TACCIMO

A CASE STUDY OF MIGRATING TO THE ARCGIS SERVER API FOR FLEX

Todd Pierce, Jeff Hicks, Amber Ramirez, Caroline Dougherty
UNC Asheville's National Environmental Modeling and Analysis Center
Jennifer Moore Myers, Rob Herring, Emyrs Treasure, Steve McNulty, and Chris Liggett
USDA Forest Service, Eastern Forest Environmental Threat Assessment Center

2011 ESRI SERUG

www.forestthreats.org/taccimotool
Overview

Purpose of presentation:
- Describe TACCIMO product
- Explain migration to Flex environment
- Provide lessons learned
TACCIMO (Template for Assessing Climate Change Impacts and Management Options)

- Helps users integrate climate change science into land management planning
- Provides land and resource managers and planners with the best available science they need to effectively and efficiently sustain forests and the services they provide under a changing climate.
TACCIMO was developed through a partnership between the USDA Forest Service’s Southern Research Station and Southern Region Planning divisions.

Initial version released in 2010 as a web-based assessment and reporting tool included:

- **Geospatial explorer** shows climate data maps
- **Geospatial report** provides climate model projections for national forests through 2090
- **Content explorer** displays climate change impacts on forests, along with management options, objectives, and design criteria for creating forest response plans
- **Report wizard** generates custom reports that provide climate change specific impacts and management options
**Geospatial Explorer**: initial version used out-of-the-box ArcGIS Server web application; data in ESRI geodatabases
Initial Version

Content Explorer and Report Wizard: HTML and ASP.NET
Geospatial Report: manually generated reports in MS Word
The initial version was successful, but the TACCIMO development team wanted to bring the various components together into one single web application.

The team also wanted to have a richer web interface using the Adobe Flex/Flash platform.

UNC Asheville’s NEMAC was brought in to migrate TACCIMO to the new platform – and to combine the components into one application (the “integrator”).
Flex Migration

New version combines previous components into one interface.

Demonstration

- GIS data in ESRI geodatabases
- Data served with ESRI ArcGIS Server 10 .NET
- Climate model projection data in SQL Server 2008 database
Flex Migration

Geospatial explorer included in GIS Viewer window

- Flex SDK 3.2
- ESRI ArcGIS Server Flex API 1.3
Geospatial explorer supports charts of climate model projections for clicked map points

- Flex SDK 3.2
- ESRI ArcGIS Server Flex API 1.3
- PHP 5.3.3
- Zend 1.11
Climate Chart window shows climate data projections in tables and charts.

- Flex SDK 3.2
- PHP 5.3.3
- Zend 1.11
Content Explorer available in Science and Planning pane – shows management options for areas

- Flex SDK 3.2
- PHP 5.3.3
- Zend 1.11
Content Explorer available in Science and Planning pane – also shows desired conditions, objectives, design criteria for forests

- Flex SDK 3.2
- PHP 5.3.3
- Zend 1.11
Flex Migration

Menu box lets user switch between states, counties, regions, and US National Forests

Flex SDK 3.2
Flex Migration

Geospatial Report now accessible from menu box

- Flex SDK 3.2
- PHP 5.3.3
- tcpdf 5.9
Geospatial Report generated on-the-fly for selected area and exported to PDF

- PHP 5.3.3
- tcpdf 5.9
- Map images from WMS services in ArcGIS Server
Lessons Learned

Communicate, communicate, communicate

- TACCIMO team in Raleigh, NEMAC in Asheville
- NEMAC team Flex beginners; TACCIMO team Flex newcomers
- Many phone calls and emails
- Four team visits over seven month period
Lessons Learned

Moving from text based code language to compiled language presents some challenges

- Flex uses ActionScript and MXML, both compiled into a SWF file for deployment
- Code can no longer easily be opened in Notepad and edited on the server environment – requires a development environment such as Flex Builder or Flash Builder (Eclipse is open source version)
- Had to coordinate versions of Flex SDK (Software Development Kit), ESRI APIs, and Flex Builder
Lessons Learned

