegrity SiteMinder
- SMIL tag pack
- Spectra
- Starbase Versions integration
- VRML

Of course, back in 1999 we were fretting over something that really turned out to be a nonissue for most, it seemed. Remember Y2K?

One topic that some may have dismissed but has since made a bit of a revival is Crystal Reports integration. The improvement in this case stems as much from improvements in the underlying Crystal tool itself.

Finally, another topic that may have seemed to become a moot point soon after 1999, but has also experienced a revival, is the Netscape versus IE debate. With the recent improvements implemented by the Mozilla organization, some find that IE is being tested, if not bested.

What's in a Code Name?

This time of reflection also seems an apt opportunity for a little diversion, to consider some of the code names that have been used for various products and betas in ColdFusion and related product history. I don't claim this to be a complete (or completely accurate) list, but it's fun to reminisce:

- Dharma (was a planned Spectra 2)
- Harpoon (dynamically generated Flash from CF)
- Harvest (server monitoring features shipped with CF 5)
- Kojak (Dreamweaver MX's beta code name)
- Neo (CFMX's beta code name)
- Nirvana (was a planned Spectra 3?)
- Nozome (JRun 4's beta code name)
- Pharaoh (overall initiative including Neo, Harvest, Tron)
- Pharaoh studio (planned IDE that would combine Studio/HS/Kawa)
- Tempest (Spectra 1's beta code name)
- Tardis (Spectra 1.5's beta code name)
- Tincan (Flash Comm. Server's beta code name)
- Trinity (release that was said to eventually follow JRun 4)
- Tron (XML/Web services/B2B features that made it into CFMX)
- Velcro/DuctTape (Dreamweaver UltraDev's beta code name)

If you can think of others, feel free to share them with me, or enter them in the comment area on the online version of this article at www.sys-con.com.

Community and Resources

I think if you ask most CF developers what the key is to their personal development, success, and ability to solve prob-

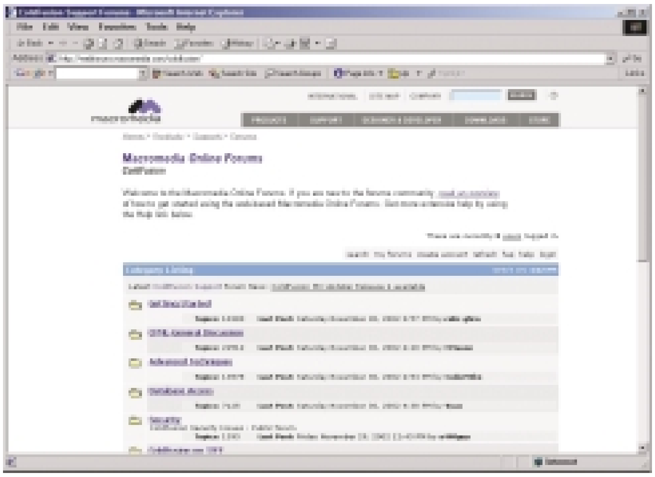


Figure 1: Macromedia CF Forums

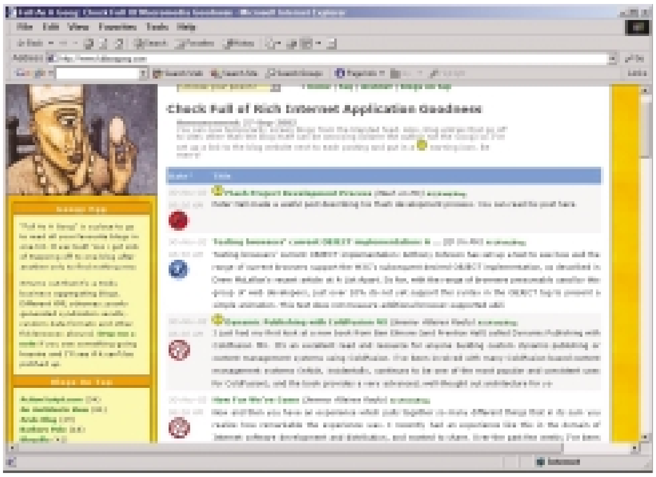

File Edit View Favorites Tools Help
Chock Full of Rich Internet Application Goodness

Figure 2: MX Blog Aggregator

lems when working with ColdFusion, they'll point to the community. It seems that from the beginning, CFers have been able to rely on each other and benefit from the tremendous community of developers (including *CFDJ* authors) and the resources that were enabled by Allaire and Macromedia.

Speaking of community support, 1999 was the year of the first DevCon, or Allaire Developer Conference, in Boston. It was the first of a long line of great conferences, as a great coming together of the CF community and a celebration of everything ColdFusion. But it wasn't actually the first ColdFusion conference. Several of us had gathered in Fort Collins, Colorado, in July, 1998 for the first "national CF conference," where speakers included Allaire founder Jeremy Allaire, chief architect Sim Simeonov, yours truly, and several others. Clearly, the ColdFusion community was taking shape on a larger scale by 1999.

In actuality, the community was already well developed in many ways, both online and in the form of local user groups, the oldest of which was the DC ColdFusion User Group run by Fig Leaf Software. I was working there at the time, and with folks like Steve Drucker, Dave Watts, and others sharing knowledge each month, there were often a hundred attendees at the meetings.

User groups have continued to thrive and you can find a local one (for ColdFusion or for other Macromedia products) at www.macromedia.com/v1/usergroups. There's also been a growth in regional events, both organized by Macromedia and run by developers. These include CF Europe, MXDU, DevCon Japan, CF North, Denver TechCon, MXDC, CFUN, CF Underground, and many more, including some attempted and still planned cruises. You can find most of these events listed at Michael Smith's "CF Conference Central" at www.cfconf.com.

Of course, not everyone lives in a city with such an active user group or event. Fortunately, there have always been plenty of online resources to help developers support themselves. In fact, I reviewed them in my March 2000 (Vol. 2, issue 3) article, "Helping Yourself – Resources for Learning and Getting Questions Answered," at www.sys-con.com/coldfusion/arti cle.cfm?id=91. Most of the resources I described then are as valuable today.

Perhaps the greatest self-help and community resource is the Macromedia ColdFusion Forums, at http://webforums.macromedia.com/coldfusion. Hundreds of messages have been traded every day for several years, and there are currently more than 60,000 messages that have been posted! (See Figure 1.) The forums may be overwhelming in their depth, and clearly no one can read and respond to every message, though Team Macromedia members (like myself) try to help where we can. More than anything, it's just another great example of developers helping each other. And the search facility allows you to find the proverbial needle in a haystack when you need it.

