

# Coldfusion Sloggers of the World Unite!

Posted At : January 15, 2007 1:11 PM | Posted By : Mark Kruger

Related Categories: Humor and Life

We are often called upon to fix or upgrade applications for our clients. In that capacity we have had the fortunate (or often unfortunate) task of reviewing code that has been written by others. Of course this gives us CF Webtools developers things to talk about around the water cooler - "Hey... did you see that loop? What a piece of work eh?" (followed by raucous laughter). Hopefully we are not referring to our own code at the time. Anyway, these applications are often written by bright individuals with little or no experience in Coldfusion. Coldfusion is accessible in a way that makes it possible for someone to churn out a site or application if they just start at the beginning and keep "slogging away" till things seem to work. Of course, after the applications is under load or the data set grows or the technology changes you have to call in the big guns (that's where we come in). Still, the sloggers produce a great deal of the Coldfusion code in the world (perhaps contributing to Coldfusion's unwarranted reputation as being too "light weight" for serious applications). Here are some of the sloggers we've run into.

## Novice Coldfusion Programmer

Novice slogger code reminds me of code I wrote when first starting out as a Coldfusion programmer. I read a book called "Mastering Coldfusion 4" by Arman Danesh and took off like a rocket adding switches, pound signs, and if elses. Arman had given me the knowledge but I lacked the wisdom to use it effectively. My code was thousands of lines long and I never neglected to do in 150 lines of code what could be accomplished in 2 or 3 lines (read it carefully - you will get it). You can identify a novice programmer by looking for code like this.

```
<cfif #isdefined("var")#>
    <cfif var neq 0>
        <cfif var neq "">
            Do something important
        </cfif>
    </cfif>
</cfif>
```

Novice programmers haven't learned about "AND" or the popular "OR" operand yet. They also have the hard and fast rule, "When in doubt put pound signs around it".

## HTML Designer Turned Coldfusion Programmer

Now many folks come to Coldfusion from the world of HTML and find a home programming in Coldfusion. Some, maybe most of these folks are really programmers at heart and have only worked with HTML to "get things done". But there are a few folks out there that are really designers who dabble in Coldfusion. Their pages are beautiful but when they write code it defies description. Usually it is a confusing conglomerate of HTML tags and Coldfusion tags with little rhyme or reason. An HTML designers Coldfusion code looks like this:

...Error....

Sorry... I don't have the heart to show you what it really looks like. After all I'm trying to *promote* Coldfusion. Actually, a classic "right brain" designer Coldfusion mistake is this:

```
<cfif ...some condition...>
    ...Do something important
<cfelse>
    <!-- do nothing here --->

</cfif>
```

It's difficult for the designer to grasp that you can just *leave out* the "CFELSE" part of the tag. Something to do with symmetry I suppose. Or maybe it's the medications they are taking.

## Academic Programmer

This is a programmer who has never made the break between what they learned in school or in a seminar and the real world. This not only makes them incapable of growing it usually makes them arrogant and truculent about the *way things should be done*. Academic programmer's code looks like this:

```
<!--
    This page will have a cookie with the users
    first name in it. We will use the cookie to
    display the words "hello _____" where the
    blank represents the first name of the user
    in question. This is done to provide a friendly
    and reassuring message to returning users.
-->

    Users who have never been here before will not
    have a cookie and therefore will not see their
    name. Project management has signed off on this
    behavior (that unknown users will not see their
    name), but would like a review in 6 months.
--->
<!-- check for the cookie --->
<Cfif isDefined('cookie.firstname')>
<!-- If the cookie exists display the message --->
Hello <cfoutput>#cookie.firstname</cfoutput>
<cfelse>
    <!-- The cookie does not exists so (and this
    is approved by the project manager) no name
    will be displayed. --->
</CFIF>
```

If you are laughing right now you get it. If you are not laughing and you are thinking, "What's wrong with that?" then you really should have a couple of drinks before you program anything.

## Prefers Other Language

This is a programmer who likes PHP or .Net or some other language. They have been forced to work in this inferior language against their will in order to meet some requirement created by a person with the mental capacity of a chrysmeth... a chrysamtheme... a big white flower. Generally they rarely miss an opportunity to point out how much better A or B would be in *their* (obviously superior) language. They have never taken the trouble to learn where Coldfusion might benefit the code. Consequently their code looks like PHP or .NET code that has been shoehorned into Coldfusion.

In fact, a well known ecommerce package (which shall remain nameless) hired a Java

programmer to "port" their CF 4 code to CFMX. Since CFMX is a Java platform, that could have worked out pretty well, but what they got was a monstrosity of servlets, classes and impossible logic. In fact, this programmer took all the Coldfusion Query objects and read them into Java arrays so he could use javaesque loop syntax for display, adding dozens of lines of code to every place where queries were used. The result was un-scalable, un-maintainable, un-upgradable and unbelievable.

Still, if it wasn't for the folks above I suspect that fewer people would need CF Webtools or its services. So for all you CF sloggers out there, getting the job done by hook or by crook, both arrogant and lovable (and you know who you are). Let me say thanks! We salute you from the heart of our bo... uh... from the bottom of our heart.