

ESRI China (Beijing)
Geographic Thinking

ArcGIS RIA开发实践

ESRI中国（北京）有限公司
吴泳锋

ESRI China (Beijing)
Geographic Thinking

【背景篇】

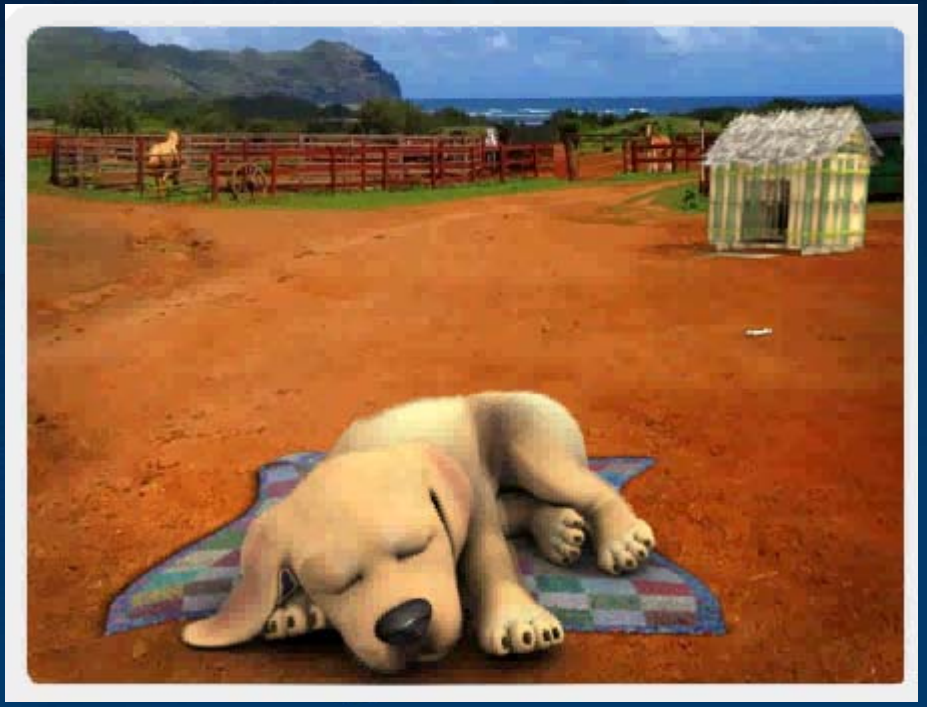
RIA介绍

- RIA相比传统Web的优势
- RIA的局限性
- 目前主流RIA技术及特点

ESRI China (Beijing)
Geographic Thinking

RIA相比传统Web的优势

- 表现力丰富
- 网络效率高
- 交互能力强
- 面向操作系统和浏览器透明
- 沙箱提供更可靠的安全性
- 易于与现有系统集成



RIA的局限性

- 客户端引擎的下载和安装
- 目前难于做SEO

ESRI China (Beijing)
Geographic Thinking

目前主流RIA技术及特点

- Adobe Flex / Flash
- Microsoft Silverlight
- Sun JavaFX
- Curl

ESRI China (Beijing)
Geographic Thinking

ArcGIS Server的REST接口

- REST的概念
- 通过REST的接口使用ArcGIS功能

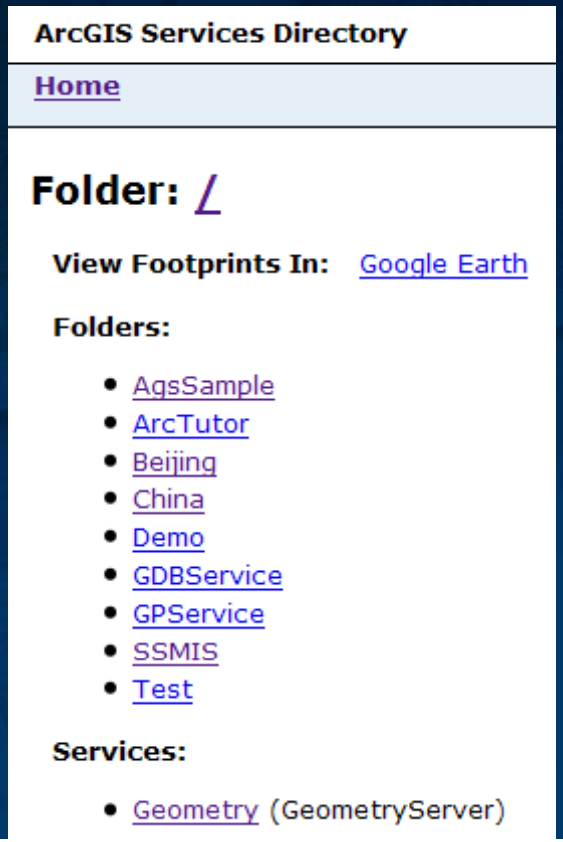
ESRI China (Beijing)
Geographic Thinking

REST的概念

- 资源通过URI来指定和操作
- 对资源的操作包括获取、创建、修改和删除资源，这些操作正好对应HTTP协议提供的GET、POST、PUT和DELETE方法
- 连接是无状态性的
- 能够利用Cache机制来提高性能

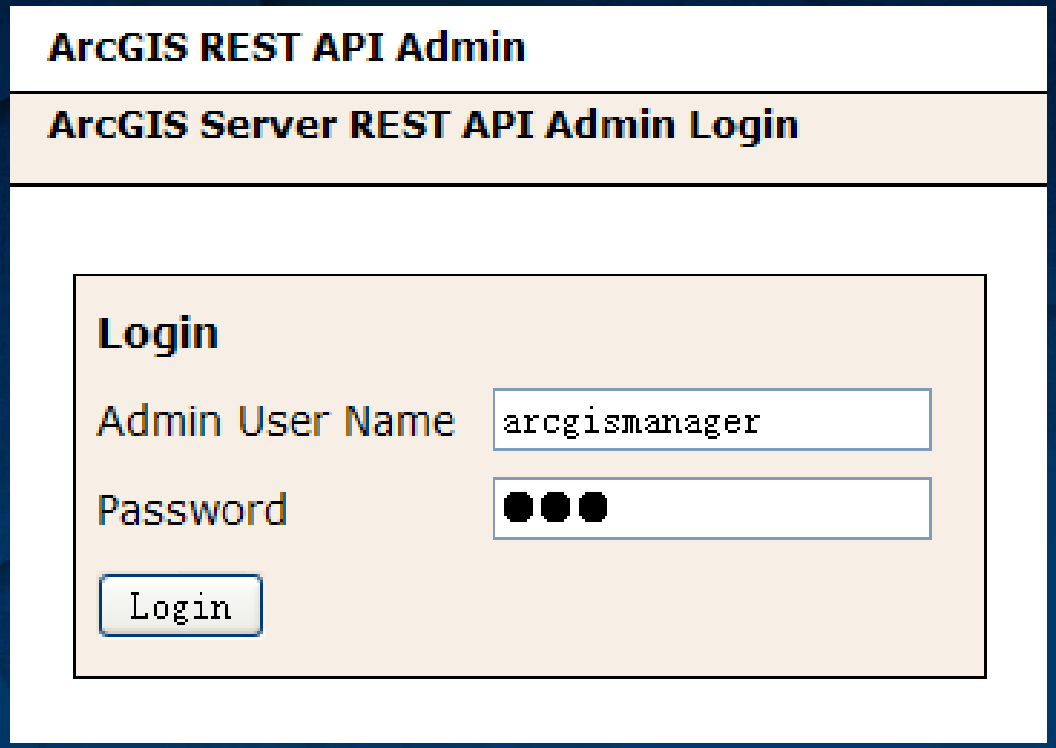
通过REST的接口使用ArcGIS功能

- REST服务目录
  Java: http://<host>:8399/argis/rest
  .Net: http://<host>/arcgis/rest



ESRI China (Beijing)
Geographic Thinking

- 清空REST的缓存
- REST管理界面
  http://<host>[:8399]/arcgis/rest/admin



ESRI China (Beijing)
Geographic Thinking

ArcGIS REST API Admin

Admin

Clear Cache Options

Clear Cache Now **(Clears the REST API Cache.)**

Clear Cache Policy Settings

Manual

Cache never expires automatically. Manually clear the cache from the admin console.

Scheduled

Schedule the cache to be cleared daily at 00:00 hours (24 hour clock - HH:MM)

Periodic

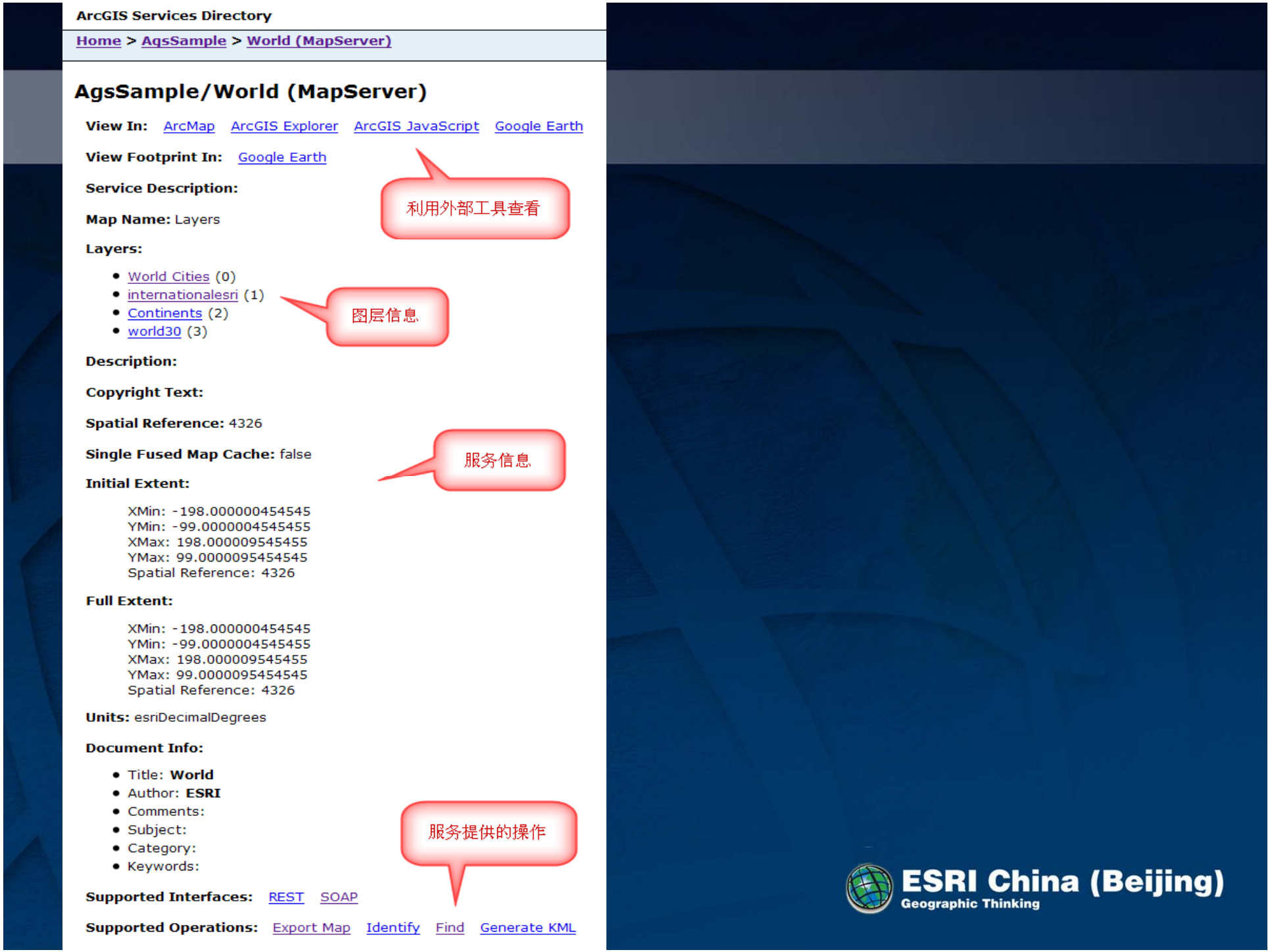
Clear the cache periodically every 60 minutes

Apply Settings

ESRI China (Beijing)
Geographic Thinking

- 某个地图服务REST URL：
  http://localhost:8399/arcgis/rest/services/AgsSample/World/MapServer

ESRI China (Beijing)
Geographic Thinking


ArcGIS Services Directory
Home > AgsSample > World (MapServer)
AgsSample/World (MapServer)
View In: ArcMap ArcGIS Explorer ArcGIS JavaScript Google Earth
View Footprint In: Google Earth
Service Description:
Map Name: Layers
Layers:
World Cities (0)
internationalesri (1)
Continents (2)
world30 (3)
Description:
Copyright Text:
Spatial Reference: 4326
Single Fused Map Cache: false
Initial Extent:
XMin: -198.000000454545
YMin: -99.0000004545455
XMax: 198.000009545455
YMax: 99.0000095454545
Spatial Reference: 4326
Full Extent:
XMin: -198.000000454545
YMin: -99.0000004545455
XMax: 198.000009545455
YMax: 99.0000095454545
Spatial Reference: 4326
Units: esriDecimalDegrees
Document Info:
Title: World
Author: ESRI
Comments:
Subject:
Category:
Keywords:
Supported Interfaces: REST SOAP
Supported Operations: Export Map Identify Find Generate KML
利用外部工具查看
图层信息
服务信息
服务提供的操作
ESRI China (Beijing)
Geographic Thinking

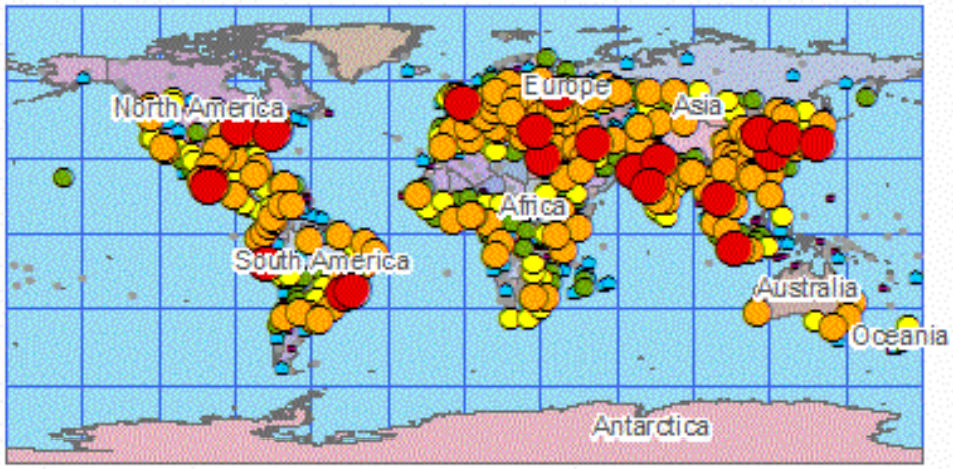
- 点击“Export Map”链接
- 跳转URL：
  http://localhost:8399/arcgis/rest/services/AgsSample/World/MapServer/export?bbox=-198.000000454545,-99.0000004545455,198.000009545455,99.0000095454545

