

# How I started in ColdFusion

Posted At : August 2, 2011 1:47 PM | Posted By : Mark Kruger

Related Categories: Business Of Development

In 1995 after 9 years of pastoring I left the ministry. Why I left is a long story of a self-destructive man hitting rock bottom and almost drowning - but finding redemption and new life through faith, family and the love of God. We won't dive into that here, but it's pretty compelling stuff I have to tell you. Meanwhile, the Muse had no skills other than piano playing and public speaking (and a penchant for Greek and Hebrew). Oh... and I could type like a banshee. So while I was trying to figure out what to do with my career (and trying to hold my marriage together), I took assignments with a temp agency. I collated and typed and copied while I scratched my head wondering what the heck I was good for anymore.

At some point I was given an assignment working the night shift at a credit union. I watched batch jobs run on a big DEC Alpha/VMS mainframe. Reports came out on 2 big green bar track printers. I watched for errors, updated reports and answered the phone. I decided I kind of liked being around all this technical stuff. I also discovered I had a knack for troubleshooting terminals and modems and the like. I checked out a "dictionary of computer terms" from the library and read it cover to cover while sitting in the lonely data center at 3 am. One of the ways I learn is by simply increasing my vocabulary and making connections. Pretty soon I knew how to ask the right questions to gain more knowledge. I believe God had given me the skills I needed to "start over".

To make a long story short, I read a bunch of books and took tests to become an MCSE and ended up working for a fellow named Jay at DTN financial services. I supported infrastructure for our division - sales tools, news feeds, laptops and desktops. Jay saw potential in me and I loved making him look good. One day Jay came to me with a ColdFusion script he needed to run. And that's where our story begins.

## The Introduction

Jay approached me with a problem on a Friday shortly before the end of the day:

- **Jay:** I can't get this script to work.
- **Muse:** How can I help?
- **Jay:** I don't know but you fixed our Qbasic issue so maybe you will see something.  
*[I had fixed a Qbasic macro parser that screen-scraped Freddie Mac data from a terminal emulator. It was my very first programming task ever. I had, as yet, never heard of "Hello World" - but I did have a 700 page book called "Qbasic Programming". I suspect I'm the only one who ever bought it let alone read it... but I digress]*
- **Muse:** Yeah... ok, what's it written in?
- **Jay:** ColdFusion. [chimes sound in my head]
- **Muse:** What is it?
- **Jay:** ColdFusion [chimes again]
- **Muse:** [shaking head] ... well ok, let's have a look.

Now Jay is one of a special breed of ColdFusion programmers. I would say maybe a third of all CF programmers fit into this category. First, he did *not* see himself as a ColdFusion programmer - any more than someone frying a grill cheese sandwich thinks they are a master chef. Instead, he was an extremely bright guy with an IT background

whose actual job was product manager. He saw a need for some internal web applications to support his sales staff. He chose ColdFusion as the straightest and fastest path to getting them done. By the time I arrived he had created data entry tools to sales to assign and track leads from the various promotions from mailers and TV commercials and subsequent reports - all using Access and Coldfusion 4.01.

Jay is what I like to think of as a fireman (and I share that trait in many areas). He sees problems and then finds quick and creative solutions. ColdFusion was a huge weapon in his problem solving arsenal. But he wasn't a CF programmer. He only wrote enough code to accomplish the task. As soon as it was working he stopped coding and whatever state the code was in was fine with him - as long as it accomplished his goal. And his goal wasn't pretty code or even maintainable code. His goal wasn't even fully leveraging the platform. It was narrowly focused on whatever form or tool he was creating.

