SYS-CON

CS3: CREATIVE INTEGRATION - STRIKING A BALANCE

FORMERLY MX DEVELOPER'S JOURNAL

Web Developer's & Designer's Journal

WWW.WEBDDJ.COM

VOLUME 5 ISSUE 2 **2007**

BUILDING ENGAGING APPLICATIONS AND CONTENT WITH ADOBE TECHNOLOGIES

Mixing Flash and After Effects

Which is video and which is a Flash object

Plus:

- Presentations for the New Web
- The Rise of RIAs
- What to Unlearn

Presorted
Standard
US Postage
PAID
St. Croix Press


puma
INTERSPEC TACULAR

Adobe® Creative Suite® 2.3

Say hello to the next generation.

It's found in the world's most inspirational places. It's pushing the boundaries of art direction and design. Introducing Adobe® Creative Suite® 2.3. Today's most talented designers and art directors are working smarter and faster because of it. Just ask the masterminds behind INTERspectacular—they rely on the Creative Suite to help bring their ideas into the world. See how they do it at adobe.com/creativemind. It's everything but the idea. Better by Adobe.™

Luis Blanco and Michael Uman,
INTERspectacular

adobe.com/creativemind

Adobe

Adobe | Solution PARTNER

INTERGRAL information solutions

</cf_bugs>

FusionDebug™ - The Interactive CF Debugger

Faster. Easier. Smarter.

fusion debug™ 2.0

www.fusiondebug.com

Announcing FusionDebug™ 2.0

- installers for Windows, MacOS, Linux...
- improved user interface layout
- easy to use configuration tool
- improved support for CF frameworks
- step though CF pages, tags and CFC's
- set breakpoints
- view and modify variables
- watch expressions

Your Web site ...

Your Web site Powered by Hot Banana

Hot Banana Web Sites Are Built Fully-Loaded!

Why settle for building and managing a vanilla Web site when you can savor a full featured Hot Banana Web site?

Hot Banana is a fully-loaded Web Content Management System with all the Web Site Optimization and Marketing Automation tools your Marketing department craves. And, it's easy-to-use, search engine friendly and fine tuned for generating and converting the maximum number of leads.

So what's the cherry on top? It's ColdFusion of course - you'll love how easy Hot Banana is to implement, integrate and customize with a robust, yet semi-open API.

Schedule your Free Taste Trial Today!

Features...

- 100% browser-based
- Updated RTE
- Powerful DAM
- XHTML & CSS compliant
- Multilingual support
- Granular security
- Flexible workflow
- Press release manager
- Keyword analysis
- Web analytics center
- A/B Testing
- Web Forms
- RSS & blogs
- SEO tools
- Content reuse & scheduling
- Lead source tagging & tracking
- Custom metadata
- Event registration
- Email manager

Contact Us Now:

hotbanana.com/cfdj-demo
1.866.296.1803
sales@hotbanana.com

hotbanana®
Web Content Management For Marketing

Web Developer's & Designer's Journal

Group Publisher Roger Strukhoff
Art Director Louis F. Cuffari

Editorial Board
Aral Balkan
Erik Bianchi
Craig Goodman
Andrew Powell
Jim Phelan
Andrew Phelps
Darron J. Schall
Stephanie Sullivan
Jeff Tapper
Jesse Randall Warden

Editorial
Editor
Nancy Valentine, 201 802-3044
nancy@sys-con.com

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
577 Chestnut Ridge Rd.
Woodcliff Lake, NJ 07677

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
Megan Mussa, 201 802-3024
megan@sys-con.com

Web Developer's & Designer's Journal (ISSN#1546-2242) is published monthly (12 times a year) by SYS-CON Publications, Inc., 577 Chestnut Ridge Road, Woodcliff Lake, NJ 07677.

Creative Integration

by Roger Strukhoff

The increasing integration of Adobe and former Macromedia products continues with Creative Suite 3, which is available in a version targeted specifically to Web developers and designers. Designers today – whether they started in an era of X-acto knives and border tape or have recently graduated with a degree in digital design – must have at least a nodding acquaintance with the various strains of code that underlies their work. Yet the tools have long been powerful enough that they don't have to consider themselves to be programmers.

Striking the balance between sheer ignorance of coding details on the one hand and programming mastery on the other is the essence of creating professional software. CS3 is not a toy. It doesn't necessarily shield its users from what is going on underneath. It's designed for the most sophisticated pros, yet is accessible enough for even non-designers who wish to learn the rudiments of the craft.

Does CS3 create better work, though? Answering that question properly requires taking the really long view. I mean really long.

The discovery of perspective by Renaissance European artists has long been recognized as one of the turning points in the history of creative arts. No more were artists constrained by the flat, non-realistic two-dimensional images of people and places found in earlier Medieval works. Yet was this necessarily an improvement?

As many critics and historians have pointed out, the integration of perspective into painting meant that most work was now a mere snapshot, an image frozen in time. By contrast, the 2D viewpoint showed motion and time. Works as diverse as the Bayeux Tapestry in the Middle Ages, and the art of the ancient Mayan and Egyptian civilizations, show societies in motion. Looking at an ancient two-dimensional work brings to life the everyday hustle and bustle of society in a way that a mere 3D snapshot cannot. It wasn't until Picasso began experimenting with multiple perspectives in the 20th century that the idea of kinetic energy expressed in art was fully reborn.

So the question of whether something as modern and powerful as CS3 creates better work can only be answered by the work that is, in fact, created. Powerful tools in inept hands do nothing. The ability to "Photoshop" something quickly, for example, is not worth much if the user has no eye.

For those who have the eye – a group that presumably includes all the readers of this publication – the idea of returning to less powerful tools, less tight integrations, slower systems, and longer processing times, etc., is something not to be desired. Adobe continues to do its part to push the envelope for its audience of creative professionals. For creative professionals, the time-frames required to create and re-create continue to get smaller. This is something that should never be underrated in the 24/7 Worldwide Web era. The ability to design superior work and get it posted quickly is of premium value to the individual designer, the small design firm, the medium-sized company, and the global enterprise.

But in the end, it's always all about the work. I would expect the work to be created with these tools to be nothing short of tremendous.

Roger Strukhoff is Group Publisher and Editorial Director of SYS-CON Media. He spent 15 years with Miller Freeman Publications and The International Data Group (IDG), then co-founded CoverOne Media, a custom publishing agency that he sold in 2004. His work has won awards from the American Business Media, Western Press Association, Illinois Press Association, and the Magazine Publishers Association. You can read his blog at "rssblog.linux.SYS-CON.com" and contact him at roger@sys-con.com.

Meet Robert

A business executive at a popular social media site

Challenges

- Encountering poor video quality due to exponential growth in traffic.
- Increasing bandwidth costs due to growing audience size.
- Experiencing compatibility issues due to user-generated video arriving in multiple formats.

Solutions

- VitalStream Streaming Services – Improved quality of end-user video experience using a scalable and global content delivery network.
- VitalStream Advertising Services – Transformed the delivery of digital media from a cost center into a profit center.
- VitalStream Transcoding Services – Automatically converted all user-generated content into the leading streaming media format.

Providing End-to-End Solutions for Your Business

VitalStream is more than a rich media delivery company. We are your professional partner providing solutions that meet your unique business challenges. To learn more about VitalStream solutions, call 800-254-7554 or visit www.vitalstream.com/go/solutions/

VitalStream®

Digital Media Solutions for Your Business

Aral Balkan

Aral Balkan is founder and managing director of Ariaware, a London-based company offering products like Ariaware Optimizer and the open-source Ariaware RIA Platform (ARP 2.0) for Flash developers. Ariaware also offers RIA development process and usability consulting and development services. Aral holds an MA in Film and Electronic Media, is a Macromedia Certified Instructor and is celebrating his 20th year as a programmer (he's only 27!). His passions include software architecture and Human-Computer Interaction - in other words, building solid, usable applications. He's co-author of "Flash MX Most Wanted Components" and "Flash 3D Cheats Most Wanted," as well as author and editor of numerous articles for Adobe Developer Center and Ultrashock.com.

Erik Bianchi

Erik Bianchi is a software engineer with more than five years of experience developing Flash-based RIAs and enterprise-wide applications for Fortune 50 and 500 companies. In his spare time he enjoys building Flash-based games, writing or tech editing Flash-related books, and when burned out on code, playing video games on his PC/console systems. You can get more info about Erik and his latest projects on his blog at www.erikbianchi.com.

Craig Goodman

Craig Goodman is the executive editor of Adobe's Developer Center. He and his team publish the tutorials and articles in the area. Craig joined Macromedia in 1995 and his past roles include managing web support and supervising product technical support for Macromedia Flash.

Jim Phelan

Jim Phelan is vice president of development for Stream57, a New York City based firm specializing in communication solution development for the enterprise. Jim's expertise in creating solutions for consolidation and collateralization of business communications has allowed his team to create applications for the management and delivery of live and on demand rich media content. Jim is a strong proponent of the Adobe Flash Platform and is a member of the editorial board of MX Developer's Journal.

Andrew Phelps

Andrew M. Phelps is in the Information Technology Department at the Rochester Institute of Technology in Rochester, NY (http://andysgi.rit.edu/).

Darron J. Schall

Darron J. Schall has been programming long before he could drive. In school he studied programming languages, ranging from Basic to Pascal to C++ and eventually moving into Java and C# throughout college. Somewhere in the middle he got hooked on Flash 5 and it's been a crazy love affair ever since. Darron is an independent consultant specializing in RIA development. He maintains a Flash Platform related weblog (www.darronschall.com) and is an active voice in the Flash and Flex communities.

Stephanie Sullivan

Stephanie Sullivan is a Web developer, partner at CommunityMX (www.communitymx.com), owner of VioletSky Design (www.violetsky.net), and contributing author of Dreamweaver MX 2004 Magic.

Jeff Tapper

Jeff Tapper, co-founder of Tapper, Nimer and Associates, has been developing Internet-based applications since 1995, for a myriad of clients including Toys R Us, IBM, Allaire, Dow Jones, American Express, M&T Bank, Verizon, Allied Office Supplies, and many others. As an Instructor, he is certified to teach all of Adobe's courses on Flex, ColdFusion and Flash development. He has worked as author and technical editor for several books on technologies including Flex, Flash and ColdFusion, such as "Object Oriented Programming with ActionScript 2.0", and "Flex 2 Training from the Source."

