

Adobe® ColdFusion® Builder™

A professional tool for ColdFusion development

Table of contents

- 1: CFEclipse
- 1: ColdFusion Builder
- 2: Code editing features
- 5: SQL editing features
- 7: Code refactoring and updating
- 8: Debugger
- 8: Extensions
- 9: Summary

Since the release of ColdFusion, the development tools have evolved from simple text editors to more robust integrated development environments (IDEs). An IDE provides a suite of services that help users write, test, and debug code. Two of the most popular IDEs for ColdFusion development today are Adobe ColdFusion Builder and CFEclipse. This white paper describes how ColdFusion Builder compares to CFEclipse.

CFEclipse

Created in 2004, CFEclipse (www.cfeclipse.org) is a free, open source plug-in for the Eclipse platform that provides basic ColdFusion development support. Its listed set of features include ColdFusion component (CFC) and CFM outlines, color coding, code folding, code completion and insight, snippets, web previewing, and frameworks support. Multiple developers have been involved in creating the plug-in, most notably Rob Rohan, Stephen Milligan, and Mark Drew, who is the current project lead. The current version, CFEclipse 1.4.2, contains support for ColdFusion 9.

Users must install CFEclipse via the Eclipse update system; no standalone version is available.

ColdFusion Builder

ColdFusion Builder (www.adobe.com/products/coldfusion/cfbuilder/features) is a commercial plug-in for the Eclipse platform that provides ColdFusion development support. Released in 2009, its listed features include color coding, code completion, code insight, code assist, an integrated debugger, SQL color coding and insight, and extensibility via CFML. The current version, ColdFusion Builder 1.01, contains support for ColdFusion 9.0.1.

Users can install ColdFusion Builder either as a plug-in to an existing Eclipse install or as a standalone program, which makes it easier and quicker for people who are not familiar with the Eclipse platform to start developing with ColdFusion.

Code editing features

Both CFEclipse and ColdFusion Builder support color coding CFML templates. The following screenshots show the same code within both editors.



```
<html>
<head>
<title>This is a test.</title>
</head>
<body>
<cfloop index="x" from="1" to="10">
  <cfoutput>This is some simple CFML: #x#<br/></cfoutput>
</cfloop>
<cfmap centeraddress="Lafayette, LA" height="500" width="500" name="Home" />
</body>
</html>
```

CFM script in CFEclipse

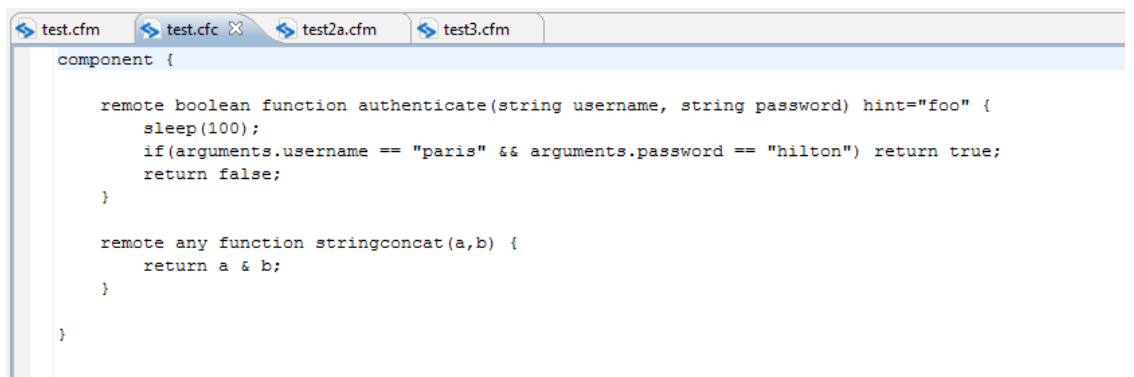


```
1 <html>
2
3 <head>
4 <title>This is a test.</title>
5 </head>
6
7 <body>
8   <cfloop index="x" from="1" to="10">
9     <cfoutput>This is some simple CFML: #x#<br/></cfoutput>
10
11
12
13 <cfmap centeraddress="Lafayette, LA" height="500" width="500" name="Home" />
14
15 </body>
16 </html>
```

head (end tag)
Non-rendered part of an HTML document that may contain meta information and may define links to scripts, stylesheets, and other documents.
Press 'F2' for focus

