

MX

developer's journal

THE LEADING MAGAZINE FOR
MACROMEDIA MX DEVELOPMENT,
DESIGN, & PRODUCTION TOOLS

volume 3 issue 6 2005 www.mxdj.com

NET FLASH



DREAMWEAVER

Standards-Compliant Web Pages



FIREWORKS

Sailing the Wine Dark Sea



COLDFUSION

Getting Started with <CFLDAP>



CAPTIVATE

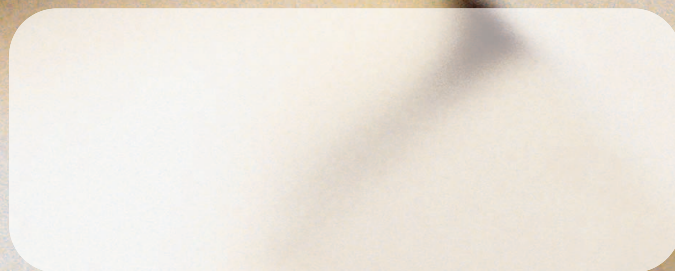
Working with Audio



FLEX

Animating Flex Charts

A NEW TOOL LETS
.NET DEVELOPERS
WRITE FLASH APPS



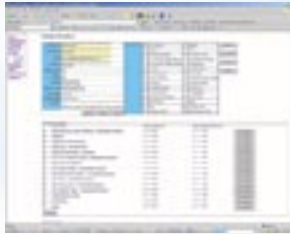


MX Kollection 3

A new era in web development

4 of the **12.226 applications** developed with MX Kollection

Intranets



E-Commerce



CMS

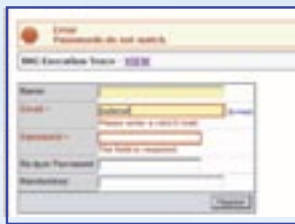


Image Gallery



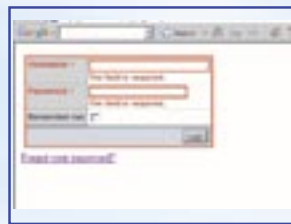
12 of the **423 features** included in MX Kollection

Form Validation



Validate form fields
Client and server side validation
Rich validation formats library
Preserve submitted values on error
Date picker, Numeric control
Masked Textfield, Restricted Textarea

User Login



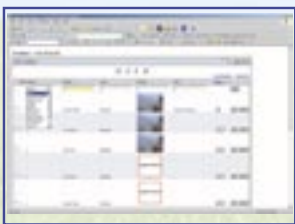
User registration and login
Encrypted password
"Remember me" feature
Account activation by e-mail
Restrict access to page
Forgot password feature

Rich Internet Applications



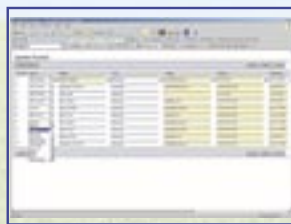
Create incredibly good looking forms
Date Picker
Dependent Drop-downs
Combo-box
Masked textfield
Numeric textfield

Manage your database



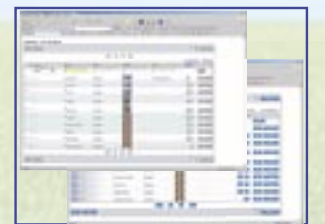
Search records in the database
Insert/Update/Delete forms
Table and CSS form generation
Automatic navigation bars
Automatic list filter
Order your database information

Edit Multiple Records at once



Excel-like multiple records form
Select Records from the list
Edit records in an HTML form
Insert multiple records at once
Delete multiple records directly from list

CSS skins for custom look



4 predefined CSS skins
Create your own skins with ease
Define fonts, colors and layouts
Duplicate edit buttons on top

Interakt



macromedia
approved



macromedia®
ALLIANCE PARTNER

1 of the **5628 customers** working with MX Kollektion

“ You cannot regard this product as an extension or an add-on or even an upgrade. I regard it as a new era in RAPID dynamic web development. You will need to come up with a new term for Dreamweaver add-ons. It definitely borders on the lines of a CASE (computer aided software engineering) tool.”

Michael Hamlett, web developer

Dreamweaver Wizards



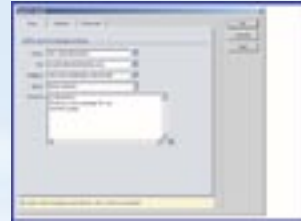
- Step by step wizards
- Contextual help in each interface
- Multi-tabbed server behaviors
- Control Panel to manage your site
- Automatic included files update

Upload Files and Images



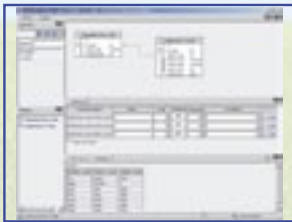
- File and Image Upload
- Proportional Image resize
- Impose maximum file size
- Dynamic Thumbnail Creation
- Download Uploaded File
- Delete files on record delete

Send E-mails



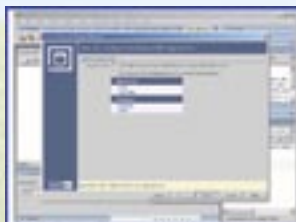
- Send e-mail on form submit
- Send e-mail to multiple recipients
- Send page section by e-mail
- CSS and image links in e-mail body
- Set e-mail priority

Create SQL Queries Visually



- Create, Edit recordsets visually
- Visually add tables to query
- JOINS between tables
- Define and apply complex conditions
- Extract information from multiple tables
- Synchronize with database changes

Horizontal loopers



- Horizontal looper
- Vertical looper
- Nested repeat regions
- Create image galleries or reports

Documentation & tutorials



- CHM Documentation with search
- In-depth tutorials
- Online troubleshooter
- Online knowledge base
- How-to section make problems history

work smart

<http://www.interaktonline.com/>



22



.NET in a Flash
A new tool lets .NET developers write Flash apps
by jim phelan



30



Using the Singleton Design Pattern
A standardized solution to a common programming problem
by danny patterson



34



Sailing the Wine Dark Sea
A Fireworks masking project
by jim babbage

7

No More Updates Ever Again
by charles e. brown

8

The MX Blogosphere
An online stroll
by mxdj news desk

50

vanguard
Interactive Ambience
by ambient interactive



16

MX Kollection 3 by InterAKT Online
Some User Perspectives
by nancy gill, et al.



40

Getting started with <CFLDAP>
A step-by-step guide to the basics
by tony paolillo



18

Creating Standards-Compliant Web Pages
Future-proofing your code
by kevin ruse



42

Working with Audio
The Hollywood effect
by lisa heselton



28

Using Electric Rain's Swift 3D 4.5
It rocks!
by matthew david



44

Animating Flex Charts
Enter another dimension
by peter ent



48

Let Flex Consume Your CFCs
A more programmatic approach to building RIAs
by dennis baldwin

ARE YOU READY FOR

MX7

WE ARE

CFDynamics

- Reseller program
- State of the art data center
- Comprehensive on line support
- 24/7 network monitoring

CFDynamics, an innovative leader in ColdFusion hosting, caters to professional developers. We're here to help our clients use this brand new technology. While other developers waited patiently for the new version of ColdFusion, our clientele has already experienced powerful new features such as a full-feature reporting engine and flash forms through free hosting of Blackstone. CFDynamics is now ready for you to experience Coldfusion MX7.

Contact CFDynamics at 1.866.233.9626 and **experience Coldfusion MX7.**

*Blackstone is the code name for MX 7. Free BlackStone hosting is no longer available because ColdFusion MX7 is now released.



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

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

Surround SCM lets you...

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

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

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

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

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

Remotely access your source code repository from Dreamweaver MX.

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

Features:

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

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

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

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

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

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

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

Want to learn more?

Visit www.seapine.com/webdev to discover how Surround SCM can save you time and headaches. Be sure to download our white paper, **Change Management and Dreamweaver**, and learn why change management is the new must-have tool for web development.



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



macromedia
ALLIANCE PARTNER



Group Publisher Jeremy Geelan
Art Director Louis F. Cuffari

EDITORIAL BOARD

Editor-in-Chief

Charles E. Brown charles@sys-con.com

Dreamweaver Editor

Dave McFarland

Flash Editor

Brian Eubanks

Fireworks Editor

Joyce J. Evans

FreeHand Editor

Ron Rockwell

Louis F. Cuffari

Director Editor

Andrew Phelps

Captivate Editor

Tom Green

INTERNATIONAL ADVISORY BOARD

Jens Christian Brynildsen **Norway**,

David Hurrows **UK**, Joshua Davis **USA**,

Jon Gay **USA**, Craig Goodman **USA**,

Phillip Kerman **USA**, Danny Mavromatis **USA**,

Colin Moock **Canada**, Jesse Nieminen **USA**,

Gary Rosenzweig **USA**, John Tidwell **USA**

EDITORIAL

Editor

Nancy Valentine, 201 802-3044

nancy@sys-con.com

Associate Editor

Seta Papazian, 201 802-3052

seta@sys-con.com

Technical Editors

Jesse Warden • Sarge Sargent

To submit a proposal for an article, go to <http://grids.sys-con.com/proposal>.

Subscriptions

E-mail: subscribe@sys-con.com

U.S. Toll Free: 888 303-5282

International: 201 802-3012

Fax: 201 782-9600

Cover Price U.S. \$5.99

U.S. \$29.99 (12 issues/1 year)

Canada/Mexico: \$49.99/year

International: \$59.99/year

Credit Card, U.S. Banks or Money Orders

Back Issues: \$12/each

Editorial and Advertising Offices

Postmaster: Send all address changes to:

SYS-CON Media

135 Chestnut Ridge Rd.

Montvale, NJ 07645

Worldwide Newsstand Distribution

Curtis Circulation Company, New Milford, NJ

List Rental Information

Kevin Collopy: 845 731-2684,

kevin.collopy@edithroman.com,

Frank Cipolla: 845 731-3832, frank.cipolla@epostdirect.com

Promotional Reprints

Dorothy Gil, 201 802-3024

dorothy@sys-con.com

Copyright © 2005

by SYS-CON Publications, Inc. All rights reserved. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopy or any information storage and retrieval system, without written permission.

MX Developer's Journal (ISSN#1546-2242)

is published monthly (12 times a year) by

SYS-CON Publications, Inc., 135 Chestnut

Ridge Road, Montvale, NJ 07645.

SYS-CON Media and SYS-CON Publications, Inc., reserve the right to revise, republish, and authorize its readers to use the articles submitted for publication. MX and MX-based marks are trademarks or registered trademarks of Macromedia, in the United States and other countries. SYS-CON Publications, Inc., is independent of Macromedia. All brand and product names used on these pages are trade names, service marks or trademarks of their respective companies.

MX
developer's journal

No More Updates Ever Again

by charles e. brown

Let me paint a hypothetical situation. There will never be another update of any software package again! All software companies feel they have gone as far as they can go!

Ok, let me give you a chance to restore your heart rhythms.

Each year I see a recurring situation. Software Company "A" releases a major upgrade to their package. Everyone is excited about the new features. Then, usually within a month, you start seeing postings about what should be put into the next update. Are we ever happy?

Now, let's go back to our opening scenario. Stop and look at your own workflow. How much of Flash's firepower do you use? Do you use all of the features of Dreamweaver? Are you using, perhaps, only 25% of Fireworks' capability?

Why do we constantly clamor for newer features when we are barely using what we have now?

As an MM Certified Instructor, I see how underutilized by most people our present tools are. I see many people still doing simple, static websites in Dreamweaver and some basic animations in Flash. As a matter of fact, in a class last week, I had a student say that they had not realized that Flash could do anything more than merely a few animations.

As I am writing this editorial, I am sitting in EPCOT center at Walt Disney World in Orlando, Florida. If you read Disney's biography, you would see that as early as 1936, he was experimenting with surround-sound systems in movie theatres for the animated movie Fantasia. He certainly pushed the animation process light-years ahead.

I admire people, such as Disney, who can take very little and yet create it into something new and innovative. But, if you think about it, the present state of our software tools exceeds anything we dreamed of just a few years ago. Yet, despite that fact, we are constantly clam-

oring for more. Is it because our creative visions exceed the tools that we have? Or is it because we want the software to give us creative ideas?

I don't think I am going too far out on a limb in saying that the Macromedia-Adobe combo will give us very powerful tools. I can only imagine the capabilities of Flash and After Effects coming together. But are we ready for it?

We need, as a community, to sit down and take a hard look at ourselves and our creative capabilities. I firmly feel that education will play an increasingly important role in this process. To that end, I very much applaud Macromedia's Jon Lloyd and Sue Hove. Jon is running the certified instructor program and Sue works ceaselessly to constantly improve the certified educational material. People like them will help match the skills of the community with the capabilities of the software. I only hope that they will be given the opportunity to continue their fine work after the merger is completed. To that end, I will lobby hard.

While we are speaking about education, I strongly recommend that everyone consider attending the annual MAX conference. This year's conference is in Anaheim, CA (home of the original Disney park) and will be October 16th - 19th.

If you examine the events at www.macromedia.com/macromedia/events/max/, you will quickly see that there are workshops for every skill level. Whether you are a beginner or advanced, plan on coming to the conference.

We had a great time last year in New Orleans. I, for one, found it invigorating and I picked up a LOT of new ideas. Also, plan on bringing tons of business cards. This is a great chance to network.

On a final note, keep an eye open for the August issue. It will be loaded with surprises! ☺



Charles E. Brown is the editor-in-chief of MX Developer's Journal. He is the coauthor of Fireworks MX, Zero to Hero and the author of Beginning Dreamweaver MX. He also contributed to The Macromedia Studio MX Bible. Charles is a senior trainer for FMC on the MX product family. charles@sys-con.com



The MX Blogosphere

This month's edition of our regular feature taking an online stroll around the MX world
by mxdj news desk

If one of the true signs of a vibrant developer community is an active blogosphere surrounding a technology, then the MX suite of technologies certainly passes that test with flying colors. Don't forget that you can blog yourself now, too, under the MXDJ domain – just follow the blog-n-play link from <http://mxdj.sys-con.com>.

Blog Topic: Flash

Macromedia Relaunches Flash as a Platform

by **oscar trelles**
from www.oscar-trelles.com/blog/

In a move that is being received with a smile by the open-source-friendly segment of the Flash development community, Macromedia announced that they will be joining the Eclipse Foundation to create a “next-generation rich Internet application development tool,” codenamed Zorn and based on Eclipse. Furthermore, Macromedia launched a new website and a series of whitepapers forecasting the future of the Flash Platform.

For many of us, the idea of Flash as a viable development platform for RIAs has been a reality for some time. Now, Macromedia is putting it in black-

and-white. Built around the strengths of the Flash Player, Macromedia has built a universe of development tools, server-side technologies, solutions and partnerships that affect directly the way we do business. With what is being announced for Maelstrom (Flash Player 8), Macromedia has proved again that they listen to the community. Some of the stuff coming in the next generation of the Flash player, according to today's announcements:

- File upload and download
- Support for GIF, PNG and Progressive JPEG image formats
- Better font readability
- Enhanced Video codecs

All of this, together with the ability of developing Flash applications within an open-source environment such as Eclipse, which has been embraced by an important segment of the Flash community, is enough to have everybody happy and waiting to see what else will come after Maelstrom is released.

Blog Topic: Flex

Using Shared Fonts in Flex

by **darron schall**
from [/www.darron-schall.com/weblog/](http://www.darron-schall.com/weblog/)

It turns out you can't directly embed a font into a Remote Shared Library (RSL) using Flex. Here's a handy little trick to get around that...

First, some background. I have a Flex application that consists of a main .swf that loads and manages several other “load on demand” .swf files. A user of the application doesn't necessarily need to use all parts of the application, so this modular approach allows them to only download the parts they'll be using. I'm using shared libraries to solve the problem of downloading the same information more than one time when using multiple .swfs in one application.

For more information on RSLs in general, check our Roger Gonzalez's great article on the subject (www.macromedia.com/devnet/flex/articles/rsl.html).

So I wanted to add a font to an RSL for a few reasons. The font should be able to be changed in only one place, it should only have to be downloaded one time, and all of the loaded .swf files should use the same font. I tried just using the embed tag to add it to an RSL, like this:

```
<embed source="../../../fonts/Arial.ttf"
newSymbol="mainFont" />
```

When I went to compile though I was getting a failed to load resource error. Apparently you can't add .ttf files to shared libraries.

I then tried to embed the font directly into the main .swf file. The css for that looks like this:

```
@font-face {
src: url("../fonts/Arial.ttf");
fontFamily: mainFont;
}
Application {
fontFamily: mainFont;
}
```

The problem with the css approach is that the font was only appearing in the main .swf file. All of the other .swf files were attempting to use the font, but the font couldn't be found (as demonstrated by all of the text being a small serif font). So, I needed a way to reference the font from the loaded .swfs so that they would use it correctly.

The workaround for this problem, suggested by Roger, is so simple that it's perfect. Create a component that embeds the font via css, stuff that component into a shared library, and then reference the component from every .swf that needs the font. By taking this approach if I ever wanted to update the font I would simple

“Macromedia has proved again that they listen to the community”

update the shared library, and all of the application would use it.

So, I made a simple component, in Fonts.mxml:

```
<mx:Canvas xmlns:mx="http://www.macromedia.com/2003/mxml">

<mx:Style>
<![CDATA[
@font-face {
src: url("../fonts/Arial.ttf");
fontFamily: mainFont;
}
]]>
</mx:Style>

</mx:Canvas>
...added that component to one of my
shared libraries:
// inside the "library" tags:
<component name="Fonts" uri="*" />
... added some css for the main .swf
file:
global {
fontFamily: mainFont;
}
```

... and then referenced the shared Fonts component inside every .swf file:

```
<local:Fonts width="0" height="0" />
```

I'm using width and height of 0 so that it doesn't actually draw anything (which would affect the layout). I suppose visible="false" would work just as well. But, by using the Fonts component, every single .swf will link to the font from the shared library, and because the global font family specifies that as the font to use, all of the text will be displayed with that font.

Good stuff... Flex handles shared libraries a million times easier than Flash does. I was pleasantly surprised to see this work, to say the least, especially thinking back to how much tweaking the Flash version of the same approach took. Go Flex go!

Blog Topic: Flash

Parent SWF Security Sandbox & Zinc

by jesse randall warden

from www.jessewarden.com/

Deploying a project yesterday and doing the unthinkable, loadMovie. A parent SWF loads a child SWF after a

successful login. These child SWFs are basically applications and the shell that loads them is just a security mechanism. We're using Zinc, and we originally we're just compiling this "shell" natively; so the EXE would be a small exe the user's download to their computer, but the SWFs remain on the server that the shell actually downloads, making it easier to keep up to date without having to distribute a new EXE since the shell just logs in and downloads SWFs.

