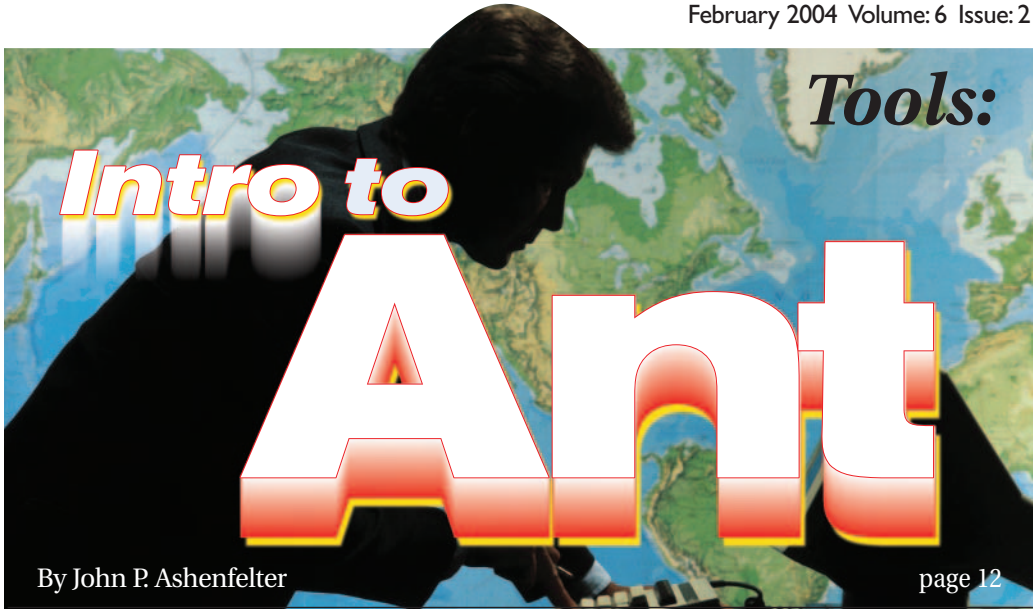
U.S. $8.99 (Canada $9.99)

ColdFusion Developer's Journal

ColdFusionJournal.com

February 2004 Volume: 6 Issue: 2



By John P. Ashenfelter page 12

For answers call us at 1-866-EDGEWEB

3 3 4 3 9 3 2

When calling your web host for support you want answers, not an annoying song stuck in your head from spending all day on hold. At EdgeWebHosting.net, we'll answer your call in two rings or less. There's no annoying on-hold music, no recorded messages or confusing menu merry-go-rounds. And when you call, one of our qualified experts will have the answers you're looking for. Not that you'll need to call us often since our self-healing servers virtually eliminate the potential for problems and automatically resolve most CF, IIS and ASP problems in 60 seconds or less with no human interaction. Plus, our multi-user support system allows you to track support requests for each of your engineers individually, lookup server availability, receive a copy of all errors on your site in real time, and even monitor intrusion attempts on your site in real time. **For a new kind of easy listening, talk to EdgeWebHosting.net**

By the Numbers:

- 2 Rings or less, live support
- 100% Guarantee
- 99.998% Uptime
- 150 MBPS Fiber Connectivity
- 24 x 7 Emergency support
- 24 Hour free backup

EDGE WEB HOSTING

What are you WAITING for?

www.edgewebhosting.net

Win a year of free hosting*

*On Shared Hosting or the equivalent value
See http://edgewebhosting.net/cfdj for details

Shared Hosting ¥ Managed Dedicated Servers ¥ Semi-Private Servers
ColdFusion ¥ SQL Server ¥ .NET ¥ Self-Healing Servers ¥ Value Priced

It's everybody's PDF™

Finally, a software company that offers affordabe yet flexible PDF solutions to meet every customer's needs. Using activePDF™ to automate the PDF creation process eliminates the need for end-user intervention so your employees can concentrate on what they do best.

Licensed per server, activePDF solutions include a COM interface for easy integration with ColdFusion. Dynamically populate PDF forms with information from a database, convert ColdFusion web pages to PDF on the fly, dynamically print reports to PDF using CF and much more. Users can also merge, stitch, stamp, secure and linearize PDF, all at a fraction of the cost of comparable solutions. Download your free trial version today!

activePDF®
Leading the iPaper Revolution

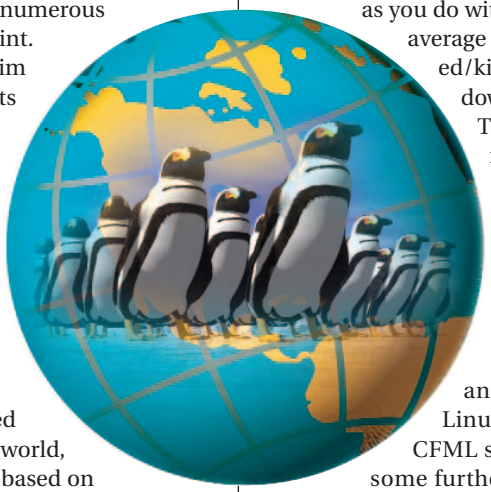
www.activePDF.com

Entering the World of Linux

By Robert Diamond

I'm writing this month's editorial from the LinuxWorld Conference and Expo in a currently frozen New York City, so it's naturally got me in a "Linux Frame of Mind." When talking to many CFMLers at the Macromedia MAX Conference this year, I found a mixed bag of opinions about Linux and a lot of fears – some justifiable, and others not. Let's jump in…

In these cost-cutting, get-more-for-spending-less economic times, everyone is looking for the maximum savings and return on investment. It's often debated whether the cost of ownership of Linux versus Windows is cheaper or not, and I could quote numerous studies to prove either point. Microsoft backers will claim that while the startup costs for Linux are less, the maintenance, consulting, and long-term costs will be higher. The Linux folks will claim the reverse – that you spend more for Windows, and continue to rack up higher bills as long as you stick with it. Like most of the associated costs in the *i*-Technology world, it's something that'll vary based on your needs, so I'd urge you to evaluate all your available options on a case-by-case basis.



We've got a combination of servers running at SYS-CON, and I can say personally that the largest leap for me to come around to using Linux was most certainly the learning curve involved. A Windows boy since 3.0, I had to jump head-on into the world of Red Hat. With some books on the subject and a number of online resources I printed off, several trees contributed to my education and security blankets as well.

Configuring an operating system and server software primarily through a text-based interface certainly wasn't my idea of fun at first, but as a natural "tweaker" I began to appreciate the flexible configurability of the whole system. You can spend hours, days, and weeks delving into as many (or as few) Linux configurability options as you'd like, and the possibilities are endless as to what it will allow you to do. There's also no shortage of free software available thanks to the open-source community.

Linux also showed some speed and performance gains on the server as a whole, compared to its Windows counterpart, thanks to (primarily) the overhead OS footprint being smaller and freeing up more resources for the server software. Uptime-wise, we had just as many startup bumps with Linux as you do with Windows, but on average the boxes are rebooted/kicked/thrown out a window 1/2 to 1/3 as often. That's certainly resulting in a slight decrease of sleepless nights – a welcome lifestyle improvement for any of us who run and admin our own servers.

Both Macromedia and New Atlanta offer Linux versions of their CFML servers. You can do some further reading about both of these at their respective sites… There are always lots of options out there, and I wish you luck in finding the best for you, your organization, and your project.

About the Author

Robert Diamond is vice president of information systems for SYS-CON Media, and editor-in-chief of **ColdFusion Developer's Journal.** *Named one of the "Top thirty magazine industry executives under the age of 30" in* Folio *magazine's November 2000 issue, Robert holds a BS degree in information management and technology from the School of Information Studies at Syracuse University. Visit his blog at* www.robertdiamond.com.
robert@sys-con.com

FREE TRIAL

ON VIRTUAL DEDICATED HOSTING*

FEATURES

- **ColdFusion Administrator Access**
  Receive full access to the ColdFusion Administrator!
- **Remote Management of Server**
  Use Terminal Server to manage your server at the root level.
- **Custom Tag Support**
  Get unrestricted tag use and ColdFusion RDS support.
- **Support for Multiple Websites**
  Host up to 50 domains. The first 10 are free.
- **Unlimited Database Support**
  Includes 100MB SQL space and unlimited DSNs.

WHY SHOULD YOU SWITCH?

- **MORE FLEXIBILITY**
  You can install custom tags and software.
- **MORE RELIABILITY**
  Your site is not affected by others on the server.
- **COST SAVINGS**
  Save by consolidating your domains into one plan. Get 5GB space for up to 50 domains.

macromedia ALLIANCE PARTNER

CALL US TODAY TO ACTIVATE YOUR FREE TRIAL!

No obligation to buy!

866-CFX-HOST

www.cfxhosting.com/vpstrial

ColdFusion Developer's Journal CFDJ READERS' CHOICE AWARD

Our servers are housed in a fully-redundant data center, monitored 24/7 by experienced technicians.

Supported Software
Latest version of CFMX (v 6.1)
Windows 2003 or 2000
Access and SQL supported

CFXHosting

www.cfxhosting.com **866-CFX-HOST**

* Free setup & first month free. Just $189/month thereafter.

SYS-CON MEDIA

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

vp, business development
Grisha Davida grisha@sys-con.com
group publisher
Jeremy Geelan jeremy@sys-con.com
technical director
Alan Williamson alan@sys-con.com

advertising
senior vp, sales & marketing
Carmen Gonzalez carmen@sys-con.com
vp, sales & marketing
Miles Silverman miles@sys-con.com
advertising director
Robyn Forma robyn@sys-con.com
director of sales & marketing
Megan Mussa megan@sys-con.com
advertising sales managers
Alisa Catalano alisa@sys-con.com
Carrie Gebert carrieg@sys-con.com
associate sales managers
Kristin Kuhnle kristin@sys-con.com
Beth Jones beth@sys-con.com

sys-con events
president, events
Grisha Davida grisha@sys-con.com
conference manager
Lin Goetz lin@sys-con.com
national sales manager
Sean Raman raman@sys-con.com

customer relations
circulation service coordinators
Shelia Dickerson shelia@sys-con.com
Edna Earle Russell edna@sys-con.com
Linda Lipton linda@sys-con.com
manager, jdj store
Brunilda Staropoli bruni@sys-con.com

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

web designers
Stephen Kilmurray stephen@sys-con.com
Christopher Croce chris@sys-con.com
online editor
Lin Goetz lin@sys-con.com

accounting
financial analyst
Joan LaRose joan@sys-con.com
accounts receivable
Charlotte Lopez charlotte@sys-con.com
accounts payable
Betty White betty@sys-con.com

subscriptions
Subscribe@sys-con.com
1 888 303-5282
For subscriptions and requests for bulk orders, please send your letters to Subscription Department
Cover Price: $8.99/issue
Domestic: $89.99/yr (12 issues)
Canada/Mexico: $99.99/yr
all other countries $129.99/yr
(U.S. Banks or Money Orders)
Back issues: $12 U.S. $15 all others

Tales from the List

Are you content with *CFDJ* content?

ColdFusion Developer's Journal has been in print for quite a long time… in fact it has now celebrated its fifth birthday. I haven't read every article published in every issue, but I've read a lot of them. One thing I've learned over the years is that the more confident you are that you know everything there is to know, the more likely you are to be humbled by someone who proves you wrong.

By Simon Horwith

This applies to all things in life – ColdFusion development is certainly one of them. *CFDJ* has been a source of educational and thought-provoking material for me, time and time again. This month's **Tales from the List** is about *CFDJ* and its content. I chose to write about this subject as a result of two list server threads – one on the ***CFDJ-List*** and one on the CF-Talk list.

What I like most about *CFDJ* is the fact that its offerings are unique in the ColdFusion world. Online resources are great, but you have to be online to take advantage of them, they aren't always kept up to date, and they are often written from one point of view. They do, however, show up in search engines, can be linked to, and can link to other resources, etc. Printed material, on the other hand, is handy in that it does not require you to be online to use it – you can take it anywhere. You also don't have to search the Web for the resource – you simply open the book and read. In the case of books, one disadvantage is that they tend to become out of date (when a newer version of the software they describe is released) as quickly as they are published.

SYS-CON makes all of the *CFDJ* content available online (http://sys-con.com/coldfusion/) which allows them to offer all of the advantages of an online resource. It is however, a periodical... a printed resource. Unlike books, a new version is released every month so the content is kept up to date.

While only one author writes each article, many authors contribute to each issue. I've found that a lot can be learned by studying the way that different developers approach and solve different problems – something *CFDJ* is fairly unique in offering. Raymond Camden and I, the two technical editors, try to ensure that the content of each article is accurate and consistent with the others. Another thing that is unique to *CFDJ* is that unlike other resources, it is reflective of the community – news from Macromedia, interviews, advertisements, community, and events reporting, etc. – all in one monthly dose of CF goodness.

The point of this article isn't to go on and on about the advantages of magazines over books and Web sites, nor is its purpose to illustrate how *CFDJ* succeeds where other resources fail. After all, as a *CFDJ* reader you probably already know all this. The goal of this article is in fact, exactly the opposite. *CFDJ*, as great as it is, could be better… but you're going to have to help.

I say that *CFDJ* could be better because there are two things that contribute to its success and there can never be too much of either one. The first thing that is required to make *CFDJ* the best that it can be is feedback and suggestions from the community.

—continued on page 17

About the Author

Simon Horwith is co-technical editor of **CFDJ**, *and chief technology officer of eTRILOGY Ltd., a software development company based in London, England. Simon has been using ColdFusion since version 1.5 and is a member of Team Macromedia. He is a Macromedia Certified Advanced ColdFusion and Flash developer and is a Macromedia Certified instructor. In addition to administering the* CFDJ List *mail list and presenting at CFUGs and conferences around the world, he has also been a contributing author of several books and technical papers.*
simon@horwith.com

The Case Against Coding for Portability

DATABASES ARE DIFFERENT

By Ben Forta

Conventional wisdom dictates that code, all code, be written with portability in mind. After all, you wouldn't want to have to revisit and rewrite code when moving between platforms or environments, would you? And while I do believe that coding for portability is a good thing in general, I also believe that when it comes to databases and SQL, coding for portability is a very bad thing indeed.

Putting It in Context

In the past few weeks I was involved in two long e-mail threads:

- One involved a discussion about the generation of primary keys and the pros and cons of using identity (or auto-number) fields versus generating primary-key values from within ColdFusion. I argued that I did not want client code generating primary keys; that just does not make sense as each client would need to recreate that code.
- The second involved the use of SQL implementation–specific features, useful functions, or stored procedures supported by a single DBMS only (in this case it was SQL Server). I explained that I did not want to have to jump through hoops to recreate functionality that my DBMS already offered; that would be a waste of my time, and a waste of runtime resources too.

In both of these discussions I took the position that it was okay, and even preferable, to use DBMS-specific features. But the individuals at the other end of the threads argued portability. Allowing the DBMS to generate primary key values means that if the DBMS was replaced, those values might all need updating. Similarly, using SQL Server–specific functions means that were I to switch from SQL Server, my client (ColdFusion) code would need updating.

And that's a compelling argument, isn't it?

Switching DBMSs Is Not the Norm

When was the last time you switched the DBMS that powered your production application? Okay, let me ask it differently: In all the years you have been writing ColdFusion applications, how many times have any of those applications been migrated between DBMSs?

I think that most of you will answer rarely, or at worst, infrequently. The fact is that few of us write apps for SQL Server and then need to port them to Oracle (one of the most difficult ports). True, many have upsized from Access to SQL Server, but that is a much cleaner and simpler port (and Microsoft actually provides wizards that can help the process somewhat).

I often see users selecting different DBMSs for new projects, but I rarely see users switch DBMSs in existing projects. Not that it doesn't happen, it does, but rarely.

And as such, does it really make sense to impose all that extra work and processing involved in creating portable SQL just in case portability becomes an issue at some point? I'd venture that the answer to that is no.

Of course, there is one exception to this. If you were to write an application that needed to be used with multiple DBMSs (commercial software, or applications distributed to other users) then portability is an obvious immediate concern. But even in that scenario I'd argue against tying your hands behind your back.

DBMS Portability Is Costly

DBMSs are big powerful applications, capable of performing complex processing and data manipulation. DBMSs are created by massive development teams, with resources far greater than you'll likely ever have. These big applications built by big teams do one thing and one thing only, they manipulate data, and they do that very well.

If you opt to not use specific DBMS functionality then you will necessarily have to do more work yourself in your client code. And regardless of how good a developer you are, the client code you write will never be as efficient as the DBMS functionality it is replacing. It can't be.

Sure, your application may be portable (emphasis on may), but is the extra cost both in development time and runtime performance worth it? I highly doubt it; my time is valuable, and runtime performance is critical.

True DBMS Portability Is Unattainable

Thus far I have implied that portable SQL is actually possible. But, being brutally honest, it isn't.

The key to writing portable SQL code is sticking to the lowest common denominator, just using the statements and syntax supported by all DBMSs. So:

One damn good Cold Fusion hosting firm!

Webcore Technologies specializes in ColdFusion, ASP and SQL hosting. We offer these solutions in both dedicated server and shared virtual environments. Webcore can design and implement clustered or load-balanced solutions, managed firewalls, VPN's, multi-site failover, and more. In addition, we also offer corporate Internet access, web site design, and technology consulting services.

Our in-house tier 1 level data center enables us to provide the most reliable hosting in the industry. Our Microsoft and Cisco certified team ensures that all support issues are resolved promptly and professionally. Unlike other hosting companies that answer with a phone attendant, you will receive a live person when you call Webcore. No phone attendant, no frustrations, just world class customer service and support the first time you call.

Cisco Systems

Microsoft Certified Partner

Webcore Technologies

Webcore Technologies, Inc.
877.WCT.HOST
www.webcoretech.com

- No triggers (they are not supported by all DBMSs)
- No stored procedures (they are not supported by Access and by most MySQL installations)
- No use of subqueries except in your newest code (as MySQL added support for them in v4.1)
- No unlimited use of DISTINCT in aggregate functions.
- No free use of wildcards (there are usage incompatibilities, and wildcards in the middle of a string behave differently in different DBMSs)
- No concatenation allowed at all (DB2, Oracle, and PostgreSQL use || while MySQL, SQL Server, and Sybase use +)
- No date arithmetic in your SQL, ever (no two DBMSs use the same syntax for that)
- No string conversions or manipulation (UPPER() or UCASE()? SUBSTR() or SUBSTRING() or MID()?)
- No using numbers other than whole integers in WHERE clauses (as PostgreSQL requires that these be cast using a syntax that other DBMSs do not support)
- No using aliases in ORDER BY clauses (Access does not allow this)
- No use of NOT in WHERE clauses except in NOT EXISTS (MySQL won't allow that)
- And the list goes on and on

In other words, you'd need to severely restrict yourself to a subset of DBMS functionality. You'd need to do just the absolute basics in your SQL code, and your client code would have to do the rest. Which goes back to the previous point, recreating DBMS functionality in client code is too costly.

But it gets worse. There are some operations that you'll never be able to do if you insist on portable code:

- There is no portable way to obtain lists of tables (sp_tables in SQL Server, SHOW TABLES in MySQL, and other DBMSs have other ways to accomplish this).
- There is no portable way to get the current date or time (NOW() in Access, GETDATE() in SQL Server, CURRENT_DATE in PostgreSQL CURRENT_DATE() in MySQL, and so on).
- There is no way to execute multiple statements in a batch (most DBMSs separate statements using ";" but Sybase does not like the ";" there at all).
- And this list goes on and on too.

So, not only is writing truly portable code too costly, when it comes down to the details it is actually impossible! Sure, for simple applications you may get away with it, but as application complexity grows, so does the likelihood that you'll be writing some proprietary code.

Database Encapsulation

Does this mean that we're doomed? I don't think so. As already stated, DBMS portability is seldom a practical concern except for code that's intended to support multiple DBMSs.

Does this mean that portability should be ignored? Absolutely not. You should definitely be coding to address portability when (or if) it becomes an issue.