Of course, not everyone enjoys Web-based forums, preferring mailing lists instead. The king of those is the CF-Talk, run by longtime CF maven and community leader Michael Dinowitz. With hundreds of messages traded per day, you can sign up to receive them by e-mail or simply read them online at www.houseoffusion.com/cf_lists. Indeed, there you'll find several other lists including active ones on CF-Flash, CF-Linux, CF-Jobs, SQL, JRun, and lots more. Other valuable mailing lists include our own CFDJList (www.syscon.com/coldfusion/list.cfm) as well as those of several user groups that nearly always welcome non-locals.

Still another, more recent means by which enthusiastic developers share their knowledge with others is the "blog," or Web log. These generally take the form of a diary of discoveries and observations. You'll find them offered by Macromedia employees, user group managers, product developers, and others, including my own at http://cfmxplus.blogspot.com. One of the best places to find most of them both listed and aggregated (so you can read them in one place) is Geoff Bowers' www.fullasagoog.com. Don't worry about the name (it's explained there). Just go fill yourself with CF (and other MX product) goodness (see Figure 2).

An equally valuable resource is the Macromedia Knowledge Base, www.macromedia.com/v1/support/knowledgebase/searchform.cfm, where several hundred TechNotes are offered on a huge range of topics. They go back to 1996, and it's fun to see what issues troubled folks then that may or not bother us now. The second oldest, TechNote 158 from November 1996, laments that cookies are not set on a page doing a CFLOCATION. Fortunately, it's been updated to reflect that this problem has gone away in CFMX. (We shouldn't say it was "fixed." You heard no announcement of it in the manuals or release notes, but good riddance nonetheless.)

ColdFusion Ship Dates

Have you ever wondered what the ship dates were for the various major releases of ColdFusion?

1.0: July 1995	3.0: July 1997	5.: June 2001
2.0: November 1996	4.0: November 1998	MX: May 2002

Macromedia has also invested heavily in its Designer & Developer Center, at www.macromedia.com/desdev, and it's become a valuable resource, with hundreds of articles covering CF and all other Macromedia products.

Recently, SYS-CON (publisher of *CFDJ* and other fine technical journals) has created a new developer portal, http://developer.sys-con.com. Those familiar with theserverside.com or slashdot will recognize the approach. Once you register, you can contribute comments on the posted topics or even post your own. The first posted thread on a ColdFusion topic had nearly 60 replies in the first two days. It's not limited to CF topics, and it does indeed open the discussion to a wider audience than the number who might participate in the Macromedia forums, for better or worse. But it's a community.

One final element of community, in the minds of many, comes in knowing who else leverages ColdFusion. We all know that it works well for us, but sometimes we're challenged to point to any examples of large and/or highly visible organizations that use it. Back in May 2001 (Vol. 3, issue 5) I wrote, "Who's Using ColdFusion?" (www.sys-con.com/coldfusion/article.cfm?id=271) where I identified several such organizations. Of course, it's been more than 18 months since then, but the list remains mostly accurate. Indeed, I've since learned of more such sites and keep that updated list on my site at www.systemanage.com/cff/who_uses_cf.cfm.

Technologies and the Future

We've seen how many things have changed in the past four years and how *CFDJ* has been there for you. More important is how much Allaire, and then Macromedia, has been constantly watching the technology landscape to give us the solutions we need, often before most of us knew we needed them!

Of course, they've always responded to the needs of developers and organizations relying on ColdFusion. The addition of support for Unix platforms and Apache integration are examples, as are the advancements in version 4 to make CF more scalable, performant, and secure. But they quickly moved forward from there.

In 2000, Allaire released their Technology Roadmap (still available at www.macromedia.com/v1/documentcenter/partners/allairebusinessplatform1.pdf). In it, they described their efforts that laid the groundwork for what would become ColdFusion 5 and MX, including those aforementioned "code name" projects for XML and Web services integration, Java integration, server management, and more.

Beyond that, we see that CFMX has introduced the concept of component-based development. On the surface, this may seem to some to be merely an extension of modular programming design, but those who are leveraging it to approach object-oriented development may find more reward. They'll certainly lay the groundwork for those who follow. The IT world in general is moving in the direction of component-based, as well as service-oriented, architectures, so these are more great examples of how Macromedia is positioning us to take advantage of this trend.

Macromedia's not only following the IT trends, but is also setting them. By now everyone reading this should be aware of the great power of Rich Internet Applications, as enabled by Flash, Flash Remoting, and the Flash Communications Server. For some, the Flash interface and platform may seem a daunting prospect, but as more developers become familiar with it and offer their own interpretations of how developers can get started with it, we'll see it become a more common tool in the kits of most CF developers.

Other examples of Macromedia bringing developers into new ways of working include Dreamweaver MX and the more recently released Contribute. Many CF developers scoff at these tools thinking they don't really apply to them. And they are new and may change in time to better suit CFers. Again, I think you'll see more and more people helping each other learn how the tools can apply or, in some way, take a load off CF developers.

With respect to all these "new" topics, we absolutely welcome contributions for articles by developers who feel they have something to offer. As we've seen before, the CF community learns best from each other. Be part of the solution!

Conclusion

We've touched a bit on how ColdFusion, the community, and the technologies we use have continued to change and improve in the past four years. They'll continue to change, of course. That's the only constant.

CFDJ has changed as well. You may have noticed the new look we took on with a graphics redesign in recent months. Another change is that Raymond Camden and I were recently appointed as technical editors. We aim to constantly improve the technical quality of the magazine. Here's to a great year five! Now on to writing my 34th topic...

About the Author

Charlie Arehart is co-technical editor of ColdFusion Developer's Journal. *He is a certified Macromedia trainer/developer and CTO of SysteManage. He contributes to several CF resources, is a frequent speaker at user groups throughout the country, and provides training, coaching, and consultation services. Charlie was recently named to Team Macromedia, and he hosts an informational site at www.systemanage.com and a blog at cfmxplus.blogspot.com.*

charlie@sys-con.com

cf user groups

National

Alabama
Birmingham CFUG
www.bcfug.org

Alabama
Huntsville CFUG
www.nacfug.com

Alaska
Anchorage CFUG
www.akcfug.org

Arizona
Phoenix CFUG
www.azcfug.com

Arizona
Tucson CFUG
www.tucsoncfug.org

California
Bay Area CFUG
www.bacfug.net

California, Fresno
Central California CFUG
www.cccfug.com

California
Inland Empire CFUG
www.sccfug.org

California
Los Angeles CFUG
www.sccfug.org

California
Orange County CFUG
www.sccfug.org

California
Sacramento CFUG
www.saccfug.org

California
San Diego CFUG
www.sdcfug.org

Southern California
Southern California CFUG
www.sccfug.org

Northern Colorado
Northern Colorado CFUG
www.nccfug.com

Colorado
Hamilton M.S. CFUG
http://hamilton.dpsk12.org/teachers/team-c/intro.html