Jesse Randall Warden

Jesse R. Warden is a senior Flash developer at Surgical Information Systems, an operating room software company, where he currently uses Flash MX, Flash Remoting, .NET, and Oracle to create next-generation rich Internet applications for the OR. He contributed four chapters to the Flash Communication Server MX Bible and has written articles for various publications, including one for Macromedia for a DRK.

cfunited

the premier coldfusion technical conference

Don't miss out on the Premier ColdFusion conference, CFUNITED-07 where you can learn from over 50 expert speakers and authors such as Ben Forta, Michael Smith, Simon Horwith, Hal Helms, Ray Camden, Charlie Arehart. Four days of sessions, networking and learning with your programming peers from around the world. Guarantee yourself a seat and register today.

"CFUNITED is a great way to not only network with fellow ColdFusion afficianados, but also to learn new techniques and methods to help your career or business. You also can get inside scooops on developments in ColdFusion and learn best practices and much more."
-Angela T , Attendee

"It is run by developers for developers. The sessions are based on real world experience and real user case studies."
-Rich P , Attendee

"CFUNITED is a great opportunity to launch your career into the next level, allowing developers to learn new tips, tricks and techniques, all while bonding with fellow ColdFusion users many of us only know by name."
-Constanty D , Attendee

"Learning more, networking more, getting NEW ideas...and now, to support CF's continued existence."
-Howard P, Attendee

TeraTech Inc.
405 East Gude Dr, Ste 207
Rockville, MD 20850
301.424.3903

www.cfunited.com

Presentations for the New Web

Flash presentations for everybody

by Alexander Kouznetsov

Most of us at one time or another has had to give a presentation – for a project, a company, or something else; we have all used PowerPoint (or other desktop applications such as OpenOffice and KeyNote) in these presentations.

We are all familiar with PowerPoint presentations, maybe even "too familiar." We've seen too many PowerPoint presentations, but only few great presenters. It's not a PowerPoint problem though as it's just a tool, and it dominates desktop presentations for a reason.

It's fine for what it was originally designed for: "Creating on the desktop – presenting on the desktop."

But what about presenting on the Web? Web publishing features were added to PowerPoint about 10 years ago, and in a Web 1.0 way (someone may say "old dog – new tricks").

An alternative way would be to "Create on the Web – present on the Web"? That's the Web 2.0 way, or "New Web" way.

There are Zoho, ThinkFree, and few more Web applications in this space. These applications do a good job imitating the desktop applications on the Web. It seems like a good idea – creating a Web clone of PowerPoint. Users would know what to expect – so it would be easy for them to adopt it.

But this approach could be both good and bad. Some design decisions in PowerPoint are good; some, questionable; and some, not portable to the Web. Besides, unlike on the desktop, PowerPoint is not a de-facto standard for a Web presentation.

And there's a reason for that too.

Let's Dig from the Other Side

How about trying a new approach – by designing from the Web side? Let's start from scratch on the Web and create a Web-native application, pretending we've never seen PowerPoint before. Maybe we'll invent something interesting and useful.

So we created Spresent – A Presentation Tool for the New Web.

Desktop vs. a Web Application

Most likely the Web platform will not replace the desktop, but would complement it. However, the Web looks more attractive to developers at this time and that's where the innovation happens. Table 1 provides are some of the key differences in presentation graphics application development.

We started Spresent back in 2001 as a Web application, then we ported it to the desktop, and then back to the Web. Soon we'll release a desktop version again. The desktop is a secondary priority platform for us, but it looks like we can't escape it – yet.

This pattern of "going back and forth" would probably apply to some other Office 2.0 apps; users would need both the desktop and the Web version.

As we know now, not every application can be originated on the desktop. For example, blogs, wikis, and some other wonderful Web 2.0 apps were not ported from the desktop.

The Web application market is still young and it's everyone's game.

Office 2.0 and the Enterprise

Recently the enterprise market started adopting Web 2.0 applications. That was inevitable and it would accelerate Web innovation even further over the desktop. In the early Web days an enterprise wouldn't trust anybody to host confidential data. Not anymore.

"What kind of app is missing, from the New Web landscape?" asked Jeremy Geelan in his blog. We think the answer is a Web presentation tool.

It should be a rich Internet application presenting rich Internet content – multimedia and dynamic content.

Content Publishing – Old Formats

"Content is king" as we know, and Web content has evolved fast. Web publishing became very easy with blogs and photo-sharing sites. But the Web content format didn't change much; it's still the same – HTML and some images – as it was 10 years ago.

If you have a story to tell on the Web, it would still look about the same is it did 10 years ago. That's because HTML is the native format of the browser. It's not going to change any time soon, though now HTML is under the strong influence of XML and CSS.

Content Publishing – New Formats

Then along came YouTube and podcasting with new, rich content and the publishing path "Create on the desktop – publish to the Web."

To view rich content, a browser needs a media player. Thanks to Flash, we all have media players pre-installed in our browsers, even on mobile devices. Flash is the ultimate media player on the Web, and it probably won't change any time soon.

Flash Platform Phenomena

As a runtime environment, Flash has won the Web; it's in the process of winning the mobile platform and is about to penetrate the desktop.

Here are some of the key benefits to using Flash as an engine for a presentation application:

- Free, robust, vector-based graphics engine (great quality, scalable for any screen)
- Open file format (SWF)
- Support for rich media formats: animations, images, video, and audio
- Ninety-eight percent computer market penetration; cross-platform
- Built-in solid scripting engine for developing RIAs
- Mobile and TV ready: 100M+ devices on the market and growing fast
- No other graphics engine alternative to Flash anytime soon
- Great development environment

It does most of what we needed to create a new Web Presentation application – Spresent.

Web Presentation Application Requirements

We started with the basic requirements:

- Create a feature-rich and easy-to-use presentation application for the Web.
- No installation required – any computer/platform could be used.
- Make it "eye candy." Presentations should look good.
- Use the best graphics engine on the Web – Flash.
- Support multimedia files (photo, audio, video, animations).
- Target Web 2.0 users: blogs, wiki, and podcasts.
- Host and grow an extensive content library (templates, clip art).
- Connect to other popular services – Flickr, YouTube, etc.
- Automatically give credit to the sources used in the presentation.
- Keep the application small, so it will load fast and run on older computers and devices.
- Publish off-line presentations – export, print.
- Release the desktop version. Notebooks outsell desktops.
- Support "dynamic" content. The presentation content can be "generated" dynamically.
- Frequent improvements and updates (seamless to the user).
- Make it cool and free. Offer subscriptions for premium services.
- Get ready for mobile and TV presentations.

Table 1

	Desktop	Web
OS Platform	One or two	Cross-platform
Deployment	"Software box" or Download	Web Server
Deployment costs	High	Low
Application Updates	2-3 years	Weekly/Monthly
Content/Templates	Packaged, limited updates	Web-based, instant updates
User Content Sharing	Very limited	Yes
File Formats	Proprietary	Open
Presentation Objects	Text, Images, Clip arts	Rich Web Media
Revenue Model	Software update sales	Free (ads), Subscription, SaaS

SYS-CON Media

CEO
Fuat Kircaali, 201 802-3001
fuat@sys-con.com

President & COO
Carmen Gonzalez, 201 802-3021
carmen@sys-con.com

Sr. Vice-President, Editorial & Events
Jeremy Geelan, 201 802-3051
jeremy@sys-con.com

Advertising
Advertising Sales Director
Megan Mussa, 201 802-3023
megan@sys-con.com

Associate Sales Managers
Corinna Melcon, 201 802-3026
corinna@sys-con.com

Events Manager
Lauren Orsi, 201 802-3022
lauren@sys-con.com

Events Associate
Sharmonique Shade, 201 802-3024
sharmonique@sys-con.com

Production
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 Directors
Abraham Addo, 201 802-3037
abraham@sys-con.com
Tami Beatty, 201 802-3038
tami@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
Richard Walter, 201 802-3042
richard@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

Customer Relations
Circulation Service Coordinators
Edna Earle Russell
edna@sys-con.com

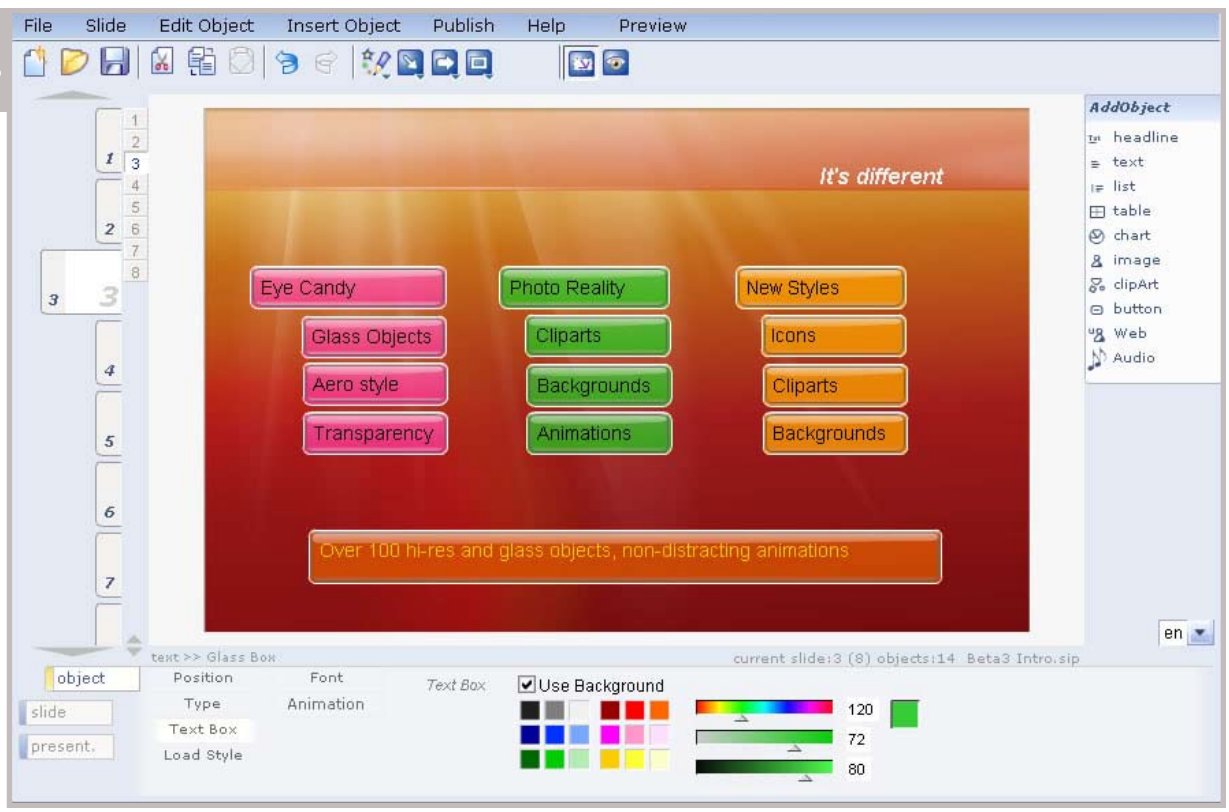
Alicia Nolan, 201 802 -3081
alicia@sys-con.com

