2005 ESRI International User Conference
July 2005
San Diego, CA

Rapid Customized Viewer Deployment

Paper UC1962
Mr. Steve Kunitzer
Mr. Todd Bosse

Pacific Disaster Center

590 Lipoa Parkway, Suite 259
Kihei, Maui, Hawaii 96753

http://www.pdc.org - skunitzer@pdc.org
1-808-891-7950 - 1-808-891-0526 (Fax)
An approach extending the functionality of the base ESRI HTML viewer using XML configuration and re-usable JSP/Java/JavaScript components for rapid, customized viewer deployment will be presented. After creating numerous HTML viewer sites and maintaining custom code across applications, the Pacific Disaster Center leveraged the ESRI HTML code and branched it into engine- and template-related components to speed development and deployment and to enhance viewer maintainability.
The methodology used and the benefits gained to application, data, and systems analysts/engineers along with Emergency Managers and other end-users are illustrated through several implemented, in-production sites as well as the viewer spawning/customization process.
Goals and Objectives

- Decrease time to deliver
- Increase reusability
- Increase customization
- Increase consistency
- Increase automation

- Use Java, JSP, XML, and Ant
- Use a modified D. Bollinger TOC
- Packaged, stand-alone WAR
- Deploy to Servlet 2.3, JSP 1.2 (Tomcat 4.1.x)
- Use ArcSDE views and database functions
- Use JDBC authentication/authorization
Benefits

- Create new viewer in hours with complete functionality
- Easy to re-package and re-deploy with new functionality using Ant
- Can customize look-n-feel to customers website color/font scheme
- Can restrict tools/functions per viewer
- Build tools once, use many times
- Learn tools once, use many times
- Leverage power of Java and related technologies including platform independence
• Started with an ArcIMS 4.0.1 HTML viewer when making a site for a County of Hawaii customer
  - Spending way too much time making viewers instead of making viewer tools
  - Spending way too much time integrating new and existing tools into other viewers
• Separated 90% of the business logic and presentation
  - Still a bit of embedded JavaScript output
  - 100% on the way
• Business logic into own CVS module
  – AKA Viewer Engine
  – Contains shared resources
    • aims*.js (aimsXML.js) -> vng-aimsxml.jsp
    • Toolbar images

• Presentation into own CVS module
  – AKA Viewer Template
  – Contains application specifics
    • Frame sources (top, refresh, metadata, print, legend)
    • Map loading and refreshing images
• Unzip template package
• Rename files with project code
  – atlas, atlhi, apnhintsu, mcfire, hcris, …
• Condition files with project code
• Edit language and CSS files
• Edit Table of Contents (next slide)
  – Best to have TOC straw man prior to spawn
• Make new Eclipse project and add Ant build
  – Packages template and engine into a WAR and deploys to Tomcat as a web application with no dependencies

• Time from unzip to deploy
  – No edits = 1 hour
• Modified Dave Bollinger dbGroup TOC code from ArcScripts
  – “Brains” moved to Engine
  – Configuration moved to Template
    • JSP page with JavaScript wrapper
    • Soon to be XML

• Root TOC has:
  – Title, caption, autorefresh, swatch, and metadata

• Groups have (including Group01):
  – Caption, opened, swatch, and metadata

• Layers have:
  – Name, caption, swatch, display, metadata, and tool
Sites in Production

- Hawaii County
- Indian Ocean Tsunami
- World Atlas
- Hawaii Atlas
Other sites

- Maui County FD
- Marikina City
- American Samoa

No link:
- Vietnam Atlas
- Makani Pahili
- Maui County
Further Work

- More transition to Java Connector Object Model and other Java components
- Abstract language constructs to ResourceBundle for translations
- Provide online interfaces to XML configuration for on-the-fly changes
- Provide support for other map servers
- Provide layer authentication/authorization
- Provide services for map and data usage outside a viewer environment
- Support for Java 1.5.x and Servlet 2.4 / JSP 2.0 (Tomcat 5.x)
- Use Ant for automated spawning process
Mr. Steve Kunitzer
Software Engineer
Pacific Disaster Center
590 Lipoa Parkway, Suite 259
Kihei, HI 96753 USA
808.891.0525 x950
skunitzer@pdc.org
http://www.pdc.org