Delaware, Dover
Delaware CFUG
www.decfug.org

Delaware
Delmarva CFUG
www.delmarva-cfug.org

Florida
Gainesville CFUG
http://plaza.ufl.edu/aktas

Florida
Orlando CFUG
www.cforlando.com

South Florida
South Florida CFUG
www.cfug-sfl.org

Florida
Tallahassee CFUG
www.tcfug.com

Florida
Tampa CFUG
www.tbcfug.org

Georgia
Atlanta CFUG
www.acfug.org

Georgia, Atlanta
Georgia CFUG
www.cfugorama.com

Georgia
Columbus CFUG
www.vcfug.org

Hawaii, Honolulu
Hawaii CFUG
http://cfhawaii.org

Illinois, Champaign
East Central Illinois CFUG
www.ecicfug.org

Illinois
Chicago CFUG
www.cfugorama.com

Indiana
Indianapolis CFUG
www.hoosierfusion.com

Indiana, South Bend
Northern Indiana CFUG
www.ninmug.org

Iowa
Des Moines CFUG
www.hungrycow.com/cfug

Kentucky
Louisville CFUG
www.loulexcfug.com

Louisiana
New Orleans CFUG
www.nocfug.org

Maryland
Annapolis CFUG
www.ancfug.com

Maryland
Baltimore CFUG
www.cfugorama.com

Maryland
Broadneck H. S. CFUG
www.cfug.broadneck.org

Maryland, Bethesda
Maryland CFUG
www.cfug-md.org

Maryland
California, MD CFUG
http://smdcfug.org

Massachusetts
Boston CFUG
www.cfugboston.org

Michigan, Dearborn
Mmaniacs CFUG
http://ciwebstudio.com/MMania/MMania.htm

Michigan, East-Lansing
Mid Michigan CFUG
www.coldfusion.org/pages/index.cfm

Minnesota, Minneapolis
Twin Cities CFUG
www.colderfusion.com

Missouri
Kansas City CFUG
www.kcfusion.org

Missouri
St. Louis CFUG
www.psiwebstudio.com/cfug

Montana, Helena
Montana CFUG
http://montanacfug.site-o-matic.com

Nebraska
Omaha CFUG
www.necfug.com

Nevada
Las Vegas CFUG
www.sncfug.com

New Jersey, Raritan
Central New Jersey CFUG
www.freecfm.com/c/cjcfug/index.cfm

New Hampshire
Northern N.E. MMUG
www.mmug.info

New York
Albany CFUG
www.anycfug.org

New York
Long Island CFUG
www.licfug.org

New York
New York CFUG
www.nycfug.org

New York
Rochester CFUG
www.roch-cfug.org

New York
Syracuse CFUG
www.cfugcny.org

North Carolina
Charlotte CFUG
www.charlotte-cfug.org

North Carolina
Fayetteville CFUG
www.schoollink.net/fcfug

North Carolina
Raleigh CFUG
www.ccfug.org

Ohio, Columbus
Ohio Area CFUG
www.oacfug.org

Ohio
Mid Ohio Valley MMUG
www.movcfug.org/

Oklahoma
Oklahoma City CFUG
http://idgweb4.ouhsc.edu/cfug

Oklahoma
Tulsa MMUG
www.tulsacfug.org

Oregon
Eugene CFUG
www.EugeneCFUG.org

Oregon
Portland CFUG
www.pdxcfug.org

Pennsylvania, Harrisburg
Central Penn CFUG
www.centralpenncfug.org

Pennsylvania
Philadelphia CFUG
www.pacfug.org

Pennsylvania
Pittsburgh CFUG
www.orbwave.com/pghcfug

Pennsylvania
State College CFUG
www.cfug-sc.org

Rhode Island
Providence CFUG
www.ricfug.com

Tennessee
Memphis CFUG
http://cfug.dotlogix.com

Tennessee
Nashville CFUG
www.ncfug.org

Texas
Austin CFUG
http://cftexas.net

Texas
Dallas CFUG
www.dfwcfug.org

Texas
San Antonio CFUG
http://samcfug.org

Utah
Salt Lake City CFUG
www.slcfug.org

Vermont, Montpelier
Vermont CFUG
www.mtbytes.com/cfug/index.htm

Virginia
Hampton Roads CFUG
www.hrcfug.org

Virginia
Northern Virginia CFUG
www.cfugorama.com

Virginia
Richmond CFUG
http://richmond-cfug.btgi.net

Virginia, Roanoke
Blue Ridge MMUG
www.brmug.com

Washington D.C.
Washington, D.C. CFUG
www.cfugorama.com

Wisconsin
Milwaukee MMUG
www.metromilwaukee.com/usr/cfug

Wyoming, Jackson
Wyoming MMUG
www.wycfug.org

International

Australia, Melbourne
Australian CFUGs
www.cfug.org.au/

Belgium, Brussels
Belgium CFUG
www.cfug.be

Brazil
Rio de Janerio CFUG
www.cfugrio.com.br

Canada
Edmonton, AB CFUG
http://edmonton.cfug.ca

Canada
Kingston, ON CFUG
www.kcfug.org

Canada
Montreal, QC CFUG
www.kopanas.com/cfugmontreal

Canada
Ottawa, ON CFUG
www.cfugottawa.com

Canada, Ottawa
Ecole Secondary CFUG
www.escgarneau.com/gug

Canada
Toronto, ON CFUG
www.cfugtoronto.org

Canada
Vancouver, BC CFUG
www.cfug-vancouver.com

Canada
Vancouver Island CFUG
www.cfug-vancouverisland.com

Central Europe, Munich
Central Europe CFUG
www.cfug.de

Finland, Helsinki
Finland CFUG
www.cfug-fi.org

France, Valbonne
France CFUG
www.cfug.fr.st

Germany
Frankfurt CFUG
www.cfug-frankfurt.de

Germany
NeussCFUG
www.cfug-nordwest.de

Ireland
Belfast CFUG
www.cfug.ie

Ireland
Cork CFUG
www.corkcoldfusion.com

Italy, Bologna
Italy CFUG
www.ingenium-mmug.org

Japan, Urayasu City
Japan CFUG
http://cfusion.itfrontier.co.jp/jcfug/jcfug.cfm

Malaysia, Kuala Lumpur
Malaysia CFUG
www.coldfusioneer.com/

Pakistan Edu, Lahore Cantt
Pakistan Educational CFUG
www.cfeugpakistan.org

Pakistan, Lahore
Pakistan CFUG
www.cfugpakistan.org

Saudi Arabia, Riyadh
CFUG Saudia
www.cfugsaudia.org

South Africa
Cape Town CFUG
www.coldfusion.org.za

South Africa
Johannesburg CFUG
www.coldfusion.org.za

Sweden
Gothenburg CFUG
www.cfug-se.org

Switzerland
Martin Bürlimann CFUG
www.cfug.ch

Thailand
Bangkok CFUG
http://thaicfug.tei.or.th

Turkey, Izmer
Turkey CFUG
www.cftr.net

United Kingdom, London
UK CFUG
www.ukcfug.org

United Kingdom
Northern England CFUG
www.cfug.org.uk

JAVA Industry >Newsletter
WebServices Industry >Newsletter
XML JOURNAL Industry >Newsletter

DEVELOPER
.SYS-CON
.COM

Macromedia Announces Director MX for Mac OS X

(*San Francisco*) – Macromedia, Inc., has announced Macromedia Director MX for Mac OS X. Director MX is the most powerful multimedia authoring environment for building high-end, rich, interactive content and applications deployed on CD/DVD-ROM, kiosks, and the Web, via Macromedia Shockwave Player. This is the first version of Director for Apple's supermodern operating system, and it takes full advantage of Mac OS X's power and stability.

