

Esri Developer Summit

March 26–29, 2012 | Palm Springs, California

esri.com/events/devsummit



Developing .NET Applications for ArcGIS Engine

Deep Dhanasekaran and John Hauck

A decorative graphic at the bottom of the slide. It features a curved orange border at the top. Below the border is a semi-transparent map of a landscape with green fields and blue water. Overlaid on the map is white text representing code snippets. The code includes symbols like 'esri.symbol.SimpleLineSymbol', 'new dojo.Color([0,0,0,0.5])', and 'feature.setSymbol(polySymbolGreen)'.

```
esri.symbol.SimpleLineSymbol  
new dojo.Color([0,0,0,0.5])  
polySymbolGreen = new esri.symbol.SimpleLineSymbol(  
    dojo.Color([0,0,0,0.5]), 1);  
feature.setSymbol(polySymbolGreen);  
}  
else if(f == 1) {  
    var polySymbolGreen = new esri.symbol.SimpleLineSymbol(  
        dojo.Color([0,0,0,0.5]), 1);  
    feature.setSymbol(polySymbolGreen);  
}  
else if(f == 2) {  
    var polyBlue = new esri.symbol.SimpleLineSymbol(  
        dojo.Color([0,0,255,0.5]), 1);  
    feature.setSymbol(polyBlue);  
}
```

Agenda

- **Introduction to Engine**
- **Runtime Binding and Licensing**
- **Authoring Content for Engine**
- **Adding Functionality to Engine**
- **ArcGIS Engine and New Runtime for ArcGIS**

ArcGIS 10 — A Complete System

Easier
More Powerful
and Everywhere



What is ArcGIS Engine?



```
esri.symbol.SimpleLineSymbol,
new dojo.Color([0,0,0]),
polySymbol,
feature.setSymbol,
} else if(f == 1) {
var polySymbolGreen =
polySymbolGreen.setOutline(
symbol.SimpleLineSymbol(esri.symbol.
Color([0,0,0,0.5]), 1));
polySymbolGreen.setSymbol(polySymbolGreen);
} else if(f == 2) {
polyBlue = new esri.symbol.SimpleLineSymbol(
new dojo.Color([0,0,0,0.5]), 1);
polyBlue.setOutline(new esri.symbol.SimpleLineSymbol(
new dojo.Color([0,0,0,0.5]), 1));
polyBlue.setSymbol(polyBlue);
}
```

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?

10.0

- Visual Studio 2008 sp1
- VS Express 2008
- Visual Studio 2010
- .NET Framework – 3.5 sp1
- ArcGIS Requirements
 - ArcGIS Engine Runtime
 - ArcObjects .NET SDK
 - ArcGIS Engine Developer Kit License



What is needed to Develop .NET Engine Applications?

10.1

- **Visual Studio 2010**
- **Visual Studio 2010 Express**
- **.NET Framework – 3.5 sp1**
- **ArcGIS Requirements**
 - **ArcGIS Engine Runtime**
 - **ArcObjects SDK**
 - **ArcGIS Engine Developer Kit License**



.NET Framework

- **.NET Framework 3.5 is min requirement**
 - **Guaranteed to be installed with Engine or Desktop**
- **.NET Framework 4.0 is fully supported**
 - **Need to handle distribution**
 - **Desktop developer Add-ins install without Admin requirements?**
- **Do NOT embed iterop types**

ArcGIS Engine Resources

- **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>
 - <http://resourcesbeta.arcgis.com>
- **Support Center**
 - Technical Articles, white papers, downloads





ArcGIS Engine Controls and SDK Tools



Runtime Binding and Licensing

```
esri.symbol.SimpleLineSymbol,
new dojo.Color([0, 0, 0, 0.5]),
polySymbol,
feature.setSymbol(polySymbol);
} else if(f == 1) {
var polySymbolGreen =
polySymbolGreen.setOutline(
symbol.SimpleLineSymbol(esri.symbol.
Color([0, 0, 0, 0.5]), 1));
polySymbolGreen.setcolor(new dojo.
Color([0, 0, 0, 0.5]));
feature.setSymbol(polySymbolGreen);
} else if(f == 2) {
polySymbolBlue = new esri.symbol.SimpleLineSymbol(
new dojo.Color([0, 0, 0, 0.5]), 1);
polySymbolBlue.setOutline(
new esri.symbol.SimpleLineSymbol(
new dojo.Color([0, 0, 0, 0.5]), 1));
feature.setSymbol(polySymbolBlue);
}
```

Runtime Binding

What is Runtime Binding?

