Get in the Right Group 22

macromedia®

One little box.
A whole lot of power.

Put it to work for you.
The shortest distance between you
and powerful web applications.

Full speed ahead. The release of Macromedia ColdFusion MX 7 is changing the whole game. This groundbreaking release introduces new features to help address your daily development challenges. These features include:

› Structured business reporting? Check. Printable web content? Check.

ColdFusion MX 7 provides a structured business reporting solution that will have you creating detailed business reports for anyone who needs them. You can also dynamically transform any web content into high-quality, printable documents with ease. The days of needing a third-party application to generate dynamic reports are going, going, gone.

› Make rapid J2EE development a reality.

So, you're heavily invested in J2EE but would love to complete projects in less time? ColdFusion MX 7 is your answer: It delivers RAD productivity on top of the power and scalability of J2EE and deploys onto standard J2EE server environments.

› New mobile applications are a go.

Innovative new features enable ColdFusion MX 7 applications to reach beyond the web. So you can rapidly create new applications, or extend existing ones, to connect with mobile phones and IM clients using a variety of protocols. The new world of mobile communications is exciting, and this your invitation to the party.

To learn more or take ColdFusion MX 7 for a test drive, go to:
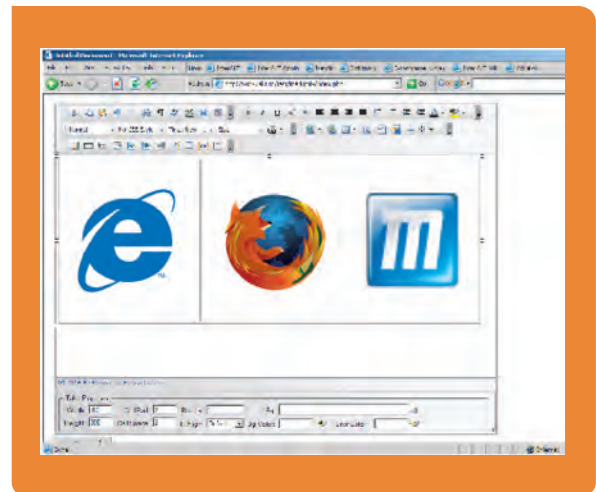macromedia.com/go/cfmx7_demo

KTML4 Word editing in browser

Along with the most affordable **UNLIMITED** licence

Revolutionary Image Editor
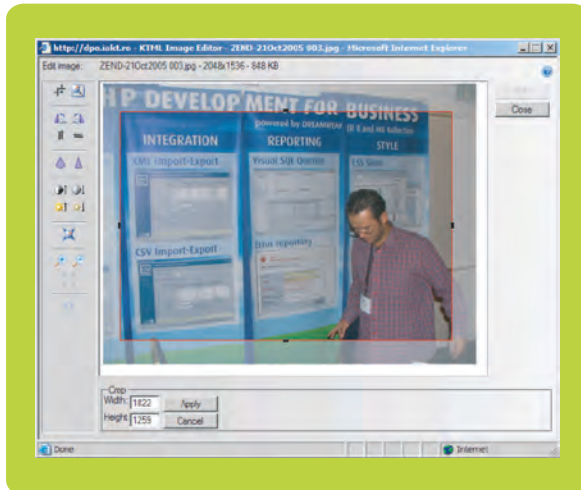
Word-like visual CSS styles
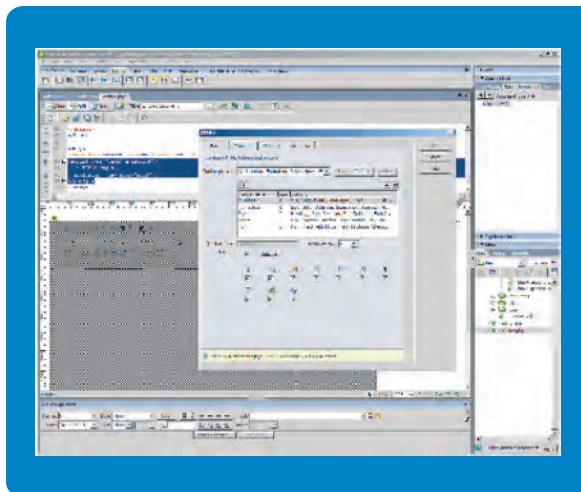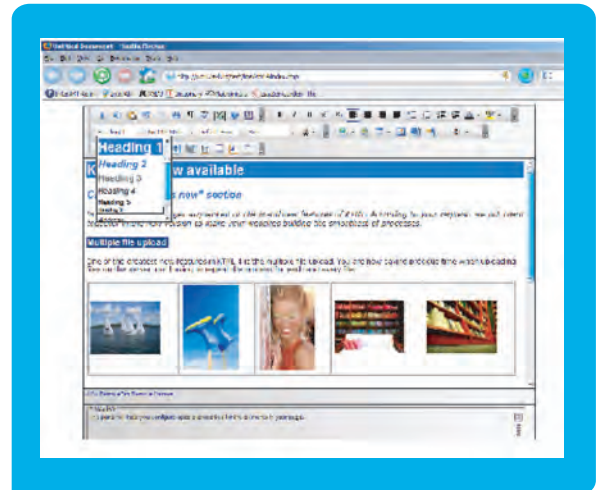
Fast Dreamweaver integration

Instant paste from Word
Incredible speed
Easy to use Word-like toolbars
Improved CSS authoring
Remote File Explorer
XHTML 1.1 compliant

Wide browser compatibility

Multiple file upload at once
HTML Table Editor
Support for multimedia (Flash, Avi)
Documents management (.doc, .pdf)
Page templates
WAI compliant

Please visit **www.interaktonline.com/ktml4/** for details

work smart

Inter**akt**

ColdFusion Hosting is our Complete Focus

⟩ POWERFUL HOSTING PLANS
FREE SQL server access | FREE account setup | Unlimited email accounts Generous disk space & data transfer
30 day money-back guarantee | Great value

⟩ RELIABLE NETWORK
99.99% average uptime | State-of-the-art data center with complete redundancy in power, HVAC, fire suppression,
bandwidth and security | 24/7 network monitoring

⟩ FANTASTIC SUPPORT SERVICES
24/7 suppport services | Knowledgeable phone support | We focus on your individual needs

CFDynamics

866.233.9626 ⟩ CFDYNAMICS.COM

For years we have been involved in the Cold Fusion community and have come to know what developers and project managers look for in a web host. The combination of our powerful hosting plans, reliable network, and fantastic suppport sets us apart from other hosts.
Real service. Real satisfaction. Real value. Real support. Real Freedom.

macromedia® **ALLIANCE PARTNER**

BlueDragonCertified
V.I.P. Hosting Partner

This Is an Exciting Time to Be a ColdFusion Developer

By Simon Horwith

I'm shocked… shocked, I say, at just how much the ColdFusion landscape is changing. I'm specifically impressed with the impact that recent trends and events have had on the community. This editorial is a look at some recent events within the community and how they will impact things to come.

I've been speaking with folks at Adobe and it appears that most of the transition period stuff is out of the way – and the developers there have been able to keep in full swing all the while! Now that the unification process is near completion, one can only imagine that it's full steam ahead. Adobe has announced several ColdFusion-related job openings: new positions and old, which is a strong indicator that they're beefing-up in-house CF expertise. After several talks with folks at Adobe and some personal observations, I can say with confidence that this acquisition is the best thing that could have ever happened to ColdFusion. One thing that many of you may have already noticed are the changes to macromedia.com. Adobe has begun integrating the adobe.com and macromedia.com sites and some of the Adobe product information is now prominently displayed on the Macromedia homepage. Many of the links from the Macromedia site now lead in to the appropriate sections of the Adobe domain. When the sites are completely integrated, we should have one excellent developer resource to go to. What's blown me away more than anything though is what's been happening on the labs site (http://labs.adobe.com/).

First, Adobe's put a bunch of new products up there in a very short amount of time, including Lightroom (a cool-looking tool for digital photographers) and the new XMP Toolkit (for working with metadata inside of image, document, and video files). They've wasted no time making good use of the labs site, a site dedicated to reaching out to the developer community by introducing new products very early. This is a good sign. Second, and most notable to myself and to CF developers in general, is the Flex 2 content on the labs site.

You can now download public betas (beta 2 at the time of this writing) of several Flex 2 products. These include Flex Builder 2, the Flex Framework and SDK, charting components, Flex Data Services, Flash Player 8.5, several cool ActionScript 3 libraries, "Mystic" – the Flex/ColdFusion connectivity module, and the Flex-AJAX bridge. I've been spending the majority of my time playing with all the new Flex offerings, and they are sweet…*especially* if you're a ColdFusion developer. In addition to being able to easily communicate with your CF server from Flex, there are now a bunch of cool wizards and RDS support inside of Eclipse, which allow

About the Author

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

ColdFusion developers to do great things. I'll save the details for an upcoming Flex super issue. At this time, if nothing else, I strongly recommend installing Flex Builder 2 and then taking a look at the Flex AJAX Bridge and Mystic – in the near future these products are likely to be commonplace in most ColdFusion shops. It's also worth mentioning that good documentation is beginning to be made available, as are many terrific sample applications. Adobe is currently holding a Flex Developer Derby – a contest to see what kind of cool applications people are building with the Flex 2 beta products. You can read the contest details on the *CFDJ* site and in this issue.

Why do I find all this Flex 2 stuff shocking? What I'm amazed at are the number of developers and companies I've already encountered that are planning major Flex 2 deployments – often times as the new presentation tier for existing ColdFusion applications. Remember, the product is still in beta; I've never seen such a quick adoption rate and so much excitement over any beta product before. It's going to be very exciting to see what kinds of applications the CF Community produces when Flex 2 is officially released.

The popularity of Flex will not only impact our application UIs – it will have an impact on how we develop our CF apps. As I mentioned in a recent editorial, in the context of Flex 2 and Web 2.0 applications in general, ColdFusion fills the role of providing a services tier. This means more focus on CFC development and how those CFCs are written. Something that developers should already be doing, but wanting to use Flex 2 might be the nudge that the community needs.

The other nudge toward more developers making better use of ColdFusion Components comes in the form of many of the frameworks, APIs, and other open source projects that have been released and are taking the community by storm. Many of these frameworks make heavy use of CFCs and encourage developers to make use of ColdFusion Components in their applications. Speaking of frameworks, the feedback from the January issue of *CFDJ* and from the CFPetMarket site has been terrific – I'm glad so many of you have

found that to be a useful resource. To those of you who have e-mailed me about the documentation for SAM, I know you're all eager and I apologize for the delay, but it is coming. In the meantime, if you haven't downloaded the CFPetmarket sample I wrote, or the sample from last year's MAX, definitely visit my blog (http://www.horwith.com), follow links to those, and check out the code.

Last, I'd like to mention a couple of noteworthy changes to the community member roster that will also have an impact on things to come. Tim Buntel, the long-time product manager for ColdFusion, resigned from his post in order to pursue other interests. What actually happened is that Tim was given the opportunity to get back to his roots (education), which I'm sure will be great for him. Tim was a great voice for the company, a terrific entertainer (especially on stage at MAX), and did a fantastic job as product manager. I wish Tim success in all his endeavors…he will definitely be missed.

Another position vacated very recently was that of CTO at New Atlanta. Charlie Arehart announced that after three years with the company he has resigned in order to pursue his CF community interests as well as other personal and professional interests. Charlie was extremely active in the ColdFusion community in its early days (and was very active at *CFDJ*)… all the way up to his assuming the CTO role at New Atlanta. Now, it appears that Charlie is back. His return to being an active member and participant in the ColdFusion community is very good news. Charlie's a smart guy with a lot (a lot) to say. You can keep up with Charlie's activities at http://www.carehart.org/.

What the availability of new ColdFusion and Flex 2 product offerings boils down to is the equivalent of an adrenaline shot straight to the heart of the ColdFusion community. We have more tools, more options, and a more robust environment for meeting the growing demands placed on Web applications in the twenty-first century. Coupled with a strong and growing community of developers, this means a more open exchange of ideas and information. Like it or not, the landscape is changing… and there has never been a more exciting time to be a ColdFusion developer.

TABLE OF CONTENTS

Get in the Right Group
By James Edmunds...22

Strip Tease
By Isaac Dealey...26

What's New with Flex, ColdFusion, and More...
Interview by Simon Horwith... 38

WEB ALL-STAR seeks an Integrated Software Suite that's up for anything. Understand my need to create cool graphics one day, while tweaking CSS the next. Should be intuitive, multi-platform and not phased by my turbo-charged pace. High-maintenance is a no-no.

Different people. Different needs. One suite solution.

With the latest versions of Macromedia Dreamweaver®, Flash® Professional, Fireworks®, Contribute™ and FlashPaper™ the new Studio 8 is quite a catch. To meet Studio 8 and find all the tools you need to design, develop and maintain online experiences, visit www.macromedia.com/go/studio8_mxdj

macromedia®
STUDIO 8

Making the Switch to CFEclipse

The obvious alternative

By Jeffry Houser

I've been using CFEclipse for about four months. When I started writing ColdFusion code, ColdFusion Studio was the editor of choice. Slowly I moved over to Dreamweaver as Homesite's support started to wane.

About a year ago, when I wrote an article on Dreamweaver (http://coldfusion.sys-con.com/read/48232.htm), I used half of Dreamweaver and half of Homesite. Today, as I write this, half my development is done in CFEclipse and the other half in Dreamweaver. Unfortunately I used Dreamweaver as a really advanced text editor, something it's not that great at. CFEclipse seemed like the obvious alternative so I got it installed and I'm slowly learning its power. The learning curve for CFEclipse has been steep, since, coming from Dreamweaver, its interface is counterintuitive. Assuming that other people are having the same problem, I thought I'd share my thoughts.

Installing CFEclipse

CFEclipse is built on top of Eclipse, the popular Open Source editor used by many Java Programmers. There are plug-ins available for just about anything you could want to do, including Web development, UML modeling, database access, and version control with utilities such as Surround or CVS. CFEclipse is just another plug-in that makes Eclipse operate like ColdFusion Studio/HomeSite+. It has tag insight, color-coding, automatic indentation, and snippets.

Unfortunately, there's no CFEclipse install/setup wizard like you'd find in many other products such as Dreamweaver or ColdFusion. So the first step in setting up CFEclipse is to download Eclipse. First you need to install Eclipse and then you'll have to install the CFEclipse plug-ins. You can download Eclipse from the Eclipse.com web site at http://www.eclipse.org/downloads/.

Eclipse is unique compared to any other product I've used in having a no-install install. Once you unpack the file you downloaded (I grabbed a Win32 zip, however. tar and gz options also exist), the product is installed. That's all you need to run the program. There are no ini files and no registry installs. You're good to go. I unzipped the zip into a directory called "C:\Eclipse." Click on the eclipse.exe in that directory and this window will pop up similar to Figure 1.

This dialog asks you to select your default workspace. A workspace is how Eclipse manages projects, plug-ins, and other settings. What would you use workspaces for? Perhaps you'll want to set up a workspace for Java development and one for ColdFusion development. Or, perhaps you want a different workspace for each project you're working on. If you like me, you might want a workspace for each client. For purposes of this tutorial, you can use the default settings for your workspace.

The next step is to install the CFEclipse plug-in in your Eclipse workspace. From the "Help" menu, select "Software Updates" then "Find and Install." This will bring up the Install/Update window with two options: Search for updates to current features and Search for new features to install. Select "Search for New Features To Install" and click Next. You should see a screen as shown in Figure 2

Click the New Remote Site button and this window should show up (see Figure 3).

For the name enter CFEclipse. For the URL enter http://www.cfeclipse.org/update. Click OK and CFEclipse should be added to your sites. Make sure to select it and click Finish. CFEclipse will show a new window with all available updates (see Figure 4).