CFM script in ColdFusion Builder

ColdFusion Builder color coding provides superior support for script-based CFCs, as seen in this comparison.



```
component {
    remote boolean function authenticate(string username, string password) hint="foo" {
        sleep(100);
        if(arguments.username == "paris" && arguments.password == "hilton") return true;
        return false;
    }
    remote any function stringconcat(a,b) {
        return a & b;
    }
}
```

CFEclipse color coding

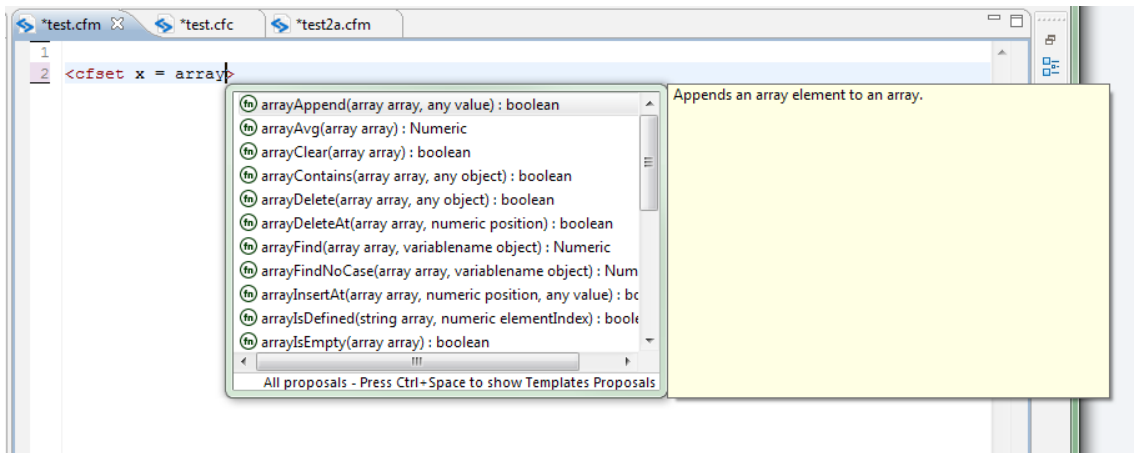
```

1 component {
2
3     remote boolean function authenticate(string username, string password) hint="foo" {
4         sleep(100);
5         if(arguments.username == "paris" && arguments.password == "hilton") return true;
6         return false;
7     }
8
9     remote any function stringconcat(a,b) {
10        return a & b;
11    }
12
13 }

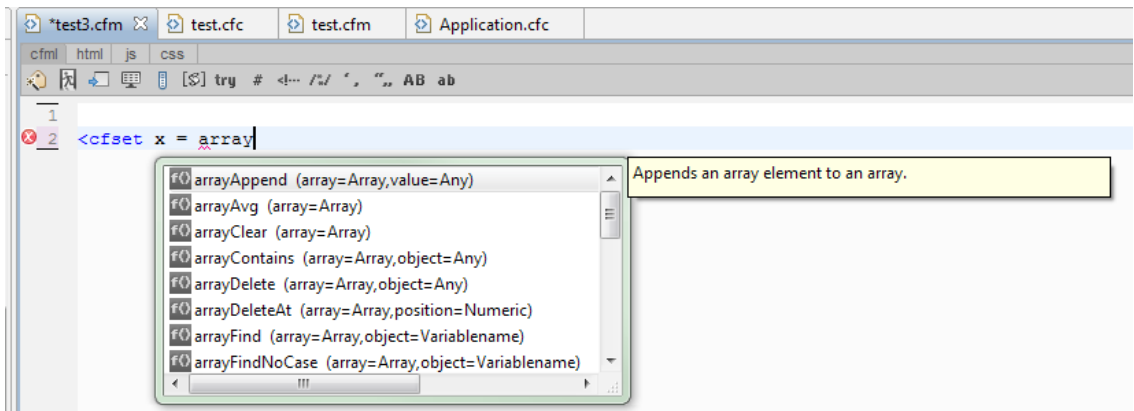
```

