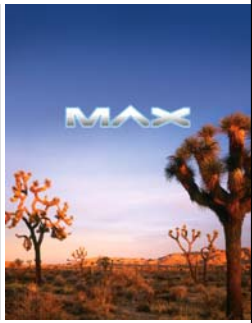
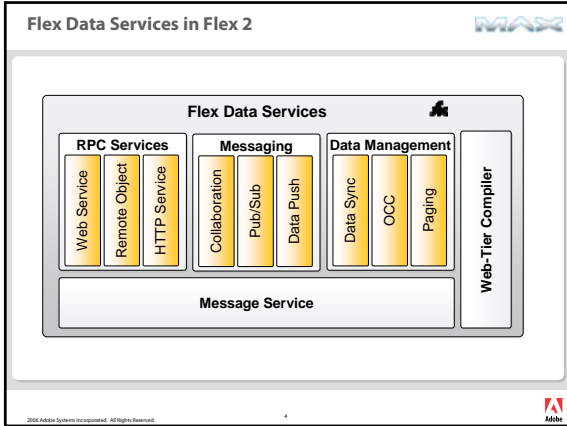


MAX 2006 Using ColdFusion w/ Flex Data Services

Tom Jordahl
Senior Computer Scientist/
ColdFusion Architect
Adobe Systems Incorporated



2006 Adobe Systems Incorporated. All Rights Reserved. Adobe



Introduction

- Architect - Adobe ColdFusion Team
- Implementer of FDS Messaging and Data Management features in ColdFusion 7.0.2

2006 Adobe Systems Incorporated. All Rights Reserved. Adobe

RPC: Flash Remoting Update

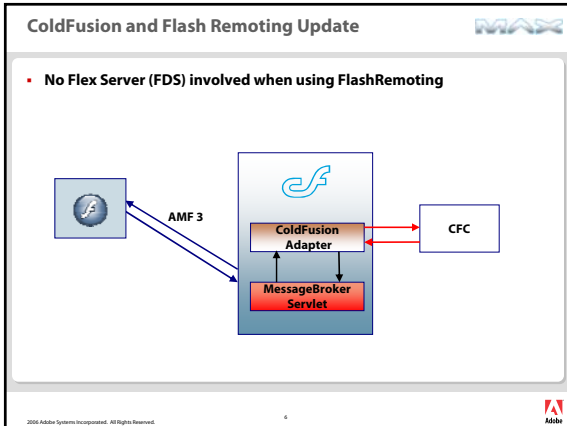
- High Performance Binary Protocol
 - AMF3
- Built in to CF 7.0.2
- No FDS required!
- New features
 - Value Object
 - Data type translations
 - Performance improvements in AMF3
- Attend Mike Nimer's session for more info
 - WD001W - An Introduction to ColdFusion Powered Flex

2006 Adobe Systems Incorporated. All Rights Reserved. Adobe

Topics

- Overview - Flex Data Services
- Messaging
- Data Management

2006 Adobe Systems Incorporated. All Rights Reserved. Adobe



Flex Messaging

- **Publish & Subscribe**
 - Producer and Consumer
- **Asynchronous communication**
- **Messages have header and body**
 - Values can be complex types
 - Values are converted to/from CFML and Actionscript types
- **ColdFusion can be both a Producer and Consumer**
 - Publish – sendGatewayMessage()
 - Subscribe – Event Gateway CFC

Flex Messaging – Connection to CF

- **Event Gateway Adapter registers itself in the RMI registry**
- **Event Gateway Name is unique key (gw1)**
- **Flex Messaging Gateway also registers in the RMI registry**
- **Destination name is unique key (destination1)**

Flex 2 Messaging: Client-Initiated Message

Flex Messaging – FDS Configuration

- **Flex uses Destinations as a basic configuration concept**
 - There are destinations for RPC, Messaging and Data Management
- **Messaging configuration is found in messaging-config.xml**
 - C:\fds2\run4\servers\samples\flex\WEB-INF\flex\messaging-config.xml
- **Clients publish to or consume from a destination**
- **Find the "ColdFusion" destination in Flex sample configuration directory**
 - C:\fds2\resources\config\messaging-config.xml