"Macromedia is entirely committed to the Mac, as all of the Macromedia MX authoring tools are built natively for Mac OS X," said Rob Burgess, chairman and CEO, Macromedia. "Director MX includes support for QuickTime 6 and is optimized to take advantage of the Mac's PowerPC G4 processor with Velocity Engine. Developers are chomping at the bit to use this industry-leading multimedia authoring tool."

"Macromedia is doing terrific work for Mac OS X, and this new version of Director MX is a prime example," said Steve Jobs, Apple's CEO. "Director MX taps into Mac OS X's advanced technologies to deliver a powerful authoring tool for multimedia developers and designers."

Macromedia Director MX has many new features, including seamless, comprehensive integration with the Macromedia Flash MX product family; adoption of the streamlined Macromedia MX user interface; major new workflow efficiencies; and the ability to create accessible content so that rich Director presentations can be enjoyed by people with disabilities. For more information, go to www.macromedia.com/go/director.

Macromedia Studio MX Named Best Development Tool by *PC Magazine*

(*Las Vegas*) – Macromedia, Inc., has announced that Macromedia Studio MX won Best Development Tool at the *PC Magazine* Awards for Technical Excellence presented at The Venetian Resort Hotel Casino in Las Vegas during Fall COMDEX. The 19th annual awards recognize individuals and products that advanced the state of technology and set new standards for technical innovation in 2002.

"Macromedia is honored to receive such an award, given the thousands of products reviewed by *PC Magazine* each year," said Norm Meyrowitz, president of products, Macromedia. "Macromedia Studio MX integrates our market-leading tools like never before to enable developers and companies to provide great end-user experiences."

More than 250,000 customers have already adopted Macromedia Studio MX since its introduction in June, making it the most successful offering in Macromedia history. Macromedia Studio MX includes Macromedia Dreamweaver MX, Macromedia Flash MX, Macromedia Fireworks MX, Macromedia FreeHand 10, and a developer edition of ColdFusion MX. For more information on Macromedia Studio MX, visit www.macromedia.com/go/studiomx.

Ektron Launches Ektron CMS300 XML Content Management Solution

(*Amherst, New Hampshire*) – Ektron, Inc., has annonced the launch of Ektron CMS300. This browser-based system offers powerful content-management capabilities with unique WYSIWYG XML editing functionality. With Ektron CMS300, organizations have powerful XML authoring functionality in an intuitive content-management environment.

Ektron CMS300 is designed to deliver the same robust features found in Ektron's other content-management solutions (see review in this issue), including intuitive work areas for contributors and administrators, scalable workflow, an advanced permissions model, and an Explorer-like content folder design. It includes the browser-based WYSIWYG XML editor, Ektron eWebEditPro+XML.

Features of Ektron CMS300 include improved consistency in Web pages via enforced page layout and content formatting; centralized control over site usability and appearance; reduced review and revision time; the ability to separate content from presentation, via XSLT and CSS; the ability to manage HTML, XHTML, and XML, with an option for adding XML capabilities where and when needed; and business-authored XML for use elsewhere (B2B transactions, CRM and portal environments, enhanced search, data analysis).

Ektron CMS300 runs on Microsoft Windows servers, and works with several Web application servers including Microsoft Active Server Pages (ASP), ASP.NET, Macromedia ColdFusion, and PHP. Ektron. CMS300 is priced at US$4,999 – $19,999. www.ektron.com/cms300.cfm

CFDJ Advertiser Index

ADVERTISER	URL	PHONE	PAGE
ADVERTISER	URL	PHONE	PAGE
ACTIVEPDF	WWW.ACTIVEPDF.COM	949.582.9002	4
CFDYNAMICS	WWW.CFDYNAMICS.COM	866.CFDYNAMICS	17
CFXHOSTING	WWW.CFXHOSTING.COM	866.CFX.HOST	3
COLDFUSION DEVELOPER'S JOURNAL	WWW.COLDFUSIONDEVELOPERSJOURNAL.COM	888.303.5282	29
COLDFUSIONHOSTING.COM	WWW.COLDFUSIONHOSTING.COM	888.250.8100	29
COOLFUSION	WWW.COOLFUSION.COM	631.737.4668	13
CTIA WIRELESS 2003	WWW.CTIASHOW.COM		51
EKTRON	WWW.EKTRON.COM/CMS		41
FUSETALK	WWW.FUSETALK.COM/NEW	866.477.7542	6
HOSTMYSITE.COM	WWW.HOSTMYSITE.COM	877.215.HOST	49
INTERMEDIA	WWW.INTERMEDIA.NET	800.379.7729	52
JDJ STORE	WWW.JDJSTORE.COM	888.303JAVA	35,43
MACROMEDIA	WWW.MACROMEDIA.COM/GO/CFMXLIGHT	877.460.8679	19
MACROMEDIA	WWW.MACROMEDIA.COM/GO/USERGROUPS	877.460.8679	23
MACROMEDIA	WWW.MACROMEDIA.COM/GO/MASTERING		27
NETQUEST	WWW.NQCONTENT.COM		9
NEW ATLANTA COMMUNICATIONS	WWW.NEWATLANTA.COM		2
PACIFIC ONLINE	WWW.PACONLINE.NET	877.503.9870	21
PAPERTHIN	WWW.PAPERTHIN.COM	800.940.3087	39
SYS-CON MEDIA	HTTP://DEVELOPER.SYS-CON.COM		44
SYS-CON SUBSCRIPTIONS	WWW.SYS-CON.COM/SUBOFFER.CFM	888.303.5282	46-47
WEB SERVICES EDGE 2003	WWW.SYS-CON.COM	201.802.3069	31

