



Moving ArcObjects Code to ArcGIS Server from ArcGIS Desktop

Allan Laframboise

Jian Huang

Presentation outline

- **ArcGIS Desktop applications and extensions**
- **ArcObjects to ArcGIS Server migration options**
- **How to leverage the ArcGIS Server Web ADF**

Prerequisites

- **Development experience with ArcObjects**
- **Development experience with .NET and ASP.NET**
- **Basic understanding of ArcGIS Server**

Questions

- **How many:**
 - **Want to migrate Desktop (VBA, VB, VC++, .NET) applications?**
 - **Want to migrate Engine (VB, VC++, , .NET) applications?**
 - **Already have ArcGIS Server development experience?**

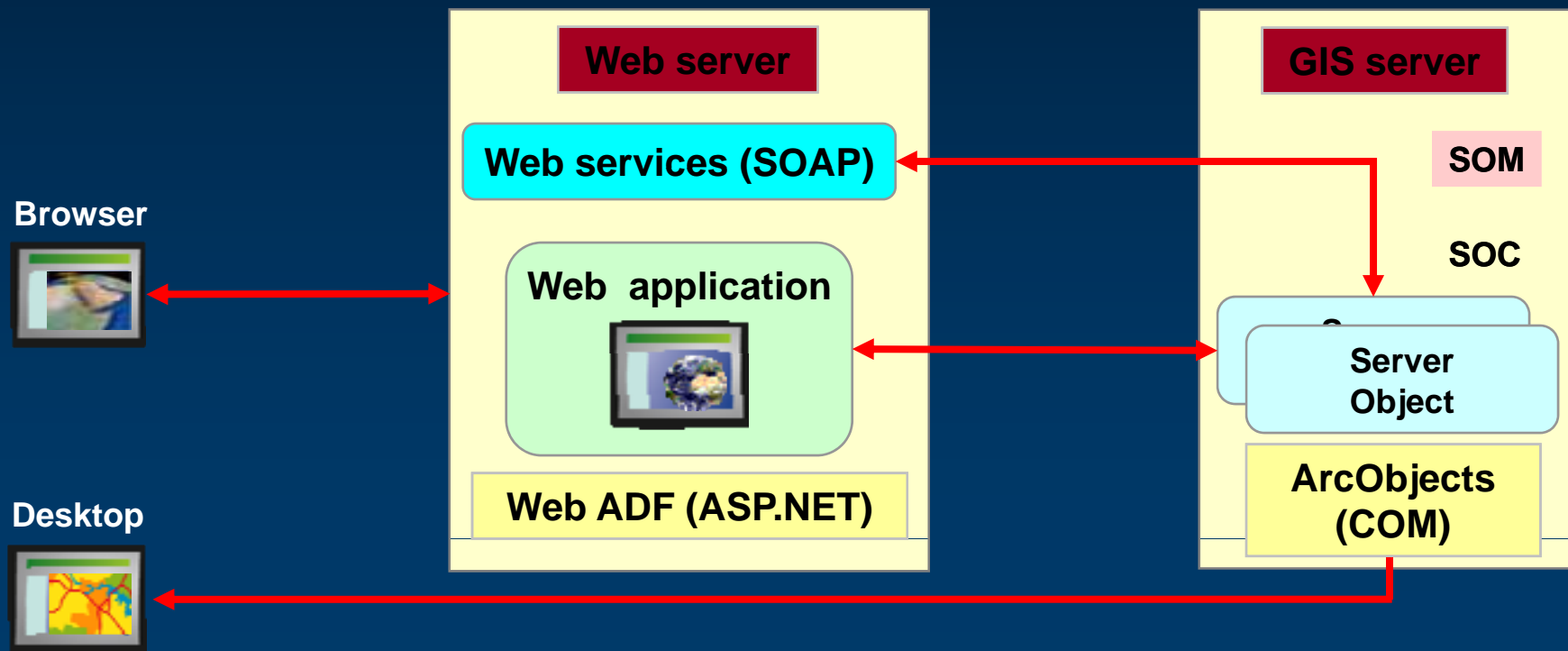
Types of ArcObjects applications and extensions

- **ArcGIS Desktop – ArcObjects Extensions**
 - ArcMap, ArcCatalog, ArcScene or ArcGlobe
 - Commands, tools, toolbars, windows, extensions
 - Mostly COM implementations (VB 6, VC++, .NET)
- **ArcGIS Engine – ArcObjects Applications**
 - Custom standalone forms-based and utility applications
 - Commands, tools, extensions (VB 6, VC++, Java, .NET)
 - COM or pure .NET implementations
- **ArcObjects Components**
 - Utility components and DLLs