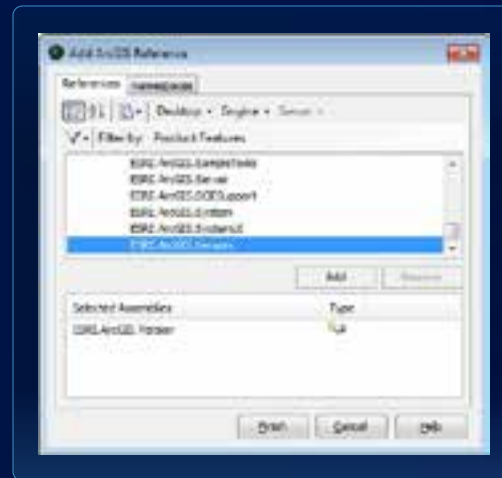
- **At ArcGIS 10 each product has its own runtime**
 - Products have separate install locations
 - Service pack products separately
 - Uninstall service packs
- **ArcObjects must be pointed to a runtime to work**
 - Before any other ArcObjects calls
 - Required for all standalone applications
 - Engine applications can bind to either Desktop or Engine Runtimes



Runtime Binding

How to bind to a runtime

- Add reference to:
ESRI.ArcGIS.Version



- Bind using the **RuntimeManager** static class

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

Runtime Binding

When to bind to a runtime?

- **Before any other ArcObjects calls**
 - Preferably in your main method (C#)
 - Preferably in your application events StartUp method (VB)
- **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

