2011 Esri Developer Summit
Palm Springs, CA

Extending ArcGIS Server Services using .NET

Sterling Quinn

Rex Hansen

Esri Training for ArcGIS Server Developers

http://www.esri.com/training

- Instructor-Led Courses
  - Introduction to ArcGIS Server
  - Creating Effective Web Applications Using ArcGIS Server
  - System Architecture Design Strategies
- Online Training Seminars
  - Free, one-hour presentation and demos by Esri technical experts
  - Live seminar broadcast on a new topic every month

In this session…

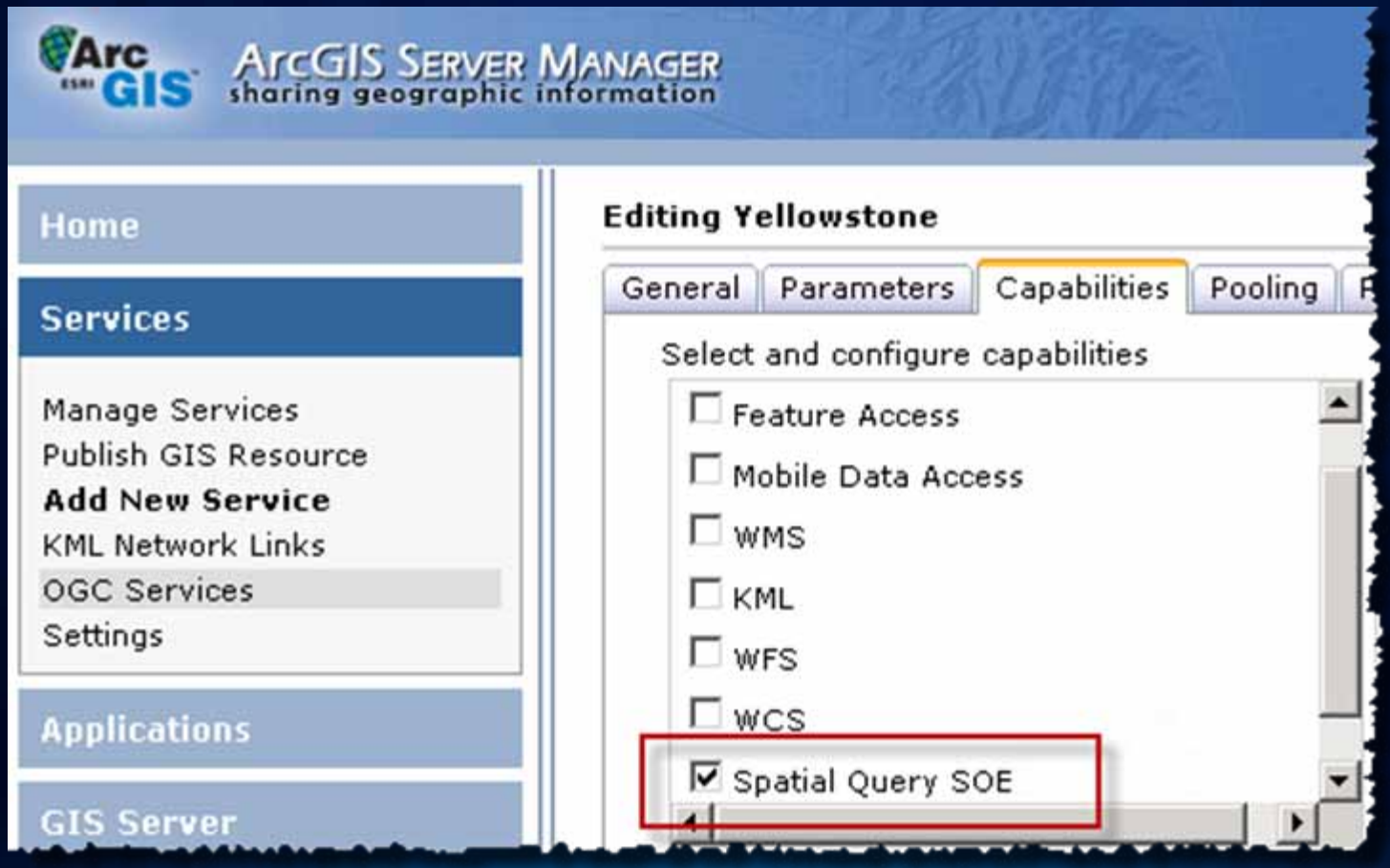
- Brief introduction to server object extensions (SOEs)
- Writing the code
- SOE registration and deployment
- Using an SOE in a client app
- Getting help

Brief introduction to SOEs

What is an SOE?