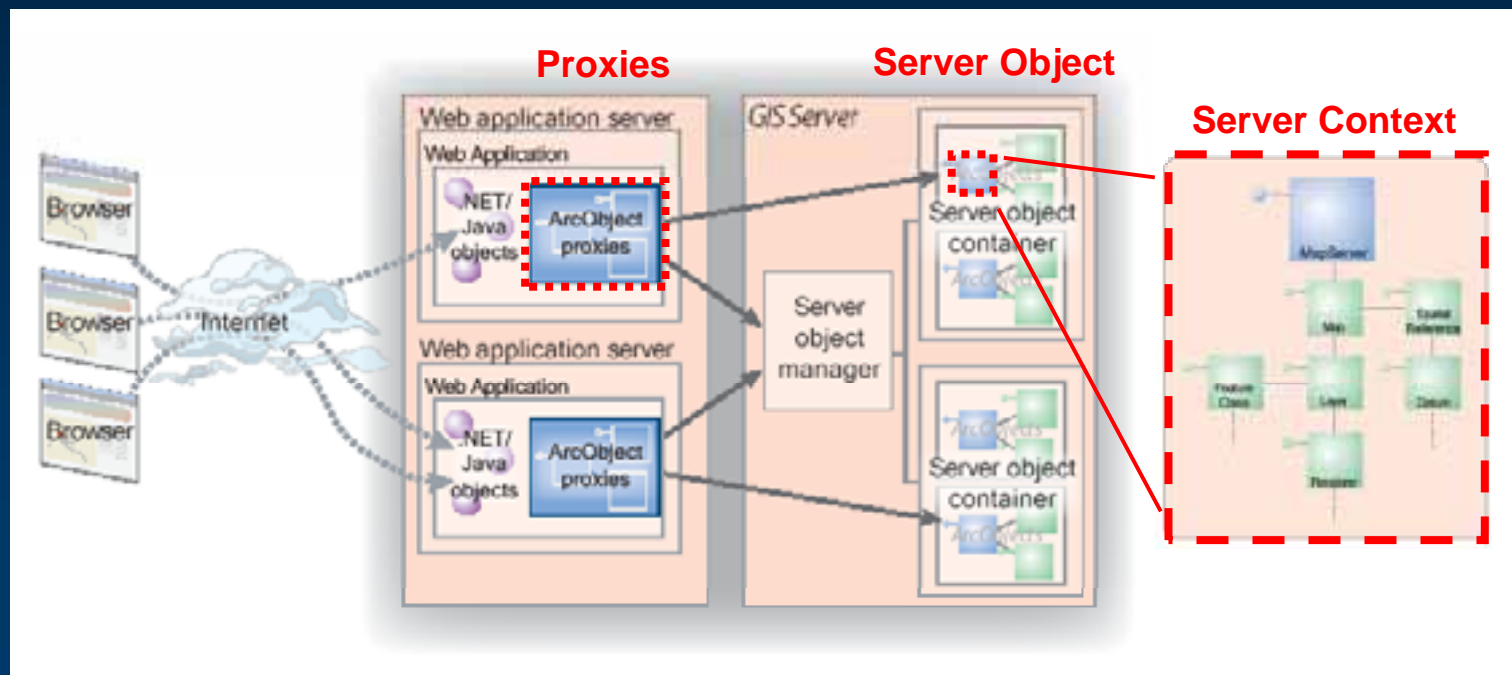
Migrating desktop applications: The challenge

1. ArcGIS Server is a multi-tiered, multi-API system
2. ArcObjects code runs remotely on the GIS Server
3. User interface is a web browser



Accessing ArcObjects remotely

1. Web app communicates through ArcObjects proxies
2. Accesses a Server Object through the SOM
3. Works with ArcObjects in a Server Context (ArcSOC.exe)



Migrating ArcObjects code to ArcGIS Server

- **Different approaches:**
 - 1. ASP.NET**
 - 2. Geoprocessing**
 - 3. ArcGIS Server Web ADF**

Which ArcObjects components can be used?

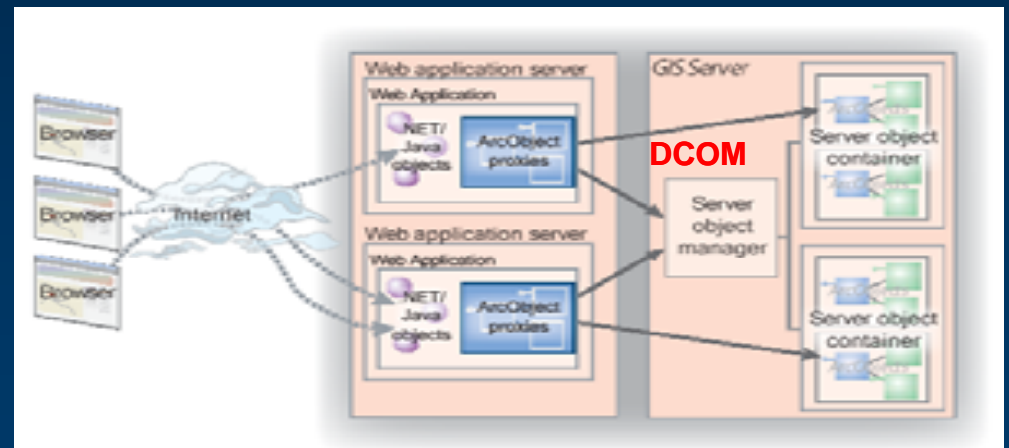
- All of the “shared” ArcObjects libraries are available
- Non-UI libraries
 - System, Geometry, Server, Geodatabase, DataSourcesGDB...
- Limitations
 - Members that return HWND, HDC, output...
- For more information, see EDN...

http://edndoc.esri.com/arcobjects/9.2/ComponentHelp/shared_libs.htm

Migrating ArcObjects code directly to ASP.NET

- Move code from .NET desktop application to ASP.NET
- ArcObjects proxies communicate through (DCOM)
- When to use

– Need to access the fine-grained ArcObjects



Migrating ArcObjects code directly to ASP.NET

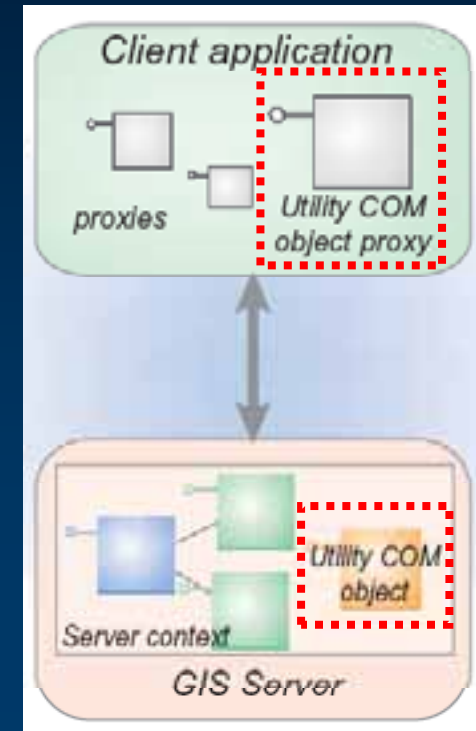
- **Steps**

- 1. Build an ASP.NET web application**
- 2. Copy/paste your ArcObjects code**
- 3. Replace “New” with “CreateObject”**
- 4. Manage objects in the server context**



