ArcGIS for Developers

Kevin Deege
Educational Services
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

- The ArcGIS System
- ArcGIS Developer Resources
- ArcGIS Desktop
  - Application Framework
  - Extend with custom components
- ArcGIS Engine
  - Application Framework
  - Visual and console applications
  - Extend with custom components
- ArcGIS Server
  - Web Application Developer Framework (ADF)
ArcGIS and ArcObjects

ArcGIS is built on ArcObjects
- Software components

ESRI uses ArcObjects to:
- develop the software and applications

Developers use ArcObjects to:
- customize the software
- build custom applications

ArcObjects is the core of ArcGIS
ArcGIS products share ArcObjects

ArcGIS Desktop

ArcGIS Engine

ArcGIS Server
ArcObjects APIs

- ArcObjects components are C++ objects
- Accessible through different APIs
  - COM, .NET, Java and C++
- Native interface is COM
The ArcGIS System

Supported ArcGIS Product APIs

- Desktop APIs (COM and .NET)
  - VBA (customize documents using MXDs and MXTs)
  - Custom components (commands, tools, windows, extensions)

- Engine APIs (COM, .NET, Java, and C++)
  - Build custom standalone applications
  - Embed into existing applications

- Server APIs (.NET, Java)
  - Build and deploy Web services and Web applications
ArcGIS Developer Resources

ArcGIS Developer Kit

- Help resources and tools to build ArcGIS applications
  - ArcObjects Help System
  - Developer documentation
  - Productivity tools and utilities
  - Object Model diagrams
  - Sample code
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
The ArcGIS System Overview

Desktop GIS
- ArcGIS Desktop
  - ArcInfo
  - ArcEditor
  - ArcView
- Desktop Developer Kit
  - .NET
  - COM

Embedded GIS
- ArcGIS Engine
  - Engine Developer Kit
  - .NET
  - C++
  - COM
  - Java
- ArcObjects

Server GIS
- ArcGIS Server
  - Java
  - ADF
  - .NET
  - ADF
  - COM
- Server Developer Kit

ArcSDE
- Geodatabase
  - File-based
  - DBMS
  - XML
ArcGIS Desktop Framework

The Basics

- **ArcGIS Desktop Applications**
  - ArcCatalog
  - ArcMap
  - ArcScene
  - ArcGlobe

- **Generic common framework**
  - Extensible and customizable
  - Documents and templates (MXD, MXT, etc)

- **Customization options**
  - Customize Dialog
  - Visual Basic for Applications (VBA)
  - Custom components (COM, .NET)
ArcGIS Desktop Framework

Framework Components

Visual Components (extending the user interface)

1) Commands and Tools
2) Menus and Toolbars
3) Embedded Windows
   • Dockable Windows, Contents Views, etc.

Non-visual Components

1) Application Extensions
2) Undo/Redo Operations
3) Command keyboard shortcuts
ArcGIS Desktop Framework

Developing custom components

- Create a COM object and plug it into an application

Steps
- 1. Create a COM/.NET project
- 2. Create a COM class
- 3. Reference the ArcGIS libraries
- 4. Implement an interface
- 5. Compile
- 6. Register in a component category
Useful development tools

- New Project templates
- Base classes
- Add Class wizard
- Code Snippets
- Quickly adding Imports/Using statements
.Net new project templates

- Available when new VS.Net project is started
- ArcGIS Project Wizard - select ArcGIS References
.Net base classes

- Inherit from commonly used interfaces
- Less implementation for users to code

```vbnet
Public NotInheritable Class RouteFinderCmd
    Inherits BaseCommand

    Public Sub New()
        MyBase.New()
        MyBase.m_caption = "RouteWindow VB"
        MyBase.m_category = "ArcObjects .NET Tools"
        MyBase.m_message = "Toggles view for the RouteFinder window"
        MyBase.m_name = "ArcObjects .NET Tools_RouteFinderCmd"
        MyBase.m_toolTip = "Displays or hides Route Finder"
        MyBase.m_bitmap = New System.Drawing.Bitmap
            (Me.GetType.Assembly.GetManifestResourceStream("ESRI.ArcObjects.AAON.RouteFinderVB.RouteFinder.bmp"))
    End Sub

    Public Overrides Sub OnCreate(ByVal hook As Object)
        ...
    End Sub

    Public Overrides Sub OnClick()
        ...
    End Sub
```

Federal User Conference 2008
How does ArcMap know to use my class?