Flex 2 Messaging: Push

Flex Messaging – FDS Configuration

- **Destination must use the ColdFusion Adapter**
 - `<adapter-definition id="cfgateway" class="coldfusion.flex.CFEventGatewayAdapter"/>`
- **Destination must specify the CF Event Gateway ID**
 - "*" means to look in the message for the target Event Gateway
- **Sample Destination:**

```

<destination id="ColdFusionGateway">
  <adapter ref="cfgateway" />
  <properties>
    <gatewayid>*/gatewayid
  </properties>
  <channels>
    <channel ref="cf-dataservice-rtmp"/>
  </channels>
</destination>

```

Flex Messaging – FDS Configuration

- Other configuration properties:
 - gatewayhost
 - Hostname or IP address of the ColdFusion server
 - allowedIPs
 - IP addresses (no hostnames!) allowed to connect to this destination
 - remote-username/remote-password
 - Credentials to pass along to the gateway, if not already specified in Message
 - Use `setRemoteCredentials()` API on the client instead
- Other Flex Messaging destination properties
 - Network: session timeout, message throttling, etc
 - Server: time to live, durable, cache size, etc.

Flex Messaging with CF

- Example – send message from CFML
- Example – receive message in CFC

Flex Messaging – Security

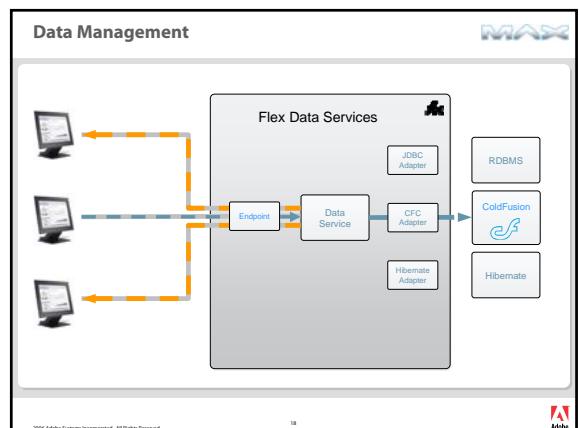
- By default, both Flex and CF sides will only accept connections from the same machine
- You can configure a list of allowed IP addresses for both sides
 - Gateway configuration file
 - Flex Destination
- If IP addresses don't match, "Permission Denied" is returned.
 - No details for security reasons
- Possible to have one side configured differently from the other!

Flex Data Management

- Manages distributed data in Flex Applications
- Data Replication & Synchronization
 - Data provided to clients
 - Changes reflected on all clients
 - Server manages changes to the underlying data resource
- Manage large collections of data
- Nested data relationships
 - One-to-one
 - One-to-many
- Occasionally Connected Clients

Flex Messaging – ColdFusion Configuration

- Configure a new ColdFusion Event Gateway
- Configuration file is optional
- Simple property file
 - Destination – Which Flex destination to send messages
 - Host – Where is Flex running?
 - allowedIPs – Which Flex machines are allowed to talk this gateway
 - Actionsript conversion options:
 - Force-cfc-lowercase
 - Force-query-lowercase
 - Force-struct-lowercase



Data Management – Connection to CF

The diagram illustrates the architecture for connecting Flex to ColdFusion for data management. On the left, a Flex application icon is connected to a box labeled 'Flex' containing a 'CF Data Service Adapter'. This adapter is connected via an 'RMI' link to a 'ColdFusion' box containing a 'Flex Assembler Service'. The 'Flex Assembler Service' is connected to a 'CFC' (ColdFusion Component) icon. Below the 'Flex Assembler Service' is a cylinder representing the 'RMI Registry' with the path 'rmi://cfassembler/identity'.

- CF Data Service Adapter registers itself in the RMI registry
- Identity as configured in the ColdFusion Administrator

Configuring FDS Data Management

- Define the ColdFusion adapter
 - `<adapter-definition id="coldfusion-dao" class="coldfusion.flex.CFDataServicesAdapter"/>`
- Define a destination:


```
<destination id="cfemployee">
  <adapter ref="coldfusion-dao"/>
  <properties>
    <component>samples.crm.EmployeeAssembler</component>
    <scope>application</scope>
    <metadata>
      <identity property="employeeid"/>
    </metadata>
  </properties>
</destination>
```