ESRI China (Beijing)
Geographic Thinking

ArcGIS Services Directory

Home > AgsSample > World (MapServer)

Export Map Image: AgsSample/World



Width: 400
Height: 400
Extent:

XMin: -198.000000454545
YMin: -198.000000454546
XMax: 198.000009545455
YMax: 198.000009545455
Spatial Reference: 4326

Scale: 4.16060621533301E8

- 总结：
  ArcGIS Server REST接口中，请求服务的URL一般应该是这样的：

  http://<host>[:<port>]/arcgis/rest/services/<服务>/<服务类型>/<操作>?<参数>

ESRI China (Beijing)
Geographic Thinking



ESRI China (Beijing)
Geographic Thinking

【Flex篇】



ESRI China (Beijing)
Geographic Thinking

【ArcGIS Flex API 基础】

Flex的历史和现状

- Flash
- Flex
- Adobe与Macromedia

ESRI China (Beijing)
Geographic Thinking

ArcGIS Flex API概述

- ArcGIS在RIA领域的第一个产品
- 开发运行于浏览器中的Web应用或者运行于桌面的AIR应用
- 非常便捷地使用地图功能和REST接口提供的GIS查询及分析功能
- 专注于GIS功能的实现而不重复地创造组件

一些基本概念

- MapServiceLayer
- GraphicLayer与Graphic
- Geometry
- Symbol

ESRI China (Beijing)
Geographic Thinking

MapServiceLayer

- **对应ArcGIS 发布的地图服务**
- **不同种类的地图服务将会对应不同的MapServiceLayer**

ArcGISDynamicMapServiceLayer	ArcGIS Server发布的动态地图服务
ArcGISTiledMapServiceLayer	ArcGIS Server发布的切片地图服务
ArcGISImageServiceLayer	ArcGIS Image Server发布的影像服务
ArcIMSMapServiceLayer	ArcIMS发布的地图服务

GraphicLayer与Graphic

- 客户端的图层
- 根据各种情况动态地在客户端显示一些符号化的几何对象——Graphic
- Graphic是一个可视化的、可响应鼠标事件的客户端要素
- Graphic3个很重要的属性：geometry、symbol和attributes

Geometry

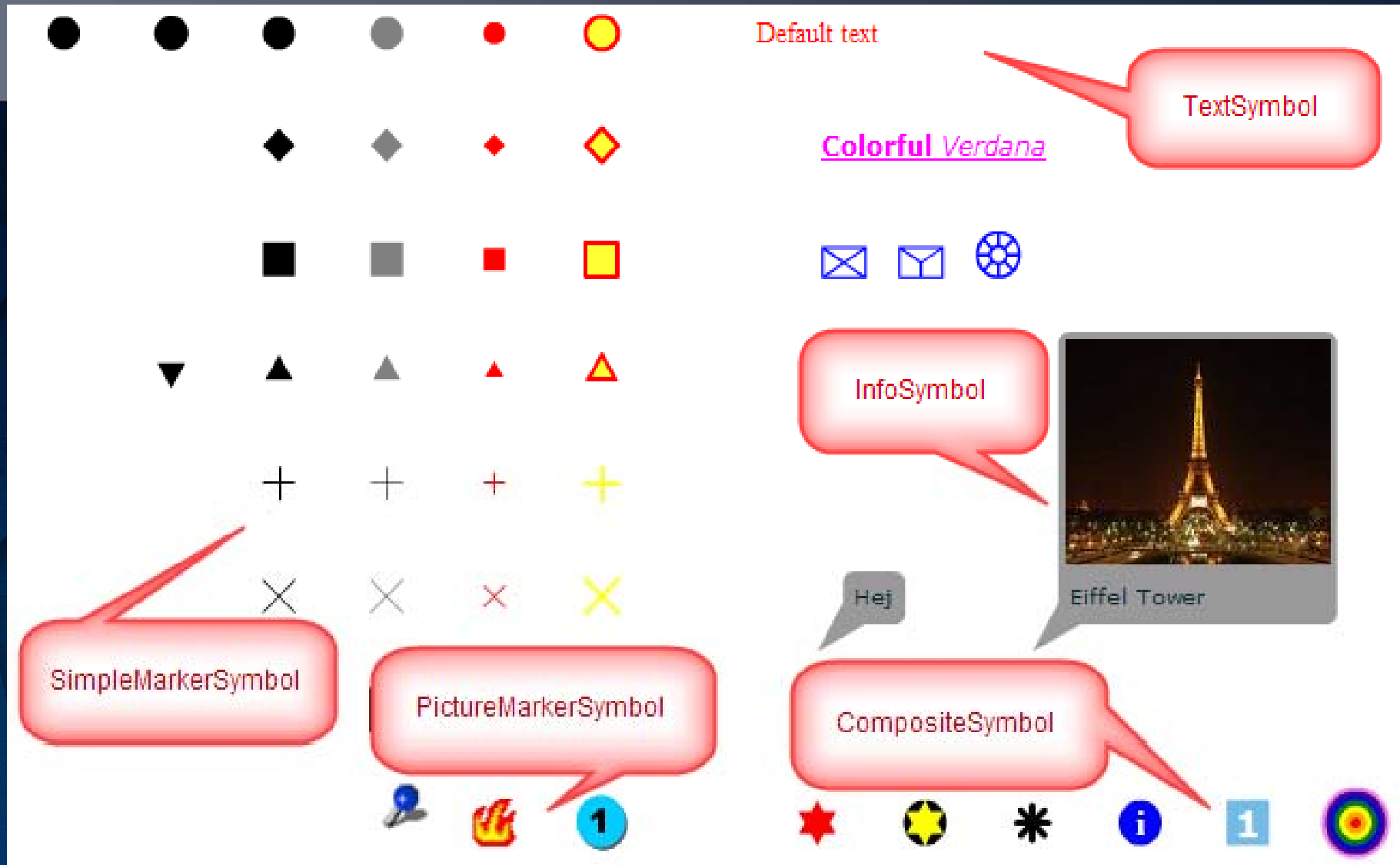
MapPoint	点
Multipoint	多点
Polyline	多段线
Polygon	多边形
Extent	边界范围

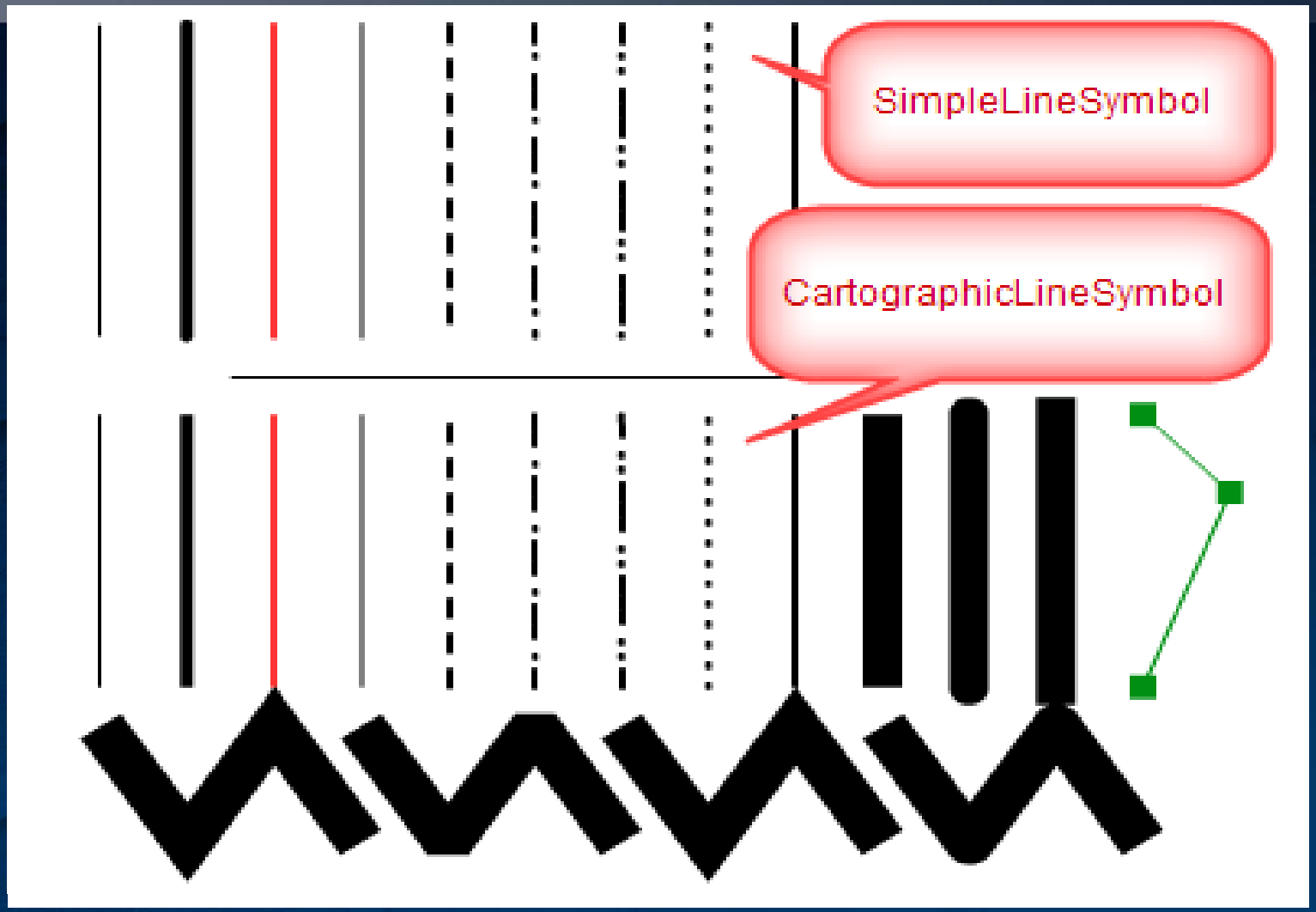
[1] 支持JPG、GIF、PNG、SWF、SVG格式
[2] 上面若干种符号的组合

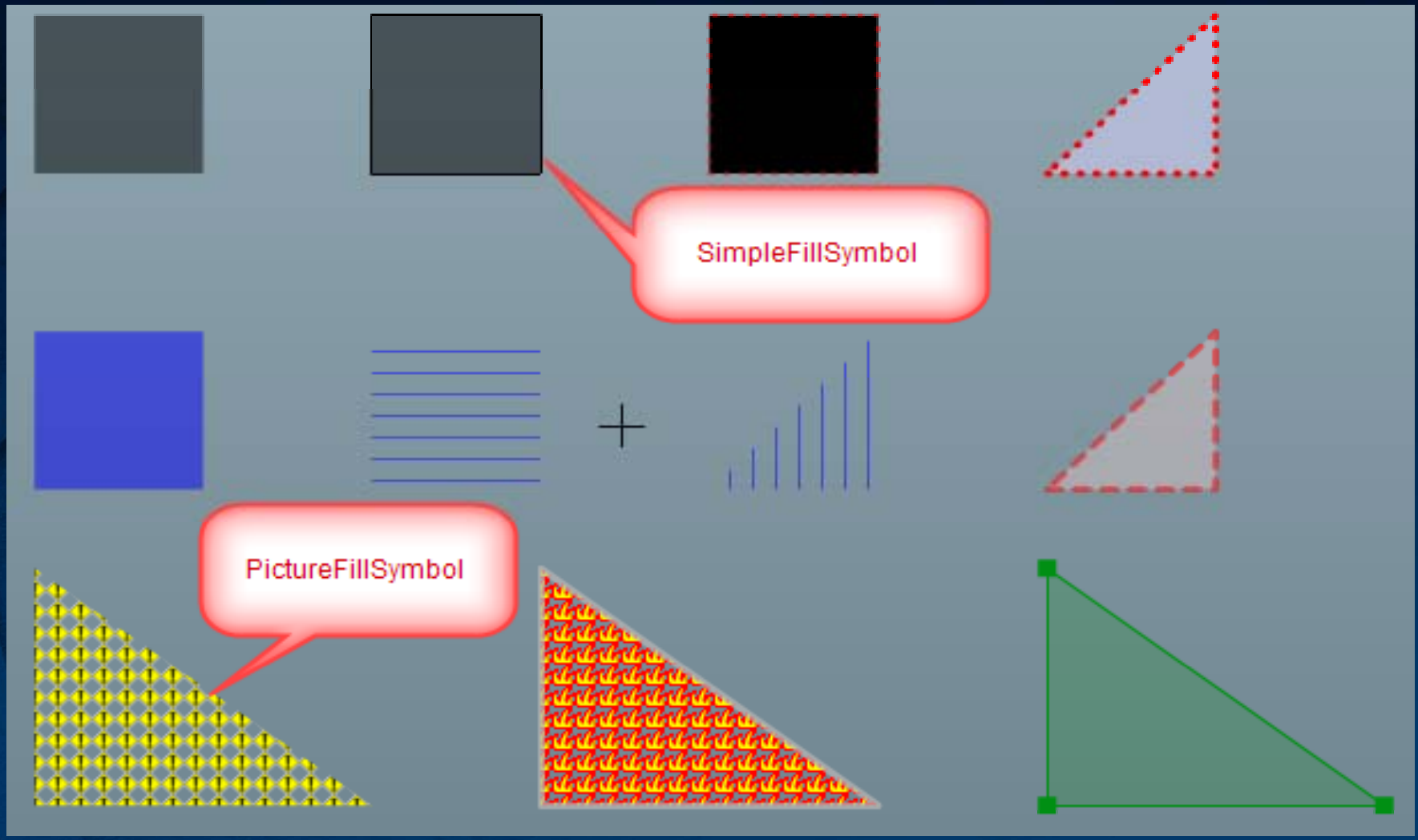
Symbol

- 定义了Geometry使用什么符号表现出来
- 一般MarkerSymbol、LineSymbol、FillSymbol
- 还有TextSymbol、InfoSymbol等其它符号帮助进行其它信息的辅助表达

SimpleMarkerSymbol	简单点符号
SimpleLineSymbol	简单线符号
SimpleFillSymbol	简单填充符号
CartographicLineSymbol	制图线符号，可定义端点、折点样式
PictureMarkerSymbol	图片点符号
PictureFillSymbol	图片填充符号
TextSymbol	文字符号
InfoSymbol	信息提示符号
CompositeSymbol	复合符号