Migrating ArcObjects code to a Utility Object

- Build a custom COM component to reduce the number of fine-grained ArcObjects calls
- Moves code to the GIS Server (SOC)
- When to use
 - Need to access a large ArcObjects code base
 - Share code with other applications
 - Optimize performance



Migrating ArcObjects code to a Utility Object

- **Steps**
 1. **Build a course-grained COM utility object**
 2. **Move fine-grained ArcObjects code to utility object**
 3. **Use “CreateObject” to create and access object**
 4. **Manage object’s lifetime in the “server context”**
- **Need to register on all SOC machines!**



Implementing a Server Object Extension (SOE)

- **Extends the capabilities of the Server Object (9.2)**
- **Lifetime is managed by the Server Object**
- **Results in better performance**
- **Steps**
 - 1. Build a utility object**
 - 2. Build a SOE object (IServerObjectExtension)**
 - 3. (Optional) Build a custom property page for ArcCatalog**
 - 4. Access SOE from web application**



Implementing Geoprocessing

- Use out-of-the-box components in the Web ADF (9.2)
- Replace ArcObjects code with a Geoprocessing Service
- Build a model and return results to a web application
- When to use
 - Large processing tasks that can be modeled
 - Optimize ArcObjects tasks



GP model

Implementing Geoprocessing

- **Steps**

- 1. Build a Geoprocessing model**
- 2. Publish the model through a Map and Geoprocessing service**
- 3. Consume as a task in the Web Server application**



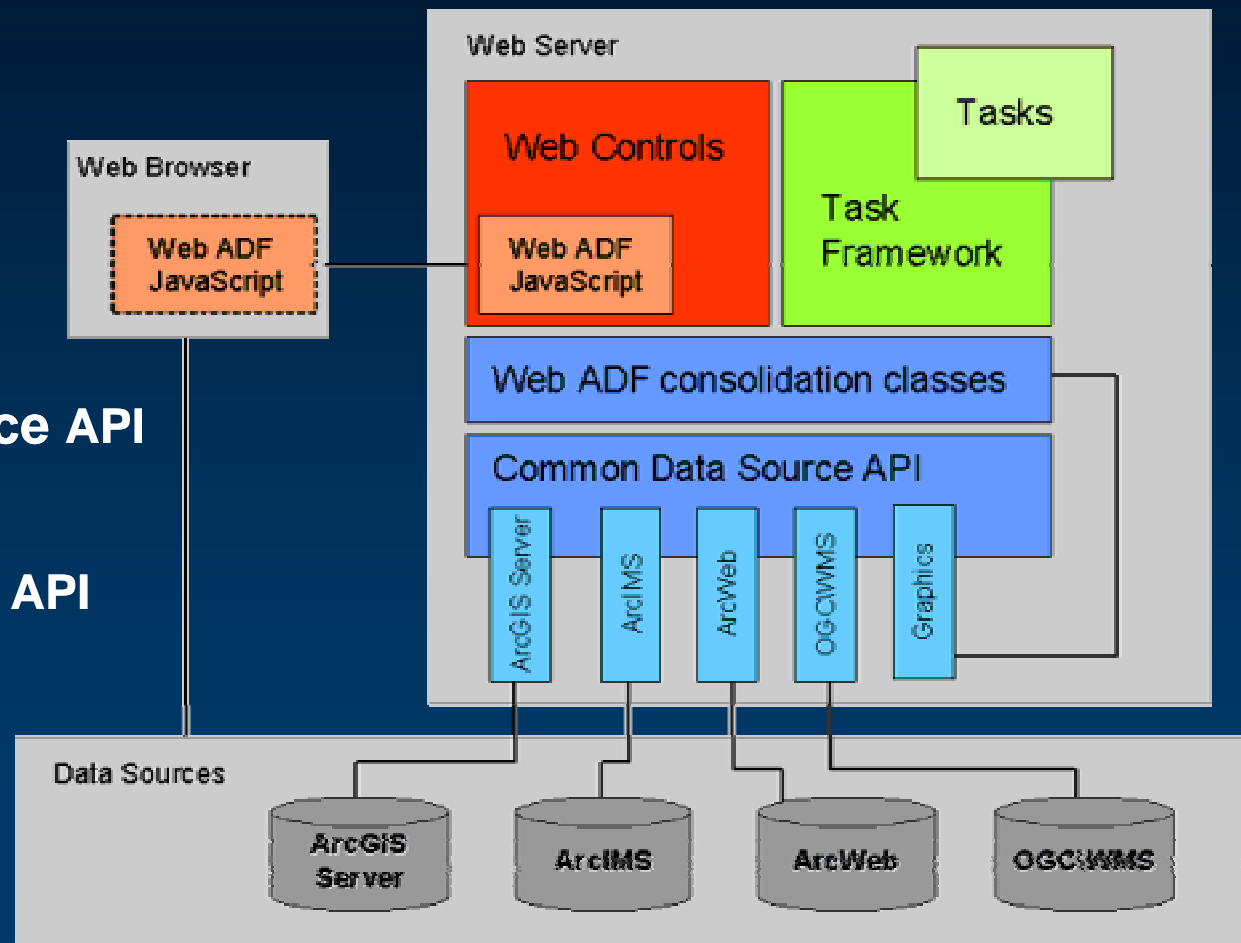
Introducing the ArcGIS Server Web ADF APIs

