## Tipical Charlie
### (Tips and Resources from a Veteran CFML Developer)
*by Charlie Arehart*

## CFFORM: Are You Sure You Want to Ignore It?

By merely mentioning CFFORM, I risk ridicule and losing the attention of some who wrote it off long ago. It has long enabled creation and use of JavaScript and Java applets, and in ColdFusion 7 it added support for Flash forms. But did you know there are uses of CFFORM and its subtags that have nothing to do with JavaScript, Java, or Flash? And did you know that changes in 7.01 have addressed many concerns about the initial release of Flash forms? More important, are you aware of all the changes that were introduced in ColdFusion 7, especially for plain HTML forms?

Much press was given to improvements regarding "rich" (Flash and XML) forms, and many would think that those were the only changes. Traditional HTML forms received as many, and perhaps more, enhancements.

## HTML Forms Enhancements

There were several changes and additions in ColdFusion 7 for those using HTML ("plain") forms:

- **Masking:** Whether plain or rich forms, you can make a value typed in by the user take on some expected format. For example, a phone number could show the parentheses and dashes or an SSN could have the dashes already in place as they type. The special characters will appear in place whether the user types them or not; they are not duplicated. The Mask attribute enables and defines the mask. The value submitted to the form action page does include the masked value. Masking applies to either cfinput type="text" or CFGRIDCOLUMN.

- **Many New Validations:** Again, whether on plain or rich forms, you can choose new types of validation, including such datatypes as Boolean, email, guid, URL, USdate, and uuid. There are also other new kinds of validation: maxlength, noblanks, and range.

- **Control Over When Validation Popups Appear:** Ever wished that the popup validation message would appear on leaving the field, rather than on submission? The new validateat attribute offers values of onSubmit or onBlur (in addition to onServer, mentioned later), and you can use any combination of the three.

- **More Input Tags Supported:** Before ColdFusion 7, you couldn't use cfinput with hidden, submit, reset, and button fields so you had to be careful converting old forms to CFFORMS. This is no longer an issue in ColdFusion 7.

- Calendars – There are two forms of calendars supported, CFCALENDAR (HTML or Flash) or CFINPUT type="datefield" (flash-only, at least until ColdFusion 8).

- CFTEXTAREA – New, with validations like required and maxlength. Version 7.01 added an HTML attribute for Flash forms to indicate if the content was HTML or not.

- CFSELECT – New QueryPosition attribute controls placement of static <option> values above/below query-generated options. Selected attribute now accepts list of values (can select multiple values).

- CFFORM – Name and Action attributes now optional, defaults to self-posting.

- JavaScript Library – Code supporting all this is in a /CFIDE/scripts/cfform.js file. Servers using multi-homing or multiple web applications may experience errors locating the file. See "Using CFFORM in a multihomed environment" in Migrating Applications to ColdFusion MX 7 (http://download.macromedia.com/pub/documentation/en/coldfusion/mx7/cfmx7_migrating.pdf) regarding how the file location can be indicated using the cfform ScriptSrc attribute or an Admin console specification.

- **Protection Against Multiple Submissions:** CFINPUT type="submit" supports a new validate="submitonce" (in HTML and XML forms, not Flash) to ignore multiple submissions of a single form.

- ❖ **Enhanced Server-Side Validation:** New validateat="onServer" option uses hidden fields, but they're built by ColdFusion with names less likely to conflict with your own form field names.

Other new features include the ability to add tooltips, to control visibility/enabling of input fields, to bind data among fields in a form, and more. Note that there were over a dozen bug fixes in 7.01 related to Flash forms. Many initial annoyances are now resolved. There was also new functionality added in 7.01, including a new CFFORMITEM type="script" to add ActionScript code to the form.

It's still true that many (though not all) of the CFFORM features for HTML forms rely on JavaScript, which your user may disable. Similarly, if you rely on any of the options for validation and security, beware that a savvy user could circumvent them by editing the HTML result of your form. You can and should back up such client-side protection with server-side equivalents coded in CFML, or using server security mechanisms.

Beyond the changes in ColdFusion 7, some classic features of CFFORM and its subtags are worth knowing, such as the Preservedata option for reloading data in self-posting forms, and the ability to build a Select control from a query using CFSELECT. Neither relies on JavaScript.

In ColdFusion 8, more CFFORM-related changes are coming which also do not rely upon Flash, including an all-HTML cfgrid, a new cfinput Type="autosuggest" and other Ajax-driven features, binding and datefield support in HTML forms, and PDF forms support. Don't dismiss cfforms so readily. Make sure you know all they're capable of, whether you are using plain HTML, Ajax, Flash, JavaScript, or Java. Even if any one of those features is undesirable, consider all the others, and remember all the new features mentioned above in both 7 and 8.

---

Charlie Arehart is a longtime contributor to the ColdFusion community and a member of the Adobe ACE program. Formerly tech editor of the CFDJ and CTO of New Atlanta, he is co-author of the CFMX Bible. A certified Advanced CF Developer and Instructor for CF 4/5/MX, he is president of the Atlanta CFUG and a frequent presenter at conferences and user groups worldwide. He is now an independent contractor.

# Letters to the Editor

Greetings, I've been meaning to send you an email for quite a while now. I just  wanted to say thanks for the great article you did on UDFs and CFCs in  FAQU 2. Between your article, your recommendation of the book *Object  Technology: A Managers Guide*, one or two other sources, and beating my  head against the wall for a while, I finally 'get it.'  I know I have a  lot more to learn still about properly using OO techniques in CF programming but at least now I can get started with the foundation of knowledge provided by you.

I'm looking forward to the next issue of FAQU.
Thanks again!
James Husum

*James, Michael was very happy to see that his article helped you 'get it'. That's our intention in publishing FAQU – that each article, as well as all of them together, would help things gel in our readers' minds. We're happy that it worked for you, and we'll try to make each issue of the journal even better!*

---

Hi,  I just read your "Getting Started with CFEclipse" article [CFEclipse CFExplained in Volume 1 Issue 1] and I am very impressed. I also want to thank you for spreading the word of CFEclipse to the CFers out there! We are trying to add a lot of info to our own website, but all the articles out there are great, especially if they come from other parties.

Again... many thanks!
Mark Drew
Lead Developer, CFEclipse Project

*We hope you enjoyed this first Letters to the Editor column. Send us a letter with your feedback, comments, suggestions on this or past issues, and you may see your name in a future issue! No contact information is shared, unless requested.*
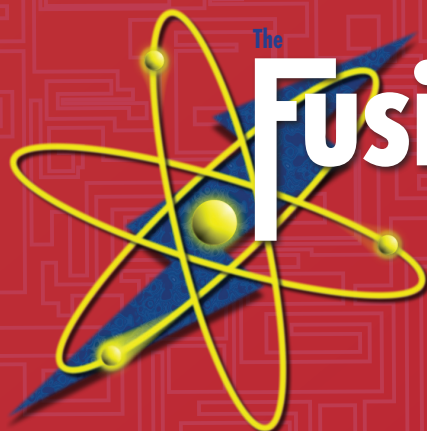
# The Fusion Authority
## Quarterly Update

## Is Your Message Getting Through? Focus on the User Interface

### You-I Design
### Separating Layout from Logic
### Prototyping for Interface Driven Architecture

## and more...

## A Step-by-Step Primer to Flex Development

### Business Reporting and Printing
### Building Flex Components for Reusability
### AIR Basics with Flex

## Plus

### Making Google Pay II