Agenda

- Introduction to Engine
  - Controls
  - SDK Requirements
- Runtime Binding
- New Functionality at 10
  - Basemap Layers
  - Working with Packages
  - GraphicTracker
  - Background Geoprocessing
- Custom Components
What is ArcGIS Engine?
What is ArcGIS Engine?

• Product used to build custom standalone GIS solutions
  - ArcObjects and ArcGIS Engine Controls
• Leverage the power of ArcGIS Desktop and Server
  - Use Desktop to author content
  - Consume Maps, Models, Geodatabases, Layers, Layer Packages, Query Layers, Services, and more…
What is ArcGIS Engine?

• Engine Controls
  - Components that can be added to windows forms
  - Add mapping and supporting GIS functionality to your custom application
  - 8 Controls
  - Over 200 built in commands
    - Editing, Add Data, Open Document, Select, Find, and more…
What is needed to Develop .NET Engine Applications?

- Visual Studio 2008 sp 1
- Visual Studio 2010
- VS Express 2008
- .NET Framework - 3.5 sp 1
- ArcGIS Requirements
  - ArcGIS Engine Runtime
  - ArcObjects .NET SDK
  - ArcGIS Engine Developer Kit License
ArcGIS Engine Resources

- Visual Studio 2008 / 2010 IDE Integration
  - Tools to make development easier and faster
- ArcObjects .NET SDK
  - Walkthroughs, samples, code snippets, Object Model Diagrams, API reference
- ArcGIS Desktop Help
  - GIS and data concepts
- Resource Centers, Blogs, and Forums
  - http://resources.arcgis.com/
- Support Center
  - Technical Articles, white papers, downloads
64 Bit Support

• ArcGIS Engine is a 32 bit application
  - Run as a 32 bit applications on a 64 Bit OS
  - Set platform to x86 in Visual Studio Configuration Manager
  - Default is “Any CPU”

• At Version 10 Engine applications are Large Address Aware
  - On 64 Bit OS 32 bit processes can take up to 4 gigabytes of RAM if available

• When compiling VS 2010 applications on a x64 machine
  - Follow KB 37879
UAC and Engine Applications

- User Account Control (UAC) on Windows
  - UAC restricts access certain parts of the system
    - Program Files directory
    - Parts of the registry
- Can be changed in the application Manifest
  - Enables application to be run as an administrator
Runtime Binding and Licensing
Runtime Binding

What is Runtime Binding?

- At ArcGIS 10 each product has its own runtime
  - Products have separate install locations
  - Service pack products separately
  - Uninstall service packs

- ArcObjects must be pointed to a runtime to work
  - Before any other ArcObjects calls
  - Required for all standalone applications
  - Engine applications can bind to either Desktop or Engine Runtimes
Runtime Binding

How to bind to a runtime

- Bind using the `RuntimeManager` static class

- Add reference to: `ESRI.ArcGIS.Version`
Runtime Binding

How to bind to a runtime

- Add the following code, before any other ArcObjects calls:

  ```csharp
  ESRI.ArcGIS.RuntimeManager.Bind(ESRI.ArcGIS.ProductCode.Engine);
  ESRI.ArcGIS.RuntimeManager.Bind(ESRI.ArcGIS.ProductCode.Engine);
  ```

- Preferably in your main method or in the application events

- Tip:
  - The Assembly is called `ESRI.ArcGIS.Version`
  - The Namespace is `ESRI.ArcGIS`
  - Bind method returns a Boolean that you can use to handle binding errors
Runtime Binding

Additional Functionality and uses

- **RuntimeManager.BindLicense**
  - Bind and License with one method
- **RuntimeCollection**
  - Identify Installed Runtimes
- **RuntimeInfo**
  - Identify the Path, Product, & Version
- **ActiveRuntime**
  - Currently bound runtime
Licensing

- Engine applications must check out a license at runtime
  - Either Desktop or Engine License
  - Product Licenses are checked out for the life of the application
- Extensions can be checked out and returned as needed
- At ArcGIS 10 Engine concurrent Engine licenses are available
Licensing

- Licensing is not the same as binding
  - Binding specifies the runtime
  - Licensing specifies the product functionality
- Can mix and match Licensing and runtimes for flexible Engine applications
  - Leverage existing Desktop runtime and licenses on client’s machine
Controls and Binding Demo
New Functionality at 10
Basemap and RasterBasemap Layers

What are Basemap Layers

• Layers that draw with a high-performance multi-threaded drawing engine
  - Improves display speed and responsiveness of your map
  - Seamless display

• Limitations
  - Layers in a Basemap layer have less functionality
    - Cannot modify layer properties
  - Some data sources, layer types, and symbols are not supported
Basemap and RasterBasemap Layers

Using Basemap Layers in Engine

• Best to author through ArcMap
  - Make use of the analyzers for optimization

• For optimal performance
  - Test to see if you can enable hardware acceleration
    - IGlobalScreenDisplaysettings.CanEnableHardwareAcceleration
  - Enable hardware acceleration
    - IGlobalScreenDisplaysettings.EnableHardwareAcceleration
Basemap Layer Demo
Working with Packages

What are packages?

