

2011 Esri China
Developer Summit

2011Esri中国开发者大会

ArcGIS Microsoft Silverlight/WPF API——初级篇

Esri中国（北京）有限公司 马克玲



分享地理价值

主要内容

- 总览
- 构建应用
 - 可配置的Viewer
 - 开始构建第一个Silverlight 地图应用
- 新动向

ArcGIS 10 —— 完整的系统

更容易
更给力
无处不在



云

企业

本地

ArcGIS API for Microsoft Silverlight/WPF

- 强大的Web和桌面地图API
- 设计用于Silverlight和WPF平台
- 使用ArcGIS Server/ Online, Bing Maps&services
- WPF支持本地64位 (x64) 平台
 - 添加了对本地GIS运行时的支持
- 整合到微软开发环境



需求

- 工具

 - Microsoft Visual Studio 2010

 - Silverlight 4 Tools for Visual Studio 2010

 - Expression Blend 4

- 平台

 - Silverlight 4

 - .NET 4

 - .NET 3.5 sp1(WPF only)

类库

- ESRI.ArcGIS.Client.dll
 - Map, ArcGIS Layers, Graphics, Geometry, Symbols
- ESRI.ArcGIS.Client.Bing.dll
 - 用于对Bing layers和 服务的支持
- ESRI.ArcGIS.Client.WebMap.dll
 - 包含用于读取和执行WebMap内容和元数据的组件
- ESRI.ArcGIS.Client.Behaviors.dll
 - 定义用于与地图及其内容进行交互的元素
- ESRI.ArcGIS.Client.Toolkit.dll*
 - 用于导航和交互地图的通用控件
- ESRI.ArcGIS.Client.Toolkit.DataSources.dll*
 - 包含支持通用数据源的一套图层

*CodePlex URL: <http://esrisilverlight.codeplex.com>

构建应用 —— 不只有一种方法...

•作为开发者，构建Silverlight 应用可以使用以下工具完成：

✓Visual Studio 2010

✓Expression Blend 4

✓KaXAML

✓Xamlon

✓MyXaml

✓More...

IDE 整合

- Design – time 支持
 - Expression Blend 4 & Visual Studio 2010
- Application 模板

没有开发人员，但想要创建一个客户化的解决方案怎么办？

ArcGIS Viewer for Microsoft Silverlight

- 可配置的方便部署的 Web 客户端
 - 使用ArcGIS API for Silverlight/WPF 2.2构建
- 很容易并且快速的创建和实现GIS Web地图应用



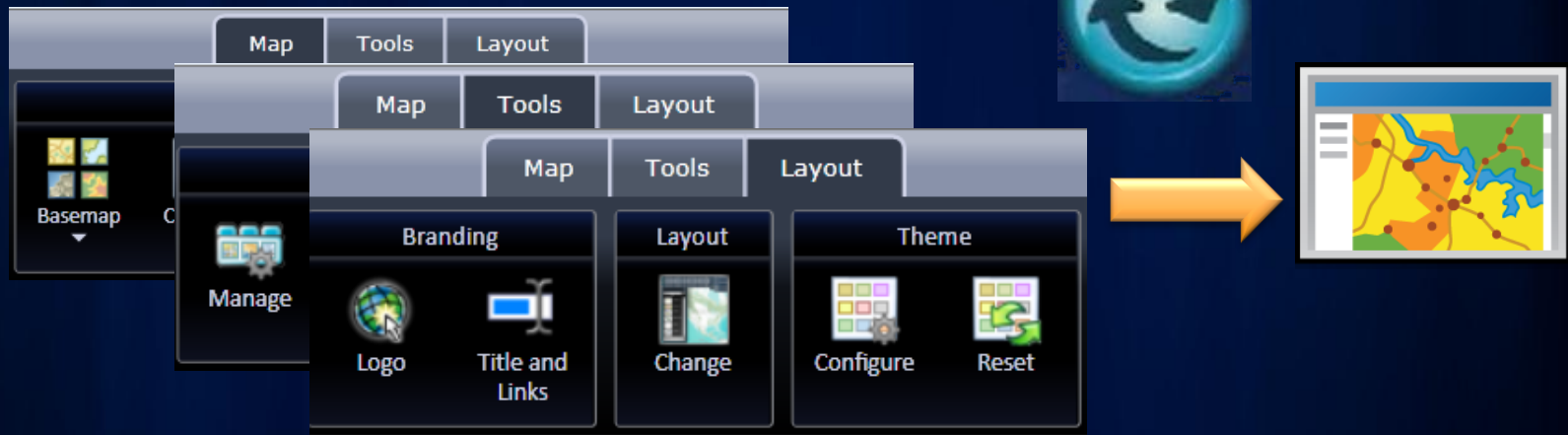
- 完成一套核心的GIS工具和功能
- 不用编程或编辑配置文件
 - 非常适合Web 应用创建的新手