Default text
TextSymbol
Colorful Verdana
InfoSymbol
Eiffel Tower
Hej
SimpleMarkerSymbol
PictureMarkerSymbol
CompositeSymbol


SimpleLineSymbol
CartographicLineSymbol



ESRI China (Beijing)
Geographic Thinking

地图功能组件

- Map组件
- Navigation工具
- Draw工具
- InfoWindow窗口

ESRI China (Beijing)
Geographic Thinking

Map组件

- ArcGIS Flex API中唯一的可视化组件
- ArcGIS Flex API中所有的操作和功能都会在Map中进行
- Map中包含各种Layer
- Map中的Layer有两种，一种是对应某个地图服务的MapServiceLayer、一种是用以绘制自定义要素的GraphicLayer

- 添加Map和Layer的代码：

```
<esri:Map id="map">
   <esri:ArcGISTiledMapServiceLayer
url="http://localhost:8399/arcgis/rest/services/China/map/MapServer"
/>
</esri:Map>
```



- 控制地图缩放的zoomSlider、地图比例尺scaleBar以及ESRI的logo
- 可以隐藏

- zoomSlider和scaleBar除了可以被隐藏掉，还可以通过定制样式来改变外观
- 以下的CSS代码将Map的zoomSlider移到了右边：

```
<mx:Style>
        .mapStyle{ navigation-style-name: navigationStyle;}
        .navigationStyle
        {
            top: 6;
            right: 6;
        }
</mx:Style>

<esri:Map id="map" styleName="mapStyle"/>
```



ESRI China (Beijing)
Geographic Thinking

- 地图四周的导航按钮
- 地图中心的十字符号
- 拉框缩放的方框样式



- Map有很灵活的可配置性
- 各种元素的外观也很容易定制
- 如果有特殊的需要，Map也可以通过隐藏所有元素，你则可以在此基础上加上自己的内容

ESRI China (Beijing)
Geographic Thinking

Navigation工具

- 提供了对Map进行导航的诸多功能
- 它可以实现漫游、拉框放大、拉框缩小等类似ADF中Tool的功能和前一视图、后一视图、全图等类似ADF中Command的功能
- 它并不是一个工具条，而是提供了一些功能，你可以把Flex组件绑定到Navigation的功能上

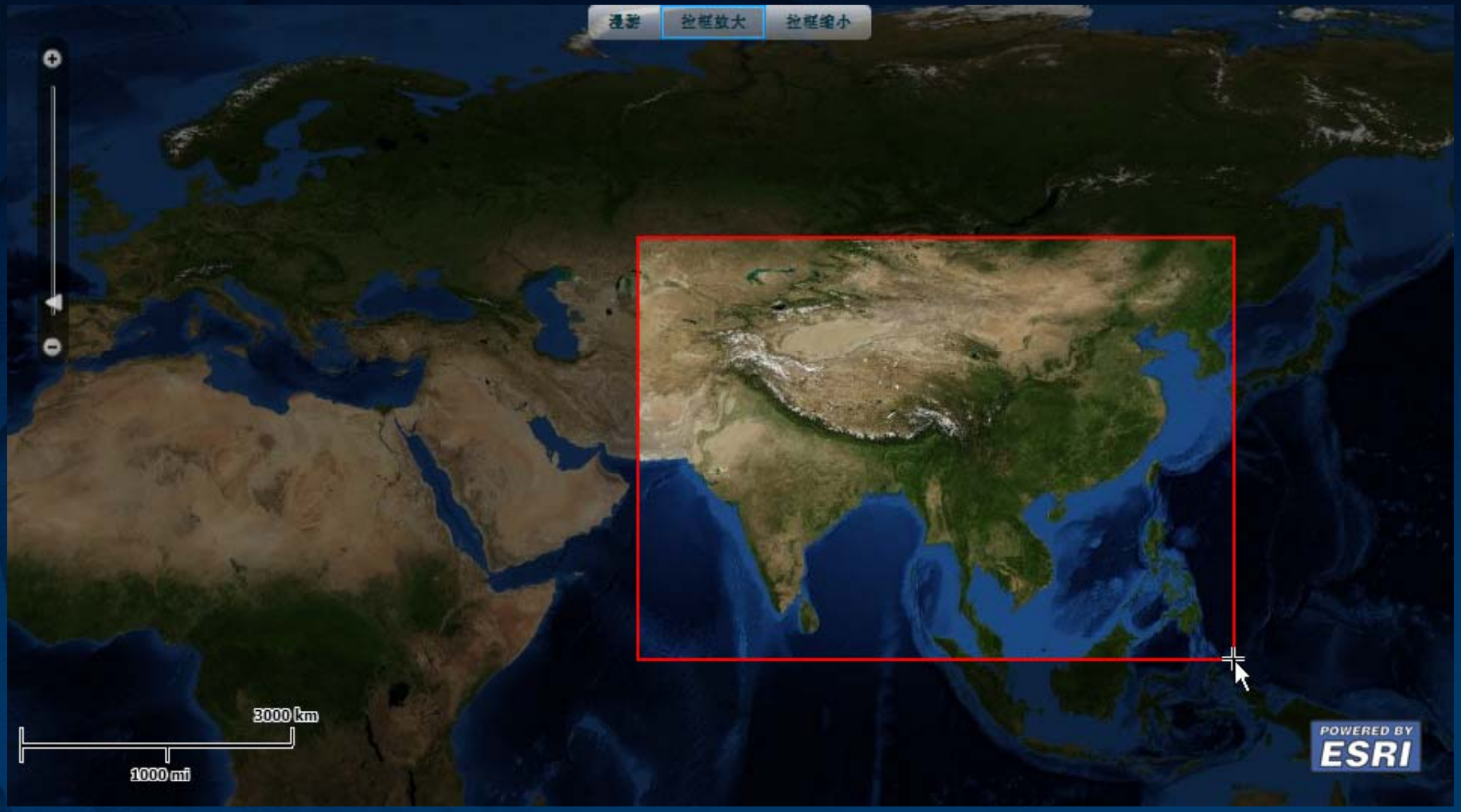
ESRI China (Beijing)
Geographic Thinking

- Navigation中漫游、拉框放大、拉框缩小等功能和工具条按钮的绑定
- 借助Flex中的ToggleButtonBar作为这些功能的宿主

```
<mx:ToggleButtonBar id="tbb" itemClick="itemClickHandler(event)">
    <mx:dataProvider>
        <mx:Array>
            <mx:String>漫游</mx:String>
            <mx:String>拉框放大</mx:String>
            <mx:String>拉框缩小</mx:String>
        </mx:Array>
    </mx:dataProvider>
</mx:ToggleButtonBar>
```

```
<mx:Script>
   <![CDATA[
      private function itemClickHandler(event:ItemClickEvent):void
      {
         switch (event.index)
         {
            case 0:{ nav.activate(Navigation.PAN , true);break; }
            case 1: {nav.activate(Navigation.ZOOM_IN , true); break; }
            case 2: {nav.activate(Navigation.ZOOM_OUT , true); break; }
         }
      }
   ]]>
</mx:Script>

<esri:Navigation id="nav" map="{map}"/>
```



ESRI China (Beijing)
Geographic Thinking

- **Navigation中前一视图、后一视图、全图等功能和工具条按钮的绑定**
- **前一视图、后一视图的按钮并不是一直可用**

```
<mx:Button label="前一视图"
   click="nav.zoomToPrevExtent()"
   enabled="{!nav.isFirstExtent}"/>

<mx:Button label="后一视图"
   click="nav.zoomToNextExtent()"
   enabled="{!nav.isLastExtent}"/>

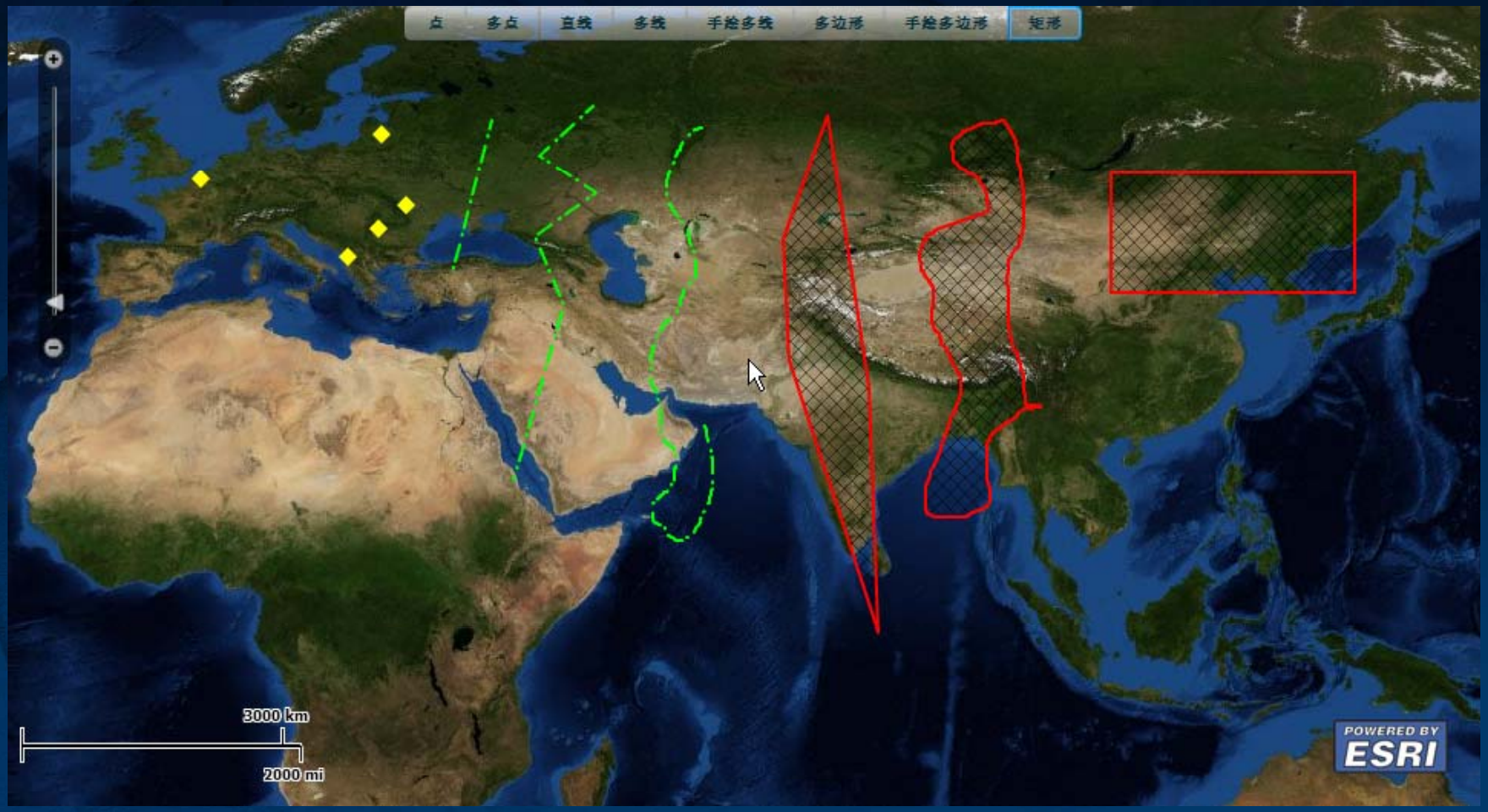
<mx:Button label="全图"
   click="nav.zoomToFullExtent()"/>
```

ESRI China (Beijing)
Geographic Thinking

Draw工具

- Draw工具提供了在客户端绘制各种几何对象的功能
- 用一个ToggleButtonBar去绑定Draw工具的功能

```
private function itemClickHandler(event:ItemClickEvent):void
      {
         switch (event.index)
         {
            case 0:{draw.activate(Draw.MAPPOINT);break;}
            case 1:{draw.activate(Draw.MULTIPOINT);break;}
            case 2:{draw.activate(Draw.LINE);break;}
            case 3:{draw.activate(Draw.POLYLINE);break;}
            case 4:{draw.activate(Draw.FREEHAND_POLYLINE);break;}
            case 5:{draw.activate(Draw.POLYGON);break;}
            case 6:{draw.activate(Draw.FREEHAND_POLYGON);break;}
            case 7:{draw.activate(Draw.EXTENT);break;}
         }
      }
```



ESRI China (Beijing)
Geographic Thinking

InfoWindow窗口

- InfoWindow窗口主要提供在Map上弹出信息提示的功能
- 定义一个Map的时候，Map会生成一个自带的InfoWindow并放置在自己的infoWindow属性中
- 下面我们在地图上弹出一个InfoWindow，并在其中放上一个图片：

```
var image:Image = new Image();
image.source = "http://www.esrichina-bj.cn/images/in_flashRig.jpg";

map.infoWindow.content = image;
map.infoWindow.show(new MapPoint(116, 40));
```



- InfoWindow使用的重点在于其content属性，它是InfoWindow窗口中的信息内容，它对应一个DisplayObject的对象，言下之意任何Flex中的可视组件都可以放置于其中。

- 从上面的内容，我们知道了如何使用一个Map来获得和显示ArcGIS Server发布的服务，然后使用Navigation工具来进行导航，用Draw工具在客户端绘制几何对象，用InfoWindow来显示提示信息。对于地图操作的浏览、交互我们都已经涉及到了。下面，我们将会在地图的基础上使用更多的ArcGIS Server的GIS功能。

ESRI China (Beijing)
Geographic Thinking

常用功能

- QueryTask
- FindTask
- IdentifyTask
- GeometryService
- Geoprocessor
- Locator

QueryTask

- QueryTask是一个进行空间和属性查询的功能类
- 可以在某个地图服务的某个子图层内进行查询
- QueryTask进行查询的地图服务并不必须加载到Map中进行显示
- QueryTask的执行需要两个先决条件：一个是需要查询的图层URL、一个是进行查询的过滤条件