You have two options: either install the latest stable release of CFEclipse or the Bleeding Edge release. The Bleeding Edge release is usually an "in process" version. I prefer to go with the stable releases. Check what you want to install and click Next. Read through the license and click "I accept the terms of the license agreement." Click Next to review your updates. Then click Finish to install. CFEclipse will download. You'll see a feature verification window. (This is your last chance to cancel the install if you're having cold feet). Click Install then Yes to restart the workbench. CFEclipse is installed.
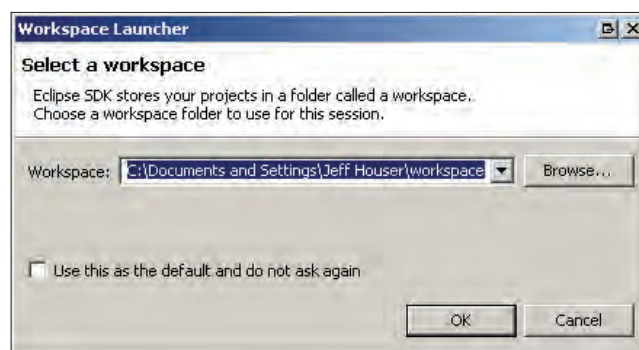
Figure 1 Select a workspace
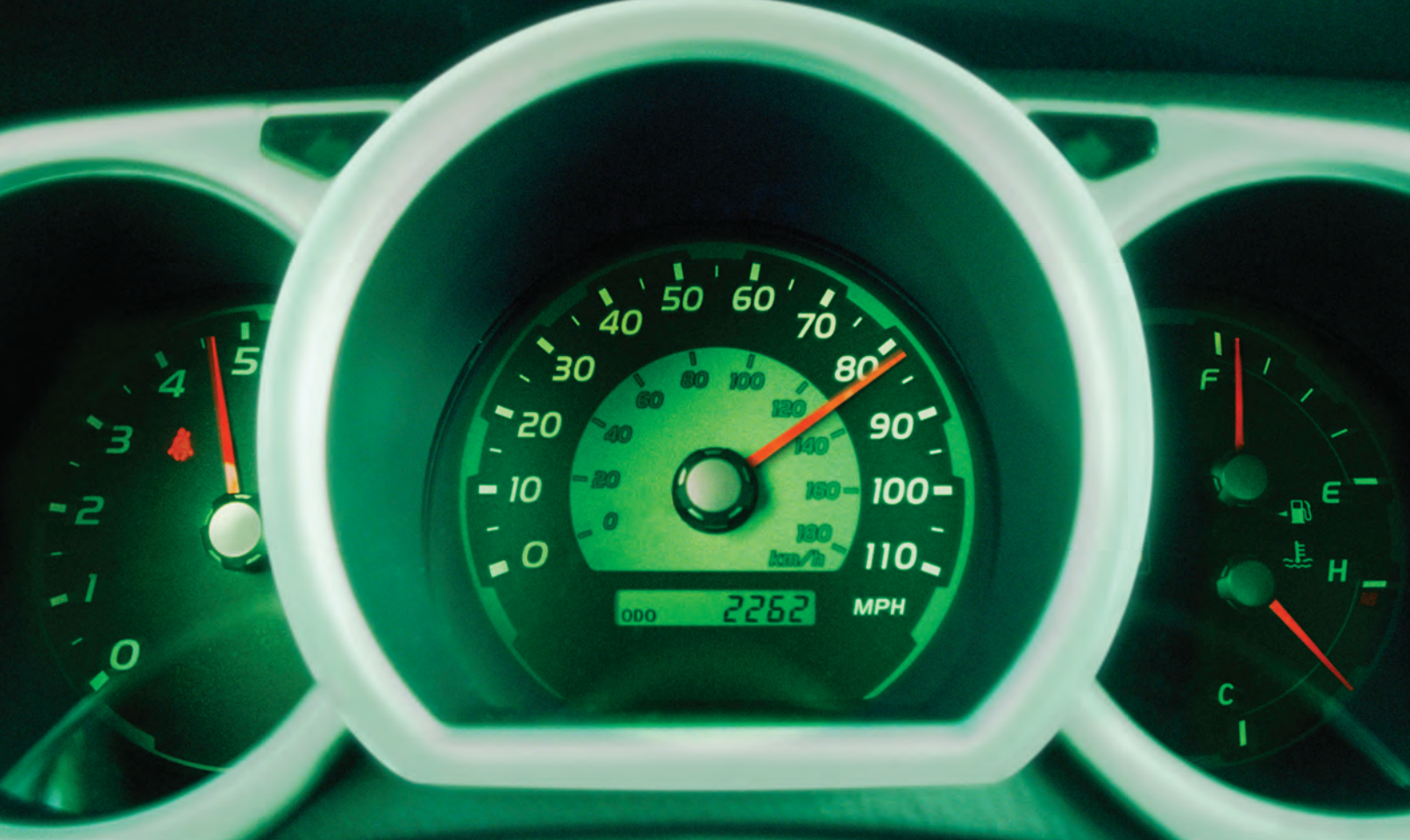
Rev Up Your Flash Video

Stay Ahead of the Competition With VitalStream and the Enhanced Video Features in Flash 8

With over two years of experience in delivering much of today's most popular media, VitalStream® is the first and most experienced Flash™ video streaming service provider.

Enhanced Flash 8 Video Features:

- New VP6 codec delivers higher quality video at the same bit rate
- 8-bit alpha channel transparency enables you to blend video with other elements
- Improved live video capabilities

VitalStream Complete Toolset for Flash:

- MediaConsole®
- MediaOps™ SDK
- Flash Authentication
- Reporting Dashboard

Integrate Streaming Media Into Your Flash Projects

Take Advantage of the Enhanced Video Features in Macromedia Flash 8
Call (800) 254-7554 or Download Tutorials at www.vitalstream.com/go/mxdj

macromedia®
FLASH VIDEO
STREAMING SERVICE

Call (800) 254-7554
Visit www.vitalstream.com

VitalStream®

You'll find a much more detailed account of this install process in the ACME Guide located at http://www.stephencollins.org/acme/. ACME stands for Apache, ColdFusion, MySQL, and Eclipse and talks about setting up all them in a development environment.
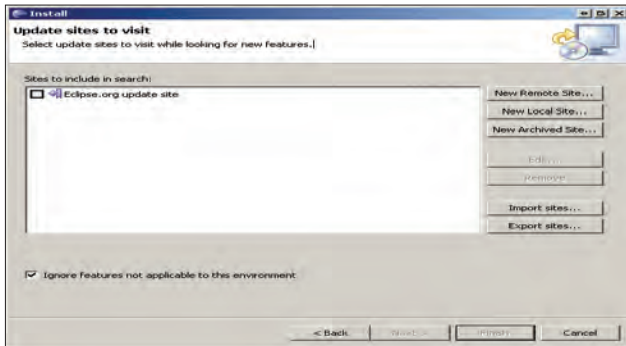
Figure 2 Install the Eclipse plug-ins
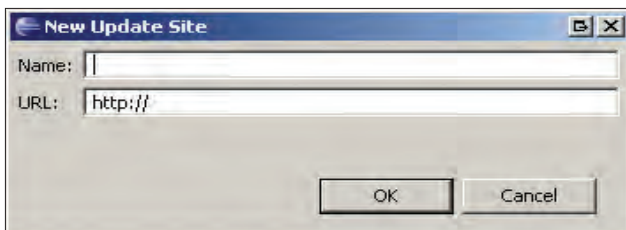
Figure 3 Add an update site
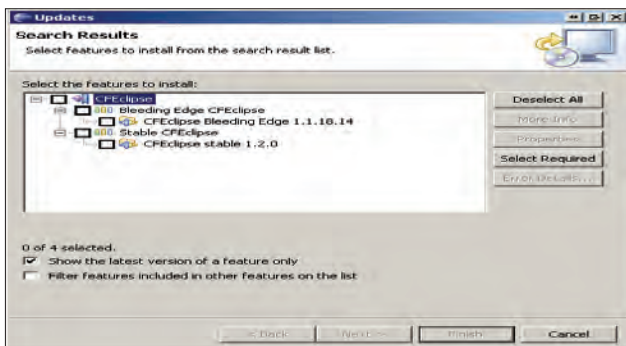
Figure 4 Update CFEclipse
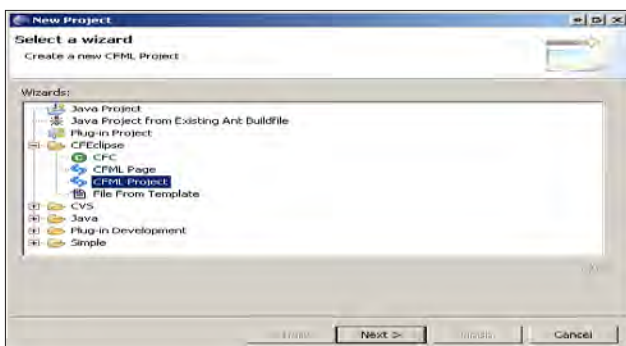
Figure 5 Creating a project in CFEclipse

Setting up a Project

Now that CFEclipse is installed, the interface doesn't look any different, does it? Nope, the interface doesn't look any different (yet). From the window menu, select Open Perspective and other. CFEclipse should be in the list of perspectives. Select it. A perspective is the default layout for your workspace.

In my Dreamweaver article, I talked about setting up a site. You'll probably want to do the same thing in CFEclipse. In CFEclipse, a project is the equivalent of a Dreamweaver site. This is how you can set it up:

1. While in the CFEclipse Perspective, select File‡ New ‡ Project. You'll get a list of all available project types as shown in Figure 5. Expand CFEclipse and then select "CFML Project." Click Next (see Figure 5).

2. This puts you in the new Project Dialog. Give the project a name and specify its directory location. Most likely you're going to be starting a new project so feel free to use the default directory, which points to the workspace directory. Click Finish. Now the project is created.

3. CFEclipse isn't as full-featured in its project settings as Dreamweaver, but there are a few that you may want to make note of. In the navigator on the left side of the screen, you should see the name of the project and a list of all files/directories that are part of the project. Right-click on the project name and select Properties. Select CFEclipse properties and you should see something like Figure 6.
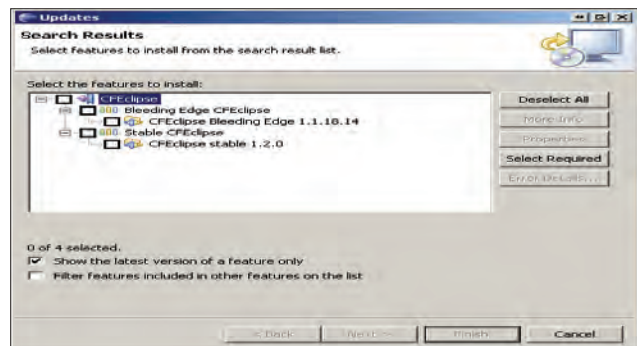
Figure 6 CFEclipse project properties
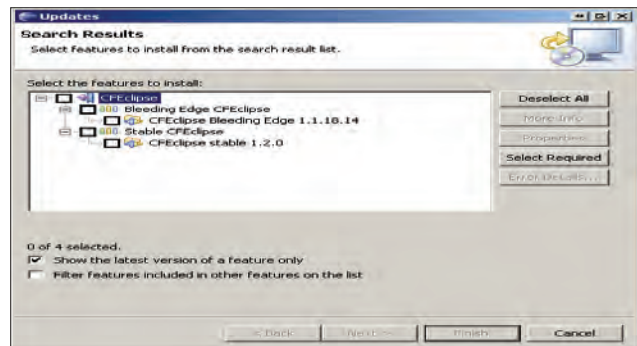
Figure 7 Component wizard

This will let you specify the project URL and the path to your snippet directory. If you're using different workspaces, but want to share snippets, you can do so here. You can also enter the docroot, which refers to the browser view, an option underneath the code view. If you want a quick browse of the files in your project, this is the way to do it without switching to a different window.

Dealing with Components

It seems that with every passing day I spend more and more time dealing with CFCs, either creating them or editing them. The CFEclipse creators didn't let me down in that regard. They have a few handy tools for creating and using components. This is how to use the wizard for creating a component:

1. From the file menu, select "New‡ Project." From the project menu (see Figure 5) select CFC. You'll get the New Component wizard in Figure 7.
2. Enter the path for the component, the component it extends, the hint, and the display name. Click Next.
3. You can add properties (a k a Instance Variables) to the component in the next window. You can specify various options including the option to generate getter and setter methods, the hint, and the type.
4. Click Next and you can add Functions to the component in this window. You can also specify each attribute in the component tag, such as access, return type, roles, hint, displayname, name, and output. Click Next.
5. In this step you can add parameters to each function you have in the component. Click Finish to create the component.

Let's examine a few of the other windows in your workspace. Open up a CFC file. (I opened the address.cfc from my December '04 article for this column). In the right half of the screen you'll see a method view, which should look something like Figure 8.

This is a nice quick outline of all the methods in your CFC. If you double-click on one of the methods, you'll jump right to its place in the component. This can be handy when jumping around long components. The Method view won't work on a regular cfm file even if you have functions inside it, however, the outline view will. The outline view gives you a collapsible hierarchical view of your files.

Miscellaneous Features

There are a bunch of things about CFEclipse that make it a joy to work with. None of these things are so amazing that they'll make you want to switch from another editor, however if you do switch I'm sure you'll find them useful.

- *Snippets:* CFEclipse has snippets. They are very similar to HomeSite-style snippets. My favorite feature here (something lacking in Dreamweaver) is that you can insert snippet variables such as the current date and time inside a snippet. You can read more details about it at the CFEclipse web site http://www.cfeclipse.org/index.cfm?objectid=C1F58502-EA69-0EC7-FF3FD40329552407.
- *Parenthesis Marking:* When dealing with nested function calls or complicated SQL queries, you'll often end up with a lot of
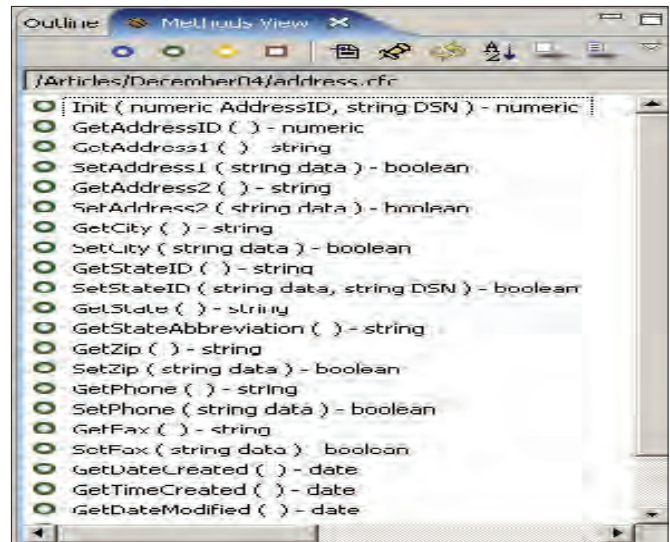
Figure 8 Method view

parenthesis. When your cursor is at one parenthesis, its foil will be highlighted so you can see the order of operations grouping. This is extremely useful.
- *Problems Tab:* CFEclipse will point out some syntax errors (such as a missing end tag) and mark them visually and in the problems tab at the bottom of the screen. I hope that future versions will advance this feature, creating a CF pseudo-compiler that will catch syntax errors.
- *Task List:* CFEclipse includes a simple task manager right in the IDE. In the middle of one thing you probably don't want to stop to fix another. So just make a quick note to come back to it.
- Search and Replace: The multiple file search and replace feature is a great way to search through all the files in your project. You can get a flat list of results like with Dreamweaver or a hierarchical list sorted like the file structure. When you open a template that has search results, CFEclipse will visually highlight those features with a bullet next to the scroll bar.
- *Hiding Panels:* By clicking the square in the top right corner of a panel, CFEclipse will expand any panel to the full size of the screen. I use this a lot to expand the code view.

Final Thoughts

CFEclipse isn't perfect yet. (Is anything ever perfect?) But it sure stands out as a quick editor that's not a resource hog. I found it does almost everything I need. And hey, it's free. It's really hard to beat free. Check it out at www.cfeclipse.org.

About the Author

Jeff Houser has been working with computers for over 20 years. He owns a DotComIt, a web consulting company, manages the CT Macromedia User Group, and routinely speaks and writes about development issues. You can find out what he's up to by checking his Blog at www.jeffryhouser.com.

jeff@instantcoldfusion.com

CFDJ Contest – Has ColdFusion MX 7 Made You a Hero at Work?

Share your success

By Simon Horwith

We're fast approaching CFUnited 06, and time for another contest to give away seats! For those of you not familiar with CFUnited, it's a conference that's grown from its grassroots beginnings to become the premier annual ColdFusion event.

As always, it will be held in the Washington, DC, area – at the Bethesda North Conference Center to be exact (yes, it's metro accessible). This year's conference runs from Wednesday, June 28, to Friday, June 30, with an optional fourth "best of" day on Saturday, July 1. If you're a ColdFusion developer, this is the one event of the year not to miss.

You must be asking yourself, "Enough, Simon. I already know how great CFUnited is…what do I have to do to win tickets?" This contest is a little different from anything I've done in the past. Rather than having a coding contest, which tends to indirectly require anyone interested in entering to come up with a lot of free time and can be intimidating for newbies, I'm having an essay-writing contest.

It's been just over a year since ColdFusion MX 7, the most feature-rich release I can think of, was released. You've all had plenty of time to play with it, deploy it at work, and build great applications. Our contest is to write about your big success story with ColdFusion MX 7. How did it make you a hero at the office? How did it enable you to make your clients happy? How did it make your existing applications more feature-rich, better performing, and/or scalable? You name it: if it's a success story thanks to ColdFusion MX 7, I want to hear it!

Essays are not going to be judged on length or grammar. Length can be anywhere from about one to five pages in Word. Don't worry about format, spelling, etc. – that's not important. I'll take submissions in Word or plain text format, though Word is definitely preferred. If you want to include screenshots or code snippets feel free, but it's not at all expected – I'm after a great story. If your company wishes to remain anonymous, that's fine – just describe them (for example, "a large health insurance company based in central PA"). All entries should be e-mailed to me at simon@horwith.com. If you have trouble sending to that address (which you shouldn't), you can try sending your entry to shorwith@aboutweb.com.

In addition to the free-entrance-to-CFUnited prize, unless the author objects, I will run the winning article in an upcoming issue of *ColdFusion Developer's Journal* for all to read. As I'm hoping and expecting to receive many great stories, it's entirely possible I will choose first and second runner-up entries as well…and also send them to the conference! This depends on the number of submissions. I hope at least to also have honorable mentions when I announce the winner. The contest deadline is May 15. I realize this may be fairly close to when many of you received your issues (one reason to also visit the coldfusion.sys-con.com site) but I don't expect anyone to spend more than an hour on their essay, unless they really feel compelled to.

Next month I will most likely run yet another contest, with a deadline right up against the conference date, for those of you still looking for a last-minute chance to attend CFUnited. Good luck to everyone – I look forward to hearing some great success stories!

Intermedia.NET...

Now serving the hottest ColdFusion.

At Intermedia.NET we go beyond the industry standard by supporting the hottest new Coldfusion software, offering power like never before. For nearly a decade, we've been providing reliable, secure hosting to thousands of companies across the globe. We can do the same for you.

Intermedia.NET's premier hosting services include:
- ColdFusion MX hosting
- Competitive plans
- Security sandboxes
- Custom tag registration
- Verity collections search engine
- Guaranteed service levels

**Unprecedented power, unmatched reputation...
Intermedia.NET is your hosting solution.**

Call us at: 1.888.379.7729
e-mail us at: **sales@intermedia.NET**
Visit us at: **www.intermedia.NET**

INTERMEDIA.NET

Creating Database-Driven Websites Using ColdFusion MX 7 on Mac OS X

This is just the beginning

By David Stutler

I t's not every day that you get the opportunity to pen an article about true love and technology. But in a few cases, true love and technology mix quite well.

I've found this to be especially true with two groups to which I belong: Apple junkies and ColdFusion developers. The first group feels like a secret club that continually wonders when the rest of the world will crash their party. The second group believes deeply in its chosen technology and consequently incorporates a fervent user community.

Until recently, it was difficult for most Apple users and ColdFusion developers to fit comfortably together. This all changed when Adobe (formerly Macromedia) released a version of ColdFusion MX 7 with official support for the Apple Mac OS X platform (www.macromedia.com/macromedia/proom/pr/2005/coldfusionmx7_updater.html).

If you want to learn how to create dynamic Websites, move ColdFusion to the top of your list. Simply put, ColdFusion is a very approachable language to learn. Users with an HTML background will love its tag-based structure; those without much programming experience will love its readability. ColdFusion developers also have reason to be excited about the availability of the language on a new platform. With ColdFusion support on Mac OS X, they can reach new markets with their application development that they could not before. That's a big advantage for the Mac Web developer crowd.