So how to balance the goals? The answer is encapsulation. Use all the DBMS-specific functionality you want, but hide that from client code. In ColdFusion apps this can be accomplished using ColdFusion Components (CFCs); all DBMS code goes into CFC methods, and client code simply invokes those methods. If you ever need to change your DBMS you'll need to change the code inside of that CFC, but your client code will continue to work as is.

And what if you need to support multiple DBMSs? That too is a job for CFCs. Create multiple versions, use different methods, leverage inheritance, and create a base CFC and then extended versions for different DBMSs…

There are lots of ways to set it up. The key is that anything proprietary (and indeed anything pertaining to SQL or DBMSs at all) is hidden from client code. And within that hidden code you are free to take advantage of all your DBMS has to offer.

Should primary key autogeneration be used? Here's a quote from my response to that thread mentioned earlier: "Using CreateUUID() to generate primary keys is inherently dangerous. You're making the assumption that ColdFusion will forever be the only client using this code, and that seems rather shortsighted. The first time I'll need to create a row using anything other than CFML code I'll be up the proverbial creek. You have a CFC method that handles record creation; make it SQL Server specific. To support Oracle and MySQL simply create new versions of the CFC inherited from the same base CFC. Sure, this will mean that I need to pick which CFC I want to use, but I have no problem putting that in some configuration script. The truth is, you could expose yet another CFC that takes a DBMS to use as a parameter, and then you'd instantiate the correct CFC internally. Or not. I don't care how you do it, just do it.

"Use all the DBMS-specific functionality you want, but hide that from client code"

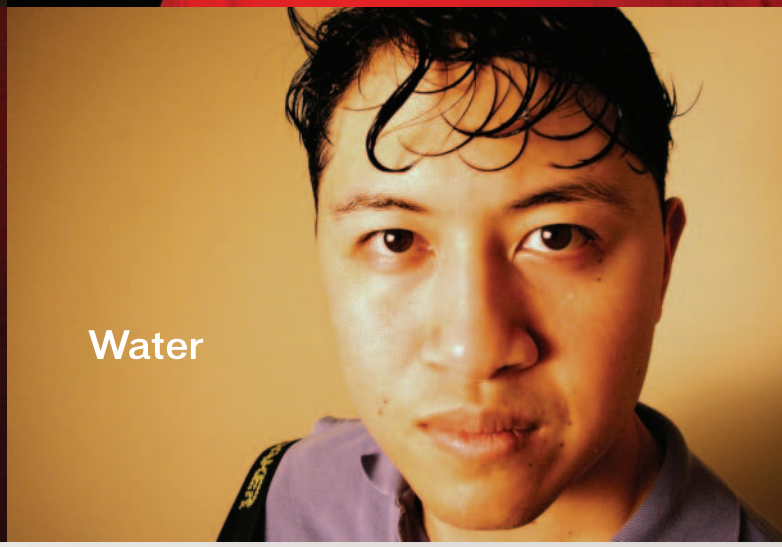
Conclusion

You've invested in a DBMS, you might as well take advantage of it. I am strongly opposed to making ColdFusion (and you) work harder than necessary just to avoid anything DBMS specific, especially as ultimately this will only be an exercise in futility. There are enough real challenges for us to address without tying our own hands behind our backs by imposing unnecessary restrictions on ourselves. Use any DBMS functionality you want, push your DBMS to its limits, and hide all that from client code. With a bit of planning and forethought you can have your cake and eat it too.

About the Author

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

ben@forta.com

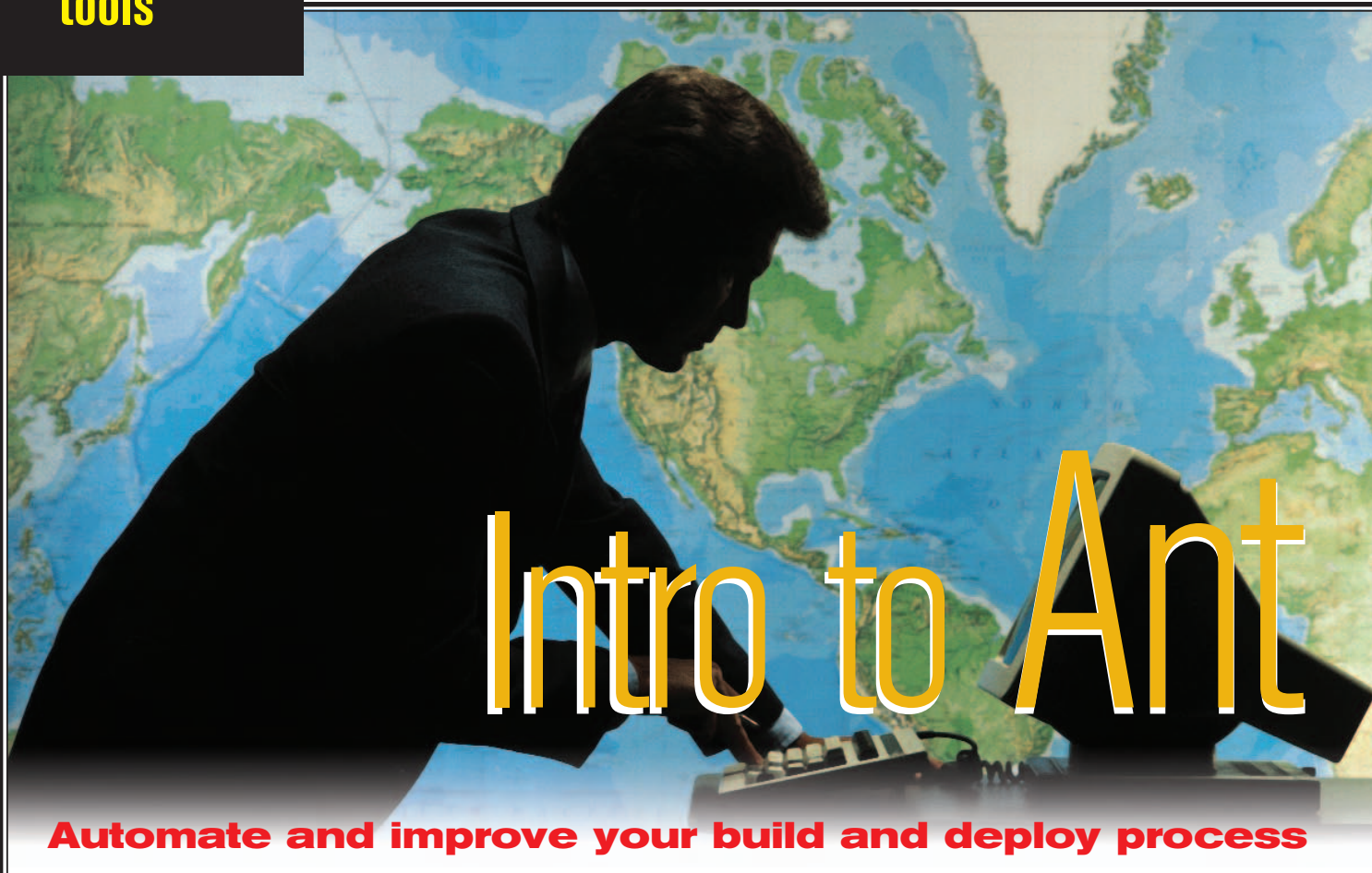


INTO

WHAT ARE YOU INTO?

At Macromedia, we're continually inspired by the passion of our customers. Visit "Into" to see their stories, or share your own. **www.macromedia.com/into**

Announcing Macromedia MX 2004:
New versions of Macromedia® Flash™, Dreamweaver®, Fireworks® and Studio. Run with it.



Intro to Ant

Automate and improve your build and deploy process

Writing shell scripts to automate the build and deploy process for ColdFusion applications is not very much fun. The Jakarta Ant project is an open-source, cross-platform alternative that makes it easy to automate the build and deploy process.

But My Build and Deploy Process is Fine....

Maybe *your* build and deploy process for your latest application is fine – you type a single command and your build process automatically retrieves your application from the source control system, configures the application appropriately for the target environment, and copies all the necessary files to the production servers while you head to the coffee shop for your morning cup of caffeine and the newspaper. But I know that the reality for the vast majority of projects *I've* seen (including many of my own applications!) are built and deployed using a written multistep checklist – some steps automated by simple shell scripts and some done by hand. With time a scarce commodity on most projects, it's not a surprise that anything other than the ColdFusion application itself gets little, if any, attention.

But how many times have you personally been burned by a bad build or deploy? Maybe forgetting to copy a custom tag to the \CFusionMX\CustomTags directory? Or deploying the wrong version of a ColdFusion template? Maybe you forgot to toggle the data source name from the development server to the production server? Or neglected to disable some debugging code? The list goes on. How much time did you waste finding and fixing the problem? Odds are it was a lot, and that the problem happened at the least opportune time, like during a major production release! If there was a simple, free, cross-platform, extensible tool that would let you write automated build and deploy scripts, wouldn't it make sense to use it?

By John P. Ashenfelter

There are some traditional tools for automating builds: the venerable **make** is familiar to anyone who's done much work on Linux or with C applications; **jam** may be familiar to some hardcore open-source developers; and native shells on Windows (DOS and third-party replacements including cygwin) and Unix (bash, csh, zsh, and their relatives) can all be used. But anyone who has used **make** or its derivatives can relate to the frustrations of accidentally putting a space in front of a tab and trying to figure out why the script is not working. And shell scripts are both platform specific and can be very limited in their capabilities (particularly the DOS shell).

One Java programmer, James Duncan Davidson, became so frustrated with the existing build tools while he was creating Tomcat (the official reference implementation for Java Servlets and JavaServer Pages) that he wrote his own build tool, which he dubbed Another Neat Tool (Ant). The Ant tool itself is implemented in Java, which allows it to run consis-

tently across most modern operating systems, and also makes it easy to extend so that it can do any tasks that Java can (more on that later). Ant-built scripts are written in XML, which makes them much easier to write (and validate!). Ant rapidly expanded into a number of Apache Jakarta projects and from there to the many Java developers who downloaded the Jakarta tools. Today, Ant is a de facto standard tool for automating many aspects of the Java software development process. Instead of the "make install" of the C world, Java developers now have "ant deploy"!

So why should you, as a ColdFusion developer, be interested in Ant? If your ColdFusion applications deploy seamlessly to Linux and Windows servers with a single command, then you might not need Ant, but otherwise it's going to become a key tool in your software development toolbox. More important, now that ColdFusion MX is so tightly coupled to the Java world, using Java tools simply makes sense. Even the ColdFusion MX updaters use Ant scripts for part of the upgrade process!

Installing Ant

The Apache Ant project home page http://ant.apache.org/ provides both source and binary distributions of Ant, as well as the manual, links to key resources, and even a Wiki. For most purposes, the binary distribution for your platform should be sufficient, especially if you're new to Ant. The latest Ant release, 1.6.0, debuted in mid-December of 2003, though there's nothing wrong with Ant 1.5.x should that version already be available in your environment. And since Ant is a Java application, you'll also need an installed JDK, version 1.2 or later – the latest Sun JDK 1.4 is available from http://java.sun.com/j2se/1.4.2/download.html.

Installing the Ant binaries is simply a matter of unzipping the archive into a directory on your computer and completing two configuration steps:

1. The Ant\bin directory needs to be added to your path.
2. Add the environment variable ANT_HOME, and have it point to the installation directory.

There is also an additional, optional, step:

3. Add the environment variable JAVA_HOME, which points to the installed JDK directory.

So on a Windows 2000/XP computer, the configuration might look like Listing 1. There are bash and csh equivalents for Unix in the Ant reference manual (http://ant.apache.org/manual/index.html).

You can test your installation by opening a console window and typing

```
ant -version
```

which should produce a version number and compile date for your Ant installation.

Using Ant

Now that you've got Ant installed, what's the next step? Why, a build file of course! You're going to need a console window and your favorite text (or XML) editor to get started. The Ant command expects a buildfile (which defaults to build.xml) that contains an XML description of build *targets*, which are steps in the build process. And each target consists of one or more *tasks* that need to occur. Listing 2 contains a very simple buildfile. If you save Listing 2 as build.xml and simply type "ant", you'll get something that looks like this:

```
Buildfile: build.xml
help:
     [echo] usage: ant [target]
     [echo] typical targets: init, dist, deploy, help
BUILD SUCCESSFUL
Total time: 4 seconds
```

It's worth noting that since we didn't pass in any particular target, Ant used the default target name defined in the <project> tag. Typing "ant help" will produce the same output. In this buildfile, we've got a single <target> that has two <echo> tasks. The <echo> task simply writes text to the screen.

Now that we've got the Ant equivalent of a "Hello World" application, let's get down to the business of creating the build process for a real ColdFusion application using Ant. To get started all we need to do is retrieve the source code from the source code control repository (I'm going to use Microsoft Visual SourceSafe for this example) and prepare it for deployment. I'm going to use the following Ant targets for our build process:

1. ***init:*** Create the directories for the project
2. ***getSource:*** Pull the latest source code from the repository
3. ***build:*** Process the source code to prepare it for the deployment environment
4. ***dist:*** Package the source for distribution/deployment
5. ***deploy:*** Copy the distribution package to the production server
6. ***clean:*** Remove all files and directories associated with the project

The entire buildfile is shown in Listing 3, and it's certainly expanded a lot!

Putting Our Buildfile Under the Magnifying Glass

There are a lot of new things to absorb in this buildfile and we have limited space, so we're going to look at each target individually in the following section and go over the major points. If you want details on any of the specific tasks (or any of the dozens of other Ant tasks) you might want to peruse the Ant online manual (http://ant.apache.org/manual/index.html) as you read along.

The init target shouldn't be too hard to interpret – it uses the <mkdir> task to create three directories: *src, build,* and *dist*. This task will create each of these three directories if they don't already exist. As an aside, these specific directory names have become conventional on Ant projects. Type "ant init" and the three directories will be created to hold your application. Since the <mkdir> in our buildfile is using relative paths, the directories will be

"Today, Ant is a de facto standard tool for automating many aspects of the Java software development process"

created underneath the directory where the buildfile is located.

The *clean* target looks very similar to the *init* target, with the <delete> task replacing the <mkdir>. As you probably intuited, we're simply deleting the directories we created with the *init*. But one new feature is the *depends* attribute in the <target> tag. The *depends* attribute can be used to describe dependencies between targets. In this case, the buildfile is indicating that the *clean* target can run only after the *init* target has been run, so Ant automatically runs *init* for you, but *only if the target needs to be run*! You can test this by running the *clean* target two or more times – if the directories exist (and thus the init dependency is satisfied), the *clean tasks* simply deletes them, but if the directories do not exist, Ant runs the dependent *init* target to create them and then deletes them. Try it and watch the output!

The *getsource* target is where things really get interesting. We want to get the source code out of our Microsoft Visual SourceSafe (VSS) repository so we can build and deploy it. Ant has built-in tasks for manipulating all of the major (and many minor) source control applications. (As an aside, there are actually eight Ant tasks for manipulating VSS; these and all of the other vendor-specific tasks are listed in the "Optional Tasks" section of the Ant manual.) In this example, we're using <vssget> to pull code from the VSS. The buildfile is passing in a number of VSS-specific parameters which result in the latest source code being pulled from the repository and put into the \src directory. Note that this target also depends on the *init* target, since the \src directory needs to exist before we start getting the source code.

The *build* target is pretty simple in this buildfile – it uses the <copy> task to copy all of the files in the \src directory to the \build directory. The <copy> task is interesting because it uses an Ant type, the <fileset>. There are over a dozen Ant types, which represent complex data structures that Ant natively understands. In this case we're defining a set of files with the <fileset> type, which is then used by the <copy> task.

> **"If there was a simple, free, cross-platform, extensible tool that would let you write automated build and deploy scripts, wouldn't it make sense to use it?"**

The *build* target depends on the getsource task, since we need source code before we can copy it to another directory, since getsource depends on *init*, we've now used the depends attribute to chain three targets together in the proper order, which you can verify by running the *build* target and watching the dependent tasks execute in the Ant output.

The dist target is exactly like the *build* target, simply copying files from one directory to another. If we were using Ant for a Java application, this is the step where the compiled class files would be packaged into a JAR file for distribution. So the net result of all of these steps so far is to pull out the source code from VSS, copy it to the \build directory, and then copy it to the \dist directory. We could have skipped the *build* and *dist* targets, but we'll talk more about why they're useful in the final section of this article.

Finally, we've got the deploy target, which in this case is copying the files in the \dist directory to another directory, perhaps a network share to the integration or production server. We're using the <mkdir> and <copy> tasks again, but this time the directory name looks a little weird. One powerful concept in Ant is the "property". In this example, the deploy target expects a property called "deploy.dir", which represents the directory the \dist directory should be deployed to. The syntax for a property is ${*varname*}. Properties can be configured a number of ways using the <property> task, ranging from assigning it directly in the buildfile to using environment variables or the command line. Let's assume for a minute that I can deploy the files to two directories, c:\apache2\htdocs\test for my local test Web server, and l:\apache2\htdocs\production for the production Web server. The property can be input from the command line using the –Dproperty=value syntax, like the following example

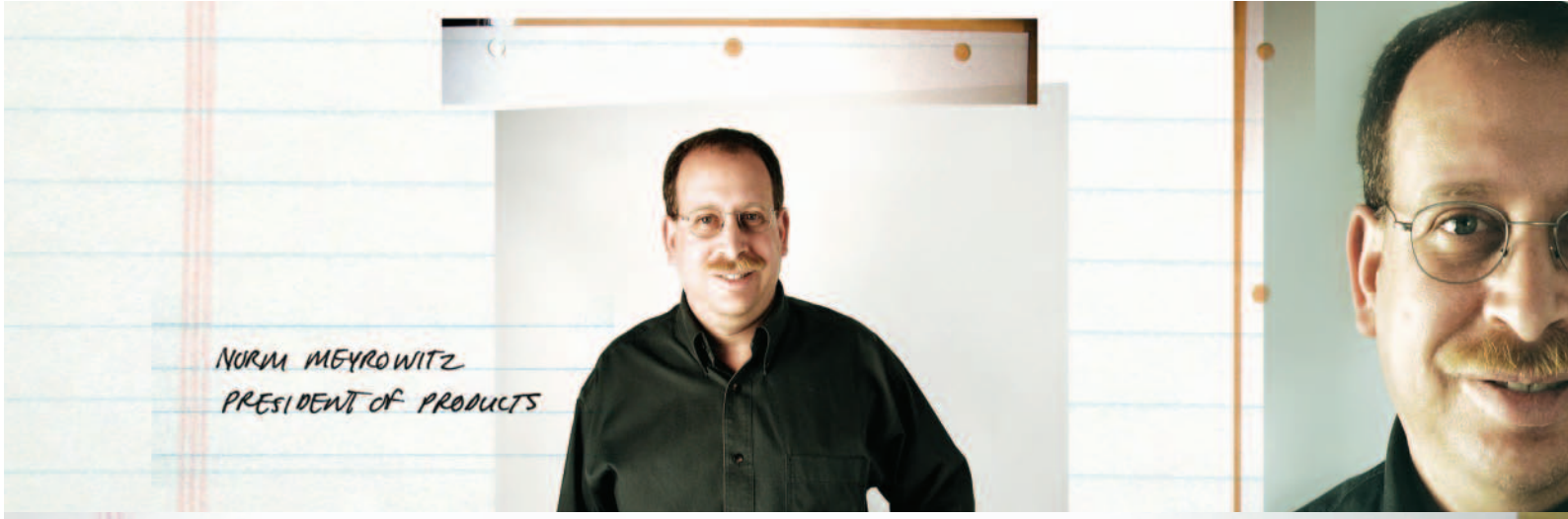
```
ant deploy -Ddeploy.dir=c:\\apache2\\htdocs\\test
```

Note that there is no space between the –D option and the property=value information. And since I'm running on a Windows machine, I do have to remember to escape the backslash just like I would have to in a Java .properties file. But the buildfile doesn't know anything about Windows, Linux, Mac, or anything else, so you can deploy to /var/httpd/htdocs/test just as easily.