- Server Object Extension
- Lets you extend the base functionality of ArcGIS Server using ArcObjects
- Functionally a .NET class library exposed through COM that implements certain interfaces and is installed on each SOC machine

You enable SOEs individually for each service



When do you need an SOE?

- Extend ArcGIS Server with your own business logic
  - Analysis beyond out of the box geoprocessing tools
  - Functions beyond what's in the Esri Web APIs
  - Fine-grained ArcObjects operations not available through other means

Alternatives to SOEs

- Geoprocessing services
  - Arcpy.mapping scripts for printing and map layouts
- Geometry service

Demo: An SOE in action

SOE in an app

SOE in the Services Directory

Why are SOEs important?

- Fast
- Pluggable
- Functional
- The way of the future for fine-grained ArcObjects access through ArcGIS Server

Why are SOEs growing in importance?

"ArcGIS Server 10.0 is the last ArcGIS Server release with support for local connections from Web ADF applications. . . .We recommend developers using local connections for accessing fine-grained ArcObjects migrate their business logic to Server Object Extensions."

- Deprecation plan for ArcGIS 10.0 and 10.1

http://downloads2.esri.com/support/TechArticles/ArcGIS10and101Deprecation_Plan.pdf

Ways to implement an SOE

- REST Web service SOE
- SOAP Web service SOE
- Traditional DCOM SOE (not available at 10.1)

Today we'll focus on REST Web service SOEs

- Easily invoked by the Esri Web APIs
  - JavaScript
  - Flex
  - Silverlight
  - iOS
  - Etc.

What you have to understand to develop an SOE

- .NET and COM technologies
- REST or SOAP communication
- ArcObjects
- Optional technologies for custom property pages
  - HTML and JavaScript for Manager pages
  - Windows Forms and ArcCatalog customization for ArcCatalog pages

Demo: Another SOE in action

Spatial Query REST

Steps for developing and deploying an SOE

1. Write the code
2. Register the SOE with COM on each SOC machine
3. Register the SOE with ArcGIS Server
4. Optionally write a property page for Manager, ArcCatalog, or both
5. Publish a service and enable the SOE on it
6. Use the SOE in a client application you develop

Writing the code

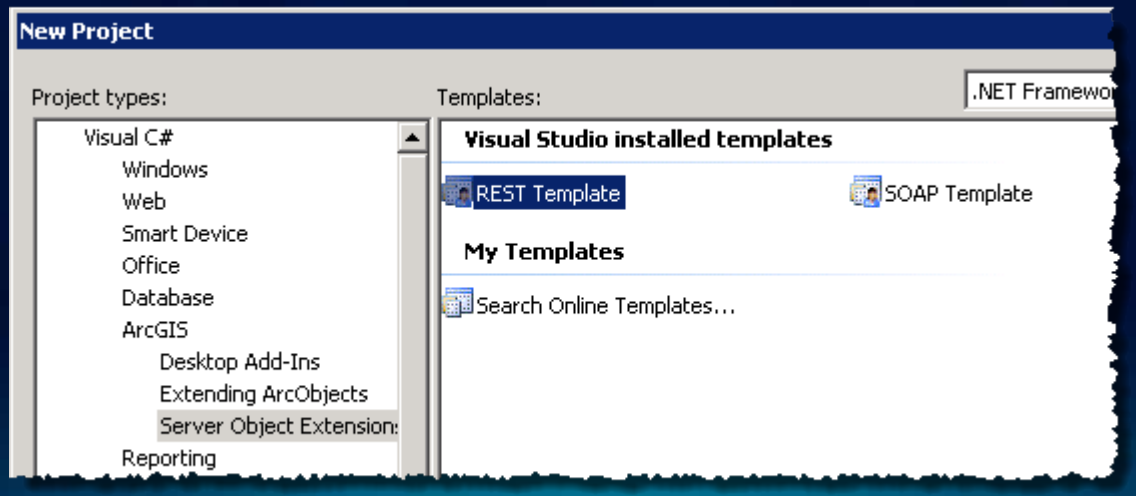
An SOE implements the following interfaces