ESRI China (Beijing)
Geographic Thinking

```
<mx:Script>
   <![CDATA[
      private function doQuery():void
      {
         var query:Query = new Query();
         query.geometry = map.extent;
         query.spatialRelationship = Query.SPATIAL_REL_WITHIN;
         query.where = "Name like '%"+keyword.text+"%'";
         query.outFields = ["Name", "Address"];
         query.returnGeometry = true;
         queryTask.execute(query);
      }
   ]]>
</mx:Script>

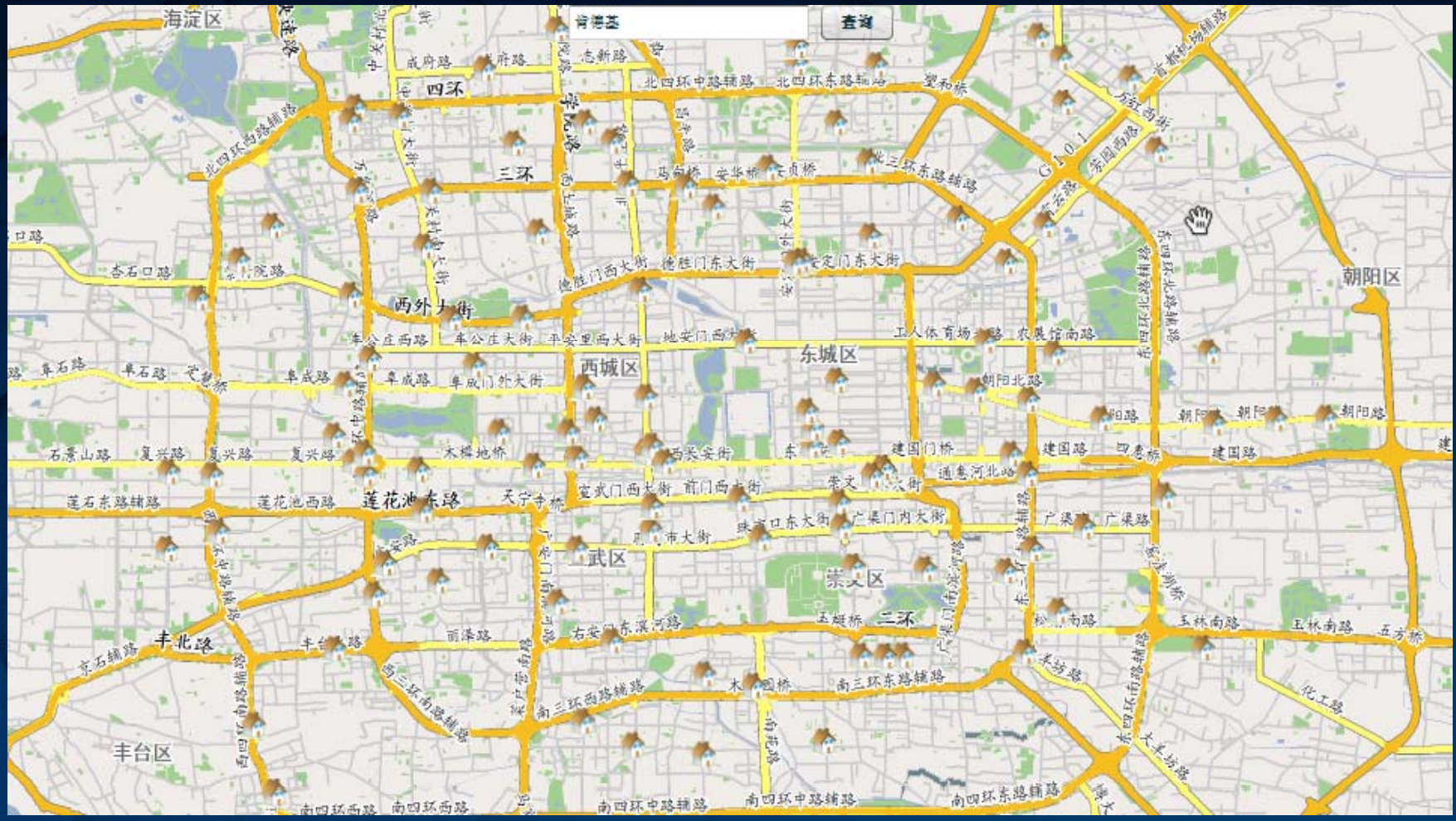
<esri:QueryTask id="queryTask”
url="http://localhost:8399/arcgis/rest/services/Beijing/beijing/MapServ
er/0"
executeComplete="onResult(event)" />
```

- QueryTask有个url属性指向需要查询的子图层的REST URL（索引从顶部图层为0开始）
- 当需要进行查询的时候我们需要调用QueryTask的execute方法
- execute方法的第一个参数就是对查询进行设置的Query对象，在这个Query对象的属性中我们可以设置过滤条件和返回内容。
- 在Query对象的属性中，geometry和spatialRelationship属性用于定义空间查询的条件、where属性用于定义属性查询的条件

ESRI China (Beijing)
Geographic Thinking

```
private function onResult(event:QueryEvent):void
{
   var featureSet:FeatureSet = event.featureSet;
    resultLayer.clear();
    for each (var g:Graphic in featureSet.features)
    {
       resultLayer.add(g);
    }
}
```

- **从onResult方法的QueryEvent参数中我们可以获得查询返回的所有结果集featureSet，featureSet的features属性中存放了所有查询结果的Graphic对象**



ESRI China (Beijing)
Geographic Thinking

FindTask

- 在某个地图服务中进行属性查询的功能类
- 需要一个FindParameters对象作为参数
- url属性需要指向所查询的地图服务的REST URL

```
private function doFind():void
{
    var params:FindParameters = new FindParameters();
    params.contains = true;
    params.layerIds = [0,1,2];
    params.returnGeometry = true;
    params.searchFields = ["Name", "Address"];
    params.searchText = keyword.text;

    findTask.execute(params);
}
```

```
private function onResult(event:FindEvent):void
{
        var results:Array = event.findResults;
        var resultCount:int = results.length;
        resultLayer.clear();
        for (var i:Number=0; i<resultCount; i++)
        {
                var g:Graphic = results[i].feature;
                resultLayer.add(g);
        }
}
```

IdentifyTask

- 在地图服务中识别要素的功能类
- 使用Draw工具绘制了一个几何对象以后，这个几何对象就可以作为IdentifyTask的参数发送到服务器进行识别

```
private function onDrawEnd(event:DrawEvent):void
{
   var identifyParams:IdentifyParameters = new IdentifyParameters();
   identifyParams.returnGeometry = true;
   identifyParams.width = map.width;
   identifyParams.height = map.height;
   identifyParams.mapExtent = map.extent;
   identifyParams.geometry = event.graphic.geometry;
   identifyParams.tolerance = 3;
   identifyParams.layerOption =
IdentifyParameters.LAYER_OPTION_VISIBLE;
   identifyTask.execute(identifyParams);
}
```

```
private function onIdentifyComplete(event:IdentifyEvent):void
{
    var results:Array = event.identifyResults;
    for each (var result:IdentifyResult in results)
    {
        resultLayer.add(result.feature);
    }
}
```

- **和FindTask的结果处理很类似，写法不同**


Point
Polyline
Polygon
内蒙古自治区
吉林省
新疆维吾尔自治区
辽宁省
甘肃省
河北省
山西省
宁夏回族自治区
山东省
青海省
陕西省
河南省
江苏省
安徽省
湖北省
上海市
西藏自治区
四川省
重庆市
浙江省
湖南省
江西省
贵州省
福建省

GeometryService

- 每个ArcGIS Server都可以发布一个几何服务
- 用以处理几何对象
- 投影转换
  简化对象
  缓冲
  量测
  判断空间关系
  计算标注点
- GeometryService功能类来使用几何服务提供的功能

ESRI China (Beijing)
Geographic Thinking

- GeometryService提供的功能比较多，因此它实现了很多监听器供不同的功能使用

project	投影转换
simplify	简化几何对象
buffer	缓冲
areasAndLengths	量测多边形的面积与周长
lengths	量测多段线的长度
relation	判断空间关系
labelPoints	计算标注点

- 下面挑一个功能进行讲解，调用relation方法
- 分别查询城市和省的Graphic集合
- 对两个Graphic集合进行计算

```
private function onQueryCity(event:QueryEvent):void
{
        graphics1 = new Array();
        var featureSet:FeatureSet = event.featureSet;
        var count:int = featureSet.features.length;
        for (var i:int=0; i<count; i++)
        {
                var g:Graphic = featureSet.features[i];
                graphics1.push(g);
        }
}
```

```
private function onQueryProvice(event:QueryEvent):void
{
	var featureSet:FeatureSet = event.featureSet;
	var g:Graphic = featureSet.features[0];
	graphics2 = [g];

	geometryService.relation(graphics1, graphics2,
GeometryService.SPATIAL_REL_INTERSECTION);
}

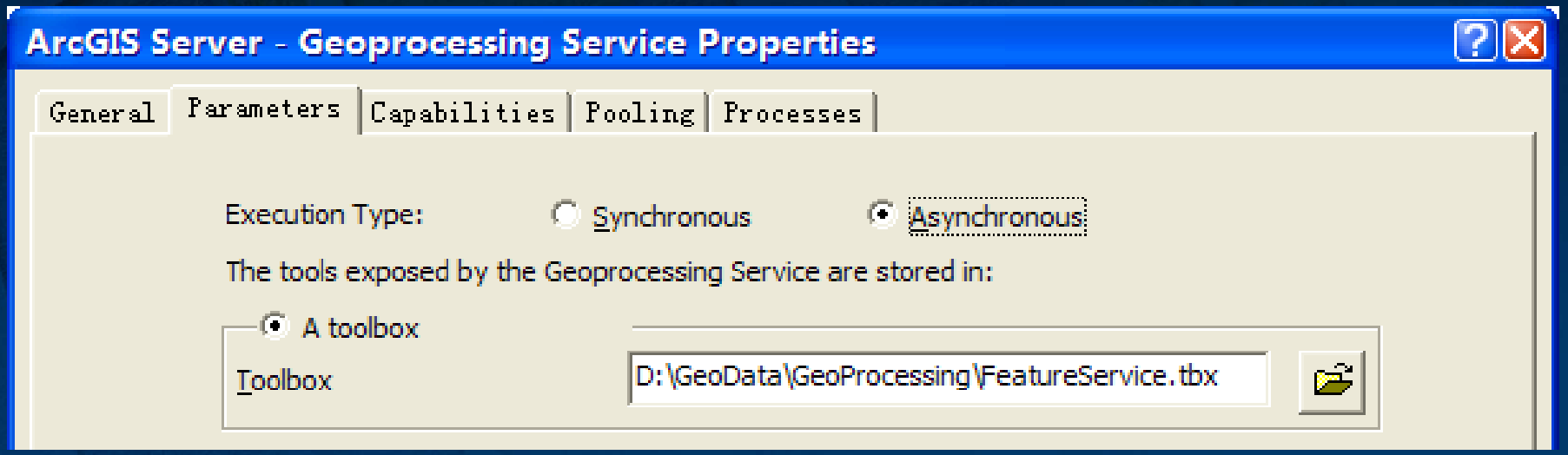
private function onRelationComplete(event:GeometryServiceEvent):void
{
	var relations:Array = event.relations;
	resultLayer.clear();
	for (var i:int=0; i<relations.length; i++)
	{
		var g:Graphic = relations[i].graphic1;
		resultLayer.add(g);
	}
}
```

- 得到江苏省内所有地级市



Geoprocessor

- GP服务（Geoprocessing Service）可以使用户能够使用高级的模型功能
- Geoprocessor功能类对应的就是GP服务的使用
- execute（同步）和submitJob（异步）方法



- 通过一个“漂流瓶之旅”的例子来看一下Geoprocessor怎样调用GP服务
- http://sampleserver1.arcgisonline.com/ArcGIS/rest/services/Specialty/ESRI_Currents_World/GPServer/MessageInABottle
- 输入：
  Input_Point（GPFeatureRecordSetLayer类型）
  Days（GPDouble类型）
  输出：
  Output（GPFeatureRecordSetLayer类型）

```
private function startGeoProcess(event:MapMouseEvent):void
{
        this.cursorManager.setBusyCursor();

        var mapPoint:Geometry = event.mapPoint;
        var startGraphic:Graphic = new Graphic(mapPoint, sms);
        resultLayer.add(startGraphic);

        var featureSet:FeatureSet = new FeatureSet();
        featureSet.features = [ { geometry:mapPoint } ];

        var parms:Object = new Object();
        parms.Input_Point = featureSet;
        parms.Days = Number(numberOfDays.text);
        geoprocessTask.execute(parms);
}
```

```
 private function onExecuteComplete(event:GeoprocessorEvent):void
{
    for each (var g:Graphic in
                event.executeResult.parameterValues[0].value.features)
    {
        g.symbol = sls;
        resultLayer.add(g);
    }

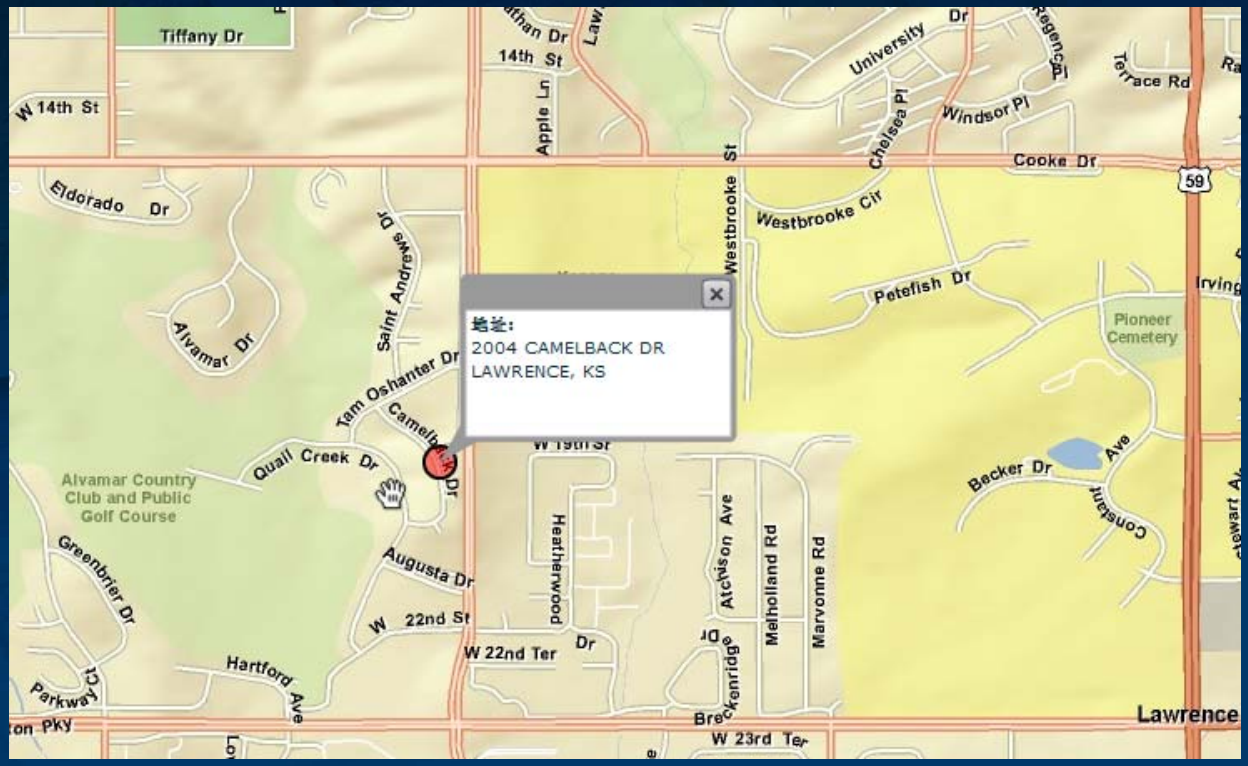
    this.cursorManager.removeBusyCursor();
}
```