Anyway, during testing we kept getting security sandbox violations. I was extremely confused since Flash as a projector or SWF running on the desktop has free reign and can do anything it wants, when it wants. I figured maybe because the new Flashout (0.2.1.2) now hosts the Flash Player in a browser window, it's bound by domain restrictions (even localhost).

...But a Zinc projector? It's an exe right? Guess not. Even using the option of Zinc to merely load the SWF from a remote URL instead of compiling the SWF in didn't work; same issue.

So, reading the docs found 2 good examples of when to use System.security.allowDomain for SWF's loading SWF's. Our's was the 2nd, where a shell SWF loads other SWFs into it, and wants to be able to interact with them; like set variables on it when it's loaded, call methods, etc. So, we did:

```
System.security.allowDomain(_parent._url);
```

in the children SWFs. You can be sure _parent's already loaded

(because the _parent loaded you), and the _url of the parent could change, but this is the exact way without hardcoding it. Works like a charm.

Still, I'm confused what Zinc is doing, making the SWFs act like they are in a browser. As an exe, I am not held to the security sandbox.

Blog Topic: Fireworks & Photoshop

Fireworks Can Do That Too!

by jim babbage

from <http://mxdj.sys-con.com/read/rss/97.htm>

I picked up a book last week at



ILLUSTRATION BY FIDEL RODRIGUEZ



Costco, in preparation for the fall semester. It's called Top 100 Simplified Tips and Tricks for Photoshop CS. Yes, I teach PS as well as Fireworks. A lot of my students will be involved in print work if they get jobs in their chosen careers of Advertising or Journalism.

Anyway, while I have not quite finished the text, I am very pleased to see that a great deal of the tips in this book could also be done quite as easily in FW - sometimes more easily. Granted, some of the stuff just will not apply as FW doesn't understand the CMYK color space. And that is fine by me; MM was always very clear that FW was a screen graphics app. I prefer a program that does one thing really well. Don't get me started on ImageReady .

Blog Topic: Flex

Flex mx:HTTPService Tricks

by **theodore patrick**

from www.power-sdk.com/ted/

HTTPService is a very powerful tag in MXML. I overlooked it for a while until I found some tricks to using it. Here is my list:

1. *HTTPService processes the data returned.*

If you load XML using HTTPService, by default it will return an object heierarchy even when useProxy='false'. Once

the data had returned you can use dot notation to retrieve nodes by name.

```
myXMLService.result.frog.dog.blog
```

A handy use of this is to set dataProviders directly. Using any List based control, you can set the dataProvider from an XML file. This is super handy given with one HTTPService XML request and setting one variable you can add complex data into a control.

Warning: There is a gotcha in here. When XML is parsed to Object, similar node names at the same level are turned into an array. To access nodes of a common name you use name[1]. The problem is that names do not become nested unless there is more than one. If you set dataProvider values this way, when 1 element is set, it will mess-up. Although you can filter for this using the length property, it will be undefined when there is only 1 node.

There are some other formats supported but I have yet to explore them. You can set the HTTPService.resultFormat property to either "object" (the default!), "xml", "flashvars" or "text". Depending on what your doing these can be very useful.

2. *Use Databinding with the URL property.*

This one is really handy and can dynamically generate GET URLs for you based on local variables.

```
<mx:HTTPService id="documentService"
useProxy="false" url="{ 'http://
server/' + docPath + '/data.xml' }"
result="documentData=documentService.
result" />
```

This defines the documentService. When documentService.send() is called, the URL is calculated based on the current value of docPath and the request is made. When data returns, the returned object is saved to the variable 'documentData'.

In using this, I would do this a ton:

```
docPath = "frog/dog"
documentService.send()
```

3. *Caching data between requests.*

```
<mx:HTTPService id="documentService"
```

```
useProxy="false" url="{ 'http://serv-
er/document.cfm?id='+ docID }" result
="documentCache[docID]=documentServic
e.result;documentServiceResult()" />
```

In this case, I use a variable called docID to vary the data returned from the HTTPService. With each request, I save a local copy according to the docID value. When calling this I use the following code:

```
if( documentCache[docID] == undefined
or ){
documentService.send()
}else{
documentServiceResult()
}
```

Instead of retrieving the data every time from the server, this only fetches the data once and caches it inside the player locally. Subsequent requests are pulled from the local cache.

4. *useProxy='false'*

This is a big deal since it allows Flash to use the players security model for loading data from another server. Just add a crossdomain.xml security file to the root of all servers you want to access and you can exchange data with another domain. Here is a sample cross-domain.xml file:

```
http://www.macromedia.com/crossdo-
main.xml
```

```
http://www.yahoo.com/crossdomain.xml
```

As I dig deeper into the practical side of Flex development, I will post some more.

Using these you can exchange data with the best of them!

Blog Topic: Flash Lite

Nokia Announces Seven New Flash Enabled Phones

by **daniel dura**

from www.daniel-dura.com/

There is no doubt that the number of Flash enabled handsets has increased dramatically during the last 12 months. The ramifications for Flash developers are very clear. More Flash Enabled Phones = Greater Demand for Flash

Four of the models use the 'Matrix-style' sliding design, that features a keyboard which slides out from under the display.

CommonSpot™

Efficient Content Management

fast. easy. affordable.



- 100 % browser-based
- Content object architecture
- Template driven pages
- 50+ standard elements
- Extensible via ColdFusion
- Content reuse
- Content scheduling
- Flexible workflow
- Granular security
- CSS support
- 508 compliance
- Personalization
- Replication
- Custom metadata
- Custom authentication
- Static site generation
- Multilanguage support

With CommonSpot Content Server you get it all. CommonSpot's exceptional blend of rapid deployment, ease of use, customization and scalability make it the leading ColdFusion content management solution.

Our rich Web publishing framework empowers developers with out-of-the-box features such as template-driven pages, custom content objects, granular security, flexible workflow and powerful ColdFusion integration hooks (just to name a few), allowing you to rapidly build and efficiently deploy dynamic, high performance Web sites.

For larger implementations, CommonSpot scales efficiently, delivering enterprise-level capabilities like replication, static content generation, multi-site clustering, personalization and custom authentication, across a diverse set of platforms.

For the past six years, PaperThin has been a leader in the ColdFusion community, and CommonSpot has been the solution of choice for organizations of all sizes, including AFL-CIO, Boeing, Kaiser Family Foundation, Ohio University, PGA.com and hundreds of others. CommonSpot's sophisticated feature set and affordable pricing are an unbeatable combination.

Call us today at [800.940.3087](tel:800.940.3087) to schedule a live demonstration of your site running under CommonSpot, or visit www.paperthin.com to learn more.

Paper | Thin



Developers. If you are interested in developing for Flash Lite enabled handsets, download the free Flash Lite CDK (www.macromedia.com/cfusion/entitlement/index.cfm?e=flashcdk) or visit the [Flash Lite product page \(http://www.macromedia.com/software/flashlite/\)](http://www.macromedia.com/software/flashlite/).

Via Digital-Lifestyles.info: "In a veritable orgy of mobile manufacturing, Nokia has launched a grand total of seven new handsets, including a dual camera 3G device and their most feature-rich CDMA phone yet.

Four of the models use the 'Matrix-style' sliding design, which features a keyboard which slides out from under the display.

The phones are powered by the somewhat elderly Series 40 user interface, although the latest version of the O/S cranks up the eye candy, adds a new "Active Standby" home screen and bundles in support for applications written in Macromedia Flash Lite."

Blog Topic: ColdFusion

Java vs. ColdFusion - String Concatenation and CSV Generation

By Darry Lyons
from dangermoose.blogspot.com

For a little while now I've been playing around with Java, and trying to replicate some common ColdFusion concepts in the language. Recently I came across a problem where generating a CSV file was taking a very long time, often making the server unresponsive.

The code in question built a large string using concatenation, and then wrote the string to a file. It occurred to me that in-line Java might be a bet-

ter tool for the job. A work colleague suggested using the StringBuffer/ StringBuilder (5.0) class, as it is much more efficient than the String class.

As you will see below, I ultimately ended up using the BufferedWriter class, and the execution time went from 5 minutes to 40 seconds!

I decided to run some tests, to see how much quicker (or slower) in-line Java is than ColdFusion.

The two tests are:

- ColdFusion vs. Java: String concatenation
- ColdFusion vs. Java: CSV generation

I created a test script that appended a string to an existing string 1000 or 10,000 (or more) times. I also created a set of test scripts that built a string and/or appended a string to a CSV file.

I concluded that Java is a lot better at string concatenation than ColdFusion. This is especially true when dealing with very large strings. I deliberately did not test the CFSAVECONTENT tag as it is not a string concatenation method.

When concatenating a string 1000 times, there wasn't too much between them. ColdFusion took 172ms, whereas the Java class took 78ms. However, when increasing to 10,000 iterations, ColdFusion took 17547ms (17s), and Java only took 406ms!

I didn't dare test ColdFusion any further, and I know from experience that it would have taken quite some time.

I pitted Java's BufferedWriter class against ColdFusion's CFSAVECONTENT and CFFILE tags. I used the CFSAVECONTENT tag, because it is a lot quicker than concatenation. The Java test simply appended data to a file as it went, whereas the ColdFusion test built the string using CFSAVECONTENT and then wrote it to a file.

Ultimately I concluded that the BufferedWriter class is a lot more efficient than the ColdFusion method. Building a 1000 row CSV file took Java 15ms, whereas ColdFusion took 47ms. When building a 10,000 row CSV file, the gap increased -- Java 64ms, ColdFusion 328ms. A 1,000,000 row file 49.6MB took Java 6547ms, whereas ColdFusion took 10860ms.

In the 1,000,000 row test,

the amount of memory used by CFSAVECONTENT was quite high. The server became sluggish after this, and memory did not seem to be freed. Using the BufferedWriter class however, you could barely notice any memory usage (as you would expect).

The good thing about using the BufferedWriter class is that it uses very little memory. As soon as the initial buffer size (8192 characters) is reached, it is written to the file, and the buffer (memory) is then flushed/released.

It is a fair statement that if you're generating a CSV file, then you should use Java. It is a much more scalable solution.

Java StringBuffer class

```
oStringBuffer = CreateObject("java",  
"java.lang.StringBuffer").init(JavaCast("int",initSize));
```

```
// Append a string  
oStringBuffer.append("A string");
```

```
// Get string back from buffer  
sString = oStringBuffer.toString();
```

Java BufferedWriter class

```
oFileWriter = CreateObject("java",  
"java.io.FileWriter").init("filename",  
JavaCast("boolean","true"));
```

```
oBufferedWriter =  
CreateObject("java",  
"java.io.BufferedWriter").  
init(oFileWriter);
```

```
// Write a string to buffer  
oBufferedWriter.write("A string");
```

Blog Topic: Flash

Beware of Automatic Octal Number Conversions

by [erik bianchi](http://erikbianchi.com/)
from www.erikbianchi.com/

When does 21 equal 17? When using Flash to convert numbers with a leading 0!

Apparently Flash will automatically convert strings starting with a leading 0 to octal numbers (base 8). However this isn't specifically mentioned in Flash's documentation for Number conversions (that I could find anyway) but is prob-

"If you're generating a CSV file, then you should use Java"

ably an ECMA standard.

To see for yourself:

```
var index="021";
trace(Number(index)); // output 17
It comes to this result by multi-
plying each value by 8: ((8*0)+1) +
(8*2) = 17
```

To compensate for this conversation I wrote a normalize to integer function that subsequently ignores any leading zeros:

```
// normalize to integer
function normalizeInt(value:String):
Number{
while(value.charAt(0) == "0"){
value = value.substring(1,value.
length);
}

return Number(value);
}
var index = normalizeInt("021");
trace(index);
```

Just something to keep in mind should you ever have a spec (as I did)

that required a fixed length integer that could start with a 0.

Blog Topic: Flash & Eclipse A Review of FDT, the Commercial Flash Eclipse Plugin

By Chafic Kazoun
from www.rewindlife.com/

I had the chance to spend a bit of time playing with FDT (www.powerflasher.com/fdt/flashsite/flash.htm) this weekend, a commercial Eclipse plugin in development and wanted to share some thoughts.

I've been an avid PrimalScript (www.sapien.com/primalscript.aspx) user for a while now, I won't get into PrimalScript vs FDT discussion now as FDT is still in beta and so is PrimalScript 4.x. I also don't want to get into open-source vs commercial products. I am a firm believer that if a tool meets my needs and speeds up my workflow then the cost is usually negligible. At the end of the day it will always depend on your workflow/needs.

This isn't the first time I use Eclipse, I've used it for Java work in the past and used it with ASDT (although I haven't

been satisfied with it and have stuck with PrimalScript for my ActionScript editing needs). Eclipse is a great IDE with lots of built in functionality (Version Control Integration, a strong community of plug-ins, etc), and since FDT is built on top of Eclipse, you gain the benefits of an existing infrastructure.

Now onto the real reason you are here, how is FDT? There is a lot to like about FDT. First is its integration with Eclipse and how far along it is in comparison to ASDT. The power flasher (<http://powerflasher.com/>) guys spent some good time on this and it shows. The good integration with Eclipse and Flash, the Flash Explorer, SWF Viewer, ease of compilation with Flash and MTASC, a good editor, help integration, and syntax checker. If you are using ASDT today and want to stay in Eclipse then FDT is exactly what you are looking for. With FDT there is no need to mess around with configuring Eclipse for Flash Development, just install FDT and you're pretty much there.

I also found myself discovering Eclipse all over again with FDT. I customized a lot of the shortcuts, the perspec-

ONCE YOU'RE IN IT...

ONCE YOU'RE IN IT...



- WebServices
- Storage & Security
- IT Solutions Guide
- PowerBuilder
- ColdFusion
- WebSphere
- LinuxWorld
- Wireless
- WLDJ
- .NET
- JDJ
- MX

...REPRINT IT



Contact Kristin Kuhnle
201.802.3026
kristin@sys-con.com



RePrints

Advertising Index

Advertiser	URL	Phone	Page
CFDynamics	www.cfdynamics.com	866-233-9626	5
CFUNITED	www.cfunitied.com	301-424-3903	51
Clearnova	www.clearnova.com		33
Flashforward	www.flashforwardconference.com	877-4FLASH4	27
Hosting.com	www.hosting.com		39
HostMySite.com	www.hostmysite.com	877-248-4678	15
InterAKT	www.interaktonline.com		2 - 3
Macromedia	www.macromedia.com/go/video5	415-252-2000	52
PaperThin	www.paperthin.com	800-940-3087	11
Seapine Software	www.seapine.com/webdev	888-683-6456	6
SYS-CON Newsletters	www.sys-con.com	888-303-5282	46
SYS-CON Reprints	dorothy@sys-con.com	201-802-3024	13
Web Services Edge 2005	www.sys-con.com/Edge2005	201-802-3066	47
WebAppCabaret	www.webappcabaret.com/cdj.jsp	866-256-7973	21

I haven't built anything from zero using FDT, just worked on some existing projects which has worked well so far

tives, and found a workflow in Eclipse that worked well for me. I do miss a couple of things in PrimalScript, the biggest of which is the performance of code-hinting which PrimalScript is really good at. Also PrimalScript generally included some useful shortcuts built-in that I've grown accustomed to, but maybe there is a way for me to get the same in Eclipse that I haven't discovered yet.

Again for now I don't want to make any judgments. I haven't built anything from zero using FDT, just worked on some existing projects which has worked well so far. Maybe I'll post more as I get some more time to work with FDT. There is a lot a lot to like about FDT, and probably a lot of it has to do with Eclipse. Being able to checkout code

directly from Subversion, perform some changes, build (FDT makes this simple), then commit my changes all without leaving the tool is great. I can see this really improving team workflow as well as the live syntax checking. PrimalScript still seems to be better on the code-hinting side and generally is faster/more light-weight than Eclipse (not FDT's fault), although it currently doesn't provide any syntax checking mechanism.

Overall I feel more confident now about Macromedia's decision to adopt Eclipse as its development platform for their next generation RIA tool, ZORN (<http://eclipse.sys-con.com/read/96869.htm>). They will have the chance to re-use so much that is already there and solid in Eclipse. ☺

“My new home is at HostMySite,
they have been incredibly
helpful and dedicated, and
went the extra step to make the
transition, well, painless.”

- Ben Forta

- ▶ Plans starting from 1GB HD storage and 20GB transfer
- ▶ Custom tags supported
- ▶ Full account management with HostMySite's Control Panel
- ▶ Call 24x7x365 for expert ColdFusion support

Visit us at CFUnited June 29th—July 1st

Go to hostmysite.com/cfdj for 2 Months Free Hosting

And \$100 off a CFUnited Conference Pass

 **HostMySite.com**

1-877-215-4678

d

f

fu

f

d

fx

MX Kollektion 3 by InterAKT Online

Some User Perspectives

by nancy gill, joyce evans, barbara o'neal and steven redden

Web development is a changing industry. Technologies are born, thrive, and then die, while web developers experience a great stress helping their clients get an Internet presence. Nowadays, the buzz word in web development is "dynamic." Not "dynamic" as in DHTML, but as in "using a database."

MX Kollektion is designed for HTML designers that want to take a plunge into database-driven development and here four of the product's beta testers report briefly on how they feel about the product.

CSS Skins

California developer and Team Macromedia member Nancy Gill reports that it's not the many functions of MX Kollektion that are her favorite as much as the simple addition of skinning the interface by site.

"I can choose Aqua for the tool's look in Site A and go on to define Site B to show the Arctic skin," she writes. "The advantage in that is that I have so many similar folders from site to site that I won't get confused about where I'm working today. It helps me sort my work flow like never before."

"The ability to fine-tune my settings on a site to site basis," Gill continues, "gives me the kind of control that I have never seen before. I can change the date format, the language for the labels and warnings and other parameters, such as mail server and such with extreme ease."

Barbara O'Neal singles out the Query Builder for praise: "I can build complex queries I would have to use Enterprise Manager to build otherwise. And this means I don't have to leave Dreamweaver MX 2004 to do it ... It's all right here." A

bonus is the ability to create a query and click on the Create CFC button to store the query in an independent document, making reuse of the same functionality instantaneous, something ColdFusion gurus have come to love in their chosen server model.

Moving along to the many wizards that power the MX Kollektion 3, Nancy Gill reckons that while some may have issues with feeling a wizard "dumbs them down", she thinks this approach is great for getting up to speed with the tool.

"Click, click in answer to the questions and it's all created for you," Gill says. Making it easy to work with and easier to get the job done faster, she adds.

Building CMS Websites

Steve Redden, Technical Director of 3-wise Internet Solutions (www.3-wise.com/), reported on MX Kollektion as follows.