Challenges

Using Flash as a platform for application development has few challenges. Here are some of the issues experienced with Spresent and the choices we made:

- ***Flash version:*** From the development perspective, it's always good to use the latest version of Flash, but it wouldn't be the best choice from a user perspective. So we agreed to stay "two versions behind." This gave us about 90% market penetration for the target Flash version every time. Also it's good to keep older versions available as it helps with mobile deployments.
- ***Components:*** Using component libraries and layout managers is attractive for developers, but it would add more Kb to the size of the application. We agreed not to use standard components and instead developed our lightweight components when we needed them, to keep the application small. We ended up with a complete application under 200Kb (not including Clip Art libraries), so it loads fast on any connection.
- ***Application interface:*** We were experimenting with designing an interface in Flash – how to make it intuitive, functional and resolution-independent. What should be the aspect ratio of slides for a cross-platform Web application, etc. It took three iterations, but we got great reviews after launching spresent.com.
- ***Desktop version:*** Porting a Flash application from the Web to the desktop is not a big problem, which is another wonderful thing about Flash. We had to write a C++ wrapper for the Flash movie to handle FS commands, add some libraries, and create an installer. We released a Windows-only version on the desktop and that seems to be okay with the users. The desktop version looks the same as the Web version, except for file dialogs. Originally we didn't plan to release a desktop version, but notebook computers are not always connected to the Web, and notebooks outsell desktop computers now.
- ***Import:*** Users expect a Web application to open documents created on the desktop. So we had to write a PPT import on the desktop first. It's just a basic import feature, helping casual users of PowerPoint to use Spresent. Our application design and objects are different than PowerPoint, so it's impossible to import and have a Web presentation that looks the same on the desktop. And we didn't want to. Since we use Flash, the Web presentations should look better.
- ***Publishing:*** This is a key feature of any Web application – it should be capable of publishing the results in different formats – for Web and for desktop applications. Spresent publishes final presentations as e-Mail, EMBED, HTML, ZIP and Flash (SWF-4 and SWF-6).
- ***SWF Export:*** Compiling a single SWF file from a presentation was tricky. We kept clip art as separate movie files (SWF). To compile the single SWF file for presentation, we had to take every single SWF object used in the presentation, "decompile" the SWF file, retrieve all internal variables and movie parameters, and then add this object to the new SWF of the presentation. When we started, we weren't sure it would be possible. But we did it twice – for Flash 4 and Flash 6 formats (SWF-4, SWF-6).
- ***Printing:*** If a presentation has animations, how do you print it? Starting with Flash 7, printing became easier.

Figure 1



JavaOne℠

Java™

OPEN POSSIBILITIES

May 8–11, 2007

The Moscone Center, San Francisco, CA
JavaOne Pavilion: May 8–10, 2007

java.sun.com/javaone

> JAVA™ TECHNOLOGY IS NOW OPEN–AND SO ARE THE POSSIBILITIES

The 2007 JavaOne℠ conference has expanded and is definitely one conference you won't want to miss. With the decision to open source Java™ technology, 2007 marks a major milestone for the Java platform. Whether your passion is scripting languages, open source, SOA, Web 2.0, mashups, or the core Java platform, this is a conference that has something for almost all technology developers.

LEARN MORE ABOUT*:

- > Scripting (JavaScript™ Programming Language, PHP, Ruby on Rails, Python, and More)
- > Open Source and Community Development
- > Integration and Service-Oriented Development
- > Web 2.0 Development
- > AJAX
- > Java Technology and the Core Java Platforms (EE/SE/ME)
- > Compatibility and Interoperability
- > Business Management

SAVE $100**

Register Today!

Please use priority code: J7PASC

* Content subject to change.
** Offer not available on-site.

Attend the JavaOne conference, and you will have many opportunities over the course of four days to network with like-minded developers; attend in-depth technical sessions; engage with your peers in Hands-on Labs and BOFs; and experience general sessions featuring speakers from Intel Corporation, Motorola, Inc., Oracle Corporation, and Sun Microsystems, Inc. Meet face-to-face with leading technology companies, and test-drive the latest tools and technologies shaping the industry.

PLATINUM COSPONSORS

intel MOTOROLA ORACLE

GOLD COSPONSORS

AMD Smarter Choice bea Think liquid. IBM NAVTEQ NOKIA SAP

SILVER COSPONSORS

InterSystems Sprint Together with NEXTEL PARASOFT We make software work. TERRACOTTA

Sun

We had to play some tricks with animated objects, but overall it came out great. Users can print the complete presentation from any computer with a single click, and usually print drivers offer handout options, such as two slides per page. Making a PDF was free with the printing feature – just print the presentation into a PDF file using the local PDF driver on the computer.

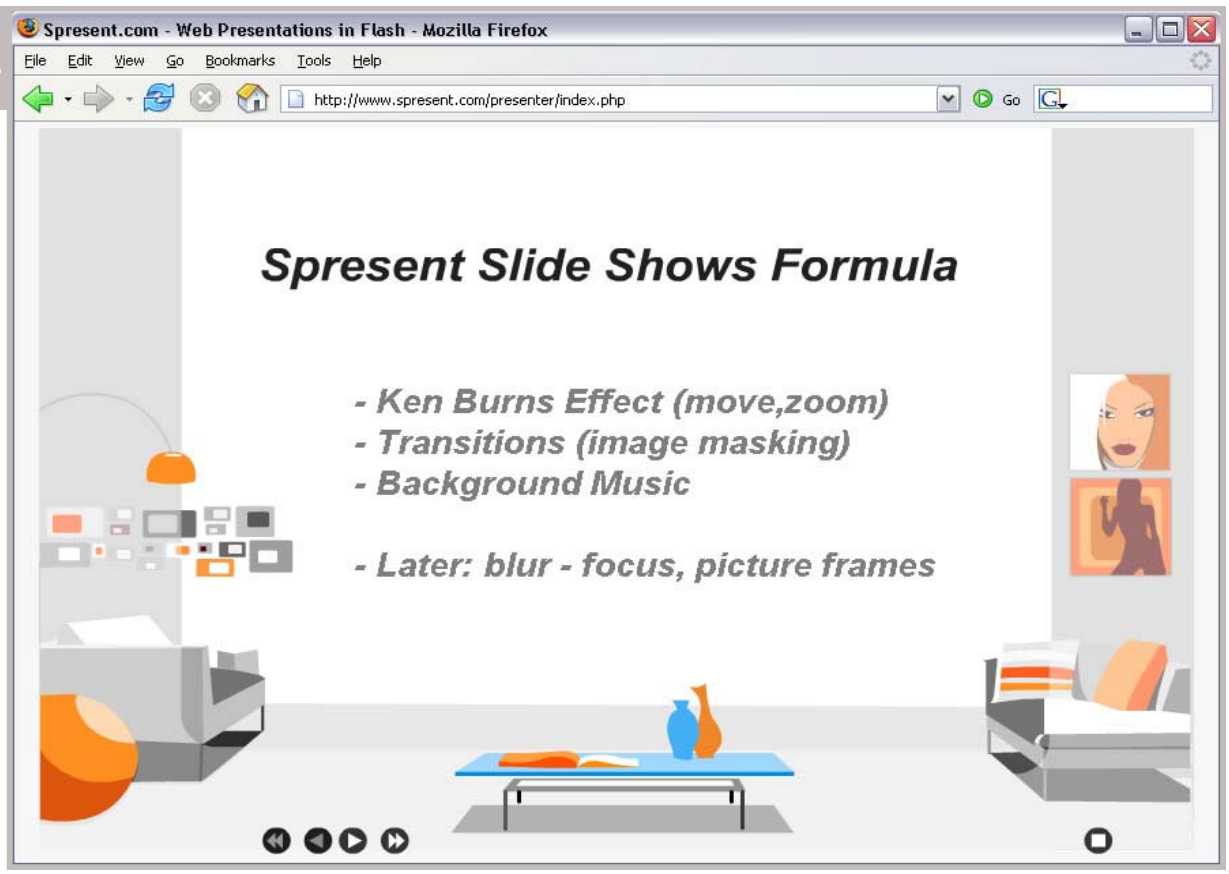
- ***To animate or not:*** Flash can animate almost anything in a presentation, but it would be a distraction. In Spresent the designers have strict rules – animations should not distract from the presentation content. On other hand, we added interaction and animations options to some traditional objects: bullets, tables, backgrounds, and charts. For example, the San Francisco background has seven non-distracting animations at the bottom of the screen.
- ***i18n:*** The Web application has to be internationalized early in development. Spresent creates presentations in any Unicode-compliant font, and the application interface comes in five languages. It's just one XML file per language, so adding new languages is easy.
- ***Auto play and audio:*** Web presentations are designed to be published to the Web and should be able to change slides on click or auto play. Spresent has an original, easy-to-use time line for arranging slide timers. We also added audio support; it really makes a difference in presentations. For slideshows and company presentations, asynchronous background music works great. For podcasts, product, and training presentations, synchronized voice (MP3) is the right choice.

Presentations for New Web Users

How can Web presentations help – for personal and business use? Here are some examples of how Spresent fits into the Web 2.0 space.