General Conditions: The Publisher reserves the right to refuse any advertising not meeting the standards that are set to protect the high editorial quality of. All advertising is subject to approval by the Publisher. The Publisher assumes no liability for any costs or damages incurred if for any reason the Publisher fails to publish an advertisement. In no event shall the Publisher be liable for any costs or damages in excess of the cost of the advertisement as a result of a mistake in the advertisement or for any other reason. The Advertiser is fully responsible for all financial liability and terms of the contract executed by the agents or agencies who are acting on behalf of the Advertiser. Conditions set in this document (except the rates) are subject to change by the Publisher without notice. No conditions other than those set forth in this "General Conditions Document" shall be binding upon the Publisher. Advertisers (and their agencies) are fully responsible for the content of their advertisements printed in ColdFusion Developer's Journal. Advertisements are to be printed at the discretion of the Publisher. This discretion includes the positioning of the advertisement, except for "preferred positions" described in the rate table. Cancellations and changes to advertisements must be made in writing before the closing date. "Publisher" in this "General Conditions Document" refers to SYS-CON Publications, Inc.This index is provided as an additional service to our readers. The publisher does not assume any liability for errors or omissions This index is provided as an additional service to our readers. the publisher does not assume any liability for errors or omissions.

A LIMITED TIME SAVINGS

SUBSCRIBE TODAY TO MULTI

Go To www.sys-con.com

and receive your **FREE** CD Gift Package VIA

Each CD is an invaluable developer resource packed with important articles and useful source code!

Pick the CDs to go with your Multi-Pack order

- Pick one CD with your 3-Pack order
- Pick two CDs with your 6-Pack order
- Pick three CDs with your 9-Pack order

□ Web Services Resource CD
□ Java Resource CD
□ WebLogic Resource CD
□ ColdFusion Resource CD
□ XML Resource CD
□ WebSphere Resource CD
□ CF Advisor Complete Works CD

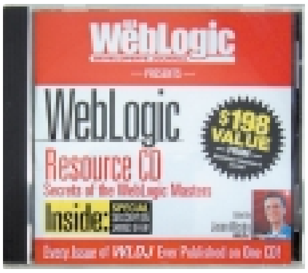
Your order will be processed the same day!

WebServices
PRESENTS
Web Services
Resource CD
Secrets of the Web Services Masters
Inside:
$198 VALUE
Every Issue of WSJ Ever Published on One CD!

More than 1,400 Web services and Java articles on one CD! Edited by well-known editors-in-chief Sean Rhody and Alan Williamson, these articles are organized into more than 50 chapters on UDDI, distributed computing, e-business, applets, SOAP, and many other topics. Plus, editorials, interviews, tips and techniques!
LIST PRICE $198

JAVA DEVELOPERS JOURNAL
PRESENTS
Java
Resource CD
Secrets of the Java Masters
Inside:
$198 VALUE
Every Issue of JDJ Ever Published on One CD!

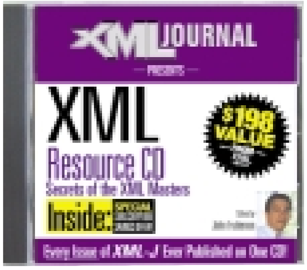
The most complete library of exclusive *JDJ* articles compiled on one CD! Assembled by *JDJ* Editor-in-Chief Alan Williamson, more than 1,400 exclusive articles are organized into over 50 chapters, including fundamentals, applets, advanced Java topics, Swing, security, wireless Java,... and much more!
LIST PRICE $198


WebLogic
PRESENTS
WebLogic
Resource CD
Secrets of the WebLogic Masters
Inside:
$198 VALUE
Every Issue of WLDJ Ever Published on One CD!

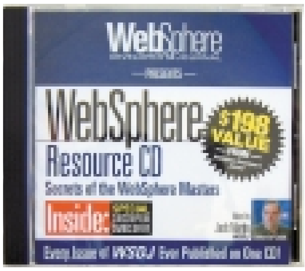
The most complete library of exclusive *WLDJ* articles ever assembled! More than 200 articles provide invaluable information on "everything WebLogic", including WebLogic Server, WebLogic Portal, WebLogic Platform, WebLogic Workshop, Web services, security, migration, integration, performance, training...
LIST PRICE $198

COLDFUSION Developer's Journal
PRESENTS
ColdFusion
Resource CD
Secrets of the ColdFusion Masters
Inside:
$198 VALUE
Every Issue of CFDJ Ever Published on One CD!

The most complete library of exclusive *CFDJ* articles on one CD! This CD, edited by *CFDJ* editor-in-chief Robert Diamond, is organized into more than 30 chapters with more than 400 exclusive articles on CF applications, custom tags, database, e-commerce, Spectra, enterprise CF, error handling, WDDX... and more!
LIST PRICE $198


XML JOURNAL
PRESENTS
XML
Resource CD
Secrets of the XML Masters
Inside:
$198 VALUE
Every Issue of XML-J Ever Published on One CD!

The largest and most complete library of exclusive *XML-Journal* articles compiled on one CD! Edited by well-known editors-in-chief Ajit Sagar and John Evdemon, these articles are organized into more than 30 chapters containing more than 1,150 articles on Java & XML, XML & XSLT, <e-BizML>, data transition... and more!
LIST PRICE $198


WebSphere
PRESENTS
WebSphere
Resource CD
Secrets of the WebSphere Masters
Inside:
$198 VALUE
Every Issue of WSDJ Ever Published on One CD!

The most up-to-date collection of exclusive *WSDJ* articles! More than 200 articles offer insights into all areas of WebSphere, including Portal, components, integration, tools, hardware, management, sites, wireless programming, best practices, migration...
LIST PRICE $198

The complete works of *CF Advisor* are now on one CD! Check out over 200 exclusive articles on topics such as e-commerce, custom tags, Fusebox, databases, object-oriented CF, WDS, upgrading CF, wireless, Verity, and more. Plus, find interviews, editorials, and source code!
LIST PRICE $198

Subscribe Online Today

OFFER FROM SYS-CON MEDIA

PLE MAGAZINES ONLINE

AND SAVE UP TO $400 AND RECEIVE UP TO 3 FREE CDs!

RECEIVE YOUR DIGITAL EDITION ACCESS CODE INSTANTLY WITH YOUR PAID SUBSCRIPTIONS

Pick a 3-Pack, a 6-Pack or a 9-Pack ☑

TO ORDER: Choose the Multi-Pack you want to order by checking next to it below. Check the number of years you want to order. Indicate your location by checking either U.S., Canada/Mexico or International. Then choose which magazines you want to include with your Multi-Pack order.

□ 3-Pack	□ 1YR	□ 2YR	□ U.S.	□ Can/Mex	□ Intl.
□ 6-Pack	□ 1YR	□ 2YR	□ U.S.	□ Can/Mex	□ Intl.
□ 9-Pack	□ 1YR	□ 2YR	□ U.S.	□ Can/Mex	□ Intl.