"Time and time again, we demo Content Management Systems and web-based applications to potential clients only to see looks of wonder on their faces when MX Kollektion 3 allows me conjure up lists that can filter, sort and re-order at the click of a mouse. NeXTensio has been our bread and butter application for the past three years in the construction of lists and create, modify and delete transactions in my developments."

Redden's clients are delighted to find positive answers to most of their questions and requirements, he says.

Editing Multiple Records at Once

There are wizards in MX Kollektion for everything from insert, update and delete forms, to the neat displaying of data. The wizards allow users to build pages for inserting and editing multiple records at the same time.

Barbara O'Neal says her favorite is the "Insert into Two Tables" wizard since this was an operation in ColdFusion or any other server model that required the functionality to be hand coded in the past. "Specify your tables, bind the appropriate fields and it's all created for you like magic," she notes.

Form Validation

As for form validation, "This is the best I've seen... it's so complete," says O'Neal. "There is point and click validation for everything I can imagine." Barbara adds that her business partner is a stickler for details and in looking over her site created with MX Kollektion validation, he couldn't find anything wrong with it. "That's nothing short of a miracle" she gushed. "Nothing gets by him, believe me."

Documentation and Support

"I'm not only a Web Designer but I teach new designers and developers so I always look for tools to make their jobs easier," said noted Dreamweaver and Fireworks author and instructor Joyce Evans, adding that what she likes about the MX Kollektion 3 is how the wizards "really help the beginner developer get up to speed quickly and produce complex applications in no time at all."

"With the Kollektion 3, you can easily use ASP, PHP or ColdFusion, even if you're not a programmer at all," Evans notes. "My clients think I'm a genius, of course it's InterAKT that are the geniuses, but we won't tell!" she jokes.

Evans welcomed the tutorials that ship with the MX Kollektion 3, calling them an "invaluable addition to any developer's toolbox."


To read more on MX Kollektion, MXDJ readers can go to: www.interaktonline.com/ 

figure 1

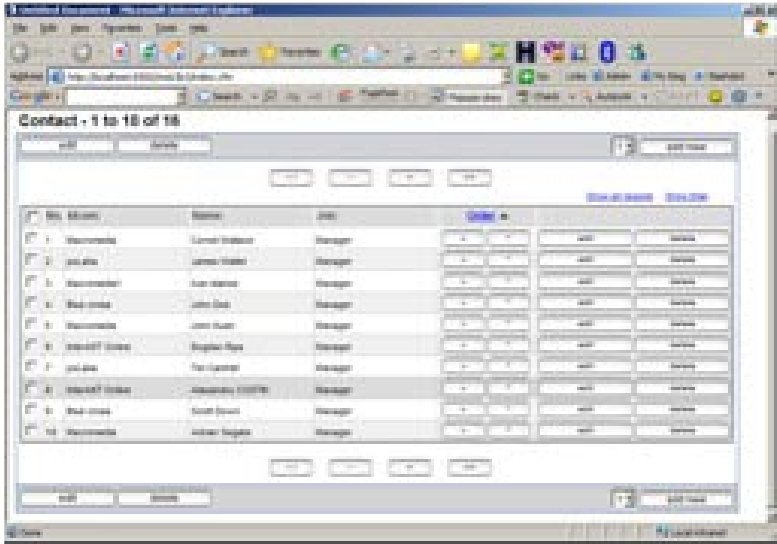


figure 2



figure 3



SYS-CON MEDIA
President & CEO
 Fuat Kircaali, 201 802-3001
 fuat@sys-con.com
Vice President, Business Development Grisha Davida, 201 802-3004
 grisha@sys-con.com
Group Publisher
 Jeremy Geelan, 201 802-3040
 jeremy@sys-con.com

ADVERTISING
Senior Vice President, Sales & Marketing
 Carmen Gonzalez, 201 802-3021
 carmen@sys-con.com
Vice President, Sales & Marketing
 Miles Silverman, 201 802-3029
 miles@sys-con.com
Advertising Sales Director
 Robyn Forma, 201 802-3022
 robyn@sys-con.com
Advertising Sales & Marketing Manager
 Dennis Leavey, 201 802-3023
 dennis@sys-con.com
Advertising Sales Manager
 Megan Mussa, 201 802-3023
 megan@sys-con.com
Associate Sales Managers
 Dorothy Gil, 201 802-3024
 dorothy@sys-con.com
 Kim Hughes, 201 802-3025
 kim@sys-con.com

PRODUCTION
Production Consultant
 Jim Morgan, 201 802-3033
 jim@sys-con.com
Lead Designer
 Louis F. Cuffari, 201 802-3035
 louis@sys-con.com
Art Director
 Alex Botero, 201 802-3031
 alex@sys-con.com
Associate Art Director
 Tami Beatty, 201 802-3038
 tami@sys-con.com
Assistant Art Directors
 Andrea Boden, 201 802-3034
 andrea@sys-con.com
Video Production
 Frank Moricco, 201 802-3036
 frank@sys-con.com

SYS-CON.COM
Consultant, Information Systems
 Robert Diamond, 201 802-3051
 robert@sys-con.com
Web Designers
 Stephen Kilmurray, 201 802-3053
 stephen@sys-con.com
 Percy Yip, 201 802-3054
 percy@sys-con.com

ACCOUNTING
Financial Analyst
 Joan LaRose, 201 802-3081
 joan@sys-con.com
Accounts Payable
 Betty White, 201 802-3002
 betty@sys-con.com
Accounts Receivable
 Gail Naples, 201 802-3062
 gailn@sys-con.com

EVENTS
President, SYS-CON Events
 Grisha Davida, 201 802-3004
 grisha@sys-con.com
National Sales Manager
 Jim Hanchrow, 201 802-3066
 jimh@sys-con.com

CUSTOMER RELATIONS
Circulation Service Coordinators
 Edna Earle Russell, 201 802-3081
 edna@sys-con.com
 Linda Lipton, 201 802-3012
 linda@sys-con.com
JDJ Store Manager
 Brundila Staropoli, 201 802-3000
 bruni@sys-con.com



d



Creating Standards-Compliant Web Pages with Dreamweaver MX 2004

Future-proofing your code
by kevin ruse

did you ever stop and wonder what the Web might look like if there were no standards? I think it would be a web of confusion.

It reminds me of a file cabinet that I have full of important electronic data that I've accumulated over the years. I have accounting information on floppy disks from years ago. The data was written in what is now long-obsolete software in file formats I can't access. If the World Wide Web were like my file cabinet it would be full of information written in Word 97 and WordPerfect and, depending on the year I generated the files, some of it would be in the Apple Macintosh format while others would be in Microsoft PC format. I have images created in formats ranging from TIFF to PCX. The files I've generated over 10 years ago are almost all in proprietary formats.

What a limited and expensive World Wide Web it would be if there were no standards.

Fortunately for us the World Wide Web's creator Tim Berners-Lee not only created a protocol for retrieving documents, the HyperText Transfer Protocol (HTTP) he also created a standard language for creating documents to be retrieved called the HyperText Markup Language (HTML).

Those of us who've traveled the Internet since it started will recall how quickly it became proprietary in its structure and abilities. Prior to the so-called browser wars, the World Wide Web was fragmenting into competing languages, overlapping functionality, and proprietary code that made any attempt at universal Web design a frustrating experience. Many of us have felt the frustration of trying to write Cascading Style Sheets that displayed our designs in an identical, if not similar, fashion across multiple Web browsers and platforms. Besides presentation problems, we also experienced functionality issues with JavaScript (from

Netscape) versus JScript (from Microsoft). The browser wars between Netscape Navigator and Internet Explorer really brought the issue to the forefront as the two competed to provide a richer, more dynamic experience for providers and end-users alike.

The subsequent creation of the World Wide Web Consortium and its recommendations (read standards) has done much to alleviate the problems caused by the browser wars. The browser makers themselves have also contributed to the solution by upgrading their browsers to render HTML code more consistently. So we've seen a return to the original plan – a Web document based on a standards-based language.

While Web designers today understand the need to write code that works in all browsers and on all platforms, many simply don't appreciate the consequences of non-standard code. Today's Web browsers ask much from the humble HTML document. The Web page has become a starting-off point for information. Eventually the Web page content is used to formulate printed pieces, populate forms including PDF pages, and some Web pages are components of an advanced Web application or software that runs via the Internet. Not all Web pages can fulfill these purposes. To use a Web page to create a PDF, or populate a back-end database, or feed a Flash application, the content must be consistent; it must be standard. The standard that promises the most possibilities is XHTML. Other standards that help multi-purpose your content include Cascading Style Sheets, XSL-T, and XML.

The purpose of this article is to demonstrate how Dreamweaver MX 2004 can help you write standards-compliant code. You'll learn how to ensure your documents are written in XHTML, or if you prefer HTML 4.0, etc. You'll learn how to build accessibility into your documents that will help PDAs and cell phones ren-

der your Web pages as well as desktop browsers and printers. By writing XHTML-compliant code and using an external stylesheet you'll be able to exploit all of the multi-purposing possibilities available.

What Type of Document Should You Write?

If you want to repurpose the data in your Web page, the most flexible format is XHTML. When your document is written in XHTML that means it's an XML application – in other words, pure data that can be rendered for any purpose. For example, using the XML stylesheet language called XSLT, you can transform XHTML Web pages into any other format including any of the new Microsoft Office Markup languages like WordprocessingML for Word and SpreadsheetML for Excel. You can also transform the XHTML into an XML file suitable for importing into many applications including Flash to create dynamic Flash applications and Adobe InDesign to create printed pieces. The possibilities are becoming endless as more applications become XML-compatible.

On the other hand, if you're reworking older Web pages that need to be viewed in early (pre-standards-based browsers) you may want to ensure that they are written according to the rules of HTML in use at the time of the browser release (HTML 4.0, for example).

The solution in both cases is to declare a document type at the beginning of your Web page that announces to the Web browser what version of HTML you're writing. In the "New Document" dialog box, Dreamweaver offers an opportunity to ensure that you create XHTML. See Figure 1, which shows the "Make document XHTML-compliant" checkbox. If you don't check this box, then Dreamweaver will write the Web page in HTML 4.01 Transitional, which means that the document may use

deprecated presentational elements such as the font tag. This makes the document backwards-compatible for older Web browsers. When creating a document that's not XHTML-compliant, Dreamweaver generates the following code:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD
HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.
dtd">
<html>
<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type"
content="text/html; charset=iso-8859-
1">
</head>

<body>
</body>
</html>
```

Enabling the "Make document XHTML-compliant" checkbox would result in the DOCTYPE shown in the following code:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD
XHTML 1.0 Transitional//EN" "http://
www.w3.org/TR/xhtml1/DTD/xhtml1-tran-
sitional.dtd">
<html xmlns="http://www.w3.org/1999/
xhtml">
```

Once you've written your Web page as XHTML, you can validate it in Dreamweaver to make sure you've not broken any of the rules of the XHTML language. If you want to exploit the XML capabilities of XHTML, your code has to be valid. To validate your code using Dreamweaver:

1. Open the Web page you've written in XHTML
2. From the Window Menu, choose "Results" or hit F7 to bring up the "Results Panel"
3. Click the Validation tab at the top of the Results Panel
4. Click the green arrow on the left of the Results Panel and choose "Validate Current Document." Note that you can also validate an entire site, or selected

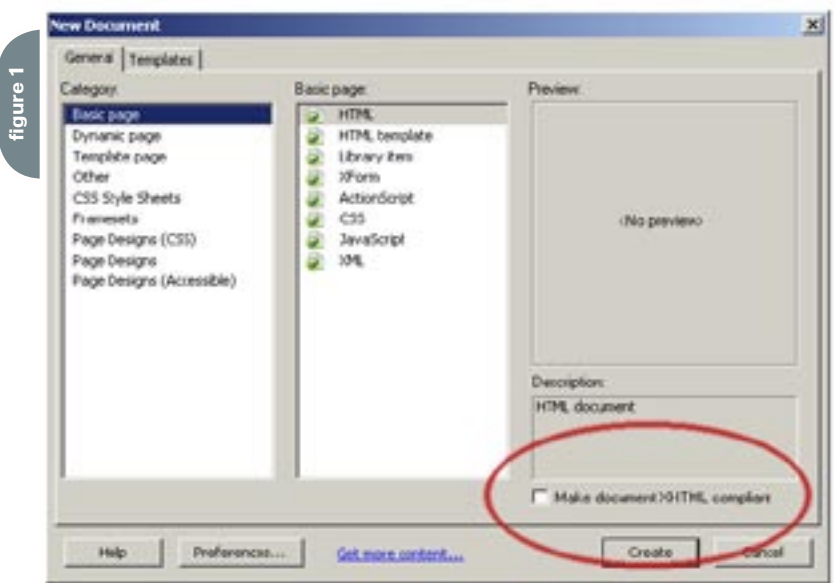


figure 1

files from the site. If the site contains no errors, the results panel will display as shown in Figure 2.

The Dreamweaver validator looks at the DOCTYPE declared at the beginning of the document and checks it for errors against that DOCTYPE. Because we chose to write XHTML Transitional, the results panel wouldn't indicate any errors even if we used deprecated tags such as and . However, if we change the DOCTYPE to XHTML strict, the validator would find our errors as shown in Figure 3. Dreamweaver doesn't provide a method for changing the DOCTYPE declaration other than the "Make document XHTML-compliant" checkbox. So, to write XHTML strict, you must manually change the DOCTYPE in the code view. See the

Macromedia exchange for an extension that allows DOCTYPE changes. For information on how to change the default DOCTYPE in Dreamweaver, see the MX Developers Journal online at <http://mxdj.sys-con.com/read/45940.htm>. The DOCTYPE should read as shown below.

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

The results panel also lets you save the validation results as an XML file and see the results in a Web browser as an HTML file. Double-clicking the error message will highlight the code in Dreamweaver.

Outside of Dreamweaver, you can vali-

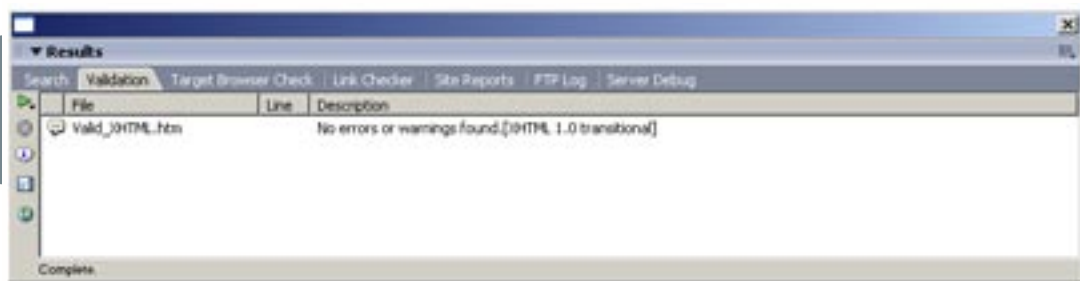


figure 2

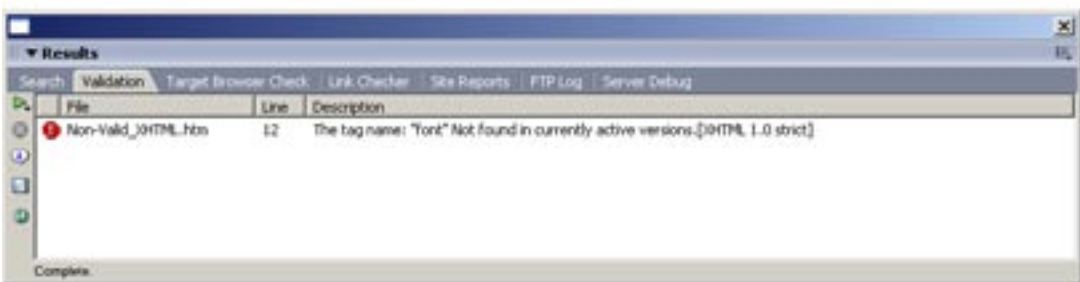


figure 3

Kevin Ruse is the author of *Web Standards Design Guide* published by Charles River Media, Inc. The book explores the current standards, XML, XHTML, CSS, Xforms, and XLink as well as workflow issues involving these standards. kevin@kevinruse.com

date your code using the W3C's Validator Service at <http://validator.w3.org/>. You can use the Validate by Upload section of the page and browse to the file you want to validate. Figure 4 shows the results of validating the same page that Dreamweaver validated earlier (with the DOCTYPE XHTML Transitional) indicating no errors. The W3C Validation Service is a bit fussier and returns an error message because our page uses an image tag without an alt attribute.

The Doctype Switch

Declaring DOCTYPES becomes especially helpful when you want to ensure that your CSS code is rendered as you intended. Today's standards-compliant Web browsers are really two browsers in one. Technically speaking, they're capable of rendering in two modes: "standards-compliant mode" and "backwards-compatible mode" also known as "quirks mode."

The Web browser reads the DOCTYPE line that triggers the correct rendering mode. For example, if your DOCTYPE declares that you are writing XHTML 1 Strict that means that your code uses only XHTML elements relating to the document structure (no deprecated markup is allowed); the presentation information is left entirely to a Cascading Style Sheet. How that Cascading Style Sheet

is rendered depends on the Browsers Rendering Mode (either standards-compliant or backwards-compatible).

Recall the browser wars, and subsequent years, when Web browsers rendered CSS code differently one browser to the other. One of the most prevalent problems has been the way the different browsers render DIV tags and their associated height, width, border, margins, and padding, the so-called box model.

Essentially, the problem rests with how Internet Explorer understands this box. If you'd like a div to be 100 pixels wide with a padding of 10 and a border of 10, you simply write a CSS width property equal to 100 pixels. With IE, however, your div box would be 40 pixels smaller than you'd expect, because it subtracts from the width the 10 pixel padding on the right and left as well as the 10 pixel border on the right and left. This is just one of several problems with the way some older (non-standards-compliant) browsers render CSS.

There is a possible solution. Imagine having to rework an old Web site that's several hundred pages long because the new browsers render the CSS in a standards-compliant way, but you wrote the page to work with browsers in use when you created it (non-standard, workarounds, CSS hacks, etc.). Rather than upgrade several hundred pages, you can

"downgrade" the browser. In other words, force the browser to render your page in "backwards-compatible mode" or "quirks mode." To do this you simply declare the correct DOCTYPE.

Unfortunately, declaring the "correct" DOCTYPE can be tricky, because browsers can respond to the same DOCTYPES differently. For example, the DOCTYPE in the following code will trigger "standards-compliant" mode in Mozilla 1.0, Netscape 6.0+, Safari 1.0+,

Internet Explorer 6.0+, Opera 7.1+, but will trigger "quirks" mode in IE 5 for the Mac.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 //EN">
```

Listing 4 A DOCTYPE for HTML 4.01 without a URL

You can use a simple JavaScript for testing purposes to see if the DOCTYPE you're using will trigger "standards-compliant mode" or "quirks mode." Copy the following code to the head section of the Web page you want to test in Dreamweaver and preview in the browser.