- ***Blogs:*** There are 50+M blogs. Bloggers use text and some images, which is very simple and needs to be improved. Web browsers support rich content formats (like Flash), but most bloggers won't buy and learn Flash authoring tools to create and publish blogs. Spresent offers them a tool to create "Rich Internet Blogs" for better storytelling.
- ***Wikis:*** It's the easiest way to collaborate; anybody can update the content of a Web site. Anybody can overwrite anybody, but it usually works great. Since a Wiki is a Web site, it's limited to what browsers do the most – text and images. Spresent offers tools to create animated, rich-media articles for Wikis.
- ***Podcasts:*** The number of podcasts grows every day with the number of listeners around 20M+. Podcasts are blogs with multimedia attachments. Today the podcast Web sites (blogs) offer two media file options: MP3 file (audio podcast) or MPEG-4 file (video podcasts). Spresent offers another option – a presentation synchronized with an MP3 file (for example, a slideshow with a podcast). This is similar to enhanced podcasts on iPod video, but in the browser.
- ***Photo sharing:*** Services such as Flickr. Spresent implemented a connector for Flickr, to search and add photos to the presentations. Users can create advanced presentations or just simple slideshows from shared photos.
- ***Video sharing:*** Services such as YouTube. Spresent supports YouTube videos on slides in presentations. Users can mix videos with photos, text, clip art, etc.

Figure 2



STREAM57

Customized Flash-based media solutions

Software and services for collaboration, video conferencing, and e-learning

visit us
stream57.com

e-mail us
streamline@stream57.com

call us
212.909.2550 x1012

- ***Audio sharing:*** Services such as PodShow+. Spresent supports an external MP3 file playback, in sync or async mode.
- ***Social Web:*** Services such as MySpace and FaceBook. Spresent can be used for presentations such as slideshows or advanced content aggregation.
- ***Education:*** Schools, colleges, and universities use presentation applications to teach. Web presentation publishing can help.
- ***Small business:*** Create a nice Flash presentation about a company or product and update it any time. Post it on the Web server, blog, or in a forum on another Web site. No contractors required.
- ***Enterprise:*** Business presentations, corporate training, Web presentations (time-shift economy), send engaging presentation to customers, and more...

The list above seems to be broad, but that's how presentations are used.

Wish List for Flash

Flash is great, but not perfect. Here are few things we wish Flash had:

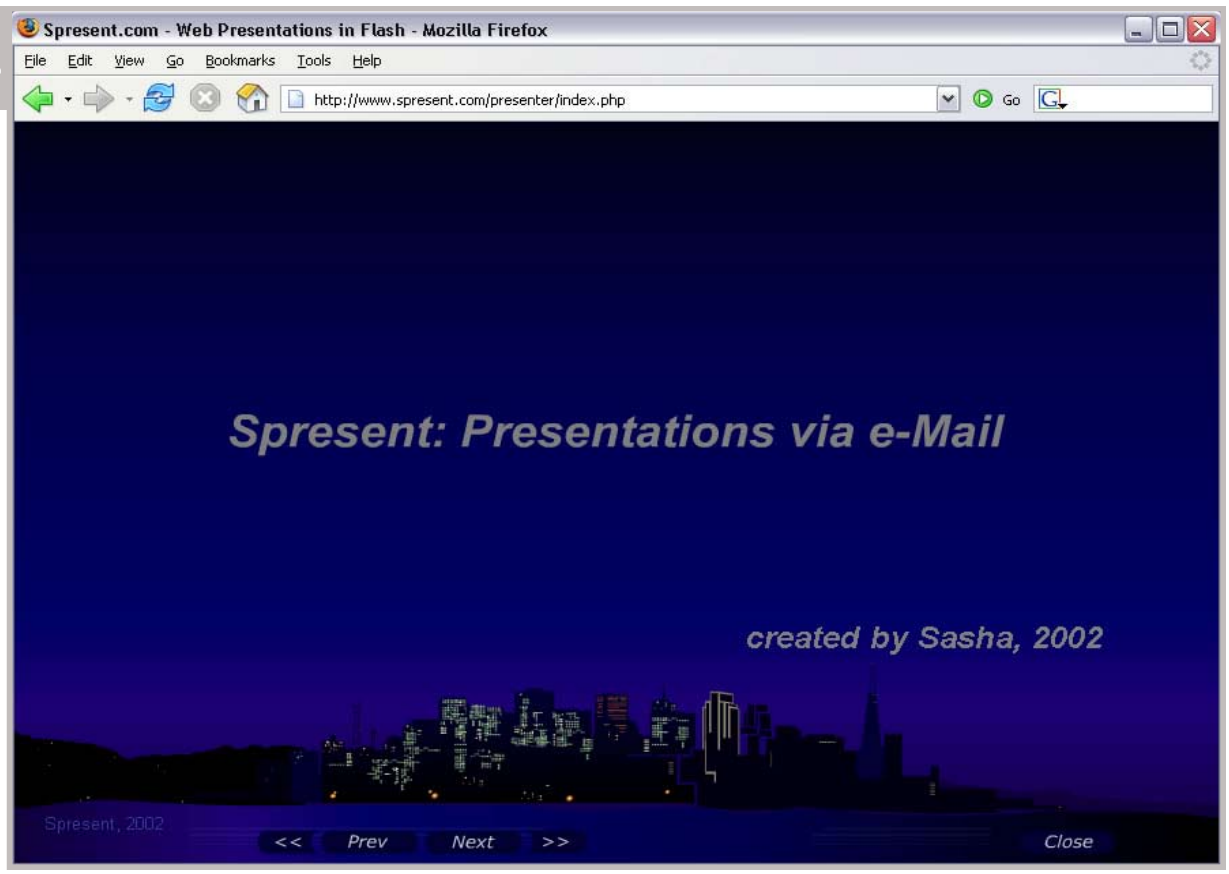
- Built-in content protection (encryption) options in Flash Player. There are third-party tools to encrypt the SWF files, but it would be better to have it in the Flash player.
- FlashLite devices are currently 2/5 versions behind the Web. Converting all platforms to Flash 9 would be great.

Free Way

Flash and Web presentation tools have come a long way; however, it feels like it's just the beginning as more great things are coming. And it's all free – like Flash and Spresent.

Alexander Kouznetsov (Sasha), PhD, is founder and president of Spresent (www.spresent.com). He holds degrees in math and computer science.
sasha@spresent.com

Figure 3



> *Businesses that ignore the potential of SOA will find themselves outpaced by rivals who improve their agility and transform themselves into new kinds of enterprises*
>
> — *Yafim Natis, Gartner Analyst*

3-DAY EVENT!

SOAWorld 2007

Plus

Enterprise OpenSource Conference & Expo 2007

TOPICS INCLUDE:

SOA Web Services

- AJAX and SOA
- Web 2.0
- Universal SOA
- Protecting Web Services
- Troubleshooting SOA
- Governance
- Open-Source SOA
- XBRL
- Service Virtualization

Open Source

- Open Source Business Models
- Open Source ESB
- OpenAjax Alliance
- SaaS and Open Source
- Spring, Hibernate and Eclipse
- Seam
- Open Source Penetration
- Monetizing Open Source
- Open Source Databases
- AMQP
- Open Source Middleware

June 25-27, 2007
Roosevelt Hotel / New York City

Register Online! www.SOAWorld2007.com

SOAEOSCONFERENCE.SYS-CON.COM
REGISTER ONLINE TODAY
SAVE $200!
(HURRY FOR EARLY-BIRD DISCOUNT)

11th International SOAWORLD 2007 CONFERENCE & EXPO

2nd Annual ENTERPRISE OPENSOURCE 2007 CONFERENCE+EXPO

2007 is to many industry insiders shaping up to be a major inflection point in software development and deployment, with SOA, Web Services, Open Source, and AJAX all converging as cross-platform and cross-browser apps become the rule rather than the exception.

Accordingly the 11th International SOA Web Services Edge 2007 again seeks to offer comprehensive coverage and actionable insights to the developers, architects, IT managers, CXOs, analysts, VCs, and journalists who'll be assembling as delegates and VIP guests in The Roosevelt Hotel in downtown Manhattan, June 25-27, 2007

Co-located with the 2nd Annual Enterprise Open Source Conference & Expo, the event will deliver the #1 i-technology educational and networking opportunity of the year. These two conference programs between them will present a comprehensive view of all the development and management aspects of integrating a SOA strategy and an Open Source philosophy into your enterprise. Our organizing principle is that delegates will go away from the intense two-day program replete with why-to and how-to knowledge delivered first-hand by industry experts.

Visit soaeosconference.sys-con.com for the most up-to-the-minute information including... Keynotes, Sessions, Speakers, Sponsors, Exhibitors, Schedule, etc.

BROUGHT TO YOU BY:

SOA World Magazine
focuses on the business and technology of Service-Oriented Architectures and Web Services. It targets enterprise application development and management, in all its aspects.

Enterprise Open Source Magazine
EOS is the world's leading publication showcasing every aspect of profitable Open Source solutions in business and consumer contexts.

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

Exhibit and Sponsorship Info:
Call 201-802-3020 or email events@sys-con.com

Mixing Flash and After Effects

Which is video and which is a Flash object

by Tom Green and Tiago Dias

We are going to introduce you to an effect that's a lot of fun to use. Items on the Flash stage are rarely static. Things move around, menus pull down, and so on. The line between what is Flash and what is video on the Flash stage has, for all intents and purposes, been erased. It makes sense, therefore, that when planning a Flash/After Effects project, that the stuff that moves, in either app, should be indistinguishable as to its source.

In this exercise, we're going to explore this concept. There is going to be motion on the Flash stage, and there is going to be text in motion in After Effects. The Flash movie will be prepared in such a way that the user can't tell which is video and which is a Flash object.
Best of all, the resulting SWF file will be less than 5K in size, meaning it should play rather quickly.

Before you start, you'll have some fun and learn how to use the Wiggler.

Text has certain properties in the timeline. What the Wiggler does is to add a degree of randomness to those properties. What you can do with this feature is have text bend, move, blur, shake, spin, and so on by changing the parameters for the effect. Where the randomness enters the picture is in how the effect is applied.

The parameters are the outer limit. This means the effect can be applied to text, and it will change using any value up to the limit you set. For example, you can enter your name to a text layer and use the Wiggler to set the maximum distance between the letters in your name to 50% of the start position. When the movie plays, each letter in your name will move any distance from 1% to 50%. This means the first letter in your name may move only a short distance while the third letter moves to the top of the Comp. OK, let's have some fun:

1. Launch After Effects and create a new Comp named MyName that is 320x240 pixels in size, uses Square Pixels, has a frame rate of 30 fps, and has a duration of 0:00:10:00.
2. Set the background color to #FFFFFF (white).
3. Select the Text tool, click the Comp once, and enter your name. Use a font, size, and font color of your choosing.
4. Twirl down the Text layer on the timeline and click the Animate button on the layer strip. A pull-down menu will appear and show you all of the properties than can be animated. Select All Transform Properties. When you release the mouse, a new animator named Animator 1 will appear on the timeline.
5. On the Animator 1 strip is a button named Add. When you click this button, you will see a pull-down menu asking you to choose between a Property or a Selector. Click Selector and choose Wiggly as shown in Figure 1. When you make your selection, all of the properties that can be "wiggled" will be listed in the Wiggly Selector that appears on the timeline.