In this case he had a new tool that he created and it required that he migrate his data into a new database and schema. But his script was failing on some of the queries. What kind of code does a fireman write? Well it's pretty dismal. The code in question looked something like this (from memory now).

```
<cfloop query="blah">
  <cfif x IS 1>
    <cfif x IS NOT 2>
      <cfset z = evaluate(x + y)>
      <cfif z IS 3>
        <Cfquery>...run a query</CFQUERY>
      </cfif>
    </cfif>
  </cfif>
  <cfif n is b>
    <cfif b is not w>
      <cfquery>...run a queyr</cfquery>
    </cfif>
  </cfif>
</cfloop>
```

You get the idea. It had variables named cryptically, nested logic with no combining, cascading if statements and it must have been 2000 lines. Jay basically wrote a procedure that cascaded his logic as he saw the data - working his way through each nuance as he dumped it out on the page. That's a fireman for you, but before you poooh poooh them, consider that there are millions of lines of fireman ColdFusion code out there that *have been running for a decade or more*. I know because I often stumble onto code that I wrote back then and wonder "what was I thinking??" In this case the error was a database error and I had a rudimentary knowledge of SQL. I told Jay I would look at it and get back to him on Monday.

As my eyes took in the ColdFusion code for the first time I must say I had an epiphany (or an apostrophe if Smee will forgive me). For some reason I "got it". Call me a CF savant but I could tell what was going on in the script. Having never read any documentation or help files I could still grasp the flow and most of the tags and their purpose. That's the power of ColdFusion - and incidentally why so many ColdFusion firemen exist (and possibly why so much poorly written code exists as well). In my mind I was able to "leap past" the things I didn't know a grasp the whole idiom right out of the gate. Moreover, I really thought it was fun and cool, and being in my manic phase I grabbed onto it as an obsession in an instant. On the way home I bought

"Mastering ColdFusion 4.0" by Arman Danesh and by Sunday afternoon I had read the whole thing and done many of the samples. Turning my attention back to Jay's script I rewrote it using a switch statement and combined logic. On Monday I presented him with the new script and he was thrilled. From then on he kept pushing ColdFusion debugging and programming tasks my way. It wasn't long before it was my full time job.

About 3 weeks later I was mulling over stock quote delivery to web sites. Our solution to date had been to run one of our quote delivery servers (that received quotes from a satellite feed - dish and all) connected to a serial cable on the server where an engine on the server received and updated the quotes. A programmer would then need to program to the socket of the engine to extract quotes. Solutions were usually COM or Jar files (although we had a nice C++ CFX tag). It was a very challenging solution and out of reach for most web site owners. Try convincing your NOC to install a 30 inch dish on the roof and run coax cable back to your rack. My mind wandered back to what I had read about WDDX. We could (I surmised) serialize any data into WDDX and deliver it directly to the server via HTTP where it could be parsed and utilized as a query object (or whatever). With Jay's support I created such a feed and we created some http request/parser objects in various languages (VB, JSP and Perl) that could be utilized on various servers. It was (in 1998) a "web service" that preceded the buzz over SOA. The product, **FinWin**, is still alive today 13 years later and the WDDX feed is still used.

## The Aftermath

From there I went on to found the **NE CFUG**, make lots of new IT friends, start a company, then a blog and generally began making a nuisance out of myself in the ColdFusion community. I was a fireman just like Jay for few years but soon I morphed into more of an engineer (in the sense that I became concerned with structure, best practices etc). And now I manage a staff of 18 and over 100 customers.

The next three or four posts are going to be about my journey in taking my skills (including CF) and building a consulting company. I've had it in my mind for some time to write such a series. I'm not sure what's kept me from it - perhaps just a feeling that I needed to progress further along before sharing this story. Recently, however, I was speaking at **D2WC** in Kansas City MO - a fabulous conference focused on both designer and developer workflows (organized by the amazing and talented Dee Sadler). My workshop was about stakeholder communication and several other tracks were about project management and what I like to call the **Business of Development**.

I was privileged to sit in on a workshop by Aaron Pederson and James Polanco from **DevelopmentArc**. It was called "Jumping Alligators" and it was summary of some of the pitfalls of estimating and project planning. As I listened I realized that I had made many of the same mistakes as these fellows were describing and learned many of the same lessons. James and Aaron both had tremendous insight and I learned a great deal, but I was also humbled by their willingness to share several stories of failure and the lessons they learned when projects blew up. It inspired me to go ahead and share my story mid-stream, even while I still have things to be embarrassed about (and seriously is that ever going to change?). Hopefully it will inspire some Muse readers who are "in the trenches" with me and we can grow together.