This article guides you through the steps for running ColdFusion on your Macintosh. You'll also build a couple of sample ColdFusion pages that demonstrate just how easy it is to create ColdFusion pages within Dreamweaver and run them on your ColdFusion server.

This article was originally published in the Adobe Developer Center: http://www.macromedia.com/devnet/. Reprinted with permission

Requirements

- ***ColdFusion MX 7.0.1***
  To try: www.macromedia.com/cfusion/tdrc/index.cfm?product=coldfusion&promoid=devcenter_tutorial_product_090903
  To buy: www.macromedia.com/software/coldfusion/buy/?promoid=devcenter_tutorial_coldfusion_090903

- ***Dreamweaver 8***
  To try: www.macromedia.com/cfusion/tdrc/index.cfm?product=dreamweaver&promoid=devcenter_tutorial_product_090903
  To buy: www.macromedia.com/go/devcenter_dw_buy

- ***Sample files:*** http://download.macromedia.com/pub/developer/cfonosx.zip
- ***Operating system:*** Apple Mac OS X (10.3.9 "Panther" or 10.4.x "Tiger" required): www.apple.com/macosx/
- ***Database server:*** MySQL (optional): www.mysql.org

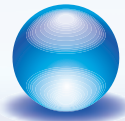## Step 1: Download the Necessary Applications and Files

To get up and running, you need a few ingredients –
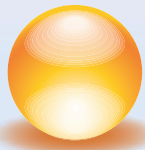

Figure 1: ColdFusion MX 7 Configuration and Settings Migration Wizard
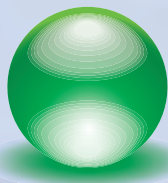
HOSTING.com

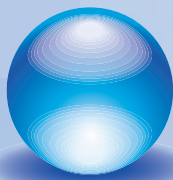THE FIRST NAME IN HOSTING

Developers!

Developers!

Developers!

We Love Them!

SERVE. SUPPORT. SECURE. STORE.

www.hosting.com
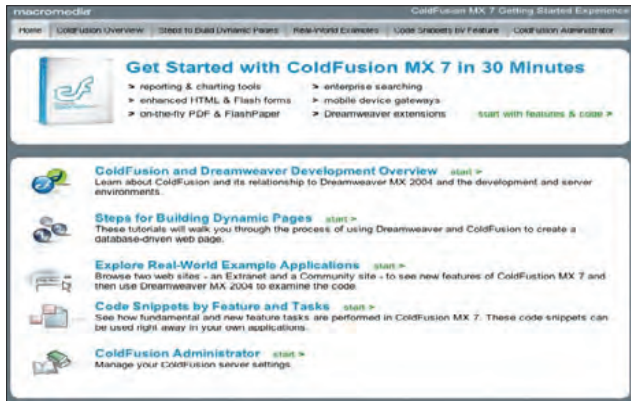
Figure 2: ColdFusion Getting Started Experience screen
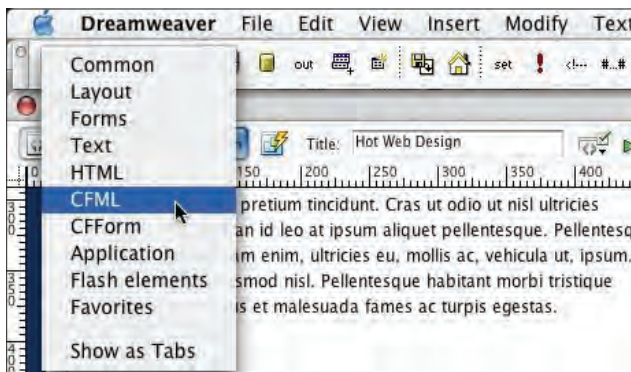
Figure 3: Selecting the CFML-related buttons on the Insert panel

primarily ColdFusion server and Dreamweaver. Luckily, both are available as free trial downloads on www.macromedia.com. The Dreamweaver trial is a 30-day full-featured evaluation copy. ColdFusion server is also a fully featured Enterprise Edition product for 30 days. After that time, the server switches to a single-user development version. You can continue to run the server locally for development without the need to purchase a commercial server license.

Optionally, you can download the MySQL database server to serve as a local data source. For this article, the MySQL server is not necessary.

Finally, download the files you'll build during the tutorial. I created them to help you learn how to use ColdFusion and speed the development process.

Step 2: Install and Set Up ColdFusion MX 7

Use the following process to install ColdFusion on your Mac:
1. Double-click the ColdFusion installer.
2. Choose your language and click OK.
3. Read the Introduction screen in the installation wizard and click the Next button to continue.
4. Accept the terms of the license agreement. Click Next.
5. Select the 30-day Trial option and click Next.
6. Select Server Configuration and click Next.
7. This tutorial assumes that this is your first installation of ColdFusion, so select No for the question and click Next.

8. The next screen shows two options. Select the one called "Getting Started Experience, Tutorials, & Documentation." Later you'll use a data source created during this step.

The second option installs a startup script that automatically launches ColdFusion server when the system starts. Select what you prefer and click Next.
9. The installer suggests a default installation folder. In most cases, this is inside the Applications folder. If you'd like a different location, enter it here. Click Next.
10. Because this is the first time you've installed ColdFusion, select No when the wizard asks if a previous version of ColdFusion has been installed. Click Next.
11. For this tutorial, select ColdFusion's built-in Web server. This allows you to access all ColdFusion services without conflicting with any of your existing Web servers. The ColdFusion server runs on port 8500. Click Next to continue.
12. On this screen, choose a password for the ColdFusion Administrator and write it down. Click Next. (The ColdFusion Administrator is a simple Web-based interface that allows you to configure things on the server, such as setting up data sources.)
13. The installer asks if you'd like to enable the Remote Development Service (RDS). By setting RDS, you enable Dreamweaver to see all your ColdFusion data sources easily from an editing interface as you build your queries. This speeds up the development process. This is very useful, so select the Enable RDS option, type an RDS password (and write it down), and click Next.
14. If you selected the option to start ColdFusion on system startup, enter your Mac user password. Click Next.
15. The next screen gives you a summary of the installation selections. To begin the installation, click Install. When the installer finishes, it notifies you that the installation was a success.
16. Leave the Launch the Configuration Wizard option selected and click Done.

Now go ahead and set up your ColdFusion installation:
1. Your default browser opens and displays the ColdFusion MX 7 Configuration and Settings Migration Wizard (see Figure 1). Enter the password you established in Step 12 above and click Login.
2. You now see the Setup Complete screen in your browser. From here you can either go to the ColdFusion MX 7 Getting Started Experience screen or the ColdFusion Administrator. You can also access the ColdFusion Administrator from the Getting Started Experience screen (see Figure 2).
3. Enter the ColdFusion Administrator. Congratulations! The administrator is a ColdFusion page, so you know that the ColdFusion server is up and running.
4. While in the Administrator, click the Data & Services link in the left navigation panel, if necessary, to expand it to show several options.
5. Click Data Sources in the expanded menu. There are several predefined data sources that the ColdFusion installation process configured for you during the server installation. You'll

use these data sources later in this article.

6. Open the sample files (cfonosx.zip) linked to in the Requirements section of this article and place them in the following location, which you must create yourself: HD:Applications: ColdFusionMX7:wwwroot:coldfusiondemo.

Step 3: Install Dreamweaver 8

If you don't have Dreamweaver 8 installed, go ahead and do that now. For your convenience, there's a download link for Dreamweaver 8 right on the ColdFusion trial download page.

The installation steps are relatively basic, with few options. Just double-click the installer, accept the license agreement, select the destination folder and drive, and click Install.

Step 4: Set Up Your First ColdFusion/ Dreamweaver Site

Setting up a site in Dreamweaver that incorporates ColdFusion pages is similar to how you set up a site for static HTML pages in Dreamweaver. The main difference lies in setting up the testing server for the site.

Follow these steps to set up a site with a testing server in Dreamweaver:

1. Launch Dreamweaver.
2. Select Site > New Site.
3. If you're not already in Advanced mode, click the Advanced

tab at the top of the Site Definition dialog box.

4. With the Local Info panel visible (select it from the top of the Category list), type ColdFusionDemo in the Site Name text box.
5. In the Local Root Folder field, browse to the folder containing the files you downloaded for this demo: HD:Applications: ColdFusionMX7:wwwroot:coldfusiondemo.
6. Select the Testing Server panel from the Category list.
7. Select ColdFusion from the Server Model pop-up menu.
8. For Access, select Local/Network.
9. The testing server folder is the folder on your hard drive where ColdFusion stores files in its Web root. For most default installations, this is as follows: HD:Applications:ColdFusionMX7:wwwroot.
10. Set the URL prefix. This is the URL that maps to your ColdFusion installation. In most cases, this URL is http://localhost:8500/ so enter that in the text box.
11. Click OK to close the dialog box.

Now that you have set up the Dreamweaver site, you are ready to begin building ColdFusion pages with Dreamweaver.

Note: For more information about defining a site and entering the proper remote settings for it so that you can upload your files to a Web server, read the following TechNote: How to Define a Site in Dreamweaver: www.macromedia.com/ go/tn_14028.

Figure 4: The cfinclude button on the Insert panel
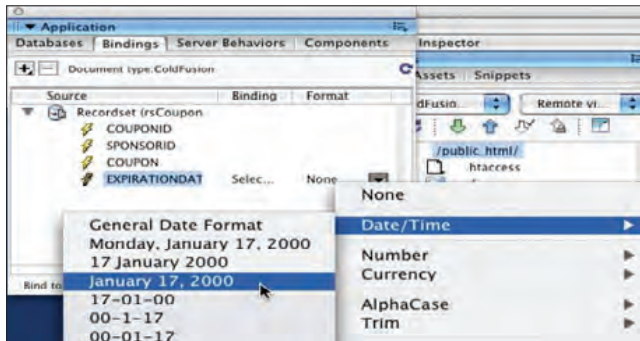
Figure 5: Choosing a date mask from the Format column pop-up menu

Step 5: Build Your First Dynamic Page

Your first dynamic page provides very basic functionality available in all application server languages – the dynamic page include. The following steps show you how to use the ColdFusion insert panel in Dreamweaver to insert a ColdFusion CFINCLUDE tag into your page:

1. In Dreamweaver, select File > Open and browse to index. cfm in the coldfusiondemo folder, where you placed all your sample files.
2. Within Design view, scroll to the bottom of the page and locate the empty area beneath the main left column. Click in the empty area.
3. On the Insert panel, click the pop-up menu and select CFML to show the CFML-related buttons on this panel (see Figure 3), which you'll need to incorporate standard ColdFusion functionality into your Web page.
4. Click the cfinclude button (see Figure 4). When you select this option, the Tag Editor dialog box appears and asks for the path for the included file.
5. Browse to footer.html inside your coldfusiondemo folder.
6. Click OK.

You've just created a simple call to include a snippet of HTML into your ColdFusion template. When you run index. html, ColdFusion will include the HTML in footer.html dynamically at runtime. Save your file and upload both index. cfm and footer.html to your testing server.

Step 6: Build Your First Database Query in a Dynamic Page

Now to build database access into your dynamic page. Over the next several steps, you'll use the Application panel in Dreamweaver to create a recordset that selects records from a database and places those records on the page:

1. Open the Application panel in Dreamweaver if you haven't

done so already. It displays a list of steps you must complete before you can do any dynamic development. Most likely, the first unchecked step is the RDS password step.

2. Click the step to provide Dreamweaver with the RDS password to connect to your ColdFusion server. (You entered this password in Step 13 during the ColdFusion installation procedure earlier.)
3. Click the Bindings tab in the Application panel. This is where you create your first recordset from the database.
4. Click the plus (+) sign in the upper left corner of the panel and select Recordset (Query) from the pop-up menu.
5. In the Recordset dialog box, name the recordset rsCoupons.
6. For Datasource, select cfbookclub_pb.
7. For Table, select BOOKCLUB.COUPONS.
8. You want to sort the coupons by date, so in the Sort section select EXPIRATIONDATE and Descending in their respective pop-up menus.
9. Click OK.

This query creates a recordset on the page that you can output using the Bindings tab in Dreamweaver. The Bindings tab displays the individual fields accessible from this recordset.
Follow these steps to output these fields on the page:

1. Highlight the text "COUPON TEXT" on the right side of the index.cfm page in Design view.
2. With that text highlighted, select the COUPON field from the list of fields in the Bindings tab in the Application panel.
3. Click the Insert button at the bottom of the Bindings tab to insert the COUPON field onto your page. This places the dynamic field directly on the page.
4. Highlight the text "Expiration Date" on the right side of the index.cfm page.
5. Select the EXPIRATIONDATE field from the list of fields in the Bindings tab.
6. Click Insert to add the dynamic field to the page.
7. With that field still selected on the page, format the data by selecting the Format column pop-up menu in the Bindings panel.
8. Format the field as a date using whichever mask you prefer (see Figure 5).

Finally, to output more than one record from this recordset, add a repeating region to the page:

1. Highlight the two new dynamic fields you just added to your page.
2. From the Server Behaviors tab in the Application panel, click the plus (+) sign and add a Repeat Region to the page.
3. A dialog box appears and asks you to specify how many records to display. Type 2 to show two records at a time.
4. Click OK.

Step 7: Test Your Page

Time to test your page:

1. Save all changes you have made to your page.
2. Upload all your files from the Dreamweaver site to your Web server.
3. View the following URL in your browser: http://localhost:8500/ColdFusionDemo/index.cfm

This page displays the included footer file and the dynamic data from the query output in the sidebar.

Congratulations! You've just created your first database-driven page using ColdFusion on a Mac.

Where to Go from Here

As you can see, developing and testing ColdFusion applications on a Mac is easy. ColdFusion is a robust application server; what you've accomplished is just the beginning.

With a little digging around in the Application panel in Dreamweaver, you can explore all that is possible in a point-and-click environment until you become more comfortable with the ColdFusion language. The functionality ranges from user authentication to inserting and updating data in a database, all through a Web page that requires little-to-no hand coding. What a great way to learn a language!

If you are an existing Dreamweaver user, jump in and try ColdFusion and learn how easy it can be to create database-driven, dynamic Websites. Start out by visiting the Getting Started page (www.macromedia.com/devnet/coldfusion/getting_started.html) in the ColdFusion Developer Center, and then see where your creativity takes you.

About the Author

David Stutler is the owner and operator of WideEyed Media (www.wideeyedmedia.net), a new media development group located in Kansas City. In addition to his production duties, David serves as the lead Flash and Dreamweaver instructor for ThinqDigital, a Macromedia Authorized Training Partner. David is also an avid musician, spending the last 11 years in the vocal group measureXmeasure.

david@wideeyedmedia.net

Get in the Right Group

The group attribute of the cfquery tag is a powerful tool for not only organizing your data but for styling and presenting it

By James Edmunds

In a previous article for *ColdFusion Developers Journal* I described a ColdFusion component (CFC) that I wrote to report on what search terms visitors used on a site's search facility when visiting the Web sites I managed (February 11, 2005 – http://cfdj.sys-con.com/read/48234.htm).

My work is largely comprised of moderate-traffic sites hosted on shared servers and, while most hosting companies offer fairly robust statistical reporting, including well-articulated reports on the referring keyword terms used to enter a site from a search engine, I wanted a report that would show in a simple and easily readable way what terms had been entered in the search facilities (usually Verity-powered) that I had created for use within the site.

One method of the component recorded each search term used when it was used in a data table, and other methods called and displayed a time-sorted count of the most frequently used search terms.

That turned out to be so handy for me and my clients that I crafted another similar CFC that reports on which data records are being called into a detail page so the client and I can see, for in-stance, what performing arts events are most looked at or e-mailed about.

Since the same collection of records can be called from different calling pages and into different presentations of the details – for instance, one page of event listings can call a detailed page regarding one event, or a detail page itself can call a page to "Email a friend" – this reporting component needed to have a field in its data table for the calling page as well as the record called.

Displaying that information is a classic use of the "group" attribute of ColdFusion's <cfoutput> tag, which specifies the query column to be used when retrieving a record set that's ordered on a query column.

For our calling page CFC, the reporting query looks like this:

```
<cfquery name="recordcount" datasource="#thedatasource#">
SELECT thecallingscript, thecalledrecordtitle, COUNT(thecalledrecordtitle)
AS TheCount
FROM tbl_calledrecords
WHERE thecalledtime BETWEEN #CreateODBCDate(arguments.thestartdate)# AND
#CreateODBCDate(DateAdd('d',1,arguments.theenddate))#
GROUP BY thecallingscript, thecalledrecordtitle
ORDER BY thecallingscript, TheCount DESC, thecalledrecordtitle
</cfquery>
```

The code for the <cfoutput> tag looks like this:

```
<cfoutput query="recordcount" group="thecallingscript">
    Here were the records called to the page        <strong>#thecallingsc
```

```
ript#</strong>:<br>
    <table width=600 cellpadding=2 cellspacing=0 border=1>
    <cfoutput>
            <cfif TheCount gte arguments.minRecords>
                    <tr>
                    <td style="font-size:11px;" valign="top" align="left"
width=80>#TheCount#</td><td  style="font-size:11px;"" valign="top" align="l
eft">#thecalledrecordtitle#</td>
                    </tr>
            </cfif>
    </cfoutput>
    </table><br>
<br>
</cfoutput>
```

And that results in a display that looks like Table 1.

Note the difference between <cfoutput>'s "group" attribute and "GROUP BY" in the SQL statement. GROUP BY is used with aggregate functions to sort rows into the specified groups before applying an aggregate function to each group.)

What <cfoutput>'s group attribute does is isolate the query column that identifies a group defined by the query's ORDER BY clause, whereupon you can, with an additional <cfoutput> tag nested inside, output the individual rows that have matches to that specified query column. In our example, we output the variable #thecallingscript# and use it as a title for an output of each set of rows creating a match for the different instances of #thecalling-script#.