ColdFusion Builder color coding

Both editors also support code assist. For example, after typing the first few letters of a function, the editors can complete the function for you.



CFEclipse code assist feature

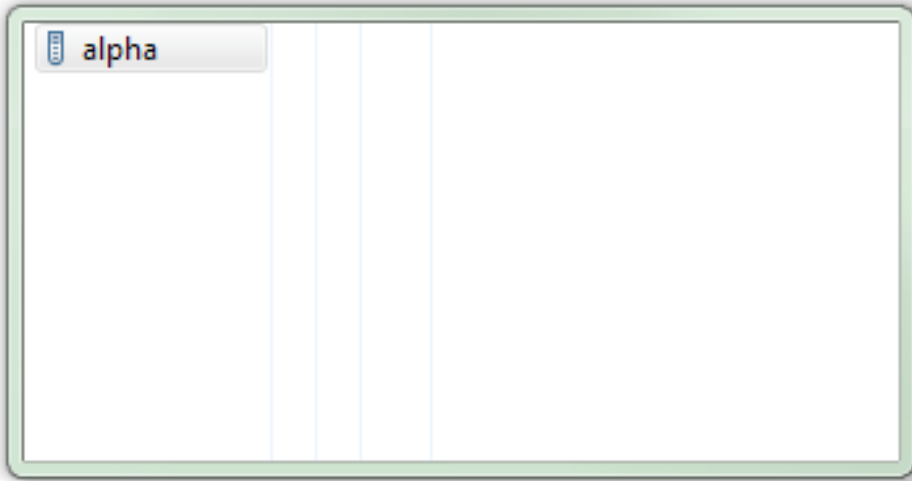


ColdFusion Builder code assist feature

However, ColdFusion Builder provides code insight in addition to code assist. It can introspect both the current open file and your project to provide help with variable names, CFC names, and methods. This can dramatically improve development time. For example, in the following figure, the template has two variables defined, alpha and beta. As you begin to type another line and enter a few characters, ColdFusion Builder is able to recognize that the variable alpha was already created. If the developer had begun with "b" instead, beta would be one of the possible options. ColdFusion Builder will actually suggest both built-in functions and your own variables.

```
<cfset alpha = 1>
<cfset beta = 2>

<cfset z = al
```