Next Steps

The first step I'd recommend taking is to automate your build process using Ant if you don't already have a solution. Now. Right now. Go! And if you do have a *build* process in place, I'd suggest taking a hard look at whether it's worth moving to Ant for its simplicity (relatively speaking), extensibility, and cross-platform support. We've literally just scratched the surface of Ant, but this buildfile is already useful and a great starting point for experimentation (and maybe another article? Let me know!). Just some quick tidbits to whet your appetite for more Ant:

- Use <filter>s to replace tokens in files with new values, maybe to toggle debugging flags and set datasource names that vary between deployment environments.
- Schedule Ant using cron, at, Windows scheduler, or the like for nightly builds, and send e-mail to the team about the results.
- Integrate automated testing tools (JUnit, HTTPUnit, etc) into the build.
- Deploy using FTP, telnet, or other methods to a remote server.
- Write your own Ant tasks that extend existing tasks or implement new functionality in Java (check out



WEB DEVELOPERS ARE USING OUR MX PRODUCTS IN WAYS WE NEVER DREAMED. JUST IMAGINE WHAT THEY'LL DO WITH THE NEW MX 2004.

I've been endlessly amazed by the things our customers have done using our MX generation of products. And with all the new features in Studio MX 2004, it's going to be even easier and faster for them to realize their visions.

Dreamweaver MX 2004 helps you get that picture in your head turned into a web site faster than ever. That should be welcome news to the millions of web professionals who use Dreamweaver to create sites and applications. We've added things like CSS support, target browser check and improved code hinting to help you get through projects in far less time. And with Fireworks MX 2004 you can optimize web graphics up to 85% faster.

The new Flash MX Professional 2004 takes our industry standard tool for building rich content and applications to a whole new level. It really helps stretch your abilities–no matter where you fall on the designer-developer continuum. For development, we've added things like data-aware components and an extensibility layer so you can add your own features. For design, we've added high-quality, long-format video that's really impressive.

These are just a few of the new features. The products are available individually, but they really work well together in Studio MX 2004. Don't take my word for it. Download a free trial and read more at our web site.

Let me know what you dream up. norm_01@macromedia.com

Introducing
Macromedia MX 2004.

www.macromedia.com/go/2004

http://sourceforge.net/projects/ant-contrib first to see if someone already beat you to it).

If you're looking for more on Ant, the Ant manual is a great resource, but I'd also heartily recommend the book, *Java Development with Ant,* by Erik Hatcher and Steve Loughran, especially if you're working with Java as well as ColdFusion (and in the interest of full disclosure, I should mention Erik was my officemate for nearly a year while he wrote the book). I hate to admit it, but since I started using Ant, I find creating a *build* process for new applications (gulp!) fun. Okay, maybe not fun, but at least much less frustrating than DOS scripts, **make**, and the manual lists I used to use. In fact, right now I'm headed off to have a cup of coffee since I just kicked off my release build… "ant deploy". Phew! Tough day. Probably enough time for a venti today…

About the Author

John Paul Ashenfelter is CTO and founder of TransitionPoint.com, a consulting firm based in central Virginia that specializes in developing and re-architecting ColdFusion Web applications for small- and mid-sized businesses. John Paul has been using ColdFusion since 1998 and is the author of three ColdFusion books as well as the upcoming Data Warehousing with MySQL.

ashenfelterj@transitionpoint.com

Listing 1

```
set PATH=%PATH%;c:\ant\bin;
set ANT_HOME=c:\ant
set JAVA_HOME=c:\j2sdk1.4.1_02
```

Listing 2

```
<?xml version="1.0"?>
<project name="demo" default="help">
  <description>Demonstration buildfile for CFDJ article</description>
    <target name="help" description="usage information for common tasks">
      <echo>usage: ant [target]</echo>
      <echo>typical targets: init, dist, deploy, clean, help </echo>
    </target>
</project>
```

Listing 3

```
<?xml version="1.0"?>
<project name="cfdj" default="help">
  <description>Demonstration buildfile for CFDJ article</description>

    <target name="help" description="usage information for common tasks">
      <echo>usage: ant [target]</echo>
      <echo>target can be all, init, dist, deploy, clean, install, help
</echo>
    </target>

    <target name="init" description="Initializes a project and all rele-
vant data">
      <mkdir dir="src"/>
      <mkdir dir="build"/>
      <mkdir dir="dist"/>
    </target>

    <target name="clean" depends="init" description="Cleans up the entire
project (reset)">
      <delete dir="src"/>
      <delete dir="build"/>
      <delete dir="dist"/>
    </target>

    <target name="getsource" depends="init" description="Gets code from
VSS">
      <vssget
        vsspath="/myApp"
        ssdir="C:\\Program Files\\Microsoft Visual Studio\\VSS\\win32"
        login="mylogin,mypassword"
        serverPath="\\dev\VSS_Share"
        localpath="src"
        recursive="true"
        writable="true"
      />
    </target>

    <target name="build" depends="getsource" description="Builds the code
from source files">
      <copy todir="build">
        <fileset dir="src"/>
      </copy>
    </target>

    <target name="dist" depends="build" description="Creates the distribu-
tion code">
      <copy todir="dist">
        <fileset dir="src"/>
      </copy>
    </target>

    <target name="deploy" description="Deploys code to network server">
      <mkdir dir="${deploy.dir}"/>
      <copy todir="${deploy.dir}">
        <fileset dir="dist"/>
      </copy>
    </target>

</project>
```

Download the Code…
Go to www.coldfusionjournal.com

—continued from page 7

If you have a suggestion or an opinion about the subject matter or content quality of the magazine or about a specific article(s), send this information to the magazine at info@sys-con.com or if you prefer, send it to me at simon@sys-con.com or simon@horwith.com and I will make sure that the message gets to the right person at SYS-CON.

The second thing that's necessary for *CFDJ* to be the best magazine (and resource) that it can be is for more developers to contribute ideas and articles. You don't have to have any previous authoring experience and you don't have to be "a ColdFusion rock star" to contribute, either. If you are comfortable writing about a topic and/or have a topic in mind that you think would be good in *CFDJ*, please consider submitting it. You can suggest a topic or submit a proposal for an article either by following the instructions at www.sys-con.com/coldfusion/writers.cfm or again by sending me an e-mail. If you do send me an e-mail, please also be sure to visit the site and read the instructions there… don't blindly begin writing. If you don't have a topic in mind but would like to write, contact me. If you have a topic you'd like to see but don't want to write, contact me. Whatever you do, don't keep it to yourself. Also remember that articles don't have to be on a specific CFML topic – anything related to CFML development is a possiblity. For example, if you recently began or completed a project, how did that go? What were the things you did right? What were the things you did wrong? These kinds of case studies are invaluable educational tools and also make for interesting reading.

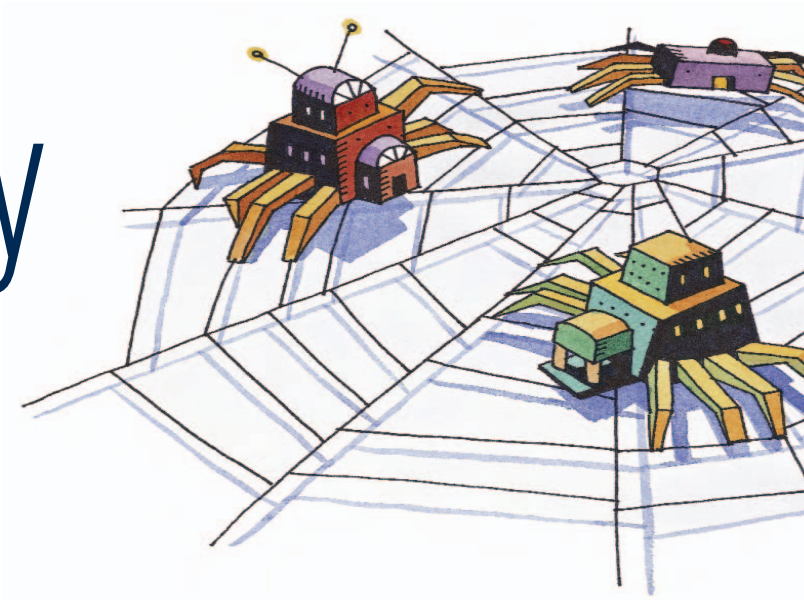
"[CFDJ] has…celebrated its fifth birthday"

There's been talk on the *CFDJ-List* and other online forums about what you, the reader, would like to see more of in *CFDJ*. Rest assured that we are listening. I've currently got a list of four or five article topics that were suggested to me that have yet to be assigned an author, and I have many other ideas as well. In response to your feedback, this month we've added a new **CF-101** column (written by Jeffry Houser) for more introductory level CF developers. Plans are also underway to add a regular column devoted to CFUG news, meeting reports, etc.

CFDJ is an excellent resource… it's also my personal favorite CFML resource and has been for quite some time. I firmly believe that as good as it is, it can be better… but only with your feedback and contributions. I urge you to submit a topic proposal and, upon acceptance, write an article. Not only will it help to make *CFDJ* a better read, it won't hurt your resume either.

The Proximity Principle

Stay close to your data



By David Shadovitz

The proximity principle holds that the best place to deal with something is as close to that something as possible. Applied to the waste industry, it yields policies that promote the management of waste as close to its point of production as possible. The idea is that transporting waste should be avoided because the transportation is costly and can lead to accidents.

Applied to marketing, it yields the advice to be "close to the customer," as Tom Peters offers in his book, *In Search of Excellence* (HarperCollins, 1982).

Applied to management, it yields MBWA, "management by walking around."

Applied to grammar, it yields the lesson that modifiers should not be separated from their subjects. What does "the bird on the pole by the pond" mean?

Let's apply it to data and software. We will look at using stored procedures, making the most of SQL, characterizing data, grouping related lines of code, and writing Boolean expressions.

Using Stored Procedures

When implementing interaction with a database, ColdFusion programmers have the choice of using stored procedures or CFQUERY with explicit SQL. The proximity principle holds that using stored procedures is the better choice because they are close to the data. In fact, stored procedures reside right there in the database, next to the data. Indeed, this recognition that a database should be a "package deal" consisting of data and functions that operate on the data is precisely why database vendors support stored procedures. This is akin to the object-oriented programming concept of a class – a set of data and functions that operate on the data.

So what benefits do we reap by using stored procedures? One benefit is that they are fast because they are precompiled. Another benefit is that they are available to *all* clients of the database, not just to your ColdFusion application. My current project involves a PostgreSQL database that is populated by Python scripts and accessed by a Web-based front end. I created an application programming interface (API) to the database. This API consists of stored procedures written in SQL and in PL/pgSQL, which is PostgreSQL's SQL procedural language. I published the API, along with instructions on how to call it from the Web-based front end and from Python scripts. The same stored procedures can be used by either of these two clients.

If you think about it, you'll see that these benefits (speed and universality) are a direct result of the proximity of stored procedures to the data.

Making the Most of SQL

When assembling a non-trivial record set, ColdFusion programmers have the choice of using a sophisticated SQL query, or a simpler SQL query followed by ColdFusion processing. By a "sophisticated SQL query," I mean one that makes use of the power of SQL's clauses, functions, and operators, many of which are not well understood and hence are underutilized. The proximity principle holds that using sophisticated SQL is the better choice, since SQL is closer to the data.

What are the benefits of using sophisticated SQL rather than post-query ColdFusion processing? One benefit is speed. I've seen people retrieve all of a table's records and then use CFIF statements to winnow out the undesired records. Those people would be better off letting the database do that work for them. In this case all that's needed is a proper WHERE clause.

Another benefit is clarity. Hudson Benson has written an article, "Using the CASE Expression in SQL Queries" (*CFDJ*, Vol. 5, issue 12), in which he explains how to replace a primitive query followed by a CFLOOP over the query results with a

more sophisticated query that makes use of SQL's SUM function and CASE statement. Not only is this faster, but it's easier to read and comprehend.

A final benefit is that you can elect to turn your sophisticated SQL statements into stored procedures, and reap further benefits.

Characterizing Data

This next item is more esoteric. There's data, and then there's metadata, which is simply data that describes other data. For example, the dollar amounts of my bank transactions are data, but the transaction dates are metadata. ColdFusion programmers often write code that processes one type of data in one way, and another type of data in another way. The proximity principle holds that when making such a decision, it is better to do so based on the characteristics of the data itself rather than on its metadata.

Here's an example. I worked on a project in which we collected data from a sensor and processed it. Well, one day we collected an unusual type of data, and it had to be processed a little bit differently. No problem; we knew the date on which we collected this unusual data, so we used this piece of metadata to control the processing:

```
<cfif collection_date = 'Mar-06-2003'>
   <cf_unusual_process data="#qryData#">
<cfelse>
   <cf_normal_process data="#qryData#">
</cfif>
```

But then we collected more of this unusual data. No problem; we just modified the test:

```
<cfif collection_date = 'Mar-06-2003' or collection_date = 'Jun-22-
2003'>
```

See where this is going? But that change (and the ones that are sure to be needed in the future) would have been *unnecessary* if we could have written the condition based on the data itself rather than on its collection date metadata. I must admit that we were unable to do so, because there wasn't much difference between the normal and unusual data. But if, for example, the normal processing algorithm required at least 1000 data points, and on those two days we collected only about 700, we could have written this much more robust condition:

```
<cfset num_data_points_required = 1000>
<cfif num_data_points ge num_data_points_required>
   <cf_normal_process data="#qryData#">
<cfelse>
   <cf_unusual_process data="#qryData#">
</cfif>
```

See how the condition is now directly tied to the data? There's no hidden knowledge, such as "on such and such a day we collected too few data points." If in the future we have another day on which we collect too few data points, this condition will catch it; there will be no need to modify the code.

If you must use metadata to make decisions, be aware that not all metadata is created equal. In my current project, an outside source provides us with tapes containing data files. Each data file consists of a metadata header followed by the actual data. The outside source also provides us with a handmade Excel spreadsheet that characterizes the data files. Much of the information in the header is also in the Excel spreadsheet. Guess what? We found that sometimes the two contradict each other, and so we asked the outside source for guidance. The response was that in the event of a conflict, trust the header. This is precisely what the proximity principle dictates, because the header and data reside in the same file and are written at the same time.

Grouping Related Lines of Code

The proximity principle holds that related lines of code should be grouped together. This makes it easier to read and to maintain. To see how well you adhere to this principle, try drawing boxes around related lines of code. If you end up with overlapping boxes, or lots of small boxes, your code may benefit from some rearranging.

Writing Boolean Expressions

Okay, this last topic may not really be a "proximity" issue, but it's worth covering. We all make great use of the CFIF tag, so let's take a look at constructing Boolean expressions for it. Consider this example:

```
<cfif (x is 5) and (y gt 10) and (pageSize is not "Env") and (dots gt
12)>
  <cf_drawshape>
</cfif>
```

It is impossible to tell the meaning of that expression. What do x, y, pageSize, and dots have to do with whether to draw a shape? You can make your code easy to read and maintain by making sure that the expression clearly states the condition. Here's a cleaned-up version:

```
<cfset bFitsOnPage = (x is 5) and (y gt 10) and (pageSize is not
"Env") and (dots gt 12)>
<cfif bFitsOnPage>
  <cf_drawshape>
</cfif>
```

Now the meaning of the expression is clear. We may not fully understand x, y, pageSize, and dots, but we do understand that we only want to draw a shape if it will fit on the page.

Conclusion

It is my hope that these applications of the proximity principle to data and software will make for better software engineers producing better software. In this vein I recommend reading Steve McConnell's book, *Code Complete* (Microsoft Press, 1993).

About the Author

David Shadovitz is a senior software engineer at Northrop Grumman in Los Angeles and a member of Team Macromedia.

david@shadovitz.com

Creating Variables in CFML

The basis of many CF programming projects

By Jeffry Houser

Welcome to **CF101**, a new column I'll be writing for ***ColdFusion Developer's Journal***. This column is dedicated to all of you beginners out there, to teach you the basics of ColdFusion development. You don't need to have a prior understanding of programming, HTML, or Web development to read this column, although if you do, it's a great way to help reinforce the basics.

For the first column, I thought I'd talk a bit about what variables are and how we can create and use them in ColdFusion. Before we jump into that, I thought you might like to know a little bit about my development experience.

Who is Jeffry Houser?

My experience in the IT world started over seven years ago. I was the "tech guy" at a small advertising firm. I did a bit of everything, but most of the development was in Lotus Notes. I worked with very early versions of the Lotus Domino Web Server. Back then, building Web sites in Domino was like trying to tie your shoes while wearing two pairs of gloves and some mittens. It was possible, just not very easy.

Around 4:59 one Friday evening, I was handed a project with a Monday morning deadline. Another developer had left the company for greener pastures and one of his projects had been left unfinished. The project was being built in ColdFusion with Microsoft Access in the database. Although I had no previous ColdFusion training, I was able to complete the project in 10 hours on a rainy Saturday. I haven't looked backed since.

I left that small advertising firm in 1999 to start DotComIt, a Web consulting company. Since starting DotComIt, I've written three ColdFusion books, including *ColdFusion: A Beginner's Guide*; spoken at a bunch of user groups; and written a handful of articles. I've dealt with a lot of technologies over the years, but ColdFusion remains my favorite Web development technology because of its simplicity. I've always had a knack for describing the complex in a straightforward way, and that is why the folks at *CFDJ* thought of me when they decided to create this column. I have also been a musician for over 20 years and own a recording studio. I mention that only because many people find it more interesting than my Web development adventures. Hopefully my life story doesn't stick in your head for too long; let's talk about variables.

Understanding What a Variable Is

Before I explain how to use variables within CFML (CFML is the programming language of ColdFusion and BlueDragon; whereas ColdFusion specifically refers to the Macromedia server product), I want to make sure that you understand what variables are. A variable in programming is very similar to a variable in algebra. It has two parts: a name and a value. You can use the variable's name to refer to the variable's value. In computing, the variable is a place in memory where a value is stored. You can assign a name to the place in memory to reference its value.

You might want to ask why we would want to use a variable instead of just using the value that it points to. That would be a good question; and there are multiple reasons. First, a variable's value can change, but the variable name never does. If we use the variable in 100 spots, then we only have to change it once, not 100 times. This could be a real time-saver when writing code. For instance, if the Webmaster's e-mail address needs to be displayed in various places in a site, storing the address in a variable and then displaying the value of that variable would mean that if the Webmaster's e-mail address changed, the value of the variable would need to be changed only in the one place where it is set, as opposed to having to change every place where the address is used.

A second reason for using variables is that you may not know what the value of the variable is going to be. One of the more common uses of this is when you are accepting input from a user, usually through an HTML form. Variables can be used to verify the user input, insert it into a database, mail it to someone, or perform whatever other processing you need to do.

Creating Variables with <cfset>

You can create variables in ColdFusion using the <cfset> tag. The <cfset> tag takes this form:

```
<cfset variableName = value>
```

As with all ColdFusion tags, it starts with the name of the tag, <cfset>. After that you specify the name of the variable that you want to create. Then comes the equal sign, followed by the value you want to give the variable.

ColdFusion has some special conventions that you must follow when chosing a variable name. If you do not follow these rules, ColdFusion will display a nasty error message when you try to execute a template. These are the rules:

- The first character of a variable name must be a letter, a dollar sign ($), or an underscore. If backwards compatibility is a requirement you should stick with an alpha character (a letter). The underscore and dollar signs work only in ColdFusion MX or BlueDragon.
- The remaining characters can be made up of any number of letters, numbers, dollar signs, or underscore characters. You are more than welcome to mix upper- and lower-case letters as part of a variable name.
- Variable names cannot contain a space.
- Variables cannot include reserved characters or words. This includes punctuation marks (other than underscore and dollar sign) such as the double quotation mark (") and names of tags or built-in functions. The full list of reserved words can be found at http://livedocs.macromedia.com/coldfusion/6.1/htmldocs/expressi.htm.

Within the scope of these few rules, you are wide open to name your variables in whatever way you please. I would always recommend that you try to choose descriptive names for your variables. The X and Y you used to name variables in algebra will not be as helpful as values like Username, FirstName, or Password; especially when you have to revisit your code six months after writing it to make some enhancements.