This use of the "group" attribute can be nested at as many levels as you need. You can have a query that orders rows by several fields, like this:

```
<cfquery name="thequery" datasource="datasource">
SELECT maincategory, subcategory, subsubcategory, description, theurl
FROM thetable
ORDER BY maincategory, subcategory, subsubcategory, description
</cfquery>
```

And you can output it by nesting at each group level like Table 1:

```
<cfoutput query="thequery" group="maincategory">
<ul>
    <li>#maincategory#</li>
    <cfoutput group="subcategory">
    <ul>
            <li>#subcategory#</li>
            <cfoutput group="subsubcategory">
            <ul>
                    <li>#subsubcategory#</li>
                    <cfoutput>
                    <ul>
                            <li><a href="theurl">#description#</a></li>
                    </ul>
                    </cfouput>
            </ul>
            </cfoutput>
    </ul>
    </cfoutput>
```

```
        </cfoutput>
        </ul>
</ul>
</cfoutput>
```

Note that the source query is identified only in the outermost top-level <cfoutput> tag, and that it's not necessary to name the "group" at the innermost level. We'll see in the next example that it's also possible (and very useful) to have more than one <cfout-put> at a given level of the nesting.

What You See Is What You Output, Made to Look the Way You'd Like It To

Because this means of grouping data comes on the output side of the process, the means of styling the display can be made an integral part of the grouping, such as our simple use of unordered list tags in the example above, and is independent of any of the query logic except the ORDER BY clause – complex queries that join two or more tables use aggregate functions. Filters by various parameters are ready for use in the "group" attribute of cfquery.

What's more the various nested levels of grouped output can each have their own display treatment, allowing for some handy integration with CSS or other display code.

A case in point came up recently when one of my clients, a very successful biomedical firm that operates plasma donation centers in a couple of dozen U.S. states, asked for a map application on its Web site that would show its donation centers by state, with detail pages as the site visitor drilled down to specific states and cities.

There are countless examples on this kind of display on various sites around the Web, a large number of which employ HTML <map> and <area> tags to define coordinates of spots on a map that can be linked. For users of an application server like ColdFusion, however, this can be made into a dynamic display and the "group" attribute of <cfquery> plays a central role in passing information from the database into CSS that makes the display work.

Here were the records called to the page **Email This:**		
9	Dora's Pirate Adventure	*Special Event*
8	Die Fledermaus	*2005-2006 Series Event*
4	Diavolo	*2005-2006 Series Event*
3	Break!	*Special Event*
2	An Evening of Classic Lily Tomlin	*Special Event*

Here were the records called to the page **Event:**	
1025	Dora's Pirate Adventure
591	Diavolo
524	Die Fledermaus
522	Break!
504	Pucho and the Latin Soul Brothers

Table 1 Output is grouped for display

state_id	state_	state_left	state_top
43	AK	44	275
41	AL	374	233
33	AR	310	213
13	AZ	106	200
8	CA	35	173

Table 2 Table of coordinates for map display

For the biomedical firm, I started by creating a graphical map of the U.S. and then measured coordinates for the spot on each state where a text abbreviation would sit; these coordinates are stored in a simple table called "tbl_states":

The "tbl_states" table gives me a set of map coordinates to join with my ongoing database of plasma center locations for this query that will get the information needed for the display:

```
<cfquery name="getmapinfo" datasource="#thedatasource#">
SELECT distinct c.state,  c.city, c.status, c.state_full, s.state_ab,
s.state_top, s.state_left
FROM tbl_states s
LEFT OUTER JOIN biocenters c
ON s.state_ab = c.state
ORDER BY state_ab, status, city
 </cfquery>
```

In Table 3 note that we get a "state_ab" result for every state, as well as the coordinates needed on our map, but for those states that are only found in "tbl_states" and not in "biocenters" we get a row with no data in the "city" field, a condition that we'll use to trigger a different result than those that do have cities defined.

Our next group level defined by the ORDER BY of the query, and called by the "group" attribute, is "status," and, finally, "city," which we'll call twice, once nested inside of each of the two possible values for "status."

You can see how this information is used by following the program listing for "MainMap.CFM" abridged from the actual site to include only the code we need to illustrate the process used to populate the map display (see Listing 1).

The first item of business once we've called the query is to offset our map coordinates to account for the spot on the page where the containing div will display it. Then we begin a rather detailed grouped output that's used to set styles for the items that will display on the raw map.

Because we used a left outer join in our "getmapinfo" query, at the level of the group="state_ab" the parameter #state_ab# will be a list of the 51 states (including the District of Columbia) whether or not there are join matches. We'll evaluate each of these rows to see whether there's a city defined (because we asked for "SELECT distinct," those cities that have more than one

match will be listed only once). The rows with no city defined (meaning there are no plasma centers in that state) are used to create one kind of div, "#state_ab#_set," and those that do have at least one city defined have created for them some additional offset information, the div "#state_ab#abbrev" and the additional "div #state_ab#detail."

That first <cfoutput>, in the <style> section of the <head> block, used only the topmost group of the query, since all we needed to do was list the states and evaluate if, based on whether they had data in their #city# fields, they needed merely the div that will write the abbreviation in place on the map; or, if they did have centers present, the richer pair of divs that we'll use to provide rollover display and links.

Digging in at All Three Levels

In the body of the page, however, we use a <cfoutput> that employs the "group" attribute at all three of the levels offered by the query's "ORDER BY" clause.

Once again, the <cfoutput> begins by being grouped on "state_ab" and we evaluate whether a given #state_ab# has city data associated with it, our tip-off that there's at least one plasma center in that state. For those with no city, we write the div that

Figure 1 The "group" attribute of <cfquery> provides the means to distribute display information about the various biocenters onto a map that packs detailed information and further links into a compact space.

	CITY	STATE	STATE_AB	STATE_FULL	STATE_LEFT	STATE_TOP	STATUS
1	Anchorage	AK	AK	Alaska	44	275	Soon
2	Montgomery	AL	AL	Alabama	374	233	Open
3	Phenix City	AL	AL	Alabama	374	233	Open
4	Tuscaloosa	AL	AL	Alabama	374	233	Open
5	Huntsville	AL	AL	Alabama	374	233	Soon
6	Mobile	AL	AL	Alabama	374	233	Soon
7	Little Rock	AR	AR	Arkansas	310	213	Soon
8	Glendale	AZ	AZ	Arizona	106	200	Open
9	Calexico	CA	CA	California	35	173	Soon
10	Colorado Springs	CO	CO	Colorado	176	154	Open
11	Pueblo	CO	CO	Colorado	176	154	Open
12	[empty string]	[empty string]	CT	[empty string]	531	170	[empty string]
13	[empty string]	[empty string]	DC	[empty string]	531	230	[empty string]
14	[empty string]	[empty string]	DE	[empty string]	531	200	[empty string]
15	Jacksonville	FL	FL	Florida	442	290	Open
16	Daytona Beach	FL	FL	Florida	442	290	Soon

Table 3 <cfdump> of the query "getmapinfo" shows the rows returned

Table 4 By being selective in using the "group" attribute, we can have newspaper columns and keep the data grouped in a way that makes an easily readable display

writes the state abbreviation in white onto the map and we're done with that state.

But for those states with city data, we write a link that will pass the #state_ab# parameter to a page that will show the detailed information, and then after we pull the #state_full# state name to head the rollover div, we have a nested <cfoutput> that is grouped on "status," so we can differentiate between centers that are already open and those in the preparation stages. Here's a case in which we use the next level of the <cfoutput> grouping twice, once for each of our two possible values of "status."

The result is a map that highlights those states that have biocenters in them, which provides those states with a link to a detail page for drilling down and summarizes the centers in the state grouped by open status in a rollover div that shows when the mouse hovers on the link.

Grouped Newspaper Columns

I'll pass along one more example of the use of the "group" attribute that I find particularly useful. There are times that I like to display query results in newspaper-like columns. Because this grouping of the data all happens on the output side, we can operate the logic that determines how a column will be distributed at whichever levels of the nested <cfoutputs> that helps us make an attractive display. In the case of listing the states and cities with biocenters in them, I wanted the cities to stay grouped with their states. First I set up the query "GetSoonAndOpenCenters" with the ORDER BY state_full, city clause, a sub-query that counts the distinct instances of state_full, and output it as shown in Listing 2.

Note that rather than use the query's rowcount as the trigger for jumping to a new column with the <td> tag, we set up a incrementer based on the "state_full" count and put the evaluation trigger for the column change at the end of each iteration of that outer <cfoutput> group. The result is a reasonably evenly distributed three-column display that keeps the cities grouped in their state.

In all of these examples, the utility of the "group" attribute rests with the fact that it resides on the output side of our data handling, giving us multiple ways of distributing and displaying the results of a given query. This includes adding styling, conditional branches in the various nested output levels, and the association of our choice of styling with each sublevel of the output. Sometimes it reminds me of my dear mom, who used to stress to me in my school years that getting in with the right group was the key to success!.

About the Author

James Edmunds is a freelance Internet developer and arts administration consultant living in New Iberia, Louisiana. After a career in journalism that included writing for national publications such as Newsweek and serving as editor for an alternative weekly newspaper he founded in southern Louisiana, James began to pursue a second career working with arts groups. Though he had no technology background, his interest in harnessing the power of the Internet to serve the interests of the arts led him into Internet development, an arena in which he has now gone beyond the arts to serve a general business clientele.

edmundsj@bellsouth.net

Listing 1

```
<cfprocessingdirective suppresswhitespace="yes">

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<script src="../scripts/common.js" type=text/javascript></script>

    <!--The query will find which states have centers, and group the
cities in which centers are located.

We use distinct to get only one city per state for the listing
purposes. The ORDER BY will be called by the GROUP parameter of the
cfquery call. LEFT OUTER JOIN will give us all state abbreviations.
---->

    <cfquery name="getmapinfo" datasource="#request.dsn#"
username="#request.dsnusername#" password="#request.dsnpassword#">
SELECT distinct c.state, c.city, c.status, c.state_full, s.state_ab,
s.state_top, s.state_left

    FROM tbl_states s

    LEFT OUTER JOIN biocenters c

    ON s.state_ab = c.state

    ORDER BY state_ab, status, city

    </cfquery>


    <!--- Have one place to make offsets ----->

    <cfset thetopoffset=146>

    <cfset theleftoffset=170>

    <style>
<cfoutput query="getmapinfo" group="state_ab">


<!---apply the offsets----->

<cfset variables.state_top = #val(getmapinfo.state_top)# + #vari-
ables.thetopoffset#>

<cfset variables.state_left = #val(getmapinfo.state_left)# + #vari-
ables.theleftoffset#>

<!----for states with no centers, create div that will put state ab-
breviation on map in white type ----->

<cfif NOT #len(getmapinfo.city)#>
###state_ab#_set {

  position: absolute;

  left: #variables.state_left#px;

  top: #variables.state_top#px;

  visibility:visible;

  Z-INDEX: 5;

  border:0px;

  padding:0px;

  background-color:##dedede;

  width:10px;

  font-size:10px;

  font-weight:bold;

  color:##fff;

}

</cfif>


<!----Set a position for the rollover which is to be called, for
those states that have centers.  This will change so that items on
the left will pop right and items on the right will pop left---->
```

Strip Tease

Formatting complex user input to prevent cross-site scripting

By Isaac Dealey

I don't like browser-based WYSI-WYG editors. There are a reasonably large number of them and several of the recent versions are even cross-browser-compatible with Mozilla and even some less popular browsers (although Safari continues to be problematic).

The technical issues surrounding the implementation of WYSIWYG editors aren't the reason I don't like them, in spite of continuing problems with their interfaces such as lack of support for tab indexes. I don't like WYSIWYG editors because users can't be trusted not to put lime green, blinking text on a hot pink background. It's inevitable, however, that if we want to keep our clients happy we have to adopt WYSIWYG editors in our applications. The genie is out of the bottle and there's no putting it back, and thanks to our friendly neighborhood spammers, there's something else we can't trust our users to do: provide HTML content without defiling our sites with pop-up ads and other malicious JavaScript.

There's a name for malicious code injected by a user into a web application: it's called a Cross-Site Scripting attack (CSS or XSS to avoid confusion with Cascading Style Sheets). There are a number of ways to deal with XSS attacks, most notably the ColdFusion native HTMLEditFormat() function that everyone should use (although almost no one does) to create form input elements (when CFINPUT isn't used). Although we still need this function and should continue to use it for forms and some other displayed text, it doesn't address the problem of dealing with input provided by WYSIWYG editors. The HTMLEditFormat() function – while crucial – doesn't allow the user to provide HTML content because it converts any characters provided in the input element into properly escaped HTML entities and so when this content is displayed the user will see HTML code instead of rendered HTML.

The problem of users providing HTML content requires a different solution: remove potentially malicious tags from the input. The process of removing these tags should always occur when the user provides the content (not when it's displayed later) to prevent the process from needlessly creating a performance bottleneck on your server.

Several years ago I contributed to the Common Function Library Project what appears to have become the de facto standard for removing potentially malicious tags from user-provided input. Every few months someone on the cf-talk mailing list asks how to remove malicious tags and inevitably, someone else on the list suggests the function I wrote. It's called StripTags() and can be found at http://www.cflib.org/udf.cfm?ID=774.

While I'm glad that ColdFusion developers have enjoyed this function and found it useful in developing their applications, this article isn't about that function. I've actually stopped using it myself in favor of what I believe is a more robust solution.

What's my secret? Extensible Style Language Transformations (XSLT) while more challenging to implement than a UDF provides a way to accomplish the same end results as the StripTags() function and provides greater control over the final result.

Why is this? One reason is that the StripTags() function uses Regular Expressions which while powerful are limited to a more or less one-dimensional view of the content, with

no built-in understanding of attributes or the parent/child relationships of nodes.

Another reason is that the arguments of the StripTags() function are simple and their simplicity (a strength with regard to ease of use and time-to-market) limit the amount of control they can provide.

XSLT for XSS and Other X-cronyms

What does using XSLT to strip malicious tags entail? First, users who provide HTML content must provide their content as properly formatted XHTML, which is slightly different from traditional HTML in ways that will frustrate most non-programming users who have been forced to become comfortable with HTML. Fortunately, if you use any of the recent WYSIWYG editors there's a good chance it will ensure proper XHTML formatting, making the whole process invisible to the user.

Second, although the ColdFusion tags and functions for handling XML have allowed us to be lackadaisical about case-sensitivity, the standard for XML is that it is case-sensitive. As a result, your XSL template will need to address the difference between a "script" tag and a "SCRIPT" tag and a "ScRiPt" tag. Frustratingly the XSLT engine that drives the ColdFusion server's native XMLTransform() function doesn't support the lower-case() function yet. I'll show an only moderately annoying workaround for this later in the article.

Lastly, there's all that learning a new language (XSL) and dealing with all the little things in the new language that seem strange or difficult to grasp, which are often not well documented or that you may not realize at first are necessary, in some cases even after testing.

If after reading the last section you're interested in moving your malicious tag removal to an XSL solution, I'll describe some of the specifics. Listing 1 is a sample XSL template that can be used with the ColdFusion native XMLTransform() function to remove malicious tags once we know the content string is valid XHTML. At the top, the doctype declaration contains several entities to ensure that the XML transformation won't throw an error if the user provides a common entity such as   which is declared in the HTML specification but not in the specification for generic XML.

Following the doctype declaration is the document root element xsl:stylesheet, with a namespace (xmlns:xsl) that tells the XML processor where to find information about "xsl:" tags. Inside the xsl:stylesheet element it's necessary to declare the output method with the xsl:output tag – this tells the XML processor what the XML will be transformed into, in this case more XML.

Following the output are several xsl:variable tags that create variables that can be used throughout the document by adding the $ symbol to the beginning of the variable name, i.e., the "ucase" variable is referenced as $ucase. The first two of these variables are used to help us make the document case-insensitive so a user can't trick the filter by changing the case of the tags. The last variable specifies a list of HTML tags that will be removed when the transformation is applied. Setting the variable here, however, doesn't remove the tags – that's done in the fourth xsl:template tag from the top.

The first xsl:template tag matching "text()" copies any text nodes the user has provided. This is the actual content – everything else is formatting, and so this is the most important element to retain. Thus, if you wanted to remove all tags, you could do so by removing all the xsl:template tags in this XSL document except for the xsl:template tag matching "text()".

The second xsl:template tag matching "comment()" lets the user to put HTML comments in their content. Since HTML comments aren't dangerous, they can be allowed even though a user isn't likely to provide them.

The third XSL template tag matching "*" copies all HTML tags into the resulting output. You may be thinking to yourself, "I thought this was supposed to remove tags" – and you're right, but to remove only the desired set of tags it's necessary to copy all the tags first.

Besides copying these tags the xsl:template tag matching "*" also removes potentially malicious attributes including any attribute beginning with "on" such as "onmouseover" or "onclick" and any attribute containing the string "javascript:" at its beginning such as href="javascript:alert('I\'m an XSS attack');".

Subsequent xsl:template tags then remove any tags they match from the resulting XHTML content by matching the tag and omitting the xsl:copy tag. For some (script, style, iframe) the tag and all children of the tag are removed, and these xsl:template tags are ended with the / at the end of the opening tag. For others such as html and body, which are removed because it's assumed this content will appear in a formed HTML document, child tags are retained by including the xsl:apply-templates tag as a child of the xsl:template tag.

Check Yourself Before You Wreck Yourself

Unfortunately, it's not enough to simply slap an XSL transformation on the incoming content. This is partly because doing so can result in an error if the content isn't well-formed XML though it's also partly because the input element may not be required. Mostly, however, the content must be pre-formatted to eliminate the need for the user to enter all the content in a document root element. Users don't know what a document root element is, nor will they ever care, they just want to make the text bold, so you can't require them to add a div tag to any content they provide, it's just going to frustrate them and reflect badly on you.

Listing 2 is a tidy little CFC containing all the methods you'll need to handle testing and formatting user-provided HTML content. The parseXHTML() method converts user-provided content into an XML document to which the XSL Transformation can be applied. This method ensures compatibility with ColdFusion version 6 or 6.1 (ColdFusion 7 doesn't require that an XML document be parsed before it's transformed) and uses the XHTML doctype to ensure that HTML entities such as   won't produce an error. The method is private because it will produce an error if the content provided by the user isn't valid XHTML. The job of the following two methods is to remove malicious tags and test the validity of the XHTML content string by attempting to execute the parseXHTML() method against the content string internally.

