Going From CF to Java
(moving between the two)
CF2001 Odyssey Conference
July 2001
Charles Arehart
Founder/CTO, Systemanage
carehart@systemanage.com

Agenda
• Considering Java as an Alternative
• Integrating CF and Java Today
• Future Integration Possibilities for CF’ers
• JRun and Java Application Servers
• Software Needed
• Learning More
• Q & A

Audience
• Intended primarily for those with:
  – ColdFusion experience
  – Aware of Java and its value to application developers
    • Though perhaps no java experience
• Anyone interested in:
  – Learning about Java integrating ColdFusion & Java
  – Extending ColdFusion capabilities via Java
**Our Goal**

- **Goals**
  - Put Java in perspective for CF developers
  - Get a flavor for opportunities, challenges
  - Suggest ways to integrate the two today
  - Muse on where the future may lead for us
  - Point to resources for learning more

**Who Am I?**

- **A programmer by trade, trainer at heart**
  - with nearly 20 years IT experience
    - 4 years ColdFusion, 1 year server-side Java
  - Monthly contributor to ColdFusion Developer Journal, bi-monthly column beginning soon in Java Developer Journal
  - Speaker at user groups, conferences worldwide
  - Macromedia Certified Advanced Developer/Instructor
- **Best practices, performance tuning consultant**
  - Contract mentor to development teams

**Considering Java as an Alternative**

- **Java: it’s here & it’s big**
- **May have been feeling pull**
  - May have been able to “fight” getting into it
  - Time has come to face the truth
    - A large market is demanding Enterprise java solutions
    - Also, many tools coming out on the market, built in Java
- **Our focus is on server-side java**
  - Not client-side
- **Microsoft’s pulling JVM out of Future Windows Operating Environments**
  - Is it a threat to Java’s future?
    - May have more effect on client side
    - Not that big a deal for server-side developers
Considering Java as an Alternative

• Not really about caving in
  – Learning how and when to best leverage Java as a CF developer
  • Not about leaving CF for Java
  • More about how to best marry the two
  – Getting easier all the time
• Not even necessarily about learning java
  – Just when and how to integrate java tools
  • Though certainly, to do much, will need to learn java
• Free to get started
  – Most platform tools are free, for developer level deployment

Why Bother?

• Benefits for current projects
  – Leveraging java library functions for features
  – Accessing existing apps/modules within enterprise
    • Whether written in java (as classes, EJB’s, Servlets)
    • Or made available by way of it (CORBA)
  – Creating new classes to extend functionality
• Benefits as a professional developer or development firm
  – Many tools available, written in java
  • Saves you having to build yourself
  • May be packaged for use with JSP/Servlets
  – Can easily figure out how to convert for use in CF

What Can Java Give Us?

• Among the things Java can offer CF developers include:
  – Stronger string manipulation
  – XML manipulation
  – Networking
  – Compression
  – Encryption
  – Graphics generation
  – Image creation and manipulation
  – PDF creation and formatting
• Enterprise integration
Macromedia Tag Gallery

- Macromedia Tag Gallery and Sun site offer downloadable examples of such things
  - Separate “java” link on left in taggallery
  - 41 things based on search of “java” at taggallery
    - Search criteria used
      - java not applet not “java script” not “no java” not “java tree”
      - ignore applets, other extraneous things, not relevant to this topic
    - Examples:
      - CFX_zipbrowser
      - CFX_html2pdf

Going Outside of CF Entirely

- Entire other discussion possible about leaving CF entirely
  - Java Servlets
  - JavaServer Pages
  - Enterprise JavaBeans
  - And more

- Will discuss this more later
  - Focus for now is on integrating CF and Java

Integrating CF and Java Today

- COBJECT for Java
  - Used to call upon java classes available on the same server running CF
    - leverage java language
    - access existing tools, business objects, including EJB’s

- CFX_Java Custom tags
  - Used to call upon Java classes on same server
    - leverage CF integration within the Java class
    - accessing/updating CF variables, queries, etc.
    - call existing CFX_ tags in Macromedia Taggallery

- CFServlet
  - Used to integrate with Servlets on same or other server
    - Either obtaining the output of the servlet
      - pass data to/from servlet
Ways To Use Java

- To call existing things
  - Native Java language offering richer environment
  - Classes, Servlets, CFX_custom tags you or someone else has built
    - Can be a class that produces a result (like a subroutine)
    - Or perhaps a JavaBean
    - Or one whose methods, properties you manipulate within CF
  - Tools that have java API
    - Such as charting tools, PDF creation tools, etc.
    - May choose to use API rather than some custom tag
  - Or ...