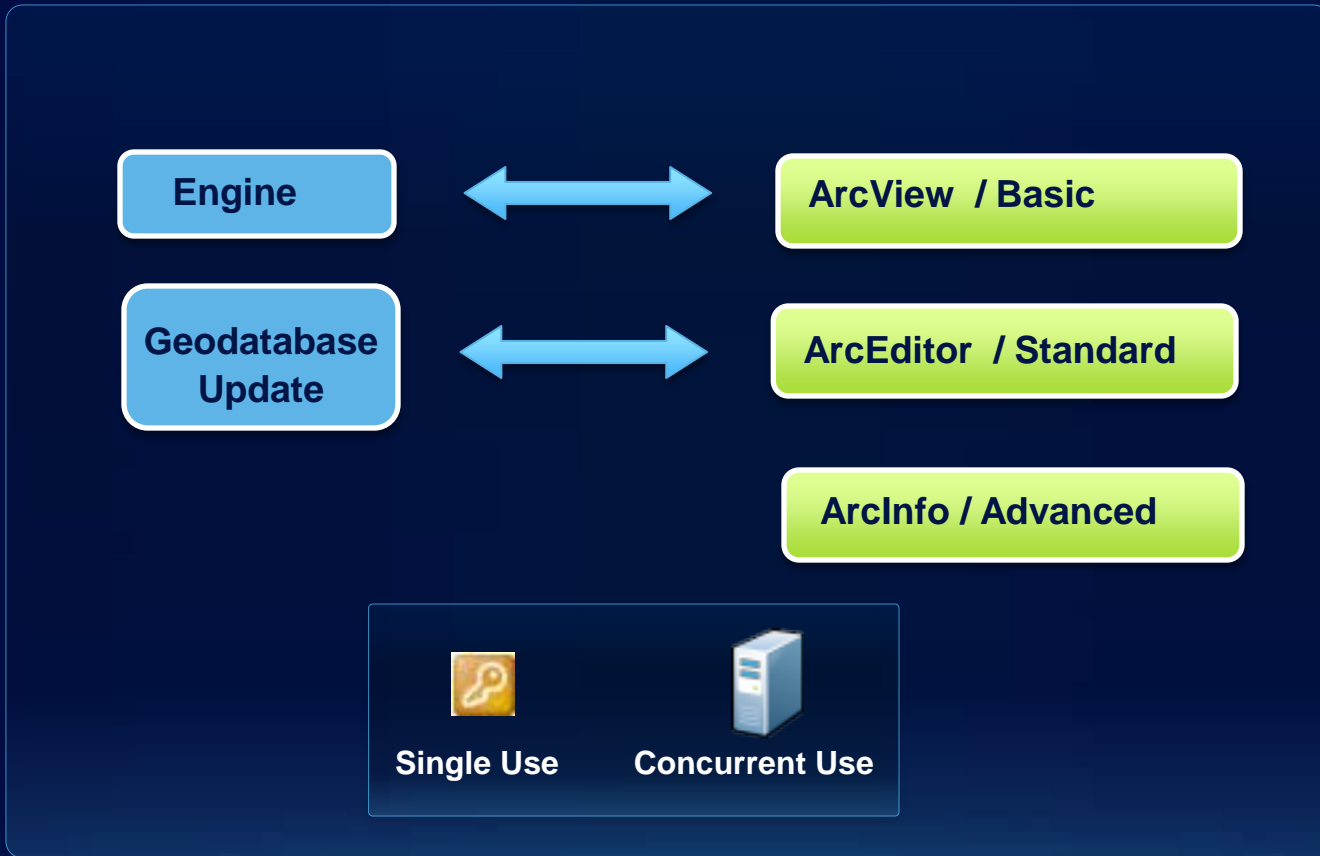
Additional Functionality and uses

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

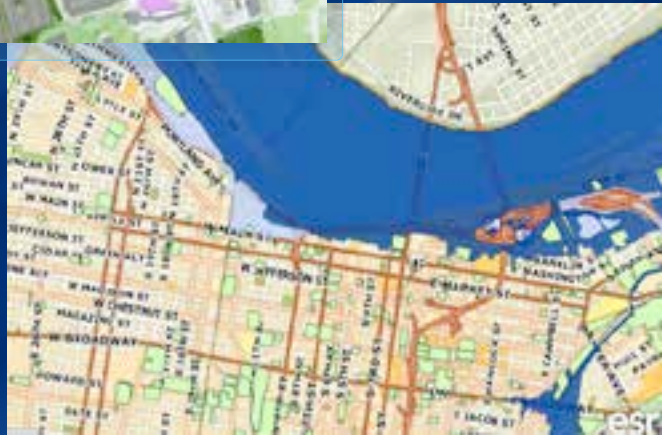
Licensing

- **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 is not the same as binding**
 - Binding specifies the runtime
 - Licensing specifies product functionality

Licensing



Binding and Licensing



Binding and Deployment

- **ArcGIS Engine is not required on the target machine**
 - Engine applications work with either a Desktop or Engine Runtime
- **Can mix and match Licensing and runtime for flexible Engine applications**
 - Leverage existing Desktop licenses and runtime on clients machine

Authoring Content for Engine

```
esri.symbol.SimpleLineSymbol({
  color: [0, 0, 255],
  width: 2
});
feature.setSymbol(new esri.symbol.SimpleLineSymbol({
  color: [0, 0, 255],
  width: 2
}));
feature.setSymbol(new esri.symbol.SimpleLineSymbol({
  color: [0, 0, 255],
  width: 2
}));
```

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 10.0 supports**
 - Layer Packages (*.lpk)
 - Map Packages (*.mpk)
- **ArcGIS 10.1 adds additional support for**
 - Geoprocessing Packages (*.gpk)
 - Locator Packages (*.apk)
 - Tile Packages (*.tpk)



Working with Packages

Using Packages in Engine

- **At ArcGIS 10 sp1**
 - 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 Packages in Engine

- Opening Layer Packages

```
Dim layerFile As ILayerFile = New LayerFileClass  
layerFile.Open("c:\Data\LayerPackages\USCities.lpk")  
Dim layer As ILayer = layerFile.Layer  
axMapControl1.AddLayer(layer)
```