If you drag the playback head across the timeline, you're in for a bit of a disappointment. Nothing happens because all of the default properties for the Wiggly Selector are set to 0.

Let's fix that and figure out what you can do with this effect.

Click each of the Position values once and drag the mouse. The value on the left moves the letters on the *X* axis, and the value on the right moves them on the *Y* axis.

What you are seeing is the start position for the effect, and the values you have in the Position area are the maximum values the letters can move. Drag the playback head across the timeline, and the letters in your name will bounce around the Comp. Reset the Position values to 0.

Click-drag the Scale value. If you scrub across the timeline, the letters in your name will appear to pulsate. When you drag the mouse across the values, notice both values remained equal. This is due to the lock icon — it looks like a chain link — beside them. If you click the lock, you can change the scale values so they are independent of each other. When you've finished playing with this, click the lock and reset the values to 100%.

Click-drag the Skew value. When you scrub across the timeline, your name jiggles as though it were made of jelly. Reset the value to 0.

The Rotate and Skew Axis properties have two values: a number and a rotation degree value. The number determines how many rotations will occur, and the degree value determines how the text will rotate. Set the Rotation number value to 2 and the degree value to 45. If you scrub across the timeline, the letters in your name will, for want of a better description, appear to jitter. Reset the Rotation values to 0.

Now that you've played with the properties, let's take a look at the selections under the Wiggly Selector. Twirl it down, as shown in Figure 2, and you will see the following:

- ***Mode:*** This pull-down determines how the selector should be combined with the objects above it in the timeline. This one is rather complex, so we will leave it alone.
- ***Max Amount, Min Amount:*** These two options specify the range for the selector. For example, assume you change the Position property values to 40, 45. If you scrub the timeline, the letters in your name will bounce around the screen. Some letters will move up and down, while others move right and left. The Min Amount and Max Amount properties will move things down and to the right. The Min Amount will move them up and to the left and are the range for the movement.
- ***Based On:*** This pull-down applies the wiggle based on a variety of choices ranging from individual letters to text blocks.
- ***Wiggles/Second:*** High values speed things up. Low values slow them down.
- ***Correlation:*** In many respects, this control applies kerning to the letters in motion. A value of 0 applies different values to each letter, and a value of 100% applies the same amount to each letter. Smaller percentages will pull the letters closer to each other.
- ***Temporal Phase:*** The key word here is temporal. The value you set here changes the timing of the wiggles in the effect. The first number determines the number of revolutions per second that will be applied, and the second number is the degrees through which the selection will rotate.
- ***Spatial Phase:*** Functions in a manner similar to the temporal phase but moves each character. In many respects, it's similar to running a wave through the text.
- ***Lock Dimensions:*** Scales the wiggled selection's dimensions by the same value. This is useful when wiggling the Scale property and the values need to be uniform.
- ***Random Seed:*** Enter a value to change the starting time for the animation by that value. When the seed is left at 0, a default value is derived based on the layer index and stream path.

6. Now that you understand what the controls can do, apply these Wiggly Selector values to your name:

Tom Green is professor of Interactive Media at Humber College Institute of Technology & Advanced Learning in Toronto. Tiago Dias works as a video producer and Flash developer at a television and news production company based in London. They are co-authors of From After Effects to Flash: Poetry in Motion Graphics, published by friends of ED (www.friendsofed.com).

Figure 1

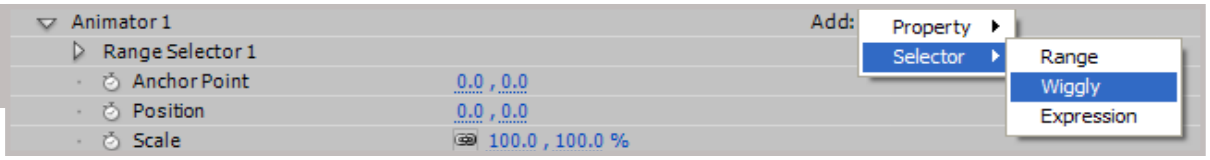
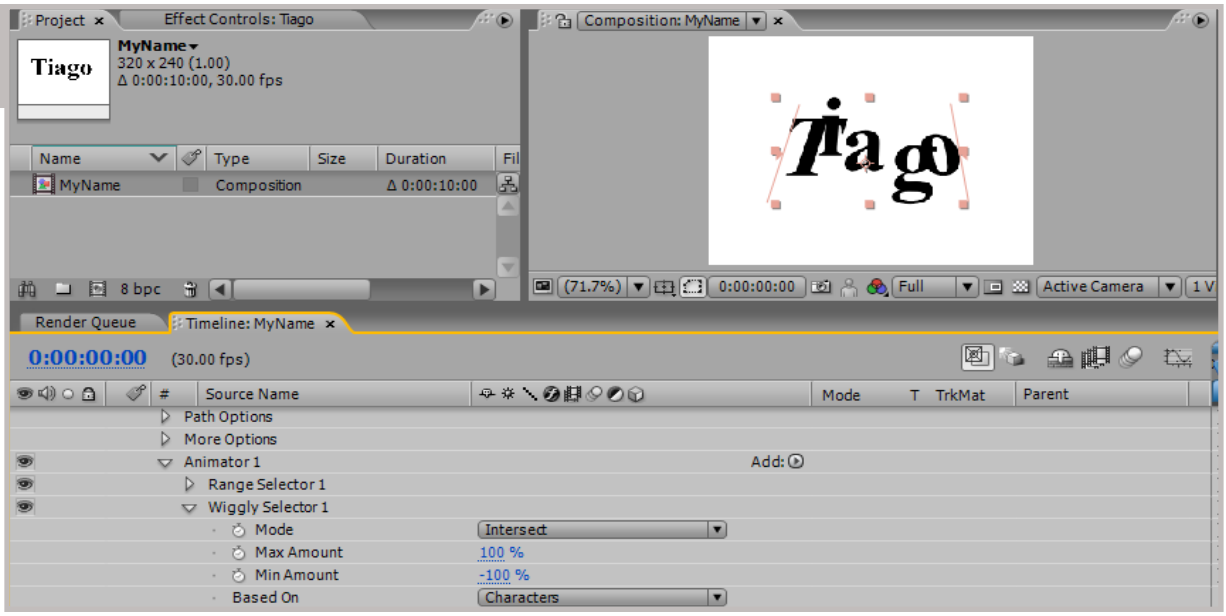


Figure 2



- ***Max Amount:*** 100%
- ***Min Amount:*** –100%
- ***Based On:*** Characters
- ***Wiggles/Second:*** 2.5
- ***Position:*** 45, 45
- ***Scale:*** 50, 50%
- ***Rotation:*** 0 x +30 degrees

If you click the Play button in the Time Controls panel or press the spacebar, the letters of your name start to dance with each other. If the viewer needs to figure out what the dancing letters mean, the letters will have to come back into their proper order. This is not difficult to do; remember, from the start of this exercise, nothing happens if the properties are reset to 0.

Let's turn this into a useful animation:

7. Add keyframes for each of the following properties at the 0 point on the timeline:
   - Max Amount
   - Min Amount
   - Position
   - Scale
   - Rotation

8. Move the playback head to the eight-second mark on the timeline and set the values for each of the properties in the previous step to 0.

If you press the spacebar, you will see the letters in your name dancing with each other, and the dance slows down as the playback head approaches the eight-second mark on the timeline. Now that you've had a chance to explore how the Wiggly Selector works, we are going to show you a practical example.

The final exercise here is rather complex, and instead of spending a lot of time reviewing a lot of what you have already done, we are going to review how the project was assembled in both Flash and After effects, and we will also review a couple of new After Effects tricks that you may want to add to your arsenal. You really don't need to do anything other than open the files. To get started, open the Wiggly.swf file located in the Wiggles folder found in the Exercise folder. (The source code can be downloaded from the online version of this article at http://webddj.sys-con.com.)

You'll notice there are four colored squares on the stage, and both the squares and the text around them are in constant motion. The most important aspect of this SWF file is the fact that an After Effects video overlies the entire Flash stage. The text that appears inside the moving blocks, shown in Figure 3, is prepared in After Effects.

The implication here is you need a high degree of precision in After Effects to ensure those letters appear directly over the blocks. If you open the wiggly.fla file in the Exercise folder, you'll see how this precision was achieved.

The Flash stage is divided into precise areas through the use of guides. If you scrub through the timeline, you will see the squares stay within the grid on the Flash stage (see Figure 4). The right side of the stage is left essentially blank and will be used for a variety of After Effects text effects.

We carefully noted the positions of each of the various grid lines. These positions will be used to construct a similar grid in After Effects

If you are lazy, like us, forget about writing down the position of the guides in Flash. The stage size used in After Effects will be the same as the Flash movie. Ensuring the Flash view is 100%, turn off the visibility of any of the objects on the Flash stage and take a screenshot of the stage and the guides. Import the screenshot into your After Effects document, and build the grid from the screenshot in an After Effects Comp.

This article is an excerpt from From After Effects to Flash: Poetry in Motion Graphics *by Tiago Dias and Tom Green, published by friends of ED (www.friendsofed.com) Reprinted with permission.*

Figure 3

Close the Flash document and let's examine how the project was assembled in After Effects.

Open the wiggly.aep file in your Exercise folder. When the project opens, you'll see that we have constructed a grid that matches the one from Flash. Here's how we did it:

1. We selected View->Show Rulers.
2. We zoomed out to a magnification level of 200% and, using the locations of the guides previously noted, dragged them from the horizontal and vertical rulers onto the Comp window. (If pixel precision were paramount, zooming out to an 800% view would set the rulers to single-pixel increments.)
3. To move around the Comp window, we held down the spacebar (if you are following along, you'll see a Grabber Hand icon replace the cursor) and dragged the mouse.
4. Once we had the grid constructed, as shown in Figure 3, we selected View->Lock Guides. By doing this, we prevented accidentally moving a grid line out of position. Note that you can also turn off the rulers by selecting View->Hide Rulers or by pressing Ctrl+R (PC) or Cmd+R (Mac).