ESRI China (Beijing)
Geographic Thinking

Locator

- 对应使用的是ArcGIS Server发布的地理编码服务（Geocode Service）
- addressToLocations和locationToAddress方法



ArcGIS Server中配置安全策略文件

- 访问跨域的ArcGIS Server服务
- 出现了安全沙箱错误
  运行的时候出不来地图
- 需要在远程的Web服务器根目录下放置一个
  “crossdomain.xml”安全策略文件

```
<?xml version="1.0"?>
<!DOCTYPE cross-domain-policy SYSTEM
"http://www.adobe.com/xml/dtds/cross-domain-policy.dtd">
<cross-domain-policy>
   <site-control permitted-cross-domain-policies="all"/>
   <allow-access-from domain="*"/>
</cross-domain-policy>
```

- 安全策略文件需要被放到特定的位置

  ArcGIS Server for .Net应该是IIS的根目录
  （如：C:\Inetpub\wwwroot）

  ArcGIS Server for Java应该是web_output目录
  （如：C:\Program
  Files\ArcGIS\java\web_output）

ESRI China (Beijing)
Geographic Thinking



ESRI China (Beijing)
Geographic Thinking

【ArcGIS Flex API 高级】

控制Map与Layer

- Map中地图服务图层的控制
- 地图服务图层中子图层的控制
- 子图层中内容的控制
- 动态投影

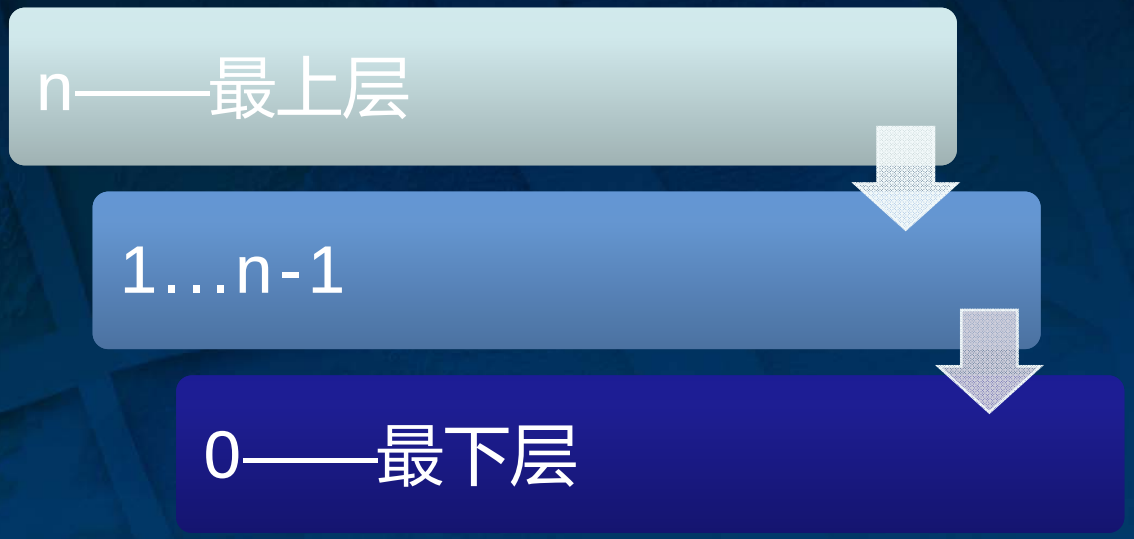
ESRI China (Beijing)
Geographic Thinking

Map中地图服务图层的控制

- Layer的visible属性指示当前图层是否被渲染
- Layer的alpha属性来控制图层的透明度



- **关于图层的叠放顺序：**
  后加入的地图被放置在最上层
  最底层的索引（index）为0



ESRI China (Beijing)
Geographic Thinking

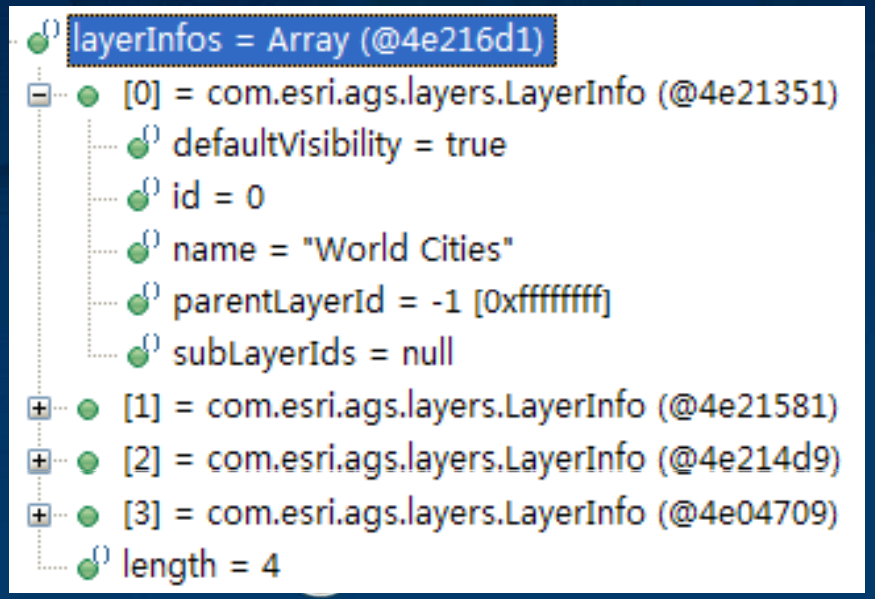
- Map中有一个layers属性存放了地图中所有Layer的集合，另外还有layerIds属性存放了对应的所有Layer的id的集合，每当添加一个Layer到Map中的时候，ArcGIS Flex API会给这个图层一个id并存放到layerIds中保存。通过调用Map的reorderLayer方法可以给这些图层重新排序

```
map.reorderLayer(map.layerIds[map.layers.length-1], 0);
```

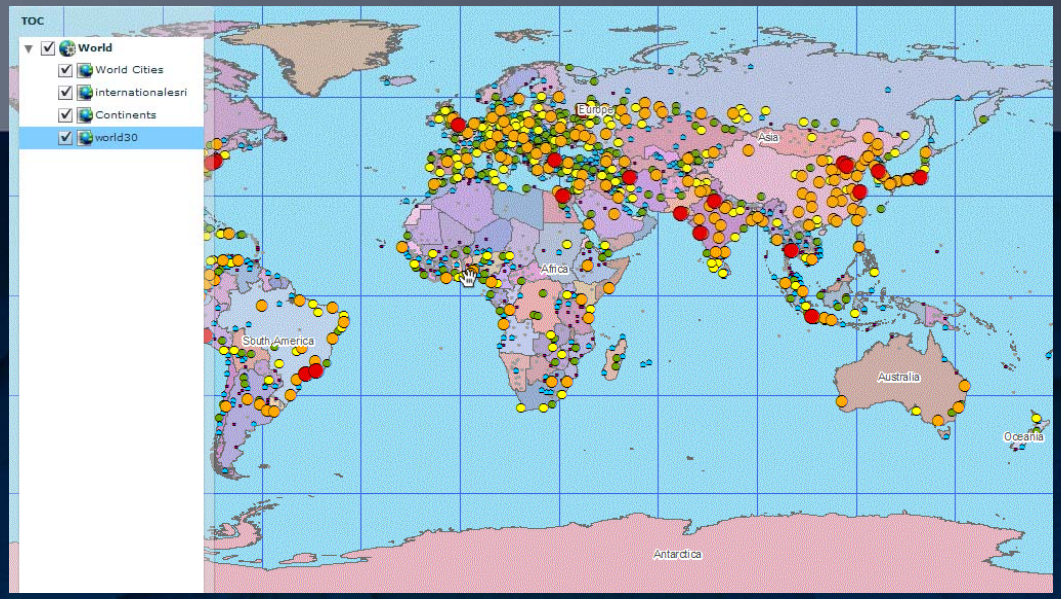
地图服务图层中子图层的控制

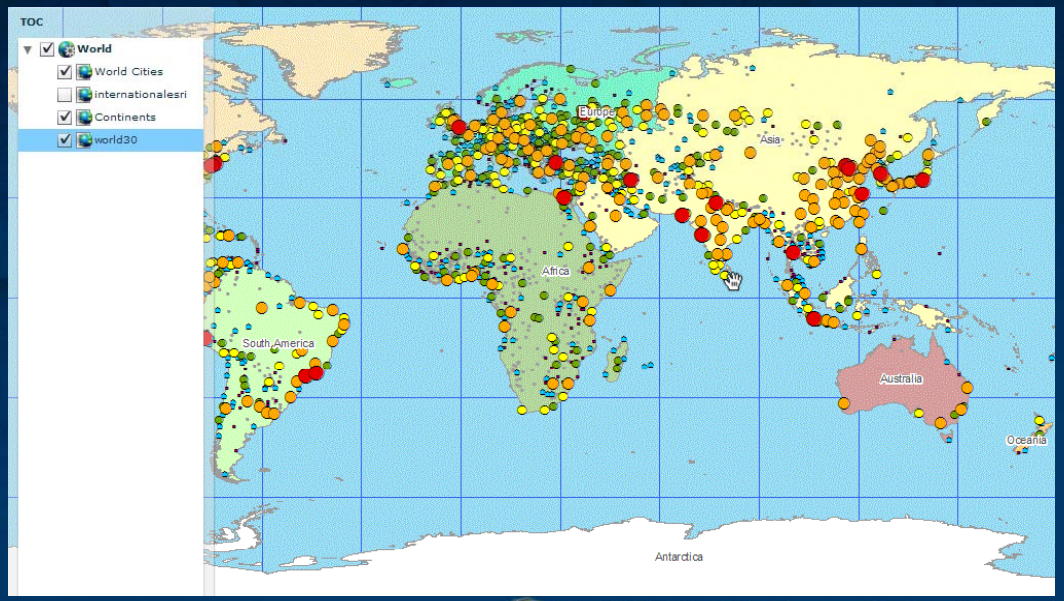
- 主要对ArcGISDynamicMapServiceLayer的控制
- 首先需要知道服务中有些什么内容
- ArcGISDynamicMapServiceLayer有一个layerInfos属性，这个属性是一个数组，它存放了这个地图服务中所有子图层的相关信息





- 值得注意的是，这里子图层的id虽然也是从0开始，但是和Map中的地图服务图层索引正好相反，最上层的子图层为0
- 关于地图服务加载， Layer的load监听
- 当我们获得layerInfos以后，我们就可以控制服务中的子图层了，控制通过ArcGISDynamicMapServiceLayer的visibleLayers属性实现，visibleLayers对应一个数组，这个数组存放了需要显示的所有子图层的id





ESRI China (Beijing)
Geographic Thinking

子图层中内容的控制

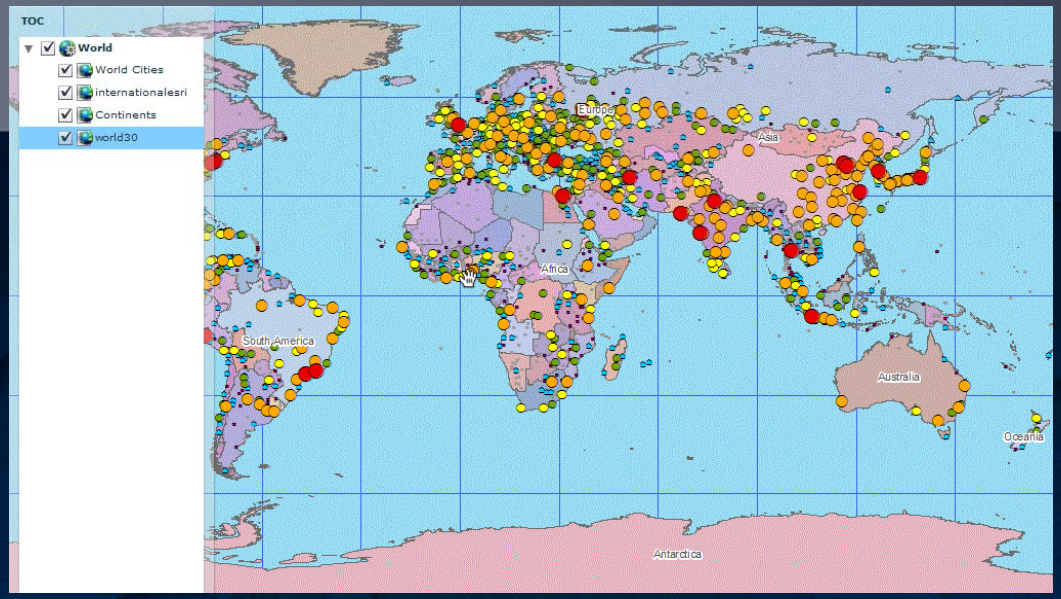
- ArcGISDynamicMapServiceLayer有一个layerDefinitions属性，这个属性对应的是一个和layerInfos长度相同的数组
- layerDefinitions数组中的对象是字符串，从索引0开始，每个字符串对应一个子图层。这个字符串是SQL语句的一部分，它可以过滤出地图服务中某个子图层中需要被显示的要素

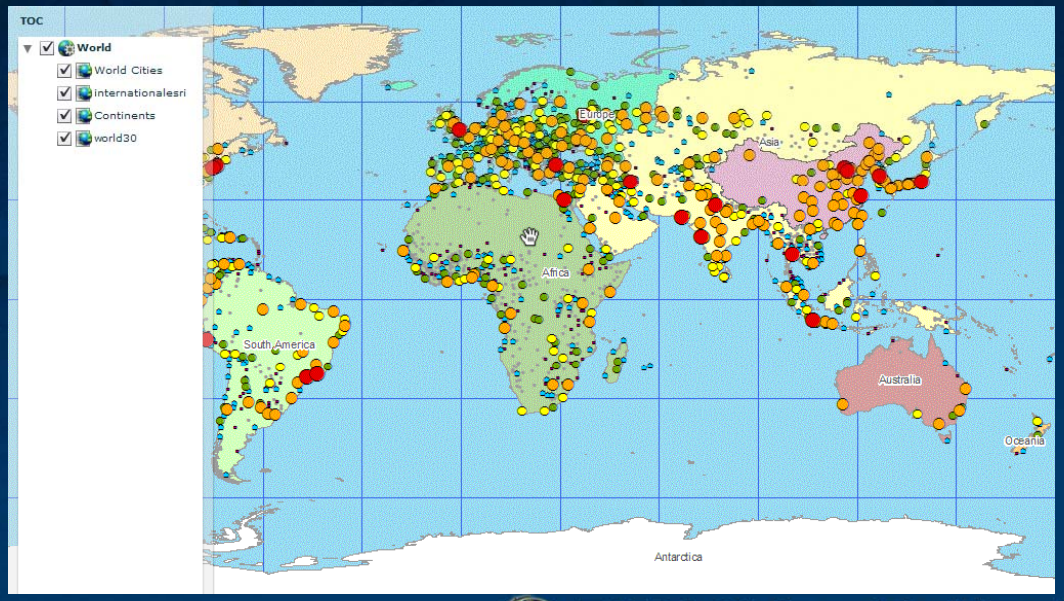
ESRI China (Beijing)
Geographic Thinking