□ **Java Developer's Journal**

U.S. - Two Years (24) Cover: $144	You Pay: $89 /	Save: $55 + FREE $198 CD
U.S. - One Year (12) Cover: $72	You Pay: $49.99 /	Save: $22
Can/Mex - Two Years (24) $168	You Pay: $119.99 /	Save: $48 + FREE $198 CD
Can/Mex - One Year (12) $84	You Pay: $79.99 /	Save: $4
Int'l - Two Years (24) $216	You Pay: $176 /	Save: $40 + FREE $198 CD

□ **Web Services Journal**

U.S. - Two Years (24) Cover: $168	You Pay: $99.99 /	Save: $68 + FREE $198 CD
U.S. - One Year (12) Cover: $84	You Pay: $69.99 /	Save: $14
Can/Mex - Two Years (24) $192	You Pay: $129 /	Save: $63 + FREE $198 CD
Can/Mex - One Year (12) $96	You Pay: $89.99 /	Save: $6
Int'l - Two Years (24) $216	You Pay: $170 /	Save: $46 + FREE $198 CD
Int'l - One Year (12) $108	You Pay: $99.99 /	Save: $8

□ **.NET Developer's Journal**

U.S. - Two Years (24) Cover: $168	You Pay: $99.99 /	Save: $68 + FREE $198 CD
U.S. - One Year (12) Cover: $84	You Pay: $69.99 /	Save: $14
Can/Mex - Two Years (24) $192	You Pay: $129 /	Save: $63 + FREE $198 CD
Can/Mex - One Year (12) $96	You Pay: $89.99 /	Save: $6
Int'l - Two Years (24) $216	You Pay: $170 /	Save: $46 + FREE $198 CD
Int'l - One Year (12) $108	You Pay: $99.99 /	Save: $8

□ **XML-Journal**

U.S. - Two Years (24) Cover: $168	You Pay: $99.99 /	Save: $68 + FREE $198 CD
U.S. - One Year (12) Cover: $84	You Pay: $69.99 /	Save: $14
Can/Mex - Two Years (24) $192	You Pay: $129 /	Save: $63 + FREE $198 CD
Can/Mex - One Year (12) $96	You Pay: $89.99 /	Save: $6
Int'l - Two Years (24) $216	You Pay: $170 /	Save: $46 + FREE $198 CD
Int'l - One Year (12) $108	You Pay: $99.99 /	Save: $8

□ **WebLogic Developer's Journal**

U.S. - Two Years (24) Cover: $360	You Pay: $169.99 /	Save: $190 + FREE $198 CD
U.S. - One Year (12) Cover: $180	You Pay: $149 /	Save: $31
Can/Mex - Two Years (24) $360	You Pay: $179.99 /	Save: $180 + FREE $198 CD
Can/Mex - One Year (12) $180	You Pay: $169 /	Save: $11
Int'l - Two Years (24) $360	You Pay: $189.99 /	Save: $170 + FREE $198 CD
Int'l - One Year (12) $180	You Pay: $179 /	Save: $1

□ **ColdFusion Developer's Journal**

U.S. - Two Years (24) Cover: $216	You Pay: $129 /	Save: $87 + FREE $198 CD
U.S. - One Year (12) Cover: $108	You Pay: $89.99 /	Save: $18
Can/Mex - Two Years (24) $240	You Pay: $159.99 /	Save: $80 + FREE $198 CD
Can/Mex - One Year (12) $120	You Pay: $99.99 /	Save: $20
Int'l - Two Years (24) $264	You Pay: $189 /	Save: $75 + FREE $198 CD
Int'l - One Year (12) $132	You Pay: $129.99 /	Save: $2

□ **Wireless Business & Technology**

U.S. - Two Years (24) Cover: $144	You Pay: $89 /	Save: $55 + FREE $198 CD
U.S. - One Year (12) Cover: $72	You Pay: $49.99 /	Save: $22
Can/Mex - Two Years (24) $192	You Pay: $139 /	Save: $53 + FREE $198 CD
Can/Mex - One Year (12) $96	You Pay: $79.99 /	Save: $16
Int'l - Two Years (24) $216	You Pay: $170 /	Save: $46 + FREE $198 CD
Int'l - One Year (12) $108	You Pay: $99.99 /	Save: $8

□ **WebSphere Developer's Journal**

U.S. - Two Years (24) Cover: $360	You Pay: $169.99 /	Save: $190 + FREE $198 CD
U.S. - One Year (12) Cover: $180	You Pay: $149 /	Save: $31
Can/Mex - Two Years (24) $360	You Pay: $179.99 /	Save: $180 + FREE $198 CD
Can/Mex - One Year (12) $180	You Pay: $169 /	Save: $11
Int'l - Two Years (24) $360	You Pay: $189.99 /	Save: $170 + FREE $198 CD
Int'l - One Year (12) $180	You Pay: $179 /	Save: $1

□ **PowerBuilder Developer's Journal**

U.S. - Two Years (24) Cover: $360	You Pay: $169.99 /	Save: $190 + FREE $198 CD
U.S. - One Year (12) Cover: $180	You Pay: $149 /	Save: $31
Can/Mex - Two Years (24) $360	You Pay: $179.99 /	Save: $180 + FREE $198 CD
Can/Mex - One Year (12) $180	You Pay: $169 /	Save: $11
Int'l - Two Years (24) $360	You Pay: $189.99 /	Save: $170 + FREE $198 CD
Int'l - One Year (12) $180	You Pay: $179 /	Save: $1

3-Pack

Pick any 3 of our magazines and save up to $275⁰⁰ Pay only $175 for a 1 year subscription plus a FREE CD

- 2 Year – $299.00
- Can/Mex – $245.00
- International – $315.00

6-Pack

Pick any 6 of our magazines and save up to $350⁰⁰ Pay only $395 for a 1 year subscription plus 2 FREE CDs

- 2 Year – $669.00
- Can/Mex – $555.00
- International – $710.00

9-Pack

Pick all 9 of our magazines and save up to $400⁰⁰ Pay only $495 for a 1 year subscription plus 3 FREE CDs

- 2 Year – $839.00
- Can/Mex – $695.00
- International – $890.00

www.sys-con.com/suboffer.cfm

SYS-CON MEDIA

ColdFusion Developer's Journal CFDJ WORLD CLASS PRODUCT ★★★ AWARD

CMS 200 from Ektron, Inc.

No-frills content management on a budget

I have always considered Ektron's eWebEditPro to be the best, easiest to integrate, and most innovative Web-based WYSIWYG component available. Many software vendors apparently concur with this assessment, for you can find the product embedded in a number of CMS systems ranging from Allaire Spectra to Microsoft Content Management Server 2002.