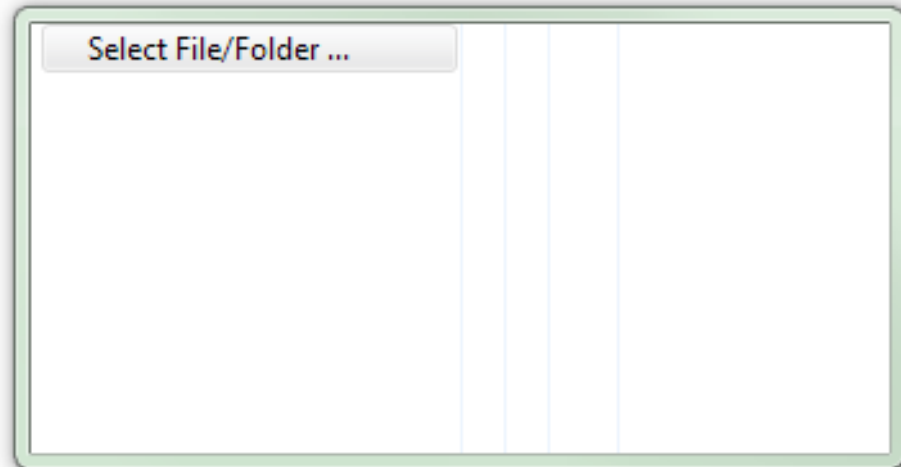


ColdFusion Builder code assist

Code assist also extends to other files in your project. The following example shows the cfinclude tag. After using the template attribute for cfinclude, you can immediately begin browsing your project to find the file that you want to include.

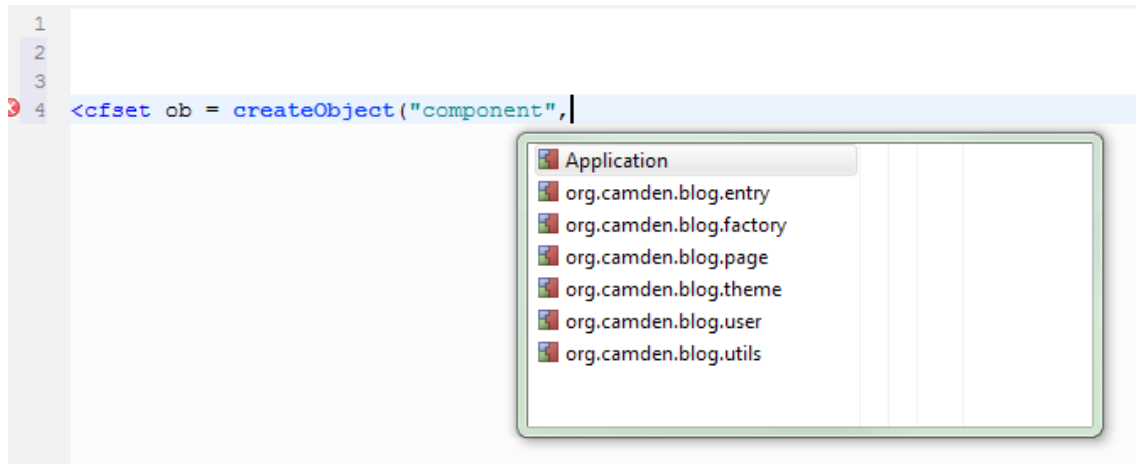
```
<cfset alpha = 1>
<cfset beta = 2>

<cfinclude template="
```