- **只显示中国的国界：**

```
<esri:ArcGISDynamicMapServiceLayer
        url="http://localhost:8399/arcgis/rest/services/AgsSample/World
/MapServer">
   <esri:layerDefinitions>
        <mx:Array>
                <mx:String></mx:String>

        <mx:String><![CDATA[CNTRY_NAME='China']]></mx:String>
                <mx:String></mx:String>
                <mx:String></mx:String>
        </mx:Array>
   </esri:layerDefinitions>
</esri:ArcGISDynamicMapServiceLayer>
```





ESRI China (Beijing)
Geographic Thinking

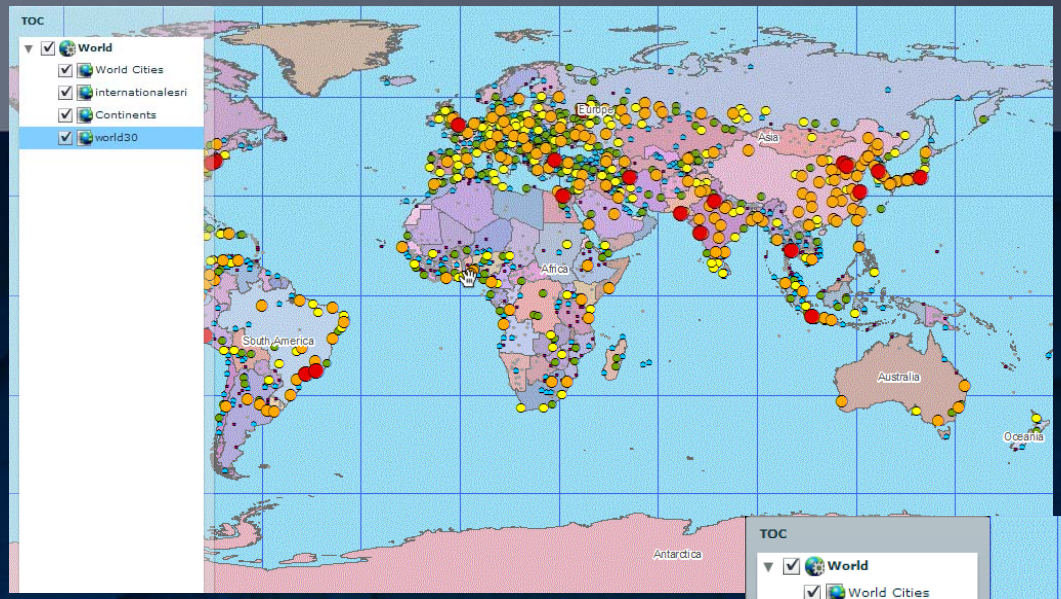
动态投影

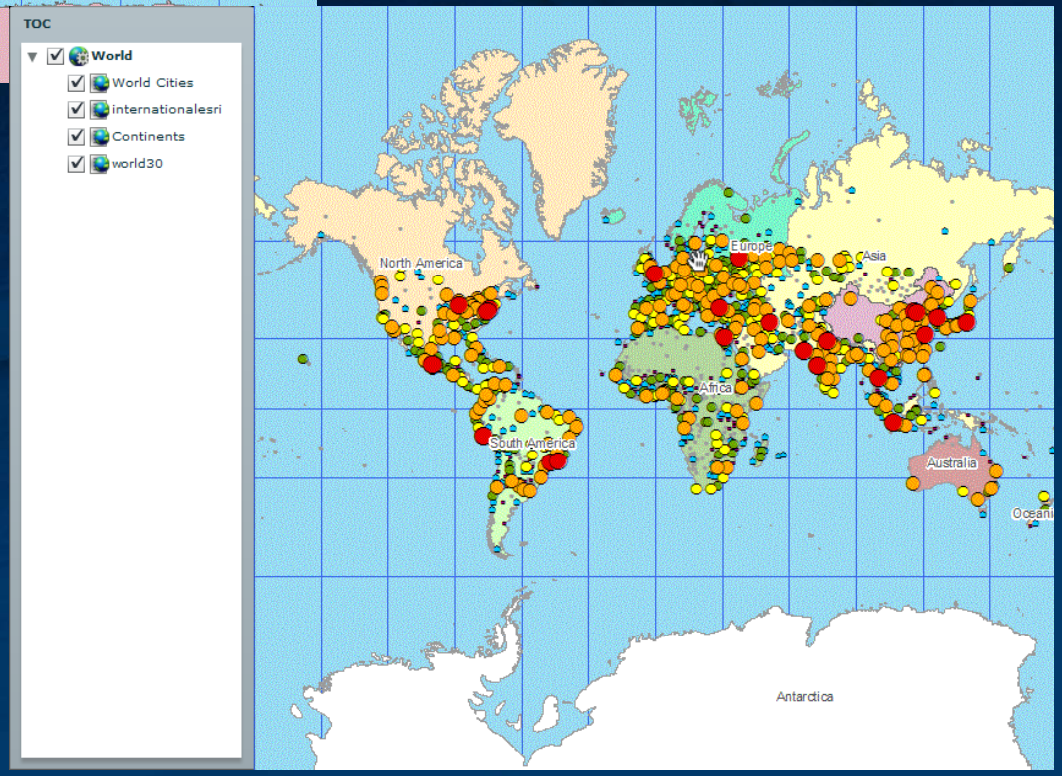
- **可以根据需要设置不同的空间参考，从而实现动态的投影**

```
<esri:Map>
   <esri:extent>
        <esri:Extent
            spatialReference="{new SpatialReference(102113)}“
            xmin="-20037508.342787“
            xmax="20037508.342787"
            ymin="-20037508.342787“
            ymax="20037508.342787"/>
   </esri:extent>
   <esri:ArcGISDynamicMapServiceLayer
url="http://localhost:8399/arcgis/rest/services/AgsSample/World/Map
Server"/>
</esri:Map>
```

ESRI China (Beijing)
Geographic Thinking

- 这个102113空间参考在ArcGIS Server中已经有定义
- ArcGIS Flex API只是保存了这个空间参考的id，本身并不作任何的计算
- 另外，设置的Well-Known ID必须是ArcGIS Server中已有的定义，具体可用的空间参考列表可以在REST SDK文档中找到


TOC
World
World Cities
internationalesri
Continents
world30
Europe
Asia
Africa
South America
Australia
Oceania
Antarctica


TOC
World
World Cities
internationalesri
Continents
world30
North America
Europe
Asia
Africa
South America
Australia
Antarctica

业务信息的表达

- 分类渲染
- 专题图
- 实时的数据变化
- 符号扩展

分类渲染

- 主要讨论GraphicLayer的分类渲染
- 比较笨拙的办法就是每绘制一个Graphic之前先判断一下它的属性，然后根据情况来给Graphic赋一个不同的符号值——代码可能比较混乱、不好维护
- GraphicLayer有个属性是symbolFunction，这个属性对应的是一个函数类型的对象，在这里我们可以定义一个函数来对符号进行判断
- 下面的代码实现了这样的功能：从美国1999年州人口统计数据中，根据人口分布区间，使用不同的颜色进行表示

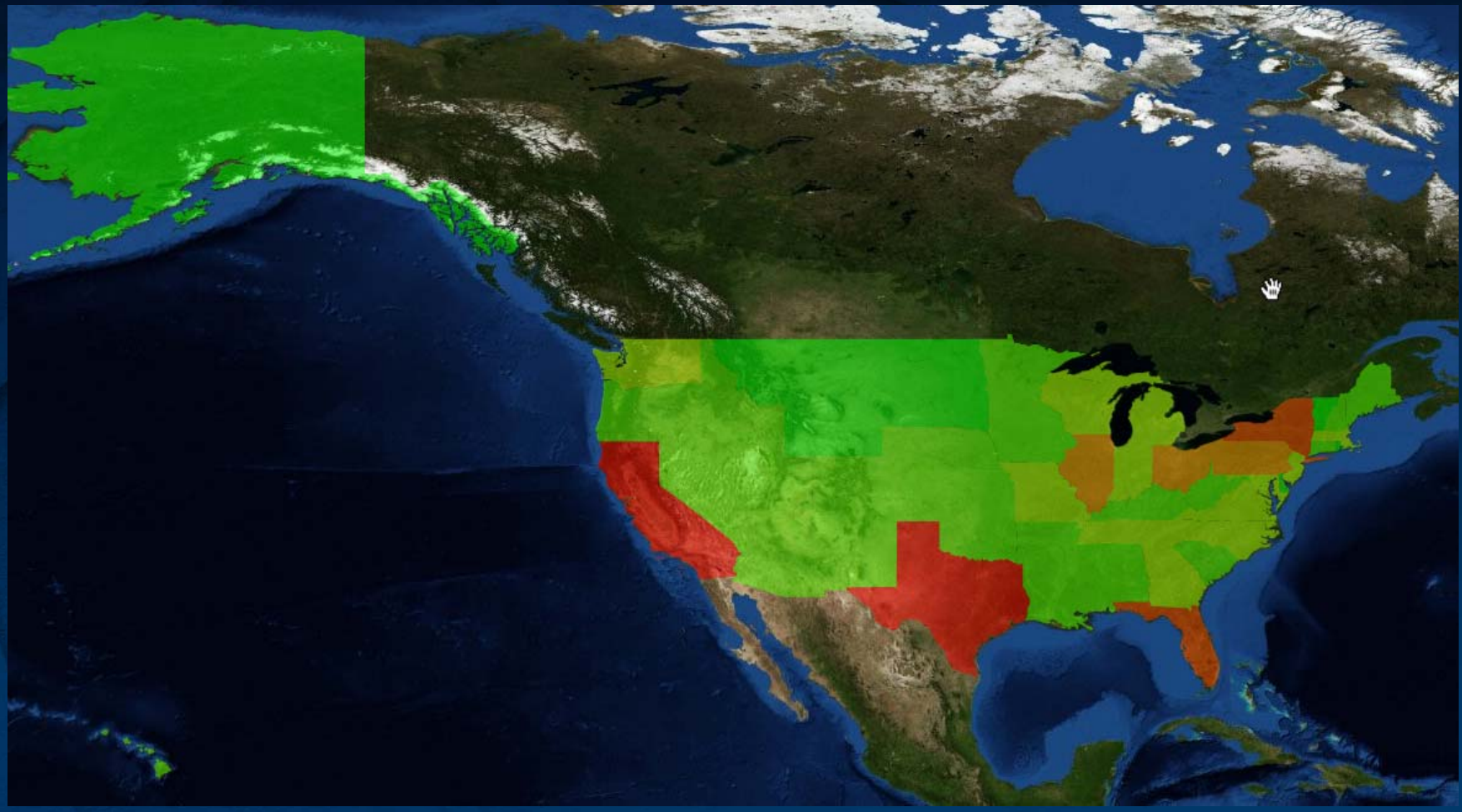
ESRI China (Beijing)
Geographic Thinking

```
<mx:Script>
   <![CDATA[
        private function getSymbol(g:Graphic):Symbol
        {
                var symbol:SimpleFillSymbol = new SimpleFillSymbol();

                var pop:Number = g.attributes["POP1999"];
                if( pop<1000000 ) symbol.color = 0x00FF00;
                ......
                else symbol.color = 0xFF0000;

                return symbol;
        }
   ]]>
</mx:Script>

<esri:Map id="map“>
   <esri:GraphicsLayer  symbolFunction="getSymbol"/>
</esri:Map>
```



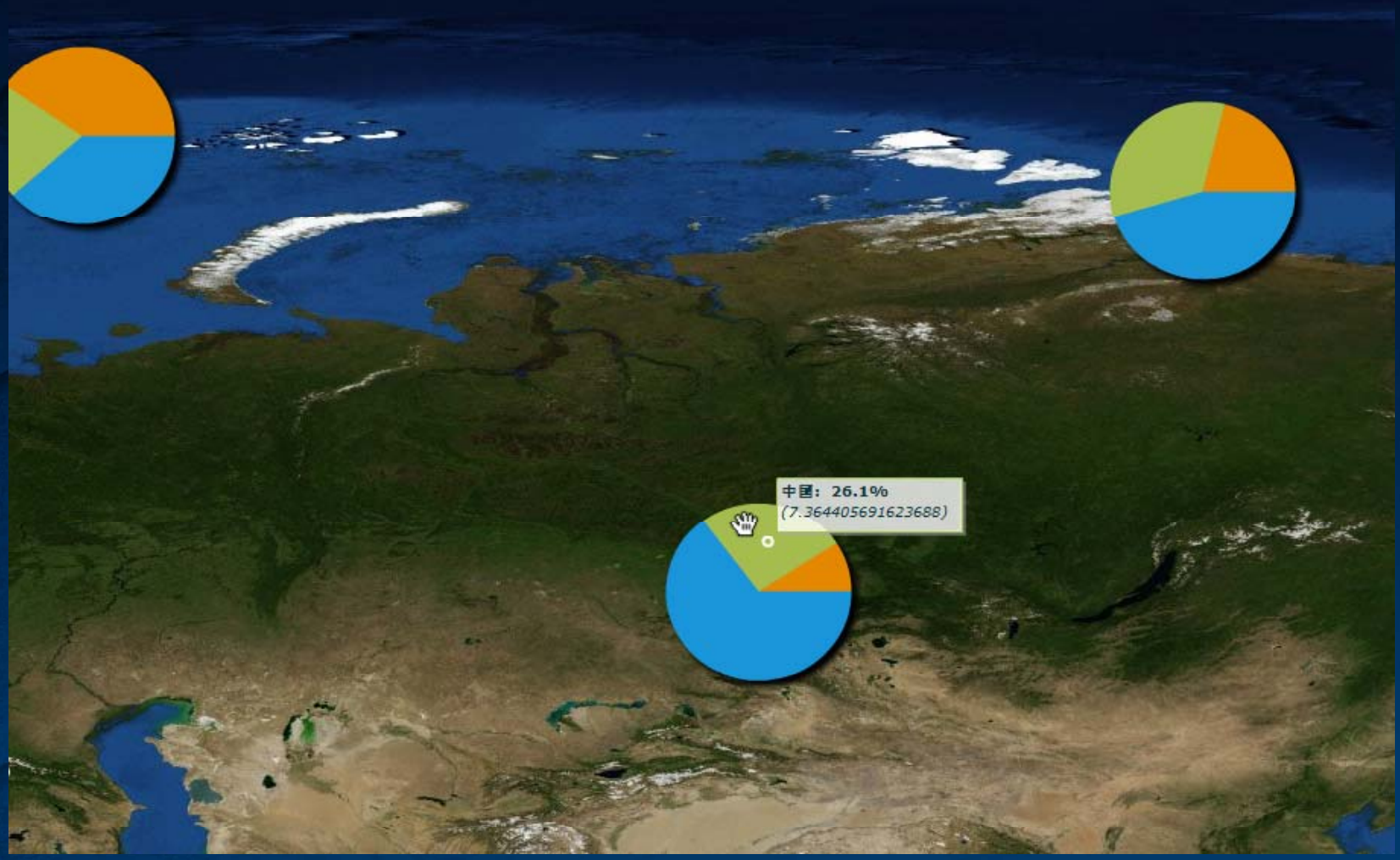
ESRI China (Beijing)
Geographic Thinking