Configuring FDS Data Management

- Must define channels
 - Channels are how Flex communicates to FDS
- Flex configuration file: services-config.xml
 - C:\fds2\run4\servers\default\samples\WEB-INF\flex\services-config.xml
- ColdFusion requires a special flag on its channels


```
<!-- CF RTMP Channel -->
<channel-definition id="cf-datSERVICE-rtmp" class="...>
  <endpoint uri="rtmp://[server.name]:2048" class="...>
  <properties>
    <idle-timeout-minutes>20</idle-timeout-minutes>
    <serialization>
      <instantiate-types>false</instantiate-types>
    </serialization>
  </properties>
</channel-definition>
```

Configuring FDS Data Management

- Specify special ColdFusion channels
 - cf-datSERVICE-rtmp, cf-polling-amf
- Specify the CFC used as the Assembler
 - Use dot notation or path
- Specify the scope
 - Application – Single CFC instance created and cached
 - Request – New CFC created for each operation

Configuring FDS Data Management

- Again we use destinations
- Configuration file: data-management-config.xml
 - C:\fds2\run4\servers\default\samples\WEB-INF\flex\data-management-config.xml
- Use the ColdFusion examples:
 - C:\fds2\resources\config\data-management-config.xml

ColdFusion Configuration for Data Management

- Enable Flex Data Management support
 - Make sure the service is enabled in the ColdFusion Administrator
- Identity
 - Used when you have more than one ColdFusion instance on a machine
- SSL encryption
 - The RMI connection between ColdFusion and Flex can be protected by SSL
 - You must configure a certificate in to ColdFusion (the "server" in this case)
 - You can use the Java "keytool" program to generate a certificate

The diagram is similar to the first slide, showing the connection between Flex, CF Data Service Adapter, ColdFusion Flex Assembler Service, and CFC. The 'RMI' link between the adapter and the service is circled in red and labeled 'RMI over SSL'.

Data Management - Assemblers

- The code that actually manages the data is called an "Assembler"
- Assemblers can be written as ColdFusion Components
- A basic assembler has four methods:
 - Fill
 - Get
 - Sync
 - Count

25

Flex Data Management with CF

- Example – Contact application
- Example – FDS Sample application

26

CFC Assemblers

- Fill**
 - Returns a data set when called with optional parameters
 - Data is returned as an array of CFCs
- Get**
 - Returns a single item (row) when passed in an identity value
- Sync**
 - Takes a list of changes and applies them to the data
- Count**
 - Returns the number of records that fill would return

26

Special considerations for CF destinations

- `<auto-refresh>`
 - Determines if FDS will re-execute a fill on updates or creates
- `<ordered>`
 - Determines whether the order of the data is important for this filled collection
 - Allows for performance optimization when order doesn't matter.
- `<hostname>`
 - The name or IP of the ColdFusion server
- `<identity>`
 - Which ColdFusion instance to talk to on a machine

27

Creating an Assembler CFC

- Show Eclipse Wizard

27

CFC Assemblers – optional function

- Flex maintains the state of all the fill functions for all clients
- When a change happens, it does an automatic refresh of each
 - Can be very expensive!
- fillContains method allows the assembler to indicate if a fill needs to be updated
 - ```


 <fill-method>
 <use-fill-contains>true</use-fill-contains>
 <auto-refresh>true</auto-refresh>
 <ordered>false</ordered>
 </fill-method>


```
- Define the method like this
  - ```

                    <cffunction name="fillContains" returnType="boolean" access="remote">
                    <cfargument name="fillArgs" type="array" required="yes">
                    <cfargument name="item" type="[CFC type object]" required="yes">
                    <cfargument name="isCreate" type="boolean" required="yes">
                
```

28


Questions? MAX



© 2006 Adobe Systems Incorporated. All Rights Reserved. 31 

Resources MAX

- Flex coders yahoo group
 - <http://groups.yahoo.com/group/flexcoders/>
- ColdFusion 7.0.2 Documentation
 - http://download.macromedia.com/pub/documentation/en/flex/2/using_cf_with_flex2.pdf
 - http://www.adobe.com/support/documentation/en/coldfusion/mx702/d702_releasenotes.html
- Flex Documentation
 - <http://livedocs.macromedia.com/flex/2/>

© 2006 Adobe Systems Incorporated. All Rights Reserved. 32 

Better by Adobe.™

© 2006 Adobe Systems Incorporated. All Rights Reserved. 33 