To put this in focus, a ColdFusion page receiving content from a WYSIWYG editor would use the following code to create

this CFC:

```
<cffile action="read" variable="xslFilter" file="#expandpath('inputFilter.
xsl')#">
<cfset Guardian = CreateObject("component","HtmlGuardian").
init(xslFilter)>
```

Once the guardian object is created, it can be used in one of two ways. The first method is to simply attempt to filter the content provided by the user and if an error occurs, remove all the content entirely. This is done with the following code:

```
<cfset form.content = Guardian.filterXHTMLInput(form.content)>
```

If the system requires the provided content, it will be necessary to test the content for validity to provide users with a validation message indicating that their content isn't valid and can't be saved. This can be done either by checking the length of the string after attempting to filter it, or by checking the length of the string, then testing its validity as below.

```
<cfif not len(trim(form.content)) or not Guardian.inputIsXHTML(form.
content)>
```

```
<!--- invalid content provided --->
</cfif>
```

The Denouement

Although I've focused on the task of removing malicious tags in this article, it's also possible (and certainly more secure) to focus instead on allowing only a specific subset of known tags to allow, say, bold and italic tags. Although I haven't covered this subject specifically the samples provided should give you a good starting point from which this technique can be streamlined to suit your specific application needs.

About the Author

Isaac Dealey has worked with ColdFusion since 1997 (version 3) for clients from small businesses to large enterprises, including MCI and AT&T Wireless. He evangelizes ColdFusion as a volunteer member of Team Macromedia, is working toward becoming a technical
instructor, and is available for speaking engagements.

info@turnkey.to

LISTING 1

```
<!DOCTYPE xsl:stylesheet  [
  <!ENTITY apos   "&#39;">
  <!ENTITY nbsp   " ">
  <!ENTITY copy   "&#169;">
  <!ENTITY reg    "&#174;">
  <!ENTITY trade  "&#8482;">
  <!ENTITY mdash  "&#8212;">
  <!ENTITY ldquo  "&#8220;">
  <!ENTITY rdquo  "&#8221;">
  <!ENTITY pound  "&#163;">
  <!ENTITY yen    "&#165;">
  <!ENTITY euro   "&#8364;">
]>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/
Transform">
  <xsl:output method="xml" indent="no" omit-xml-declaration="yes" />

  <xsl:variable name="lcase" select="'abcdefghijklmnopqrstuvwxyz'" />
  <xsl:variable name="ucase" select="'ABCDEFGHIJKLMNOPQRSTUVWXYZ'" />
  <xsl:variable name="tags"  select="',head,meta,script,frame,frameset,
iframe,style,'" />

  <xsl:template match="text()">
    <xsl:value-of select="." disable-output-escaping="no" />
```

```
  </xsl:template>

  <xsl:template match="comment()">
    <xsl:comment><xsl:value-of select="." /></xsl:comment>
  </xsl:template>

  <xsl:template match="*">
    <xsl:copy>
      <xsl:copy-of select="@*[
        starts-with(translate(local-name(),$ucase,$lcase),'on')=false()
        and starts-with(translate(normalize-space(.),$ucase,$lcase),'ja
vascript:')=false()]" />
      <xsl:apply-templates />
    </xsl:copy>
  </xsl:template>

  <xsl:template match="*[contains($tags,concat(',',translate(local-
name(),$ucase,$lcase),','))=true()]" />

  <xsl:template match="*[contains(',html,body,',concat(',',translate(lo
cal-name(),$ucase,$lcase),','))]">
    <xsl:apply-templates select="./node()" />
  </xsl:template>

  <xsl:template match="*[starts-with(translate(local-name(),$ucase,$lca
se),'cf')=true()]">
```

```
      <xsl:apply-templates select="./node()" />
    </xsl:template>
</xsl:stylesheet>
```

LISTING 2

```
<cfcomponent displayname="HtmlGuardian">
  <cfset variables.filter = "">

  <cffunction name="init" access="public" output="false">
    <cfargument name="filter" type="string" required="true" hint="XSL">
    <cfset variables.filter = arguments.filter>
    <cfreturn this>
  </cffunction>

  <cffunction name="parseXHTML" access="private" output="false"
  hint="converts user provided XHTML content into an XML document
object">
    <cfargument name="input" type="string" required="true">
    <cfxml variable="arguments.xml" casesensitive="false">
      <cfoutput>
        <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
        "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
        <html>
        <body>
        #REReplaceNoCase(trim(arguments.input),"<\?xml[^>]*\
?>","","ONE")#
        </body>
        </html>
      </cfoutput>
    </cfxml>
    <cfreturn arguments.xml>
  </cffunction>

  <cffunction name="inputIsXHTML" access="public"
  output="false" returntype="boolean"
  hint="tests a user provided XHTML content string for validness">
    <cfargument name="input" type="string" required="true">
    <cfif not len(trim(input))><cfreturn true></cfif>
    <cftry>
      <cfset parseXHTML(input)><cfreturn true>
      <cfcatch><cfreturn false></cfcatch>
    </cftry>
  </cffunction>

  <cffunction name="filterXHTMLInput" access="public"
  output="false" returntype="string"
  hint="removes potentially malicious tags from user provided XHTML
content">
    <cfargument name="input" type="string" required="true">

    <cfif not len(trim(variables.filter)) or not len(trim(input))>
      <cfreturn input>
    </cfif>

    <cftry>
      <cfset arguments.input = parseXHTML(input)>
      <cfreturn XMLTransform(arguments.input,trim(variables.filter))>
      <cfcatch type="any">
        <cfreturn "">
      </cfcatch>
    </cftry>
  </cffunction>
</cfcomponent>
.
```

Flash Forms to the Rescue

The checkbox issue

By Sami Hoda

Six months ago I was anxious to get the chance to upgrade our Web servers to Adobe ColdFusion MX 7 Enterprise Edition. During a product re-design, I realized how advantageous it would be for our customers to start using the new Flash Forms feature versus the standard HTML tables they had been using for years. The sorting and scrolling capabilities alone would be a hit.

Initially I thought that the Flash Forms would function like other HTML forms, but, as it turned out, there were some definite quirks to be dealt with. Fortunately I was able to resolve all the issues except one, which kept me stumped. I read up as much as possible online, using sites such as ASFusion.

The Problem

Our Web application had tables of data throughout the site. The first column of each table was always a checkbox so the user could check off whatever record he wanted to modify. The checkbox contained the primary key value for each of the records.

The issue with moving to Flash Forms was that there was no apparent checkbox-type functionality built-in. What needed to be done, to ASFusion's credit, was to introduce a new Bit column into the database table and show the column in the Flash Form as editable.

Here is some sample code that works well with Microsoft SQL Server.

Before:
SELECT *
FROM tblCustomers

```
<cfgridcolumn name='custID' display='no' />
<cfgridcolumn name='customerCode' header='Code' select='no' width='65' />
```

```
<cfgridcolumn name='username' header='Username' select='no' width='65' />
```

After:
SELECT *, checked="
FROM tblCustomers

```
<cfgridcolumn name='checked' header="" type="boolean" width='25' />
<cfgridcolumn name='custID' display='no' />
<cfgridcolumn name='customerCode' header='Code' select='no' width='65' />
<cfgridcolumn name='username' header='Username' select='no' width='65' />
```

As you can see, cust ID is the hidden primary key column, and I added a cfgrid column called "checked" to reflect the new database column. By default none of the columns are checked since the SQL value for each column is empty. You'll also notice that none of the columns are selectable except the "checked" cfgrid column. That means the column can be checked on and off by the user just like the checkboxes before.

And I thought I was done.

The Problem (Continued)

What I found was that I could easily check off the record I wanted and get the custID to show on the other end. However, there was a problem. In HTML you can check and uncheck checkboxes and it returns a value only when the item is checked. In Flash Forms, when the item is essentially "touched," it becomes on. In other words, I would check a box, uncheck it, and submit the form, and it would show the custID of the one I checked and unchecked because Flash Forms thinks this record was "edited". This was going to be a problem. How was I to determine which item was "truly checked" versus which record was just "touched"?

The Solution

After checking

around, I found myself alone in trying to find a solution. In essence what I needed was a piece of code that would parse through the submitted form fields and return the Primary Keys necessary for me to go about my work.

Here's a breakdown of the custom tag I developed to help resolve this issue. *Note:* This tag is in beta and will undergo some additional cleanup and should be posted on my blog soon.

The Custom Tag <!---

Problem: Flash Form Grids with CheckBoxes, when submitted, return all items that were checked AND unchecked, i.e. Edited.

Solution: getIDSelectedFromFlashFormGrid.cfm, with the user-specified hidden ID column name that's passed over via Flash Forms grid, cleans up the returned form scope, returning only IDs of the items that were truly selected.

Note: In Progress, works as is, Copyright Sami Hoda (c) 2005-06

Usage: <cf_getIDSelectedFromFlashFormGrid ID="**Primary Key Column Name **">

Output: variable listSelected, list of Primary Keys selected --->

```
<CFSETTING ENABLECFOUTPUTONLY="YES">

<CFPARAM NAME="Attributes.ID" default="" type="string">

<!--- Checks to see how many checkboxes were 'touched' assuming check-
boxes were named checked --->
<cfset numEdited = ArrayLen(GRID.checked)>

<!--- Clean form values by getting only values for 'GRID.*' and make a
query of those values without the 'GRID.' prefix --->
<cfset newQ = queryNew("name,value")>

<cfloop collection="#form#" item="i" >

    <cfif left(i,5) EQ "GRID." and left(i,13) NEQ "GRID.ORIGINAL" AND
left(i,14) NEQ "GRID.ROWSTATUS">
            <cfset temp = queryAddRow(newQ)>
            <cfset wOutGrid = right(i, len(i) - 5)>
            <cfset temp = QuerySetCell(newQ, "name", "#wOutGrid#")>
            <cfset temp = QuerySetCell(newQ, "value", "#Evaluate(i)#")>
    </cfif>

</cfloop>

<!--- Loops through the array values dependent on the number 'touched'
and create a new proper query with simple values --->
<cfif numEdited GT 0>
    <cfset newQQ = queryNew("ID")>
    <cfset temp = queryAddRow(newQQ,numEdited)>
```

```
    <cfloop index="z" from="1" to="#numEdited#" >
            <cfset temp = QuerySetCell(newQQ, "ID", z,z)>
            <cfset count = 1>
            <cfloop query="newQ">

                    <cftry><cfset temp = queryAddColumn(newQQ, #newQ.
name#, arrayNew(1))><cfcatch type="any"></cfcatch></cftry>
                    <cfset foo = newQ.value[count]>
                    <cfset fooboo = foo[z]>
                    <cfset temp = QuerySetCell(newQQ,newQ.name,
fooboo,z)>
                    <cfset count = count + 1>
        </cfloop>
    </cfloop>

</cfif>

<!--- Now, from the clean query, get only those who were truly selected,
not just 'touched' --->
<cfquery dbtype="query" name="getSelected">
    SELECT *
    FROM newQQ
    WHERE checked != 'false'
</cfquery>

<cfset myList = "">
<cfoutput query="getSelected">
    <cfset myList = ListAppend(myList,#evaluate(Attributes.ID)#)>
</cfoutput>
<cfset caller.listSelected = myList>

<CFSETTING ENABLECFOUTPUTONLY="NO">
```

So, on my target page, I can write code like this:

```
<cfset numEdited = ArrayLen(GRID.checked)>
<cfif numEdited EQ 0>
<cflocation url="#errorURL#">
</cfif>

<cf_getIDSelectedFromFlashFormGrid ID="custID">
```

And now a variable called "listSelected" gives me a clean list of custIDs to use.

Some of the changes I have planned for the Custom Tag include:

- Naming the return variable
- Renaming variables for consistency
- Enhanced comments inside the code
- Enhanced error checking
- Speed enhancements

So I hope this shows how you can effectively use checkboxes inside Flash Form grids now.

samihoda@gmail.com

Enter the Adobe Flex Developer Derby

Show off your killer apps

By Matt Chotin

We're very excited about the upcoming release of Adobe Flex 2.0, an integrated set of tools and technology enabling developers to build and deploy scalable rich Internet applications, and hope that you as developers are too.

We just released Flex 2.0 beta 2, and, to celebrate, we're introducing the Flex Developer Derby with the opportunity for you to show off your killer apps and the chance to win great prizes.

Download the Adobe Flex 2.0 beta 2 (http://www.macromedia.com/cfusion/entitlement/index.cfm?e=labs) and see what you can build with Flex today. If you're new to Flex, see "Flex 2.0: Enabling the Next Generation of Rich Internet Applications" (http://www.macromedia.com/devnet/flex/articles/flex2_intro.html) and read more about the Flex 2.0 beta in Adobe Labs.

Categories and Judging Process

We've put together a great panel of judges from the developer community (these are not Adobe employees, though I'll be moderating) who will look at the applications and select one winner from each of the following categories:

- ***Product Configurators/Self-Service Guided Applications:*** Allow configuration or filtering of a service or product using the changes to the visual presentation and associated data to guide the user to a more informed decision.
- ***Components and Gadgets:*** Not all Flex and Flash sites are all Flex and Flash. Use components or user interface gadgets in an intuitive and appropriate way to increase usability of DHTML or standard Web applications.
- ***Data Dashboards:*** Provide a live visual window into business processes or snapshots of business intelligence that drive insight in a variety of configurations, potentially from multiple applications or databases. Ease of use and the ability to save or print views is also paramount since the target is usually a business user that has specific reporting milestones.
- ***Mashups:*** Combine content from more than one source to create new applications. Content often comes from public APIs or data feeds like RSS and Atom.
- ***Communication/Collaboration Applications:*** Allow two or more users to work together, in real time and in context, to achieve a specific objective. If communication between users is required, it uses some combination of audio, video, and text.
- ***Just Freakin' Cool (aka Wild Card):*** An application that doesn't fit into one of the other categories.

Using a points system, the judging panel will judge each submission on the following criteria:
- Originality
- Ease of use
- Design
- Use of Flex features

See the full Terms and Conditions (http://labs.adobe.com/showcase/special/flexderby/contest_terms.pdf) for more details on the judging process.

Prizes

The six winners, one from each category, will be selected. Each winner will receive:
- A 42-inch Samsung plasma TV and a Microsoft Xbox 360
- Your application showcased on the Adobe showcase Website (http://labs.macromedia.com/showcase/) along with the application name, description, and a link to your personal Website or blog (if applicable).
- Your application showcased in the Adobe booth at the JavaOne Conference, May 16–29, 2006, in San Francisco.

Early-Bird Submission Prizes

In addition, the first 25 developers who submit applications

that are chosen to be posted in our online gallery showcase will receive a $50 electronic gift certificate to Amazon.com. Applications must meet eligibility requirements set by the Terms and Conditions.

How to Enter

Entering the contest is easy:

- Read the Official Contest Rules and Terms and Conditions (http://www.macromedia.com/devnet/logged_in/mchotin_flexderby.html#rules).
- Download the Adobe Flex 2.0 beta 2 (http://www.macromedia.com/cfusion/entitlement/index.cfm?e=labs) and develop an application that will blow away the judges.
- Host your application on a server. If you need server hosting, check out CFDynamics (http://www.cfdynamics.com/), which is providing limited-time free server hosting for the contest.
- Submit your application by 11:59 PM Pacific Time, May 1, 2006 (http://www.macromedia.com/cfusion/mmform/index.cfm?name=flexderby_submission).

Official Contest Rules

Submission Guidelines

All applications must be hosted on a server and submitted to the Flex Developer Derby by providing a URL link to the hosted application. If you need free limited-time server hosting, please contact CFDynamics.

Official Contest Rules

Applications must be submitted by completing the form on Adobe's Website. All applications must be hosted on a server and submitted to the Flex Developer Derby by providing a URL link to the hosted application on the official submission form. No file attachments will be accepted. All submitted applications must be built with Flex Builder 2.0 beta 2 or 3 software from Adobe. Each submission is subject to the eligibility requirements, as listed in the Terms and Conditions.

NO PURCHASE NECESSARY TO ENTER OR WIN. CONTEST IS OPEN ONLY TO LEGAL RESIDENTS OF THE UNITED STATES OR CANADA, EXCLUDING QUEBEC, WHO ARE AGE 18 YEARS OR OLDER. Void where prohibited by law. Limit one (1) submission per participant. The Flex Developer Derby Contest opens March 20, 2006, and closes at 11:59:00 PDT on May 1, 2006. Six (6) winners will receive one of six (6) 42-inch plasma TVs plus a Microsoft Xbox 360 gaming system, valued at US$4,080 each. No alternative prize, cash equivalent, or other substitution is permitted. Prize is nontransferable. Winner is responsible for all taxes. The winners will be selected by a panel of independent judges, who will use an objective points system to judge in six categories. Participants agree to hold Adobe and its agents/representatives harmless from any and all losses, damages, rights, claims, and actions of any kind in connection with the Contest. Participation constitutes entrant's full unconditional agreement and acceptance of these Official Rules.

Other limitations may apply. Contest is sponsored by ADOBE SYSTEMS INCORPORATED ("Adobe" or "Sponsor") at 345 Park Ave., San Jose CA 95110. See the full Terms and Conditions (PDF, 34K).

This contest is available to United States and Canada only, and residents of jurisdictions prohibiting such promotions should consider this program void (i.e., Quebec). While Adobe makes a great effort to accommodate the international community, we cannot always guarantee that our contests will be available in every country. Please note that although this contest is available only in the United States and Canada, there may be future opportunities to participate in international contests and promotions. The Labs Showcase Gallery (http://labs.macromedia.com/showcase/)is open to residents of all countries and we welcome your submissions (http://labs.macromedia.com/wiki/index.php/showcase).

Terms and Conditions

Terms and Conditions: Free gift certificate offer available only to the first twenty-five (25) participants who submit applications that meet the following eligibility requirements: Application must be built with Flex 2.0, beta 2 or beta 3 version; application must include interactive elements; application must be virus-free; and application may not contain any unlawful, libelous, offensive, defamatory, obscene, or pornographic materials. Adobe reserves the right to verify that each submission meets eligibility requirements prior to offering gift certificates to participants and prior to posting submissions in the Showcase Gallery. Offer valid only to legal residents of the United States and Canada, excluding Quebec, who are age 18 years or older. Limit one (1) certificate per eligible submission. Applications must be received between March 20, 2006, and May 1, 2006, via the online submission form. The selected twenty-five participants will be notified within three to five days of submission by e-mail, which will detail the steps for redeeming the gift certificate. The first twenty-five eligible submissions will receive a gift certificate to Amazon.com, valued at US$50. Certificate is nontransferable. No alternative prize, cash equivalent, or other substitution is permitted. Offer subject to change or withdrawal without notice. Supplies are limited. Void where prohibited or restricted by law. For more information on the Flex Developer Derby Contest, see the full Terms and Conditions (http://labs.adobe.com/showcase/special/flexderby/contest_terms.pdf).