By Steve Drucker

It should come as no surprise then that Ektron, seeking to capitalize on the success of their award-winning editor, launched a series of content-management products that interoperate with ColdFusion 4.x, 5.x, Microsoft Active Server Pages, Microsoft ASP.NET, and PHP. Ektron CMS 200 is their mid-range offering, falling between their entry-level product (CMS 100) and their enterprise CF-only product eMPower.

Out-of-the-Box Functionality

Ektron's CMS 200 includes the following "out-of-the-box" features:

- "Anywhere Authoring" through the award-winning eWebEditPro editor supporting spell checking, import/export to Microsoft Word, CSS, XHTML, Section 508.
- Limited multilingual support (English, French, German).
- Content organized into "objects" which may be published to multiple locations in a site and syndicated to other sites.
- Simple security model for logins, viewing, and editing content; integration with Active Directory planned for Q1 2003.
- WYSIWYG content is versioned
- Content may go through a configurable approval process prior to publishing.
- Management reports detail content contributor activities.

Is WYSIWYG Editing the Most Important Facet of a CMS?

According to a June 2002 Forrester Research report entitled, "Enterprise Content Management Delusions," improved usability ranked as the most pressing need of respondents. To be sure, modifying Web content through a Web browser has never been a picnic. However, CMS 200's reliance on the eWebEditPro editor serves it well. Content contributors may participate using either IE 4.x+ or Netscape 4.x+ on a PC platform (Macintosh support is currently unavailable).

While its use requires a significant client download (up to 4.5MB), its behavior more closely approximates Microsoft Word than any other editor I've tested. As an added bonus, eWebEditPro can actually invoke Word as an OLE-Automation object so that you can modify and transfer content seamlessly. Automatic spell checking within eWebEditPro is also available; however, this feature tended to bog down my system with Word running hidden in the background.

Prevent Content Contributors from Using the <BLINK> Tag

There is an old saying: "With great power comes great responsibility." As an administrator, you can restrict content authors to choosing from a list of approved fonts and styles in their content. Surprising, however, is the apparent inability of an administrator to restrict access to specific editor features such as HTML view, table creation, or color selection. Additionally, you are restricted to using a single content "type" consisting of the following attributes:

- Title
- Comment
- Start date/end date
- WYSIWYG content
- WYSIWYG summary
- Metadata (customizable)

Creating Templates Using the Developer API

The CMS 200 developer API is comprised of eight CFML custom tags that you will insert into CFM files to create dis-

TAG	FUNCTION
<cf_ecmlogin>	Places login, logout, and admin navigation buttons on-screen
<cf_ecmContentBlock>	Displays database content on a page
<cf_ecmSearch>	Displays a text box, with a search button on the template
<cf_ecmSearchDisplay>	Displays search results from criteria entered into a <cf_ecm search> box
<cf_ecmListSummary>	Displays a list of content summaries on a page.
<cf_ecmSingleSummary>	Displays the summary of one content block on a page
<cf_ecmCollection>	Retrieves links to articles and can be used to drive dynamic navigation and output "teaser" content
<cf_ecmMetaData>	Displays metadata for one or more content blocks

Table 1: CFML custom tags

play templates for your content (see Table 1). Ektron ships a Dreamweaver MX extension that helps facilitate this process by inserting analogous ASP call syntax into your document. A CFML version of the extension should be available in early Q1.

The CFML custom tags that ship with the product are unencrypted and customization is encouraged. However, since the underlying kernel is actually a set of Windows .DLL files and most administrative functions are coded in ASP, you may find advanced customization somewhat challenging.

No Support for ColdFusion MX

Ektron's CMS core libraries (excluding the eMPower product) were developed as a series of Windows-based DLLs invoked through COM. ColdFusion 4.x, 5.x support is achieved by supplying developers with a series of CFML custom tag wrappers that invoke these COM objects through <CFOBJECT> invocations. Like most other vendors in the same COM predicament, Ektron plans to release a future version of the product architected around a more open, Web services-based model. At the time of this writing, however, no release date for CFMX compatibility is being disclosed.

Does a Lightweight CMS Inevitably Lead to High Total Cost of Ownership?

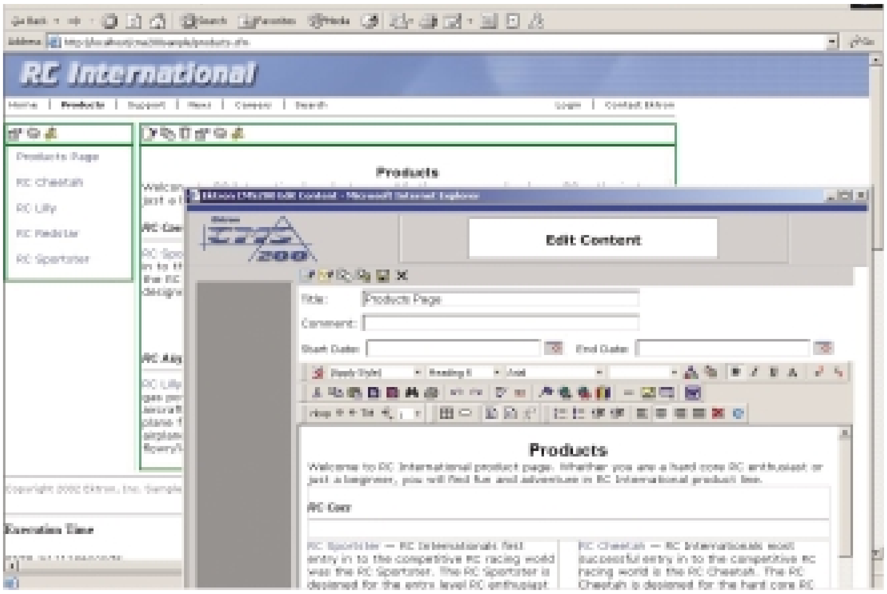
Carefully consider your total cost of ownership before purchasing any content management system. This includes not only the price of the CMS, but also the application server, database server, and perhaps most significantly – the development time to create your required functionality. Twenty years in software development have taught me the following:

1. 90% of customer requirements can usually be met in 10% of the budget. Meeting the remaining 10% of the requirements usually costs 90% of the budget.
2. "Enterprise" software generally costs as much as the "Starship Enterprise" and frequently performs as well as that ship's transporter.
3. Murphy was an optimist.

In other words, choose a CMS that most closely meets your functional requirements with "out-of -the-box" functionality. Ektron's CMS 200 carries one of the lowest starting price-points in the industry. At $2,999 USD for one URL and 10 content contributors operating using an MS Access datasource and an ASP application server model, small workgroups or departments could feasibly deploy the system for well under $10K on a single server (including hardware, excluding services). The functionality, as described in Table 1 is limited; however, if it fulfills many of your requirements, then you've just discovered a hard-to-find bargain in a very competitive industry.