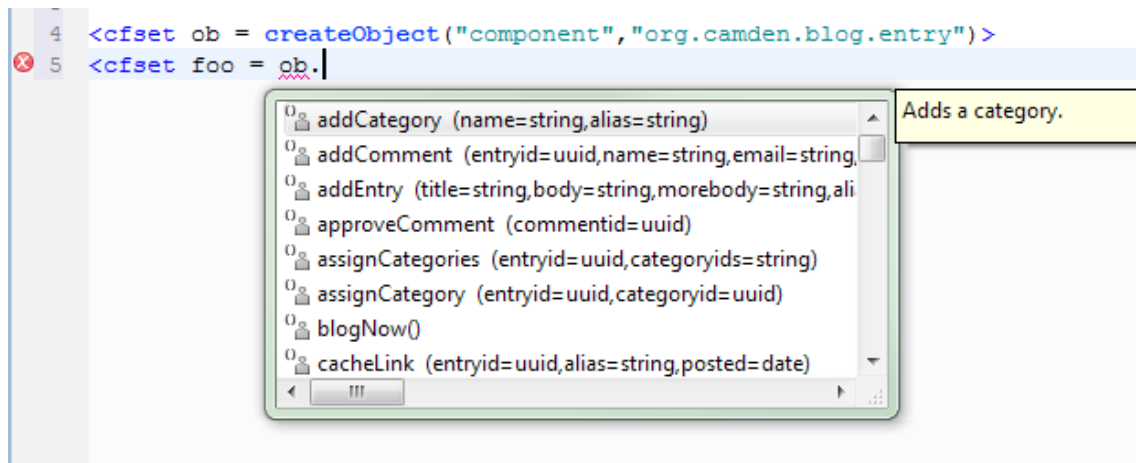
File selection for the cfinclude tag

Code assist is also helpful for CFC development. It provides easy access to all your CFCs within a project as well as their methods. In the following example, notice that ColdFusion Builder has detected the createObject() function and automatically provides a list of available components.



ColdFusion lists the available components

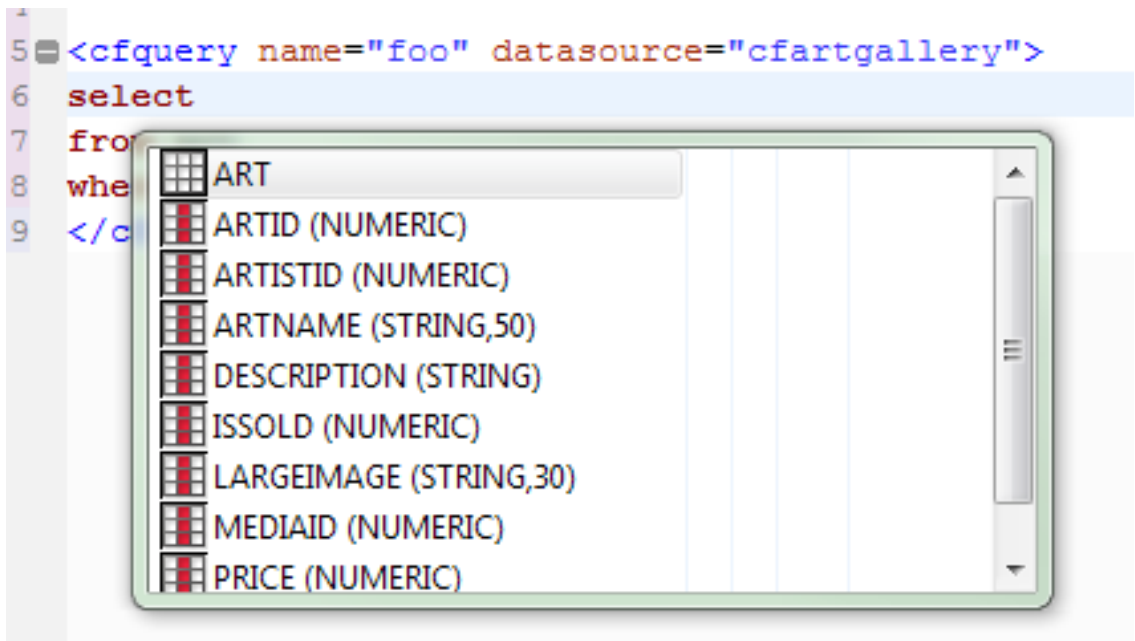
After a CFC is instantiated, ColdFusion Builder then associates that variable with the component. This allows you to get code hinting for methods. For example:



List of component methods, arguments, and hints

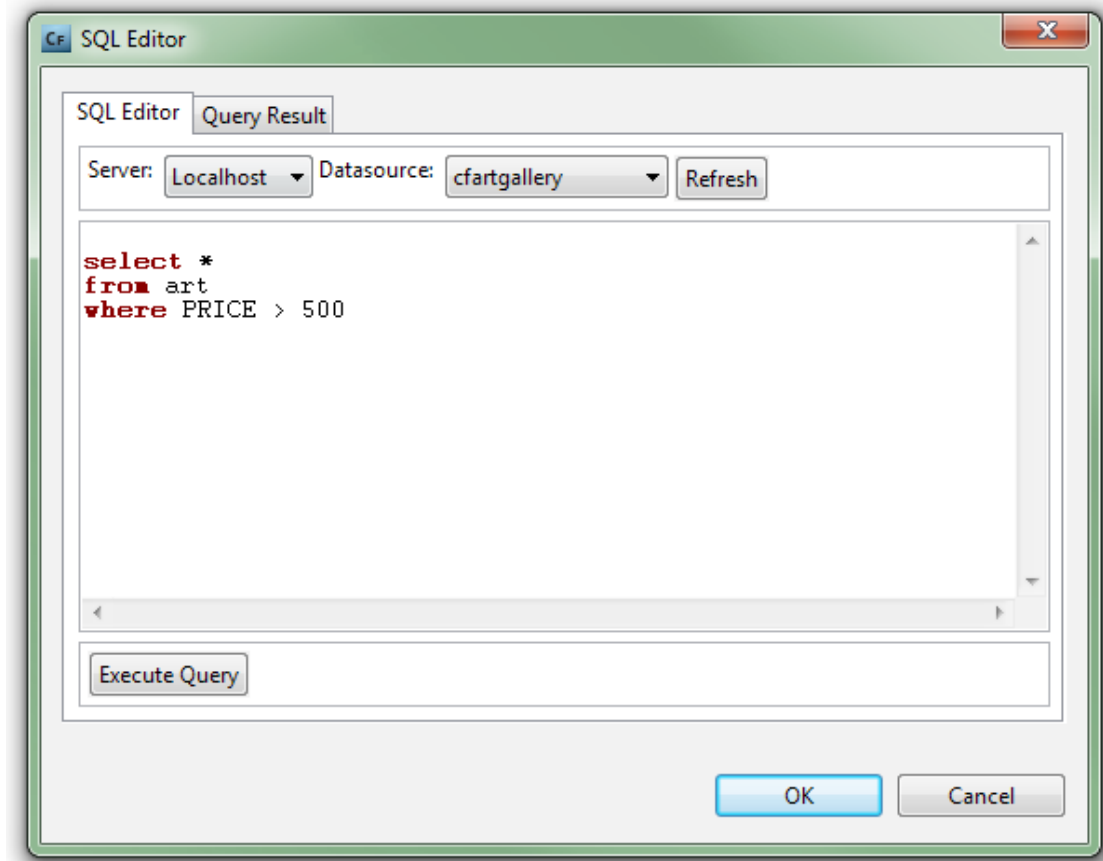
SQL editing features

ColdFusion Builder code hinting and assist features also extend to SQL and the object-relational model (ORM). Working with databases is an important part of any web application. Both CFEclipse and ColdFusion Builder support SQL color coding. However, ColdFusion Builder provides multiple enhancements that greatly improve database work, such as code assist. While typing, the editor provides assistance for the name of the datasource, the name of all the tables within that datasource, and all the columns for a selected table.