Ways To Use Java

- To call existing things (cont.)
  - EJB's
    - Those built within company to support applications
    - Tools that have EJB interfaces
      - (NetCharts EJB, for example)
  - Or to create any of these new, for your apps

Installation Support for Java Integration

- To leverage CFOBJECT for Java and CFX
  - You must have 4.5.1 of ColdFusion
  - Must have Java Runtime Environment 1.2
  - Must configure CF server Admin to setup classpath and other parameters
    - Critical points are
      - JVM path: where JVM is installed (and jvm's name)
      - Classpath: where your classes will be stored
      - CFX_jar path: where cfx.jar used by CFX’s will be stored
Installation Support for Java Integration

- See “A Cold Cup O’ Joe” Part 1
  - Jan 01 CFDJ
  - Great discussion of setup issues, challenges
- See later slide “Software Needed”
  - how to obtain software, versions, etc.

CFOBJECT for Java

- New support in 4.5 for CFOBJECT
  Type="Java"
  - Also createobject() function for use in CFSCRIPT
    ```
    <cfscript>
    classref = createobject("java","classname");
    classref.methodname(parameters);
    </cfscript>
    ```
  - Allows call to java objects
    - Either native java language objects or custom ones
    - Just need to know methods and attributes

Calling Java for Extended Functionality

- If there’s something Java can do, call it
- Example:
  ```
  <!--- // Create a java.util.GregorianCalendar Object // --->
  <CFOBJECT ACTION="CREATE" TYPE="Java" CLASS="java.util.GregorianCalendar"
              NAME="myCalendar">...
  </CFOBJECT>

  <!--- Default Constructor is being called implicitly --->
  <!--- Check if 2001 is a leap-year --->
  <CFSET myCalendar = myCalendar.getInstance(2001)>...

  <!--- Retrieve current date --->
  <CFSET theYear = myCalendar.get(myCalendar.YEAR)>...
  <CFSET theMonth = myCalendar.get(myCalendar.MONTH) + 1>
  <CFSET theDay = myCalendar.get(myCalendar.DATE)>...
  <CFOUTPUT>
  <html>
  <body>
  <p>Is 2001 a leap-year? <b>#is2001LeapYear#</b></p>
  <p>Today is the <b>#theMonth# / #theDay# / #theYear#</b></p>
  </body>
  </html>
  </CFOUTPUT>
  ```
  ```
  <!--- // End Example // --->
  ```
Extending with Java (cont)

- Notice that that involved no compilation of any java classes
  - Was leveraging java language from within CF
- See also call_javacalendar.cfm
  - Converted from code offered in a java book
  - Just need to see how to convert to CFOBJECT
  - Invoke methods like functions
  - Get/set properties like normal variable use
- Can only really benefit if you understand Java
  - More to the point, know the various capabilities of the language libraries
  - See http://java.sun.com/j2se/1.3/docs/API/index.html
    - For online documentation of libraries

Calling a Class to Execute Some Tool

- Many tools now provided as java classes
- May be able to call upon them from within CF
  - With no java coding, per se, required
  - An example: PopChart from corda.com
    - Java-based dynamic server-side charts
    - Relatively inexpensive,
    - Also supports CF w/out Java (using URL’s to pass data)
- Just need to see how to call as CFOBJECT
  - Must set classpath in CF admin to wherever that code is stored; restart CF server

Calling a Java Class You Build

- Can treat the opportunity like a subroutine, calling a java class you’ve created
  - May provide business rule functionality
- Need to build and compile the java class
  - Put it somewhere in the defined classpath, or modify the classpath, as defined in the CF admin
  - Need to know how to write java
    - More in a moment
- When creating own Java classes, beware:
  - Must restart CF server after recompile of any class instantiated via CFOBJECT (or change of classpath)
    - Fixed in CF5 with new dynamic class load path
Building Java Classes