- **ArcMap application start up cycle starts**
  1. Accesses the appropriate component category
Application start up cycle

- **ArcMap starts**
  1. Accesses the appropriate component category
  2. Creates an internal objects that implements ICommand

ESRI Mx Commands
ESRI Mx CommandBars
ArcMap starts

1. Accesses the appropriate component category
2. Creates an internal objects that implements ICommand
3. Creates your UI component (command, tool, toolbar, or menu)
Application start up cycle

1. Accesses the appropriate component category
2. Creates an internal objects that implements ICommand
3. Creates your UI component (command, tool, toolbar, or menu)
4. Adds the CommandItem to the CommandItem list
Register in a component category

- Manual registration
  - ArcMap Customize dialog box
- Add from file (.tlb)
  - CategoryManager.exe
- Developer tools

```csharp
#Region "ArcGIS Component Category Registrar generated code"
    Private Shared Sub ArcGISCategoryRegistration(ByVal registerType As Type)
        Dim regKey As String = String.Format("HKEY_CLASSES_ROOT\CLSID\{{0}}", registerType.GUID)
        MxCommands.Register(regKey)
    End Sub
    Private Shared Sub ArcGISCategoryUnregistration(ByVal registerType As Type)
        Dim regKey As String = String.Format("HKEY_CLASSES_ROOT\CLSID\{{0}}", registerType.GUID)
        MxCommands.Unregister(regKey)
    End Sub
#End Region
```
Using component categories

- Folders in the registry
- Desktop Component Categories
  - ESRI MxCommands
  - ESRI MxCommandBars
  - ESRI Mx Extensions
- View component categories with Categories.exe
- Engine Component Categories
  - ESRI Controls Commands
  - ESRI Controls Menus
  - ESRI Controls Toolbars
  - ESRI Controls Palettes

- Can register your components in these categories
The ArcGIS System Overview

ArcGIS Desktop
ArcGIS Server
ArcGIS Engine
Desktop GIS Embedded GIS Server GIS
Geodatabase DBMS File-based XML ArcSDE
ArcEditor ArcView ArcInfo ArcView
ArcObjects Engine Developer Kit Server
Desktop Developer Kit Embedded GIS Server Developer Kit
.NET COM .NET COM .NET ADF ADF COM
.NET C++ COM Java Java ADF .NET ADF COM
ArcObjects
ArcSDE
Geodatabase
File-based DBMS XML

Federal User Conference 2008
ArcGIS Engine for Developers

Overview

- Consist of two products:
  - ArcGIS Engine Developer Kit
    - For development use only
  - ArcGIS Engine Runtime
    - The platform for custom solutions
    - Included with the ArcGIS Desktop Install

- Designed for easy:
  - Development
  - Testing
  - Deployment
The available extensions for ArcGIS Engine Runtime are

- Spatial extension
- 3D extension
- Geodatabase Update extension
- Network extension
- Data Interoperability
- Schematics
- Maplex
- Tracking
ArcGIS Engine Applications

- GUI-based - visual
- Console
- Embedded applications
Engine Controls

- MapControl
- PageLayoutControl
- ToolbarControl
- TOCControl
- LicenseControl
- SceneControl
- GlobeControl
- SymbologyControl
Building ArcGIS Engine applications

1. Start with the IDE integration tools

2. License the application

3. Add custom buttons and tools as necessary

4. Use code snippets where possible
Engine application licensing

- An Engine application runs with:
  - Existing ArcGIS Desktop 9 license
  - ArcGIS Engine Runtime license

- Application Developer has control over what license is required to run an application
ArcGIS Product Licensing

- Engine and Desktop are functionally similar
- Two levels of licensing
  - Product
    - ArcGIS Desktop
    - Engine standard
    - Engine GDB Update
  - Extension
    - Spatial, 3D, Network, etc.
Engine customizations

- A number of different development options...
- Similar customizations to ArcGIS Desktop
The ArcGIS System

Overview

Desktop GIS
- ArcGIS Desktop
  - ArcInfo
  - ArcEditor
  - ArcView
- Desktop Developer Kit
  - .NET
  - COM

Embedded GIS
- ArcGIS Engine
  - Engine Developer Kit
  - .NET
  - C++
  - COM
  - Java

Server GIS
- ArcGIS Server
  - Server Developer Kit
  - Java
  - ADF
  - .NET
  - ADF
  - COM

ArcObjects

ArcSDE

Geodatabase
- File-based
- DBMS
- XML

Federal User Conference 2008
ArcGIS 9.2 ADF Highlights

- **ArcGIS Server Manager**
  - Create Services, Manage servers
  - Build Web Applications

- **Supports multiple data sources**
  - ArcGIS Server, ArcIMS, ArcWeb, WMS custom, etc.