```
<script language="javascript">
function displayMode() {
  theMode = document.compatMode;
  alert(theMode);
}
</script>
```

You'll have to add a button to trigger the JavaScript. Copy the code shown below to the bottom of the Web page you want to test.

```
<form>
  <input type="button" name="Button"
  value="Display Mode" onClick="display
  Mode()" >
</form>
```

If your code triggers "standards-compliant mode" in IE, you'll see the alert box shown in Figure 5; otherwise IE will display "Backcompat." Opera 7.5+, Mozilla 1.0, Firefox 1.0, and Netscape 7.2+ will also display "CSS1Compat." If Opera is rendering in Quirks Mode the alert box will display "Quirksmode." If Firefox 1.0, Netscape 7.2+, or Mozilla 1.0 is rendering in quirks mode the dialog box will display "Backcompat."

Dreamweaver's ability to write standards-compliant code lets you create Web pages that can then be used for other purposes. For example XHTML can be transformed into XML, which can then be used to drive a Flash Web application. In a similar way XHTML code can be reused through ColdFusion where the XML file replaces a database. Perhaps the greatest benefit in writing and validating our standards-compliant code is knowing that our code is ready for future browser releases, meaning less rework for us. As Web designers, we should commit ourselves to the future of the Internet by supporting standards. Macromedia as a software developer has committed itself to creating a product that writes code in accordance with the accepted standards of today. ☺

figure 4



figure 5



- Home Help
- App Server
- ColdFusion Admin
- JRun MC
- Restart
- System Logs
- Heartbeat
- Configure
- Site Management
- Domains
- Emails
- Horde Web Mail
- File Manager
- Site Builder
- Web Counter
- AWSstats
- Web Logs
- Database
- Apache
- Applications
- Advanced

testmycfm DNS, Domain Host, and Email Options:

Domain Host and DNS Configuration | Email Configuration

:: Domain Host and DNS Configuration ::
Please refer to this topic in the [Help](#) section before proceeding.

DOMAIN
Select Domain: (1)
Dedicated Address: [testmycfm.webappcabaret.net] Server Address: [pointer.webappcabaret.net]

Domain Name (mydomain.com)
IP/Host Address: testmycfm.webappcabaret.net

Apache Document Root
Apache Directory Index: index.html index.jsp index.do index.html.ver index.
AppServer Virtual Path(s)

DNS & SUB-DOMAINS
DNS Service

MX Address (separate multiple MX with commas):

Sub Domain	Apache Document Root	Directory Index	Virtual Path(s)	IP/Host Address	CNAME
<input type="text"/>	/usr/ngasi/contexts/testmy	index.html index.jsp index.	<input type="text"/>	testmycfm.webapp	<input type="checkbox"/>
<input type="text"/>	/usr/ngasi/contexts/testmy	index.html index.jsp index.	<input type="text"/>	testmycfm.webapp	<input type="checkbox"/>

[Email Configuration](#) ▲

Imagine a hosting company dedicated to meet the requirements for complex web sites and applications such as those developed with ColdFusion MX. At WebAppCabaret our standards based process and tools make deploying ColdFusion MX applications as easy as a point-and-click. We call it **Point-and-Deploy Hosting**. Our advanced NGASI Web Hosting management Control was designed for the hosting and management of ColdFusion web sites and applications thus cutting down on maintenance time.

Backed by an experienced staff as well as a **Tier 1 Data center** and network. Our network is certified with frequent security audits by reputable security firms.

All ColdFusion hosting plans have separate and individual installation AND instances of ColdFusion MX 6.1 Enterprise with **full access to ColdFusion Administrator** and JRun Management Console; so there is virtually no restriction or customization required for your application.

Complete power and control at the tip of your fingers. We take care of the system and hosting infrastructure so you can concentrate on development and deployment of your application. That is **real ROI**.

Log on now at <http://www.webappcabaret.com/cdj.jsp> or call today at 1.866.256.7973



d

f

fu

cf

df

fx

NET in a FLASH

A NEW TOOL LETS .NET DEVELOPERS WRITE FLASH APPS

by jim phelan

Not long ago developers and IT professionals looked at Flash with disdain. The anti-Flash camp had contempt for its lack of accessibility, minimal standards compliance, and of course there was the infamous back button debacle. But things have changed. Large companies are embracing Flash for rich media delivery. Developers are jumping on the bandwagon in swarms. In the last year there's been an explosion in Flash development tools, including several full-featured development environments and a homespun SWF compiler or two. Flash video suddenly seems to be everywhere, and if you're not talking about AJAX you're talking about rich Internet applications with Flash. So, now that we're confident that Flash is here for the long haul, the question becomes: Is Actionscript coming along for the ride?

Recently, a flurry of developer tools has made Flash development a more attractive proposition. Now, with a plug-in, developers can use Eclipse (the Java favorite) as their IDE, and even compile to SWF with a



Macromedia's MXML was one of the first commercially available application markup languages, and it seems that Xamlon Web will actively compete against Flex in some respects

compiler called MTASC. Still, some developers turn to another solution, such as Flex, for full-featured layout components and true rapid application development.

Interestingly, one of the most exciting tools to surface lately doesn't have anything to do with Actionscript at all. Xamlon Web, created by Xamlon, Inc., lets .NET developers create Flash applications purely in C# or VB.NET. Entire program can be written in Visual Studio .NET and then deployed to the SWF format. Although there are undoubtedly some nuances in developing for Flash Player using .NET, the process is strikingly similar to developing for a typical Windows forms application.

Xamlon, Inc., is the brainchild of Paul Colton, the original author of the JRun server software purchased by Allaire and, subsequently, Macromedia. Originally, Paul's concept was to make Microsoft's Extensible Application Markup Language (XAML) available to .NET developers for immediate use in developing Windows

desktop applications. XAML, which will be released with .NET 2.0 and Windows Longhorn, is an application markup language comparable to Mozilla's XUL or Macromedia's MXML. Although Colton's original product, Xamlon Desktop, was well received, he found that the product's real power was in deploying to the Web using SWF.

As Colton described his vision for Xamlon Web, "The idea is to let developers use Visual Studio to write in .NET and deploy directly to the Web without knowing anything about the target, which in this case is Flash."

Colton feels that making the Flash format a viable option for .NET developers is a big step. "We've taken the very best of Microsoft and deployed it to the very best coming out of Macromedia, and we're delivering one product that neither company really can because of competitive forces," he said, suggesting that the combination of Flash delivery with the Microsoft development workflow and IDE

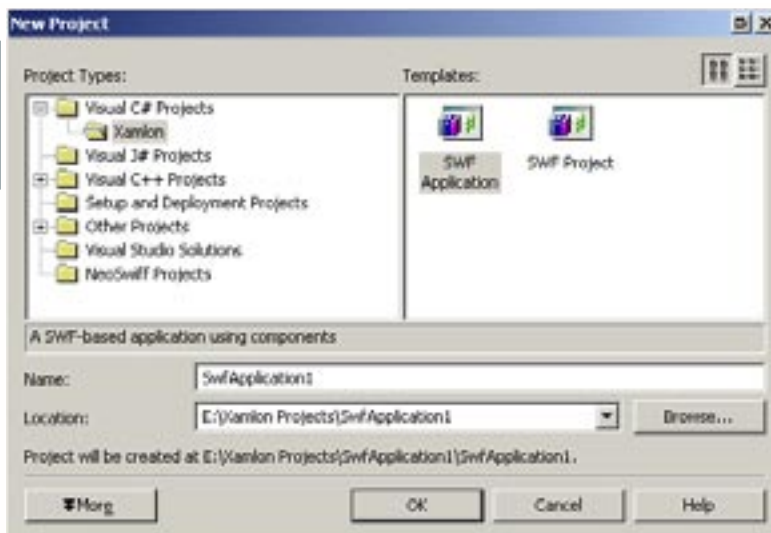
creates a best-of-breed product that their parents can't deliver.

The jury is out on how effective Xamlon is and will be until the 1.0 release arrives in mid-July, but even the current beta version has been used to do some interesting things. One example on the Xamlon Web site uses SOAP Web Services to create a Flash-based airline seat selector entirely in XAML and C#. Another connects to an XML source and displays local weather information. And, while the XAML-based layout is one of the more interesting features, it's actually not needed to develop applications. In fact, a developer could write a Xamlon application using only standard C# with no XAML at all.

The Xamlon process leverages existing Microsoft technology and tools to achieve a streamlined process that's familiar to .NET developers. Xamlon Web offers complete Visual Studio integration, letting developers create their apps in a familiar way and then deploy to the SWF format. Perhaps the most interesting component is that, unlike competing tools, Xamlon doesn't compile the C# or VB.NET code; Microsoft's compiler compiles the code into Microsoft's Intermediate Language (IL). Xamlon then takes over, creating a SWF file from the IL bytecode. This means that Xamlon will evolve more or less as the Microsoft .NET platform does, and should support any .NET language. Xamlon Web will also allow for some extensibility, including a component framework for extending Xamlon's capabilities.

Xamlon Web claims to be fully .NET-compliant and support most common .NET classes. It only includes a subset of .NET functionality that Flash can support,

figure 1



which is actually quite powerful. The product is still in beta, so we won't have full functionality until the July release, but Colton is confident that version 1.0 will be production-ready.

For the experienced Flash developer, Xamlon offers the SwfNative namespace that lets developers access most of the functionality available in an Actionscript application. The only glaring omission is the remote NetConnection, so connecting to a Flash Communication Server isn't possible. Xamlon plans to investigate this more thoroughly, so hopefully an early version will rectify this problem. Flash Remoting isn't supported either (although SOAP Web Services are).

Macromedia's MXML was one of the first commercially available application markup languages, and it seems that Xamlon Web will actively compete against Flex in some respects. Although Xamlon's Web site flatly says that it won't, Colton spoke of a server-based version of Xamlon Web following 1.0's release. When asked what advantages Xamlon Web had over Macromedia Flex, he replied, "XAML is much more thorough and involved and has a much broader set of capabilities than MXML." While this may be true, it's also true that only a subset of XAML and .NET can really be applied in the context of a Flash application. If the two products ultimately do compete, there will be irony in the fact that Flex runs atop JRun, the server software originally written by Colton.

And, since Colton did create JRun, why does Xamlon support .NET and not Java? Xamlon's rationale as a company was Microsoft's XAML, so it makes sense that the software supports .NET, but it poses problems for developers who write in Java or on the Apple platform – in fact, they're pretty much excluded from Xamlon. Colton says he intends to correct this in the future by creating a Java version of the software and it will be interesting to see if the Java version supports XAML markup as fully as the .NET version. Colton didn't have a projected release date for the Java version, but stated that it's "in the works."

He repeatedly said that Xamlon's main benefit was multi-channel delivery: in theory, a Xamlon application can be delivered via the Web, a desktop application, and a mobile device with little extra

effort. This, of course, is something developers have been attempting to attain for years, so it'll be interesting to see if Xamlon's unique blend of Microsoft and Macromedia technologies achieves it. It has yet to deliver the proof-of-concept.

Does Xamlon deliver on its promises? The Xamlon Web beta comes with more than 20 examples that try to show that it does, and the Xamlon Web site has some more intriguing real-world examples of applications developed in Xamlon. Since no programming article would be complete without the perennial "Hello World" example, let's dive into how Xamlon can be used to create the simplest of applications. This is the XAML markup code:

```
<?xml version="1.0"?>
<Canvas Width="600" Height="400"
Background="LightGreen" />
```

This is the C# code:

```
using System;
using SwfNative;
```

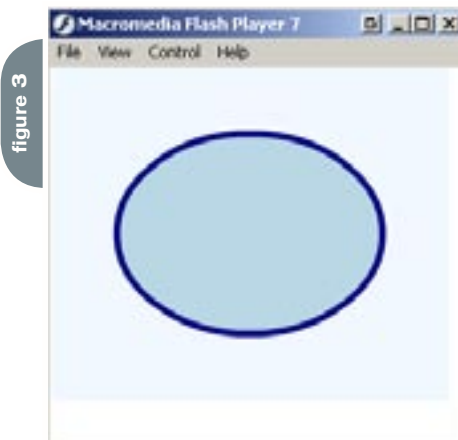
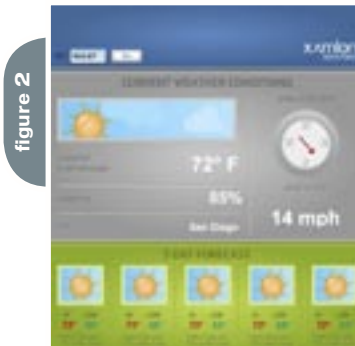
```
namespace Demo
{
    public class HelloWorld
    {
        public static void Main()
        {
            TextField cn = MovieClip.Root.
CreateTextField(55,55,100,33);
            cn.Variable = "Console";

            Console.WriteLine("Hello
World!");
        }
    }
}
```

The Hello World example seems a bit cumbersome, but Xamlon comes through on user interface controls. This is the C# code to create a button and attach an event handler to it.

```
Xamlon.Swf.Controls.Button button =
new Xamlon.Swf.Controls.Button();
button.Click += new
ButtonDelegate(button_Click);
```

This sample doesn't completely deliver on the multi-channel delivery promise, since the namespace (Xamlon.Swf.Controls) isn't the same as the .NET



controls namespace, but Colton assured me that the final release will completely support the standard Windows.Forms. Controls namespace as well as Microsoft's visual designer. Developers will be able to redraw controls using GDI+ the same way they do now. Xamlon Web will also support skinning controls using XAML markup.

And, surprise, Xamlon Web also supports pure XAML for vector-based drawing and layout. The results are quite astonishing, and open up some interesting possibilities for streamlining the vector graphic creation and management process. The company supplies a free evaluation of a XAMLConverter tool that converts Adobe Illustrator-generated SVG into XAML. The following shows a simple ellipse drawn in XAML.

```
<?xml version="1.0"?>
<Window ID="root" xmlns="http://
schemas.microsoft.com/2003/xaml"
xmlns:def="Definition" Width="230"
Height="255">
    <TransformDecorator ID="zoom"
AffectsLayout="false">
        <Canvas Width="300"
Height="250" Background="AliceBlue">
            <Ellipse
                CenterX="150"
```




figure 4