- Opening Map Packages

```
Dim mapDocument As IMapDocument = New MapDocumentClass  
mapDocument.Open("c:\Data\LayerPackages\MyMapPackage.mpk", "")  
axMapControl1.Map = mapDocument.get_Map(0)
```


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=a50c645f83bb4f5dbddd457df29d639>

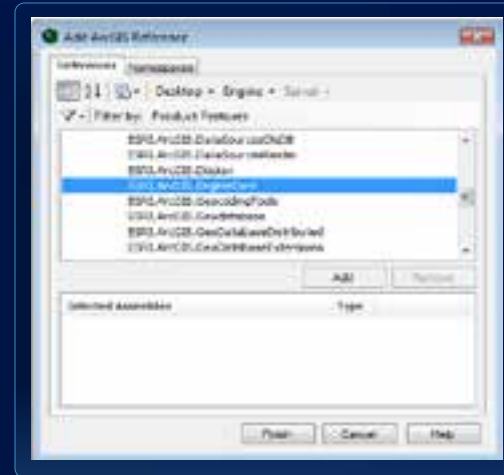
```
private void addLayerFromOnlinePackage(string packageId)
{
    string lyrPackage = "http://www.arcgis.com/sharing/content/items/" + packageId + "/item.pkinfo";
    ILayerFile layerFile = new LayerFileClass();
    layerFile.Open(lyrPackage);
    ILayer layer = layerFile.Layer;
    axMapControl1.AddLayer(layer);
}
```



Working with Packages

Using IPackageFile at 10.1

- Add reference to:
ESRI.ArcGIS.EngineCore



- Unpack using IPackageFile.Unpack

```
private void unPack(string packagePath, string unpackToLocation)
{
    IPackageFile packageFile = new PackageFile();
    IStringArray stringArray = packageFile.Unpack(packagePath, ref unpackToLocation);
    axMapControl1.AddLayerFromFile(stringArray.get_Element(0));
}
```

Working with Packages

Why use Packages in Engine?

- **Easily deploy maps, data, and functionality with your Engine solution**
 - Simple deployment of a single file
 - Easy to update off cycle
- **Packages can be uploaded to ArcGIS Online**
 - Once a package is downloaded it can be used locally
 - Use ArcGIS Online groups to manages access

Adding Functionality to Engine

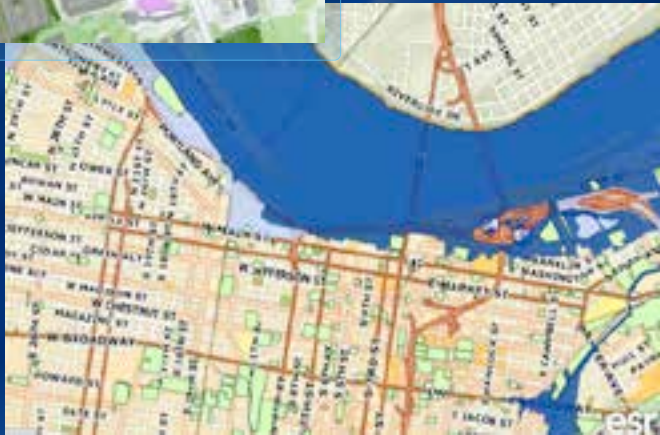
```
esri.symbol.SimpleLineSymbol,
new dojo.Color([0,0,0]),
polySymbol,
feature.setSymbol(polySymbol);
} else if(f == 1) {
var polySymbolGreen =
polySymbolGreen.setOutline(
symbol.SimpleLineSymbol(esri.symbol.
Color([0,0,0,0.5]), 1));
polySymbolGreen.setSymbol(polySymbolGreen);
feature.setSymbol(polySymbolGreen);
} else if(f == 2) {
var polySymbolBlue = new esri.symbol.SimpleLineSymbol(
new dojo.Color([0,0,255,0.5]), 1);
polySymbolBlue.setOutline(polySymbolBlue);
feature.setSymbol(polySymbolBlue);
}
```