A variable can hold many different types of values. They can take simple values such as integers, real numbers, strings, or Booleans. An integer is a whole number, such as 12 or 205. A real number is a number with a decimal, such as 12.5 or 19.3. Strings are text values such as "This is a cfset example." Strings are always placed inside quotes, while numeric values are not. A Boolean value has one of two possible values, true or false. In CFML, the words "true" and "false" and "yes" and "no" are Boolean values. A number can also be used as a Boolean value – zero is false and all other values are true.

Complex values such as arrays, structures, or ColdFusion Components can also be assigned to variables with the <cfset> tag. We'll reserve the discussion of complex values for a future column. An expression can also be contained in the value portion of the <cfset> tag. ColdFusion will execute the expression and use the result as the value of the variable. We'll reserve an in-depth discussion of ColdFusion expressions for another article. Right now you can refer to the Macromedia documentation to learn more: http://livedocs.macromedia.com/coldfusion/6.1/htmldocs/cfml_b14.htm#wp1160241.

Creating Some Variables

With an understanding of creating variables, here's some code that will create a few variables using the <cfset> tag:

```
<cfset FirstName = "Jeffry">
<cfset LastName = "Houser">
<cfset Age = 28>
<cfset State = "California">
<cfset State = "Connecticut">
<cfset State = 28>
```

The first line creates a variable called FirstName with a string value of Jeffry. The second variable creates a variable called LastName with a string value of Houser. The third creates a variable called Age with an integer value of 28. Then we create a variable called State and give it the value of "California". The next line changes the value of the state variable from "California" to "Connecticut".

The final line of the code segment changes the state variable's value from a string to an integer. This is an example of changing a variable's type – something that's very easy to do in ColdFusion. This is different from the way many other programming languages operate, and is part of the reason why ColdFusion is very easy to work with. You can easily change a value from an integer to a string and back just by placing another <cfset> tag.

Outputting (Displaying) Variables

Now that you've created a bunch of variables, what do you want to do with them? There are an unlimited number of things that you can do with variables, many of which I'll cover in future columns. In this column we'll just show a simple example of outputting a variable. We can do this with the <cfoutput> tag and a ColdFusion expression.

```
<cfoutput>
 #FirstName#<br>
 #LastName#<br>
 #Age#<br>
 #State#<br>
</cfoutput>
```

Before going too much further, I want to specify that this code segment must be put in the same template as the previous code segment to execute properly. This listing starts with the <cfoutput> tag. When the ColdFusion server sees the <cfoutput> tag, it knows to process all the text between the beginning <cfoutput> and the end </cfoutput>. The difference between the start and end tag is that the end tag has a slash before the tag name. This is identical to the way that HTML handles start and end tags. Not all tags have both a start and end tag. The <cfset>, for instance, has no closing tag.

You can place any sort of text that you want in the <cfoutput> tag block. The way to tell ColdFusion the difference between normal text and an expression that is to be evaluated is with the use of the pound sign, "#". Text between two separate pound signs is evaluated as an expression. Make sure that there are no spaces between the pound signs and the expression. Any text without pound signs is completely ignored by the ColdFusion server and is passed on to the browser unmodified. This text could be HTML, JavaScript, CSS, or any plain text.

The second line of the template contains a ColdFusion expression and an HTML tag. ColdFusion will look at the expression and return the value of the FirstName variable, "Jeffry". The HTML <br> tag will go straight to the browser. The same happens for the LastName, Age, and State. The code segment finishes with the end cfoutput tag. The output should look like this:

```
Jeffry
Houser
28
28
```

You'll notice that the State variable contains the most recent value it was set to, and the other values, "California" and "Connecticut", are lost.

Variable Scopes

At this point, you should have a good understanding of how to create variables within ColdFusion. There is one more important topic you'll need to understand about variables. This is the concept of variable scopes. A scope is used to define the extent of a variable's life before the server destroys it (removes it from memory), and determines how it can be used. To access a variable in a specific scope, you can specify the scope before the variable's name, and separate it from the variable name with a period, like this:

```
ScopeName.VariableName
```

Just as you could mix case sensitivity with variable names, you can also do so with variable scopes. They are not case sensitive. Here are some common variable scopes:

- ***Variables:*** This is the local variable scope of a page, the default scope ColdFusion stores variables in, and the first place ColdFusion will look for variables whose scope has not been specified. It is available throughout the execution of a single template and all the include files that the template uses. It is not available to custom tags or ColdFusion Components called from within the template.
- ***Request:*** The request scope is very similar to the local variable scope, except that it is available across a whole request, including inside custom tags or ColdFusion Components called from the main template.
- ***Form:*** When you use an HTML form, its values are placed inside the form scope on the form-processing page. The life of the form scope is a single template execution. Like the request scope, it will be available to custom tags or ColdFusion components.
- ***URL:*** The URL scope contains variables specified in the query string portion of a URL. They have a life similar to the form and request scopes.
- ***Cookie:*** The cookie scope refers to variables that are stored by the user's Web browser. These variables are available during a single template's execution, in much the same way

that form, URL, and request scopes are. However, unless you do something to erase a cookie variable from the user's browser, the value will be available on all page requests.

- ***Application:*** The application scope is unique to a particular ColdFusion application. You can set up a ColdFusion application using the cfapplication tag. You can read more about the cfapplication tag at http://livedocs.macromedia.com/coldfusion/6.1/htmldocs/tags-pa3.htm. The application scope is known as a persistent scope, because it exists between page requests. An application variable will exist once, no matter how many users are using your application.
- ***Session:*** The session scope is another persistent scope. A single session exists for each user of a particular application. Just like the application scope, session usage is set up using the cfapplication tag. A different set of values in the session scope will exist for each individual user on your application.

This is a list of some of the more commonly used scopes, but it is not comprehensive. You will most likely discover more scopes during your development adventures in ColdFusion.

In our code example from the previous section, we did not specify a variable scope. When no scope is specified, the variables are placed in the local scope, which is named "variables". We could easily rewrite the previous listing to look like this:

```
<cfoutput>
 #variables.FirstName#<br>
 #variables.LastName#<br>
 #variables.Age#<br>
 #variables.State#<br>
</cfoutput>
```

This code segment works identically to the one that did not specify the variables' scope before the variable name.

Conclusion

Variables will be the basis of many of your programming projects within ColdFusion. This article should have given you a strong introduction to creating ColdFusion variables, and an overview of some of the uses they provide. In my next column, I'll take an in-depth look at ColdFusion expressions. If there's something you'd like to see in this column, feel free to drop me a line to let me know.

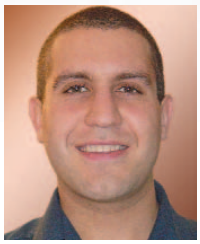
About the Author

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

jeff@farcryfly.com

Collaborative Filtering

Using predictive analysis to make recommendations



By Joe Danziger

Collaborative filtering on the Web has existed for a long time, dating all the way back to the original incarnations of sites like CDNow and Amazon.com. Recommendation systems are a powerful tool for businesses to extract additional value from their e-commerce and customer databases. They benefit customers by enabling them to find products they like, and help businesses by generating more sales.

We're going to look at some of the basic principles of predictive systems and introduce some methods you can utilize to make recommendations in your own applications. Along the way, we'll attempt to point out the benefits and limitations of each type of system.

Basic Predictions

At the most basic level, predictive information can be provided manually for your items. This can be built into the backend administration of the site. When adding products to an e-commerce site, you could include a multiple select box listing all of the additional items in that category. Selecting items in the list would create a list of product IDs to be stored in an additional "related items" field in your database. With one additional query on a product detail page, you can pull up details on all of the related items that have been associated with the item being viewed.

This scenario can provide quick, quality recommendations as the computer is not guessing at the association and also does not have to perform any on-the-fly calculations. The technique suffers, however, by requiring your product administrator to have a deep knowledge of the products in your store, which may be unrealistic for larger sites. It also requires you to continuously update the "related items" lists of older items as new products are added to the catalog.

User-Based Collaborative Filtering

A second approach to providing recommendations is to use collaborative filtering, which is a technique to make predictions without any explicit relationships defined within the database. There are two types of collaborative filtering that are common: user-based and item-based. User-based filtering works by building a database of ratings for products by consumers (see Listing 1).

We'll assume an Items table and a Users table in the database with respective primary keys of ItemID and UserID, and we'll rate using a scale of 1 (lowest) to 5 (highest). You can go as high as you'd like, though statistically there's not much value in going above 7. The system will determine, on the fly, a community of like users whose ratings of items most closely match those of the current user. We'll set up a sample table of five users providing ratings for each of the colors of the rainbow (Figure 1).

To determine our community of users, we'll use the "Mean Squared Differences (MSD)" algorithm. This measures the degree of dissimilarity between two user profiles. Squaring adds more weight to the larger differences, which is appropriate since points further from the mean may be more significant (we care more about things that a user has a positive or negative feeling about versus items they are ambivalent about). To perform the calculation in laymans' terms: take the difference between the two users' rankings on each item that

they have both rated, square that number, add those all up, and take the average. The lower the result, the closer that user's preferences are to the current user. Listing 2 provides the query used to determine the community of users with the lowest mean squared difference to the user. Figure 2 provides the results of the query and the MSD values. We'll use a TOP value of 5 at the beginning of our query to display only the five most similar users to userID 1.

We're going to use the three most like-minded users to come up with predictions on what colors this user would like. We see from Figure 2 that our three closest neighbors are Mike, Laura, and Sam, since they have the lowest MSD values. Products that this community likes most will then be recommended to the user, as he will probably also like them. We loop over each member in the community and assign a weighted rating (based upon their MSD value) to each of the other items that they have rated (see Listing 3). These weighted ratings from the query in Listing 3 are then inserted into a database table (see Listing 4) to aid with our calculations.

Now that we have all of our weighted ratings in the database, we total up the weighted ratings and divide by the total MSDs to give us the items with the highest weighted averages that have not already been rated by the user (see Listing 5).

Our final results are shown in Listing 6. Although this is a simplified example, it allows us to see where our recommendations come from. Better predictions would be gained by increasing the neighborhood size (up to a point), so you should experiment to find a reasonably large neighborhood size that does not significantly affect processing time. Since we were using a scale of 1–5, the higher the weighted average for the prediction, the more likely this user is to desire this item (or color in our case).

Although we used the Mean Squared Differences algorithm, there are several other mathematical formulas each with their own drawbacks and limitations. The model presented could easily be modified to provide recommendations of favorite artists, authors, or whatever your site calls for. You could also base recommendations on the demographics of your users, or you may want to provide an explicit survey for all of your users to fill out to gain knowledge of your users' preferences on whatever topic your site deals with.

	Red	Orange	Yellow	Green	Blue	Indigo	Violet
Harry (ID: 1)		2		3		4	1
Mike (ID: 2)	5		2		5		2
Sam (ID: 3)	1	4		2		2	
Laura (ID: 4)		2	3		3		3
Betty (ID: 5)	3		2	1		2	
Sue (ID: 6)		5		2	4		2

Figure 1: Sample user-item rating (scale of 1-5)

User ID	User Name	MSD
2	Mike	1.000
4	Laura	2.000
3	Sam	3.000
6	Sue	3.667
5	Betty	4.00

Figure 2: Closest matching users and MSD values

Drawbacks of User-Based Collaborative Filtering

One of the major drawbacks of user-based predictive systems in general is that they do not scale well. The computational complexity of these methods grows linearly with the number of customers and items, which in commercial applications can each grow to be several million. Another problem deals with the sparsity of recommendations on the data set, which might be quite large. In large e-commerce sites, even active customers may have purchased well under 1% of the products. Therefore, a system based upon nearest neighbors may be unable to make any product recommendations for a particular user. To address these scalability concerns, item-based recommendation techniques have been developed to identify relationships between the items themselves, and to use these to compute a list of recommendations.

Item-Based Collaborative Filtering

One way to make item-based recommendations is to simply look at items that a user has purchased together or that were part of the same transaction. Items that appear the most in orders in which the specific item appears would be the most likely to be a successful prediction. A sample query is provided in Listing 7.

This is the simplest way to provide quality item-based recommendations. It should perform quickly on the fly, but could always be run offline as a scheduled job for your entire database. A more in-depth discussion is beyond the scope of this article, but you can visit the link below for articles that will lead you in the right direction.

Conclusion

The recommendation technique you choose depends on the nature of your users and your application. You may have a small, controlled site with a limited set of users where user-based collaborative filtering may work just fine, or you may have a very large site with many items, which would necessitate an item-based solution. The key is to choose carefully and test things out to make sure they perform and scale correctly. It should also be noted that in many cases, it may make sense to perform the predictions themselves as a scheduled job and just store them in the database as part of the record for each item. Other cases may allow you to perform the recommendations on the fly in a brief amount of time.

Credit should also be given to Peter Boot who put out the first collaborative filter custom tag back in 2001. For more info on the science of collaborative filtering, you can visit http://jamesthornton.com/cf/ to find links to more than 40 articles and research papers that deal with the subject. Much research continues to be done on the science of determining which collaborative filtering algorithms work best.

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 since version 1.5.

joe@djcentral.com

Listing 1: Create Database Table

```
CREATE TABLE CollabRatings (
    UserID int(8),
    ItemID int(8),
    Rating float
)

PRIMARY: UserID,ItemID
```

Listing 2: Determine Community of Like Users Using Mean Squared Differences

```
<cfset userID = 1>

<cfquery name="getNearestNeighbors" datasource="CF">
SELECT TOP 5 MIN(c.UserID) AS OtherUserID, AVG(POWER(ABS(a.Rating-
c.Rating),2)) AS MSD
FROM CollabRatings a, Users b, CollabRatings c
WHERE a.UserID=b.UserID AND b.UserID = #userID#
AND c.ItemID=a.ItemID AND c.UserID<>a.UserID
GROUP BY c.UserID
ORDER BY MSD ASC
</cfquery>

Note: It's a good idea to use TOP X in your query to to improve scalability
```

Listing 3: Assigning Weights to Other Items Ranked by Community Members

```
<cfquery name="getList" datasource="CF">
SELECT ItemID, (1/#NumberFormat(MSD,"00.0000000")#) AS MSDCalc, Rating,
Rating*(1/#NumberFormat(MSD,"00.0000000")#) as WeightedRating
FROM CollabRatings2
WHERE UserID=#OtherUserID#
ORDER BY WeightedRating DESC
</cfquery>
```

Listing 4: CollabCalcTable for Aiding Calculations

```
CREATE TABLE CollabCalcTable (
    CalcID                    int
    ItemID          int,
    MSD                       float,
    Rating          float,
    WeightedRating  float
)

PRIMARY: CalcID - generated randomly using <cfset CalcID =
RandRange(1000000,9999999)>
```

Listing 5: Get Final Weighted Averages

```
<cfquery name="getWeightedAverages" datasource="#request.dbsource#">
SELECT TOP 3 MAX(b.ItemIdentifier) AS ItemIdentifier, MAX(b.ItemName) AS
ItemName,
SUM(a.WeightedRating)/SUM(a.MSD) AS PredictedRating
FROM CollabCalcTable a, CollabItems2 b
WHERE CalcID=#iCalcID# AND a.ItemID=b.ItemID
AND a.ItemID NOT IN (SELECT ItemID FROM CollabRatings2 WHERE
  UserID=#getUserID.UserID#)
GROUP BY a.ItemID
ORDER BY PredictedRating DESC
</cfquery>

Note: We USE TOP X equal to the number of predictions we'd like
```

Listing 6: Final Recommendations

```
Color      Weighted Average
Blue       4.333
Red 4.000
Yellow     2.333
```

Listing 7: Most Items Ordered Along with Another Item

```
<cfquery name="MostItemsOrdered" datasource="CF" >
SELECT TOP 3 OrderItems.ItemID,Items.Title,count(OrderItems.ItemID) as
ItemSaleCount
FROM OrderItems,Items
WHERE OrderID IN (SELECT OrderID FROM OrderItems WHERE ItemID = 3)
AND OrderItems.ItemID != 3
AND OrderItems.ItemID = Items.ItemID
GROUP BY OrderItems.ItemID,Items.Title
ORDER BY ItemSaleCount DESC
</cfquery>

Note: The TOP X value determines how many similar items to recommend
```

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

Now More Than Ever Before...

JAVA DEVELOPER'S JOURNAL

JDJ

Means Business

Subscribe Now!

$69.99

1 YEAR/
12 ISSUES

www.SYS-CON.com/JDJ

or call

1-888-303-5282

The World's Leading *i*-Technology Magazine Just Got Better!

Now with expanded business coverage, *JDJ* is the world's premier independent, vendor-neutral print resource for the ever-expanding international community of IT professionals.

The Most Popular *i*-Technology Magazine in the World !

SYS-CON MEDIA

The World's Leading i-Technology Publisher

Adventures in Encapsulation PART I

Creating array-like functionality, without using an array

By Hal Helms

I've just returned from a four-day class on Mach-II (www.mach-ii.com) that Ben Edwards and I gave to a group of developers at Macromedia in San Francisco. It was great fun to be working with so skilled a group. (You can read about the class from the perspective of Macromedia's director of architecture in IT at www.corfield.org/blog.)

We spent the first day working on object orientation, placing great stress on encapsulation, a subject that I've written about several times in *CFDJ*. Encapsulation is as much a philosophy as a practice and it centers on the idea that data and behavior should be tied together. What does this mean? Perhaps an example will serve best.

Let's assume that we have a table in a relational database called **TaxRates**. One column has the abbreviation for the state or province and the other has a numerical value. Let's further assume that you are working on your application and I on mine and we both need the tax rate for California. The answer is straightforward enough:

```
<cfquery datasource="MyDSN" name="tax">
   SELECT rate
   FROM TaxRates
   WHERE stateprov = 'CA'
</cfquery>
```

Now, you can use **tax.rate** in your program and I can do the same in mine and it will work fine – right up to the time that the company that supplied us with the information on the myriad of localities with their different tax rates goes broke. Management finds a new company but their information is stored in a different database and both of us have to rewrite our queries. Then later, the new company decides that rather than supplying information for databases, they will use Web services. They still provide the database info for now, but for how much longer isn't certain. We must both rewrite our queries to work with Web services.

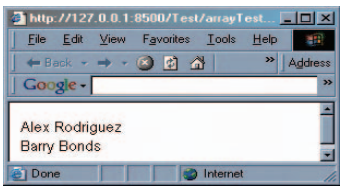


Figure 1: Using the Iterator to display the contents of ArrayList

Now, this is a simple example. Multiply it, though, to the degree of complexity of modern corporate life and, as the Apollo 13 astronauts stated, "Houston, we have a problem." But by encapsulating both data and behavior into a single component, the problem is very easily handled. The component is named **taxMan** and the method we will use to get our needed information is **getTaxRate**. The method requires a two-letter abbreviation.

In an encapsulated world, both of us would get the information we need by this simple invocation: **taxMan.getTaxRate**('CA'). The invocation provides us with an abstraction from the sometimes-strange world of databases (and the always-strange world of taxes). Regardless of how often the tax rate changes or how the provider chooses to store this information, our programs are insulated from these changes.

This idea of encapsulation is taken to great lengths by serious programmers. In the Java world, it means that all and any components that need the **taxRate** info would need to call the **getTaxRate** method – even other methods within **taxMan** itself. Look at this code for **calculateMaxTax**, a method that applies logic to help ensure that no one is taxed unfairly. In order to do its job, **calculateMaxTax** needs access to the **taxRate** of a particular province, but rather than directly accessing the tax rate, it respects the encapsulation and asks **getTaxRate** for help.

```
<cffunction name="calculateMaxTax" returntype="numeric">
   <cfargument name="personalIncome" type="numeric" />
   <cfreturn Max(getTaxRate() * arguments.personalIncome,
arguments.personalIncome) />
</cffunction>
```

But this level of encapsulation—desirable as it is—can present us with a problem in a special case with ColdFusion. Consider this code:

```
<cfcomponent displayname="BaseballTeam">
   <cfset variables.team = ArrayNew(1) />

   <cffunction name="getTeam" returntype="array">
         <cfreturn variables.team />
   </cffunction>

   <cffunction name="addPlayer">
         <cfargument name="player" type="BaseballPlayer" />
         <cfset ArrayAppend(getTeam(), arguments.player) />
   </cffunction>
</cfcomponent>
```

The problem is that arrays are returned by *value*, not by *reference*. This means that **getTeam** will return a *copy* of variables.team, not a *reference* to the actual team property. Because of this, **addPlayer** will not add a player where you want it – in the **BaseballTeam**'s **team** property.