- Packages are a single file that contains a map or layer[s] and supporting data
  - Also can contain references to SDE data
- Easy to share
  - Single file
- ArcGIS 10 supports
  - Layer Packages (.lpk)
  - Map Packages (.mpk)
Working with Packages

Using Packages in Engine

- At ArcGIS 10 sp 1
  - Programmatically consume packages
- IMapDocument.Open
  - Map Packages
  - Layer Packages
  - Web Maps
- ILayerFile.Open
  - Layer Packages
- Just point to the path of the package
Working with Packages

Using Online Content

- **Consume data on ArcGIS Online**
  - Pass in a URL with the id as the filename
    - [http://www.arcgis.com/home/item.html?id=224ee2a012154bbf84bcb5b04ea35fb5](http://www.arcgis.com/home/item.html?id=224ee2a012154bbf84bcb5b04ea35fb5)

- **URL to ArcGIS Online Data**
  - Point to the item.pkinfo file online
    - [http://www.arcgis.com/sharing/content/items/224ee2a012154bbf84bcb5b04ea35fb5/item.pkinfo](http://www.arcgis.com/sharing/content/items/224ee2a012154bbf84bcb5b04ea35fb5/item.pkinfo)
Working with Packages
Working with Packages

Why use Packages in Engine?

• Provides a mechanism to easily deploy maps and data with your Engine solution
  - Simple deployment single file
  - Easy to update data off cycle

• Data can be uploaded to ArcGIS Online
  - Once a packages is downloaded it can be used locally
  - Use ArcGIS Online groups to manage access to data
GraphicTracker

What is the GraphicTracker?

- **Simple API**
  - Add, remove, update, and move graphics
  - Pass in a geometry and symbol
    - Works with Points, Lines, and Polygons
- **Same API for Map, Globe, and Dynamic Display**
  - GraphicTracker manages the display
- **All objects passed in ByValue**
  - Objects managed by the GraphicTracker
GraphicTracker

Tips when using the GraphicTracker

💡 • Pause with IGraphicTracker.SuspendUpdate
  - For adding groups of items

• IGraphicTracker.Add method returns an integer to reference the graphic
  - Store this integer into a table for easy reference to use later
  - GraphicTracker ids may not be sequential

• Use multiple GraphicTrackers
  - Separate GraphicTrackers for points, lines, and polygons
GraphicTracker Demo
GraphicTracker

Performance Considerations

💡 How many graphics does the GraphicTracker support?
  - Number of graphics
  - Complexity of the graphics and symbols
  - Complexity of your map
  - Update interval for moving graphics
  - Using labels

VS.
Background Geoprocessing

What is Background Geoprocessing

- Framework to allow Geoprocessing tools to execute in a separate processes
  - Great alternative to multi-threaded application
- Allows User Interface to remain responsive while processing
Background Geoprocessing

What is Background Geoprocessing

- Execute tool using ExecuteAsync method on the Geoprocessor object
- Wire in the events
  - ToolExecuted
    - Must handle to know when tools completes
  - ProgressChanged
  - MessagesCreated
  - ToolExecuting
- Run system tools, models, and script tools.
Background Geoprocessing Demo
Background Geoprocessing

Tips for working with the Geoprocessor

• Tips:
  - Set OverwriteOutput = True
  - Tools require different license levels or extensions
  - Desktop help is your friend
  - Understand GP Messaging
  - Learn about the Result Object
Custom Components
Custom Components

- Engine applications that use custom components in a component categories
  - Components shipped in a separate dll
  - Components that are used by both Engine and Desktop
  - Register using EsriRegAsm.exe
    - ESRIRegAsm.exe <MyDll.dll> /p:Engine
  - Creates an *.ecfg file

  %CommonProgramFiles%\ArcGIS\Engine10.0\Configuration\CATID
New ArcGIS Runtime – Early Look at the APIs (WPF, Java & Qt)
- Thursday March 10th 1:30pm – 3:30pm (After Closing Session)
- Mesquite B in the Palm Springs Convention Center
Questions?

- Please don’t forget to fill out the surveys