- To build your own classes, need to know how to write Java
  - More than just learning language
    - On simple level, must also learn how to create java code
      - Various issues such as class name, case sensitivity
      - Class compiled via javac
      - How to invoke compiler
    - On larger level, should learn Java language
      - Object-oriented programming, inheritance, polymorphism
      - Packages, interfaces, inner classes, etc.
      - Many concepts foreign to procedural programmers
  - Plenty of good books on Java
    - More on "reading list" later

Some Challenges

- For more complex uses
  - Need to understand calling an object’s default constructor or overloaded constructors
    - Does not currently support overload by attribute datatype
    - Available javacast function to control type passed to java
  - “Cold Cup O’Joe” Part 3 discusses much of this
    - April 01 CFDJ
    - Release 5 adds new GetException() function
      - To manage java exceptions from CFOBJECT more effectively

CFX_ Java Custom Tags

- Similar to calling upon objects
  - Class can read/manipulate CF elements
    - Variables, queries
    - See simple example
  - Can also pass it attributes
    - Processed with request.getAttribute in class
    - See simple example

- Must be registered in the CF administrator
  - Then called as <cfx_objectname>
  - See call_cfx_hellocoldfusion.cfm

- May hear of old way of doing this, CFX_J
- Often a CFX will perform better than CFOBJECT
  - Though some have experience to suggest opposite
Java CFX Interface

- 3 steps: compile, define in admin, call
  - In CF5, no longer need to restart server when changed
- Several available methods, attributes for developing Java Custom Tags
  - Query interface
  - Request interface
  - Response interface
- See “Developing Web Apps” for details
  - Chapter 18

JRun and Java Application Servers

- JRun is a Java Application Server
  - And a very well respected one, at that
  - Reasonably priced, free developer edition
- Others include
  - IBM Websphere
  - BEA Weblogic
  - Apache TomCat
- Enables execution of Java Server Pages, Servlets
  - JRun also includes features to run EJBs, and more
- JSP’s ease entry into Java world

Why Consider JSP/Servlets?

- Lots more clients clamoring for JSP/Servlets than CF
  - Still plenty of market for CF, just larger for JSP/Servlets
- “Learn once/Run Anywhere”
  - Skills learned in working with JRun migrate for the most part to other Java application servers
  - IBM Websphere, BEA Weblogic, Apache TomCat
- What can JSP/Servlets do that CF can’t?
  - On a simple-to-intermediate level, not much
  - More the integration possibilities
  - But now, knowing how CF can integrate Java, not as critical
    - Choose the best platform for each task

Choose the best platform for each task
Integrating CF and JRUN: CFServlet

- Only works with JRun
- Allows call to servlet running under JRun
  - Also allows sharing of data between them
- May hear of old way of doing this: CF_Servlet

EJB's

- Enterprise Java Beans
  - EJB's run in an EJB container (JRun is one such)
  - Provide substantial functionality for managing distributed computing
- Can be called from within CF via CFOBJECT
  - Demonstrated in Macromedia “Developing Applications with JRun”, chapter 42
- Some tools now available as EJB’s
  - NetCharts EJB from visualmining.com

Future integration possibilities for CF’ers

- ColdFusion 6, a.k.a Neo
  - Changing CF from running on a C++ platform to running on a Java platform
  - Does NOT propose to change CF developers into Java developers
    - CF templates will run unchanged
    - Under the cover, will be turned into Java Servlets
  - Will further open integration possibilities
  - More important for Macromedia
    - Allows them to leverage the Java platform
      - And many features that they currently must build from scratch to make CF as powerful as it is
TagServlet from n-ary.com

- Can process a file of CFML on a java server
  - Has more limited tag support than Neo proposes
  - Is improving
- Currently in beta, will eventually be sold
- Called TagFusion for a time

Software Needed

- To do any java integration, need Java Runtime Engine on server
  - Can obtain free from javasoft.com
- To run JRun or do JSP/servlet integration, will need a Java application server
  - Macromedia offers free 3 person developer edition
  - Will need Java 2 EE (J2EE) SDK on top of that
    - Available on install, also free from javasoft.com
  - Other java app servers from IBM, BEA (beasys.com), Apache Tomcat (apache.org)

Software Needed (cont.)