Conclusion

Ektron's CMS 200 fulfills the somewhat peculiar need to add lightweight content management to a preexisting hybrid HTML, ASP, PHP, or CF 4.x/5.x Web site. If you're on a tight budget supporting 20 (or fewer) content contributors, have no immediate plans to support ColdFusion



RC International

Home | Products | Support | News | Careers | Search | Login | Contact Ektron

Products Page
RC Cheetah
RC Lilly
RC Pedal Lite
RC Sportster

Products

Ektron CMS200 Edit Content - Microsoft Internet Explorer

Edit Content

Title: Products Page
Comment:
Start Date: End Date:

Products

Welcome to RC International product page. Whether you are a hard core RC enthusiast or just a beginner, you will find fun and adventure in RC International product line.

RC Cars

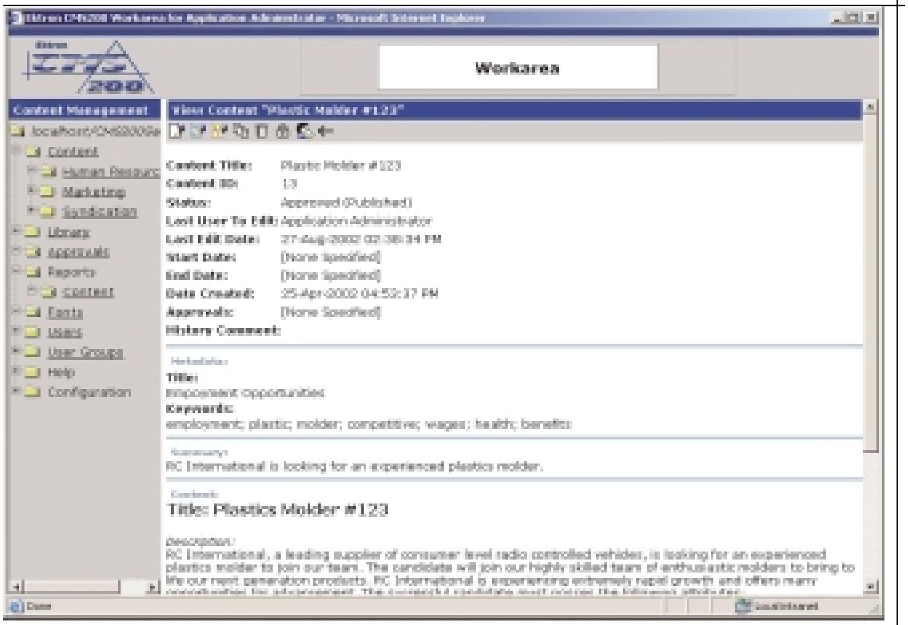
RC Sportster — RC International's first entry in to the competitive RC racing world was the RC Sportster. The RC Sportster is designed for the entry level RC enthusiast.

RC Cheetah — RC International's most successful entry in to the competitive RC racing world is the RC Cheetah. The RC Cheetah is designed for the hard core RC

Copyright 2002 Ektron, Inc. Sample

Execution Time

Figure 1: Anywhere authoring in CMS 200



Ektron CMS200 Workarea for Application Administrator - Microsoft Internet Explorer

Workarea

Content Management
localhost/CMS200Sa
Content
Human Resourc
Marketing
Syndication
Library
Approvals
Reports
Content
Fonts
Users
User Groups
Help
Configuration

View Content "Plastic Molder #123"

Content Title: Plastic Molder #123
Content ID: 13
Status: Approved (Published)
Last User To Edit: Application Administrator
Last Edit Date: 27-Aug-2002 02:38:34 PM
Start Date: [None Specified]
End Date: [None Specified]
Date Created: 25-Apr-2002 04:53:37 PM
Approvals: [None Specified]
History Comment:

Metadata:
Title:
Employment Opportunities
Keywords:
employment; plastic; molder; competitive; wages; health; benefits

Summary:
RC International is looking for an experienced plastics molder.

Content:
Title: Plastics Molder #123

Description:
RC International, a leading supplier of consumer level radio controlled vehicles, is looking for an experienced plastics molder to join our team. The candidate will join our highly skilled team of enthusiastic molders to bring to life our next generation products. RC International is experiencing extremely rapid growth and offers many

Done Local intranet

Figure 2: The "CMS 200" administrative system, coded entirely in ASP, allows users to locate content through a tree-based metaphor

MX, and have minimal CMS requirements, then Ektron's CMS 100/200/300 product line will provide exceptional value as a "no-frills" entry-level system. Otherwise, you may want to consider the more fully featured "native CF" implementations of Ektron eMPower, CommonSpot 3.0 from Paperthin, Inc. (reviewed in ***CFDJ*** Vol. 4, issue 10), and NQcontent from NetQuest, which I will be reviewing in a future issue.

About the Author

Steve Drucker is the CEO of Fig Leaf Software, a Macromedia premier solutions and training partner with offices in Washington, DC, and Atlanta, GA. He is also a certified Macromedia instructor and a Macromedia-certified Dreamweaver, Flash, and Advanced ColdFusion MX developer.

sdrucker@figleaf.com

Vitals: CMS 200

Ektron, Inc.
5 Northern Blvd. • Amherst, NH 03031
Phone: 603-594-0249
Web: www.ektron.com

Test Environment:
Dell Inspiron 5000, 450MHz P-III
ColdFusion 5.0, 512MB RAM
Windows 2000, SP 2
Microsoft Access

Pricing:
U.S. $2,999 (one URL, 10 contributors)
U.S. $12,999 (one URL, unlimited contributors)

PRODUCT SNAPSHOT

Target audience:
Small, price-sensitive organizations and workgroups that require simple anywhere authoring and workflow.

Pros:
- Low cost
- Utilizes the exceptionally robust eWebEditPro WYSIWYG editor
- Supports editing content through MS Word

Cons:
- No timetable for ColdFusion MX support
- Requires Microsoft ASP application server
- No support for Macintosh content contributors
- Cumbersome process for creating new sites, adding full-text search capabilities (through Microsoft Index Server)
- WYSIWYG features may overwhelm some nontechnical users

Client platform:
Microsoft Internet Explorer 4.x+, Netscape 4.x / PC only. Client download of the Ektron eWebEditPro editor is required and may vary between 1.5–4.5 MB.
Server OS: Windows only (requires Microsoft Active Server Pages). Developer API for Macromedia ColdFusion 4.x, 5.x. PHP, Microsoft ASP, ASP.NET
Database support: Microsoft Access, Microsoft SQL Server, MySQL

Note:

Since this review was written, Ektron has unveiled its newest product, Ektron CMS300. This content-management solution, base-priced at US$4,999, adds powerful XML functionality to Ektron CMS200.