Let's now look at how many of the effects used in the video were created.

1. Twirl down the Poetry layer and scrub the playback head across the strip in the timeline. It fades in and out and tends to jiggle. We accomplished this by physically moving the text block from one location to the next and fading it in by tweening its opacity. Twirl up the Poetry layer.
2. Twirl down the In layer. If you scrub across its strip, you'll see the blurred word zip in from the right and, as it bounces off of the guide, come into focus. We did this through the addition of a Tracking animator and a Directional Blur effect.

   The Tracking animator was applied by selecting Tracking from the Animate pull-down. This animator adjusts the spacing between the characters in a word. We tweened the tracking by changing the tracking value from 0 to 232 in the Tracking Amount property. This is how the word splits apart at the end of the animation.

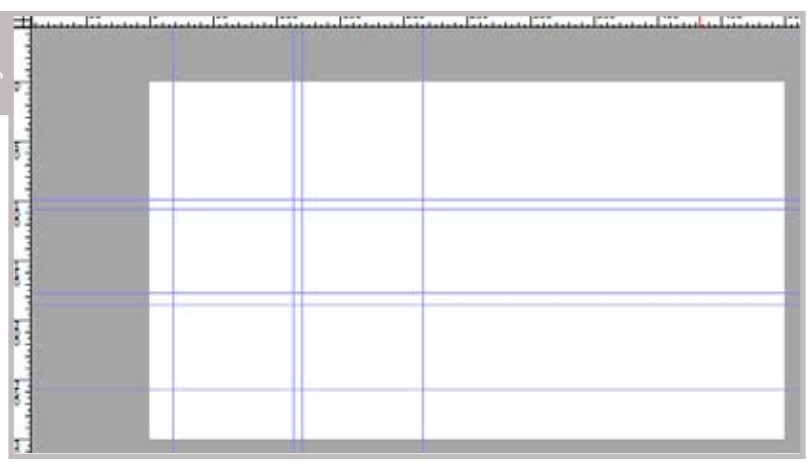
   The Directional Blur found in Effects & Presets->Blur and Sharpen, gives a layer the illusion of a motion blur. In fact, in earlier versions of After Effects, this was the name of the effect. Note that we changed the blur length at various points in the timeline. Twirl up the In layer. The other two In layers are nothing more than variations of the first one.
3. Twirl down the first Motion layer — Layer 11 — and scrub across its strip. You'll see the word Motion travel along the jagged line. We did this by adding a solid to the layer, drawing the path, adding the text, and dragging the Path Text effect from the Text folder in the Effects & Presets panel onto the solid.

   As you may have surmised, this is how you can do a "text on a path" animation effect in After Effects. You can define a path as a straight line, a circle of any diameter, or a Bezier curve. You can also import a path created in another application, such as Adobe Photoshop or Adobe Illustrator. This didn't make sense in our project, but it is something to be aware of if you need a more complex path. You can animate text on an existing layer or, for additional control over placement of text as we have done, create a solid and animate the text on it.

   You are probably wondering how the text moves from the top to the bottom. To move text along the path, create keyframes for the left margin or right margin and change the margin value in the keyframe. In our case, we used keyframes to adjust the Left Margin property of the effect as shown in Figure 4. In many respects, all you're doing as creating one seriously big indent in a paragraph.

   Twirl up the Motion layer.

Figure 4



The final "how to" involves how we created the series of letters that appear over each of the boxes in the Flash movie. The effect is a classic example of our axiom: "Let the software do the work." In this case, we used a Unicode character table to change a letter into a series of disjointed characters and finish with the capital of the letter we started with.

1. Twirl down the F layer — Layer 4. Select the layer strip and twirl down Window->Character. Change the text color from white to black. The text is white because it will reverse out of the solid color of the color block over on the Flash stage.
2. Scrub the playback head across the layer strip and you'll see the text start as an e, cycle through a series of random characters, and finish as an uppercase F. This is done is by assigning a Character Value animator to the text and changing the value using a series of keyframes. If you twirl down the Animator 1 in the layer, you'll see this

property and the keyframes (as shown in Figure 6).

Essentially what we did is to start with a lowercase e that has a Unicode value of 101 and finish the sequence with an upper case F, which has a Unicode value of 0. By starting with one value and ending with another, you can change the character on the fly.

3. With the text selected, reset the text color in the Character panel back to white. Select File-> Revert, and the project will revert to the state it was in the last time you saved the file.

At this stage of the process, once we were satisfied with the piece, we saved the project and rendered it out as a QuickTime movie using the Animation codec with the color depth set to Millions of Colors+. These settings ensure the white background color for the Comp is ignored when the video is created. If it isn't, the FLV file will hide the Flash stage.

The FLV file was created using the Flash 8 Video Encoder and the ON2VP6 codec with Encode alpha channel selected. The frame rate was set to 15 fps and the data rate was set to 200 Kbps. The video was encoded, and we quit the Encoder.

In Flash, we added a video object to the Library and then added the object to its own layer on the Flash timeline. To ensure the video was properly positioned, we set its X and Y coordinates to 0, 0 in the Flash Property inspector, gave it the instance name of video, and then locked the video layer.

Finally, there's another way to approach this project that results in an even quicker-loading Flash file, and many developers and designers are now just discovering it. The technique involves approaching complexity from a position of simplicity.

This project involves an FLV file that is 500 pixels wide and 300 pixels high. Even though the FLV file is about 600K in size, the user will still face a delay as the FLV loads. The After Effects timeline also has a lot of layers, and the effects applied in each of the layers are quite sophisticated. In short, this is one very complex project.

If you look at the Flash stage, you see that it is, for all intents and purposes, based on a grid design. All of the After Effects content plays within a defined section of the grid. This means that instead of one great big FLV file playing over the entire stage, you could have each of the animations contained in a separate FLV file. Suddenly one complex project is broken down into a series of rather simple pieces.

For example, the lettering effect that plays with the Character Values and is used in the boxes could be created in an After Effects Comp that matches the dimensions of the grid square it appears in. This would result in a series of four four-second animations appearing in separate videos that are 96 pixels wide by 72 pixels high. The upshot is a series of very small QuickTime movies and a series of FLV files that should come in well under 20K each.

Figure 5

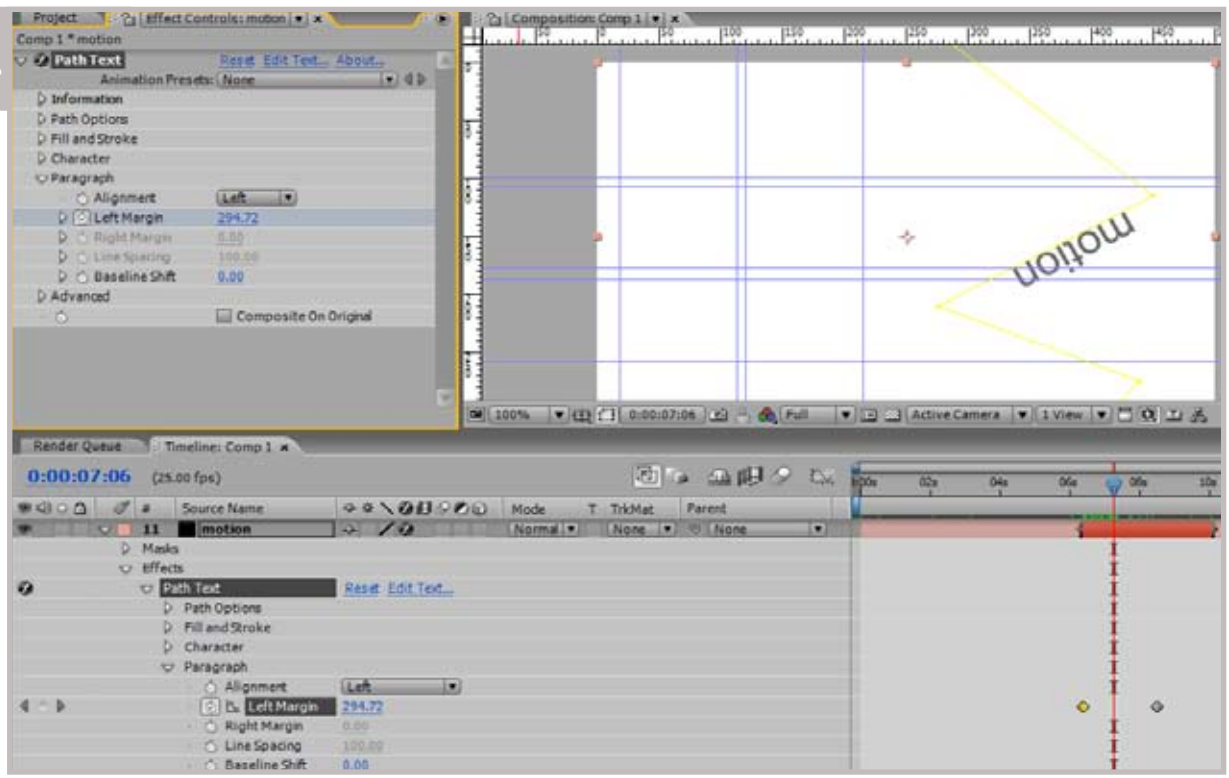
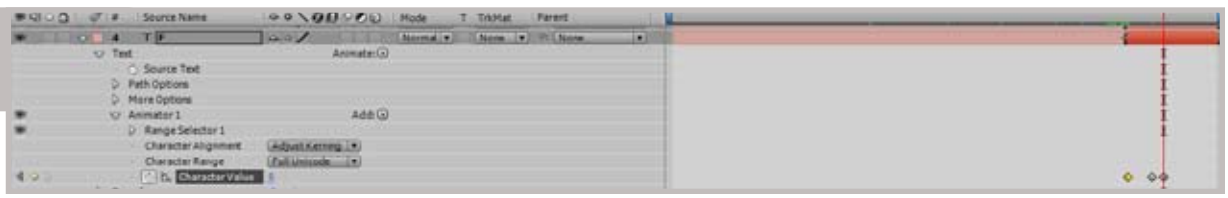
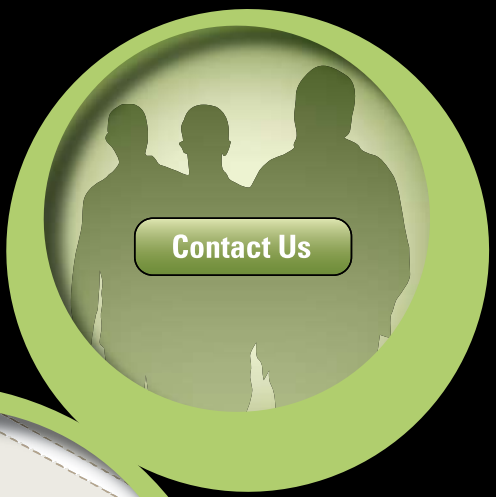


Figure 6



Scary Question.