Adding Functionality to Engine

- **Leverage existing commands and tools included with the Engine SDK**
 - On a **ToolBarControl**
 - **Programmatically**
- **Build your own components**
 - **Such as Commands, Tools, Extensions, Custom Layers, etc...**
 - **Specific to your Engine Application**
 - **Generic for all Engine applications**
 - **Work in both Engine and Desktop applications**



Adding Functionality to Engine



Using Geoprocessing in Engine

- **Why use geoprocessing in Engine?**
 - Why reinvent the wheel?
 - ArcGIS comes with hundreds of Geoprocessing tools
 - Developed by specialists in their field
- **Provides a framework to author Model and Script tools in ArcGIS Desktop**
 - Consume these tools in Engine

Using Geoprocessing in Engine

- Using system tools

```
Imports ESRI.ArcGIS.Geoprocessor
'System Toolboxes have their own Assembly
Imports ESRI.ArcGIS.AnalysisTools

Dim gp As Geoprocessor = New Geoprocessor

'Create the clip tool
Dim clipTool As Clip = New Clip
clipTool.In_features = "C:\Data\Test.gdb\InFeatures"
clipTool.clip_features = "C:\Data\Test.gdb\ClipFeatures"
clipTool.out_feature_class = "C:\Data\Test.gdb\ResultFeatures"

'Execute the Tool
gp.Execute(clipTool, Nothing)
```


Using Geoprocessing in Engine

- Using custom tools

```
Imports ESRI.ArcGIS.Geoprocessor

Dim gp As Geoprocessor = New Geoprocessor

'Add the toolbox
gp.AddToolbox("C:\Data\MyToolbox.tbx")

'Populate the parameters
Dim parameters As IVariantArray = New VarArray
parameters.Add("C:\Data\Test.gdb\InFeatures")
parameters.Add("C:\Data\Test.gdb\ProcessFeatures")
parameters.Add("C:\Data\Test.gdb\ResultsFeatures")

'Execute the Tool
gp.Execute("MyTool", parameters, Nothing)
```

Using Background Geoprocessing in Engine

What is Background Geoprocessing

- **Framework to allow Geoprocessing tools to execute in a separate process**
- **Great alternative to multi-threaded applications**
- **Allows User Interface to remain responsive while processing**



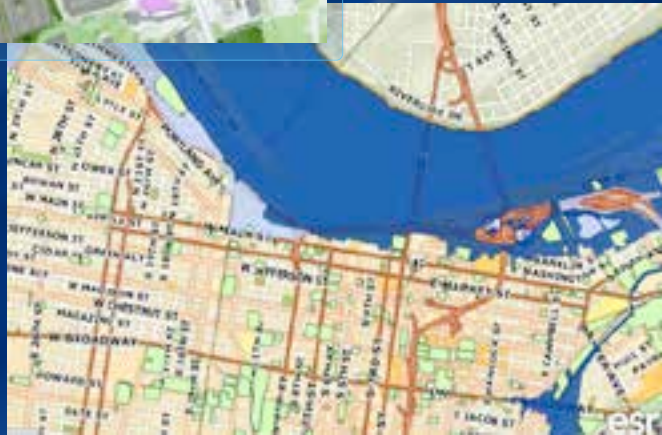
Background Geoprocessing

What is Background Geoprocessing

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



Using Geoprocessing in Engine



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**

Graphic Tracker

```
esri.symbol.SimpleLineSymbol,
new dojo.Color([0,0,0]),
polySymbol,
feature.setSymbol(polySymbol);
} else if(f == 1) {
var polySymbolGreen = new
polySymbolGreen.setOutline(
symbol.SimpleLineSymbol(esri.symbol.
Color([0,0,0,0.5]), 1));
polySymbolGreen.setcolor(new dojo.
feature.setSymbol(polySymbolGreen);
} else if(f == 2) {
polyBlue = new esri.symbol.SimpleLineSymbol(
setOutline(new
esri.symbol.SimpleLineSymbol(
new dojo.Color([0,0,0,0.5]), 1));
polyBlue);
```

