

What is this OO?

Posted At : May 28, 2009 6:00 AM | Posted By : Steve

Related Categories: ColdFusion, OO Principles

The recent discussion on OO programming has created quite a bit of discussion, some with more heat than light, but other thoughts that have been illuminating. Some discussions covered whether or not OO has "won" over procedural programming. Some have suggested that those avoiding OO do so because they are unable or unwilling to learn new things.

My favorite thing, however, was "sipping the [OO] kool-aid" by [Matt Williams](#) (I owe you a beer, Matt!).

This was posted in a comment in an already [great entry](#) by [Peter J. Farrell](#). Both suggested taking OO development in slowly, rather than all at once. It really got me thinking (more on this in a minute). I particularly liked Matt's wording as it provided a nice juxtaposition to [Marc Funaro's](#) "That was the Kool-Aid I *sucked* down, little by little." - emphasis mine).

Many of the comments stressed the benefits of OO over "[Procedural Programming](#)". One comment even suggested (somewhat facetiously) that you would find more resumes listing object oriented program than procedural programming. This really helped me realize that I don't understand the terms being used. It seems like "Procedural Programming" is being discussed as though anything that isn't [Object Oriented Programming](#) is (by definition) procedural programming (what about Functional Programming?).

If that is the case, then of course no one would list it on their resume. I wouldn't have listed it on mine when I first started. I wasn't doing procedural programming (at least not intentionally), I was just throwing code together and hoping it worked.

The Matt Wilson throws out this "sipping the Kool-Aid" comment and it really got me thinking. I have always thought you are either "doing OO" or you aren't (stupid on my part? Maybe so). Can you sip the OO kool-aid? If so, are you "doing OO"? [Matt Woodward](#) suggests this term isn't really helpful and mentions that we should "begin to intelligently use OO principles".

So, maybe words are getting in the way here. I think the term "OO" can be useful, but I think it often gets in the way for those of us not doing it. Maybe it would be more helpful to talk about more specific techniques (encapsulation, DRY, separation of concerns) without talking about "OO" from time to time.

It reminds me of a television add where the husband is reading the health benefits of some kids food and his wife keeps cutting him off. The kids like the food, so the mom doesn't want them to know it is healthy. ("Here is why you should use encapsulation. Is that OO? No, no - nothing hard like OO. We're just talking about encapsulation here.")

In my case, I keep my code DRY, I use CFCs, I use separation of concerns, I ([almost always](#)) use encapsulation, I even sometimes use polymorphism. I read about [design patterns](#) and attend conferences to learn about OO principles (among other things). Am I sipping the OO Kool-Aid? I always say that I am not doing OO, because I am not using *objects*. Maybe my dim comprehension of the terms is a barrier to good communication.

Is the term OO getting in the way of a more fruitful discussion or am I missing something here?

If you missed the discussion, here are the relevant blog entries:

- [How OO Almost Destroyed My Business](#) by [Marc Funaro](#)
- <http://www.cfwwheels.org/blog/index.cfm/2009/5/24/How-OO-Almost-Destroyed-a-Business> by [Chris Peters](#)
- [OO Nearly Destroyed A Business?](#) by [Andy Jarrett](#)
- [On Responding to Criticism](#) by [Sammy Larbi](#)
- [The Trouble with OO?](#) by [Steve Bryant](#)
- [OO Can't Ruin Businesses, but People Can](#) by [Brian Kotek](#)
- [How do you learn OO?](#) by [Peter J. Farrell](#)
- [Throwing Down the Gauntlet](#) by [Scott Stroz](#)
- [This OOP Debate Isn't About OOP At All](#) by [Brian Rinaldi](#)

- [OOP + CF Debate Response](#) by [Kyle Hayes](#)
- new! [Who the !@\\$^# is Marc Funaro and What's Wrong With Him???](#) by [Hal Helms](#)

Apologies if I missed yours. I know I sometimes feel slighted when my posts get missed, but we all just miss posts that we meant to pick up.

I'll end this post by saying that although I have at some points found the recent discussion frustrating, overall I am really excited by all of this. I really love how this community can discuss these things in a mostly civilized manner and I really appreciate how many people take the time to help others learn and grow.