专题图

- 专题图主要通过InfoSymbol来实现
- 在InfoSymbol中放置比如饼图、柱状图等组件后，其对应的Graphic就可以被绘制成一个信息非常丰富的符号
- 符号内的数据和Graphic的attributes是相关联的

```
<esri:InfoSymbol id="infoSymbol" infoPlacement="center">
   <esri:infoRenderer>
         <mx:Component>
               <mx:PieChart id="chart"
showDataTips="true" width="150" height="150"
dataProvider="{data.thematic}">
                     <mx:series>
                        <mx:PieSeries field="Gold"
                              displayName="资源占有率"
                              nameField="Country"/>
                     </mx:series>
               </mx:PieChart>
         </mx:Component>
   </esri:infoRenderer>
</esri:InfoSymbol>
```

ESRI China (Beijing)
Geographic Thinking



ESRI China (Beijing)
Geographic Thinking

实时的数据变化

- Flex应用可以和服务器建立Socket连接，从而真正实现从服务器向客户端“推送”数据
- 使用服务器刷新数据这种方式来构建“准”实时的应用，那么从ArcGIS Flex API的角度来说也没什么区别
- 主要看一下刷新数据的时候我们应该怎么改变原先Graphic对象的位置，以及在构建实时的应用时需要注意的一些特殊的细节

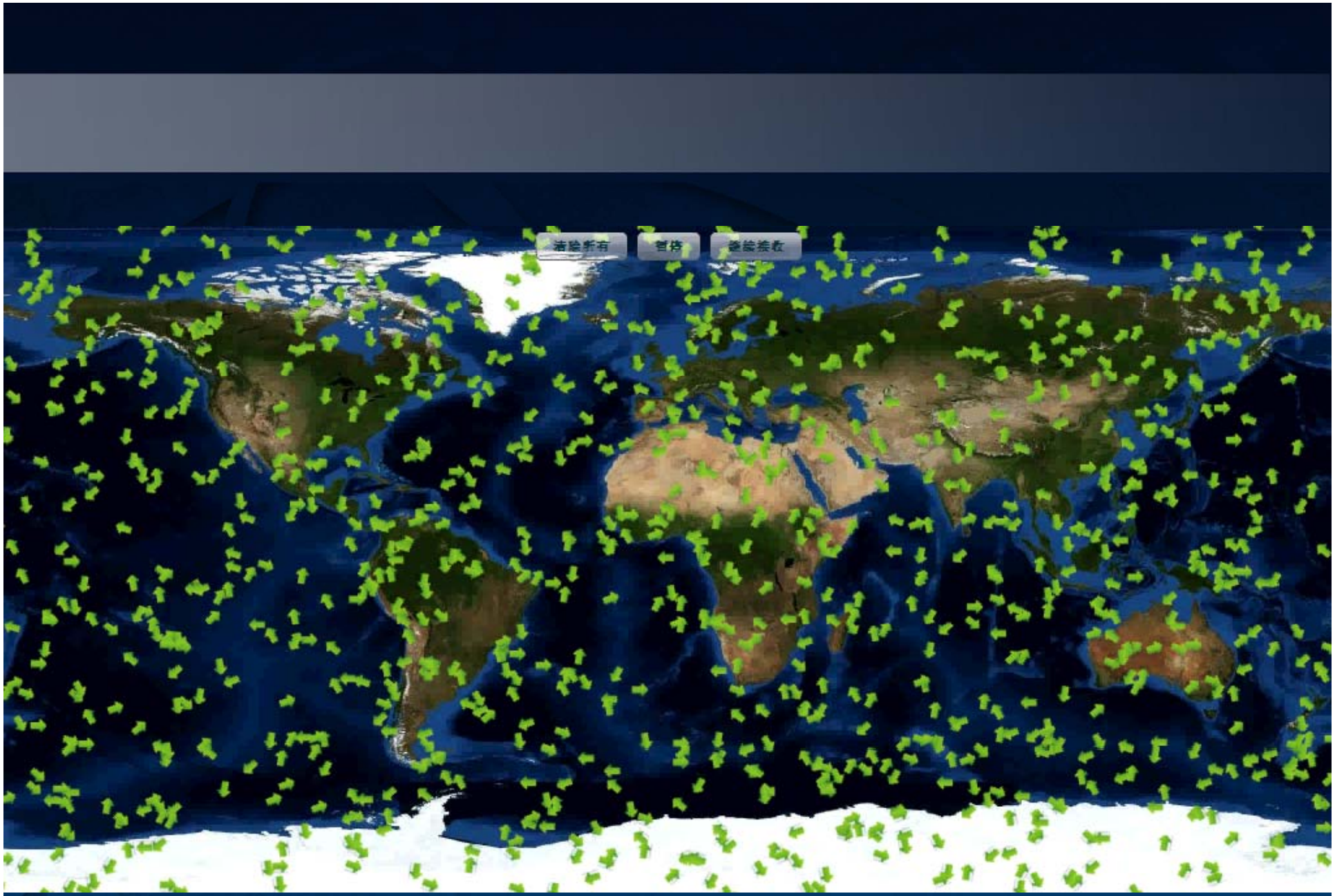
- 首先需要从一个GraphicLayer上获得那个特定的Graphic
- 由于所有Graphic都是GraphicLayer的子对象，因此，我们可以通过GraphicLayer的getChildAt或getChildByName方法来获取一个Graphic
- 更改这个Graphic的位置其实就是更新这个Graphic的geometry属性，把这个属性更新以后，这个Graphic会自动重绘到新的位置去

- 资源的嵌入
- Graphic的autoMoveToTop属性
- 浏览器中的Flash Player大概可以承受的Graphic的数量级

ESRI China (Beijing)
Geographic Thinking

- 实时监控Demo
  监控1000个不同的对象
  服务器每0.5秒给客户端推送一次数据
  每次推送的数据改变所有这1000个对象的位置
- 在这样的压力下，这个Demo的交互操作基本上没有延迟

ESRI China (Beijing)
Geographic Thinking



ESRI China (Beijing)
Geographic Thinking

符号扩展

- 对符号有更多的需求
- 比如希望实现渐变效果、使用贝塞尔曲线来绘制符号等等
- draw方法的原型

```
override public function draw(sprite:Sprite, geometry:Geometry,
attributes:Object, map:Map):void
```

- 绘制符号需要做两件事情：
  首先需要给sprite定位、定尺寸
  然后再在sprite中绘制图形
- 下面的代码使用渐变填充绘制了一个圆形符号

ESRI China (Beijing)
Geographic Thinking

```
var half:Number = size/2;

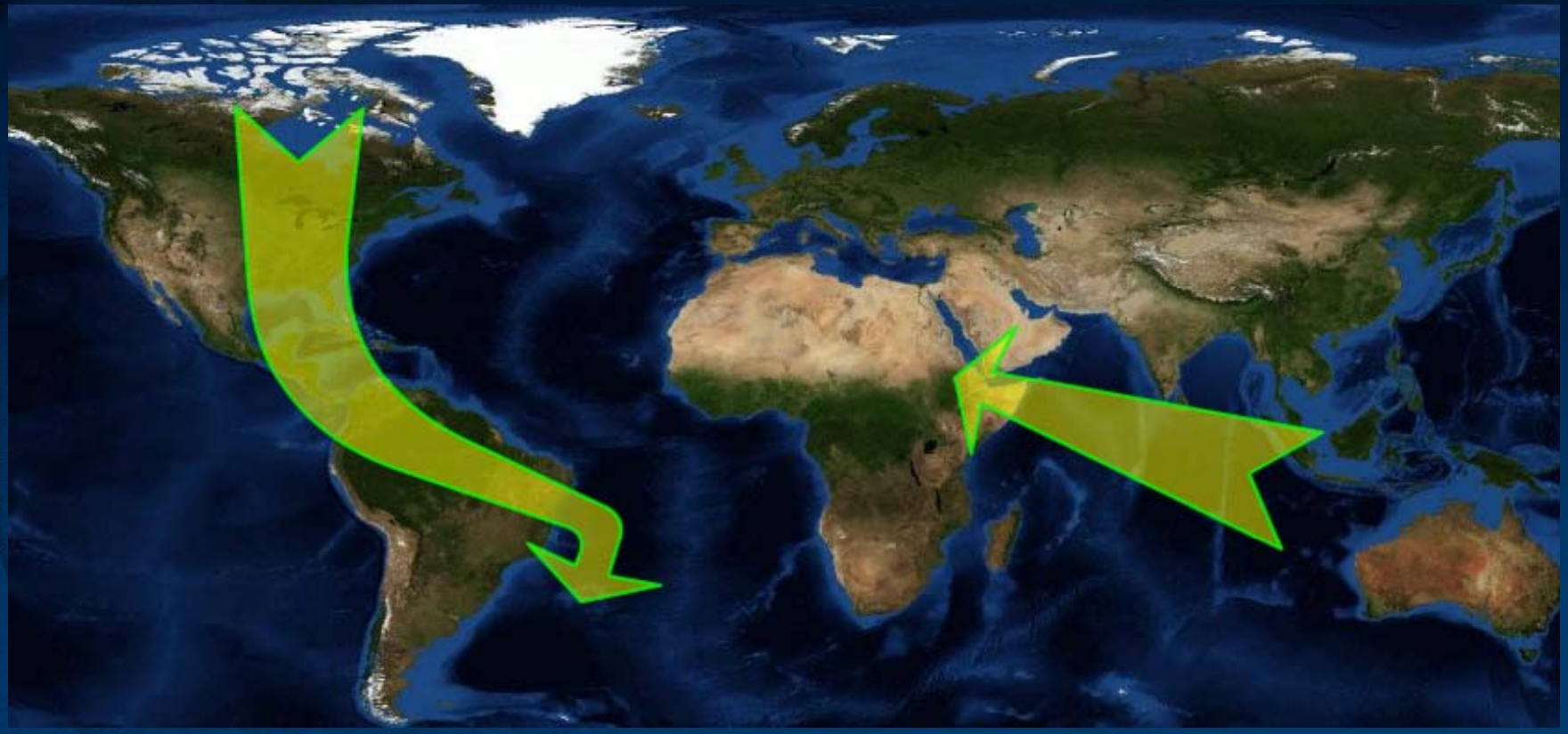
//定位、定尺寸
sprite.x = toScreenX(map, mp.x);
sprite.y = toScreenY(map, mp.y);
sprite.width = size;
sprite.height = size;

//绘制图形
sprite.graphics.beginGradientFill(GradientType.RADIAL, colors, alphas,
ratios);
sprite.graphics.drawCircle(0, 0, half);
sprite.graphics.endFill();
```



ESRI China (Beijing)
Geographic Thinking

- 绘制曲线行军箭头符号



ESRI China (Beijing)
Geographic Thinking

与其它业务系统的整合

- 与JavaScript交互
- HTTPService与WebService
- RemoteObject与BlazeDS

ESRI China (Beijing)
Geographic Thinking

与JavaScript交互

- Flex可以触发页面上的JavaScript函数
- 页面上的JavaScript也可以触发Flex中的函数
- 这使ArcGIS Flex API应用可以以非常独立的方式嵌入到某个页面

- 某客户原来的MIS系统已经开发结束并具备了完整的功能，查询信息都能输出到页面的表格中
- 因此，给这个客户设计的解决方案就是采用ArcGIS Flex API，将GIS功能封装到Flex中，数据交互则通过JavaScript进行

- 原先的报警点是输出到当前页面的表格中的，我们改写了JavaScript，使其在刷新报警点的时候，调用另外一个函数把这个表格中的所有数据发送到Flex中去，然后由一个GraphicLayer来负责绘制

```
private function init():void
{
    //供JavaScript调用
    ExternalInterface.addCallback("clearHotPoints", clearHotPoints);
    ExternalInterface.addCallback("addHotPoint", addHotPoint);
}
```

- 反过来，当用户在Flex中浏览到新的地图范围时，需要刷新数据以获得新的报警点，这是就需要从Flex调用JavaScript函数

```
private function refresh():void
{
    //调用JavaScript函数
    ExternalInterface.call("refreshHotPoints", xmin, xmax, ymin, ymax);
}
```

HTTPService与WebService

- Flex通过HTTPService类从Web服务器获取数据
  通过WebService类从Web Service获取数据

- 以下两个例子分别使用HTTPService和WebService

ESRI China (Beijing)
Geographic Thinking

```
<mx:Script>
   <![CDATA[
        private function addPoint():void
        {
                var mapPoint:MapPoint = graphic.geometry as MapPoint;

                var params:Object = new Object();
                params.x = mapPoint.x;
                params.y = mapPoint.y;

                this.httpAddPoint.send(params);
        }
        private function onAddPointResult(event:ResultEvent):void
        {
                var str:String = event.result as String;
                var json:Object = JSON.decode(str);
                //对JSON数据的处理
        }
   ]]>
</mx:Script>

<mx:HTTPService id="httpAddPoint"
url="{ROOT_URL+'/point.php?action=add'}" method="POST"
showBusyCursor="true" result="onAddPointResult(event);" />
```

```
<mx:WebService id="ws"
        wsdl="http://livecycledata.org/services/ProductWS?wsdl"
        useProxy="false" showBusyCursor="true"/>

<mx:DataGrid dataProvider="{ws.getProducts.lastResult}" >
   <mx:columns>
      <mx:DataGridColumn dataField="productId" headerText="编号"/>
      <mx:DataGridColumn dataField="name" headerText="名称"/>
      <mx:DataGridColumn dataField="price" headerText="价格"/>
   </mx:columns>
</mx:DataGrid>

<mx:Button label="获取产品信息" click="ws.getProducts()"/>
```