- IServerObjectExtension
- IObjectConstruct (put initialization logic here)
- IRESTRequestHandler *

* Other interfaces are used with SOAP and DCOM SOEs

Getting started with the code

- REST SOE template in Visual Studio
- Installed with .NET ArcObjects SDK
- Has an example SOE stubbed out for you



Demo: A tour of the REST SOE template

Visual Studio REST SOE template

What happens in the template code

- Implements required interfaces
- Creates the schema
- Handles resources and operations

ESRI.ArcGIS.SOESupport

- Helper classes
  - SoeRestImpl for REST
  - SoeSoapImpl for SOAP
- Allows for
  - Message de/serialization
  - Error handling
  - Logging

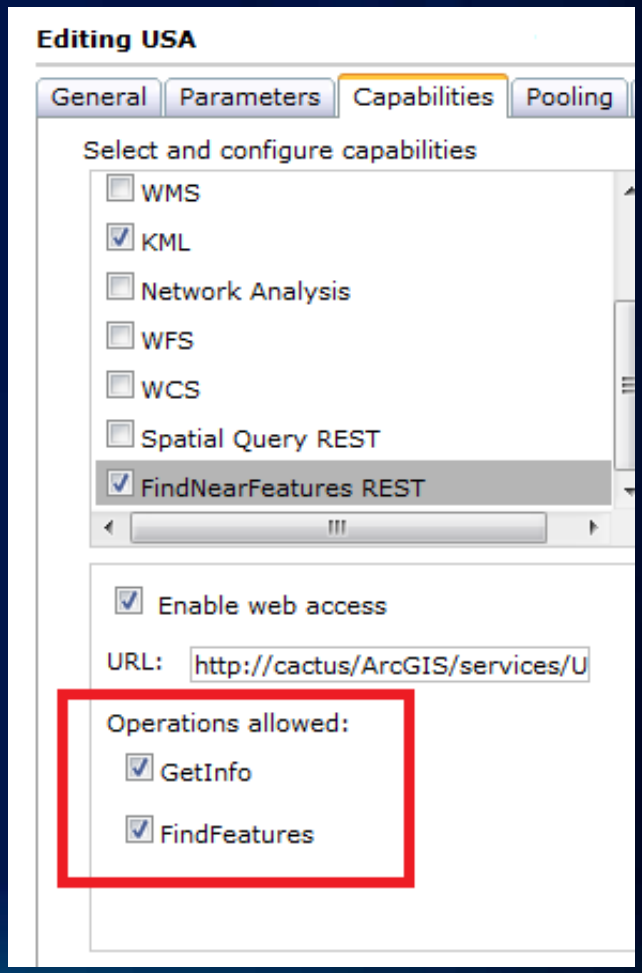
Creating the schema

- What resources and operations will your SOE expose?
  - Resources give you back information
  - Operations do something
- Snap them together to make a schema
- You may need to diagram this

SOE capabilities (or “Operations allowed”)

- Determine which schema items clients can access
- Configured as a parameter on schema items

```
RestResource customLayerResource =
    new RestResource("customLayers",
    true, CustomLayer, "GetInfo");
```



Demo: Exploring REST SOE schemas

Spatial Query REST SOE

Find Near Features REST SOE

Writing handler functions

- Each resource and operation has a handler
- Operation handlers are where most of your business logic is invoked

Working with JSON

- Your handlers have to…
  - Deserialize incoming JSON
  - Do something with it (often using ArcObjects)
  - Serialize the output into a JSON response
- Important class: ESRI.ArcGIS.SOESupport.JsonObject

Serialization methods on SOESupport.JsonObject

- **Deserialization (receiving a request)**

```
TryGetArray(string, out object[])
TryGetAsBoolean(string, out bool?)
TryGetAsDate(string, out System.DateTime?)
TryGetAsDouble(string, out double?)
TryGetAsLong(string, out long?)
TryGetJsonObject(string, out ESRI.ArcGIS.SOESupport.JsonObject)
TryGetObject(string, out object)
TryGetString(string, out string)
```