- **Goal: Replace ArcObjects functionality with the ADF**

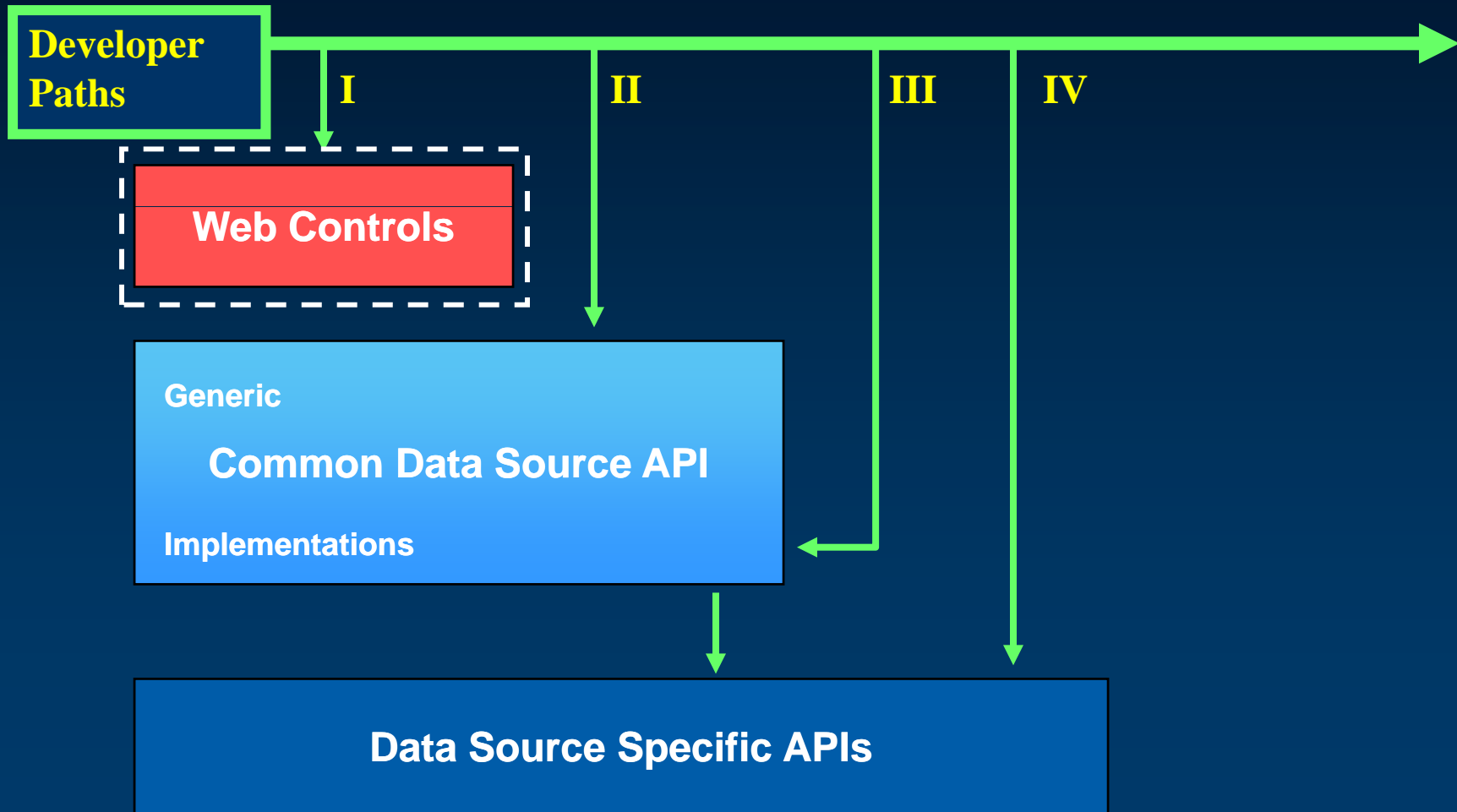
1. Web controls API

2. Common Datasource API

3. Datasource Specific API



ArcGIS Server Web ADF – Development Paths



Migrating the User Interface – Web Controls API

- **Start with the Visual Studio Map template**
 - Contains visual Web controls Javascript functions
- **WebControls API**
 - Pure ASP.NET controls
 - AJAX enabled
- **Functionality**
 - Visualization
 - User interaction

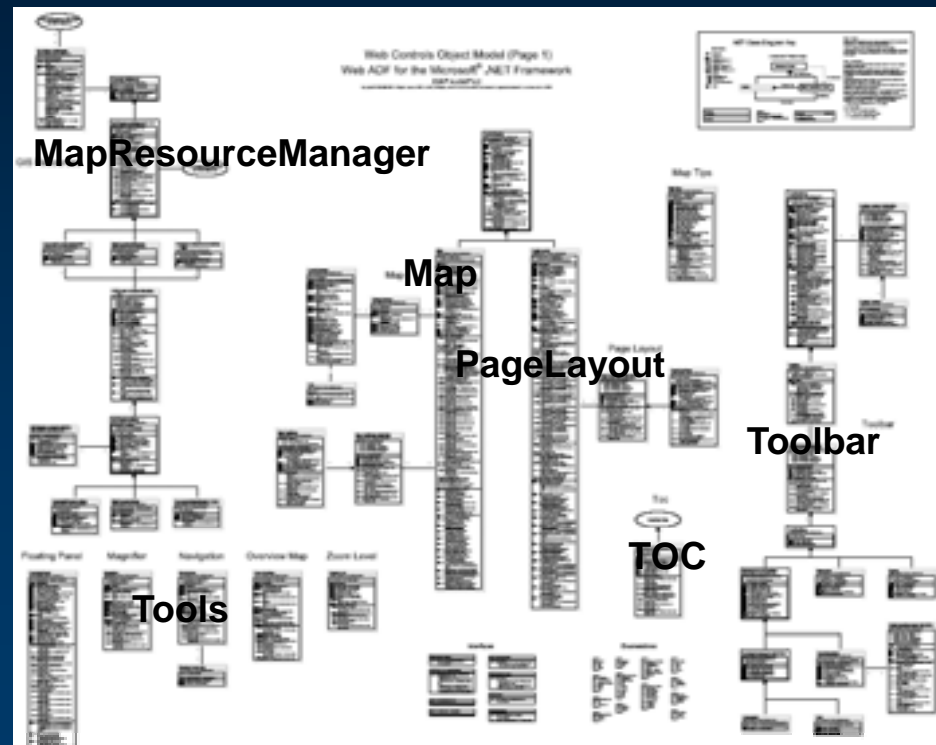
WebControls API



```
ESRI.ArcGIS.ADF
ESRI.ArcGIS.ADF.ArcGISServer
ESRI.ArcGIS.ADF.ArcGISServer.Editor
ESRI.ArcGIS.ADF.ArcWebService
ESRI.ArcGIS.ADF.Connection
ESRI.ArcGIS.ADF.IMS
ESRI.ArcGIS.ADF.Tasks
ESRI.ArcGIS.ADF.Web
ESRI.ArcGIS.ADF.Web.DataSources
ESRI.ArcGIS.ADF.Web.DataSources.ArcGISServer
ESRI.ArcGIS.ADF.Web.DataSources.ArcWebService
ESRI.ArcGIS.ADF.Web.DataSources.Graphics
ESRI.ArcGIS.ADF.Web.DataSources.IMS
ESRI.ArcGIS.ADF.Web.DataSources.OGCWMSService
ESRI.ArcGIS.ADF.Web.UI.WebControls
```