Questions? Contact us at: developerderby@adobe.com.

About the Author

Formerly of Macromedia, Matt Chotin is a principal engineer on the Adobe Flex team. He focuses on nonvisual features like databinding and data services but tries to answer questions of all ilk on the Flexcoders list.

```
<cfif #len(getmapinfo.city)#>
<CFIF #Val(state_left)# gt 320>
<CFSET theDetailLeft = #Val(variables.state_left)# -240>
<CFELSE>
<CFSET theDetailLeft = #Val(variables.state_left)# + 30>
</CFIF>
<CFSET theDetailTop = #Val(variables.state_top)# -20>

###state_ab#_abbrev {
  position: absolute;
  left: #variables.state_left#px;
  top: #variables.state_top#px;
  visibility:visible;
  Z-INDEX: 10;
  border:0px;
  padding:0px;
  background-color:##dedede;
  width:10px;
  font-size:10px;
  font-weight:bold;

}

###state_ab#_detail {
  position: absolute;
  left: #variables.TheDetailLeft#px;
  top: #variables.theDetailTop#px;
  width:220px;
  visibility:hidden;
  border:1px solid black;
  padding:3px;
  background-color:##f8eee3;
  z-index:30;
  font-family:Verdana, Arial, Helvetica, sans-serif;
  font-size:11px;
  color:##8f420f;
}

</cfif>
</cfoutput>
</style>

</head>

<body>

<div id="themap">
<img src="images/BaseMap100.gif" alt="US Map" title="" />
</div>


<cfoutput query="getmapinfo" group="state_ab">
```

```
<!---First, put in the state abbreviations without rollovers and
popups. We identify this by the city field being empty----->
    <cfif NOT #len(getmapinfo.city)#>

        <div id="#state_ab#_set">#state_ab#</div>

    </cfif>

  <!----Now we put in the divs for those that will have rollovers---->
    <cfif #len(getmapinfo.city)#>
        <div id="#state_ab#_abbrev"> <a href="OneStateMap.cfm?state_
ab=#state_ab#" class="map" onMouseOver="show('#state_ab#_detail')"
onMouseOut="hide('#state_ab#_detail')">#state_ab#</a> </div>

    <!---Next, create the div that will popup on mouseover------------>
        <div id="#state_ab#_detail"> <strong>#state_full#</strong><br />
            <cfoutput group="status">
                <cfif #status# is "Open">
                    <cfoutput> &middot; #city# </cfoutput> <br />
                </cfif>
                <cfif #status# is "Soon">
                    <em>Opening soon in:<br />
                    <cfoutput> &middot; #city# </cfoutput> </em> <br />
                </cfif>
            </cfoutput> <strong><em>Click now for details...</em></
strong> </div>
    </cfif>
</cfoutput>


</body>

</html>


</cfprocessingdirective>
```

Listing 2

```
<cfset variables.numberofcolumns = 3>
<cfset variables.thecolumncounter = 1>
<cfset variables.theitemcounter = 1>
<cfset variables.thetotalitems = statecount.thestatecount>
<table width="370" cellpadding="5" cellspacing="0" border="0">
    <tr>
        <td class="jobs" valign="top">
        <cfoutput query="GetOpenAndSoonCenters" group="state_full">
        <strong>#state_full#:</strong><br />
        <cfoutput>
            <cfif #city# neq #variables.thecity#>
                #city#<br />
            </cfif>
            <cfset thecity = "#city#">
        </cfoutput>
    <br />
        <cfset variables.theitemcounter = variables.theitemcounter
+ 1>
        <cfif variables.theitemcounter gte ((variables.thecolumn-
counter / variables.numberofcolumns) *
variables.thetotalitems)>
            <cfset variables.thecolumncounter = variables.thecolumn-
counter +1>
            <cfif variables.thecolumncounter lte variables.numberof-
columns>
                </td>
                <td class="jobs" valign="top">
            </cfif>
        </cfif>
        </cfoutput>
        </td>
    </tr>
</table>
```

FACULTY

Featuring... **Real-World AJAX Rock Stars!**

Jesse James Garrett *(San Jose April 24)*
Father of "AJAX" Who Coined the Term in 2005

Jesse James Garrett is the Director of User Experience Strategy and a founding partner of Adaptive Path, the world's premier user experience consulting company. He is author of The Elements of User Experience (New Riders), and is recognized as a pioneer in the field of information architecture. Jesse's clients include AT&T, Intel, Crayola, Hewlett-Packard, Motorola, and National Public Radio. Since starting in the Internet industry in 1995, Jesse has had a hands-on role in almost every aspect of Web development, from interface design and programming to content development and high-level strategy. Today, information architects around the world depend on the tools and concepts he has developed, including the widely acclaimed "Elements of User Experience" model. He is co-founder of the Information Architecture Institute, the only professional organization dedicated to information architecture. He is also a frequent speaker and writer whose work has appeared in numerous publications, including New Architect, Digital Web, and Boxes and Arrows.

Adam Bosworth *(San Jose April 24)*
Vice President of Engineering, Google
One of the Fathers of XML & the Creator of MS Access

Adam Bosworth is Vice President of Engineering, Google. He joined Google in 2005 from BEA Systems, where he was Chief Architect & Senior VP of Advanced Development. Prior to joining BEA, Bosworth co-founded Crossgain, a software development firm acquired by BEA. Known as one of the pioneers of XML, he previously held various senior management positions at Microsoft, including General Manager of the WebData group, a team focused on defining and driving XML strategy. While at Microsoft he was also responsible for designing and delivering the Microsoft Access PC Database product and assembling and driving the team that developed the HTML engine of Internet Explorer 4.0.

Dion Hinchcliffe *(San Jose April 24)*
Cofounder & CTO, Sphere of Influence Inc.
Editor-in-Chief, Web 2.0 Journal

Dion Hinchcliffe, newly appointed editor-in-chief of SYS-CON's pioneering Web 2.0 Journal, is cofounder and chief technology officer for the enterprise architecture firm Sphere of Influence Inc., in McLean, Virginia. A veteran of software development, Dion works with leading-edge technologies to accelerate project schedules and raise the bar for software quality. He is highly experienced with enterprise technologies and he designs, consults, and writes prolifically. Dion actively consults with enterprise IT clients in the federal government and Fortune 1000. He is a frequent speaker on AJAX, Web 2.0 and SOA and is currently the top-read SYS-CON.com blogger.

Christophe Coenraets *(San Jose April 24)*
Senior Technical Evangelist, Adobe
AJAX/Flex Integration Guru

Christophe Coenraets currently works as a Senior Technical Evangelist at Adobe. Before joining Adobe, Christophe was an evangelist at Macromedia, focusing on Rich Internet Applications and Enterprise integration. Prior to Macromedia, Christophe was the head of Java and J2EE Technical Evangelism at Sybase, where he started working on Java Enterprise projects in 1996. Before joining Sybase in the US, Christophe held different positions at Powersoft in Belgium, including Principal Consultant for PowerBuilder, and Manager of the Professional Services organization. Before joining Powersoft, Christophe worked as a developer and architect on several retail and BPM projects. Christophe has been a regular speaker at conferences worldwide for the last 10 years.

Paul Rademacher *(San Jose April 24)*
Google, Creator of HousingMaps.com

Paul Rademacher is the creator of HousingMaps.com, which combined Craigslist and Google Maps for the first web mashup. Paul holds a Ph.D. in Computer Science from UNC-Chapel Hill, and worked as an R&D Engineer at Dreamworks Animation on such movies as Shrek 2 and Madagascar. Since creating HousingMaps, Paul is now at Google.

Jouk Pleiter *(San Jose April 24)*
Co-Founder & CEO of Backbase

Jouk Pleiter is the CEO of Backbase, a leader in the field of Rich Internet Applications and AJAX development software. Backbase's clients include ING, ABN AMRO, TNT, KPN, Comsys and Heineken. Backbase operates globally with offices in San Mateo (North America) and Amsterdam (Europe). Since 1995, Jouk has been an entrepreneur: he founded three successful software companies. Prior to Backbase, Jouk was part of the founding team at the web content management company Tridion, where he led the product management operations, and was driving the company's efforts to become a leader in the European WCM software market. Jouk previously was part of the founding team at the Interactive Agency Twinspark where he grew the company to a leading market position in Europe and was instrumental in the sale of Twinspark to Agency.com. He has an MBA from the University of Groningen.

Kevin Hakman *(San Jose April 24)*
Director of Product Marketing for TIBCO
General Interface TIBCO Software

Kevin Hakman is the director of product marketing for TIBCO General Interface, the award winning AJAX and Rich Internet Application framework and toolkit. Kevin Hakman pioneered AJAX in the enterprise co-founding General Interface in 2001. Since that time General Interface (aka 'GI') has been powering Web applications that look, feel and perform like desktop applications, but run in the browser at Fortune 500 and U.S. Government organizations. General Interface was also the first to use its own toolkit to provide full visual tooling for AJAX when it released it's 2.0 Version in 2003. TIBCO acquired General Interface in 2004 to extend its vision for service oriented applications to the end user. Kevin is a contributor to the SOA Web Services Journal and the AJAX Developer's Journal.

Shanku Niyogi *(San Jose April 24)*
Product Unit Manager of the UI Framework and Services Team Microsoft Corporation

Shanku is Product Unit Manager of the UI Framework and Services (UiFX) team, which is responsible for delivering high-productivity UI framework technologies for the .NET platform, including ASP.NET, Atlas, Windows Forms, and frameworks for smart clients. Prior to his current role, Shanku was Group Program Manager of the Web Platform and Tools team on the Whidbey release of ASP.NET and Visual Web Developer. Shanku joined Microsoft in 1998 as a developer, having spent several years shipping products in the Windows ISV industry. Shanku holds a Bachelor of Mathematics degree in Computer Science from the University of Waterloo.

Coach Wei *(New York June 5-6)*
Chairman, Founder and CTO, Nexaweb
The Creator of First Commercial AJAX Applications

Coach Wei combines in-depth IT industry expertise with extensive education and research experience at MIT to drive technology innovation and business direction for Nexaweb. He founded Nexaweb in 2000 and served as CEO until summer 2003. Before founding Nexaweb, Coach architected and designed enterprise software for managing storage networks at EMC Corporation. As a graduate researcher at MIT, Coach developed software and hardware systems for non-destructive evaluation as well as signal/image processing algorithms. Coach was a finalist in the 1999 MIT $50K entrepreneurship competition and holds several U.S. patents. An accomplished writer and speaker, Coach has published numerous articles on topics including: AJAX, J2EE and .NET, RIA development, XML, signal/image processing, composite materials and ultrasonic imaging. He has spoken at top industry events, such as JavaOne and Web Services Edge. Coach holds an MS in information technology from MIT.

Ajit Jaokar *(New York June 5-6)*
CEO, futuretext
Author, "Mobile Web 2.0"

Ajit Jaokar, based in London (England), is the CEO of a publishing company, futuretext (www.futuretext.com). He is currently writing a book about Mobile Web 2.0 (Mobile Web 2.0: The Innovator's Guide to Developing and Marketing Next Generation Wireless / Mobile Applications). Ajit also chairs Oxford University's Next-Generation Mobile Applications Panel and, since January 2006, has been a member of the Web 2.0 Workgroup. In his "Real-World AJAX" conference session, Ajit will discuss the "AJAX Use in Mobile Applications" as part of the wider impact of Web 2.0, sometimes referred to as the "Global SOA."

Jonas Jakobi *(New York June 5-6)*
AJAX Evangelist and Co-Author, "Ajax and JSF: Friend or Foe?"
Jonas will autograph a copy of his book for all delegates!

Jonas Jacobi is a principal product manager and evangelist for Oracle's Java/J2EE tool offering, JDeveloper, and over the past three years has been responsible for JavaServer Faces, Oracle ADF Faces, and Oracle ADF Faces Rich Client development features within Oracle JDeveloper. Jonas has been in the software business for 15 years. Prior to joining Oracle, he worked at several software companies in Europe, covering many roles including support, consulting, development, and project team leadership. Jonas' new book "Ajax and JSF: Friend or Foe?" released by Apress on February 25, 2006.

John Fallows *(New York June 5-6)*
AJAX Evangelist and Co-Author, "Ajax and JSF: Friend or Foe?"
Jonas will autograph a copy of his book for all delegates!

John Fallows, former lead developer for Oracle ADF Faces Rich Client, has been working in distributed systems for over a decade. After five years spent focused on designing, developing the JavaServer Faces standard to provide AJAX functionality, playing a leading role in the Oracle ADF Faces team, he recently joined an AJAX start-up. Originally from Northern Ireland, John graduated from Cambridge University in the United Kingdom and has worked in the software industry for more than ten years. Prior to joining Oracle, he worked as a research scientist for British Telecommunications Plc.

Steve Benfield *(New York June 5-6)*
Well-known AJAX Evangelist and CTO of Agentis Software
Steve's first talk on "Aspect-Oriented Programming & AJAX"

Steve Benfield is CTO of Agentis Software and one of the pioneers of AJAX technology, a gifted writer and a technical visionary. A technology marketeer and strategist with 20 years of software entrepreneurism experience, a combination of qualities that made him the perfect choice of editor-in-chief for SYS-CON Media's inaugural publication 12 years ago. Steve's proven ability to determine marketing and technology strategies that align with market needs led to successful stints at SilverStream, where he started as technology evangelist and ended as CTO, and at ClearNova, an open source AJAX company, where he was CTO and AJAX evangelist.

Jeremy Geelan
Conference Chair
Group Publisher & Editorial Director, SYS-CON

Jeremy Geelan is group publisher and editorial director of SYS-CON Media, and is responsible for all print titles and online i-technology portals for the firm. He regularly hosts SYS-CON.TV, is executive producer of the "Power Panels with Jeremy Geelan" iTV series, and represents SYS-CON at conferences and trade shows, speaking to technology audiences both in North America and overseas. His i-Technology Blog is at jeremy.linuxworld.com and he is conference chair of the upcoming iTVcon - "Internet TV Conference & Expo 2006".

What you'll walk away with...
A New understanding of AJAX and how to make it work for you, plus:

A) **Conference Proceedings Binder**
   Full-conference proceedings in a self-contained commemorative binder.

B) *Ajax in Action* **Book**
   A copy of the Best-Selling AJAX book by Dave Crane.

C) *Real-World AJAX "Secrets of the Masters"* **Book + DVD**
   New AJAX book edited by Dion Hinchcliffe (Release Date: Summer 2006)

D) **Notebook**
   Executive style keeps you looking sharp for note-taking wherever your business might take you!

E) **Computer Back-Pack**
   A hip new way to conceal your laptop! The Urban Wonder Compu-Pack is an ingenious creation - a 600-denier polycanvas and nylon knapsack!

F) **T-Shirt**
   100% preshrunk 6.1 oz. heavyweight cotton tagless shirt features shoulder-to-shoulder tape, and double-needle sleeves and bottom.

G) **Porcelain Mug**
   Lightweight and durable, you can use it as a teacup, coffee mug or keep it as a pencil holder.

H) **Satin Silver Contemporary Pen**
   Satin-silver plastic barrel with the look and feel of brass. Rubber comfort grip improves writing control and comfort.

I) **Real-World AJAX Seminar on DVD**
   Watch complete seminar video and slide presentations (Release Date: Summer 2006)

For more information...
Call 201-802-3022 or email events@sys-con.com

SYS-CON EVENTS   **For more great events visit www.EVENTS.SYS-CON.com**

NOTE: SPEAKER LINE-UP SUBJECT TO CHANGE WITHOUT NOTICE
VISIT WWW.AJAXSEMINAR.COM FOR THE MOST COMPLETE UP-TO-DATE INFORMATION

What's New with Flex, ColdFusion, and More

Interview with Ben Forta

Simon Horwith: *Today I'm speaking with Ben Forta, the technical evangelist for Adobe. The first thing I have to ask you is – the acquisition has happened, Adobe is now controlling ColdFusion and has taken over everything that Macromedia was doing, and this is my first chance to really speak with you since that happened – what is going on in the ColdFusion world? How has the acquisition gone for the ColdFusion team and yourself and what's in store for us in the near future?*

Ben Forta: Thus far, not a whole lot has changed. Before the acquisition we were hard at work on ColdFusion Scorpio. We first mentioned it over a year ago, the next major version of ColdFusion. That work began pre-acquisition and it's ongoing. We're still working through the specs and futures and what the price should be. Independent of that, also pre-acquisition, we started work on the Mystic bits, which is the update of the ColdFusion 7.01 full Flex integration. That work is still happening. As far as the short-term implications and the data impact of ColdFusion immediately, not a whole lot. It's still pretty much the same team doing exactly what they did before with the same immediate goal and longer-term goal.

Beyond that it starts to get very interesting. You're starting to really see the beginnings of integration between products, so ColdFusion and Flex are obviously extremely important, but once you're beyond that then there are things ColdFusion developers have been asking for awhile that Adobe's in a really great position to help with, such as improving PDF generation and some of the imaging functionality people ask for. These are things that are Adobe's area of expertise and they're definitely the ones to help us build this stuff. Just looking at the future of ColdFusion, there's definitely a lot of Adobe expertise that needs to make its way into ColdFusion.

Beyond that, there are other parts we have to start thinking about integrating with, so some of the lifecycle servers, for example, it might make sense to provide some ColdFusion functionality to them and to expose some of that functionality back to CF. While we are building the next releases of the product, we are looking at the wider array of product offerings now, technologies, and trying to figure out what the right mix is and how we can add value everywhere. The end result is a much better product and a better set of services and solutions for customers.

SH: *Can't wait to see that. You mentioned Flex, and I know that currently you're on the road meeting with user groups to talk about Flex 2.0 and Flex Builder 2.0. For the benefit of our viewers who aren't familiar with it, could you briefly describe what the major differences between Flex 2.0 and Flex 1.5 are and maybe even just quickly introduce what Flex is, so if there's anybody out there who hasn't seen this yet…*

BF: Oh, wow. You want me to take the full couple hour presentation and turn it into a 30-second sound bite, great.