FlexBuilder 3 interface
Connecting Flex to databases requires use of server side language separate from ActionScript and MXML

- For a data driven application, the Flash SWF app (client) must make requests to a database (server)

- First attempts using ColdFusion worked easily, but USFS needed an open source solution

- Final version uses PHP with Zend library to connect PHP to Flex (Zend located at http://framework.zend.com/)
Lessons Learned

Connecting Flex to databases requires use of server side language separate from ActionScript and MXML

- Zend connection required several steps
  - Write PHP queries to database as functions
  - Create a PHP ‘gateway’ file to load Zend and PHP functions
  - Edit services-config.xml in Flex to point to gateway file
  - Add the services XML file into the Flex project compile arguments
  - Set up Remote Object in Flex to connect to the PHP functions through the gateway file
  - Write functions to call the Remote Object and to process results returned from PHP
Lessons Learned

Flex requires a team of programmers with different skills

- Previous projects at NEMAC done by one or two persons
- TACCIMO required a larger team with multiple skills
  - One member specialized in the interface – colors, styles, graphics, transitions, effects
  - One member focused on the GIS-Flex interaction using the ESRI API for Flex
  - One member focused on the database schema and the PHP code needed to query the database
  - One member focused on the Flex-database interactions and Zend for PHP
Lessons Learned

Dedicated team programming efforts make a big difference

- NEMAC staff usually work on several projects at once
- For this project, 2 members were 100% dedicated, and the other 2 members were roughly 75% dedicated

- Led to rapid strides in development; initial prototype completed in just a couple months
- Final version delivered five months after project start

- Such concentration of resources often not possible – has not been done again due to competing project deadlines
Lessons Learned

Large teams need code management software and version control

- Subversion plugin (open source) used for Flex
- Supports code check in/out and change synchronization
- Worked well most of the time – as long as team remembered to commit changes, and to get latest changes when opening code
- Requires setting up a subversion repository
Lessons Learned

Use external configuration file(s) to minimize coding changes

- Similar to approach used by ESRI’s Flex Viewer application
- Have XML configuration file store items that end user might want to change, without need for developers to edit code and recompile SWF file
  - ArcGIS services to load to map
  - Choices for background map (streets, image, terrain, etc)
  - Initial extent of map and bookmarks for quick zooming
  - Hyperlinks to external resources in links bar
  - Combinations of services into map ‘themes’
  - Settings for map tools (print, find, identify, export, etc)
Lessons Learned

Use external configuration file(s) to minimize coding changes
Lessons Learned

Use external configuration file(s) to support portability

- Similar to approach used by ESRI’s Flex Viewer application

- Have configuration file store items likely to change as application is moved across servers, or as data changes
  - Path to ArcGIS Server REST folder (different server names)
  - Location of needed images
  - Connection information for SQL server database
  - Path to export reports and resulting URL
Lessons Learned

Don’t be afraid to hard code things that don’t change, if you get faster performance.

- List of states, counties, forests – not expected to change – store in code file instead of querying database.
Lessons Learned

Don’t be afraid to value speed over disk space

- Maps of climate change projections for areas in geospatial report – data not expected to change – create pre-generated images at needed scale rather than query WMS server on the fly (saved several minutes per report)
- But, took 735 MB of space just for states, regions, forests – counties not done (would add another 9 GB)
Lessons Learned

Transferring working application to another organization requires considerable planning and probably on-site work

- Setting up same environment on server (PHP, ArcGIS Server)
- Moving spatial and SQL databases
- Setting up ArcGIS Services
- Finding hard coded paths and URLs and making generic
- Handing off needed documentation and metadata
- Training on needed skills to keep application running
Conclusion

The TACCIMO product serves as a useful case study of migration to Flex

Many lessons learned being applied to other development projects at NEMAC

tpierce@unca.edu
http://nemac.unca.edu/

jmooremyers@fs.fed.us
www.forestthreats.org/taccimotool