

MAX 2006 Beyond Boundaries

Ben Forta
WD203H
Building ColdFusion Powered
Flex Applications

2006 Adobe Systems Incorporated. All Rights Reserved. 1

Objectives

- Understand the ColdFusion Flex relationship
- Become familiar with Flex Builder and the Flex Builder ColdFusion extensions
- Define Flex projects for use with ColdFusion
- Use a variety of Flex components
- Use data bindings and data providers
- Work with renderers and formatters
- Use the FlexBuilder debugger

2006 Adobe Systems Incorporated. All Rights Reserved. 2

The Flex Product Line

Flex SDK 2 (FREE)

- MXML and ActionScript 3.0
- Flex Framework and Class Library
- Command-line Compiler & Debugger

Flex Builder 2

- Visual Layout
- Code Hinting
- Debugging
- Skinning and Styling

Flex Charting

- Extensible Charting Components

Flex Data Services 2

- Message Service
- Data Management Service
- RPC Services

eclipse

2006 Adobe Systems Incorporated. All Rights Reserved. 3

What You Will Need

- ColdFusion MX 7.0.2 (any edition)
- Flex Builder 2
- Flex Builder ColdFusion extensions

2006 Adobe Systems Incorporated. All Rights Reserved. 4

Flex Builder

- Flex Builder is an Eclipse based IDE used to build Flex applications
- Flex Builder can be installed as:
 - A standalone IDE
 - As a plug-in in an existing Eclipse installation
- Key areas of interest:
 - The editor
 - Design view
 - Navigator (Project panel)
 - Outline panel
 - Problems panel
 - ColdFusion extension panels


2006 Adobe Systems Incorporated. All Rights Reserved. 5

Flex Builder ColdFusion Extensions


- Flex Builder ColdFusion extensions support:
 - RDS Dataview panel
 - RDS Fileview panel
 - Services Browser panel
 - SQL query builder
 - CFC->AS and AS->CFC conversion wizards
 - Database table abstraction wizard
 - ColdFusion / Flex application generation wizard
 - Extensive CFML and ColdFusion documentation
- During Flex Builder installation, ColdFusion extensions are saved locally but are not actually installed into the IDE
 - Install using Help -> Software Updates -> Find and Install

2006 Adobe Systems Incorporated. All Rights Reserved. 6

Exercise 1 MAX




- Follow along with your instructor as we explore Flex Builder, and the Flex Builder ColdFusion extensions


2006 Adobe Systems Incorporated. All Rights Reserved. 7 

The ColdFusion / Flex Application Wizard MAX


- The ColdFusion / Flex Application Wizard can be used to build a complete ColdFusion powered Flex database front end:
 - Unlimited pages/screens
 - Data drill down
 - Full editing capabilities
 - MDI style UI
 - Best practices based code generation
 - Safely reentrant

2006 Adobe Systems Incorporated. All Rights Reserved. 8 

Exercise 2 MAX





- Follow along with your instructor as we use the ColdFusion Flex Application Wizard to build a complete ColdFusion powered Flex database front end (supporting data retrieval, insert, update, and delete) – all without writing a single line of code!

2006 Adobe Systems Incorporated. All Rights Reserved. 9 


It All Starts With A Project MAX

- Every Flex Builder application you build is defined as a "Project"
- Projects may be created underneath the ColdFusion root, if needed
- A new Flex application should be of type "Flex Project"
- To simplify ColdFusion integration, be sure to select "ColdFusion Flash Remoting Service" as the means by which your application will access data




2006 Adobe Systems Incorporated. All Rights Reserved. 10 

Exercise 3 MAX




- Create a new Flex Builder project for use with ColdFusion Flash Remoting

2006 Adobe Systems Incorporated. All Rights Reserved. 11 

Use Flash Remoting MAX

- Flex applications can access back-end data a variety of ways:
 - HTTP access to XML
 - Web Services (SOAP)
 - Flash Remoting (AMF)
- As a general rule, use Flash Remoting to connect to ColdFusion
 - Simplest integration
 - Best performance
 - Automatic object mapping between CFCs and AS classes

2006 Adobe Systems Incorporated. All Rights Reserved. 12 

Flex & ColdFusion Components

- Flash Remoting connects to ColdFusion Components
 - Yes, you need to be using ColdFusion Components (but you should already be using these anyway!)
- CFCs used with Flash Remoting must NOT be web-centric in any way:
 - NO output, NO assumptions about environment, NO reliance on CGI or cookies, NO use of sessions
- Any ColdFusion data types can be used, both when returning data and when receiving arguments
- More sophisticated applications will pass "objects" back and forth, CFCs on the ColdFusion side and ActionScript classes on the Flex side
 - Flash Remoting will auto convert between the two

2006 Adobe Systems Incorporated. All Rights Reserved. 13

<mx:RemoteObject>

- The Flex `<mx:RemoteObject>` tag is used to define a ColdFusion Component to be accessed via Flash Remoting:


```
<mx:RemoteObject id="svc"
  destination="ColdFusion"
  source="Path.To.CFC"
  showBusyCursor="true" />
```
- `<mx:RemoteObject>` does not actually connect to the CFC or invoke any methods, it simply defines the connection, so that when `svc` is referred to Flex will know the CFC to be accessed

2006 Adobe Systems Incorporated. All Rights Reserved. 14

Using Flash Remoting Results


- Important: ActionScript is case-sensitive!
- The results returned by a Flash remoting call to a CFC method can be accessed as:


```
svc.method.lastResult
```
- In MXML `{ }` is used to delimit an ActionScript expression, much like how `##` is used in CFML
- To use results as a data provider use:


```
dataProvider="{svc.method.lastResult}"
```

2006 Adobe Systems Incorporated. All Rights Reserved. 15

Exercise 4



- Use Flash Remoting to populate a DataGrid with results returned by ColdFusion

2006 Adobe Systems Incorporated. All Rights Reserved. 16


Creating Your Own Flex Components

- Custom Flex components may be created in MXML or ActionScript
- One use of custom components is "renderers", code you write to control how content is presented:
 - Changing the behavior of grid columns
 - Overriding the default display of tree control elements
 - Specifying how TileList items should be displayed
- An MXML custom Flex component is simple a file with the name of the base control as the outermost tag


```
<?xml version="1.0" encoding="utf-8"?>
<mx:VBox xmlns:mx="http://www.adobe.com/2006/mxml">
  <mx:Image source="{data.image}" />
</mx:VBox>
```

2006 Adobe Systems Incorporated. All Rights Reserved. 17

Exercise 5



- Create a custom Flex component to use as a TileList renderer

2006 Adobe Systems Incorporated. All Rights Reserved. 18

The Flex Builder Debugger



- Flex Builder features a sophisticated interactive debugger which supports breakpoints, step-by-step debugging, expression evaluation, and more
- The debugger is used to debug ActionScript (it tends to be far less useful for MXML)
- When running the debugger Flex Builder will switch to the "Flex Debugging" perspective



Exercise 6



- Follow along with your instructor as we explore the Flex Builder debugger



Using States



- States are used to define different views hierarchically.
- States define the differences between views
- Every application has a base state, whether it is used or not
- Defining states creates an MXML block

```
<mx:states>
  <mx:State name="detail">
  </mx:State>
</mx:States>
```
- State are defined in Design View States panel



Exercise 7 (Optional)



- Use States to create an application detail that can be displayed and hidden on demand



Where To Go From Here



- Visit flex.org regularly!
- Visit the Adobe.com Flex Developer Center for more lessons and examples
- Learn about Flex Data Services and ColdFusion / FDS integration
- Have fun!



Better by Adobe.™