- **Serialization (preparing a response)**

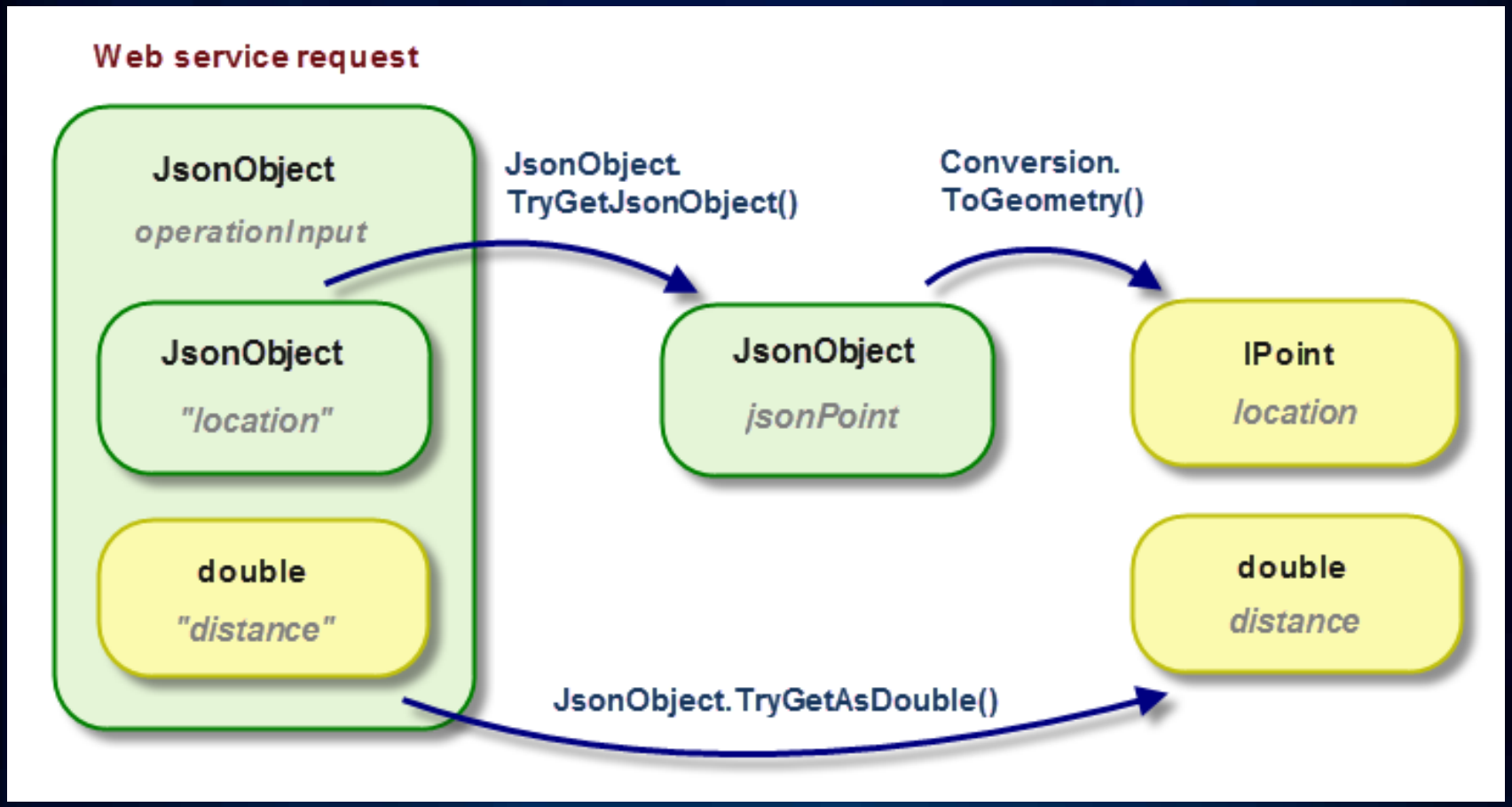
```
AddArray(string, object[])
AddBoolean(string, bool)
AddDate(string, System.DateTime)
AddDouble(string, double)
AddJsonObject(string, ESRI.ArcGIS.SOESupport.JsonObject)
AddLong(string, long)
AddObject(string, object)
AddString(string, string)
```