```
CenterY="125"
RadiusX="100"
RadiusY="75"
Fill="LightBlue"
Stroke="DarkBlue"
StrokeThickness="5"/>
</Canvas>
</TransformDecorator>
</Window>
```

The actual XAML support, however, leaves something to be desired, because Xamlon Web only offers a subset of XAML capabilities. It doesn't provide examples that illustrate binding XAML controls to a codebehind file, one of the most important features of XAML, nor does the beta support the visual designer component of Visual Studio. This functionality is expected in version 1.0, and how well the company pulls it off will decide whether Xamlon's version of XAML is a lifesaver or just a neat feature.

Xamlon's most intriguing examples are found on its Web site. One is a Google Maps client that shows a front-end for Google Maps developed exclusively with Xamlon Web. It lacks the complexity and completeness of Google Maps, but it's an interesting achievement nonetheless; Xamlon claims it was created over a weekend without the backbreaking job of writing an AJAX app. The company has coined the term AFLAX (Asynchronous

Flash and XML) presumably to jump on the AJAX bandwagon and make a case for Flash. However, one wonders whether the world needs another acronym. In this respect, it seems that Xamlon missed the point about one major advantage of Flash, which is that wonderful things can happen when Flash Communication Server or socket-based connections are used to create stateful applications (something an AJAX application can only mimic).

Another example that deserves attention is the StockWatcher (see figure 4). This RIA is a living breathing stock analyzer and keeps a live feed of stock data available to update datagrids and draw graphs. Possibly the most compelling application created with Xamlon to date, it instills a sense of confidence that Xamlon does in fact work and is moving in the right direction.

It seems that Xamlon might offer the .NET developer a lot, but what does it offer us, the few, the proud, the Actionscript developers? I posed this question to Colton, who quickly replied, "The positive for Actionscript Developers is this: C# is a mature, strongly typed language. It has an awesome development environment, and that's lacking in Macromedia; there isn't a very powerful development environment for Actionscript. Actionscript is great, but

how would you like to use a better language with a better IDE, and still target the Flash that you know and love? We're going to bring real languages and real development environments to the Flash community."

While that statement may not exactly warm your heart, it's good to know that Xamlon is keeping us in mind. Xamlon Web has separate tools that let experienced Flash developers access native commands, and ships with components that are more or less specific to Flash development like the Accordian.

Colton also told me that he had met with numerous Flash developers during Xamlon Web development to get their input.

For the most part, Xamlon does its job, but it's definitely a beta product. If Xamlon delivers on its version 1.0 promises it will be hard to knock. .NET developers aren't currently developing for SWF and the adoption of Xamlon might take a while, so don't trade in your copy of Essential Actionscript 2.0 for Learning C# just yet. In fact, Actionscript isn't going anywhere. Macromedia has made a strong commitment to the language and products like Flex use it extensively. The question of whether Xamlon (or competing products) will be effective enough to allow real SWF delivery for .NET developers has yet to be answered. As a developer who uses both Actionscript and .NET, Xamlon and XAML are very exciting propositions, but I'll still be developing in Actionscript for the foreseeable future. ☺

Jim Phelan is VP of Development at Stream57, a New York City firm specializing in creating Rich Internet Applications for live video events, videoconferencing, and e-learning. Jim, who had been programming since age eight, is an avid enthusiast of Flash and now develops primarily in ActionScript.

jim@flashalchemy.com

The jury is out on how effective Xamlon is and will be until the 1.0 release arrives in mid-July, but even the current beta version has been used to do some interesting things



FLASHFORWARD & FLASH® FILM FESTIVAL

JULY 6-8 2005 NEW YORK CITY
NEW YORKER HOTEL

THE WORLD'S PREMIER FLASH EVENT

The 14th Flashforward conference and Flash® Film Festival, the largest and longest running Flash user conference in the world, shares the latest in design, development, education and inspiration.

FLASHFORWARD 2005 FEATURES:

- 6 In-Depth Flash Workshops
- 25 One-Hour Seminars
- 20 Q&A Sessions
- 17 "Ask the Experts" Sessions
- 11 Technology Showcase Seminars
- Flash® Film Festival
- Macromedia Keynote
- Exhibition Area
- Meet the Speakers and Flash Authors Reception
- Networking Receptions
- Exclusive Conference Workbook with Speaker Notes

GOLD SPONSOR



SILVER SPONSOR



BRONZE SPONSORS



MEDIA SPONSORS



REGISTRATION:

- \$599 through June 3
- \$699 through July 1
- Additional \$50 Macromedia User Group discount
- \$799 at the door

ADDITIONAL INFORMATION:

- Call toll free 1.877.4.FLASH.4
- www.flashforwardconference.com
- 1-303-593-1977 outside the US and Canada



Using Electric Rain's Swift 3D 4.5

It rocks!
by matthew david

It may only be a point release but that doesn't mean that Electric Rain, makers of the #1 3D modeling software tool for Flash animators hasn't come out swinging. What you'll find with this new release is a tight focus on the rendering options in Swift 3D.

Quick Run Down on Swift 3D

First I have to admit I'm partial to Electric Rain's Swift 3D. To me, it has always been the small product that stood shoulder to shoulder with the likes of Soft Image, Max and the other big 3D boys. Swift 3D isn't the only Flash 4D design product on the market. It's simple the one outselling its competition. According to erain.com over 40,000 developers have bought Swift 3D.

So what makes it special? The answer lies with its focus; it's designed for Flash animation by Flash animators. You see the Flash animation group isn't like other groups. We have special wants and needs, a real desire to deliver solutions over the Internet and deliver high-impact solutions. When companies launch discreet release products such as plasma that die on the vine they miss that point. Flash developers don't want a huge animation suite that costs thousands of dollars. They want a tight tool focused on 3D animation that can be effectively delivered on the Web.

How Easy Is It To Use Swift 3D? This easy

The easy-to-use Swift 3D layout lets you jump in and begin modeling. Imagine that you've never designed or modeled in 3D in your life. Heck, you wouldn't know a 3D model if it hit you in the face. But, the boss said you have to have 3D on your Web and you open up Swift 3D. What do you do?

First, ask yourself the question, do you have an EPS version of the company logo? A spinning logo would be cool. Enter Swift 3D. You can create a new file in Swift 3D, select File – Import and directly import an EPS file.

The EPS file opens into Swift 3D. The colors and design have been retained. You'll also see that the file has been extruded to give it some depth.

If you want to choose the different elements that make up the logo, you select the Hierarchy Tools. Here you can expand the groups to gain deeper control over the objects.

OK, you now have a 3D logo. So what? This stuff is supposed to spin and dance. No problem there. You can leverage a powerful tool called the Gallery Tool, which is made up of six different tool sets:

- Materials
- Animations
- Lighting
- Environments
- Models
- Bevels

Each of these solutions helps build a more stunning model. For instance, we can replace the Materials used to build the Electric Rain model with reflective surfaces.

Next, add one of the many instant animation sequences.

Add some funky lighting to show off your model's shadows and effects.

You can keep adding all sorts of effects. The next step is to go to the

preview and export screen. Here you can choose to export as either a vector animation or a bitmap animation. Here is the logo as a vector animation:

It's as the export stage, however, that the new upgrade really shines.

Exporting with Swift 3D

Most of the new features in Swift 3D 4.5 are hosted in the rendering part of the product. The first feature that will delight you is the new choice of export options. Check out this list:

- Windows AVI
- QuickTime MOV
- Macromedia Flash Video (FLV)
- Flash Player (SWF)
- Swift 3D Flash Importer (SWFT)
- Encapsulated Postscript (EPS)
- Adobe Illustrator (AI)
- Scalable Vector Graphics (SVG)

Yeah, you saw it right; now there's support for SVG. To get to these lovely choices you'll need to select the Preview And Export Editor tab and click on the Vector button. This brings up the updated Ravix 4 encoding engine. Cool stuff.

The EPS export has been enhanced to support level 3, which includes area shading.

You'll also find greater control over shadows in your export under the Fill Options. Here you can choose shadow color, density, reflections, and specular lighting.

For me, the best feature to finally make it into Swift 3D is support for Pen Outlines. Now you can really make cartoon animations look like classic cartoon animations.

What's Missing?

As with any product, I always come away wishing there was more. Two features that could have a huge impact on the application and audience using it are missing from Swift 3D. The first is support for ActionScript. It would be great



to be able to build applications and interfaces directly in Swift 3D without having to crack open Flash. The other feature that I long to see is the support for plug-ins or a way to extend the feature set in Swift 3D beyond what you get from eRain. Think of PhotoShop plug-ins for Swift 3D.

but can be upgraded for as little as \$138.

If you're seriously into Flash animation you seriously have to consider adding Swift 3D to your toolset. I don't rave about many products, but the price and value Swift 3D brings to your design make it easy to rave about. ∞

The Bottom Line

Currently Swift 3D sells for \$350

mdavid@matthewdavid.ws

figure 2



figure 3



figure 4

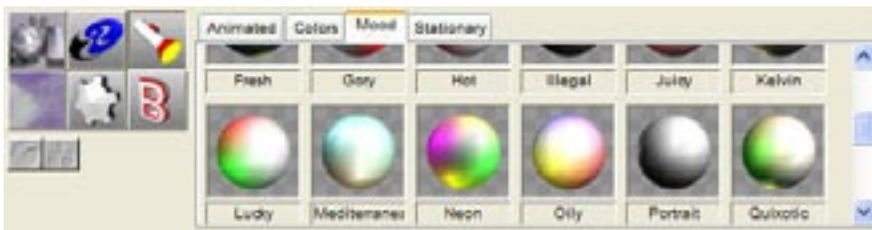


figure 5



d

f

fu

f

d

f



Using the

Singleton Design Pattern

in Flash

by danny patterson

*A standardized solution to
a common programming problem*

Design patterns are standardized solutions to common programming problems. It's good to use design patterns for two simple reasons. First, it's faster (and many times better) to implement a time-tested design pattern as opposed to developing a custom solution. Second, collaboration with other developers increases when common practices are used.

The Singleton Pattern

The Singleton design pattern is one of the most widely known design patterns; and it's also one of the simplest. It's used to insure that a class is only instantiated once. The pattern also provides a global point of access to the sole instance of that class. Many classes have to be able to limit their instantiation to a single instance. For example, you may write a focus manager for you application. If you had two instances of the focus manager, they might conflict with each other and create race conditions. So it's essential that you have only one instance per application.

There are two goals for the Singleton pattern: one instance and global access. The global access point insures that all of your classes have access to that single instance. But the main task of the Singleton pattern is to ensure that there is only one instance of the class. That's why the Singleton pattern lets the class manage its own instantiation. Instead of creating an instance of the class by using a new keyword, outside access will be given via a static method that returns the lone instance of the class. If the instance doesn't exist then it creates the instance before returning the reference.

There are two things you must do to make the Singleton pattern work. The first is to use a static method to retrieve the instance of the class and a static property to store

the instance. The static keyword indicates that the following method or property is created only once per class rather than being created in every instance of the class.

The second thing you must do is make the constructor private. A private constructor is rare in ActionScript. Since the constructor is private, instantiation can only happen in the class. By making the constructor private, you can insure that no one can instantiate your class outside of the static method.

The Singleton pattern may seem complex, but the code is actually simple. The following example contains all the code you would need to make your class follow the Singleton pattern:

```
class Singleton {
    static private instance:Singleton;
    private function Singleton() {}
    static public function getInstance():
    Singleton {
        if(instance == undefined) {
            instance = new Singleton();
        }
        return instance;
    }
}
```

The instantiation of a Singleton class is a little different from a normal class. Normally we'd call the constructor directly like this:

```
var foo:Singleton = new Singleton();
```

However, in the Singleton pattern our constructor is private so that code would fail. We must get the single instance of this class by calling the static getInstance method like this:

```
var foo:Singleton = Singleton.getInstance();
```

Singleton in Action

In the following example, we'll build a class for tracking the navigation history of a given application. We'll use the Singleton pattern so all of an application's navigational classes track this history information in a central location. An example of some navigational classes might be a menu bar, a link, or even a button. All of these objects will have to be able to track the application's history. But ultimately there can be only one navigation history for an application so we use the Singleton pattern. This class is simple. It will simply store a string to describe a history destination. Any variety of information your application may need to determine the history destination could substitute. Our class will contain one private array that will hold

all of the items in our application's history. We will also have an additem method for adding destination items to the history. Then we'll have previousItem and nextItem methods for going forward and backward in our history.

```
// History.as
class History {
    private var historyItems:Array;
    private var currentPosition:Number;
    static private var instance:History;
    private function History() {
        historyItems = new Array();
    }
    static public function getInstance():History
    {
        if(instance == undefined) {
            instance = new History();
        }
        return instance;
    }
    public function addItem(item:String):Boolean
    {
        if(item != historyItems[currentPosition]) {
            if(currentPosition < (historyItems.length
            - 1)) {
                historyItems = historyItems.slice(0,
                (currentPosition + 1));
            }
            historyItems.push(item);
            currentPosition = historyItems.length - 1;
            return true;
        }
        return false;
    }
    public function getNextItem():String {
        if(currentPosition != undefined && current-
        Position <
        (historyItems.length - 1)) {
            return historyItems[++currentPosition];
        }else {
            return null;
        }
    }
    public function getPreviousItem():String {
        if(currentPosition != undefined && currentPo-
        sition > 0) {
            return historyItems[--currentPosition];
        }else {
            return null;
        }
    }
}
```

The implementation of this history class is also simple. The example above uses two navigation components. The following is the class code for those two components:

```
// SiteNavigation.as
class SiteNavigation extends MovieClip {
```

```
private var buttonOne:MovieClip;
private var buttonTwo:MovieClip;
private var buttonThree:MovieClip;
private var buttonFour:MovieClip;
private var history:History;
function SiteNavigation() {
    history = History.getInstance();
    buttonOne.onRelease = function() {
        if(this._parent.history.addItem('One')) {
            this._parent._parent.gotoAndPlay('One');
        }
    }
    buttonTwo.onRelease = function() {
        if(this._parent.history.addItem('Two')) {
            this._parent._parent.gotoAndPlay('Two');
        }
    }
    buttonThree.onRelease = function() {
        if(this._parent.history.addItem('Three')) {
            this._parent._parent.gotoAndPlay('Three');
        }
    }
    buttonFour.onRelease = function() {
        if(this._parent.history.addItem('Four')) {
            this._parent._parent.gotoAndPlay('Four');
        }
    }
}
// HistoryNavigation.as
class HistoryNavigation extends MovieClip {
    private var backButton:MovieClip;
    private var forwardButton:MovieClip;
    private var history:History;
    function HistoryNavigation() {
        history = History.getInstance();
        backButton.onRelease = function() {
            this._parent._parent.gotoAndPlay
            (this._parent.history.previousItem);
        }
        forwardButton.onRelease = function() {
            this._parent._parent.gotoAndPlay
            (this._parent.history.nextItem);
        }
    }
}
```

Conclusion

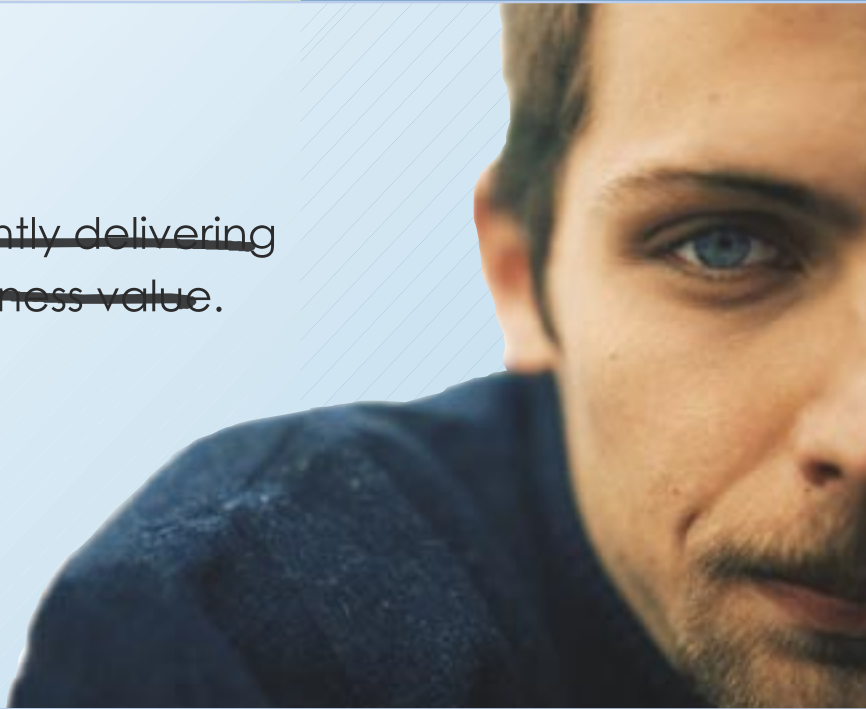
The Singleton pattern is both a simple and proven design pattern for restricting instantiation and for providing a global access point. A variation of this class is the Multiton pattern, which also restricts instantiation, but allows for a fixed number of instances. ☁

This article was originally published on Community MX (www.communitymx.com).

Danny Patterson is a Consultant specializing in Flash and Web technologies. Danny is a Partner and Author at Community MX (communitymx.com) and a member of Team Macromedia Flash. He is a Certified Advanced ColdFusion MX, Flash MX and Flash MX 2004 Developer. You can check out his weblog at DannyPatterson.com. info@DannyPatterson.com

Think Java development is complex & overengineered?

~~Think again.~~



~~I want to be known for consistently delivering applications and providing business value. That is why I use ThinkCAP.~~

ThinkCAP™

ThinkCAP 6.0 simplifies and accelerates the development and maintenance of J2EE-based web applications by 50%.

ThinkCAP's Visual Workbench and integrated MVC framework bring high productivity to corporate application developers (those with PowerBuilder, RPG, or Oracle-like skills) while letting J2EE programmers build reusable business logic & components using their tools of choice. The result is productivity for the whole development team.

Because of features such as Smart Data Binding, Visual Page Flow, and seamless Hibernate/Castor support, your team focuses on value added functionality — not low-level "plumbing." ThinkCAP integrates over 20 open source projects providing a practical mix of high productivity, robust functionality, and standards support.

Download a free trial version or view a demo at www.clearnova.com.

CLEARNOVA
Deliver Web Applications at the Speed of Business

Highly Productive Visual Workbench

Actions-based MVC Framework

Page Flow Designer & Generator

Easy to Learn Event Model

Advanced Data-Aware Components

Forms, DataViews, Queries, Navigations

Workflows, Graphs, Treeviews, Grids, Tabs

Multirow Updatable DataViews

Extensible Browser & Server-side Validation

Point & Click Smart Data Binding™

Service Flow Designer for Web Services/SOA

Content Management & Workflow Engine

Platform Independence: Any Java Server/OS/IDE/Database

Integrated Role-Based Security

Seamless Integration with over 20 Open Source Libraries

Team Development with CVS Support



SAILING

THE WINE DARK SEA

A FIREWORKS MASKING PROJECT

by jim babbage

SAILING, sailing over the bounding sea... Or in our case, inside a wine glass. We've all seen the ship-in-a-bottle knick-knack. Well, I'm here to show you how to set a sailboat afloat in a wine glass, using Fireworks MX or MX 2004, vector AND bit-map masks. Heave ho, matey, and don't forget to buckle your swash!

Combining images is part of the fun with programs like Fireworks. Making those images look like they belong together – to fool the eye, as they say – is challenging and rewarding. Take this unlikely pair of images, for example: a studio beauty shot of a wine bottle with a glass and a sailboat. Wouldn't it be neat to make the sailboat seem as if it were floating inside the wine glass?

If you answered “yes,” continue on. If you answered “no,” re-read the previous paragraph and answer, “yes,” this time.

Now that we’re all in agreement, let’s set sail – errr – get to work.

The final result of this collage makes use of a few layers and two masks, one vector mask and one bitmap mask: To neaten things up a little, I also work with the Rubber Stamp, Blur, Sharpen, Dodge, and Burn tool and Replace Color tools. This is a full project from start to finish, so we’ll be doing more than just adding the masks.

On that note, I would also mention that it’s a good idea if you at least have a passing understanding of the Layers Panel, general masking techniques, and the Pen tool. There are several good introductory tutorials on CMX that can help you get acquainted with these Fireworks Features:

- Understanding the Pen Tool
- The Elements of Vector Graphics
- Fireworks MX – The Basics – Part 2
- Using Masks in Fireworks
- Retouching Images in Fireworks MX
- Fireworks MX 2004 Piece by Piece - Bitmap Tools: Part 1
- Fireworks MX 2004 Piece by Piece - Bitmap Tools: Part 2

The winery is a real one here in the Niagara region, and it’s truly awesome. If you’re a red wine drinker, try their Black Cab – very nice. I created the wine shot as a quick test image for some new lighting equipment I was trying out. The winery is a real one here in the Niagara region, and it’s truly awesome. If you’re a red wine drinker, try their Black Cab - very nice. (See Figure 1) The sailboat was shot while on holiday last summer. (See Figure 2)

Preparing the Images

1. I started by opening the file named “wine,” then selecting the object itself by clicking on the thumbnail in the Layers Panel.
2. The image needs a couple adjustments to the Levels. I’d like a bit more snap from the wine shot, but not so much that we lose highlight detail. I find that almost any digitized image will benefit from a bit of tweaking with Curves or Levels.
3. Go to Effects in the Properties

- Inspector (P.I.) and click on the “+” sign.
4. Choose ADJUST COLOR > LEVELS. I set my Input Levels to the following from left to right: 10, 0.92, 236. Your Levels dialog box should look like Figure 3 below if you’re using the wine image.
5. Click OK.
6. Save this file with a new name, such as “wine_sail.png.”
7. I then opened the sailboat shot. This file has two layers: the background layer is the original image, untouched. The mask layer contains a masked version of the sailboat..
8. I want to darken this image a bit, so we have better separation and contrast when the two shots are merged. Once again, go to EFFECTS > ADJUST COLOR > LEVELS. The Input Levels for this image should be: 79, 0.94, 255.

Who Was That Masked Schooner?

Our first mask is for the sailboat. We want to hide as much of the background as possible. To do this, we’ll create a Vector Mask by using – gasp – the Pen tool. The Pen tool will give us a lot of control when creating the mask, and flexibility in editing the mask after it’s initially created.

1. Zoom in to 150%, and using the Pen tool, click at the top of the sail to start the path, then follow the main shape of the boat, clicking the mouse to change direction, to create an outline around the major shape of the sailboat. To create a curve section of the path, click, and drag the mouse. Don’t worry if you’re not 100% accurate at this point; after we’ve created the main outline, we can tweak the individual points with the Sub-Selection tool (try doing that with the Polygon Lasso!). Basically, we need to trace the main outline of the sailboat. You can see the completed path in Figure 4. Set the fill for white, since we’ll be using this same shape as a bitmap mask later on in the tutorial, and without the white fill, it won’t work.

For much more information on the Pen tool, be sure to check out Kim Cavanaugh’s Pen tool tutorial, complete with video segments. In fact, you may want to see Kim’s tutorial BEFORE you go any further. It’s a freebie on CMX.

2. Select both objects in the Layers panel by Shift-clicking them.
3. Go to MODIFY > MASK > GROUP AS MASK. Instant hidden background! Well, most of it, anyway. Take a look through the sails; zoom in if you need to. We can still see water in the background as in Figure 5. No worries though, and no need to start over. Fireworks will let us add to the mask while it’s grouped with our bitmap image. This next part is cool, I must admit.
4. Select the Mask object in the Layers panel by clicking on it. You’ll see it outlined in yellow when it’s active, and the path itself also displays in yellow. Zoom in to at least 300%, and select the Pen tool. We’ll start at the top of the sail. If you don’t see the tip of the sail, hold down the space bar and your Pen tool will switch to the Hand tool. Click and drag to scroll higher in your image.
5. When you’re at the top of the sail, release the space bar and click once at the highest point where you still see the sky/water background. Continue to click the Pen tool, following the outline of the sail (it’s important that you go in this direction first, rather than following the easy straight line of the





mast). If you run out of canvas area, just hold the space bar down again and scroll lower. The Pen tool will wait until you've let go of the space bar before it starts to work again.

- Continue to follow the outline of the sail until you reach the bow of the boat. Click to change direction, and begin to follow the top railing, back towards the mast. Don't panic if part or all of your sail gets covered by the mask; this is only temporary. As you trace the outline, feel free to click the Pen tool to set a straight point, or click and drag to create a curved path section. If you need to adjust the path after it's been created, use the sub-selection tool.
- When you get to the mast, click one last time to change direction, and

- move your Pen tool back to your starting point at the top of the sail. Click on the starting point to close the path. You've just added to your vector mask, knocking out more of the background. Cool, huh? And you didn't even need to ungroup the two objects!
- Continue this process to mask out other parts of the background until all you see is the sailboat. Make sure to save your file as you go; you're doing a lot of work and I'd hate for it to get lost in a crash.

A Tempest in a Teapot or a Sailboat in a Wine Glass

Now for the first real step in our collage. Make sure you can see both images in the workspace.

- Create a new layer in the wine glass image. Make sure this is the active layer.
- Using the Pointer tool, select the bitmap object in our sailboat image.
- Click and drag it onto the canvas of the wine shot over the top of the wine glass. By dragging the bitmap object, we bring along the image and the mask.
- Use the Transform (scale) tool to shrink the boat to a size that fits in the bowl of the wine.
- Position the boat so that it's resting on the wine, centered on the surface of the wine.

If you take a good look at the new image, it looks like the boat is on top of the glass. We need to change this to make our illusion more realistic. There are a couple ways to do this:

- Use the Eraser tool
- Use a Bitmap mask

Guess which one we're going to use? Yes, we're going the Bitmap mask route, for a couple reasons. While this method isn't as fast as using the eraser tool, it's non-destructive; we're not deleting any pixel data. It's also more flexible in terms of editing. It's not so easy to "un-erase" pixels, but it's a snap to change what a mask shows or hides.

- Create a duplicate layer of the background (wine glass) layer, by dragging the background layer onto the New Layer icon in the Layers Panel. Move this layer to the top of the Layers panel

by clicking and dragging it. Our sailboat is now hidden.

- Rename this layer to "glass mask."
- Adding the bitmap mask is quite easy in this case. Make sure the glass mask layer is active.
- Move your cursor over top of the sailboat vector mask.
- Hold down the Control/Command key and click once. Instantly, you will see a bitmap selection – identical to the vector mask – appear on the canvas.
- Invert the selection by going to **SELECT > SELECT INVERSE**
- Ensuring that the glass mask layer is still active, click on the Add Mask icon at the bottom of the Layers panel. Our sailboat reappears! But wait – it looks just like it did before. Not to worry; that's why we created the bitmap mask. Notice that this mask is the opposite of the vector mask. The white area represents the part of the image we can see through the mask. It may be a little strange to wrap your head around in this case; we usually think of a mask as hiding a lot and showing a little. In this case, we're showing a lot and only hiding the area occupied by the sailboat.
- Choose the Zoom tool, then click and drag a box where the sail and glass rim are located. This should zoom you in to about 650%.
- Now, we're going to alter this bitmap mask. Unlike a vector mask bitmap masks are changed by using the Brush tool. White color subtracts from the mask, black adds to the mask.
- Select the Brush tool, and set its size to 1 pixel and choose Soft Rounded from the Stroke Category dropdown menu in the Properties inspector. Make sure the stroke color is set to white and that the mask is selected.
- Carefully draw a line through the sails where the rim of the wine glass should be. The goal here is to show just the rim; that's why we chose a very small brush, and zoomed in so close to the work area. If you make a mistake, or go off course, don't worry; change the stroke color to black and paint over the mistake. Problem solved, and no need to use the undo command. Figure 8 shows a close-up of this in action.
- Use this same technique, but with a larger soft brush (15, for example) to

figure 3

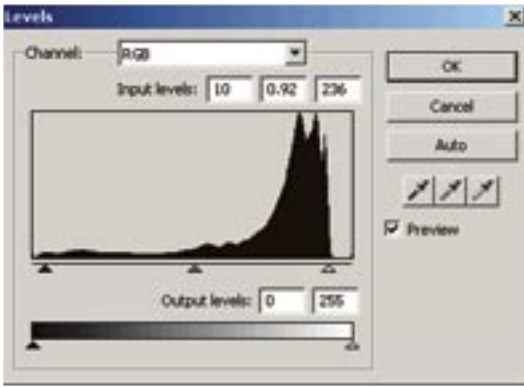


figure 4



figure 5



"Combining images is part of the fun with programs like Fireworks; making them look like they belong together – fooling the eye, as they say – is challenging and rewarding"

touch up any areas around the boat that seem to stand out. For example, your original vector mask may have left behind a few dark blue pixels from the original water background. You can add to the bitmap mask to hide any of these artifacts. I also used this method to soften up the area where the hull of the boat and the wine meet.

This Is NOT the Blue Nose!

1. One thing you might notice now – if not before – is that our sails look a little on the blue side. Too cool a color with all that wine, so we'll warm things up a little by using the Replace Color tool, found in Fireworks MX 2004. This tool will selectively replace one color – or range of hues – with another color.
2. Select the sailboat layer. Make sure the sailboat image is active, NOT the mask.
3. Zoom in so that the sailboat fills the workspace.
4. Choose the Replace Color tool from the tool bar. Look down at the Properties Inspector.
5. Click on the color well that says Change. When the Color Picker appears, move your cursor midway up the sail at the bow of the boat. Click once. You'll see that the Change color has switched to a pale blue.
6. Now click on the To color well. Rather than picking a color, put this value in the input box of the Color Picker: #FFF3CE. Use the following settings for the tool's attributes:
 - Size: 40
 - Shape: Round
 - Tolerance: 55
 - Strength: 100
 - Colorize: Un-checked
7. Now just paint over the sailboat. Don't

even worry about going outside the lines. Between the tolerance setting and the mask, there will be no problems to see. Below are before and after examples of using the Replace Color tool.

If you're using Fireworks MX or earlier, you can get the same effect by using the Hue/Saturation Live Effect, or the Curves Live Effect. It takes a bit more time, but the results should be decent.

The Clean Up

With our boat afloat in the wine, and the color cast taken care of, we just have a last few touch-ups to make, all of which are applied to our glass mask layer. As I mentioned earlier, this wine shot was done fairly fast, and as a result, there are a few reflections in the wine bottle and glass that I'm not happy with. Our friend, the Rubber Stamp tool will be coming to our aid.

1. Make the glass mask layer active, and click on the bitmap object to make sure it's active.
2. Choose the Rubber Stamp tool from the Toolbar (Shortcut key: S). Sample a clean area of the wine in the wine glass, and stamp over the weird reflections in the wine glass.
3. Use the Blur Tool to soften the edges of your Rubber Stamp work where necessary. Try an Intensity of 50 or lower for the Blur tool. You can also use the Blur tool to soften up some of the shadowy reflections on the wine bottle itself.
4. Select the Sharpen tool. Set the size to 20 and the Intensity to 15. Paint over the VQA logo only once to give it a bit of added snap. Don't keep painting it, or the sharpening will become heavy and undesirable.

Adding a Better Highlight

One last thing to adjust is the highlight on the wine bottle. This image was photographed with a diffused hot light, and the circular highlight is a mirror reflection (also called a specular highlight) of the light source. I'm partial to a nice long, clean highlight, so we'll add one that appears to follow the contour of the bottle.

1. Create a new layer and call it "highlight."
2. Start by drawing a rectangular vector shape that's close to the width of the circular highlight, and that runs in height from the middle of the original highlight right down to the label on the bottle.
3. Fill this shape with white, and set the Feathering to "2" in the P.I.



figure 6

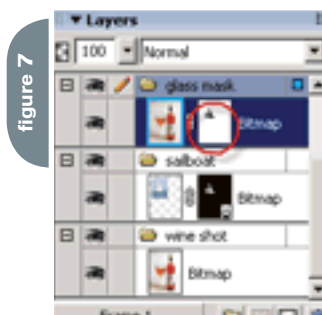


figure 7



- Ungroup the rectangle by pressing CTRL/CMD+SHIFT+G.
- Using the Pen tool, add three more control points: one in the middle of the top path of the rectangle, the other two on the side of the path about 70 pixels from the top. Figure 11 shows the arrangement of the control points and the position of the rectangle.
- Select the Sub-Selection tool and move the top center control point so that it is at the top of original highlight. You will have a pointed end now. Grab the Pen tool again, and click and drag the center control point horizontally. This will extend the Bezier control arms for the center control point. Drag the right handle down slightly so the curve of our new highlight matches the original circle. To reposition the control point itself, switch back to the Sub-selection tool. You can temporarily toggle between the Pen tool and the Sub-Selection tool by holding the Control/Command key down.
- Set the Bezier handles for the two control points on the right side of the highlight, and make minor adjustments so the highlight appears to curve following the shape of the wine bottle.



figure 8



figure 9



figure 10

Set the Bezier handles for the lower control point on the left of the highlight to have it curve slightly. This is delicate work so you may want to zoom in while you are doing this. Figure 12 shows a close up of the finished highlight.

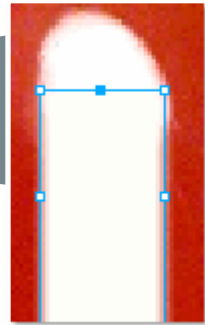


figure 11

In the case of a couple curves, I found it useful to use only ONE Bezier arm. To get rid of one arm, use the sub-selection tool to click and drag the unwanted control arm back to the actual control point. Don't be alarmed if your remaining Bezier arm goes a little crazy; you can bring it back in line once you have the other one disabled.

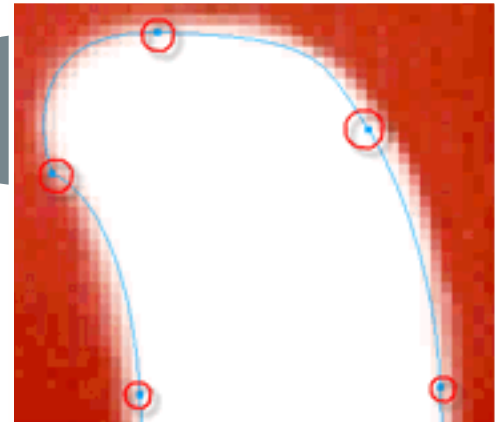



figure 12

And there you have it; we have set sail on the roseate sea, creating an interesting and relatively believable photo-illustration.

Conclusion

This Fireworks project may take you a little longer than some others, but I believe the results are worthwhile. We've done some pretty cool masking, and touched up an original image to boot. Techniques like this have multiple uses, limited only by your imagination. The completed file is above. You can click on it to see a larger version. 

Jim Babbage (contributor from CommunityMX) comes from the photographic world, and has spent many years as a professional studio photographer. His involvement with the Web began in the mid-90s when the company he worked for had just gotten online. Born and raised in Toronto, Jim teaches imaging, Web design, and photography at Centennial College's Centre for Creative Communications (www.thecentre.centennialcollege.ca). He is a partner in Newmedia Services (www.nms123.ca), a small communications company, specializing in the things he teaches. He is a regular contributing partner to Community MX (www.communitymx.com), where he's written many articles and tutorials for Fireworks, Dreamweaver, and other general Web topics. He has been a guest speaker at TODCON for several years. jim@communitymx.com



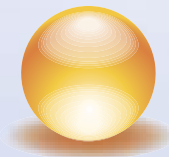
figure 13

H  S T I N G . c o m

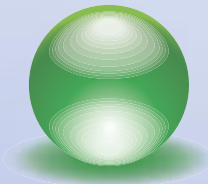
THE FIRST NAME IN HOSTING



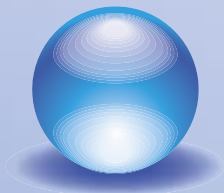
Developers!



Developers!



Developers!



We Love Them!

SERVE. SUPPORT. SECURE. STORE.

www.hosting.com

Getting started with <CFLDAP>

A step-by-step guide to the basics
by tony paolillo

The <cldap> can be very simple or very complicated. It all depends on what you're looking to do and how you want to authenticate your users.

For quite some time I wanted to authenticate the users on my intranet through Active Directory. I spent countless hours searching the Web for someone to explain the basics in a way I could understand being a basic ColdFusion developer. Most articles and tutorials I came across were for intermediate or advanced users.

I wound up learning most of it on my own after getting an LDAP browser and snooping around in Active Directory for what I was looking for. To my surprise accessing Active Directory wasn't as complicated as it may seem. There are tutorials out on the Web that show you different ways to access Active Directory and references that show the different attributes of Active Directory that you can query.

Every time I ask someone about accessing Active Directory using the <cldap> tag, they're like "No Way, that's too complicated for me." Or they respond with "I know nothing about Active Directory."

Well, this article will show you how to authenticate through Active Directory with little or no <cldap> or Active Directory experience.

I have a Windows 2000 Server and Windows 2003 Server and had to change my code for each domain because of the differences in Active Directory. Trial and error led me to find a way to use the same code for both Windows 2000 and Windows 2003 domain controllers, so I decided to pass on my knowledge.

<cldap> can be very simple or very complicated. It all depends on what you're looking to do and how you want to authenticate your users. I'm going to show you how you can use <cldap> to authenticate your users using just the

basics. Nothing complicated!

Let's Get Started

First you should understand the basics of the <cldap> tag. I'm only going to show you the basic options to use for this example.

First let's assume that I have a domain controller called "ns1" and my domain is "adtest.com."

Here's a snapshot of a cldap query that I used to find a user in my Active Directory:

```
<cldap action="QUERY"
  name="GetUserInfo"
  attributes="dn"
  start="dc=adtest,dc=com"
  scope="subtree"
  filter="(&(objectclass=user)(samaccountname=#form.cfusername#))"
  server="ns1.adtest.com"
  username="administrator@adtest.com"
  password="password"
>
```

Let's start with the attributes, the attributes are the information that we want to query from Active Directory. Think of this like a "Select" statement in a regular query. In this case we just want "dn". (distinguishedName).

In the start field, you only need to define the "dc" (dc means domain content rather than domain controller in this context). Notice I used "adtest" as the first dc and "com" as the second dc. So if you have a .NET domain, just replace the dc="com" with dc="net."

The next item is the "scope." I think this is where some users make the mistake of not defining. The first time I tried to access my Active Directory I thought I didn't need to define the "scope." My Active Directory is set up with many OUs and with about three levels. The default option for "scope" is "onelevel." If you let it default you will only be querying one level below entry. So in my case, users

within the one-level OUs could authenticate just fine; the other users could not. Imagine my headache figuring that out!"

So now I like to use the "subtree" option. The "subtree" option queries the entry and all levels below it.

The next item is filter. In the cldap query above notice that I used the "objectclass=user." This is what we are querying for. If I just wanted to query the Active Directory for a computer name, I would have "computer" instead of "user." There are many other objectclass types to choose from, but I don't want to confuse you with objectclasses we don't need for this example.

The next filter is the samaccountname. This is the same account name as in the Windows Active Directory. It's basically the user login name. Here we put the login name that came from the form (#form.cfusername#).

Next is the "server." This is straightforward. Just put in your complete server name including the domain name like this "ns1.adtest.com".

The username is where I found the difference between Windows 2000 and Windows 2003 domain controllers. Windows 2000 requires you to have the "@adtest.com" at the end of all names and Windows 2003 doesn't. I found that if I just add it into my code like I did above I wouldn't have to worry about either domain since Windows 2003 accepts it. Notice that I used the administrator to authenticate to Active Directory. You can use whatever username and password you want that has access rights to query your Active Directory.

What this query does is find the user in the Active Directory. It does a lookup to determine if there's a samaccountname that matches the #form.cfusername#. If the user exists then we can move on to the next section. If not, then we should kick an error saying that the username wasn't found. I do a recordcount against the "GetUserInfo" query. If it comes back



with a 0, then it didn't find the user in AD.

Authenticating a User

Okay, here we're going to assume we got a 1 with our recordcount "cfif" statement. Here's the next query example that actually authenticates the user:

```
<cfif #getuserinfo.recordcount# gt 0>
<cftry>
<cfldap action="QUERY"
name="AuthenticateUser"
attributes="givenname,samaccountname,
dn,cn,mail"
start="dc=adtest,dc=com"
maxrows="1"
scope="subtree"
filter="(&(objectclass=user)(samacco
untname=#form.cfusername#))"
server="ns1.adtest.com"
username="#form.cfusername#@adtest.
com"
password="#form.cfpassword#">
<cfset LoginMessage = "User
Authentication Passed">
<cfcatch type="any">
<cfset LoginMessage = "User
Authentication Failed">
</cfcatch>
</cftry>
<cfelse>
<cfset LoginMessage = "Username not
found">
</cfif>
```

I left in the <cftry> statements for you to see how I catch if the authentication fails.

Most of the tag is the same as before. The main difference is that we use the "#form.cfusername#" and "#form.cfpassword#" to authenticate to Active Directory. If we successfully authenticate to AD then the script just moves on, but if not we get an error and that's when I catch the error and set a LoginMessage as failed.

If we get a successful login we can use the attributes from the cfldap query. There are many attributes that you can choose from. The main ones I use are "givenname," "cn," and "mail." "givenname" is the user's first name and "cn" is the user's "Display Name" in Active Directory. You can use a lot of information that AD has in it. Now you can display the user's name on a welcome screen or whatever. Like this: "Welcome # AuthenticateUser.cn#".

If you want your users to put the "@adtest.com" after their login name, then take out the "@adtest.com" in the username field. Otherwise you need to leave it in if you're using Windows 2000.

That's really it. You don't need to get any fancier for a simple Active Directory authentication system. There are many other options available and some of them might work for you depending on your environment. In my next article I'll go into a lot more options and attributes.

Here's my example index page and login page.