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

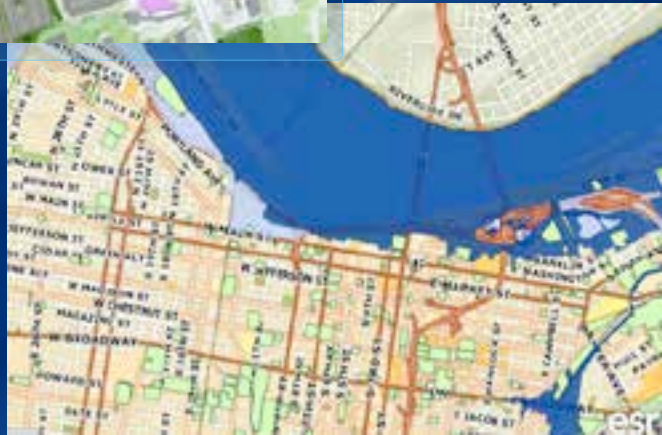


GraphicTracker

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 set 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**

New Runtime for ArcGIS

```
esri.symbol.SimpleLineSymbol,
new dojo.Color([0,0,0,0.5]),
polySymbol(feature, setSymbolColor);
} else if(f == 1) {
var polySymbolGreen =
polySymbolGreen.setOutlineColor(
symbol.SimpleLineSymbol(esri.symbol.
Color([0,0,0,0.5]), 1));
polySymbolGreen.setSymbol(polySymbolGreen);
} else if(f == 2) {
polySymbolGreen = new esri.symbol.SimpleLineSymbol(
new dojo.Color([0,0,0,0.5]), 1);
polySymbolGreen.setSymbol(polySymbolGreen);
}
```

New Runtime for ArcGIS

- **New Architecture**
- **Easy to Deploy Solutions**
 - Software, maps and data
- **64 bit Native Support**
- **Modern API**
 - WPF and Java
- **Reduced Memory and Disk Footprint**

New Runtime for ArcGIS

- **Not a functional equivalent to Engine**
- **Not an ArcObjects replacement**
- **New APIs**
 - **Code will need to be re-written**
 - **Should leverage general patterns for data sharing and distribution**

Questions?

- Please don't forget to fill out the surveys



esri

Additional Tips

```
esri.symbol.SimpleLineSymbol,
new dojo.Color([0,0,0]),
polySymbol(feature, setSymbolColor);
} else if(f == 1) {
var polySymbolGreen = new
polySymbolGreen.setOutlineColor(
symbol.SimpleLineSymbol(esri.symbol.
Color([0,0,0,0.5]), 1));
polySymbolGreen.setSymbol(polySymbolGreen);
feature.setSymbol(polySymbolGreen);
} else if(f == 2) {
polyBlue = new esri.symbol.SimpleLineSymbol(
setOutlineColor(new dojo.Color([0,0,0,0.5]), 1));
esri.symbol.SimpleLineSymbol(
new dojo.Color([0,0,0,0.5]), 1));
polyBlue);
```


64 Bit Support

- **ArcGIS Engine is a 32 bit application**
 - Run as a 32 bit application on 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 GB 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 to 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**

Disabling Windows Aero



The image shows a video player window with a blue border. The video content is a slide with a black background and white text. The slide title is "Disabling Windows Aero Themes". Below the title is a bulleted list of instructions. The first bullet point is "Esri recommends disabling Windows Aero themes in Engine Applications". The second bullet point is "KB [38485](#)". The third bullet point is "Use P/Invoke". To the left of the third bullet point is a small yellow lightbulb icon. Below the third bullet point, there are two sub-bullets: "Load the Desktop Windows library if it exists" and "Remember XP does not support Aero". The fourth bullet point is "Disable Aero for the life of the application". At the bottom of the video player, there is a progress bar and a volume control icon. The text "Medium | Large" is visible below the progress bar.

Disabling Windows Aero Themes

- Esri recommends disabling Windows Aero themes in Engine Applications
- KB [38485](#)
- Use P/Invoke
 - Load the Desktop Windows library if it exists
 - Remember XP does not support Aero
- Disable Aero for the life of the application

Medium | Large