Conversion methods on SOESupport.Conversion

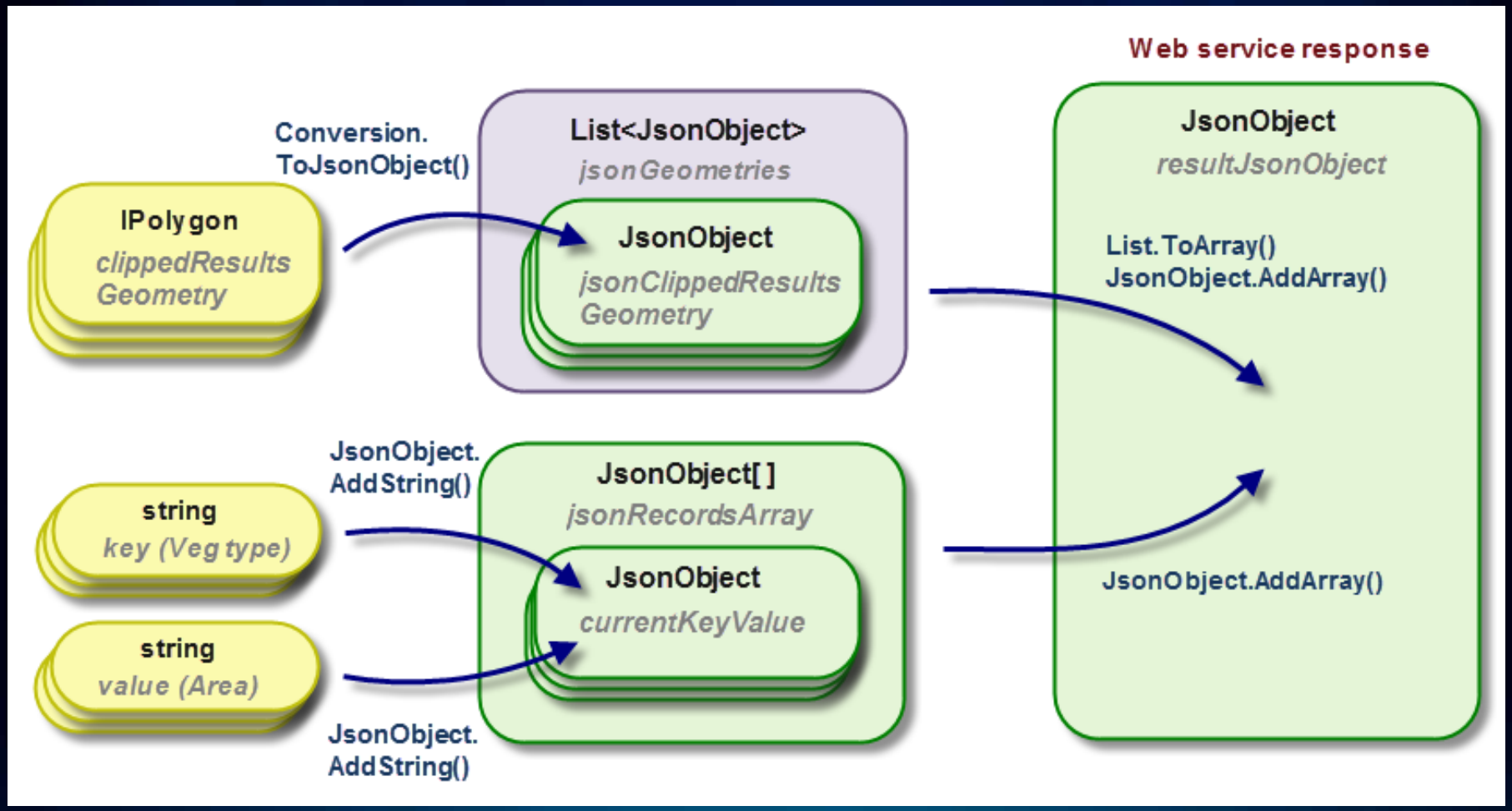
```
Conversion()
ToGeometry(ESRI.ArcGIS.SOESupport.JsonObject, ESRI.ArcGIS.Geometry.esriGeometryType)
ToGeometry(string, ESRI.ArcGIS.Geometry.esriGeometryType)
ToJson(ESRI.ArcGIS.Geodatabase.IRecordSet)
ToJson(ESRI.ArcGIS.Geometry.IGeometry)
ToJsonObject(ESRI.ArcGIS.Geometry.IGeometry)
ToSpatialReference(string)
```