Migrating the User Interface – Web Controls API

- Controls for all common GIS user interactions
- MapResourceManager
- Map and Pagelayout
- Toolbar
- TOC
- Buttons and tools
- Scalebar
- Northarrow
- FloatingPanel
- TreeviewPlus

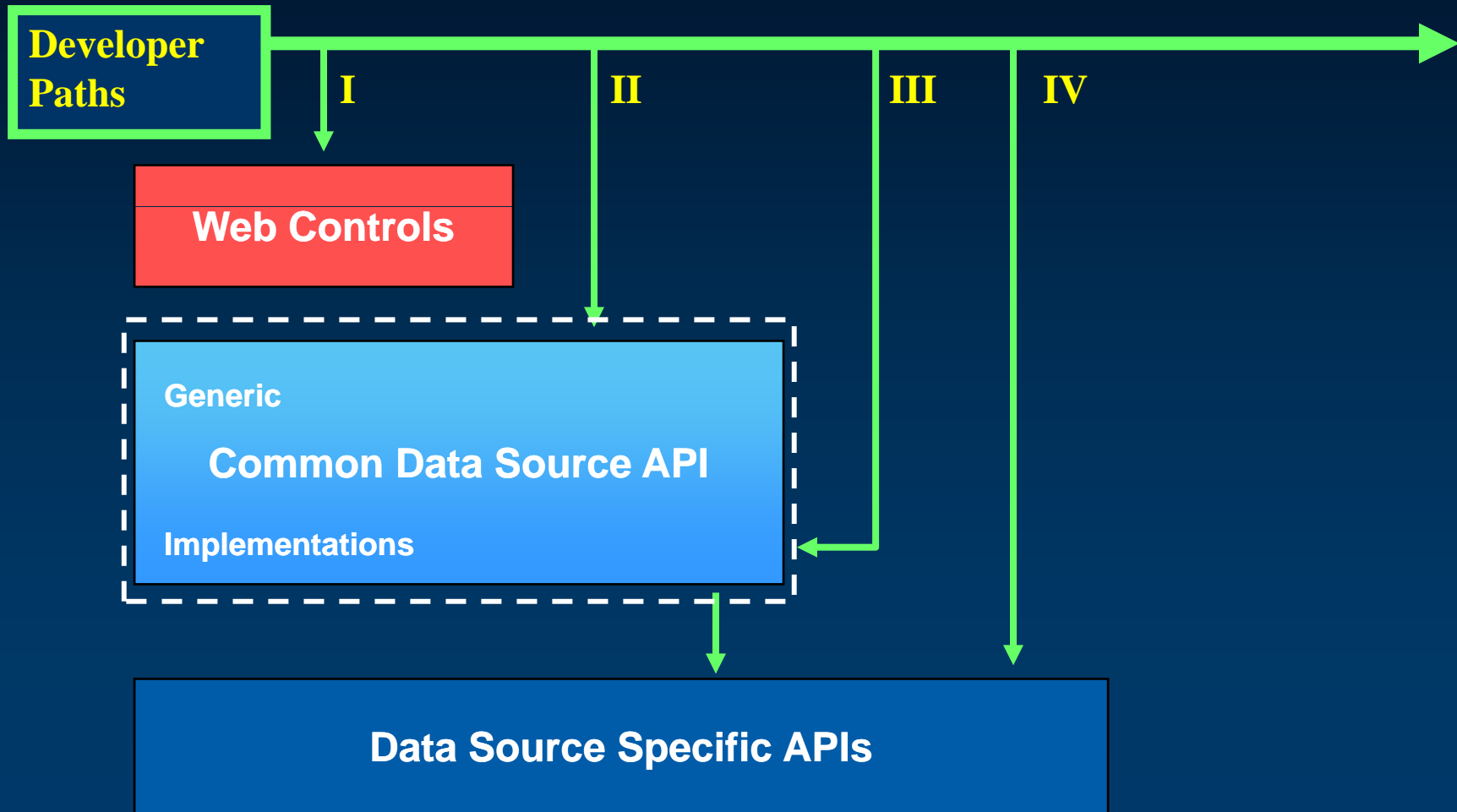


Migrating the UI, Commands and Tools

- **Use web controls to replace desktop components**
- **Use ADF framework to replace custom tools**
 - **ICommand and ITool**
- **Steps**
 - 1. Start with the Map template**
 - 2. Create new buttons and tools**
 - 3. Implement client-side action (JavaScript or DTHML)**
 - 4. Implement sever-side action**
 - 1. IMapServerCommandAction or IMapServerToolAction**
 - 2. Add ADF or ArcObjects logic if necessary**



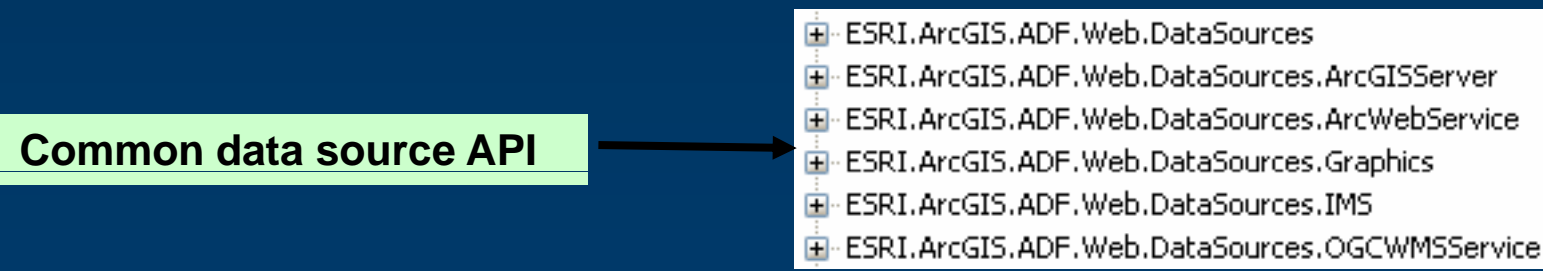
ArcGIS Server Web ADF – Development Paths