Let's talk about Flex currently; Flex is like solving a very specific problem. The Flash player, the Flash platform, is a very capable platform for delivering and employing rich, engaging experiences. It's primarily used for design work and has been for a long time, but it's actually capable of a lot more. It's capable of being a front end for applications, for business apps, and for being the client-side of a real application, talking to real back-end data beyond the design-centric applications that you often see in Flash. While the platform is capable of doing that, the

experience, the process of building those apps has been complex or difficult for most coders and developers, the way they think about building apps. The Flash IDE is something that designers are far more comfortable with than coders, and while there were changes made in the last few versions of Flash, particularly in Flash 2004, to make Flash a better tool for coders, it still is something that designers are a lot more comfortable with than those who also write code for a living. The tool itself, the way to build apps, how the application is put together, the way code is embedded in the app, this entire process and tooling is something that is a little foreign to coders.

Flex is a whole different way of thinking about building Flash applications. Instead of starting with a stage, a graphical stage that you drop elements on and a timeline that you script around, simply start the code. You have tags, script, and a server. Create these files using a tag-based markup language called MXML and use ActionScript as a script language in conjunction with that, and what you end up creating is a SWF. It's a Flash binary that runs in the player just as if you had created it in the Flash IDE, but the way you get there is totally different. It's all code based. It's designed for a different type of developer, a different type of application, and it's ideally suited for the types of apps that coders would build, although it's entirely possible to still build apps like that in Flash. That's what Flex is right now.

The current shipping version of Flex is Flex 1.5, and the products have been well-publicized successes. Obviously the highest profile one is the Yahoo Maps beta, which is running on Flex, an extremely powerful application and very responsive, a very engaging experience, and is really, as I said, a benchmark for what Flex apps should be.

However, there were some limitations around Flex 1 that made it perhaps less than attractive for some developers, particularly ColdFusion developers. The objections that we primarily hear form three categories. One is pricing; Flex 1 is a server product and it's priced as such. When you look at the demographic of who the ColdFusion developers are, the type of apps they're building, the budgets those apps have, perhaps the price point is too high for them.

Number two is the whole deployment model. When you build an app in Flex 1, you need to deploy the Flex server on a J2EE server, and the majority of ColdFusion developers avoid J2EE servers. Even those who have ColdFusion Enterprise will be exposed to the J2EE server, but they don't use it for the most part, and so they use a simpler deployment model. Number three is the tooling, which is simply an easy way to build the apps in general. All of those are addressed by Flex 2 and there's a lot more to Flex 2 than just that but, at least through the feedback we're getting from ColdFusion developers, what they like about Flex 2 is that these three issues have been addressed. One is the pricing. We haven't released final pricing yet but we have publicly stated that the Flex Builder tool will sell for under a $1,000 per developer seat, so it's priced at a price point that's a lot more appealing to ColdFusion developers.

In addition, you will now be able to build applications with the tool, with the IDE of a Flex Builder, and build the app, compile it, generate a SWF, and then deploy that SWF like any other asset on your server. As you deploy your HTML files and your CFM files and GIFs, JPEGs, and CSS files, you also deploy your SWF files in the server and they get served by the HTTP server like any other pages; you don't need these services running on the server that you did previously so it's a far simpler deployment.

There are additional services, the Flex Enterprise services, and these are covered in detail in the user group presentations. These are additional functionality, messaging, and data services, and many developers will want to leverage that functionality, but for those who want to build basic applications, applications that talk to back ends, perhaps to ColdFusion via Flash remoting or via SOAP, and get the results back again, that would be possible using Flex Builder without needing to deploy services in the server. This makes the whole deployment a lot easier as well. Third is the new tool. Flex Builder is a tool that is really designed for coders. It's built on Eclipse, which has proven itself as a coder's platform, a coder's IDE, and so the combination of these three things really make Flex 2 very compelling to ColdFusion developers, and that's just a small part of what Flex 2 is about. There's a new ActionScript language; there's dramatically improved performance; there are enhancements to all the controls to it, there's a whole lot to it.

SH: *You mentioned that Flex can be very appealing to ColdFusion developers. Typically when you go out on the road, which you do almost every year doing a user group tour, it's almost always coinciding with a major release, typically in the ColdFusion product. This tour you're going around and telling everybody about Flex 2.0 and Flex Builder 2. Are ColdFusion developers the target audience for your presentations and for the product itself? Care to talk just a little bit about who the real target audience is here for Flex 2.0?*
BF: Flex 2 is not targeted solely at ColdFusion developers. It's being targeted at developers in general. ColdFusion developers are a very important part of that demographic because they're building apps that Flex can help solve real problems for and improve, and the integration between the two is so clean and so seamless that it just becomes another tool that they can work with; the end result is something vastly superior what they have right now. I think the ColdFusion user base is an important part of who Flex is for but it's not only for the ColdFusion developers. We have really ambitious numbers as to how we expect to grow Flex Builder use and that's going to be far bigger than just the ColdFusion audience, so it appeals to Java developers, to .NET developers, to PHP developers, and it should. The products, Flex and ColdFusion, shouldn't be dependent on each other. They have to be independently successful, but when you use the two together, you end up with something better than the sum of their parts. That's Flex in general.

As far as the user groups, yes, most of the prior tools have been around ColdFusion releases and this is a little different in that it's not around a ColdFusion release; it's more around Flex 2, which will have ColdFusion data associated with it, but the intent here is to expose the ColdFusion community to Flex. Many of them saw Flex 1 in the past, and some tried it with varying degrees of success. I really want them to take a look at Flex again, because Flex 2 is radically different from what they're used to and based on responses thus far they're really excited about it. We're now a week into the tour and I'm getting all sorts of feedback and comments that people are already downloading Flex Builder, they're already trying it, they're already building the sample apps and have been very successful with it, which was a little harder for them to do in the Flex 1 era. The initial feedback from the ColdFusion community and those who attended the groups is very good, but you're right, it is also attracting a slightly wider audience. I did a presentation in Philadelphia last night and there were attendees who were Flash

developers, who were using Flash in talking to PHP with no ColdFusion experience at all, so it definitely has a wider reach.

SH: *You mentioned that Flex Builder 2 is built on Eclipse and you're aware that there is also a CFEclipse project out there that offers ColdFusion developers a good Eclipse-based environment that's different for ColdFusion developing and, obviously, Java developers and other developers have been using Eclipse as well. What does this mean? Do you think that Adobe is going to begin promoting Eclipse as the recommended development platform for both ColdFusion and Flex development? I know Macromedia joined the advisory panel for CFEclipse. Are we looking at having one unified tool for doing all of our hard-core development?*

BF: Eclipse is definitely the IDE for Flex development and there are other initiatives in the company using Eclipse in other ways, too. So yes, you're going to see a lot of momentum around Eclipse. But with ColdFusion specifically, I think we've taken this dual approach to it, and for many developers Dreamweaver is still a good IDE for ColdFusion developers, especially for those who are beginners or just starting off, who are already very comfortable with Dreamweaver, or are doing a lot of HTML work or a lot CSS work, in addition to ColdFusion. Dreamweaver in the early days didn't have as sophisticated support for ColdFusion as it would have liked, but it has gotten better, and in fact it's gotten really good. The CFC support, for example, in Dreamweaver is superb, and the integrated debugging is very, very good. Even the wizards extensions we wrote, where you can, for example, add data sources to ColdFusion directly to Dreamweaver. There are some very important features within Dreamweaver that have made it a very compelling tool for some ColdFusion developers, but it isn't for all developers and we recognize that.

I don't think there was ever the intent that Dreamweaver would replace the old ColdFusion Studio, which some people still use. That's why every time you bought a copy of Dreamweaver, it included a copy of Home Site Plus, which is essentially ColdFusion Studio, because there was the recognition that Dreamweaver really was not a replacement for what the old ColdFusion Studio Home Site–based product was, and it appealed to a different audience and developer base. CFEclipse is an important project. It does answer part of the void left by there being no new additions to ColdFusion Studio Home Site for quite a while. There have been tag updates but nothing else. So the ColdFusion team has gotten behind the CFEclipse project. We announced support for it at CFUnited last year. We are working on features already. At the user group this week I've been publicly demo-ing, for example, an RDS plugin that we built to be used with Eclipse.

We will be getting very involved in the project; we are already and continue to do so. We're not going to take over the project, that needs to be community driven. That's the way it will be successful. We are very involved with this and if we end up with this dual approach for IDEs, for those who want Dreamweaver, it's still the IDE of choice; for those who want an Eclipse-based IDE, it's a very different experience, great. CFEclipse is there. It's a powerful tool. It's been kind of light for a while but it's gotten a new burst of life recently and that's really good to see. We are going to keep writing functionality to it and it can be used with

Flex Builder as well. You can install Flex Builder in Eclipse, and the end result is something very useful.

SH: *Did you tell me there was an announcement by Adobe about the planned pricing model for the Flex 2 SDK. They basically announced that it's going to be made available for free. I was wondering if you could tell us a little more about that and particularly what exactly do you get in the SDK?*

BF: There are several parts to the Flex story. Flex Builder obviously is the IDE. It's where you do your developments and it has an integrated compiler. It has a very sophisticated debugger built in, all sorts of help, and you end up building an application that gets deployed on the server. In addition to the Flex Builder, there are a set of services that can be installed at runtime in the server. These are the Flex Enterprise services, and we haven't really announced pricing for that part of it yet but it will be a scale pricing starting off at something very manageable. As your needs increase, there will be a way to buy more use of it and scale appropriately, but we haven't discussed the final Flex Enterprise services pricing yet. When you build an app with Flex Builder, you can build apps that are stand-alone or apps that are built in conjunction with and deployed using the Flex Enterprise services. Those are the two products that we sell.

In addition to that, what we did announce was that the Flex 2 SDK would be made available at no cost. That essentially is the framework and the compiler. If you want to build an application using MXML and ActionScript and don't plan to use the Enterprise services, and don't plan to use our IDE for building the app, you actually don't pay us anything at all. You just download the framework and the compiler and use an editor of your choice. If you feel like using Notepad, so be it. Build your XML, build your script, compile it, and you can actually build scripts and build an actual application that way. It's an important part of making the Flash platform far more prevalent by encouraging people to really tinker with it, see what they can do. I think they'll be very pleased.

SH: *One thing I also wanted to ask you about, being a developer myself, just something that is a concern of mine and I'm just interested to hear your take on this. Prior to Flex 2.0, the Flex server was priced out of the range for most smaller companies and Internet developers. Now obviously it's going to be much more reasonable financially for developers and smaller companies to use. It's also a free public beta, so we have tons of developers who are downloading and playing with Flex and really learning it and using it for the very first time. It immediately makes me question whether or not Flex is going to become the blink tag of the 21st century.*

BF: You didn't like the blink tag?

SH: *Yeah, I really don't miss the blink tag. Definitely, you have so many people who have no prior knowledge of Flex and they're hoping to become proficient overnight and be able to implement this into their projects in the near future. Flex really makes it easy to do some very flashy things, such as drag-and-drop, hiding, and turning on and off elements on the screen and making things zoom –there are some neat effects that it can do. Is there any*

ENGAGE AND EXPLORE...

The Technologies, Solutions and Applications that are Driving Today's Initiatives and Strategies...

concern at Adobe or that you personally have about the developers going overboard with this use or even if they don't, just not learning how to implement best practices and build applications the right way with this tool?

BF: Yes, but actually I don't think that's as much a Flex concern as much as it is a core Flash concern. People abuse Flash already right now. We've all seen the pop-up ads that you move your mouse over and they kind of roll over and take over your screen and you can't close them. Often I'll ask people what association – what do you think of when you think of Flash and a lot of people still say skip intro. So I think Flash itself allows people to build really cool things, and also now it allows people to probably go overboard, and there's an education process. You have to learn what's appropriate and what's not. It depends upon the message. It depends on the type of application. It depends on how it's being delivered. It depends the target audience. That is something that people need to be educated about and I think we've been doing a pretty good job in general for that around Flash.

With Flex it will be more of the same, yes. An important part of this is going to be educating people to what the right way to build apps is, understanding that a Flex app isn't a replacement for a single page in an HTML application because that won't scale, it won't be comfortable for people to use. The formats won't be what you want and there is a difference. A Web app isn't the same as a Rich Internet App if you have to rethink how you build the apps and how the pieces are put together, and the flow through the application, user interaction.

There are a lot of people who will learn as they go. If they build it incorrectly, if they suddenly take an HTML form, guess what, I'm going to turn this into a Flex app, built by Flex; it's a Flash form, drop in an HTML page. They're going to learn very quickly that isn't the result people want.

I think actually you'll be getting two fundamental questions; one is Flash in general and educating people what's correct to use. I think we've made real progress there. The other part is just explaining to people how the migration from a Web app to a Flex-powered Rich Internet Application, how that process should go. It really is taking a step back and rethinking your app; we've already started showing people the example apps, show that when you install Flex and you get the snippets in the explorer and how to write code, those are obviously little snippets and that's not how we actually build real applications. Building real apps does require taking a step back, rethinking the flow, rethinking the communication between presentation for the clients and the server. That has to happen and that is happening. We've already started building examples and writing up docs for that as well. There will be a lot more around it as well. There will be training, there will be documents that explain the best practices and how to use the technology, so that's what we're working on. It's important and I really think it needs to be addressed and we are working on that.

SH: *I'm glad you mentioned that developers have to rethink the way they actually present these interfaces to people and the sample applications. That was the other thing that I really wanted to bring up with you. At the MAX conference in California, the Macromedia Extreme Design Group showed an application that,*

for lack of a better word, just absolutely blew me away. It was absolutely amazing and everybody's jaw dropped when we saw the entire video store application to rebrand itself. "Spiderman," I guess, was the sample, when the guy was typing in the search box. The Extreme Design Group was showing their view of what user experiences are going to be like in the very near future with these next generation of Web 2.0 type applications. That application was built with Flex and the tools are available today to build those kinds of things. Is the Extreme Design Group or any other group within Adobe planning to do anything to help developers really learn how to rethink the way they approach presenting data, services, and interfaces to end users?

BF: I think the answer to that is an obvious yes. I mean that's part of the core education I was just talking about. There is an education process in teaching people how to use technology and how to use it effectively. Part of that is actually in the core product, so a lot of things we've done at Flex 2, some of the new controls, some of the way you can do skinning now and styling to get the look that you want, make that a whole lot easier as well. I think some of that thinking is really making this the core product, and, yes, there is additional education that needs to happen around that and that's a work in progress.

SH: *There's been a lot of buzz around Flex, and I think it's fantastic so far with everybody getting to see it, and everybody is really anticipating this release. Do you have any other thoughts, anything else that you want to tell our viewers before I let you go?*

BF: The only other thing I'll add is that I would really encourage people to play with it. We've made this beta really open, actually made the alpha open, as well, and we really want people to play with this technology and see what it can do. It's a lot of fun. It's addictive and there's an instant gratification factor to it, and people are having a whole lot of fun with it. What I really want people to do, though, is to start to play with the ColdFusion Flex integration. The truth is, if you build an app in Flex 1 it could talk to a ColdFusion back end by default because the Flash player knows how to make HTTP calls and knows how to talk to – via Flash remoting – talk to Web services. ColdFusion obviously can respond to all of those, but there was nothing really built into the product on either end. I mean, one particularly suited to the other. If you're a ColdFusion developer, obviously you used ColdFusion as a back end for your Flex app but there was nothing particularly compelling as far as the integration goes between the two.

That has changed. In Flex 2 the integration is really nice and really clean, and so the new Flash remoting adapter that gets installed for ColdFusion 7 does really important things like automatic diversion of objects between ColdFusion CFCs and a Flash group and then back again. Some of the wizards we're working on for Flex Builder are specific to ColdFusion, so a right-click and ActionScript object can generate the appropriate CFC with all the properties and so on. If you're using the Flex Enterprise services, the gateway that we install into ColdFusion makes it really easy to send the data from ColdFusion, push down to the client with a single function call, probably far easier than any other back-end platform. I think ColdFusion developers are going to discover when they start playing with ColdFusion and Flex together that Flex doesn't just support ColdFusion, but

that the two work very well together and have been designed to do just that.

ColdFusion is a first-class citizen when it comes to being a back end for Flex. The two are designed to work very well together and ColdFusion developers in that way have an edge in that they have all the power, the simplicity, and productivity that is ColdFusion, which is why they use the product already. In addition, they are now in a really good place when it comes to building these next-generation Rich Internet Applications with a powerful Flex front end and a ColdFusion back end. I really encourage people to download the beta, to go through to the lab site, labs.Adobe.com, download Flex Builder 2 and download Flex Enterprise services if you want, play with that as well, make sure you download the updated bits for ColdFusion. They do require Cold Fusion 7.01 and download those bits as well; if you want to play with the charting components, grab those. There are good examples up there, demos, postings all over the place, and blogs elsewhere with examples to try, so just play with it. It really is a lot of fun to play with and the results are pretty amazing.

SH: *Thanks for talking with me, Ben. I look forward to seeing what happens with Flex 2.0 when all of its various components are officially released and, obviously, we are also looking forward to seeing what's in store for ColdFusion in the next year or two.*

CFDJ Advertiser Index

ColdFusion U

For more information go to...

U.S.