Luckily, encapsulation comes to the rescue. Rather than using the built-in array function of ColdFusion, we can create our own, encapsulated CFC that deals with collections of things that need array-like functionality. Let's call this new CFC, **ArrayList** (see

Listing 1). **ArrayList** gives us array-like functionality, but does so without using an array. **ArrayLists** are copied by reference.

We can then create another encapsulated component that can accept any **ArrayList** and assists us in looping over any **ArrayList**. We'll call this CFC an **Iterator** (see Listing 2).

To illustrate this, we'll create a **BaseballTeam** CFC. Of course, what would any team be without its players, so let's create another CFC, **BaseballPlayer**. Now, look at how simple the code is that uses **ArrayList** and **Iterator** in the context of a baseball team:

```
<cfset dreamTeam = CreateObject('component', 'BaseballTeam') />

<cfset p1 = CreateObject('component', 'BaseballPlayer').init("Alex
Rodriguez") />
<cfset p2 = CreateObject('component', 'BaseballPlayer').init("Barry
Bonds") />

<cfset dreamTeam.addPlayer(p1) />
<cfset dreamTeam.addPlayer(p2) />

<cfoutput>
<cfset it = dreamTeam.getTeam().getIterator() />
<cfloop condition="#it.hasNext()#">
   #it.next().getName()# <br />
</cfloop>
</cfoutput>
```

Gone is the undesirable behavior we had when dealing with a ColdFusion array; **ArrayList** has solved the problem for us. Hopefully, you see the possibilities in both **ArrayList** (and encapsulation generally) and you're thinking: "Boy, that Hal Helms guy sure is smart!" Well, hold that thought for I must confess that this concept was stolen and then adapted from the way that Java handles collection objects. It's one of the ways that knowing Java can help us be better ColdFusion programmers.

If you'd like to download the code for **BaseballPlayer** or **BaseballTeam**, visit halhelms.com where you can sign up for my "occasional newsletter" on techie topics I find interesting.

About the Author

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

hal.helms@teamallaire.com

Listing 1: ArrayList.cfc

```
<cfcomponent displayname="ArrayList" hint="I provide array-like function-
ality but am passed by reference">

    <cfset variables.elements = StructNew() />

    <cffunction name="init" access="public" returntype="ArrayList" out-
put="false">
            <cfreturn this />
    </cffunction>

    <cffunction name="add" access="public" returntype="void"
output="false">
            <cfargument name="element" type="any" required="true" />
            <cfset var elements = getElements() />
            <cfset elements[StructCount(elements)] = arguments.element />
    </cffunction>

    <cffunction name="getElements" access="public" returntype="struct"
output="false">
            <cfreturn variables.elements />
    </cffunction>

    <cffunction name="getElementByIndex" access="public" returntype="any"
output="false">
            <cfargument name="index" type="numeric" required="true" />
            <cfset var elements = getElements() />
            <cfreturn elements[arguments.index] />
    </cffunction>

    <cffunction name="getIterator" access="public" returntype="Iterator"
output="false">
            <cfreturn CreateObject('component', 'Iterator').init(this) />
    </cffunction>

</cfcomponent>
```

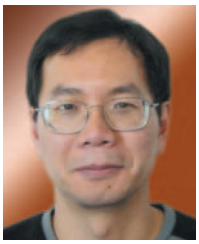
Listing 2: Iterator.cfc

```
<cfcomponent displayname="Iterator" hint="I provide capabilities for
looping over collections">
    <cfset variables.collection = "" />
    <cfset variables.pointer = 0 />

    <cffunction name="init" access="public" returntype="Iterator" out-
put="false">
            <cfargument name="collection" type="ArrayList" required="true"
/>
            <cfset setCollection(arguments.collection) />
            <cfset setPointer(0) />
            <cfreturn this />
    </cffunction>

    <cffunction name="hasNext" access="public" returntype="boolean" out-
put="true">
            <cfif StructCount(getCollection().getElements()) EQ 0>
                    <cfreturn false />
            </cfif>
            <cfif getPointer() LT
StructCount(getCollection().getElements())>
                    <cfreturn true />
            </cfif>
            <cfreturn false />
    </cffunction>

    <cffunction name="next" access="public" returntype="any"
output="false">
            <cfset var elements = getCollection().getElements() />
            <cfset var currentPointer = getPointer() />
            <cfset setPointer(currentPointer + 1) />
            <cfreturn elements[currentPointer] />
    </cffunction>

    <cffunction name="getCollection" access="public"
returntype="ArrayList" output="false">
            <cfreturn variables.collection  />
     </cffunction>
    <cffunction name="setCollection" access="public" returntype="void"
output="false">
            <cfargument name="collection" type="ArrayList" required="true"
/>
            <cfset variables.collection  = arguments.collection  />
     </cffunction>

    <cffunction name="getPointer" access="public" returntype="numeric"
output="false">
            <cfreturn variables.pointer  />
     </cffunction>
    <cffunction name="setPointer" access="public" returntype="void" out-
put="false">
            <cfargument name="pointer" type="numeric" required="true" />
            <cfset variables.pointer  = arguments.pointer  />
     </cffunction>

</cfcomponent>
```

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

Writing Queries for Oracle vs SQL Server

Utilize the database in CF apps



I-Lin Kuo

Have you ever written an application using one database only to have it break when you switched to a different database? How often have you decided to limit the functionality of your database by writing only basic SQL because you just weren't sure if your queries would work on a different database?

Or maybe you've written ColdFusion code to manipulate query results without realizing that the database could do the same manipulation quicker and more efficiently. In general, the database tends to be seriously underutilized in most applications. This article aims to provide a quick overview of some important SQL differences between the two most popular commercial databases – Oracle9*i* and SQL Server 2000 – to help the CF developer write more portable and database-smart applications.

Sequences and Identity Columns

One of the major differences between Oracle and SQL Server is in the treatment of sequential primary keys. To obtain an automatically generated sequential primary key for a table in SQL Server, all you have to do is make the column an IDENTITY column. Oracle, on the other hand, requires that you first create a sequence and insert its value explicitly.

```
SQL Server:
CREATE TABLE products (
   ID     IDENTITY,
   Desc   char(50)
);
INSERT INTO products(desc) VALUES('Acme widget');

Oracle:
CREATE SEQUENCE products_seq
   START WITH     1000
   INCREMENT BY   1;
CREATE TABLE products (
         ID         NUMBER,
         Desc       char(50)
);
INSERT INTO products(ID,desc)
VALUES(products_seq.NEXTVAL,'Acme widget');
```

Another difference is that SQL Server allows you to conveniently retrieve the auto-generated ID from the insert via the following code:

```
SQL Server:
<cfquery name="insertProducts">
SET NOCOUNT ON;
INSERT INTO products(desc) VALUES('Acme widget');
SELECT SCOPE_IDENTITY() AS [SCOPE_IDENTITY];
SET NOCOUNT OFF
</cfquery>
```

Warning: there are variants of the above code that retrieve the ID using @@IDENTITY rather than SCOPE_IDENTITY. @@IDENTITY is not as good as SCOPE_IDENTITY because it will give an incorrect result if you have a trigger on the table that inserts into another table.

In Oracle, because you're managing the sequence explicitly, you would retrieve the sequence value before you do the insertion rather than retrieving the value after insertion.

```
Oracle:
<cfquery name="getSeqValue">
   SELECT products_seq.NEXTVAL AS ID from DUAL
</cfquery>
<cfset ID=getSeqValue.ID[1]>
<cfquery>
   INSERT INTO products (ID,desc) VALUES(#ID#,'Acme widget')
</cfquery>
```

However, these advantages of the SQL Server approach to sequential primary keys come with a price – you cannot insert a value into an IDENTITY column. Thus, if you try to bulk import rows from another table into a table with an IDENTITY column, you will have to omit the original column corresponding to the IDENTITY column. This becomes especially painful to manage properly if the omitted column is the foreign key value by which other tables reference your original table. For this reason, despite

its ease of use, it is often a good idea to avoid the use of the IDENTITY column in SQL Server.

Derived Tables and Top *N* Queries

A common question that keeps coming up in various CF forums is: How do I retrieve only the first *N* results of a query? While this is a task that's possible to do in a database-independent way by using CF to control the looping over the query, in general that would not be a good idea. That's because in order to do this, CF has to retrieve all the rows of the query and then filter out the rows that it does not want. As a result, a large amount of memory and network bandwidth is consumed to process unwanted rows. It is a much better idea to get the database to do the work of selecting and returning only those results that you want.

To illustrate our next examples, we'll be using the small Employees table shown in Table 1. The SQL to create the table is located at the end of the article.

Using the sample employees table as an example, I would run the following queries in SQL Server to retrieve the names and salaries of the three highest-paid employees:

```
SQL Server:
SELECT TOP 3 name, salary FROM Employees ORDER
BY salary DESC
```

In Oracle, it's slightly more complicated. We have to make use of a derived table to sort the results first (a **derived** table is a select query that is used in place of a table). Then we use the ROWNUM pseudocolumn to select the first 3 results (a **psuedocolumn** is a column that doesn't actually exist in the table that nonetheless has a value that can be used in a query).

```
Oracle:
SELECT name, salary
FROM (
SELECT name, salary
FROM Employees
ORDER BY salary desc
) WHERE ROWNUM <=3;
```

Note that since I want the three highest salaries, I have to order my results in descending order of salary. The ROWNUM condition on the outer query restricts the results to the first three.
Both queries return:

```
Name      salary
George    100000
Ling      90000
Sandeep   60000
```

For the sake of consistency, the ORDER BY clauses in top *N* queries are not optional. Without an ORDER BY clause, the database is free to return results in any order it chooses, so the top *N* may be very different from one minute to the next depending on the internal state of the database, even if the data itself does not change.

Derived Tables and Results by Pages

A common variation on the previous question is: How do I retrieve only the first *N* through *M* results of a query? For instance, this kind of question might come up when paging through a large query 100 results at a time. In both SQL Server and Oracle, this can be accomplished by using a derived table.

In SQL Server, to retrieve the first *N* through *M* results, we must select the top *M* results in a derived table, reverse the order, and then select the top M-N+1 results of the reversed derived table. As an example, the following query selects the third through seventh highest paid employees.

```
SQL Server:
SELECT TOP 5 name, salary
FROM (
SELECT TOP 7 name, salary
FROM Employees
ORDER BY salary desc
)
ORDER BY salary asc
```

Now, how would we get the same results with Oracle? You might think that by modifying the previous Oracle query that the following would work:

```
SELECT name, salary FROM (
SELECT name, salary FROM Employees
ORDER BY salary, desc
) WHERE ROWNUM >=3 and ROWNUM <=7
```

Unfortunately, there are some oddities with the use of Oracle's ROWNUM pseudocolumn that cause the above query to return zero results. The query that actually works requires the use of a derived table within a derived table:

```
Oracle:
SELECT name, salary
FROM (
    SELECT ROWNUM as rnum, name, salary
    FROM (
          SELECT name, salary
          FROM Employees
          ORDER BY salary DESC
    )
)
WHERE rnum <= 7 and rnum >= 3;
```

ID	NAME	DEPT	SALARY	HOBBIES
1	George	Finance	100000	golf
2	Ling	IT	90000	golf
3	Sandeep	Finance	60000	tennis, volleyball, ballooning
4	Manuel	Finance	30000	stamps
5	Shaya	Finance	30000	NULL
6	Akane	IT	55000	comic books
7	Bjorn	IT	35000	cricket
8	Jean	IT	35000	

Table 1: Emlpoyees

Note: it is very important that the ROWNUM in the derived table is given an alias, otherwise the query will return zero results.

Both queries return:

```
NAME            SALARY
Sandeep         60000
Akane           55000
Bjorn           35000
Jean            35000
Manuel          30000
```

NULL

NULL behavior is an extremely tricky issue. To the average CF developer, NULLs are often dealt with unobtrusively – a column with a value of NULL returns an empty string. For example, the following CF code results in "Shaya likes []".

```
<cfquery name="myQuery">
SELECT hobbies FROM Employees WHERE name='Shaya'
</cfquery>
<cfoutput>Shaya likes [#myQuery.hobbies#]</cfoutput>
```

Things get trickier when there are numeric expressions involving NULL. ANSI standards dictate that NULL should propagate during a calculation. In layman's terms, this means that any expression involving a NULL should evaluate to a NULL, thus 8+NULL and cos(NULL) should both evaluate to a NULL, which to CF looks like an empty string. Both SQL Server and Oracle are compliant in this respect.

However, with string expressions involving NULL, we see noncompliant behavior from Oracle. According to ANSI standards, concatenating a string to a NULL should also produce a NULL. In Oracle, a NULL will automatically be converted to a zero-length string for concatenation. For SQL Server, the default ANSI-compliant behavior can be overridden by disabling the **concat null yields null** setting of **sp_dboption**.

```
Oracle:
SELECT 'Shaya likes [' || hobbies || ']' FROM Employees WHERE
name='Shaya'
returns "Shaya likes []"- not ANSI-compliant

SQL Server:
   SET CONCAT_NULL_YIELDS_NULL ON
   SELECT 'Shaya likes [' + hobbies + ']' FROM Employees WHERE
name='Shaya'
returns NULL - ANSI-compliant
But
   SET CONCAT_NULL_YIELDS_NULL OFF
   SELECT 'Shaya likes [' + hobbies + ']' FROM Employees WHERE
name='Shaya'
returns "Shaya likes []"- not ANSI-compliant
```

Furthermore, Oracle states that this noncompliant behavior may change in future versions. Nonetheless, in my opinion, this noncompliant behavior is more user-friendly. Fortunately, though, there is a uniform way to be compliant as well as user-friendly in both databases using the COALESCE() function. The COALESCE() function takes a string of arguments and returns the first non-NULL argument. Thus, COALESCE(column_name,'') returns the column value if it is not NULL, and returns the second argument, the empty string, if the column is NULL. Rewriting the above queries as

```
Oracle:
SELECT 'Shaya likes [' || COALESCE(hobbies,'') || ']' FROM Employees
WHERE name='Shaya'

SQL Server:
SELECT 'Shaya likes [' + COALESCE(hobbies,'') + ']' FROM Employees
WHERE ID=0
```

returns the same result "Shaya likes []" in Oracle and SQL Server regardless of future versions or of the setting of CONCAT_NULL_YIELDS_NULL.

However, there's one more wrinkle to Oracle's noncompliance. In Oracle's own words, "Oracle currently treats a character value with a length of zero as null. However, this may not continue to be true in future releases, and Oracle recommends that you do not treat empty strings the same as nulls." Consequently, when a row is inserted or updated with an empty string, what actually gets stored in Oracle is a NULL. By ANSI standards, however, a NULL and an empty string should be distinct objects. In our Employees table example this means that the statement

```
INSERT INTO Employees(ID,Name,Salary,reportsTo,hobbies)
```

```
VALUES(8,'Jean' ,35000 ,6,'');
```

actually inserts a NULL into the hobbies column for Jean rather than an empty string. As a result, the following two selects

```
SELECT count(*) FROM Employees WHERE hobbies IS NULL;
SELECT count(*) FROM Employees WHERE hobbies = '';
```

will return 1 and 1 in SQL Server but 2 and 0 in Oracle, respectively. Regrettably, this is one of those issues for which there is no satisfactory workaround, and it's just something to be aware of when developing for different database platforms.

CASE and Relabeling Results

CASE is one of those SQL features that doesn't get used as much as it should. Here are just two of its uses. The following simple example shows how CASE can be used to label the results of a query.

```
SELECT name, salary,
   CASE
         WHEN salary >= 60000 THEN 'Manager'
         ELSE       'Grunt'
   END
   AS jobTitle
FROM Employees
WHERE name IS NOT NULL
```

returns

NAME	SALARY	JOBTITLE
George	100000	Manager
Ling	90000	Manager
Sandeep	60000	Manager
Manuel	30000	Grunt
Shaya	30000	Grunt
Akane	55000	Grunt
Bjorn	35000	Grunt
Jean	35000	Grunt

CASE and User-Defined Ordering

Another question that comes up in the ColdFusion mail list is: How do I return the results in an order that I define, which is neither alphabetical nor numeric? We'll use our Employees example to illustrate how this can be done. Suppose I want a list of all the employees ordered alphabetically, but I always want George to appear first. How do I accomplish that? I can do this by creating an ORDER BY attribute using CASE, which always places George first, as in the following query:

```
SELECT name FROM Employees
ORDER BY
   CASE
         WHEN name='George' THEN 1
         ELSE 2
```

```
    END, name;
```

which returns

```
NAME
George
Akane
Bjorn
Jean
Ling
Manuel
Sandeep
Shaya
```

Calculating Department Subtotals in a Single Query

Now, let's suppose we're given the following problem: retrieve a list of employees and their salaries sorted by department along with department salary subtotals in a single SQL query. While this is easy enough to do with a combination of SQL and CF, let's try to find a SQL-only solution.

It's fairly easy to retrieve a list of department subtotals. The query

```
SELECT dept, SUM(salary) FROM Employees GROUP BY dept
```

retrieves the subtotals

```
Finance    220000
IT  215000
```

However, we need to combine these results with the Employees table. This can be done with the UNION keyword, which simply combines the results of two queries. However, the use of UNION requires that the two queries match up exactly in column type and column order. Since we are retrieving name, salary, and department from Employees, we need to add an extra column to our subtotals query.

```
SELECT name, dept, salary FROM Employees
UNION
SELECT 'Total' as name, dept, SUM(salary) FROM Employees GROUP BY dept
```

The result of this query

```
NAME                DEPT       SALARY
Akane               IT         55000
Bjorn               IT         35000
George              Finance    100000
Jean                IT         35000
Ling                IT         90000
Manuel              Finance    30000
Sandeep             Finance    60000
Shaya               Finance    30000
Total               Finance    220000
Total               IT         215000
```

is almost what we want, but the order of the rows is wrong. We would like the employees to be ordered by department, and furthermore, have the department subtotals follow all the employees in the department. An ORDER BY department will take care of the first order condition, but how do we make the department subtotals follow the employees? We can take inspiration from how we used CASE for user-defined ordering by defining a new order_helper column to implement our ordering. The final query follows:

```
SELECT name, dept, salary, '1' AS order_helper
FROM Employees
UNION
SELECT ' Total' as name, dept, SUM(salary), '2' as order_helper
FROM Employees
GROUP BY dept
ORDER BY dept,order_helper;
```

returning in the desired order

```
NAME                DEPT                SALARY     ORDER_HELPER
George              Finance             100000                1
Manuel              Finance             30000                 1
Sandeep             Finance             60000                 1
Shaya               Finance             30000                 1
 Total              Finance             220000                2
Akane               IT                  55000                 1
Bjorn               IT                  35000                 1
Jean                IT                  35000                 1
Ling                IT                  90000                 1
 Total              IT                  215000                2
```

Use Stored Procedures

This final tip isn't about SQL difference, but rather a tip about SQL development in general. I strongly recommend as a best practice that, as much as possible, all SQL be removed from ColdFusion code and placed in stored procedures. There are two very good reasons for this, neither of which involve performance:

- ***Security:*** Stored procedures can be restricted in privileges. This means that a stored procedure with only select privileges can never be used to delete or insert rows in your table via an SQL injection attack.
- ***Centralized organization:*** Having all the SQL in a central location means that the SQL can be easily modified and tuned. For example, during the prototype phase before the database design has started, all the stored procedures can contain query stubs. Then, during development, the query stubs are replaced with actual working queries. Finally, during the tuning phase just before deployment, the working queries can be replaced by optimized or database-independent queries. By placing SQL in stored procedures, all this SQL work can proceed independently of the ColdFusion development.