The Common Datasource API

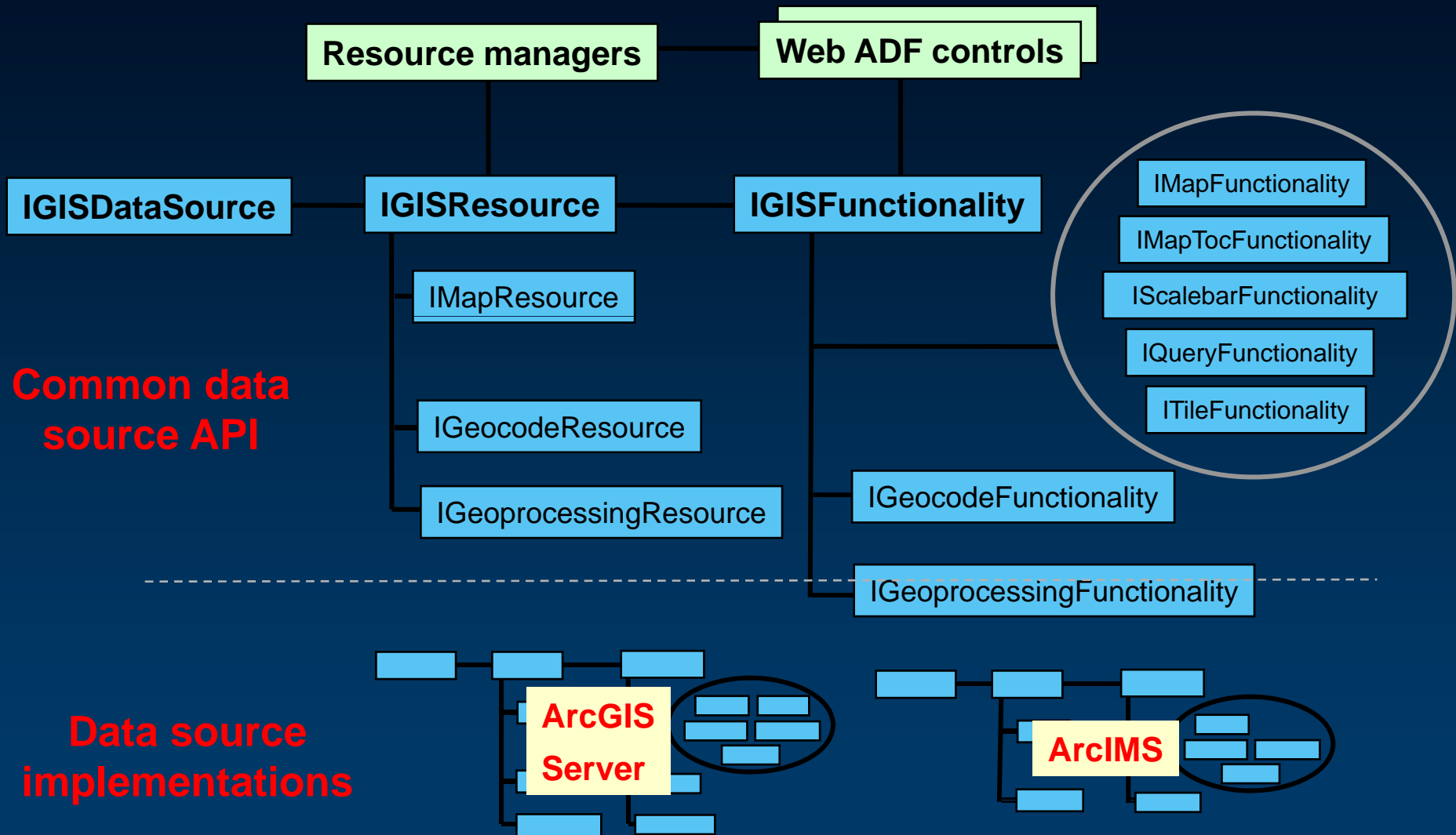
- **ArcGIS Server supports multiple data sources**
 - ArcGIS Server, ArcIMS, ArcWeb services, OGC, Graphics
- **Common Datasource API**
 - Pure .NET classes for the Web ADF
 - Access and interact with all data sources the same way!
- **Provides different functionalities - query, find, identity...**

Common data source API



```
ESRI.ArcGIS.ADF.Web.DataSources
ESRI.ArcGIS.ADF.Web.DataSources.ArcGISServer
ESRI.ArcGIS.ADF.Web.DataSources.ArcWebService
ESRI.ArcGIS.ADF.Web.DataSources.Graphics
ESRI.ArcGIS.ADF.Web.DataSources.IMS
ESRI.ArcGIS.ADF.Web.DataSources.OCWMSService
```