Viewer 概览

- 基于Web浏览器的应用
- 由3个部件构成：
 - 可配置的Viewer
 - 应用构建器
 - 可扩展包
- 利用ArcGIS Server 10 的功能
- 支持ArcGIS.com Web地图
- 应用按预制模式宿主

Application Builder

- 交互式创建Web apps -> 所见即所得的用户体验
 - 地图内容
 - 工具
 - 外观和体验



- 发布可配置的Viewer 应用

Demo1: 预览Viewer

```
function init() {  
  var map = new esri.Map("map");  
  var tl = ledmapServiceLayer = new  
  esri.Layer({url: "http://services.esri.com/arcgis/rest/services/Esri_Imagery",  
    map: map, addLayer: tl, ledmapServiceLayer});  
}  
  
function getDriveTimePolys(results) {  
  var features = results[0].features;  
  for (var f=0; f<features.length; f++) {  
    var feature = features[f];  
    if (feature.attributes["time"] < 30) {  
      var polys = feature.attributes["polys"];  
      new dojo.Color(0, 0, 0, 0.5).stroke(polys, 1, true);  
      new dojo.Symbol(Stroke, 0, 0, 0, 0.5).stroke(polys, 1, true);  
      feature.attributes["stroke"] = polys;  
    }  
  }  
}
```

Application Builder

- 配置客户化的Web应用
 - 设置属性用以满足商业需求
 - 可伸缩的工作流
- 很容易发布和部署
- 修改一个现有已发布的站点
 - 使用Builder做配置改变
- 可选的 “Getting started mode”
- Builder可以通过 “add-ins” 扩展



可扩展包

- Viewer功能通过“ add-ins” 定义
 - “Add-ins” -> module code
- 开发者可以创建自定义的 “add-ins”
 - 需要ArcGIS API for Silverlight 2.2
- Viewer能力可以被扩展
 1. Tools -> 客户化功能
 2. Behaviors -> 用以客户化非UI逻辑
 3. Layouts -> 客户化应用的 “外观和体验”
 4. Layers -> 客户化数据层类型

Viewer功能亮点

- Application builder – 交互性WYSIWYG UE
 - 无需编程或配置文件编辑
- 可伸缩的配置 workflow
- 可配置的Geoprocessing(GP)工具
- 多个外观选项
- 部署后web 应用仍可编辑
- 可扩展包

ArcGIS Viewer for Silverlight – 时间线

- 现在发布了Beta 版
 - 可以从ArcGIS Beta Community下载
<https://betacommunity.esri.com/>
- 最终
 - 目标：2011年7月（美国用户大会）

Demo2：开始创建Silverlight地图应用

- Resource Center

- Interactive SDK
- Documentation
- Samples
- CodePlex

- 构建第一个应用

- 一个简单的应用
- 客户化模板

新动向

- V2.2目前已经发布
 - ArcGIS API for Silverlight
 - ArcGIS API for WPF
 - ArcGIS API for Windows Phone
- 最终版本会在2011年6月发布

V2.2中的新特性

- WGS 84和Web Mercator空间参考，地图支持以环绕模式连续漫游——WrapAround
- 加强了ArcGIS.com整合——在Web地图中支持地图备注和弹出
- 提供新的graphic shape绘制类型：Arrow, Triangle, Circle和Ellipse
- 支持交互式缩放和旋转graphics
- 提供新的ScaleLine和Attribute控件

Demo3: V2.2先体验

```
function onInfo {  
    var map = new esri.Map("map");  
    var tl = ledmapServiceLayer = new  
    esri.layers.ArcGISLayer("ces/esri_imagery",  
    "ces/esri_imagery", true);  
    map.addLayer(tl);  
}  
  
function getDriveTimePolys(results) {  
    var features = results[0].features;  
    for (var f=0; f<features.length; f++) {  
        var feature = features[f];  
        if (feature.attributes["type"] == "road") {  
            var polys = feature.attributes["polys"];  
            for (var i=0; i<polys.length; i++) {  
                var poly = polys[i];  
                new dojo.Color(0, 0, 0, 0.5);  
                new dojo.Symbol("stroke", {color: new dojo.Color(0, 0, 0, 0.5)});  
                feature.attributes["stroke"] = new dojo.Symbol("stroke", {color: new dojo.Color(0, 0, 0, 0.5)});  
            }  
        }  
    }  
}
```



分 享 地 理 价 值