Conclusion

While Oracle and SQL Server have greatly improved their compliance with ANSI standards, true universal SQL portability remains elusive. This has put off many developers from using

more advanced SQL due to compatibility concerns. Nonetheless, SQL is designed to manipulate query results and does so in a far more elegant and efficient manner than ColdFusion. Thus, it's best to retrieve the exact desired results from the database via correct SQL than to manipulate the results of a simple query via ColdFusion. Hopefully, this article has demonstrated some of the ways in which more portable SQL can increase the utilization of the database in CF applications.

```
Employees Table creation script:

CREATE TABLE Employees (
    ID              INTEGER,
    Name            CHAR(20),
    Salary          NUMBER,
    Dept            CHAR(15),
    hobbies         CHAR(50)
);
INSERT INTO Employees(ID,Name,Salary,dept,hobbies)
VALUES(1,'George',100000, 'Finance','golf');
INSERT INTO Employees(ID,Name,Salary, dept,hobbies)
VALUES(2,'Ling ',90000, 'IT' ,'golf');
INSERT INTO Employees(ID,Name,Salary, dept,hobbies)
VALUES(3,'Sandeep' ,60000, 'Finance' ,'tennis, volleyball, balloon-
ing');
INSERT INTO Employees(ID,Name,Salary, dept,hobbies)
VALUES(4,'Manuel ',30000 , 'Finance' ,'stamps');
INSERT INTO Employees(ID,Name,Salary, dept,hobbies)
VALUES(5,'Shaya',30000 , 'Finance',NULL);
INSERT INTO Employees(ID,Name,Salary, dept,hobbies)
VALUES(6,'Akane' ,55000 , 'IT','comic books');
INSERT INTO Employees(ID,Name,Salary, dept,hobbies)
VALUES(7, 'Bjorn',35000 ,'IT','cricket');
INSERT INTO Employees(ID,Name,Salary, dept,hobbies)
VALUES(8,'Jean' ,35000 , 'IT','');
```

Resources

- ***T-SQL reference at MSDN online:*** http://msdn.microsoft.com/library/default.asp?url=/library/en-us/tsqlref/ts_tsqlcon_6lyk.asp
- ***Oracle Technology Network online documentation (free registration required):*** http://otn.oracle.com/pls/db92/db92.homepage
- ***Oracle SQL Reference online (free registration required):*** http://download-west.oracle.com/docs/cd/B10501_01/server.920/a96540/toc.htm

About the Author

I-Lin Kuo is a developer at the Inter-University Consortium for Social and Political Research at the University of Michigan, Ann Arbor. He is also a coordinator of the Ann Arbor Java Users Group as well as a frequent contributor to the **CFDJ List** *mail list.*

cfrefactoring@yahoo.com

CFDJ Advertiser Index

Add Localization to Your Web Site

Adapting an application to a specific locale

By Raymond Camden

Welcome to another installment of **Extending ColdFusion**. In this edition, we are going to look at one of the ways you can add localization to your Web site. What do we mean by localization? Localization or L10N (L10N is an abbreviation for the 10 letters between the "L" and "N" in localization) describes the process of adapting an application to a specific locale.

You can think of L10N as the process of applying a locale or language "skin" to an application. That can mean many things like date, currency, and number formatting; calendars; text direction, and so on. Typically localization is done after a process called internationalization. Internationalization or I18N (I18N is an abbreviation for the 18 letters between the "I" and "N" in internationalization) is the design and development of an application so that it functions in at least two locales. You can think of I18N as making an application language or locale neutral.

In this article we are going to focus on one aspect of localization, and that is the site user interface and static text. In a later edition we will talk about how dynamic content can be localized. We are going to talk specifically to the concept of "resource bundles." These are files that contain a set of translations. For example, here is a representation of a simple English resource bundle:

```
Cancel=Cancel
Save=Save
```

Not very exciting, but the basic concept here is that we have a list of keys (the strings on the left side of the equal signs) and values (the strings on the right side). This becomes clearer when we look at the French version of the bundle:

```
Cancel=Annuler
Save= Enregistrer
```

The author wishes to give special thanks to Paul Hastings, who he says is, without a doubt, the localization/internationalization master, and was a big help in writing this article. His blog can be found at http://cfg11n.blogspot.com.
(Watch for a series by Paul in future issues of *CFDJ*.)

As you can see, we have the exact same keys, but the strings on the right-hand side, the actual values, are the French versions. Extending this a bit farther, you can use a larger set of keys and values, one key for each piece of text on your Web site. To add support for a new language, you simply create a new resource bundle and do the translations. You should understand that managing the translation of resource bundles can become quite a task for anything complicated. One of the more popular tools for this task is IBM's Java-based rbManager, which you can download from http://oss.software.ibm.com/icu4j/demo_tools/RBManager.html.

One note about resource bundles: we use a Java object, java.util.PropertyResourceBundle, to handle parsing in the resouce bundle. This Java object requires that the resource bundle files be encoded using escaped ASCII. For example, here is the Thai version of "Cancel": \u0E22\u0E01\u0E40\u0E25\u0E34\u0E0. Besides the rbManager tool mentioned above, Sun provides a command-line tool, native2ascii, that handles the creation of properly escaped Unicode text. This program is included in the "bin" directory of standard Java installs. You can find documentation for it at: http://java.sun.com/j2se/1.4.1/docs/tooldocs/windows/native2ascii.html.

How do we use this in ColdFusion? Listing 1 demonstrates a simple CFC that we will use to load resource bundles and display them on our Web site. Let's begin by looking at the getResourceBundle method. This method will take a resource bundle and return a structure of keys and values. The method takes two arguments. The first argument is the file name of the resource bundle. (We actually do some magic with this argument, which I will go into a bit later.) The second argument is optional, and represents the locale. Java uses a different format for locales than ColdFusion. Notice how the default is "en_US." In CFML, this is the same as English (US).

After initializing our arguments, we then have a set of var statements. Do not forget that it is vital to var scope any variable created with a CFC method. As I said above, we are going to do some magic with the resource bundle file name. If the file doesn't exist as passed, we check to see if a locale version exists. What do we mean by that? Imagine you have a set of resource bundles in your c:\projectX\resources\text folder. Each one is named "language_en_US.properties", where the only part of the name that changes is the locale version. In other words, the French version would be "language_fr_FR.-

properties." The getResourceBundle method allows us to simply pass in "c:\projectX\resources\text\language.properties" as the file name. When it discovers that the file does not exist, it will automatically try to load "c:\projectX\resource\text\language_en_US.properties". This is because the rbLocale argument defaults to en_US; to load the French version you would simply pass in fr_FR.

Once the method finds the file (it will throw an error if it cannot), we then use a set of Java objects we created in the constructor area of the CFC:

```
<cfscript>
variables.rB = createObject("java",
"java.util.PropertyResourceBundle");
variables.fis = CreateObject("java", "java.io.FileInputStream");
</cfscript>
```

These objects then enable getResourceBundle method to load the properties file:

```
variables.fis.init(arguments.rbFile);
```

and then initialize the PropertyResourceBundle object:

```
variables.rB.init(variables.fis);
```

Once we have done that, we can use another method, getKeys(), to get the keys of the resource bundle. We then simply loop over our keys and create a structure of the values. Once we have the structure we return it.

So, let's take a quick look at how this could be used on a sample page. Listing 2 contains a simple form, in this case, one with just a submit and cancel button. However, the labels for these buttons need to be localized. To get the correct strings, we use <cfinvoke> to get the resource bundle from a property file, and to specifically grab the fr_FR version. We can then use this returned structure to populate the values of our buttons. As you can see from the code, switching to English would be a trivial change.

Let's take a look at another method in the CFC, loadResourceBundle. This is an extremely short method. All it does is take a file and an optional locale. It calls the getResourceBundle method we described earlier, and then stores it in the CFC's local variable scope. This allows us to create an instance of the CFC and have it parse the resource bundle once.

Finally, the method getResource allows us to grab one particular string from the structure. Why would you use this when you could getResourceBundle? This method will slightly modify the result when debugMode is detected. Instead of returning just the string, it will actually return the string wrapped in * characters. Visually, this makes it easy to see which parts of your page are localized and which are not. This method will also "swallow" any errors that involve missing resources, so if you try to retrieve a resource for the French locale that doesn't exist, you will get a blank string. (This may or may not be preferable. What's nice is that you can modify the CFC to handle it differently. You may want it to retrieve the en_US version if one doesn't exist for the fr_FR version. You could also have it optionally log a warning, or e-mail a localization team.)

One more example before we end. In the code in Listing 2, we had a hard-coded locale. Normally you probably want your user to select a locale. You may think it's a good idea to store the resource bundle in the user's session scope. However, this means that every user in locale N will have a copy of the bundle, which isn't efficient. Listing 3 shows a modified version of Listing 2. This time we allow the user to select a locale. We store the user's preference, but keep the bundles stored in the application scope.

About the Author

Raymond Camden is co-technical editor of ColdFusion Developer's Journal *and a senior software engineer for Mindseye, Inc. A longtime ColdFusion user, Raymond is a co-author of the "Mastering ColdFusion" series published by Sybex Inc, as well as the lead author for the* ColdFusion MX Developer's Handbook. *He also presents at numerous conferences and contributes to online webzines. He and Rob Brooks-Bilson created and run the Common Function Library Project (www.cflib.org), an open source repository of ColdFusion UDFs. Raymond has helped form three ColdFusion User Groups and is the manager of the Acadiana MMUG.*

raymond@sys-con.com

Listing 1:

```
<!---
   This code is a modified version of the javaRB.cfc by Paul Hastings.
   You can find the original code + examples here:

   http://www.sustainablegis.com/unicode/resourceBundle/javaRB.cfm
--->

<cfcomponent displayname="resourceBundle" hint="Reads and parses resource
  bundle per locale" output="false">

   <cfscript>
   variables.rB = createObject("java",
    "java.util.PropertyResourceBundle");
   variables.fis = CreateObject("java", "java.io.FileInputStream");
   </cfscript>

   <cffunction name="getResource" access="public" output="false"
     returnType="string" hint="Returns bundle.X, if it exists, and
     optionally wraps it in ** if debug mode.">
          <cfargument name="resource" type="string" required="true">
          <cfargument name="rbFile" type="string" required="false">
          <cfargument name="rbLocale" type="string" required="false"
            default="en_US">

          <cfset var val = "">
          <cfset var rb = "">

          <!--- If passed a file name for the RB, load it. --->
          <cfif isDefined("arguments.rbFile")>
                    <cfset rb = getResourceBundle(arguments.rbFile,
                     arguments.rbLocale)>
          <!--- else, they MUST have used loadRB() --->
```

```
        <cfelseif not isDefined("variables.resourceBundle")>
                <cfthrow message="Fatal Error: Resource bundle file
                  must be passed to this method if
loadResourceBundle is not used.">
        <cfelse>
                <cfset rb = variables.resourceBundle>
        </cfif>


        <cfif not structKeyExists(rb, arguments.resource)>
                <cfset val = "">
        <cfelse>
                <cfset val = rb[arguments.resource]>
        </cfif>

        <cfif isDebugMode()>
                <cfset val = "*** #val# ***">
        </cfif>

        <cfreturn val>

   </cffunction>

   <cffunction name="getResourceBundle" access="public" output="false"
     returntype="struct" hint="Reads the RB into a struct">
        <cfargument name="rbFile" required="true" type="string"
          hint="The file name of the bundle.">
        <cfargument name="rbLocale" required="false" type="string"
          default="en_US">

        <cfset var resourceBundle = structNew()>
        <cfset var thisRBFile = "">
        <cfset var keys = "">
        <cfset var thisKey = "">
        <cfset var thisVal = "">

        <!--- Translate rbFile if it does not exist --->
        <cfif not fileExists(arguments.rbFile)>
                <!--- Try to change X.FOO to X_#argumentslocale#.FOO
                --->
                <cfset thisRBFile =
getFileFromPath(arguments.rbFile)>
                <cfset thisRBfile =
getDirectoryFromPath(arguments.rbFile) & listFirst(thisRBfile,".") & "_" &
arguments.rbLocale & "." & listLast(thisRBfile,".")>
                <cfif not fileExists(thisRBFile)>
                        <cfthrow message="Fatal Error: Resource
                          bundle file (#thisRBFile#) not found.">
                <cfelse>
                        <cfset arguments.rbFile = thisRBFile>
                </cfif>
        </cfif>

        <cfscript>
        variables.fis.init(arguments.rbFile);
        variables.rB.init(variables.fis);
        keys=variables.rB.getKeys();
        while (keys.hasMoreElements()) {
                thisKey=keys.nextElement();
                thisVal=rB.handleGetObject(thisKey);
                resourceBundle[thisKey]=thisVal;
        }
        fis.close();
        </cfscript>

        <cfreturn resourceBundle>
   </cffunction>

   <cffunction name="loadResourceBundle" access="public" output="false"
     returnType="void" hint="Loads a bundle for the persisted CFC">
        <cfargument name="rbFile" required="true" type="string">
        <cfargument name="rbLocale" required="false" type="string"
          default="en_US">

        <cfset variables.resourceBundle =
          getResourceBundle(arguments.rbFile,arguments.rbLocale)>
   </cffunction>

</cfcomponent>
```

Listing 2:

```
<cfinvoke component="resourcebundle" method="getResourceBundle"
  returnVariable="bundle">
   <cfinvokeargument name="rbFile" value="#expandPath("./testJavaRB.
     properties")#">
   <cfinvokeargument name="rbLocale" value="fr_FR">
</cfinvoke>

<form>
<cfoutput><input type="submit" value="#bundle.save#"> <input type="reset"
  value="#bundle.cancel#"></cfoutput>
</form>
```

Listing 3:

```
<cfapplication name="rbDemo" sessionmanagement="true">

<cfparam name="session.locale" default="en_US">

<cfif isDefined("form.newlocale")>
   <cfset session.locale = form.newlocale>
</cfif>

<cfif not isDefined("application.localeBundles")>
   <cfset application.localeBundles = structNew()>
</cfif>

<cfif not structKeyExists(application.localeBundles, session.locale)>
   <cfinvoke component="resourcebundle" method="getResourceBundle"
     returnVariable="bundle">
        <cfinvokeargument name="rbFile"
          value="#expandPath("./testJavaRB.properties")#">
        <cfinvokeargument name="rbLocale" value="#session.locale#">
   </cfinvoke>
   <cfset application.localeBundles[session.locale] = duplicate(bundle)>
<cfelse>
   <cfset bundle = application.localeBundles[session.locale]>
</cfif>

<cfoutput>
<form action="#cgi.script_name#" method="post">
<select name="newlocale">
   <option value="en_US" <cfif session.locale is
     "en_US">selected</cfif>>English</option>
   <option value="fr_FR" <cfif session.locale is
     "fr_FR">selected</cfif>>French</option>
</select><br>
<input type="submit" value="#bundle.save#"> <input type="reset"
  value="#bundle.cancel#">
</form>
</cfoutput>
```

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

A new tool for MX professional developers and designers...

MX developer's journal

4TH Annual International Deve

edge 2004 EAST

DEVELOPMENT TECHNOLOGIES EXCHANGE

February 24-26, 2004

Hynes Convention Center, Boston, MA

ARCHITECTING JAVA, .NET, WEB SERVICES, MX, XML, AND OPEN SOURCE

PROVIDING DEVELOPERS THE LATEST IN:

- Application Integration
- Desktop Java
- Mobility
- ASP.NET
- VS.NET
- JavaServer Faces
- SOA
- Interoperability
- Rich Internet Applications

Preregister for *FREE* full-day tutorials featuring the hottest Java, .NET, XML, Web Services, and Security issues (ADVANCED REGISTRATION REQUIRED)

FREE Exhibition Floor (ADVANCED REGISTRATION REQUIRED)
Meet with industry-leading companies showcasing the newest products and services.

- Product Demonstrations Theater
- Opening Night Reception
- Internet Café
- Case Study Forum

Register by FEBRUARY 21, 2004 SAVE Up To $200

MX WebSphere XML .net

For more information

visit www.sys-con.com/edge

call 201 802-3069

e-mail events@sys-con.com

Over 100 participating companies will display and demonstrate over 300 developer products and solutions.

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

Over 100 sessions – training, certifications, seminars, case studies, and panel discussions – promise to deliver real-world benefits, the industry pulse, and proven strategies.

Full-Day Tutorials Targeting

- Java
- .NET
- XML
- Web Services
- Open Source
- Security

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

HOSTED BY SYS-CON MEDIA

POWERED BY SEVENTH

eloper Conference & Expo

Attend a FREE One Day Security Tutorial Presented By

westbridge

Strategies for Web Services Security Success

Are you a developer, software architect, IT operator, or security administrator deploying or planning to deploy XML Web services? If so, this technical seminar is designed to arm you with the practical information and best practices you need to securely deploy XML Web services in your environment. Many questions will be addressed, such as: Why do Web services need special security? What standards are being created and what do they address? How do I leverage my existing environment to secure Web services? What are the different architectural and technical approaches to solving the problem? How can I centrally manage security in a decentralized environment?

Course Highlights/Benefits

- Overview of XML Web services security: Why is it important?
- Discussion of various standards (WS-security, SAML, XML-Enc, XML-Sig, XKMS)
- Architectural considerations
- Malicious Web services attacks
- Strategies for securing XML Web services today and in the future
- "Nuts and bolts" demonstrations of security solutions

As part of the tutorial, we will show you how to secure your entire XML environment without adding any additional code.

Seating is limited.
Full-conference attendees will receive priority seating for all tutorials; all other seating is on a first-come, first-served basis.

Attend a FREE One Day Tutorial Presented By

Microsoft

.NET–The Smart Client Perspective

For the business world, one of the most exciting promises of the information age is the opportunity to provide employees with access to information and the tools to act on that information whenever and wherever they need to. To a certain extent, that promise has already been achieved. Today, most businesspeople work on PCs that provide access to information, applications, and resources far beyond the boundaries of their local machine.

There are limits, however. Today's Internet model for information and application distribution assumes access to a network connection, but ubiquitous Web connectivity still lies in the future. And some computing tasks require robust functionality that can only be provided efficiently by "rich" client applications that reside on the local computer.

A challenge arises when your organization requires both the flexibility and immediacy that comes with online access to data and applications, and the full functionality of traditional client software.

The answer: smart client software.

This day-long tutorial focuses on developing and deploying smart client applications.

Seating is limited. Full conference attendees will receive priority seating for all tutorials; all other seating is on a first-come, first-served basis.

VISIT www.sys-con.com/edge FOR DETAILS and REGISTER TODAY

2004 EAST

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

JAVA SESSIONS

Aspect-Oriented Programming and Java

This session introduces Aspect-Oriented Programming (AOP) and how it applies to enterprise Java application development, with an emphasis on applications for service-oriented architectures such as Web services. AOP has become a major topic in the future of enterprise Java development. This session will present a conceptual road map, including tangible examples of how AOP works, and provide an understanding of both the potential and challenges of applying AOP in a J2EE context.

Squeezing Java

Java is a very powerful language; while it offers the developer a rich array of tools, the fundamentals mustn't be overlooked. Improving your code at the core layer will result in great improvements in efficiency and produce (hopefully) fewer bugs. We'll look at the do's and don'ts of programming and learn lots of hints and tips that will accelerate your Java coding.

Enterprise Architecture and Open Source

Use of open source software within the enterprise is gaining momentum. The vast majority of organizations are using some form of open source software in production environments, including Linux, Apache, and JBoss. The enterprise architecture, however, needs to incorporate the best thinking of the industry; this includes not only using open source but contributing to it.

The model in which open source software gets developed has practices that could assist an organization in becoming agile in their software development practices and allow them to develop software faster, with cheaper costs and better quality. In this session, you will learn:

- Two models of development: the cathedral and the bazaar
- The value proposition of using open source
- Harnessing the power of the mob: the value proposition of contributing to open source
- Making the build versus buy decision: additional thoughts

J2EE v1.4

Day-to-day work with deadlines makes it difficult to keep abreast of the rapidly evolving landscape of J2EE, especially given the numerous constituent J2EE technologies. J2EE v1.4 is chockful of new services that affect and benefit a wide range of enterprise development tasks. This talk will extract core material from the speaker's new J2EE Developer's Handbook and describe what's embodied in J2EE v1.4. In particular, the new Web services features provided by J2EE v1.4 will be highlighted. The talk will also briefly address those services missing from the current J2EE standards but still needed when building enterprise applications.