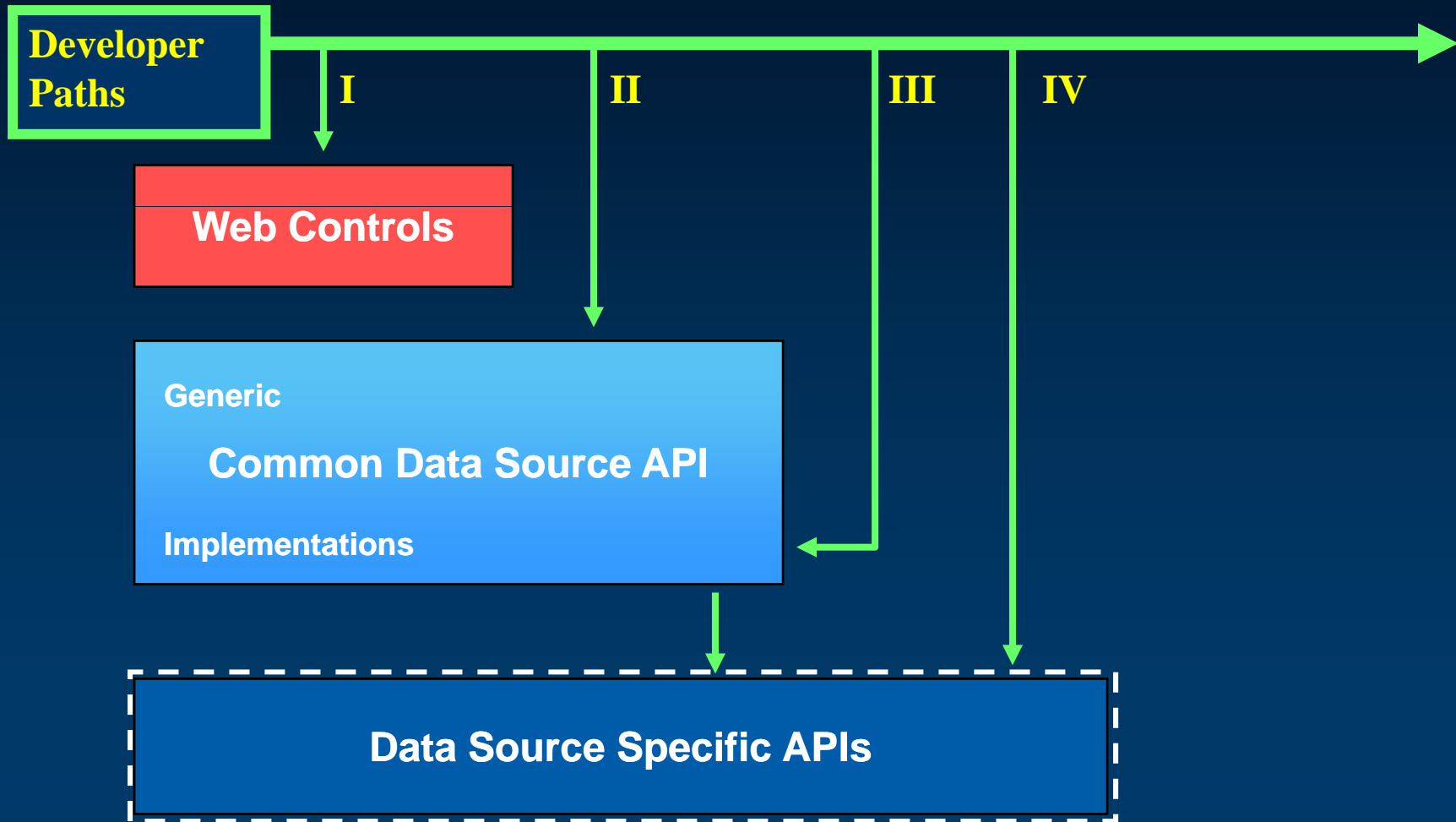
The Common Datasource API



Using the Common Datasource API

- Where possible, replace ArcObjects code
- Examples – Query, Identify, Find
- Steps
 1. Identify the ArcObjects code to be replaced
 2. Implement the Common API
 3. Reuse code to access other data sources
- Review SDK, code samples and OMDs...

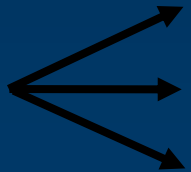
ArcGIS Server Web ADF – Development Paths



The Datasource Specific APIs – ArcGIS Server

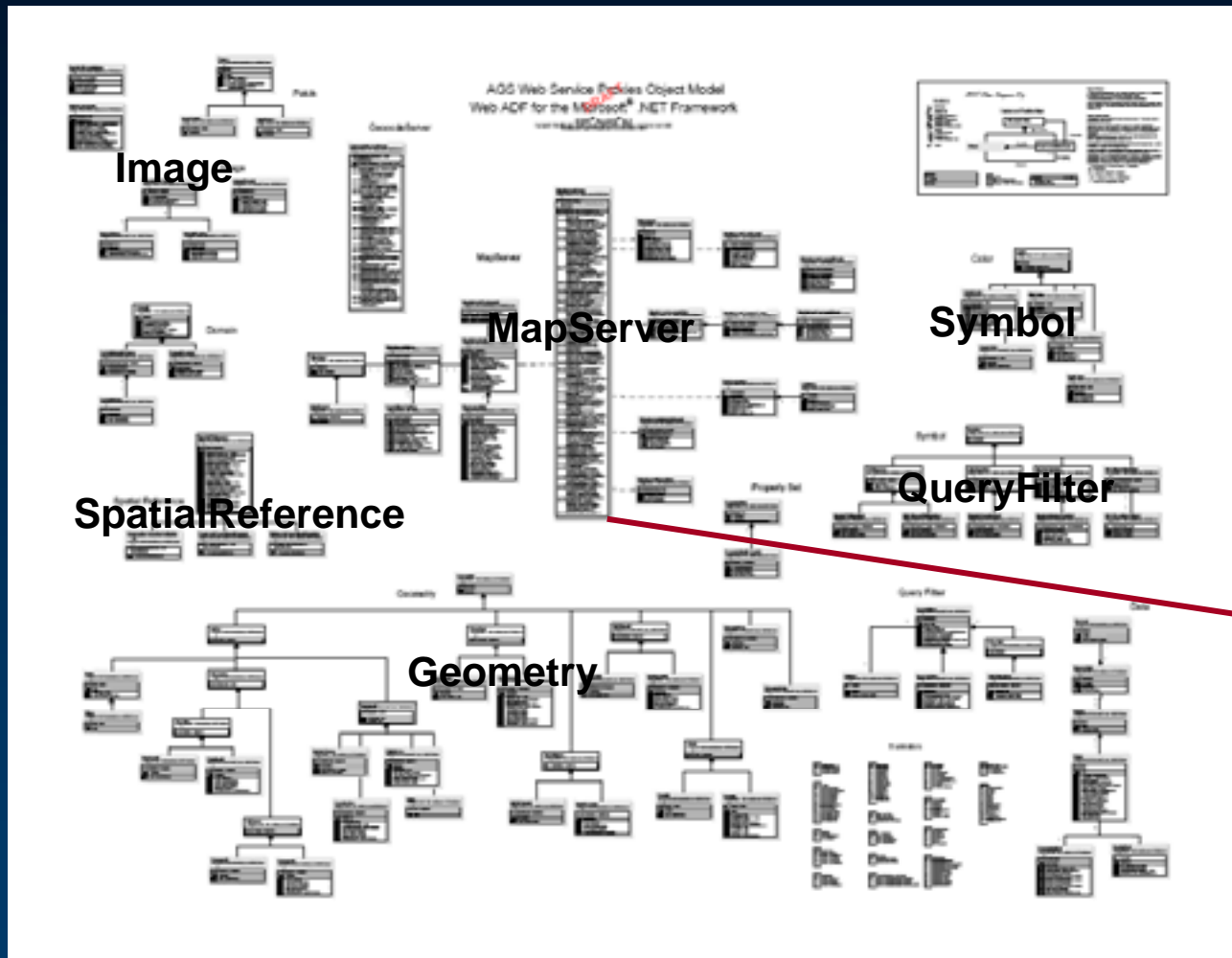
- Use when functionality is not available in Common API
- ArcGIS Server SOAP API
 - Gives you access to a Local ArcGIS Server Object
 - Exposes a sub-set of ArcObjects functionality
 - .NET objects called “Value objects”
 - Stateless and serializable
 - Optimized