Exactly who is developing your next app?



The Rise of RIAs

Reshaping users' expectations and experiences

by Luis Polanco

Today's online experiences often fall short of user expectations. For many consumers and business users, it is no longer enough to simply interact with static pages of information. Instead, a growing number of users want Web applications that deliver the responsiveness, performance, rich media and interactivity typically found in desktop applications.

The shortcomings of today's online experiences lie in how traditional Web browsers function. Built on an early paradigm – the request/response page metaphor – Web browsers include forward buttons, back buttons, and stateless communication protocols that fail to retain input once users leave a screen. The reality is that much of the demand and potential for Web services today has outgrown the capabilities of existing browsers and applications.

The rise of rich Internet applications (RIAs), a term Macromedia coined in 2002 with the release of Flash MX and Flash Player 6, is a response to changing business requirements and to the frustrations that developers and end users have with traditional Web applications. RIAs combine the rich interaction of desktop applications with the development and deployment model of the Web. Supporting real-time interactions and an intuitive user interface, RIAs make user experiences more engaging and meaningful. Users can interact directly with Web services, access and analyze data, and take advantage of functions like dynamic charting and drag-and-drop for richer, more streamlined interactions.

The Benefits of Thick and Thin

Adoption of RIAs is well underway. Industry leaders like Google and others are transforming Web applications to offer engaging experiences devoid of tedious "click-wait-reload" processes long familiar to Web users. Companies are also realizing RIA advantages in-house, creating business dashboards for managers to drill down, analyze, and report on critical sales and customer data in real time. Given the power and potential of RIAs, it's not surprising that the Gartner Group predicts by 2010 that at least 60 percent of new application development will include RIA technology.

In many ways, the growing acceptance of RIAs is a natural progression of the Web. For years, HTML fulfilled its duty by delivering static content to users requesting information. People were excited by the ease of accessing information and simply accepted limited interaction, presentation, and printing capabilities as the way of the Internet.

The restrictions of HTML, however, became more apparent as Web developers wanted to emulate the richness of desktop applications. RIAs offered a viable option, with applications that can harness the local processing power of desktop hardware and systems, which reduces server loads and also provides better user experiences.

Equally important, RIAs address the need to provide more compelling and productive end-user interactions that go beyond the capabilities of traditional Web applications. Improved responsiveness, the ability to work with large or complex data sets, the flexibility to manipulate screen views, and other routine functions – all common and popular in desktop applications – are prompting IT groups to look to RIAs.

A Maturing RIA Market

RIAs are ideal for IT managers who want to capitalize on the reach of the Internet and the benefits of Web deployment, without sacrificing application richness and interactivity. The applications can be developed and maintained at a fraction of the costs compared to creating and managing desktop applications. IT staff appreciate the cost-effective development, deployment, and maintenance of applications, while end users enjoy easy access to interactive and powerful tools.

RIA development options are keeping pace with increasing business demand for the applications. For instance, AJAX (Asynchronous JavaScript and XML) describes an RIA development model using existing Web technologies, including HTML, XHTML, Cascading Style Sheets (CSS), JavaScript, and XML. The intent of AJAX is to increase Web page usability, interactivity, and application speed

for the end user by exchanging small amounts of data with the server behind the scenes, so that the entire Web page doesn't need reloading each time the user interacts with the page.

The popularity of AJAX is a clear sign that the market needs RIAs. Part of the appeal of AJAX is that it overcomes some of the limitations that developers encountered using only HTML. AJAX is one approach for building RIAs but it has its own boundaries as well. While the Web programming is great, AJAX lacks the ability to provide application or UI constructs to build more dynamic and flexible RIAs that end users will demand.

Enhanced Engagement, Better Performance

What are developers' options when they need levels of functionality beyond HTML, JavaScript, and AJAX?

There are several solutions for building Web-based RIAs – some leverage existing Web design patterns while others introduce new ones. IBM and Sun offer a variety of solutions, ranging from RIA development frameworks and runtimes to components that bring rich Internet capabilities to existing Web applications. Microsoft too is introducing an additional programming model for desktop developers to build RIAs.

Adobe's RIA tools, such as Adobe Flex, allow Web developers to build compelling RIAs for the Web or desktop using technologies and design patterns already familiar to Web developers. Adobe Flex provides a programming model for building RIAs for both the browser and the desktop. RIAs built with Flex also leverage an enterprise-class runtime based on Adobe Flash Player, so end users can easily manipulate data and blend audio and video on any device at anytime. Achieving similar capabilities in a traditional HTML environment would be cost-prohibitive and time-consuming for developers.

With Adobe Flex, developers can take advantage of server-side data management services and data channels for enterprise integration. Enhanced data services in Flex 2 benefit applications requiring high-performance data transfer, real-time data streaming, message-based publish and subscribe, transparent cross-tier data synchronization, automated paging of large data sets, and occasionally connected applications. The dynamic data services simplify the move of Web services from static, request-response modes to dynamic "push" models.

Flex is built on J2EE as a plug-in to the Eclipse Development Environment. The data that RIA developers need to access is often stored in databases that are connected to J2EE servers. Flex 2 data services makes it easy for developers to connect RIAs to a J2EE database, such as a Salesforce automation or CRM system, through Web services and implement a rich, interface so users can more easily engage with vital enterprise information.

RIAs developed in Flex currently support a wide range of essential business and end-user applications, from guiding call center representatives through service calls to providing customers with real-time, online support using video, chat, and co-browsing help. With RIAs, insurance companies can link multiple systems to handle new account enrollment via an intuitive UI, while banks can streamline complex, multi-step workflows for mortgage application processing.

Overcoming Browser Limitations

As is evidenced by the tools available today, developers can continue to push the boundaries of RIAs, getting closer to desktop functionality to provide more expressive, immersive experiences. Users can enjoy offline or online interactions with applications, but also readily engage with rich media that includes integrated and seamless multimedia content.

To deliver on the full promise of RIAs, Adobe is providing capabilities that allow developers to easily combine the richness and power of desktop applications with the development and deployment model of the Web. Adobe's Apollo project is a new cross-OS, cross-device application runtime that will extend the reach of RIAs to the desktop. With Apollo, Web developers will be able to leverage their existing skills in HTML, XML, JavaScript, AJAX, Flash, or Flex to build RIAs that break free of browser and platform constraints, allowing them to run on the desktop.

"Successful RIA development depends on the flexibility to meet a wide range of application requirements, **whether it is simply extending capabilities in existing Web applications or creating full client-side applications running in browsers."**

As a result, developers can build applications previously inaccessible to them due to technology or learning-curve barriers; they can create desktop applications with ease using the tools that have been successful for building Web applications. In addition, with Apollo, updating desktop applications will be as simple as it is now to update Web applications.

Apollo will provide a set of APIs that can facilitate different capabilities, such as system file access, windowing, background processing, and system tray/toast notifications. These APIs will be accessible within JavaScript and ActionScript (both ECMAScript standard languages) for use by applications running on Apollo. Developers won't have to worry about the underlying implementation for each operating system because capabilities will be abstracted. The Apollo runtime will take care of integration with the desktop across all supported operating systems. For Web developers and companies, the advantages are clear, with both groups having the opportunity to bring new classes of applications and services to users.

Because the business logic is in the client, users can repeatedly manipulate content without reconnecting to the server. Communication with servers happens only as needed, such as when users save their work. In addition, developers can build RIAs that support automatic synchronization of data between servers and client-side objects. As back-end content is updated, changes are reflected automatically on users' devices.

Committed to Standards and Ease of Development

Adobe understands that RIAs can be composed of many technologies and works with leading organizations, such as ECMA and the OpenAjax Alliance, to ensure that standards are extended to meet RIA needs.

For example, Adobe Flex and open AJAX technologies complement each other well. The company's FABridge – a small, unobtrusive library of code that can be inserted into a Flex application, a Flex component, or an empty SWF file to expose it to scripting in the browser – helps integrate rich Flex components with AJAX applications. The FABridge is freely available to the community under an open source license. Spry, also free, is a client-side framework for AJAX development that is server and design-tool agnostic, allowing Web developers to incorporate XML data into their HTML documents using HTML, CSS, and a minimal amount of JavaScript, without needing to refresh the entire page.

More Open, Responsive Development

Successful RIA development depends on the flexibility to meet a wide range of application requirements, whether it is simply extending capabilities in existing Web applications or creating full client-side applications running in browsers. Of course, it is important to consider where RIAs should run – in a browser, on the desktop, or a combination of both. What is clear is that any approach should be as open as possible and have the ability to create RIAs for any platform.

That is one of the main reasons Adobe integrates AJAX with Flash and Flex. Developers can start with AJAX techniques and then use the open source Flex-AJAX bridge to take RIAs to the next level, streamlining the integration of Flash and Flex-based components – as well as vector graphics, audio, video, and charting – into AJAX-style applications. As user requirements expand, developers can respond with more dynamic applications developed in Flex and delivered in the upcoming Apollo runtime, which will allow developers to take their browser-based RIAs to the desktop.

Much like the familiarity of today's Web applications, RIAs will become increasingly common. Developers will be asked to build more sophisticated applications, and users will regularly access services filled with real-time updates, data manipulation, quality presentation and printing, blended video and documents, and other capabilities once limited to costly desktop applications. Already, the impact of RIAs are evident on Web sites and corporate intranets worldwide, with even the simplest RIAs helping to reshape users' expectations and experiences of what is possible with Web services.

Luis Polanco is senior product manager, Platform Business Unit, Adobe Systems.

