



TeraTech News

Tools for Programmers

We make you a computer hero every day

12221 Parklawn Dr., Ste 200
Rockville, MD 20852
<http://www.teratech.com>

January 2003
Liz Arroyave, Editor

(800) 447-9120 • (301) 881-1440
Fax (301) 881-3586
info@teratech.com

Read about:

- What is new in ColdFusion MX?
- Is programming getting harder?
- Stored Procedures
- Vendor Retention
- ColdFusion Tip

What is new in ColdFusion MX?

ColdFusion MX, or CFMX, is the next release of ColdFusion, which brings much, much more to the table than just new CF features. Driven by Macromedia's drive to create an entirely new set of web application development tools (linking Flash, DreamWeaver, and more), it also brings to CFers the tools needed to take part in the latest development trends, including XML, web services, .NET and J2EE integration, and lots more.

In fact, there's really a lot more to CF than just those things. Sure, you may have heard about the ColdFusion component (CFC) capability, as well as the new graphing capabilities, etc. But did you know that there's really quite a bit more to CFMX than those top features that are marketed so heavily?



You can find out exactly what all those extras are at the CF204 class in March. We will identify lots of things that you didn't hear as much about and that indeed you might not have easily found on your own. Of course, we'll also briefly highlight those most popular and well-discussed features as well. And talk about migrating your code to CFMX.



As a bonus, there are many features you can leverage in CFMX that aren't discussed in any CFMX documentation, because they're really features of the underlying J2EE infrastructure. We'll point those out as well. Cool stuff!

For more details on CF204 and other ColdFusion and Fusebox classes see <http://www.teratech.com/training/>

Is programming getting harder?

Based on an article by Joel Spolsky

There was a time when if you read one book by Peter Norton, you literally knew everything there was to know about programming the IBM-PC. Over the last 20 years, programmers around the world have been hard at work building abstraction upon abstraction on top of the IBM-PC to make it easier to program and more powerful. But the law of leaky abstractions means that even as they built the abstractions that are supposed to make programming easier, the sheer amount of stuff you have to know to be a great programmer is expanding all the time.

Becoming proficient, really proficient, in just one programming world takes years. Sure, lots of bright teenagers learn Delphi one week and Python the next week and Perl the next week and think they are proficient. Yet they don't have the foggiest clue how much they're missing. Leaky abstractions mean that we live with a hockey stick learning curve: you can learn 90% of what you use day by day with a week of learning. But the other 10% might take you a couple of years catching up. That's where the really experienced programmers will shine over the people who say "whatever you want me to do, I can just pick up the book and learn how to do it." If you're building a team, it's OK to have a lot of less experienced programmers cranking out big blocks of code using the abstract tools, but the team is not going to work if you don't have some really experienced members to do the really hard stuff.

So for now, my advice is this: don't start a new project without at least one architect with several years of solid experience in the language, classes, APIs, and platforms you're building on. If you have a choice of platforms, use the one your team has the most skills with, even if it's not the trendiest or nominally the most productive. And when you're designing abstractions or programming tools, go the extra mile to make them leak proof.

<http://www.joelonsoftware.com/articles/LordPalmerston.html>

Stored Procedures (continued)

SPs are like super Views that can contain multiple SQL statements and other programming logic like loops and if statements written in the T-SQL language. They are compiled.

Q: When would you *not* want to use stored procedures to run queries etc. on a database? Wouldn't a stored procedure almost always run faster than one done through CF?

- If you're building a completely dynamic bit of SQL, then you just can't do it in a SP
- Also, SPs aren't always faster - I have some SQL where the tables are indexed to hell and back - the CFQuery version is faster than the SP by about 10%.
- Also, also, quite often we have tons of queries in a site and it's easier to manage without hitting hundreds of SPs than allow the SQL to be updated in the templates
- Also, also, also (this is just getting silly now), you just HAVE to do some things in CFQuery - like altering a table structure - it'd be silly making a SP for a one-shot SQL statement
- Although I use stored procedures for about 95% of my database calls, I would agree that there are some situations in which you might choose not to use them.
- You can do quite a bit of complex programming inside your SPs (e.g., using Transact-SQL in SQL Server), but you may not know that language. If you don't have the time or inclination to learn a new language, you might stick with using CF for some complex processing that could otherwise have been encapsulated within an SP.
- I both agree and disagree with regarding building dynamic SQL. I cannot think of many cases where you cannot do it in an SP, but if you are actually building your SQL statement from within the SP, then you lose one of the performance benefits of SPs (it cannot be pre-compiled).
- I would add another reason not to do SPs - they are harder to debug from CF than straight SQL - when it doesn't work you don't get a good error message from CF... Of course the regular CFQUERY error message isn't great either but at least you can cut and paste the SQL back into your query builder to test...
[more next month on SPs]

Vendor Retention (continued)

By Al Negrin

Just because a company is starting to experience hard times is no reason to communicate less effectively to vendors. Yet, suddenly many vendors aren't being given timely information about postponements and reallocations of purchases, aren't being involved in cost-saving design changes, and, worst of all, aren't being paid on time. When frantic vendors call, they are either told "she's in a meeting and will call back later" (which doesn't happen) or are diverted to subordinates, who "can't approve these invoices." Suddenly, the once active lines of communication go dark.



Frankly, I believe this is stupid and immature. Just because executives find their company in difficulty is no reason to clam up and try to pretend everything is all right. Vendors aren't dumb, and they know what's happening. It's happening to everyone. So why refuse to communicate? Do you think if a vendor's bills are 30 days overdue the vendor won't notice it if you keep silent and don't mention the subject? Come on; get real!

No one likes to be stiffed, but after all, if a vendor's been a partner in good times, they'll be one in bad times, too, *if you keep the lines of communication open*. So instead of retreating to your office and refusing to take their calls, why don't you take the initiative? "Hi, Bob, this is Amelia at Acme. You know that the business slump is really hitting us hard. Our volumes are off and cash flow is really slow. Can you help us out? I know it's hard, but can we work out a delayed payment schedule for the amount we owe you?"

Bob won't like it, but the alternatives are worse, so he will try to work out a plan. And what are the alternatives for you? If you don't tell him what's going on, he may be sore enough to take legal action. He may put you on C.O.D. And when times get better (and they will), you'll be at the bottom of Bob's list when it comes to getting his product. "Hi, Bob this is Amelia at Acme. I wonder if we can get 5,000 of your MCX3118's in two weeks?" Guess what Bob's answer will be?

Copyright ©2002, RampUp Solutions, Inc. By Al Negrin
www.RampUpSolutions.com
Contact us at articles at RampUpSolutions.com for permission to reprint this article.

ColdFusion Tip:

Did you know that ColdFusion string comparison is case insensitive? So in CF "a" EQ "A" is true. This is similar to many databases. If you need to do a case sensitive compare use the compare function.



Compare(str1,str2)

It returns 0 if str1 and str2 are EXACTLY equal.

If you missed previous articles, you can locate them at <http://www.teratech.com/news.cfm>

Copyright TeraTech 2002
TeraTech, Inc.
12221 Parklawn Drive, Suite 200
Rockville, MD 20852

TERATECH CUSTOM PROGRAMMING

- ColdFusion
- VisualBasic
- SQL, Oracle, Access, FoxPro
- Numeric Analysis