Data source-specific APIs



```
ESRI.ArcGIS.ADF.ArcGISServer
ESRI.ArcGIS.ADF.ArcGISServer.Editor
ESRI.ArcGIS.ADF.ArcWebService
ESRI.ArcGIS.ADF.Connection
ESRI.ArcGIS.ADF.IMS
```

The ArcGIS Server SOAP API – Value Objects

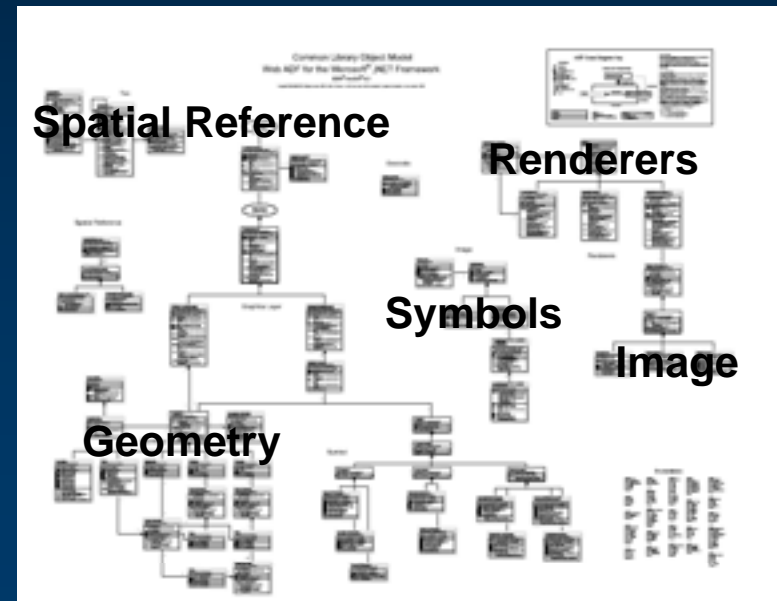


ArcObjects



The Datasource Specific APIs – Graphics

- **ArcGIS Server Graphics API**
 - .NET classes for local graphics operations for the Map
 - Graphics, geometries and renderers
 - Implemented as ADO.NET tables
- **Functionality**
 - Independent data source
 - Contains multiple layers
 - Can be queried
 - Refreshed independently
 - Can be used with other AJAX controls



Using the Datasource Specific APIs - Graphics

- **Replace graphics operations for the desktop**
 - Any drawing of graphics to the screen – buffers, selection...
 - Alternative for GraphicsContainer concept in ArcObjects
- **Replace IMapDescription::customGraphics (9.1)**
- **Steps**
 1. Identify the ArcObjects code to be replaced
 2. Implement the Graphics API
 3. Implement AJAX if necessary...
- **Review SDK, code samples and OMDs...**



ArcGIS Desktop vs ArcGIS Server Comparison

- **ArcGIS Desktop/Engine**

- **Commands, tools and toolbar**
- **TOC**
- **DockableWindow**
- **Map and PageLayout**
- **Extension**
- **ArcObjects**
- **Graphics**

- **ArcGIS Server Web ADF**

- **Buttons, tools, toolbar and tasks**
- **TOC**
- **FloatingPanel, Treeviewplus**
- **Map and PageLayout**
- **SOE**
- **Common API**
- **Datasource Specific API**
- **ArcObjects**
- **Graphics API**

Summary

- Two parts to migrating Desktop applications to Server
 1. Migrating pure ArcObjects code
 - ASP.NET, DCOM and Utility objects (9.1)
 - SOE and Geoprocessing (9.2)
 2. Migrating the User Interface
 - ArcGIS Server Web controls
 - ArcGIS Server Web ADF framework (Comamnds and Tools)
- Replacing ArcObjects
 - Common Datasouce API (Query, Identify, Find...)
 - Datasource Specific API (ArcGIS Server SOAP API and Graphics)

***Leverage the new Web ADF APIs as much as possible!**

Presentation materials

- PowerPoint presentation and code are posted on the conference web site
 - <http://www.esri.com/events/devsummit/index.html>
- EDN – downloads and videos

Further questions?

- **TECH-TALK AREAS**

- **What:** Further opportunity to discuss questions and concerns with presenters and subject matter experts

- **Where:**

- **When:** during the next 30 minutes

- **ESRI Showcase**

- **Meet the teams**

- **ESRI Developers Network (EDN) website**

- <http://edn.esri.com>

The end – Thank you!