Developing Web Applications with ArcGIS Server

Kevin Deege
Educational Services
ESRI-Washington DC
Introductions

- Who am I?
- Who are you?
  - ESRI Product Development Experience
  - What development languages are you using
  - What types of applications are you developing
Session Overview

- ArcGIS Server Overview
- ArcGIS Server Developer Resources
- Introduce the Web ADF
- Building Web ADF applications
  - Manager, template, controls
- Extending Web ADF applications
  - Custom Tools
  - Custom Tasks
ArcGIS Server 9.2

- Complete & Integrated server-based GIS
- Out-of-the-box applications and services
- Tremendous developer opportunities
Out of the Box Web based GIS

- Allows GIS Analysts to easily configure Web Applications
  - No programming required

- Includes out of the box Desktop Clients
  - ArcGIS Explorer
  - ArcGIS Desktop

- Allows analysts to publish rich GIS Services
  - 2D and 3D Mapping services
  - Geocoding, Geodata, Network Analysis, Geoprocessing services
  - SOAP, WMS, and KML based services
Development Platforms

- **Support for multiple platforms**
  - .Net
  - Java

- **Cross-platform development**
  - Available on Windows, Linux, and Solaris

- **Comprehensive SDKs for Application Development**
  - *Web ADF* – for Web Applications and Web Services
  - *Mobile ADF* – for Mobile Applications (.Net only)
  - *Enterprise ADF* - for Enterprise JavaBeans (EJB) (Java only)
Product Platforms

- **ArcGIS Server for the Microsoft .Net Framework**
  - Windows

- **ArcGIS Server for the Java Platform**
  - Windows, Linux, Solaris

- Platform specific install, documentation
  - SDK and IDE integration

- Both platforms have the same GIS functionality
  - Rich GIS Services
  - Management Applications
  - Web Mapping Applications
  - ArcGIS Explorer and Desktop clients
GIS on the Web in 3 steps

1. **Author GIS content**
   - Create GIS resources
   - ArcGIS Desktop applications

2. **Serve content**
   - Publish GIS resources as services
   - ArcCatalog and ArcGIS Server Manager

3. **Use GIS services**
   - Web mapping applications
   - ArcGIS Explorer
   - ArcGIS Desktop
   - Many others
ArcGIS Server Web Services

- Server Objects exposed as web services
  - Types
    - Map service
    - Geocode service
    - Globe service
    - Geodata service
    - Geoprocessing service
    - More at 9.3 (Image Service)
  - Capabilities
    - Map/Data/Query
    - Geocode/Reverse Geocode
    - 3D Visualization
    - Data Replication
    - Geospatial Analysis
- Consumed by both Java and .NET
ESRI Developer Network (EDN)

Access to the ArcGIS Development Platform

- The Developer Product
  - ArcGIS Server (all Levels and Editions)
  - ArcGIS Image Server
  - ArcIMS
  - ArcGIS Engine
  - ArcGIS Desktop*

- Software Developers Kits
  - .NET and JAVA

- Website for Developers
  - http://edn.esri.com
Developer Resources

ArcGIS Server Development Blog

- http://blogs.esri.com
Introducing the Web ADF

- Libraries used to build GIS Web applications
  - Web controls (ASP.NET, JSF)
    - AJAX enabled
    - Map image blending
  - Supports multiple data sources
    - ArcGIS Server, ArcIMS, ArcWeb, WMS
    - Web ADF graphics
  - Task Framework
    - Extensible Architecture

- ArcGIS Server Manager
  - Build Web Applications

- IDE integration
Java Web ADF Architecture

Java Web ADF

- Web Controls
- Tasks & Framework
- WebContext & Attributes

AGS | IMS | WMS | AWS | GR

Common API

Resource APIs
.NET Web ADF controls

- Resource manager controls
  - Not visible at run time
  - Manage resources
- Mapping controls
  - Map
  - Overview map
- Related controls
  - Toolbar
  - Table of contents

Warning: You must enter a valid value for the ResourceItems property.
Java Web ADF controls

- Set of custom JavaServer Faces Web controls
- Expose server object functionality
  - Context, Map, Overview, Toc, and Toolbar
  - Gives access to other APIs (common or resource specific)
    - com.esri.adf.web.faces.component
- SDK
  - JavaDoc
  - Web control tag library
  - Samples
Web Graphics