For answers call us at 1-866-EDGEWEB
3 3 4 3 9 3 2

When calling your web host for support you want answers, not an annoying song stuck in your head from spending all day on hold. At Edgewebhosting.net, we'll answer your call in two rings or less. There's no annoying on-hold music, no recorded messages or confusing menu merry-go-rounds. And when you call, one of our qualified experts will have the answers you're looking for. Not that you'll need to call us often since our self-healing servers virtually eliminate the potential for problems and automatically resolve most CF, ASP, .NET, SQL, IIS and Linux problems in 60 seconds or less with no human interaction.

Sleep soundly, take a vacation, and be confident knowing your server will be housed in one of the most redundant self-owned datacenters in the world alongside some of the largest sites on the Internet today and kept online and operational by one of the most advanced teams of skilled Gurus, DBAs, Network and Systems Engineers.

By the Numbers:

- 2 Rings or less, live support
- 100% Guarantee
- 99.999% Uptime
- 2.6 GBPS Redundant Internet Fiber Connectivity
- 1st Tier Carrier Neutral Facility
- 24 x 7 Emergency support
- 24 Hour Backup
- Type IV Redundant Datacenter

EDGE WEB HOSTING

For a new kind of easy listening,
talk to EdgeWebHosting.net

http://edgewebhosting.net

CFDJ READERS' CHOICE AWARD
MXDJ READERS' CHOICE AWARD
2003 - 2006

◦ Shared Hosting ◦ Managed Dedicated Servers ◦ Managed Colocation ◦ Semi-Private Servers
◦ ColdFusion ◦ BlueDragon ◦ ASP ◦ .NET ◦ .Linux ◦ .Java ◦ SQL Server ◦ .MySQL ◦ Self-Healing Servers

What to Unlearn

Career guidance

by Jesse Warden

Some things that I learned early in my career that originally helped me succeed, I believe are now hurting me in job interviews. One of the pros to typing via dynamic languages and forgiving compilers such as ActionScript 1.0, Ruby, JavaScript, and others is that you can quickly code things that work. In a lot of the early agency, multimedia, and small software company work that I did, these technologies were great. They didn't get in your way, and they empowered you to quickly create programmatic solutions that were enhanced or even driven by good designers. You could hit insane deadlines, recompile changes quickly for a client, and react flexibly to ever-changing, almost fluid requirements...if any.

I believe those early successes helped me become very pragmatic, agile, and optimized. While to this day I still question a lot of what is considered "good programming" practices, I believe my pragmatism allowed me to judge which OOP, design patterns, and frameworks were appropriate, and how much of each were used based on a project's need. As my projects increased in scope, I learned to love strict-typing. Design patterns helped me organize my code enough to be able to quickly change the design for whatever reason and still have my code work. I learned to love frameworks such as ARP and Cairngorm as I started getting more development help on projects (I used to be the only developer).

Things have changed. With the release of Flex 2, 2006 opened the door for traditional programmers, both server-side and client-side, to now contribute and benefit from the SWF format. Now they can create RIAs, and really enjoy doing so. New blogs from talented developers you may not have known of, if you were strictly in the Flash development community, are popping up daily, all with great code and development techniques to contribute to the greater community. Some were sleepers, others are just now getting around to acquiring a blog because other traditional developers are doing it too, and some are just coming into the echo chamber that is the Flash/Flex development blogosphere. All bring with them their own style of development. More often than not, these are real developers who don't have the learned expectations of a lot of the early pioneers and frontiersmen, those first brave programmers like Branden Hall and Colin Moock. They are the ones who use test-driven development, Inversion of Control, and some even care about your CMM level. Some are all about the Flex component and FDS APIs, while others are swimming around the boilerplate, low-level Flash Player AS3 APIs, such as ByteArray and Socket, as if they were nothing foreign and new.

One thing is abundantly clear, though: these are programmers and they mean business – business in the sense that they have lingo, processes, and development setups that only used to trickle into the Flash development community. Granted, there is plenty of prior art to showcase all of these things in some shape, form, or fashion in our community, but never has it been on such a paradigm-changing scale as this.

Why Does This Matter?

For one thing, these guys 'n gals don't know the Flash Player; they just know they like what it can do. Most don't know, nor care to know, what an 8-bit alpha channel is and how it revolutionized what we can do with design on the Web. However, they know we do, and they want us on their teams because they need to focus on getting their J2EE back ends to work wonders with our Flex front ends. If you know Flash and/or have a minuscule amount of Flex experience with a design background, you are in high demand.

The common theme I keep seeing is that Flex is being used more and more in a lot of traditional enterprise development setups. This means having an actual quality assurance team, maybe even user testing, all with the assumption you'll use the tools and processes that traditional Java developers on large teams use. I feel a lot of Flash developers have spent their entire careers learning more and more about traditional software development to augment their ability to be more successful in their traditional multimedia and agency work. That is, of course, unless they truly have crossed over to full appli-

cation development and are just waiting for Flash Player 9 to hit the magic 80% penetration mark before they make the dive to AS3 or even Flex.

Either way, I feel we've all had an open mind about learning new things, since our industry in particular has changed so fast. While I may be extremely suspicious of using test cases, or appear quite unconvinced about having Spring lend some of its concepts to the client, I'm actually very open. Again, while we've had a lot of great traditional developers contribute their learning to our industry to help move us forward, never have we had an entire industry merge like this and on such a scale. I'm still learning every day about how a myriad of development teams do their work based on their varied backgrounds (C, C++, Java) and what tools and processes they use to make better software.

It's just really hard to match the lessons they've learned with my own pragmatic ones. They don't complement each other at all. They come to a harsh clashing in the old consulting adage of "fast, cheap, and good... choose two." If you want me to develop a widget for a client in one week (which is definitely more than reasonable), I see little value in creating test cases and implementing ARP. I've been burned too many times not to use source control, but what if I had two months? Would those tools and those who corroborate their joy of being converted now be justified? I'm sure the traditional software developers are nodding their heads saying, "Obviously." Of course, if you asked them if they blindly accepted new frameworks and processes, I bet most wouldn't say "No"; they'd look at them logically, and want to see proof of ROI of their time invested before learning them.

But is there any return? Do I have anything to give? What about all the accelerated development talents, tricks, and skills I've learned over the years to hit impossible deadlines with design and code – how do I convey value to them that a traditional software developer would understand? Obviously the Ruby guys "get" the advantages of loosely typed languages, and the agency non-coders love when we have good attitudes to make their designs do awesome dynamic things. How do you translate that to AS3, when you are rewarded twice for strong-typing: speed and maintainability? I could choose not to strongly type, turn off compiler warnings, type far less code, and get things done a lot faster. But to what gain? We are no longer suffering the agency problem, underbidding on both price and time, forcing designers and programmers to slave away for immense hours of time (well... some of us). Yes, clients in the enterprise sphere are as demanding as well, but we're talking months or years here, not days and weeks.

The software developers coming to Flex already have preset ways they develop; Flex is just a technology in their repertoire, while I think for a lot of us from a Flash background, it's a way of life. That way of life has drastically changed. I think it'll continue to change as swaths of more traditional software developers, both enterprise and small, get on board in such numbers as we've never seen before. For them, it's a nice, new opportunity to create more rich GUIs that are easier to deploy, easier to develop for, and more enjoyable for all involved. As for us old skoolerz? Our life is changing. What I'm having problems accepting is that it's "supposed" to fit into traditional development molds. It seems just obvious now that the computer scientists have their true IDE, command-line compiler, and strongly typed runtime. To me, it just seems one of the many ways to utilize the SWF platform, and, therefore, nothing is implied or inferred.

I feel like I have a lot to offer. I know how the Flash Player works. I know how to get a variety of designs to work. I know how to talk to any back end. I know how to get complementary technologies and tools to help make my development and design implementations efficient. I remove headaches from server-side developers, and they remove mine. I have client development experience that translates to both Web application and desktop development. I can get things done and, if need be, get them done quickly. The J2EE guys like this. What they don't understand is that us Flash and early Flex developers weren't around for the EJB wars, or all

"The common theme I keep seeing is that Flex is being used **more and more in a lot of traditional enterprise development setups."**

of the other battles fought over blogs, e-mail lists, and conferences about how Spring and Hibernate revolutionized server-side Java development and that industry as a whole. When we ask questions, it's because we didn't participate in that. We had our own battles dealing with the maturation of a fledgling language moving from a dynamic environment to a more traditional programming one. There were growing pains, yes, but they don't necessarily translate to process ones. This doesn't mean we have the proper context to appreciate what you all have to offer. Just something to keep in mind when you explain something that's wonderful in your work flow, and you're greeted with a blank stare. Something to think about. We're not dumb; we love new toys, and love this stuff as much as you do. An ActionScript developer's life is fraught with learning new things all the time. That's what makes it fun. It's just there are a lot more opinions and options now than five years ago.

I know that insane deadlines, hastily written ActionScript 1.0, and non-source controlled projects will still exist for some time. This could even be for me via the side contract jobs I take to supplement my full-time income. To me, those development decisions exist because you can thrive on them in such fast-paced environments. It's just been really hard to reconcile the lessons I learned from those areas into traditional software development, specifically in interview conversations. You can really sound like a buffoon to those people even if you've had a long history of proven successes and notes of merit. So far, I've had a few instances of tension with the bigger companies, and none with the startups. Basically, those who use more enterprise development technologies struggle to understand why I wouldn't want to use their uber-involved processes, whereas the startups and I really have a synergy in "getting things done." When I bring this up to the enterprise potential employers, they are definitely in agreement that they too like to get things done.

...but it's not the first thing, they say.

Anyway, just wanted to point out it's something that's been challenging for me to reconcile: past learned lessons on "how to survive as an engineer in a fast-paced environment" coupled with "how to develop maintainable, long-lived software." Different worlds, man, different worlds.

Jesse R. Warden, a member of the Editorial Board of Web Developer's & Designer's Journal, is a Flex, Flash and Flash Lite independent contractor. He maintains a Website at jessewarden.com where he writes about technical topics that relate to Flash and Flex. jesterxl@jessewarden.com

Advertising **Index**

Relax.

You're hosting with CFDynamics

We at CFDynamics work hard to have the most reliable servers, 99.9% uptime and outstanding customer service! We hope our customers enjoy just a little more "Relax" time because of our efforts.

Try us out today and see why our customers are kicking back a little more often.

- **99.9% Average Uptime!**
- **State-of-the- Art Data Center including HVAC, fire suppression, bandwidth & security**
- **24/7 Network Monitoring**

CFDynamics

The web host that helps you maximize your website

1.866.233.9626 customerservice@cfdynamics.com cfdynamics.com

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

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