- **These are helpful with (de)serialization**

Deserialization in the SpatialQueryREST sample



Serialization in the SpatialQueryREST sample



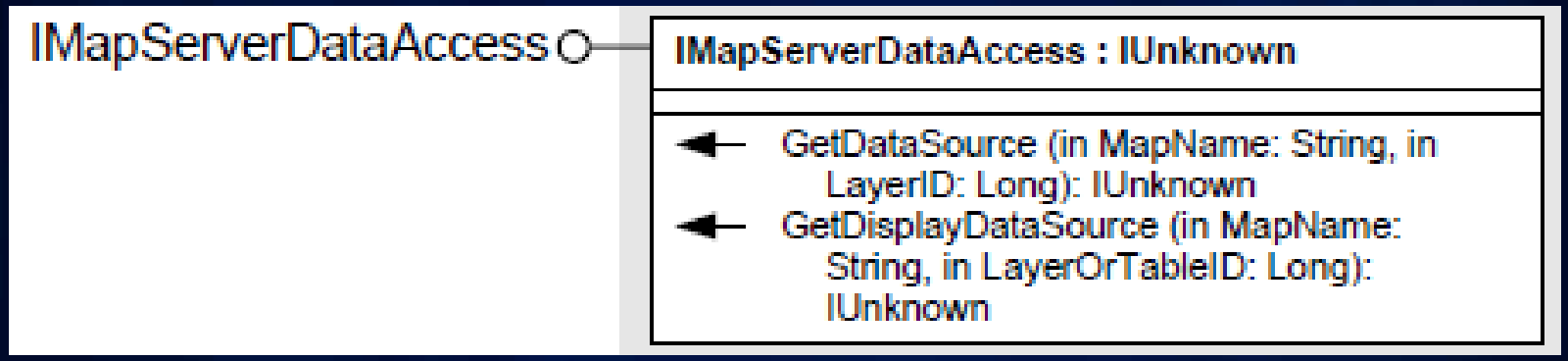
Demo: JSON serialization

Spatial Query REST SOE

Supporting MSD-based services

- Avoid MXD-specific ArcObjects (IMap, ILayer, etc.)
  - Carto object model
- Use IMapServerDataAccess to get at underlying data of MSDs
  - IFeatureClass, IRaster, or ITable
- When you register the SOE with ArcGIS Server, set “SupportsMSD” to “true”
  - `serverObjectExtensionType.Info.SetProperty("SupportsMSD", "true");`

IMapServerDataAccess



Demo: MSD support

Spatial Query REST SOE

SOE registration and deployment

Registering the SOE on each SOC

- Sign and build the project
- Register on each SOC machine with this command:

```
regasm <path to DLL> /codebase
```

- SOC account needs access to DLL location

Registering the SOE with ArcGIS Server

- Done through code using IServerObjectAdmin2.AddExtensionType()
- Re-use the registration app in the SOE SDK samples
- Verify registration by viewing <ArcGIS Server install>\server\system\ServerTypesExt.dat

Writing a property page

- Manager and ArcCatalog support pluggable “property pages” for SOEs
- Useful if server admin must set properties at the service level
- Best for advanced developers
  - Some examples available in the SDK
  - Must register these with COM as well

Creating a service and enabling an SOE on it

- Publish a service
- Enable the SOE in the Capabilities tab of the Service Properties
- Choose the allowed operations (“capabilites” in SOE-speak)

Testing an SOE in the Services Directory

- Services Directory detects parameters
- Allows for easy testing

Using an SOE in a client app

Using the SOE in a client application

- Use types designed for HTTP requests to Web services
- Different techniques for JavaScript, Flex, and Silverlight

JavaScript clients to REST SOEs

- Use esri.request
- Set up content variable with JSON inputs
- Pass content to esri.request, then work with the response object
- Online SDK has a new sample

Flex clients to REST SOEs

- Extend the BaseTask class
- url property on BaseTask should point at the SOE
- BaseTask.sendURLVariables to call the SOE

Silverlight clients to REST SOEs

- Two choices
  - DataContractJsonSerializer
  - JsonObject and LINQ

- Silverlight SDK has samples
  - For example: http://help.arcgis.com/en/webapi/silverlight/samples/start.htm#SOEElevationData

Getting help

Where can I get more help?

- Recording of this session will be on Resource Center
- SOE doc is in the ArcObjects SDK developer help

Recent additions to the SOE help

- Simplified conceptual help for .NET SOEs
- Added more detail on REST SOEs and the templates
- Added a walkthrough for creating a REST SOE
  - Includes new sample REST SOE that does a spatial query
  - Shows a JavaScript client for the SOE
  - http://esriurl.com/spatialqueryrest

Demo: New online help for SOEs

Questions?

esri