```
Index.cfm
<html>
<head>
<title>Active Directory Login
Example</title>
</head>
<body>
<div align="center">Active Directory
Login Example
</div>
<cfform action="login.cfm"
name="login" method="post">
<table border="0" align="center" cell-
padding="0" cellspacing="0">
<tr>
<td>UserName:</td>
<td>
<cfinput type="Text"
name="cfUserName" size="20"
required="yes">
</td>
</tr>
<tr>
<td>Password:</td>
<td><cfinput type="Password"
name="cfPassword" required="yes"> </
TD>
</td>
</tr>
<tr>
<td>&nbsp;</td>
<td><input name="Submit"
type="Submit" value="Submit"> </TD>
</td>
</tr>
</table>
</cfform>
</body>
</html>

Login.cfm
<cfparam type="string"
name="LoginMessage" default="">
```

```
<cfldap action="QUERY"
name="GetUserInfo"
attributes="dn"
start="dc=adtest,dc=com"
Scope="subtree"
filter="(&(objectclass=user)(samacco
untname=#form.cfusername#))"
server="ns1.adtest.com"
Port="389"
username="administrator@adtest.com"
password="password">
```

```
<cfif #getuserinfo.recordcount# gt
0>
<cftry>
<cfldap action="QUERY"
name="AuthenticateUser"
attributes="givenname,samaccountname
, dn,cn,mail"
start="dc=adtest,dc=com"
maxrows="1"
Scope="subtree"
filter="(&(objectclass=user)(samacco
untname=#form.cfusername#))"
server="ns1.adtest.com"
Port="389"
username="#form.cfusername#@adtest.
com"
password="#form.cfpassword#">
<cfset LoginMessage = "User
Authentication Passed">
<cfcatch type="any">
<cfset LoginMessage = "User
Authentication Failed">
</cfcatch>
</cftry>
<cfelse>
<cfset LoginMessage = "Username not
found">
</cfif>

<html>
<head>
<title>Active Directory Login</title>
</head>
<body>
Login Status:<br><br>
<cfoutput>#LoginMessage#</cfoutput>
<br><br>
<cfif #getuserinfo.recordcount#
gt 0 AND #LoginMessage# neq "User
Authentication Failed">
<cfoutput>#AuthenticateUser.cn#</
cfoutput>
</cfif>
</body>
</html>
```

Tony Paolillo is a Microsoft Certified System Engineer (MCSE) and a Microsoft Certified System Administrator (MCSA). He has been working with computers for over 20 years and is currently running the IT department of a large petroleum distributor. Part of his duties is to develop an intranet so he chose ColdFusion as his programming language. He has been using ColdFusion for the past year and a half. He enjoys writing tutorials and helping others with problems that he has overcome. He has developed three of his own applications including one that authenticates users through Active Directory and assigns security levels. See <http://www.cfpurchasepickup.com/cfadsecurity.htm>. tpaolillo@gmail.com

Working with Audio in Captivate

The Hollywood effect
by lisa heselton

In part one of our article, we created a very basic Captivate movie. Now we'll enhance our project with some audio elements – narration, sound effects, and music.

Recording Narration

A word about narration and audio: remember to think of the environment your users will be in when viewing your project, and make sure that any audio used will enhance the project and not distract from learning.

We've already recorded our demonstration, so we'll need to do the following steps to put a script into Captivate and then record our narration:

1. Open an existing Captivate project. For this article, we'll open the demonstration movie we previously made. Make sure your microphone is connected and active.
2. Select the slide to which you want

to add audio (or where you want the audio to begin) in either the Storyboard or Edit mode.

3. Choose Edit > Properties then click the Notes button.
4. Type in the text you want to be spoken, and click OK. You can also insert a Caption for users who can't hear the audio to ensure that users can navigate successfully and that they can comprehend the information even when the audio is off. Continue this process for each of the slides that you'll narrate.
5. Click the Audio button on the Main toolbar.
6. Under View Script, select either of the following:
 - Captions (to view the contents of a Caption that's present on the current slide)
 - Slide notes (to view notes for this slide)

Now that you have the text to speak, you can begin recording your voiceover.

1. Open Background Audio preferences and click the "Record new" button to open the Record Audio dialog box.
2. Click the Options button. The Audio Options dialog box opens.
3. Select Microphone as your input source.
4. Click the "Calibrate input..." button. The Calibrate microphone dialog box opens.
5. Speak into the microphone. The Recording Level meter on the right side of the dialog box changes color and will tell you if the input level is acceptable or not. Once it's okay, click OK to return to the Audio Options dialog box.
6. Select your audio quality from the Audio Quality pop-down menu and click OK to return to the Record Audio dialog box.
7. Click the Record button and speak the caption text or slide notes into the microphone.
8. When you've finished recording your narration for this slide, click Next.
9. Read the contents of the next slide note or caption.

10. Repeat Steps 8 and 9 until you've read all of the narration for the entire movie.
11. Click the OK button.

Captivate now adds each of the audio clips to the appropriate slide.

Listening To and Modifying Narration
Once you've finished, the audio files are converted into MP3 format and added to each of your project slides.

To listen to the audio track, press the spacebar. A red line (also known as a playhead) moves from left to right across the timeline to indicate where in the timeline you're currently hearing the audio. You can drag this line back and forth to pick up the sound at that point.

Editing Audio

So now that we've played back our narration, we may have found parts we want to edit. We may have left too long of a pause, said "um," or left out a dramatic pause. Whatever the reason, editing the audio is simple. You can modify the audio file using Captivate's built-in audio editor.

Here's one way to remove an unwanted sound snippet:

1. Choose Audio > Edit Audio. You can also open the Edit Audio dialog box by right-clicking the audio slide of the timeline and selecting Edit Audio from the context menu, or by clicking the speaker icon located on each slide with audio.
2. Play the audio track by clicking the Play button or pressing the spacebar.
3. Locate the unwanted audio and make a note of where it starts and stops.
4. Position the playhead at the beginning of the noise (start time) and then click and drag until it's at the end (stop time) to select the entire area. You can play the audio to make sure you haven't missed any of the sound or accidentally selected too much.
6. Click the Delete Selection button to remove the sound.
7. Play the selection to ensure that you're happy with the deletion.
8. Click OK to close the Audio dialog box.



Adding Silence

The Insert Silence command lets you remove mistakes, unwanted hissing sounds, or create pauses in your audio track. You access the Insert Silence command through either the Edit Audio or Edit Timing dialog boxes.

To insert silence, do the following steps:

1. Open the Edit Audio dialog box.
2. Select the block of audio where you want to add the silence, pressing the spacebar to play the audio track if needed.
3. Click the Insert Silence button (on either the Edit Audio or Edit Timing dialog boxes) to open the Insert Silence dialog box.

You can specify the amount of silence (in seconds) to add to the audio file. If you select a snip of audio that you want to silence, Captivate will calculate how many seconds to add and select the Current Selection option.

The menu options are:

- _ Current selection
- _ Playhead position
- _ Start of audio
- _ End of audio

4. Click OK to add the seconds of silence you specified.

Importing Audio

Now that we've added some narration and done some basic edits, let's look at how to add a sound effect or supporting audio file to our movie.

Tip: You can also attach sounds to objects in your movie to emphasize actions or reinforce learning objectives.

You can import audio into Captivate that's in MP3 or WAV format. To import an audio track, do the following:

1. Choose Audio > Import.
2. Browse to the file you want to import and click Open. Or, you can import an audio track by right-clicking the slide to which you want to add the audio track, then choosing Properties, selecting the Audio tab, and clicking the Import button.
3. Browse to the file you want to import and click Open.

Captivate imports the audio file and converts it into MP3 format if necessary.

If the audio file is longer than the slide, Captivate will display the Audio Split Options dialog box.

You will have three options:

_ Extend the display time of the selected slide to fit this audio file: Use this option only if you are importing audio for the current slide. Captivate will put the entire audio track on the slide and increase the slide's display time.

_ Edit the audio timing and manually distribute the audio file over several slides: Use this option if you want to edit the timing of the imported audio file and distribute it manually over several slides. In the Edit Audio Timing dialog box, you can specify how your audio files are distributed. When you import the audio, slide dividers appear at the end of the audio waveform. To manually distribute the audio across a number of slides, do the following:

1. Click the Play button or press the spacebar to hear the audio.
2. When you want the audio to be split onto the next slide, click the Start Next Slide button (or press Ctrl+S). The slide divider for Slide 2 moves over to the point where the insertion point is located.
3. Repeat Steps 1 and 2 until the entire audio file has been distributed as desired.
4. Click OK. Captivate distributes the audio file and creates a new audio track for each slide.

_ Retain the current timing and distribute this audio file over several slides: Use this option if you want Captivate to distribute the audio across several slides without altering the timing. Captivate will do it automatically.

Adding Background Audio

Captivate lets you add a soundtrack to the background of an entire movie.

To add background audio, do the following:

1. Choose Movie > Preferences.
2. Select the Background Audio tab.
3. Select one of the following buttons:
 - _ Record New to record a new audio file
 - _ Import to import an existing audio file
 - _ Audio Library to select a clip from the Audio Library

The Background Audio tab in the Movie Preferences dialog box includes the following options:

- Fade In and Fade Out areas to specify fade in and fade out values.

- "Lower background audio volume on slides with additional audio" to reduce the background volume on slides that have other audio files such as voiceover narration.
- "Loop audio" to play back the background audio over and over again.
- "Stop at end of movie" to stop the background audio when the movie ends.

Tip: There are a number of pre-recorded background sound loops and files in the MP3 format that ship with Captivate in the Program Files/Macromedia/Captivate/Gallery/Sound folder.

Select the file and click Open. The Import Audio dialog box closes, and you are returned to the Background Audio dialog box.

To preview your sound file, click the Play button.

When you are finished, click OK to close the dialog box. The background audio is added to your movie.

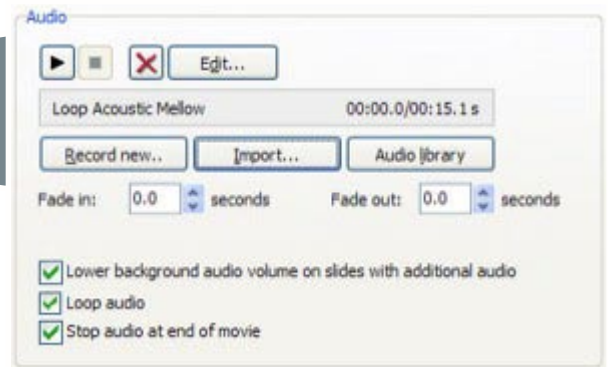
Don't be afraid to experiment with the audio properties of your movie and see what exciting audio effects you can come up with. Have fun! ☺

Lisa Heselton is eBusiness manager for NISH in Vienna, VA, a national non-profit organization working on behalf of the severely disabled, and she runs their internal user group NISHMMUG (<http://www.dhsmmug.us>). Lisa's background includes marketing, video production, writing, business/project/artist management, presentations, sales, and training. Her blog is at <http://lisa.mxdj.com>. l.kavka@gmail.com

figure 3



figure 4



Animating Flex Charts

Enter another dimension
by peter ent

Using pictures to convey information is as old as man. We first used cave drawings to get a point across. The old saying “a picture is worth a thousand words” is never truer when it comes to looking at spreadsheet data. Perhaps cave art was primitive man’s way to consolidate a lot of information.

Flex has its own way of drawing pictures – Chart controls. Since Flex is based on Flash, you can expect a lot from the graphics package built into Flex. The Flex Chart controls not only draw your standard business graphics, they can be animated to drive home the point.

In this article I’ll introduce you to the Chart controls and show you how to make a simple chart and animate it to add another dimension to the information the chart already displays.

As you know, looking at a column of numbers in a spreadsheet is tedious. Charts help to decipher the information in meaningful ways. But what happens if the data isn’t just last quarter’s sales, but last year’s sales, month by month? To see any significant changes in the figures you could either flip between charts or digest those numbers into a single chart and show trend lines or regression fits: something to indicate what’s been happening to your sales.

Another way to look at the data is to animate it and control the animation. Your chart begins with January and morphs into February, then into March. You can watch as the numbers show the sales of Widgets dropping as Cogs rise. You can even stop the animation and go back and see it over and over again.

This kind of animation is easy to do with Flex. First you need your data. For this article, I’m going to keep it simple because of space constraints, but you should get the picture (no pun intended).

Our sales figures for January are shown in Table 1.

This data set has one row for each sale item: widgets and cogs. Each column represents a region.

A bar chart is used to show the figures. The time is advanced by telling the chart to use a different data set from moment to moment. This is similar to what a real-life application would do: a query would be made to a server to fetch the sales figures by region for a particular month. The data from the server would be bound to the chart.

Figure 1 shows the application. You can’t see the animation in print, but just imagine the bars sliding gracefully up and down as the date advances along the slider below the chart.

Building the Chart

This kind of chart is easy to do in Flex using the `<mx:ColumnChart>` tag. You define the chart and each series. In this case there are three series: one for each region. The series is told which fields from the data set to graph.

```
<mx:ColumnChart id="salesChart"
showDataTips="true"
dataProvider="{zero_data}">
  <mx:horizontalAxis>
    <mx:CategoryAxis
categoryField="product"
dataProvider="{zero_data}" />
  </mx:horizontalAxis>
  <mx:verticalAxis>
```

```
<mx:LinearAxis maximum="3600" minimum="0" name="Sales" />
  </mx:verticalAxis>
  <mx:series>
    <mx:Array>
      <mx:ColumnSeries
name="Northeast" yField="northeast"
showDataEffect="morph" />
      <mx:ColumnSeries
name="Mid-west" yField="midwest"
showDataEffect="morph" />
      <mx:ColumnSeries
name="Southeast" yField="southeast"
showDataEffect="morph" />
    </mx:Array>
  </mx:series>
</mx:ColumnChart>
```

The `ColumnChart` tag provides the space for a chart as a series of columns (Flex also has a `BarChart` if you want horizontal bars). Since the data being charted doesn’t contain any X coordinates, you can use a `CategoryAxis` to divide up the horizontal elements. In this case, those elements are the products widgets and cogs so the `categoryField` is set to the column name product.

Flex automatically calculates the minimum and maximum values for the vertical axis from the data. Because the chart is going to display many sets of data, the chart’s axis range might change, which would look odd. To prevent that, a vertical axis is specified with specific minimum and maximum values.

Finally the `ColumnSeries` is defined for each region. You can see how the `yField` attribute relates to the data. The `name` field is used as a label. The `showDataEffect` attribute names the special effect to use when the data changes.

Effects

The animation is also easy since Flex comes with several effects specifically for charts. For this example I chose the

table 1

Product	Northeast sales	Midwest sales	Southeast sales
Widgets	3000	1400	985
Cogs	1450	1100	200



SeriesInterpolate, which allows for a smooth transition as the data changes.

```
<mx:Effect>
  <mx:SeriesInterpolate name="morph"
/>
</mx:Effect>
```

That is, if the value of Widgets in the Northeast for March was 2400 and then changes to 2500, the SeriesInterpolate effect would gradually increase the height of the bar from the 2400 mark to the 2500 mark.

You can see how easy it is to specify animation for the chart. The effect's name (morph) is used as the value of the showDataEffect on each series.

You can also control aspects of the effect such as how long the effect should last and how much the graphic should change.

Animating the Chart

Some of the data for the chart is shown in Listing 1. For the sake of this demonstration, I have coded the data in ActionScript in a <mx:Script> block. But just imagine that this data is available on a server.

You would call these ActionScript statements from within a function that would be invoked when you retrieved data from the server.

Summary

While charts certainly make presenting numbers a lot more appealing and easier to understand, animation makes the data come alive. Being able to see how the data is changing, in real-time, can give you more insight into the problem you're analyzing.

Setting up a chart in Flex is easy; animating it is a breeze.

```
var sales_data:Array = [
  {month:"January",
    data:[{product:"Widgets",
north-east:3000, midwest:1400, south-east:985},
        {product:"Cogs", north-east:1450, midwest:1100, south-east:200} ]},
  {month:"February",
    data:[{product:"Widgets",
north-east:3300, midwest:1450, south-east:1180},
        {product:"Cogs", north-
```

```
east:1500, midwest:450, south-east:350} ]},
    {month:"March",
    data:[{product:"Widgets",
north-east:3100, midwest:1500, south-east:1550},
        {product:"Cogs", north-east:1900, midwest:600, south-east:700} ]},
    .
    .
    .
    {month:"December",
    data:[{product:"Widgets",
north-east:760, midwest:1200, south-east:710},
        {product:"Cogs", north-east:3100, midwest:1700, south-east:2600} ]}
]; // end of sales_data
```

The sales_data Array has 12 items: one for each month. Each item in the sales_data Array as two components: the name of the month and the data making up the sales figures.

If you look at the MXML definition for the chart above, you see that the dataProvider is set to "{zero_data}" and not to "{sales_data}." This is another trick. Setting up some empty data when the chart is created will let the first real data be animated when it's added to the chart. In the code below shows the definition of the zero_data array.

```
var zero_data:Array = [
  {month:"",
    data:[{product:"Widgets",
north-east:0, midwest:0, southeast:0},
        {product:"Cogs", north-east:0, midwest:0, southeast:0} ]}
];
```

The idea is to tell the chart to present the data for January (or sales_data[0]). After a short time the chart is told to present the data for February (or sales_data[1]). This continues until all the months' data have been shown.

An interval function runSales is used to do this. The function takes the current month, tells the chart to display that month, then goes to sleep, and repeats the step when it wakes up.