Apache Axis

Apache Axis is the popular SOAP engine that includes everything you need to start producing Web services. Discover just what Axis is, and how you can utilize the power of this free engine to kick-start your Web services.

Empowering Java and RSS for Blogging

One of the fastest-growing areas over the last few years is the blogging community. The ease with which you can post and publish information has enabled everyone to become their own publisher. One of the powers of blogs has been the syndication of data via the RSS (XML) protocol. Discover how you can easily produce and consume RSS feeds within your Java applications for wider appeal and hook into JavaBlogs, for example.

JUnit/Ant

A defined and easily repeatable process is one of the most necessary but often least used aspects of good software development. A defined build process ensures that your project's software is built, deployed, and tested identically each time. Without this type of control and predictability, valuable time is often lost chasing down bugs that don't exist or rejecting solutions that were only partially implemented.

A critical measure of the success of software can be found in whether or not it executes successfully. Equally important, however, is whether or not that software does what it was intended to do. JUnit is an open source testing framework that provides a simple means for developers to define their intentions of how their software should work. JUnit then provides test runners that process your intentions and verify that your code performs as intended. The result is software that not only works, but works in the correct way.

Apache's Ant is a powerful scripting tool that enables developers to define and execute routine software development tasks using the simplicity and extensibility of XML. Ant provides a comprehensive mechanism for managing software development projects, including compilation, deployment, testing, and execution. In addition, it is compatible with any IDE or operating system.

Next Phase in the Evolution of J2EE

J2EE has been making major inroads in the enterprise space for a number of years. However, it is only with the 1.4 release that we have had uniform and easy access to Web services. Discover how to leverage the new features of J2EE 1.4 and why this release is a significant milestone in the evolution of J2EE.

Simplifying J2EE Applications

J2EE is a large, complex specification for server-side, Web-enabled application development. Over the past few years, the presenter has led many teams through the J2EE jungle, trying to steer them away from the hype and keep them focused on delivering rock-solid, end-user applications. This tutorial will discuss a variety of tips, tricks, and lessons that he has learned so you and your teams can develop J2EE applications better, faster, and simpler than before.

WEB SERVICES SESSIONS

Exploring the Dark Side

The growing use of service-oriented architectures puts pressure on application developers relying on Web services for key features of their applications. Performance, scalability and reliability of these components affect the ability of applications to meet service-level agreements, yet can't easily be analyzed as a part of the application when developers have a problem. In fact, the Web service may be on a different software platform than the rest of the application. This session describes how developers can shed light on memory use in Web services written in either .NET or Java, even if they didn't write the code and do use another platform.

Web Services Progress Report

Web services have been the buzz for the last couple of years. There have been many technical and some market success stories but the concept remains confusing. New "standards" are proposed on a regular basis, but they overlap one another and seem to form rifts along the same fault lines as previous industry politico-strategic controversies. A group of people from the W3C Web Services Architecture working group have been arguing that many of the ideas coming from the Web services community are antithetical to the principles of the Web itself and are unlikely to ever work on an Internet scale. This presentation provides a progress report on the effort to distinguish Web services architectural principles from the marketing agenda of individual companies.

ID, Please. The Case for Giving Web Services an Identity

Without identity management, Web services can be consumed by anyone. The challenge for Web services developers is to provide appropriate access based on the user's identity. As identity management moves into the forefront of technology, directory services will evolve from simple LDAP repositories used for authentication and storage to robust engines that provide identity integration, access management, and policy enforcement. This presentation will discuss how identity management and directory services provides a robust solution for Web services authentication, authorization, and single sign-on.

Web Services Orchestration, Management, and Security: Will They Play Together?

Web services orchestration, management, and security are among the principal challenges facing implementers of service-oriented architectures today. There is still much confusion in the IT community about the standards themselves, which are at various stages of maturity. Their relevance to enterprise IT and how they might someday be able to effectively work together is often unclear. This session provides an overview of standards in these three critical areas; and more important, how each affects the other. Attendees will then gain practical knowledge and a deeper understanding of future trends and the need to address certain real-world issues in order to create a more cost-effective and agile IT infrastructure.

Service-Oriented Integration: Making the Right Choices To Support The Next-Generation of Integration

Applications are increasingly being developed "built to integrate," providing the ability to easily expose key functionality through commonly defined interfaces. Gartner calls this concept SODA, or service-oriented development of applications. When applied to the ever-present integration challenge, SODA represents a transition to service-oriented integration.

But making the right architectural decisions is absolutely vital to ensuring success with service-oriented integration projects – whether applications were built to integrate or not.

This presentation will examine the leading choices for supporting service-oriented integration: enterprise service buses, integration brokers, and application suite platforms.

Government Real-Time Fraud Detection Using Web Services

Government agencies are faced with increasing amounts of data and are challenged to make sense of, and act on, that data in real time. Failure to interpret and execute on data can result in security threats and potentially, loss of life. Government agencies are increasingly investing in Web services solutions to address their need for real-time access to information.

The Canadian Passport Office is an example of a government agency leveraging Web services to exchange information in real time to combat terrorism and other illicit uses of fraudulently obtained passports. They selected IT consulting firm Pentalar and Sybase, Inc. technologies to electronically authenticate identity document data through the use of Web services and ebXML.

This session will discuss this pilot project and highlight the ebXML capabilities that enable the Canadian Passport Office to address real-time information exchange.

WS-CAF: Standardized Web Services Transactions and Composite Applications

The Web Services Composite Application Framework is a collection of three specifications – Web Service Context (WS-CTX), Web Service Coordination Framework (WS-CF), and Web Service Transaction Management (WS-TXM) – designed to solve problems that arise when multiple Web services are used in combination ("composite applications") to support information sharing and transaction processing. As coauthor of the specification, we will discuss how WS-CAF addresses the underlying issues of Web service context propagation and transaction management to expand the scope, usability and reliability of Web services for business process automation.

Securing Web Services: What Can Be Done Today?

Security is listed as one of the main barriers to the adoption of Web services today. With the proliferation of security standards, there is a lot of confusion over which ones are mature enough to use and how they fit together. This session will present the current and emerging security standards for Web services and show how they can fit together architecturally to address various security concerns.

XML SESSIONS

Universal Business Language

Web service technologies promise to revolutionize electronic business, but global interoperability of business processes cannot occur without the semantic standardization of the messages exchanged in business transactions. This session will describe the OASIS UBL project to create standard XML Schemas for basic business documents, explore the relationship of UBL-based business to traditional EDI, and note the explosive potential of standard markup combined with reliable XML messaging.

Real Best Practices for XML Web Services Management and Security

Companies deploying Web services in a meaningful way are increasingly finding they need to address Web services management and security early in the architectural phase. Basic Web services connections are easy to do, but managing the security, performance, scalability, and inevitable changes to the production environment requires some knowledge, expertise, and planning. This session cuts through the hype and outlines real-world mistakes many companies make when deploying Web services and the real best practices from companies that have successfully captured the value of XML Web services. It provides practical advice on how to successfully manage and secure your XML Web services environment.

SOA Foundation Components: Building an XML Content Router

One of the fundamental components for any burgeoning SOA will be an XML content router. This session explores the concepts, patterns, and open source software available that facilitate building an XML content routing system. The system can be exposed as a Web service or simply as a standalone J2EE component for use in your enterprise. The 'restaurant' pattern is introduced as the principal design pattern for building the service, and this pattern's applicability to building generic services is discussed. Applying the router as an XML data integration tool is also discussed, as well as its potential for acting as a service orchestrator.

What's New in XSLT 2.0?

XSLT 2.0, which may achieve W3C Recommendation by conference time, offers unparalleled power in conjunction with XPath 2.0 for transforming XML documents. In this engaging, example-rich session, Steve Hedder demonstrates the most important new features of XSLT 2.0, including Sequences, new data types and XML Schema support, regular expressions, multiple document output, grouping, new control-flow operators, and much more. Current and future support for XSLT 2.0 on the Java and .NET platforms will also be discussed. Most examples will use Saxon, but .NET examples will be included if .NET supports XSLT 2.0 by conference time.

Using XML Schemas Effectively in WSDL Design

Developers building Web services today are beginning to see the value of using the document-style approach over RPC. Recent experience shows that to take full advantage of document-style Web services requires a strong knowledge of XML Schemas and related XML standards. This presentation presents a number of important tips and techniques for properly using XML Schemas in the design of a Web service interface (WSDL), including XML-based development tools, binding considerations between XML and underlying objects, WSDL reusability through XML Schemas, and XML Schema naming best practices.

XML: A Manager's Guide

As more and more IT projects utilize XML and its derivatives as fundamental technologies, it is key for today's manager to be aware of the various ingredients of XML. The objective of the session is to provide an essential introduction around XML from a manager's perspective. From core XML processing; transformation; metadata definition and schemas; applications in Web, wireless, and speech applications; Web services; industry-standard vocabularies; and more, this session provides a comprehensive review of the various technologies related to XML.

Using Rules to Clean Up XML

Garbage in, garbage out – it's an axiom that applies to many aspects of enterprise development, but none more so than building reliable and robust Web applications and integration projects with XML. Since its inception, XML has been seen as the cure-all for problems related to Web applications and integration projects. However, poorly written XML can slow down an integration project, or worse, cause the integration project to collapse. The key to successfully using XML in an integration project is to first understand the inefficiencies that may cause poorly written XML, and then apply a rule-based system that establishes policies to follow.

EXPO HALL
TUESDAY, FEBRUARY 24,
11:00AM-4:00PM, Welcome Reception 4:00PM
WEDNESDAY FEBRUARY 25,
11:00AM-4:00PM

XForms: Simplifying the Development of Transactional Web Forms

XForms is a W3C specification that specifies a declarative language for solving a common requirement for advanced user interaction, data validation, and XML processing. XForms is designed to be integrated into XHTML, but is not restricted to being a part of that language alone. It can be integrated into any suitable markup language. This session will give an introduction to XForms and explain how XForms fits in the client tier of the J2EE application architecture. In addition, it will cover the benefits of XForms and why it is a perfect fit for interacting with J2EE and Web services. A demonstration of XForms in a J2EE environment, using an XForms-compliant browser and a sample application, will further illustrate the advantages.

.NET SESSIONS

.NET Compact Framework Performance Tips and Tricks

Learn the techniques that can be used to increase the responsiveness of user interface and network operations for users of applications built on the .NET Compact Framework. Look under the covers at advances and changes in the "Whidbey" release to significantly improve performance. Get a general overview of how the .NET Compact Framework works under the hood at runtime, with specific focus on performance implications. Then we will cover general user interface tips to increase performance. Explore how asynchronous infrastructure, such as threading, in the .NET Compact Framework can be leveraged to optimize both user interface and network operations. Learn about the architectural guidelines for creating applications that perform well under frequently changing network conditions.

Best Practices and Techniques for Building Secure ASP.NET Applications

When the enterprise depends on your application, careful attention to security is essential. This session provides specific recommendations to follow when developing secure ASP.NET Web applications and services, and focuses on the details of configuring IIS for security. Understand how to use authentication, authorization, threat modeling, configuration settings, and secure database access to create secure systems and learn common coding techniques for storing secrets, error handling, data validation, and code access security.

Using the Enterprise Instrumentation Framework

The Microsoft .NET Framework 1.1 and Windows Server 2003 offer a number of new features to help developers instrument their code. In this session we'll learn about the challenges facing application management in today's distributed world. We will examine the new unified instrumentation API in the Enterprise Instrumentation Framework (EIF), including the new Windows Event Trace available in Windows Server 2003, configurable at-source event filtering, and how request-based event tracing using EIF allows you to put a request context around the trace messages that map to a business process flow in your application. We will also discuss the benefits of using EIF in your application for both the developer and the application administrator.

.NET Framework: Exploring What's New in the Base Class Library for "Whidbey"

The base classes serve as the essential libraries for any developer. Continued evolution of the base classes provides numerous benefits, including the ability to develop more reliable, faster solutions, easier-to-write code, and more solutions entirely in managed code. Take a look at the many features that are a part of that evolution, including features in IO, event logging, and various features in System.Collections.Generic classes and interfaces.

Microsoft Office 2003: A Solutions Platform

For all developers who would like to integrate custom business solutions with Microsoft Office products, this session will introduce you to the expanded developer features that have been included in the newest version of Microsoft Office. Come explore new XML-based programmability in everything from Word 2003 and Excel 2003 to FrontPage 2003 and SharePoint. Build powerful, modular solutions with Web services. Learn about InfoPath 2003 support for XML standards. Discover how to use the Microsoft Visual Studio Tools for the Microsoft Office System to automate and extend Microsoft Office Word 2003 and Microsoft Office Excel 2003 using Visual Basic .NET and Visual C# .NET. More than ever, Office has a solution for you.

BizTalk Server 2004 Technical Drilldown

BizTalk Server 2004 is designed to enhance Enterprise Application Integration (EAI), Business Process Automation (BPA), and Information Worker Integration. Join us for a technical drilldown into the new features and toolsets available.

Moving your Architecture to .NET

This session's emphasis is on how to migrate existing business components from VB6 COM objects to VB.NET assembly components. We'll spend time discovering how to best move different tiers of a multi-tiered application from COM to .NET, as well as effective strategies on how to wrap existing COM components for interoperability. We'll also examine best practices for moving your application from a COM-based architecture to a .NET-based architecture.

Who Should Attend

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

MX SESSIONS

Enterprise Infrastructure for Rich Internet Applications

Learn how Macromedia's technology initiative "Flex" fits seamlessly into today's new service-oriented architectures (SOA). We'll cover design patterns for rich clients, accessing Web services, and securing your Flex application.

ColdFusion Components from the Ground Up

ColdFusion Components (CFCs) are considered to be the most important enhancement to the CFML language since it was created some eight years ago. CFCs combine the power of objects with the simplicity of CFML, and in this session, you'll discover that not only are they incredibly powerful, they're also remarkably easy to master.

Code-Based Rich Internet Applications

Learn how to use Macromedia's technology initiative "Flex" to create rich Internet applications. This session will cover using components, layouts, and managers to build user interfaces, as well as using Flex's XML-based language to create and manipulate client-side data models.

Tips and Tricks for Writing and Using CFCs

ColdFusion Components are simple to write and simpler to use. But that simplicity hides a series of powerful features and technologies that you can (and should) take advantage of. In this session, you'll learn how to use (and how to not use) inheritance, "super" persistence, constructors, and more.

Leveraging Web Services

Web services technology is changing the way we think about designing and building applications. Come and learn what all the fuss is about, find out exactly which problems Web services solve, see Web services created and used, and even discover how Web services expose the world of .NET.

Building an RIA with Macromedia Flash and ColdFusion Web Services

Learn about the quickest and easiest way to build rich Internet applications using Macromedia Flash with connections to Web services built in ColdFusion.

ColdFusion Components

ColdFusion Components combine the power of objects with the simplicity of CFML. This is the way object-based development was intended to be, and in this session, you'll learn about this combination first hand. Starting with a simple data-driven application, you'll gradually convert it into a highly scalable and manageable multitier application, and in the process, will be amazed at just how easy ColdFusion makes this process.

Rapidly Build Web Services Applications with ColdFusion and Studio MX

The last year has shown that Web services are not just another passing fad and their promise of platform-independent distributed applications has been realized. Compared to other application server platforms, ColdFusion makes creating Web services easy. This session covers how to create a ColdFusion Component (CFC) in Dreamweaver, as well as how to expose that CFC as a Web service by just toggling one attribute of the CFC. That's right: in ColdFusion, it is just that easy.

Using Macromedia Flash with Web Services

Web services, a technology that allows developers to execute remote procedures, is emerging as a revolutionary tool for Web application development. Macromedia Flash MX 2004 Professional is a powerful tool for building applications that consume Web services built in any technology, including Macromedia ColdFusion, Java, ASP.NET, and PHP. In this session you will explore the vision any computing model that Web services represent as you use Macromedia Flash components to develop a Web service–based application. You will learn how to discover Web services, work with data and UI components, perform data binding, examine security issues, and aggregate multiple Web services into a cutting-edge Web service consumer.

Using Web Services with ColdFusion

You're a ColdFusion user and want to be able to take advantage of Web services? You're in luck. No other language or platform makes Web services consumption as painless as ColdFusion does. In this hands-on session, you'll experience Web services for yourself by building a complete application around some of the most popular Web services available today.

Sheraton Boston Hotel

Official Hotel of Edge 2004 East

REGISTRATION FORM

Edge 2004 East
Development Technologies Exchange

CONFERENCE: Feb. 24 – 26, 2004 EXPO: Feb. 24 – 25, 2004

Hynes Convention Center, Boston, MA

THREE WAYS TO REGISTER FOR CONFERENCE

1) On the Web: Credit Card or "Bill Me" Please make checks payable to SYS-CON Events.
2) By Fax: Credit Card or "Bill Me" 201-782-9651
3) By Mail: 135 Chestnut Ridge Road, Montvale, New Jersey 07645, Attention: Registration

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

Please complete sections 1, 2, 3 and 4

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

First Name _____ Last Name _____

Title _____

Company _____

Street _____

Mail Stop _____

City _____

State _____ Zip _____ Country _____

Phone _____

Fax _____ E-Mail _____

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

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

Check #: Amount of Check $

Charge my: ☐ Visa ☐ MasterCard ☐ American Express ☐ Discover

Name on Card

Card # Exp. Date

Signature

Billing Address (if different from mailing address)

3 PLEASE INDICATE YOUR CONFERENCE CHOICE

Total Registration Fee $.

	before 01/9/04	before 02/6/04	before 02/20/04	Onsite
☐ GP: Gold Passport Three-Day Conference Good for all three days of the conference program, including keynotes, all conference sessions & one tutorial	$1,395.00	$1,495.00	$1,595.00	$1,795.00
☐ 3D: Three Day Conference Good for all three days of the conference program, including keynotes, all conference sessions (excludes tutorials).	$1,295.00	$1,395.00	$1,495.00	$1,695.00
☐ 2D: Two Day Conference Good for two days of the conference program, including keynotes, all conference sessions (excludes tutorials).	$995.00	$1,095.00	$1,295.00	$1,395.00
☐ 1D: One Day Conference Good for one day of the conference program, including keynotes, all conference sessions (excludes tutorials).	$595.00	$695.00	$795.00	$795.00
☐ VIP PASS FREE with preregistration Select one: ☐ FREE - The Smart Client Perspective (Feb. 24) ☐ FREE - Strategies for Web Services Security Success (Feb. 25)	FREE	FREE	FREE	$99.00
☐ EO: Expo Hall Only Includes keynotes, expo presentations, and workshops.	FREE	FREE	FREE	$50.00

SPECIAL DISCOUNTS AVAILABLE

Take advantage of the Early Bird and Preregistration values available right now, or save even more with a group of 5 or more. For special group discounts contact Michael Lynch at mike@sys-con.com, or by phone at (201) 802-3068.