ESRI China (Beijing)
Geographic Thinking

RemoteObject与BlazeDS

- RemoteObject功能类可以通过AMF（Action Message Format）数据格式来访问远程应用服务器上的类（Java、.Net、PHP等）
- AMF是一种压缩的二进制数据格式，专门用于ActionScript与服务器端通讯
- RemoteObject可以使你专注于Flex客户端开发和服务器端功能开发，而隐藏掉服务器提供Web功能的细节
- Adobe公司针对Java发布了一个很好的开放源代码的服务组件BlazeDS，Flex应用通过BlazeDS的帮助就可以无缝地调用远程服务器上的Java方法提供的功能

- 下面是一个从数据库获得污染节点并计算污染影响范围的例子。首先，我在服务器端编写了一个类Wind用以获得污染节点并通过风力参数来计算其影响范围
- Java代码：

ESRI China (Beijing)
Geographic Thinking

```
public ArrayList<HashMap<String, String>> getEffectZones(double xMin,
double yMin, double xMax, double yMax)
{
        ArrayList<HashMap<String, String>> result = new
ArrayList<HashMap<String, String>>();
        try
        {
                ......
                ResultSet rs = st.executeQuery(sql);
                while (rs.next())
                {
                        ......
                        result.add(map);
                }
                rs.close();
        }
        catch(Exception e)
        {
                System.err.print(e);
        }

        return result;
}
```

- 要想这个类可以在Flex应用中通过RemoteObject可以访问到，我们需要在“remoting-config.xml”文件中添加这个类的定义：

```
<destination id="Wind">
        <properties>
                <source>wuyf.Wind</source>
                <scope>application</scope>
        </properties>
</destination>
```

ESRI China (Beijing)
Geographic Thinking

- 下面让我们来看一下Flex中相应的代码

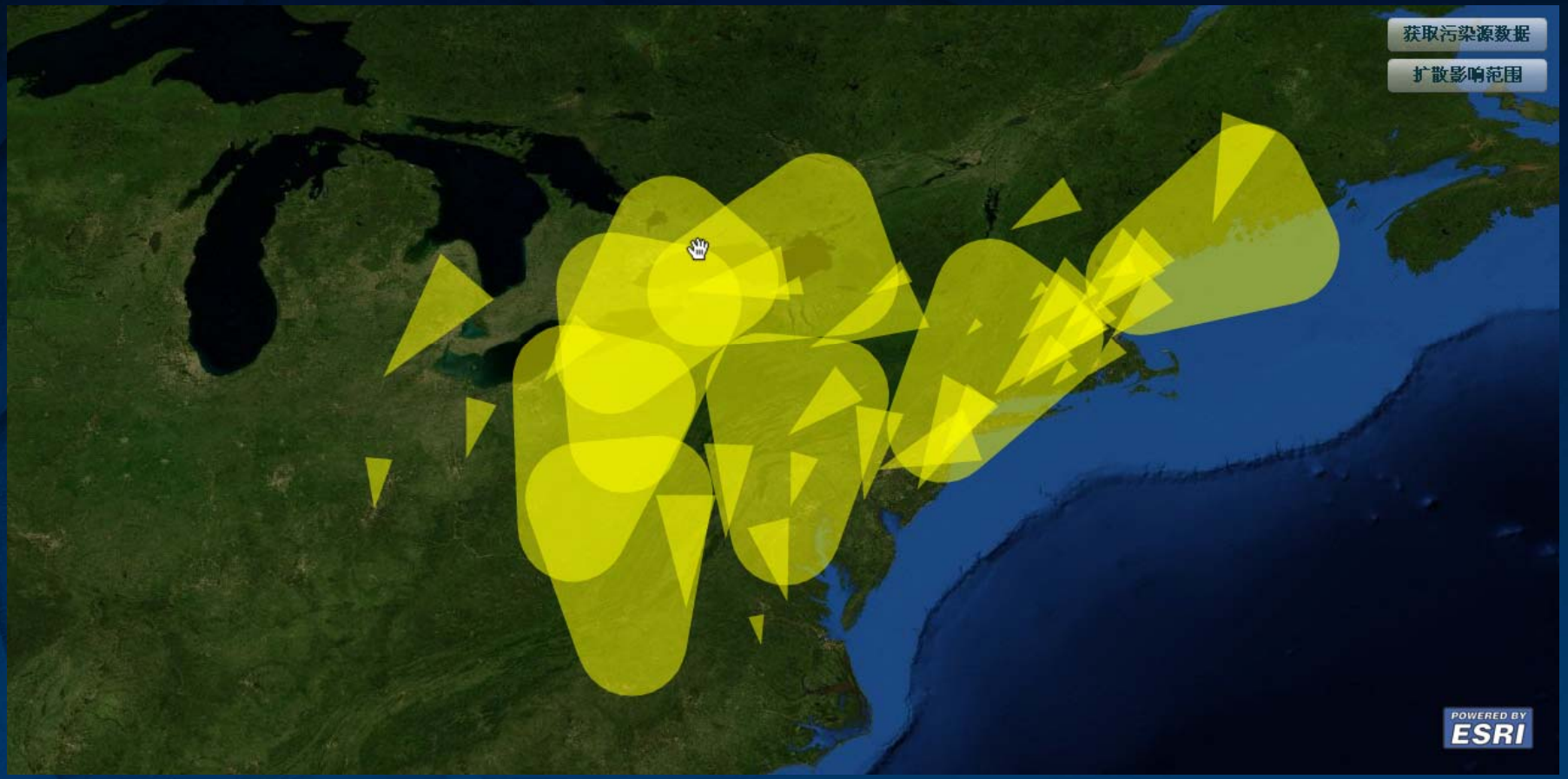
```
<mx:RemoteObject id="roWind" destination="Wind" />
```

ESRI China (Beijing)
Geographic Thinking

```
private function getEffectZones():void
{
        var extent:Extent = map.extent;

        var args:Array = new Array();
        args.push(extent.xmin);
        args.push(extent.ymin);
        args.push(extent.xmax);
        args.push(extent.ymax);

        var opt:AbstractOperation =
this.roWind.getOperation("getEffectZones");
        opt.addEventListener(ResultEvent.RESULT,
onGetEffectZonesResult);
        opt.arguments = args;
        opt.send();
}
```



ESRI China (Beijing)
Geographic Thinking

综合场景演示

- 某系统演示
- 其它一些案例

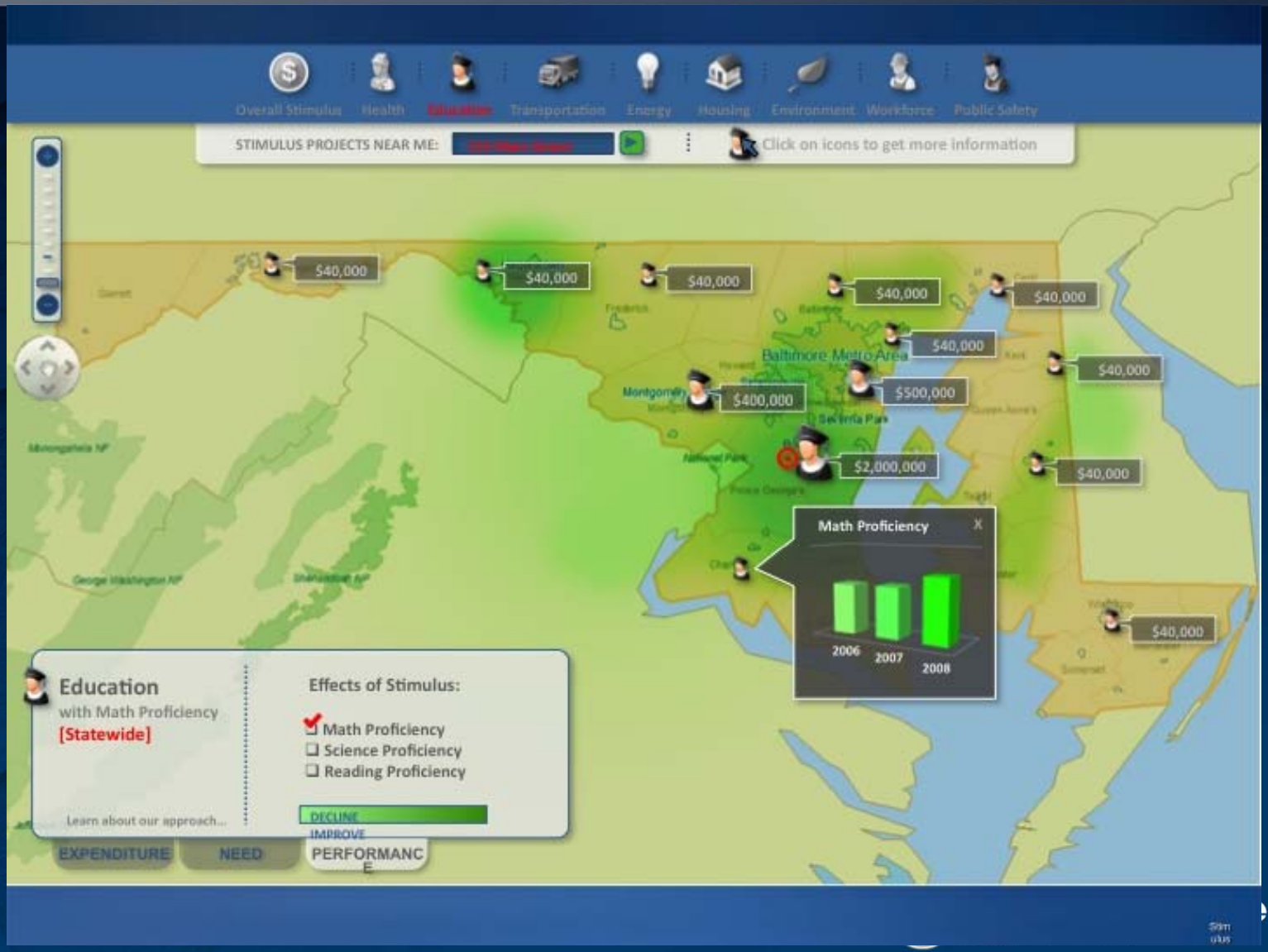
ESRI China (Beijing)
Geographic Thinking

某系统演示



ESRI China (Beijing)
Geographic Thinking

其它一些案例

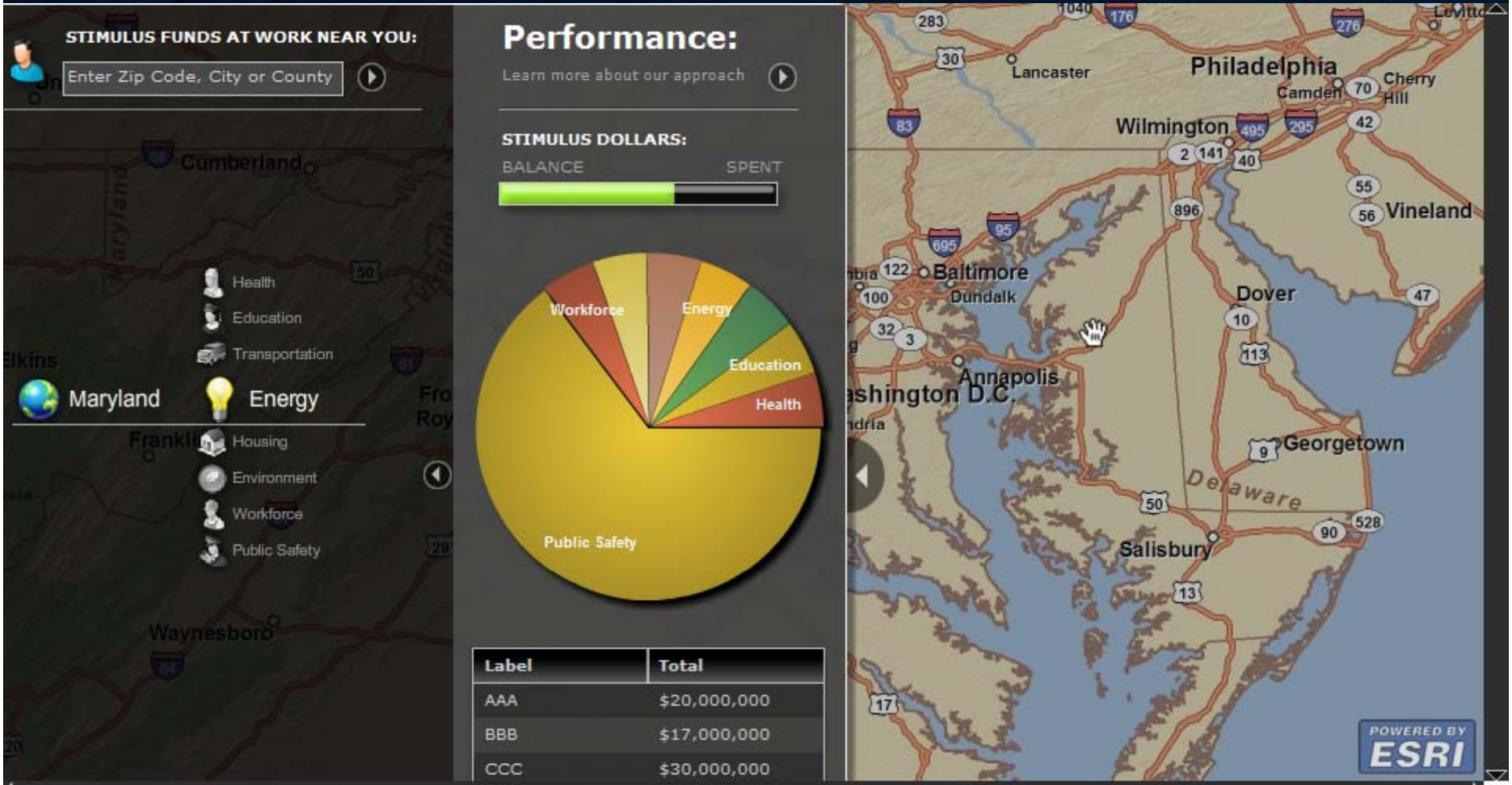


ijing)





Geographic Thinking


STIMULUS FUNDS AT WORK NEAR YOU:
Enter Zip Code, City or County
Cumberland
Health
Education
Transportation
Maryland
Energy
Housing
Environment
Workforce
Public Safety
Waynesboro
Performance:
Learn more about our approach
STIMULUS DOLLARS:
BALANCE
SPENT
Workforce
Energy
Education
Health
Public Safety
Label
Total
AAA
$20,000,000
BBB
$17,000,000
CCC
$30,000,000
Lancaster
Philadelphia
Camden
Cherry Hill
Wilmington
Vineland
Baltimore
Dundalk
Dover
Annapolis
Georgetown
Delaware
Salisbury
POWERED BY ESRI