- Allow developers to add simple graphics to the map
- Supported symbols
  - Line
  - Marker
  - Polygon
  - True type marker
- Supported renderers
  - Unique value
  - Class break
  - Simple
- Integrated into Web and Browser tiers
Graphics Layers

- Display geometry and text dynamically within a map

Client-tier

Web-tier

Server-tier

Map image
Graphics layer
Map data layers
Web ADF resources

ArcGIS Server
ArcIMS
Web ADF Graphics

ArcGIS Server
ArcIMS

Create Image

Federal User Conference 2008
Web ADF – Tasks

- Allow the end user to perform useful work
- Visual components
  Perform a set of related actions and generate a result
- Configurable from Manager
- Product ships with out of the box tasks
  - Search by attributes
  - Query Attributes
  - Find Place
  - Find Address
  - Editing
  - Geoprocessing
- Developers can extend the system with Custom tasks
Visual Studio 2005 integration

- Web controls
- Web Mapping Application template
- Context menus
- Developer Help
IDE integration - Eclipse

- **Eclipse plug-in**
  - Template applications
  - Reusable code available as snippets
  - Integrated documentation
  - Samples are integrated to run from within the IDE
Building Web Applications

- Use the Web ADF
  - ArcGIS Server for .Net - Web ADF
  - ArcGIS Server for Java – Web ADF

- Leverage AJAX Enabled ASP.Net or JSF Web Controls

- Work with GIS Web Services using the SOAP API

- Work with GIS Server Objects using either
  - The SOAP API
  - Finer Grained ArcObjects API
Web ADF Development Paths

- Increasing complexity and functionality

Developer Paths

I

Web Controls

II

Common Data Source API

III

Data Source Implementations

IV

Data Source (GIS Server) Specific APIs

Federal User Conference 2008
Creating Web ADF applications

1. Use ArcGIS Server Manager
   - Web site builder
   - Modify in Visual Studio, Eclipse

2. Use template
   - Same template used by ArcGIS Server Manager
   - Visual Studio, Eclipse, Creator

3. Create using Web controls
Web Mapping Application template

- Web controls
- Themes
- Resources
- Custom tools
- Help system
Advantages of using the template

- Start with a complete GIS Web application
- The template provides:
  - Measure user control
  - Identify tool
  - Layout DIVs – resize, etc.
  - Code for docking items, closing application
  - Themes
  - Help system
.NET Map control

- Associate with a Map resource manager control
- Works with multiple map resources
  - Create dynamic map images
  - Fetch pre-generated image tiles (cached services)
- Image blending
  - Web tier or browser
- Game-style navigation
  - Seamless pan and zoom
  - Keyboard and mouse integration
- AJAX enabled!
.NET Toolbar control

- Work with Map control using callbacks
  - Write server-side code only
  - Client-side code provided by the Web ADF

- Out-of-the-box map actions:
  - Zoom In
  - Zoom Out
  - Pan
  - Full Extent

- Tools and Commands configurable in Visual Studio design-time dialog
# Java ADF - MVC architecture

<table>
<thead>
<tr>
<th>View</th>
<th>Controller</th>
<th>Model 1</th>
<th>Model 2</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;a:context../&gt;</code></td>
<td>ContextControl</td>
<td>WebContext</td>
<td>GISResource</td>
</tr>
<tr>
<td><code>&lt;a:map../&gt;</code></td>
<td>MapControl</td>
<td>WebMap</td>
<td>Map Functionality</td>
</tr>
<tr>
<td><code>&lt;a:overview../&gt;</code></td>
<td>OverviewControl</td>
<td>WebOverview</td>
<td>OverviewFunctionality</td>
</tr>
</tbody>
</table>

```
faces-config.xml
```

```
context-attributes.xml
```

```
***-functionalities.xml
```
The Context control

- Establishes a connection to the GIS Server (non-visual)
- Container of all attributes and resources
- Has attributes registered to it
  - Notifies attributes about changes in application state
  - `WebContext.refresh()`
- Stored in session scope
  - Data objects can be retrieved from context
- Works with a `WebContext` object
The Map control