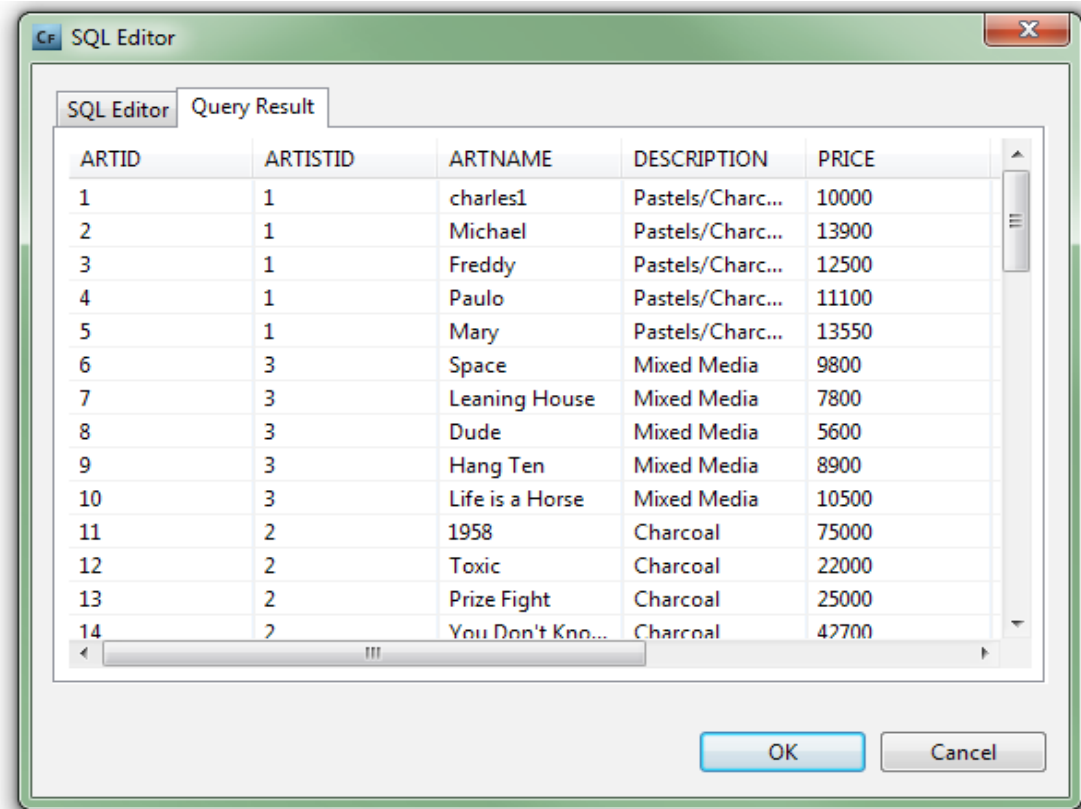


ColdFusion code assist displays the columns for the currently selected table in the query

ColdFusion Builder also provides a SQL editor that lets you write SQL statements and test the output. This can dramatically speed up development time. You can launch the editor by simply right-clicking an existing query.



The ColdFusion SQL editor



Testing the results of a query

Along with simple SQL support, ColdFusion Builder assistance also works well with projects using ORM. The editor provides code hinting for entities and any implicit or explicit methods that they define.