4

A. Your Job Title
☐ CTO, CIO, VP, Chief Architect
☐ Software Development Director/Manager/Evangelist
☐ IT Director/Manager
☐ Project Manager/Project Leader/Group Leader
☐ Software Architect/Systems Analyst
☐ Application Programmer/Evangelist
☐ Database Administrator/Programmer
☐ Software Developer/System Integrator/Consultant
☐ Web Programmer
☐ CEO/COO/President/Chairman/Owner/Partner
☐ VP/Director/Manager Marketing, Sales
☐ VP/Director/Manager of Product Development
☐ General Division Manager/Department Manager
☐ Other (please specify) _____

B. Business Industry

☐ Computer Software
☐ Computer Hardware and Electronics
☐ Computer Networking & Telecommunications
☐ Internet/Web/E-commerce
☐ Consulting & Systems Integrator
☐ Financial Services
☐ Manufacturing
☐ Wholesale/Retail/Distribution
☐ Transportation
☐ Travel/Hospitality
☐ Government/Military/Aerospace
☐ Health Care/Medical
☐ Insurance/Legal
☐ Education
☐ Utilities
☐ Architecture/Construction/Real Estate
☐ Agriculture
☐ Nonprofit/Religious
☐ Other (please specify) _____

C. Total number of employees at your location and entire organization (check all that apply):

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

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

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

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

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

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

G. Which of the following products, services, and/or technologies do you currently approve, specify or recommend the purchase of?
☐ Application Servers
☐ Web Servers
☐ Server-side Hardware
☐ Client-side Hardware
☐ Wireless Device Hardware
☐ Databases
☐ Java IDEs
☐ Class Libraries
☐ Software Testing Tools
☐ Web Testing Tools
☐ Modeling Tools
☐ Team Development Tools
☐ Installation Tools
☐ Frameworks
☐ Database Access Tools/EJB Servers
☐ Application Integration Tools
☐ Enterprise Development Tool Suites
☐ Messaging Tools
☐ Reporting Tools
☐ Debugging Tools
☐ Virtual Machines
☐ Wireless Development Tools
☐ XML Tools
☐ Web Services Development Toolkits
☐ Professional Training Services
☐ Other (please specify) _____

SYS-CON Events, Inc. and SYS-CON Media make no warranties regarding content, speakers, or attendance. The opinions of speakers, exhibitors, and sponsors do not reflect the opinion of SYS-CON Events and SYS-CON Media, and no endorsement of speakers, exhibitors, companies, products, or sponsors is implied.

If you require special assistance covered under the Americans with Disabilities Act, please call 201 802 3068 by February 10, 2004

CANCELLATIONS, SUBSTITUTIONS, REFUNDS

Any written request to SYS-CON Registration 201 782 9651. Requests for refunds received prior to January 16, 2004 will be honored less a 10% handling charge; requests received after January 16, 2004 and before February 6, 2004, will be honored less a 25% handling charge. No requests for refunds will be honored after February 6, 2004. Requests for substitutions must be made in writing prior to February 20, 2004. No one under 18 permitted to attend. No commitments are made regarding the content of sessions or materials. Speakers, sessions, and schedules are subject to change without prior notice.

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

ColdFusion User Groups

For more information go to...
http://www.macromedia.com/cfusion/usergroups

New England

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

Massachusetts
Boston, MA CFUG
www.cfugboston.org

Rhode Island
Providence, RI CFUG
www.ricfug.com

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

Midatlantic

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

New York
Albany, NY CFUG
www.anycfug.org

New York
Long Island, NY CFUG
www.licfug.org

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

New York
Rochester, NY CFUG
www.roch-cfug.org

New York
Syracuse, NY CFUG
www.cfugcny.org

Pennsylvania, Harrisburg
Central Penn CFUG
www.centralpenncfug.org

Pennsylvania
Philadelphia, PA CFUG
www.pacfug.org

Pennsylvania
Pittsburgh, PA CFUG
www.orbwave.com/pghcfug

Pennsylvania
State College, PA CFUG
www.cfug-sc.org

Southern

Alabama
Birmingham, AL CFUG
www.bcfug.org

Alabama
Huntsville, AL CFUG
www.nacfug.com

Delaware, Dover
Delaware CFUG
www.decfug.org

Delmarva, Dover
Delmarva CFUG
www.delmarva-cfug.org

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

Florida
Orlando, FL CFUG
www.cforlando.com

Florida
Tallahassee, FL CFUG
www.tcfug.com

Florida
Tampa, FL CFUG
www.tbcfug.org

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

Georgia, Atlanta
Atlanta, GA CFUG
www.acfug.org

Georgia, Atlanta
Georgia CFUG
www.cfugorama.com

Georgia
Columbus, GA CFUG
www.vcfug.org

Kentucky
Louisville, KY CFUG
www.loulexcfug.com

Louisiana
New Orleans, LA CFUG
www.nocfug.org

Maryland
Annapolis, MD CFUG
www.ancfug.com

Maryland
Baltimore, MD CFUG
www.cfugorama.com

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

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

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

North Carolina
Charlotte, NC CFUG
www.charlotte-cfug.org

North Carolina
Fayetteville, NC CFUG
www.schoollink.net/fcfug/

North Carolina
Raleigh, NC CFUG
www.ccfug.org

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

Oklahoma
Tulsa, OK MMUG
www.tulsacfug.org

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

Tennessee
Nashville, TN CFUG
www.ncfug.org

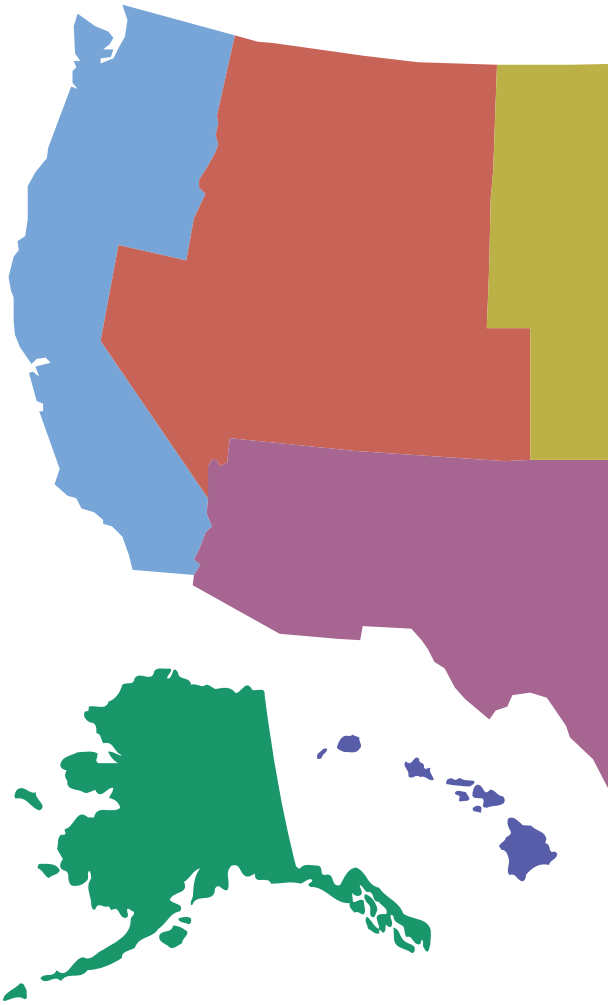
Virginia
Hampton Roads CFUG
www.hrcfug.org

Virginia
Northern Virginia CFUG
www.cfugorama.com

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

Virginia, Roanoke
Blue Ridge MMUG
www.brmug.com

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



Midwest

Northern Colorado
Northern Colorado CFUG
www.nccfug.com

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

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

Illinois, Chicago
Chicago, IL CFUG
www.cfugorama.com

Indiana
Indianapolis, IN CFUG
www.hoosierfusion.com

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

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

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

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

Minnesota, Minneapolis
Twin Cities CFUG
www.colderfusion.com

Rockies

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

Nevada
Las Vegas, NV CFUG
www.sncfug.com

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

Wyoming, Jackson
Wyoming MMUG
www.wycfug.org

West Coast

California
Bay Area CFUG
www.bacfug.net

California, Fresno
Central California CFUG
www.cccfug.com

California, Inland Empire,
Inland Empire CFUG
www.sccfug.org

California
Los Angeles CFUG
www.sccfug.org

California
Orange County CFUG
www.sccfug.org

California
Sacramento, CA CFUG
www.saccfug.org

California
San Diego, CA CFUG
www.sdcfug.org

Southern California
Southern California CFUG
www.sccfug.org

Oregon
Eugene, OR CFUG
www.EugeneCFUG.org

Oregon
Portland, OR CFUG
www.pdxcfug.org

Alaska

Alaska
Anchorage, AK CFUG
www.akcfug.org

Hawaii

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

Missouri
Kansas City, MO CFUG
www.kcfusion.org

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

Nebraska
Omaha, NE CFUG
www.necfug.com

Ohio, Columbus
Ohio Area CFUG
www.oacfug.org

Ohio
Mid Ohio Valley MMUG
www.movcfug.org

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

Southwest

Arizona
Phoenix, AZ CFUG
www.azcfug.com

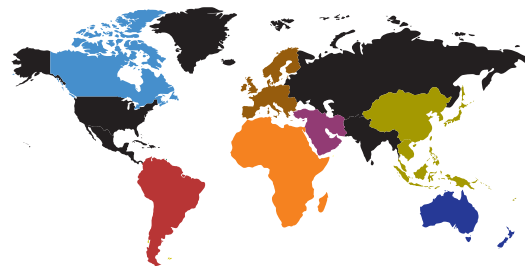
Arizona
Tucson, AZ CFUG
www.tucsoncfug.org

Texas
Austin, TX CFUG
http://cftexas.net

Texas
Dallas, TX CFUG
www.dfwcfug.org

Texas
San Antonio, TX CFUG
http://samcfug.org

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.



Africa

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

South Africa, Johannesburg
Johannesburg, South Africa CFUG
www.coldfusion.org.za

Asia

Malaysia, kuala lumpur
Malaysia CFUG
www.coldfusioneer.com

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

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

Australia

Australia-Melbourne
Australian CFUGs
www.cfug.org.au

Canada

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

Canada
Kingston, ON CFUG
www.kcfug.org

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

Canada
Ottawa, ON CFUG
www.cfugottawa.com

Canada, Ottawa (HS group)
Ecole Secondary CFUG
www.escgarneau.com/gug

Canada
Toronto, ON CFUG
www.cfugtoronto.org

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

Canada
Vancouver Island CFUG
www.cfug-vancouverisland.com

Europe

Belgium, Brussels
Belgium CFUG
www.cfug.be

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

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

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

Germany, Frankfurt
Frankfurt CFUG
www.cfug-frankfurt.de

Ireland
Belfast, Ireland CFUG
www.cfug.ie

Ireland
Cork, Ireland CFUG
http://viewmylist.com/cork

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

Sweden
Gothenburg, Sweden CFUG
www.cfug-se.org

Switzerland
Martin Bürlimann, Switzerland CFUG
www.cfug.ch

United Kingdom, London
UK CFUG
www.ukcfug.org

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

Middle East

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

Pakistan, Lahore
Pakistan CFUG
www.cfugpakistan.org

Saudi Arabia, Riyadh
CFUG Saudia
www.cfugsaudia.org

Turkey, Izmer
Turkey CFUG
www.cftr.net

South America

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

Macromedia Announces Macromedia Director MX 2004

(San Francisco) – Macromedia has announced Macromedia Director MX 2004, the latest version of the industry-standard multimedia authoring tool. Macromedia Director has been at the forefront of multimedia development for more than a decade, and this latest version adds support for JavaScript, Flash MX 2004 content, DVD-Video, and the ability to create projector files for both Mac and Windows platforms in one simple step. Macromedia Director MX 2004 enables developers to unleash their creativity and build rich, interactive experiences that deliver measurable results.

"Macromedia Director defined both this company and this industry, and it remains the most sophisticated multimedia authoring tool to this day," said Norm Meyrowitz, president of products, Macromedia. "Macromedia Director MX 2004 builds on the product's rich legacy and adds new features that will enable richer content and easier development due to its seamless workflow with other MX 2004 products and the addition of an industry-standard, widely used scripting language."

Macromedia Director MX 2004 empowers professional multimedia developers, advanced Flash developers, professional DVD developers, e-learning designers, and game developers who need to create multimedia content that can be deployed anywhere, whether it be CD, DVD, intranets, kiosks, or the Internet. Director MX 2004 supports most major video, audio, bitmap, 3D, and vector formats to give developers the broadest content palette from which to deliver sophisticated, compelling user experiences. Extensive video capabilities within Director MX 2004 allow developers to stream video files in DVD-Video, Windows Media, RealMedia, QuickTime, and Flash formats.

Macromedia Director MX 2004 is also tightly integrated with other products and servers in the Macromedia MX family. In addition to adding support for Flash MX 2004 content and applications, Director also has the ability to launch and edit both Flash and Fireworks to enable a seamless workflow. Director also includes support for Flash MX 2004 components, including user interface components. Flash content can be integrated into Director projects, and Flash performance within Director projects has been dramatically improved over previous versions.

The product has also added new features to make developers more efficient and productive. Interactivity within projects can now be scripted directly using JavaScript, Lingo, or a combination of both. Content can be published to both Macintosh and Windows, and online across different platforms, in one simple step. The ability to name sprites and channels enabled workflow efficiencies for the production of content within Director. For a detailed list of new features in Director MX 2004, visit www.macromedia.com/go/dirmx2004new.

Macromedia Director MX 2004 is expected to ship in English in February, and is priced at $1,199 for new users, and $399 for upgrades from Director 8.5 and Director MX. Education, government, and volume licensing options are available. The software works on Mac OS X v10.2.6 or higher, and for Windows 2000 or Windows XP. French, German, and Japanese versions will be available at a later date.

SYS-CON Media Premieres MX Developer's Journal at MAX

(*Montvale, NJ*) – *MX Developer's Journal*, SYS-CON Media's latest highly anticipated new title, launched on September 10, 2003, reaching more than 2 million Macromedia MX developers using Flash, Dreamweaver, Fireworks, FreeHand, ColdFusion, and Director. Targeted at the professional developers and designers who use the award-winning Macromedia MX product family to build Web sites and applications, *MXDJ* is now available on newsstands and in specialty bookstores such as Barnes & Noble and Borders. Every issue includes:

- Hands-on tutorials covering every aspect of Flash MX, Dreamweaver MX, Fireworks MX, FreeHand MX, ColdFusion MX, and Director MX
- Insightful technical articles and feature stories on how best to harness the MX family of technologies
- How-to programming tips/code listings, and more...

A special U.S. charter subscription rate is being offered to *CFDJ* readers, with substantial savings off newsstand prices. For more information, see www.sys-con.com/mx/charter.cfm, or call 1 (888) 303-5282.

Former Microsoft Executive and Symbian Cofounder Juha Christensen to Lead Macromedia Mobile and Devices Division

(*San Francisco*) – Macromedia has announced the appointment of Juha Christensen as president of its mobile and devices business unit. Christensen recently served as corporate vice president of the mobile devices division at Microsoft Corporation, building Microsoft's franchise in the mobile industry. Before joining Microsoft, Christensen cofounded Symbian Ltd., a joint venture between Nokia, Ericsson, Sony-Ericsson, Matsushita, Samsung, Siemens, and Psion, whose cross-vendor operating system is emerging as an industry standard. Previously Christensen worked at Psion Plc., bringing out some of the first PDAs in the industry.

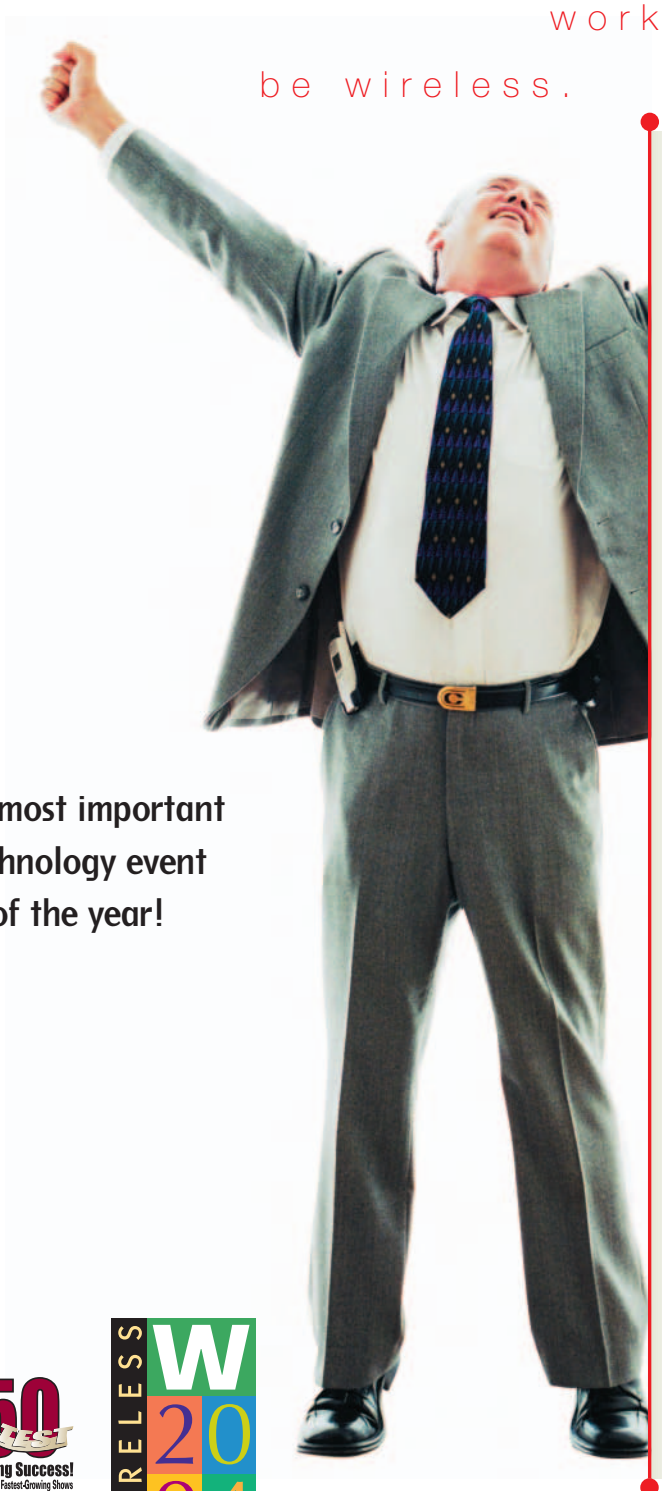
"Juha Christensen brings entrepreneurial leadership, an amazing track record with mobility, and clear vision to our growing mobile and devices business," said Rob Burgess, chairman and CEO, Macromedia. "As president of this important division, Juha will help Macromedia lead the field in delivering innovative and practical solutions for this fast-growing industry."

While at Microsoft, Christensen brought to market the Pocket PC, Pocket PC Phone Edition, and Smartphone, in addition to Mobile Information Server and Server ActiveSync. He was also responsible for Microsoft's participation in several worldwide wireless standards bodies and developed the strategy for company-wide mobility solutions and scenarios. During his time at Symbian, Christensen wrote the company's first business plan, which defined the company's overall strategy, vision, and operational platform.

live wireless.

work wireless.

be wireless.



The most important technology event of the year!

CTIA WIRELESS 2004

is the one show where wireless standards are created and the technological direction of the industry is set. With the largest gathering of wireless engineers and technologists, this is where you will find the tools you need to help build and advance the wireless industry.

Look at all CTIA WIRELESS 2004 has to offer the wireless engineer and technologist:

- *6 CTIA educational sessions dedicated to exploring wireless technology*
- *IEEE Wireless Communications Network Conference (WCNC) 2004 – the industry's foremost conference for developing wireless standards and engineering*
- *WiFi Summit – the CTIA Smart Pass program, a cutting edge look at WiFi strategy and security*
- *A 400,000 square foot exhibit floor displaying the latest in wireless technology and applications*

CTIA Global

TSW 50 Fastest Celebrating Success! Tradeshow Week's 50 Fastest-Growing Shows

Produced by CTIA

WIRELESS W 2004 CTIA

welcome the wireless generation.

March 22-24, 2004 Georgia World Congress Center Atlanta, GA, USA www.ctiashow.com

John T. Chambers
President & CEO
Cisco Systems

Scott McNealy
Chairman & CEO
Sun Microsystems, Inc.

Russell Simmons
Chairman & CEO, Rush Communications,
Co-founder & Chairman, Def Jam Records

be the pilot!

Intermedia.NET is the #1 ColdFusion Hosting Provider*

FREE SETUP on Shared Hosting Accounts With ColdFusion MX Support

Use Promo Code CFDJ2003

INTERMEDIA.NET

WE DARE YOU TO TAKE A FREE TEST FLIGHT!

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

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

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

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

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

Managed Hosting • Shared Hosting • Microsoft Exchange Hosting

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

*CFHub.com