Alabama
Huntsville
Huntsville, AL CFUG
www.nacfug.com

Alaska
Anchorage
Alaska Macromedia User Group
www.akmmug.org

Arizona
Phoenix
www.azcfug.org

Arizona
Tucson
www.tucsoncfug.org

California
San Francisco
Bay Area CFUG
www.bacfug.net

California
Riverside
Inland Empire CFUG
www.sccfug.org

California
EL Segundo
Los Amgeles CFUG
www.sccfug.org

California
Irvine
Orange County CFUG
www.sccfug.org

California
Davis
Sacramento, CA CFUG
www.saccfug.org

California
San Jose (temporary)
Silicon Valley CFUG
www.siliconvalleycfug.com

California
San Diego
San Diego, CA CFUG
www.sdcfug.org/

California
Long Beach
Southern California CFUG
www.sccfug.org

Colorado
Denver
Denver CFUG
www.denvercfug.org/

Delaware
Kennett Square
Wilmington CFUG
www.bvcfug.org/

Delaware
Laurel
Delmarva CFUG
www.delmarva-cfug.org

Florida
Jacksonville
Jacksonville, FL CFUG
www.jaxfusion.org/

Florida
Winter Springs
Gainesville, FL CFUG
www.gisfusion.com/

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

Florida
Tallahassee
Tallahassee, FL CFUG
www.tcfug.com/

Florida
Palm Harbor
Tampa, FL CFUG
www.tbmmug.org

Georgia
Atlanta
Atlanta, GA CFUG
www.acfug.org

Illinois
East Central
East Central Illinois CFUG
www.ecicfug.org/

Indiana
Avon
Indianapolis, IN CFUG
www.hoosierfusion.com

Indiana
Mishawaka
Northern Indiana CFUG
www.ninmug.org

Iowa
Johnston
Des Moines, IA CFUG
www.hungrycow.com/cfug/

Kentucky
Louisville
Louisville, KY CFUG
www.kymug.com/

Louisiana
Lafayette
Lafayette, LA MMUG
www.cflib.org/acadiana/

Maryland
Lexington Park
California, MD CFUG
http://www.smdcfug.org

Maryland
Rockville
Maryland CFUG
www.cfug-md.org

Massachusetts
Quincy
Boston, MA CFUG
www.bostoncfug.com

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

Minnesota
Brooklyn Park
Twin Cities CFUG
www.colderfusion.com

Missouri
Overland Park
Kansas City, MO CFUG
www.kcfusion.org

Missouri
O'Fallon
St. Louis, MO CFUG
www.stlmmug.com/

New Jersey
Princeton
Central New Jersey CFUG
http://www.cjcfug.us/

Nevada
Las Vegas
Las Vegas CFUG
www.sncfug.com/

New York
Albany
Albany, NY CFUG
www.anycfug.org

New York
Brooklyn
New York, NY CFUG
www.nycfug.org

New York
Syracuse
Syracuse, NY CFUG
www.cfugcny.org

North Carolina
Raleigh
Raleigh, NC CFUG
www.ccfug.org

Ohio
Dayton
Greater Dayton CFUG
www.cfdayton.com

Oregon
Portland
Portland, OR CFUG
www.pdxcfug.org

User Groups

http://www.macromedia.com/cfusion/usergroups

INTERNATIONAL

Australia
ACT CFUG
www.actcfug.com

Australia
Queensland CFUG
www.qld.cfug.org.au/

Australia
Southern Australia CFUG
www.cfug.org.au/

Australia
Victoria CFUG
www.cfcentral.com.au

Australia
Western Australia CFUG
www.cfugwa.com/

Brazil
Brasilia CFUG
www.cfugdf.com.br

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

Brazil
Sao Paulo CFUG
www.cfugsp.com.br

Canada
Kingston, ON CFUG
www.kcfug.org

Canada
Toronto, ON CFUG
www.cfugtoronto.org

Ireland
Dublin, Ireland CFUG
www.mmug-dublin.com/

Italy
Italy CFUG
www.cfmentor.com

Japan
Japan CFUG
cfusion.itfrontier.co.jp/jcfug/jcfug.cfm

Scotland
Scottish CFUG
www.scottishcfug.com

South Africa
Joe-Burg, South Africa CFUG
www.mmug.co.za

South Africa
Cape Town, South Africa CFUG
www.mmug.co.za

Spain
Spanish CFUG
www.cfugspain.org

Switzerland
Swiss CFUG
www.swisscfug.org

Thailand
Bangkok, Thailand CFUG
thaicfug.tei.or.th/

Turkey
Turkey CFUG
www.cftr.net

United Kingdom
UK CFUG
www.ukcfug.org

Pennsylvania
Carlisle
Central Penn CFUG
www.centralpenncfug.org

Pennsylvania
Exton
Philadelphia, PA CFUG
www.phillycfug.org/

Pennsylvania
State College
State College, PA CFUG
www.mmug-sc.org/

Rhode Island
Providence
Providence, RI CFUG
www.ricfug.com/www/meetings.cfm

Tennessee
LaVergne
Nashville, TN CFUG
www.ncfug.com

Tennessee
Germantown
Memphis, TN CFUG
www.mmug.mind-over-data.com

Texas
Austin
Austin, TX CFUG
www.cftexas.net/

Texas
Corinth
Dallas, TX CFUG
www.dfwcfug.org/

Texas
Houston
Houston Area CFUG
www.houcfug.org

Utah
North Salt Lake
Salt Lake City, UT CFUG
www.slcfug.org

Washington
Seattle
Seattle CFUG
www.seattlecfug.com

About CFUGs

ColdFusion User Groups provide a forum of support and technology to Web professionals of all levels and professions. Whether you're a designer, seasoned developer, or just starting out – ColdFusion User Groups strengthen community, increase networking, unveil the latest technology innovations, and reveal the techniques that turn novices into experts, and experts into gurus.

An In-Depth Look at Mapping APIs

The benefits and drawbacks

By Joe Danziger

Back in the November 2005 issue of *CFDJ*, we looked at how to create a live map of your Web site visitors using Google Maps and other freely available services.

Now that Yahoo! and Microsoft have joined the fray and released their own mapping APIs, we'll revisit our example while learning about each implementation along the way. All of these APIs are extremely powerful, but each has its own advantages that may work better for your particular solution. We'll take a look at each in detail and outline some of the benefits and drawbacks of each.

Geo-locating by IP Address

Let's go over our example one more time. We're going to use these tools to create a visual map of users visiting your site. You'll be able to show these on a custom map along with any other text or HTML data you'd like to display. One of the key ingredients to this technique is the ability to geo-locate an IP address. This refers to the ability to locate a user's geographic latitude and longitude from his IP address. The most cost-effective source of this information is to query one of the freely available public databases. NetGeo was a project originally set up by the Cooperative Association for Internet Data Analysis (CAIDA). Although it has been defunct for several years, the database is still freely available and provides a good resource for determining rough locations.

We'll set up a table (see Listing 1) to keep a log of user visits. Each time a new user visits the site, we'll query the geolocation database and insert a record of the user's IP address, latitude, longitude, city, state, country. In CFMX 7, this code can be put inside the onSessionStart() function in Application.cfc.

Queries are executed via HTTP against the NetGeo database by calling the URL http://netgeo.caida.org/perl/netgeo.cgi?target=64.19.164.74. It will return the following data to the browser:

```
VERSION=1.0
TARGET: 64.19.164.74
NAME: MONMOUTH-BLK2
NUMBER: 64.19.128.0 - 64.19.191.255
CITY: RED BANK
```

```
STATE: NEW JERSEY
COUNTRY: US
LAT: 40.35
LONG: -74.08
LAT_LONG_GRAN: City
LAST_UPDATED: 01-Jun-2001
NIC: ARIN
LOOKUP_TYPE: Block Allocation
RATING:
DOMAIN_GUESS: MONMOUTH.COM
STATUS: OK
```

ColdFusion's string functions can be used to parse out the relevant latitude and longitude data, as well as the city, state, and country. We'll then insert that info into the database and use it to populate a map overlay when each point is clicked. Listing 2 contains the code to parse out these fields.

Another source of freely available geo-location data is the community-based hostip.info project located at http://www.hostip.info. Although this is a newer database, it's being actively and continuously updated and is improving with time. This database can also be queried via HTTP and returns the results in a format similar to NetGeo. A URL query to hostip.info would be structured as http://api.hostip.info/get_html.php?ip=64.19.164.74&position=true.

You can also download the full hostip.info database directly to your server for quicker response time. There's also the commercial GeoIP City database maintained by MaxMind (see http://www.maxmind.com/app/city).

Drawing the Map Using Google

The first thing you'll need to do is to sign up for a Google Maps API key at http://www.google.com/apis/maps/. This key will have to be included on any pages on which you'll be drawing maps.

To begin, put the script calls in Listing 3 in the <head></head> section of your page.

Next, include an empty div right after your <body> tag. This will contain the actual map and determine its size:

```
<div id="map" style="width: 700px; height: 500px"></div>
```

Using Yahoo!'s New Flex-Based API

You'll also have to get an application ID from Yahoo! before using its service. Once that's done, you'll put script calls in Listing 4 in the <head></head> section of your page.

Then include an empty div right after your <body> tag. This

div will contain the actual map:

```
<div id="mapContainer" style="width: 700px; height: 500px"></div>
```

Microsoft's Virtual Earth Implementation

Interfacing with Microsoft's Virtual Earth is also relatively painless. Most of your code will go in the <head></head> section of the page. The code you'll need to include is in Listing 5. Then you'll have to add an onLoad call to your body tag as follows: <body onLoad="OnPageLoad()">.

As with most of Microsoft's technologies, Virtual Earth will probably take a few revisions before they get it right. Currently there are advantages to using the Google or Yahoo! APIs. Google offers simplicity and the capability to add detailed HTML overlays complete with images. Yahoo!'s advantage comes in its rich integration possibilities with Flash-based technologies as well as AJAX.

Troubleshooting

One thing to note is that some browsers will throw a security error when trying to access a JavaScript file from a different domain. You can get around this by downloading the .js file to your server and serving it up locally.

Taking It Even Further

The Yahoo! map that we built, though Flash-based, used the JavaScript API to interact with the map. There's also an ActionScript API available for direct integration with Flash as well as a Flex API to interact with maps directly from your RIAs. This comes at a great time with Macromedia's recent announcements regarding the Flex product line. Check this link under resources for a great example of what can be done with the Flash API: http://tinyurl.com/avpv8 (from Justin's Rich Media Blog). The APIs will also accept geo-coded RSS feeds. This lets you fuse live data right on to your maps, such as being able to plot apartments currently listed for rent through craigslist.

Conclusion

We've only scratched the surface of what can be done using the new mapping APIs that are now available. It's possible to create all kinds of applications based on geographic data. Although the licensing terms for each API are slightly different, they are generally pretty liberal and even allow for commercial use in some cases. Visit the links provided for full documentation on each of them.

Resources
- Google Maps API – http://www.google.com/apis/maps/documentation/
- Yahoo! Maps API – http://developer.yahoo.net/maps/
- MS Virtual Earth Developer Center – http://www.viavirtualearth.com/
- NetGeo Public Database – http://www.caida.org/tools/utilities/netgeo/

Welcome to the Future of Video on the Web!

LIVE SIMULCAST!
AROUND THE WORLD ON SYS-CON.TV

iTVCON.com
INTERNET TV CONFERENCE & EXPO 2006

Coming in 2006 to New York City!

> "Internet TV is wide open, it is global, and in true 'Web 2.0' spirit it is a direct-to-consumer opportunity!"

For More Information, Call 201-802-3023 or Email itvcon@sys-con.com

Welcome to the Future!

Did you already purchase your ".tv" domain name?
You can't afford not to add Internet TV to your Website in 2006!

2005 was the year of streaming video and the birth of **Internet TV**, the long-awaited convergence of television and the Internet. Now that broadband is available to more than 100 million households worldwide, every corporate Website and every media company must now provide video content to remain competitive, not to mention live and interactive video Webinars and on-demand Webcasts.

20 years ago the advent of desktop publishing tools opened the doors for the creation of some of today's well-known traditional print media companies as well as revolutionized corporate print communications. Today, with maturing digital video production, the advent of fully featured PVRs, and significant advances in streaming video technologies, **Internet TV** is here to stay and grow and will be a critical part of every Website and every business in the years to come.

It will also very rapidly become a huge challenge to network and cable television stations: **Internet TV** is about to change forever the $300BN television industry, too.

The Internet killed most of print media (even though many publishers don't realize it yet), Google killed traditional advertising models, and **Internet TV** will revolutionize television the way we watch it today. You need to be part of this change!

Jeremy Geelan
Conference Chair, iTVCon.com
jeremy@sys-con.com

PRODUCED BY
SYS-CON EVENTS

List of Topics:

> Advertising Models for Video-on-demand (VOD)
> Internet TV Commercials
> Mastering Adobe Flash Video
> How to Harness Open Media Formats (DVB, etc)
> Multicasting
> Extending Internet TV to Windows CE-based Devices
> Live Polling During Webcasts
> Video Press Releases
> Pay-Per-View
> Screencasting
> Video Search & Search Optimization
> Syndication of Video Assets
> V-Blogs & Videoblogging
> Choosing Your PVR
> Product Placement in Video Content
> UK Perspective: BBC's "Dirac Project"
> Case Study: SuperSun, Hong Kong

Track 1	Corporate marketing, advertising, product and brand managers
Track 2	Software programmers, developers, Website owners and operators
Track 3	Advertising agencies, advertisers and video content producers
Track 4	Print and online content providers, representatives from traditional media companies, print and online magazine and newspaper publishers, network and cable television business managers

- Hostip.info Community Database – http://www.hostip.info/
- MaxMind Commercial Database – http://www.maxmind.com/app/city/
- Locate Geographic Points – http://geonames.usgs.gov/pls/gnispublic
- Cool Google Maps – http://coolgooglemaps.blogspot.com/.

About the Author

Joe Danziger is the founder and president of DJCentral.com, an online promotional tool for disc jockeys and other members of the electronic dance music industry. He has been developing professional ColdFusion solutions for over six years since version 1.5.

danziger@yahoo.com

Listing 1:

```
Setting Up Your Database Table
CREATE TABLE login_record
(IPADDRESS varchar (15),
     LATITUDE varchar (10),
     LONGITUDE varchar (10),
     CITY varchar (50),
     STATE varchar (30),
     COUNTRY varchar (50))
```

Listing 2:

```
<cfscript>
citystart  = find('CITY: ',cfhttp.filecontent)+6;
cityend    = find('<br>',cfhttp.filecontent,citystart);
city       = mid(cfhttp.filecontent,citystart,cityend-citystart);
statestart  = find('STATE: ',cfhttp.filecontent)+7;
stateend    = find('<br>',cfhttp.filecontent,statestart);
state  = mid(cfhttp.filecontent,statestart,stateend-statestart);
countrystart = find('COUNTRY: ',cfhttp.filecontent)+9;
countryend     = find('<br>',cfhttp.filecontent,countrystart);
country = mid(cfhttp.filecontent,countrystart,countryend-countrystart);
latstart  = find('LAT: ',cfhttp.filecontent)+4;
latend = find('<br>',cfhttp.filecontent,latstart);
lat       = mid(cfhttp.filecontent,latstart,latend-latstart);
longstart  = find('LONG: ',cfhttp.filecontent)+5;
longend    = find('<br>',cfhttp.filecontent,longstart);
long       = mid(cfhttp.filecontent,longstart,longend-longstart);
</cfscript>
```

Listing 3:

```
// include .js file for Mapping API
<script src="http://maps.google.com/maps?file=api&v=1&key=YOUR_API_KEY"
type="text/javascript"></script>

// create map and add overlays
<script type="text/javascript">
  //<![CDATA[
  var map = new GMap(document.getElementById("map"));
  // adds control for panning/zooming
  map.addControl(new GLargeMapControl());
  // adds control for switching between Map & Satellite Mode
  map.addControl(new GMapTypeControl());
  // draw map centered at this point - last param is zoom level [1
(street level) to 17]
  map.centerAndZoom(new GPoint(-74.04, 40.32), 15);

  <cfloop query="getVisitLog">
  <!--- WE'LL USE #CURRENTROW# TO MAKE SURE EACH MARKER IS UNIQUE --->
  var point = new GPoint(#trim(long)#,#trim(lat)#);
  var marker#currentRow# = new GMarker(point);
  // set HTML for popup balloon
  var html#currentRow# = "#city#, #state# #country#";
  // adds popup event to marker
  GEvent.addListener(marker#currentRow#, "click", function() {
      marker#currentRow#.openInfoWindowHtml(html#currentRow#); });
  // plot marker on map
  map.addOverlay(marker#currentRow#);
  </cfloop>
  //]]>
```

```
</script>
```

Listing 4

```
// include .js file for Mapping API
<script type="text/javascript" src="http://api.maps.yahoo.com/v2.0/fl/
javascript/apiloader.js"></script>

// create map and add overlays
<script type="text/javascript">
  // Create a latitude/longitude object to center map at Kansas City,
MO
  var latlon = new LatLon(39.12, -94.55);
  // Display map centered on location with zoom level 3. Include ap-
plication ID.
  var map = new Map("mapContainer", "YOUR_APPLICATION_ID", latlon,
15);
  // Make the map draggable
  map.addTool( new PanTool(), true );

  <cfloop query="getVisitLog">
  <!--- WE'LL USE #CURRENTROW# TO MAKE SURE EACH MARKER IS UNIQUE --->
  // Create a POI marker object
  var marker#currentRow# = new CustomPOIMarker( 'ICON TEXT', 'EXPANDED
TEXT',
          'POPUP BALLOON TEXT/HTML', '0xFF0000', '0xFFFFFF' );
  // create a latitude/longitude object for the current location
  var latlon#currentRow# = new LatLon(#trim(lat)#, #trim(long)#);
  // Add the POI marker to the map and display it
  map.addMarkerByLatLon(marker#currentRow#, latlon#currentRow#);
  </cfloop>
</script>
```

Listing 5

```
// define the CSS for the map markers
<style type="text/css" media=screen>
 .marker { width:120px;height:17px; font:bold 8pt Arial,sans-serif;
         color:White;text-align:center;background:#0000FF;z-in-
dex:5; }
</style>

// include the JavaScript file for Mapping Control
<script src="http://dev.virtualearth.net/standard/v1/MapControl.
js"></script>

// create map and add overlays
<script>
   var map = null;
   function OnPageLoad() {
     map = new VE_MapControl(39.12, -94.55, 4, 'r', "absolute", 10,
10, 700, 500);
       <cfloop query="getVisitLog">
       <!--- WE'LL USE #CURRENTROW# TO MAKE SURE EACH MARKER IS
UNIQUE --->
       map.AddPushpin('marker#currentRow#',#trim(lat)#,#trim(long)#,
10,10,
                    'marker','#CITY#, #STATE#');
     </cfloop>
     document.body.appendChild(map.element);
   }
</script>
```

Other companies in this magazine spent a lot of time on pretty ads. As you can see, we did not. We spent our time hiring the best people and training them to deliver outstanding support for your website. We spent our time building a state of the art datacenter and staffing it with people who care about your website like it's their own. Compassion, respect, credibility, ownership, reliability, "never say no," and exceed expectations are words that describe our service philosophy. From the first time you interact with us, you'll see what a difference it really makes. And you'll also forgive us for not having a pretty ad.

HostMySite.com

WEB HOSTING • MANAGED DEDICATED SERVERS • COLOCATION • VPS • ECOMMERCE • BLOGGING • EMAIL