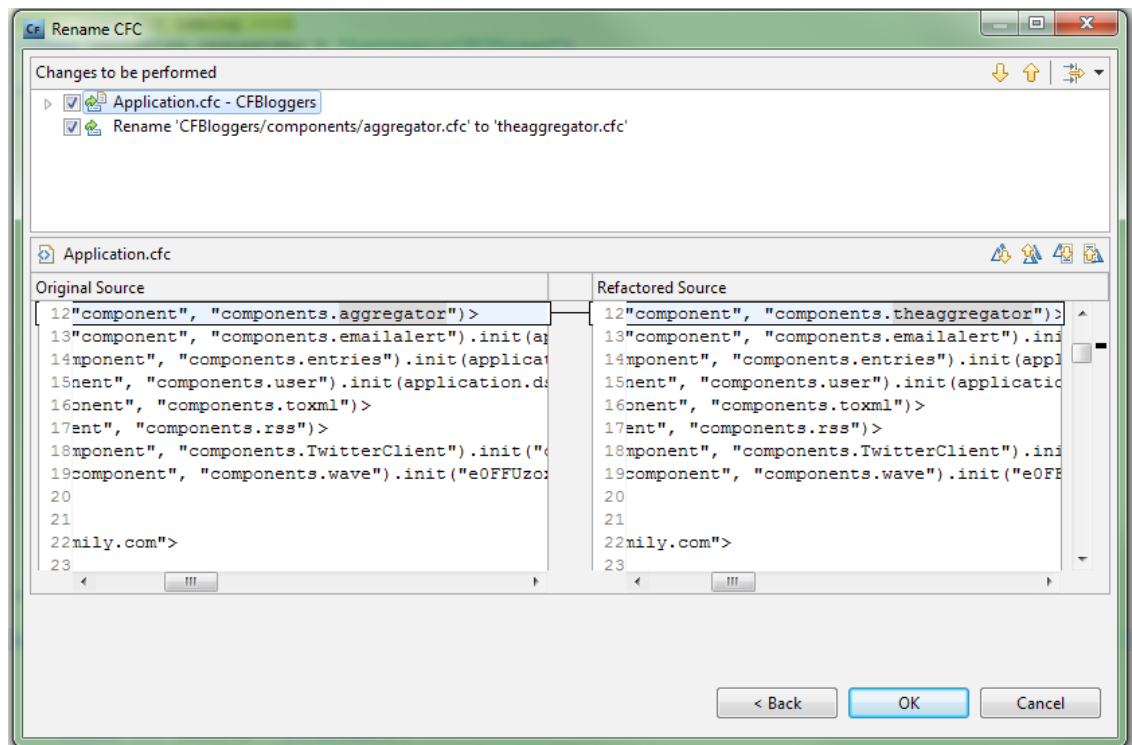
Code refactoring and updating

All projects evolve over time. Unfortunately, sometimes minor code changes in one area can have a dramatic impact on the rest of the project. ColdFusion Builder provides support to help with such changes, specifically focusing on renaming CFM or CFC files and methods. The following screenshot shows how you can right-click a CFC file to rename it.



Renaming a file

Notice that you cannot simply perform the refactoring, but must preview it first to ensure that the refactoring correctly updates the project. After clicking Preview, ColdFusion Builder scans the entire project for all uses of the component. It presents a list of the matches along with the changes it would make.



ColdFusion Builder lists where the component is referenced and how it updated the component name.

Debugger

One of the biggest features an IDE can provide is an integrated debugger. CFclipse can work with third-party ColdFusion debuggers. With ColdFusion Builder, this feature is built into the core product. The debugger allows you to step through your code as it is being processed in the web page. You can add breakpoints that pause the active requests and precisely check the value of variables at a given point of time. You can then step through the rest of the request and watch as these variables change. This is an incredibly effective way to track down problems within projects and goes a long way to speeding up the resolution of bugs.

Extensions

ColdFusion Builder differentiates itself in the capability to build extensions using CFML instead of Java, which is traditionally required to extend an Eclipse based IDE. Regardless of the IDE, it is impossible to provide every imaginable feature that a developer may require; however, ColdFusion Builder provides an extension API that allows developers to create their own tools within the editor using CFML. This API provides access to projects, files, and databases recognized within the tool. ColdFusion developers can use their existing skills to quickly design their own extensions for ColdFusion Builder. Today RIAForge provides 40 open source extensions at www.riaforge.org/index.cfm?event=page.category&id=14.

In addition to the numerous extensions available, organizations can create their own extensions that do everything from internal due diligence checks to company-specific code formatting.

Summary

Both CFEclipse and ColdFusion Builder allow you to write ColdFusion code, so which is appropriate for your needs? CFEclipse is a free, open source tool with a large community behind it, but developers and organizations looking for professional support will most likely prefer ColdFusion Builder. Professionals who make their living writing code and need a product that offers support and a consistent bug release and update cycle will also prefer ColdFusion Builder.

For more information

Product details: www.adobe.com/products/coldfusion



Adobe

Adobe Systems Incorporated
345 Park Avenue
San Jose, CA 95110-2704
USA
www.adobe.com

Adobe, the Adobe logo, ColdFusion, and ColdFusion Builder are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries. Java is a trademark or registered trademark of Oracle and/or its affiliates. All other trademarks are the property of their respective owners.

© 2010 Adobe Systems Incorporated. All rights reserved. Printed in the USA.

91033354 11/10