- **Multi-source controls**
  - Map image blending (browser, web tier)
  - AJAX enabled (cache tile retrieval, map refresh)
  - Seamless navigation

- **Task Framework**

- **Object oriented, AJAX-enabled JavaScript library**

- **IDE Integration**
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
Server Development Guidelines:

- Use the Web ADF
  - Create a Web Application that consumes GIS Services
  - Resource Management Controls
    - Map, Geocode, Geoprocessing

- Leverage AJAX Enabled ASP.Net or JSF Web Controls
  - Extend with Custom Tools

- Work with GIS Web Services using the SOAP API

- Work with GIS Server Objects using either
  - The SOAP API
  - Finer Grained ArcObjects API
Increasing complexity and functionality

Web Controls

Common Data Source API

Data Source Implementations

Data Source (GIS Server) Specific APIs

Developer Paths

I

II

III

IV
Creating Web ADF applications

1. Use ArcGIS Server Manager
   - Web site builder
   - Modify in Visual Studio 2005

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

3. Create using Web controls
Common Data Source API

Resource Managers
- IGISDataSource
- IGISResource
  - IMapResource
  - IGeocodeResource
  - IGeoprocessingResource

Web ADF Controls
- IGISFunctionality
  - IMapFunctionality
  - ITileFunctionality
  - IMapTocFunctionality
  - IGeocodeFunctionality
  - IGeoprocessingFunctionality
  - IQueryFunctionality

ArcGIS Server
ArcIMS

Implementations

Generic

Federal User Conference 2008
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
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
The ArcGIS Server SOAP API

- Composed of a number of value and proxy objects
- Objects work with both Internet and local resources
- Value objects
  - Geometry, symbology, query filters, spatial reference…
- Proxy objects
  - Emulate functionality provided by coarse-grained server objects (MapServer, GeocodeServer)
Value and Proxy objects?

- **Value objects:** native .NET classes
- **Use proxies to communicate with server end points**
- **Proxy objects perform two main tasks:**
  - Serializing Value objects to SOAP that is sent to resource
  - Deserializing SOAP responses to Value objects for client
Using a SOAP service

- Work with raw SOAP XML (rare)
  - or -
- Create and utilize Value objects and proxies
  - Generated from a WSDL using a SOAP toolkit
  - Value objects “model” server objects
  - Proxies pass information from value objects to web service
Web ADF implementation

- ArcGIS Server MapResource types:
  - MapResourceInternet – Connect to a Web service endpoint
  - MapResourceLocal – Connect to the Server Object Manager
    - Access ServerContext
    - Use ArcObjects

- Each ArcGIS Server service type provides:
  - Web Services Description Language (WSDL)
    - `<ArcGIS Install>\XMLSchema`
  - Web service proxy
  - Distributed Component Object Model (DCOM) proxy

- Value objects are shared for different service types
Using ArcObjects

- **COM utility objects**
  - Initialized when required by the GIS server
  - May happen several times
  - Not registered with a specific server object instance
  - Created “ad-hoc” using the server context

- **Server object extensions**
  - ArcGIS Server 9.2
  - Initialized once during server object startup
  - Can benefit from caching logic
  - Registered with specific server objects (Map Services)
  - Configurable in ArcCatalog through custom property page
Where do you go from here?

- **Instructor Led Training**
  - Desktop:
    - Introduction to Programming ArcObjects (VBA, .NET, JAVA)
    - Extending ArcGIS Desktop Applications
  - Engine:
    - Developing Applications with ArcGIS Engine (.NET, Java)
- **ArcGIS Server**
  - Introduction to ArcGIS Server
  - Developing Applications with ArcGIS Server (.NET, Java)