- To do EJB integration, need EJB server
  - JRun Enterprise provides that, as do other products
- ColdFusion 4.5 +
  - added java custom tags and CFObject support for java
- Pre 4.5
  - Cfx_j custom tag
  - Cf_servlet custom tag
## A Reading List

- **Macromedia JRUN documentation**
  - Even without Java experience, they make using understanding JSPs/servlets rather easy
- **Java Developers Journal magazine**
- **JavaServer Pages Application Development**
  - Ben Forta and company
  - Doesn’t presume Java knowledge
- **Beginning Java Objects**
  - Great book for fundamentals of objects/design
- **Other good J2EE Books**
  - Core Servlets and JSP
  - Professional Java Server Programming (J2EE Ed.)
  - Enterprise JavaBeans
- **Other good foundational Java books**
  - Core Java; Beginning Java; Thinking in Java

## Beware of Some Books

- **Servlet/JSP books teach to Java developers**
  - Focus more on showing them how to do web application development (stateless programming, passing data from URL’s and forms, etc.)
  - Presume that reader already knows Java
  - Java Server Pages App Dev (Forta) presumes no java
- **Java books often focus too much on client-side Java (applets, Swing)**
  - Not as relevant to us interested in server-side Java
  - But important to consider for their core Java content

## Learning More

- **Many CFDJ articles**
  - Especially current 8 part “Cold Cup O’ Joe” series by Guy Rish
  - Matching ColdFusion with Server-side Java, by Christian Schneider
  - 3 part “Java for CF’ers” series by Ben Forta
  - 6 part “Online Ticket Store” series by Ajit Sagar
  - 4 parts of which were in Java Dev. Journal
- **Macromedia documentation**
  - CFObject, CFX, custom tags, CFServlet
  - Language Reference and “Developing Web Apps”
  - Administering CF (for admin setup)
Learning More

• Excellent Macromedia JRun documentation
  – Download the Macromedia JRun 3-user demo
  – No time expiration
• Macromedia now offering several Java classes
  – Java for Web Developers (3 day)
  – FastTrack to JSP (2 day)
  – Also coming soon, Building J2EE Apps
  – These all replace former ‘Servlets, Java and JSP’ class
• I’m planning a class on Integrating CF & Java
  – Will expand on many of today’s topics, walkthroughs, tips & tricks, more substantial applications
  – See last slide for contact info

Topics Covered Further in My Forthcoming Class

• My class would cover more, and w/ walkthroughs
  – Actual tools for integrating CF/Java
  – Setting up the environment/challenges
  – Solving nasty classpath problems, compile tricks
  – Some Java programming fundamentals
  – Finding, leveraging other libraries, classes
  – Trying different database integration alternatives
  – Integration with more products/services
    • XML/XSL, PDF, Graphing engines, Servlets etc.
  – Converting Java, Servlet code to CFOBJECT/CFX
• …
* Topics are tentative and subject to change

Topics Covered Further in My Forthcoming Class (cont.)

– Performance implications of calling objects
– Integrating with EJB’s
  • For distributed computing, database calls, more tools
– Comparing Java to other object-oriented CF approaches (Spectra, FuseBox, CFObjects)
– More about Neo, tagservlet
– And more
* Topics are tentative and subject to change
For Further Questions

- For follow-up questions
  - Or more information on Java or ColdFusion consulting, training, or presentations

- Please contact:
  Charlie Arehart, CTO, Systemanage
carehart@systemanage.com
(voice) 301 604 8399
http://www.systemanage.com

- Available for training, mentoring, short-term consulting (1-5 days):
  - All manner of CF topics, beginner-adv
    - Maximizing developer productivity
    - Best practices, performance tuning
  - J2EE (JSP/Servlets/EJBs), Integrating CF/Java
  - Database design, In/Adv SQL topics
  - Testing, Graphing, Source Code Control, and more
  - Wireless applications

www.systemanage.com