- AJAX-enabled
- Works with multiple map resources
  - Creates dynamic map images
  - Supports image caching
- Map service blending
  - Web tier or browser
- Required attributes
  - id
  - value

```
<map id="map1"
    value="#{mapContext.webMap}" />
```
<%@taglib uri="http://java.sun.com/jsf/core" prefix="f"%>
<%@taglib uri="http://java.sun.com/jsf/html" prefix="h"%>
<%@taglib uri="http://www.esri.com/adf/web" prefix="a"%>

<f:view> <h:form id="f">
  <a:context value="#{mapContext}" />
  <table> <tr> <td>
    <a:map id="map1" value="#{mapContext.webMap}" width="600" height="600"/>
  </td><td>
    <a:toc mapId="map1" value="#{mapContext.webToc}" style="height:200" />
    <a:overview mapId="map1" value="#{mapContext.webOverview}" />
  </td>
  </tr> </table>
</h:form>
</f:view>
.NET - Creating a custom tool

To create a tool/command:

1. Write server-side code to handle map action
2. Add a new toolbar item to toolbar.
3. Set server action
4. For tools, choose client action (point, rectangle, etc.)
5. Set images
Common Data Source API benefits

- Removes business and GIS logic from Web controls
- Support for multiple data sources using the same API
- Easy to program against different data sources
- Implement your own custom data sources
Data source-specific APIs

- Each data source a different set of capabilities
  - ArcGIS Server
    - SOAP, ArcObjects
  - ArcIMS
    - ArcXML
  - ArcWeb Services
    - SOAP

- What does this mean?
  - Many data source-specific classes available
  - More business/GIS logic
  - Different APIs may use different communication protocols
  - Requires different programming patterns for each data source
ArcGIS Server APIs

- **SOAP**
  - Available for services and server object extensions
  - Designed for stateless interaction

- **ArcObjects**
  - Available for Local ArcGIS Server services
  - Designed for stateless and stateful interaction
Web ADF Tasks

- **Visual components**
  - perform a set of related actions
  - generate results

- **Encapsulate a workflow**
  - UI for gathering input
  - Perform some action based on inputs
  - Generate results
Web ADF Tasks

- Configurable from Manager
- Out of the box tasks
  - Navigation
  - Geoprocessing
  - Search by attributes
  - Editing
  - Find direction
  - Predefined query
  - Find place
  - Print Task (9.3)
- Custom tasks
The Task Framework (.NET)

- Framework provides for:
  - Configuring tasks in VS.NET and Manager
  - Organization of tasks
  - Feedback when a task is executing
  - Management of results
    - Display
    - Visualization on Map
    - Delete, Refresh, Re-run
  - Distributable UI components
    - Plugs into Manager

- Extensible
.NET Web Custom Tasks

- Basic task implements ITask
- Create a custom Web control
  - Extend Task or FloatingPanelTask abstract base classes

```
ESRI.ArcGIS.ADF.Web.UI.WebControls
FloatingPanelTask
CustomTask
ICallbackEventHandler
IBuddyControlSupport
ITask

ESRI.ArcGIS.ADF.Web.UI.WebControls
FloatingPanel

ASP.NET Web Controls
```
Java Custom Web Tasks

- **Parameters**
  - Inputs needed by the task
  - E.g. Layer name, zoom factor, etc ...

- **Commands (Actions)**
  - Business logic Is executed without user interaction with the map
  - E.g. Zoom to full extent

- **Tools**
  - User interaction with the map is required
  - Client-side action is required
  - E.g. Identify, dynamic navigation
Java Custom Task Implementation

1. Create a standard Java Class
   - Signature of the method will dictate command or tool

2. Register the Java class as a managed bean
   - Faces.config
   - Make it available in the ADF

3. Add the task tag to the jsp
   - Reference the managed bean
   - To visually display the UI to the user
Implement a custom task (1)

1. Create a standard Java Class
   ```java
   public class MyTask {
   }
   ```

2. Register Java class as a managed bean
   ```xml
   <managed-bean>
       <managed-bean-name>myTask</managed-bean-name>
       <managed-bean-class>myPackage.MyTask</managed-bean-class>
       <managed-bean-scope>session</managed-bean-scope>
   </managed-bean>
   ```
Implement a custom task (2)

3 Add the control to the jsp

- Value attribute = value of managed-bean-name in faces-config

```jsp
<a:task value="#{myTask}" mapId="map1" />
```

```xml
<managed-bean>
  <managed-bean-name>myTask</managed-bean-name>
</managed-bean>
```
Questions?
Thank You