```
function runSales() {
  adjustChart();
```

```
  if( ++currentMonth >= sales_data.length ) {
    clearInterval(timerID);
    timerID = 0;
  }
}
```

The adjustChart function tells the chart to display the proper data.

Listing 4 Adjusting the chart

```
function adjustChart() {
  salesChart.dataProvider = sales_data[currentMonth].data;
  salesChart.horizontalAxis.dataProvider = sales_data[currentMonth].data;
}
```

Chart Building

Sometimes you may not be sure of the number of series you'll need until you examine the data. For example, suppose you get your data from a server and the query you run asks for a year's worth of sales data. But you don't know how many regions you'll get back. In the example above the ColumnChart was hard-coded to display three ColumnSeries with specific names for the regions. If you had a fourth region it wouldn't get graphed unless you added a ColumnSeries to show it.

Adding a series to a chart using ActionScript isn't terribly difficult once you have some insight as to how MXML maps to ActionScript.

Take the ColumnChart definition above. Notice how the ColumnSeries is specified using a nesting of <mx:ColumnSeries> in <mx:Array> in <mx:series>? The <mx:series> MXML tag is analogous to the series property in ActionScript. That is, you reference the Array of ColumnSeries objects from ActionScript like this:

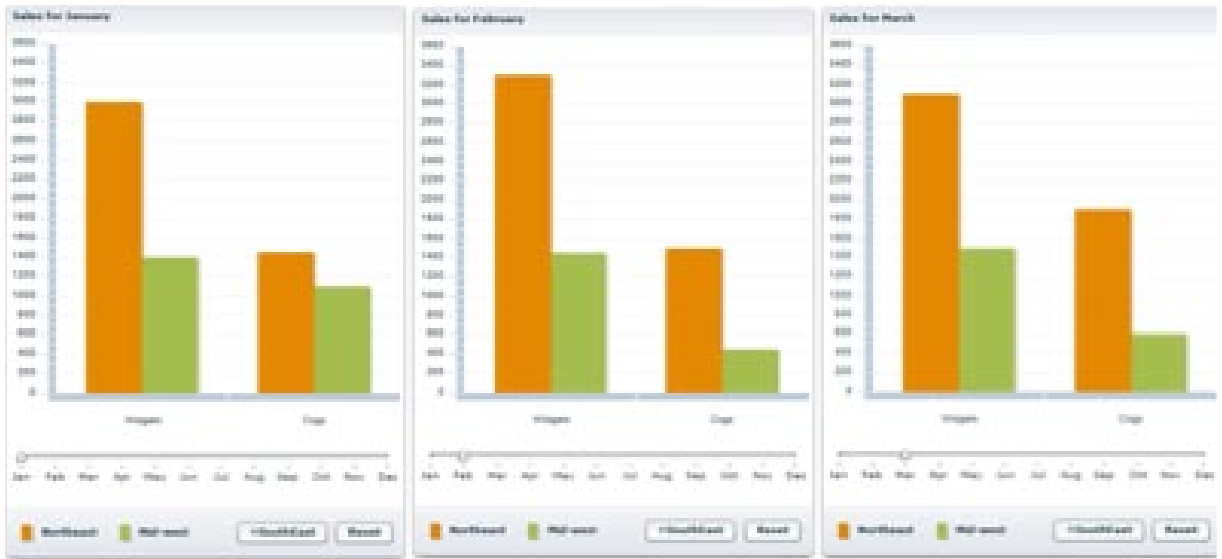
```
salesChart.series // the Array of the series
salesChart.series[0] // the ColumnSeries for northeast
```

Having access to the series means you can change the style or property of any series, or even add new series on-the-fly.

If you know you're going to be adding series to a chart dynamically, set up the chart



figure 1



in MXML but include an empty Array so you don't have to create the Array when you go to add the first series (just a shortcut).

```
<mx:ColumnChart id="salesChart"
showDataTips="true"
dataProvider="{zero_data}">
  <mx:series>
    <mx:Array>
  </mx:Array>

```

```
</mx:series>
</mx:ColumnChart>

var colSeries = new ColumnSeries();
colSeries.yField = regionID; //
retrieved from server
colSeries.name = regionName; //
retrieved from server
colSeries.showDataEffect = "morph";
salesChart.series.addItem(colSeries);
```

References

To see live examples of the Flex Charts, go to: <http://flexapps.macromedia.com/flex15/chartexplorer/explorer.mxml>.

You can download the code for this article from my blog at: www.markme.com/pent.

peter.ent@keaura.com



Looking to Stay Ahead of the *i*-Technology Curve?

Subscribe to these **FREE** Newsletters >

Get the latest information on the most innovative products, new releases, interviews, industry developments, and *i*-technology news

Targeted to meet your professional needs, each newsletter is informative, insightful, and to the point.

And best of all – they're FREE!

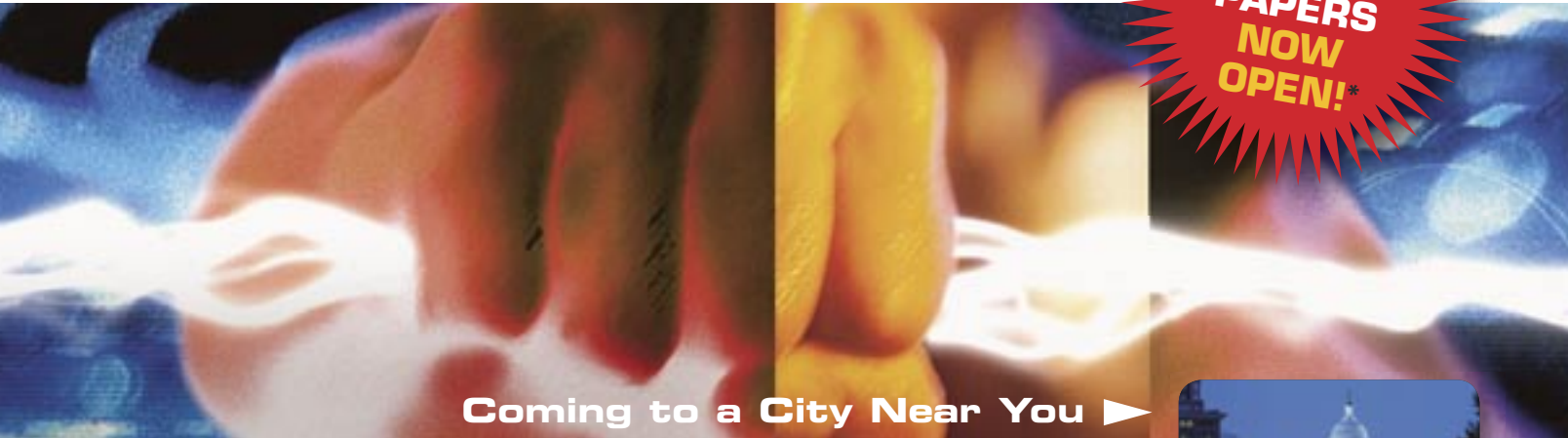
Your subscription is just a mouse-click away at www.sys-con.com



The World's Leading *i*-Technology Publisher



[Engage and Explore] the technologies, solutions and applications that are driving today's **Web services** initiatives and strategies...



Coming to a City Near You ▶

Web Services Edge Fall Conference Series

3 Dynamic Conference Programs Targeting Major Industry Markets

20+ seminars within 5 tracks will address the hottest topics & issues:

- ▶ Web Services: The Benefits and Challenges
- ▶ Web Services Security
- ▶ SOA (Service-Oriented Architecture) and ESB (Enterprise Service Bus) Strategies
- ▶ Interoperability, Incremental Integration, & Open Source
- ▶ The Management Process in Developing a Web Services Strategy

Program Features:

- ▶ Keynotes
- ▶ Tutorials
- ▶ Panel Discussions

Why Attend:

- ▶ Improve the return on your technology investment
- ▶ Develop & sharpen your strategy and identify key action steps
- ▶ Find new ways to reach and impress customers with Web services
- ▶ Maximize the power of your enterprise
- ▶ Protect your business from security threats
- ▶ Assess Web services as a viable option



Attention Exhibitors:

- ▶ An Exhibit-Forum will display leading Web services products, services, and solutions



The Westin Washington, D.C.
Washington, D.C.

September 7-8, 2005

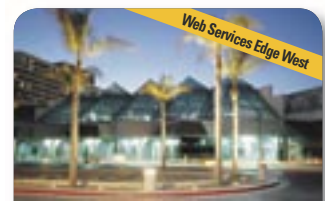
1



Helmsley Hotel
New York, NY

September 19-20, 2005

2



The Westin Santa Clara Convention Center
Santa Clara, CA

October 24-25, 2005

3

Register Today! www.SYS-CON.com/Edge2005

Sponsored by



*Call for Papers email: grisha@sys-con.com

For Exhibit and Sponsorship Information ▶ Call 201 802-3066

Produced by SYS-CON EVENTS

Let Flex Consume Your CFCs

A more programmatic approach to building RIAs
by dennis baldwin

In previous articles I've written about the power of integrating Flash and ColdFusion to build Rich Internet Applications (RIAs).

In this paradigm Flash provides the user interface while ColdFusion is responsible for handling the business logic and sending data to Flash either through Flash Remoting or Web Service calls. In this article I'd like to shift focus a bit and provide a more programmatic approach to building RIAs. By this I mean using XML to create the Flash interface. The answer lies in Macromedia's Flex.

This article will walk you through a simple application that retrieves employee records from a database so the user can update them. This is by no means a new concept, but I'd like to try a different approach. Forget about HTML, or even Flash forms for that matter, and see what Flex has to offer. If you've done any ColdFusion component development then I want to challenge you to experiment with Flex as your presentation tier. We'll

dig into this concept in greater detail but let's first configure the application.

Walking Through the Example Application

You can download the source code for the example application at www.db75.com/dev/mxdj/mxdj_0605.zip. Please refer to the Readme.txt file included in the zip archive. This file has all the information necessary to get the example application up and running.

Once you've deployed the files to their proper locations and updated flex-config.xml, you should be able to access EmployeeForm.mxml through its fully qualified name, i.e., <http://localhost:8080/flex/EmployeeForm.mxml>. Once Flex compiles this file you should see something similar to Figure 1.

Let's open EmployeeForm.mxml so we can walk through the code and get a general understanding of what's going on. You can review the mxml code in the following code.

figure 1

First Name	Last Name
Aaron	Smith
Carolynn	Peterson
Dave	Heartsdale
David	Barnes
Dominiquea	Branden
Erica	Manley
Francisco	Cabrera

First Name:

Last Name:

Email:

```
<?xml version="1.0" encoding="utf-8"?>

<mx:Application xmlns:mx="http://www.
macromedia.com/2003/mxml" creationComp
lete="getEmployeeList();">
```

```
<mx:Script>
<![CDATA[

    var selectedEmployeeID:Number;
    var selectedIndex:Number;

    // Gets the list of employees and
is bound to the combo box
    function getEmployeeList():Void
    {
        employeeWS.getEmployeeList.send();
    }

    // Get the employee when the selec-
tion is made in the datagrid
    function getEmployee(event):Void
    {
        selectedEmployeeID = event.target.
selectedItem.EMP_ID;
        selectedIndex = event.target.
selectedIndex;
        employeeWS.getEmployee(selectedEmpl
oyeeID).send();
    }

    // Update the employee info
    function updateEmployee():Void
    {
        var employeeObj:Object = new
Object();
        employeeObj.first_name = first_
name.text;
        employeeObj.last_name = last_name.
text;
        employeeObj.email = email.text;
        employeeObj.id = selectedEmploy-
eeID;
        employeeWS.updateEmployee(employeeO
bj).send();
        employeeDG.replaceItemAt(select
edIndex, {FIRSTNAME: employeeObj.
first_name, LASTNAME: employeeObj.
last_name});
```



```

    }
  ]]}
</mx:Script>

<mx:WebService
  serviceName="EmployeeService"
  id="employeeWS"
  showBusyCursor="true"
  fault="mx.controls.Alert.show('Error
Loading Data');" />

<mx:Panel title="Employee Form">

  <mx:DataGrid dataProvider="{emp
loyeeWS.getEmployeeList.result}"
  id="employeeDG" width="100%" change="g
etEmployee(event);">
    <mx:columns>
      <mx:Array>
        <mx:DataGridColumn columnName="EMP_
ID" width="0" />
        <mx:DataGridColumn
columnName="FIRSTNAME"
headerText="First Name" />
        <mx:DataGridColumn
columnName="LASTNAME" headerText="Last
Name" />
      </mx:Array>
    </mx:columns>
  </mx:DataGrid>

  <mx:Form>

    <mx:FormItem label="First Name">
      <mx:TextInput id="first_name"
width="200" text="{employeeWS.getEm-
ployee.result[0].FIRSTNAME}" />
    </mx:FormItem>

    <mx:FormItem label="Last Name">
      <mx:TextInput id="last_name"
width="200" text="{employeeWS.getEm-
ployee.result[0].LASTNAME}" />
    </mx:FormItem>

    <mx:FormItem label="Email">
      <mx:TextInput id="email"
width="200" text="{employeeWS.getEm-
ployee.result[0].EMAIL}" />
    </mx:FormItem>

    <mx:FormItem>
      <mx:Button label="Update" click="up
dateEmployee();" />
    </mx:FormItem>

  </mx:Form>

```

```

<mx:ControlBar />
</mx:Panel>
</mx:Application>

```

First off, when the application's creationComplete event has fired, we call the CFC and ask for a list of all employees. You'll notice that a WebService object is created and assigned the ID employeeWS. That's what we'll call the instance of this object throughout the application. The WebService object also has a serviceName attribute, which is bound to a named entry in our flex-config.xml file:

```

<service name="EmployeeService">
  <wsdl>http://localhost/mxdj/Employee.
cfc?wsdl</wsdl>
  <endpoints>
    <endpoint>http://localhost/mxdj/
Employee.cfc</endpoint>
  </endpoints>
</service>

```

This adds the Web Service to Flex's whitelist of authorized services. For development purposes you can enable any service to be accessible through the proxy or bypass the proxy altogether.

When the getEmployeeList method of the CFC is called, we handle the result by binding it to the DataGrid control. In Flex this is called Databinding. This is a powerful concept and I definitely encourage you to learn more about it. Databinding can help minimize the amount of code you have to write as well as simplify your design.

Now that the DataGrid is populated let's click on an employee record and you'll see a few items (first name, last name, and email) appear in the form below. This is done using the DataGrid's change event. If you've dealt with change events in the Flash IDE you can appreciate the fact that Flex's approach is cleaner. Each time a record is selected the change event is fired and the appropriate method is designated to handle the event; in our case it's the getEmployee(event) method. This method asks the CFC to return the employee object that matches the ID sent from Flex. You may have noticed that the employee ID is actually in the DataGrid but hidden in a column with zero width. For demonstration purposes I decided to hide the ID but in certain situations it definitely makes sense to show it.

When the employee object is returned

from the CFC you'll see the appropriate information displayed in the employee form. The form fields are bound to specific elements in the employee object. Whenever a result is returned, the form is automatically updated. You might notice the difference here compared to the way this has been approached using the Flash IDE. In most instances you'd have to set the value of the each text field explicitly in the result handler (i.e., first_name.text = result[0].first_name). Once again, Databinding makes this process cleaner. Click around on different employee records and you'll see the contents of the form change. Also notice that each time an employee record is selected the mouse cursor changes to an animated clock. This is done with the simple parameter showBusyCursor="true" on the WebService object.

The last feature we'll look at is how employee updates are handled. If you have an employee selected you can modify his first name, last name, or e-mail and click update. You'll see the update take effect instantly. When the update button is pressed a new employee object is created and then passed to the CFC, which in turn updates the database. Instead of retrieving the employee list again, we immediately update the selected entry in the DataGrid. This saves a roundtrip to the server, but leaves you with some work to do for a production environment. I mention this because we don't check that the update was successful before updating the DataGrid.

Conclusion

It's worth noting that with the small amount of code in Listing 1 we added a good deal of functionality and a very rich interface. It's hard to do this in any other technology with as little work, but the ease of use and rapid development comes at a hefty price tag. I can't say that Flex will be the answer to all your RIA woes, but it's definitely worth considering if you're involved in enterprise development. Before you balk at the price tag I recommend downloading the 60-day eval and giving it a run. Macromedia has also announced an Eclipse plug-in, code named Zorn, that enables Flex application development in one of the industry leading IDEs. When it comes to your next project, or if you're looking for a change, consider Flex. ☺

Dennis is currently a software engineer for SensorLogic, an M2M application service provider. You can read more about Flex as well as mobile software development on his blog at www.db75.com. dennisbaldwin@gmail.com

Interactive Ambience

ambient Interactive is an emerging Boston-based interactive agency specializing in innovative interactive design, online advertising, guerilla marketing, and brand identity. The company is made up of a small group of unique designers, artists, and creative thinkers who all bring a certain style and edge with them. Founded in early 2005, the first project the team took on was the design of ambientinteractive.com. Since its launch in April 2005 the site has been acknowledged with numerous interactive design awards and has been featured in online and print publications around the globe. The company is in its infancy stages and is still looking for that initial client with a breakthrough project opportunity. ☁



New Pre-Conference Classes
 Beyond Basic SQL for CF - Nate Nelson
 Intro to .NET for CFers - Geoff Snowman
 Designing & Developing OO Applications w/ CFCs - Hal Helms
 Maximizing Code ReUse w/ CFCs & Custom Tags - Simon Horwith
 "Forms and Beans" applying simple OO principles to standard CF forms methodology - Joe Rinehart
 Leveraging .Net Strengths from CFML - Charlie Arehart
 Advanced Cascading Stylesheets - Sandra Clark
 FLIP and Fusebox walkthrough - Jeff Peters

CFML Platform Integration
 Geoff Snowman
Deployment
 Tim Buntel
 Simon Horwith
 Charles Arehart
 Christian Cantrell
 Jeff Peters
Manager Empowered Programming

coldfusion

Raymond Camden
 Michael Smith

Steve Drucker
 Ben Forta

Advanced CF

Sean Corfield

MX Integration

Hal Helms

Dave Watts

Shlomy Gantz

Accessibility / Usability

Boot Camp



June 29 - July 1, 2005
 Washington DC Area
 7th Annual ColdFusion Conference
 3 full days!
 New Venue
 75% Bigger



www.cfunitied.com

Produced by
 TeraTech
 Programming
 www.teratech.com
 301.424.3903
 info@cfunitied.com

- macromedia
- TeraTech
- Microsoft
- NewAtlanta
- CFDynamics
- Fog Creek
- Savvy
- Interakt
- UNIVERSAL Mind
- Paper Thin
- MX
- COLD FUSION Journal
- HostMySite.com



CFUN has become the premier CF specific event, and Michael Smith and his team deserve all sorts of praise for their hard work in pulling it all off yet again.

Ben Forta

"...this event really is the best gathering in the world for people developing or managing CF systems. It's here that we can understand what happened, hear what's happening, and learn what's going to happen. You can't beat it."

Chuck Hoffman

"Great place to network yourself and pick up new techniques and ideas. Also to meet ones peers, and see what the future holds for all those involved with ColdFusion"

Daniel Gregorio

"Introductions to the latest developing techniques. Get inspiration in new ways to develop projects. Ge'erally, to "re-ignite" your cf "fire" by being part of a group excited by and interested in cf development."

Kathleen Ballard



We figured it was about time that web video stopped looking like web video.

This is a scene from one of the world's best websites, with video playing next to traditional web animation. The look of full-screen video, without ugly web players or pop-ups.

To tour the site—and see the future of video—visit:
macromedia.com/go/video5