



ArcServer Flex API on an Android Device

Aaron Nash



Connecticut GIS User to User Network
A voluntary association of GIS Users



ArcServer Flex API on an Android Device



What is Android?

- Is it an Australian Rock Band?
 - No that's The Androids
- Is it a type of anabolic steroid?
 - Sure...but who uses methyltestosterone anymore...at least I stopped
- Is it a Robot designed to look like a human?
 - It is but I have not seen any lately.

Google's Android



- Android is a software stack for mobile devices that includes an operating system, middleware and key applications.
- Its an open source operating system for mobile devices based off the Linux Kernel.
- Google purchased Android Inc. in 2005



Why Android?

- Google's Open Source Platform
- More Options Available
 - Multiple devices
 - Multiple networks
- Android is catching ground
 - Who knows when the Android Market will overtake Apple's App Store
- Google has a plan
- A lot of Developer Help
 - XDA Forums
 - Android Forums



Why Adobe Air not Java?

- Vernon already paid for a license of Flash Builder
- Provides a development environment for the delivery of applications across devices and platforms for:
 - Desktop operating systems
 - Android devices, including smartphones and tablets
 - BlackBerry PlayBook
 - iOS devices including the iPhone and iPad
 - Supported television devices
- Adobe Air is free



ADOBE® AIR™



You Can Port Over Your Existing ArcServer Flex Code

- You can...really...How you ask?...Easily
 - All you need is Flash Builder 4.5
 - The new SDK Flex 4.5 (Open Source)
 - Android SDK (not needed but helps)
 - Android Phone (also not needed but a must to show off your new program)
- ESRI's new ArcServer Flex API supports mobile devices



ESRI

Is it Hard to do?

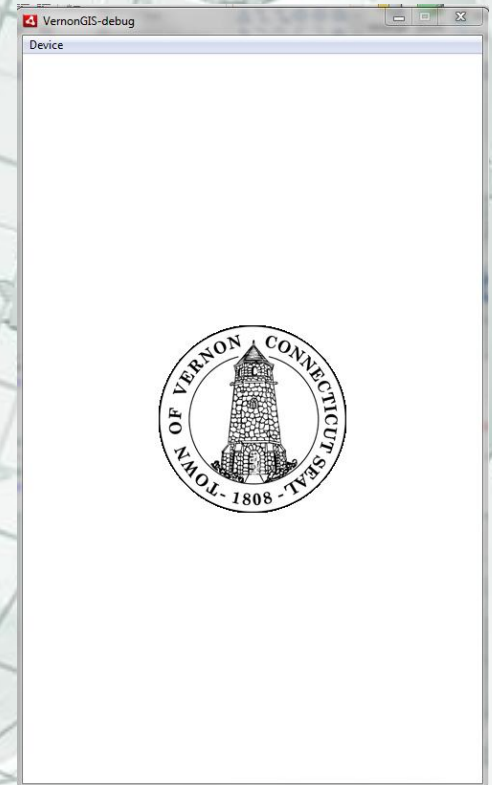
- Not at all...
- You can use any Flex Projects or code samples already created
- ESRI Flex API Resource center
 - Lots of code samples
- Lots of Sample Adobe Air projects and help
 - Tour De Mobile Flex



Step 1

- Create your Flex Mobile Project
- Application type with Splash screen

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <s:ViewNavigatorApplication xmlns:fx="http://ns.adobe.com/mxml/2009"
3     xmlns:s="library://ns.adobe.com/flex/spark"
4     firstView="views.VernonGISHomeView"
5     splashScreenImage="@Embed('assets/VernonSeal4_300.png') ">
6 <fx:Declarations>
7     <!-- Place non-visual elements (e.g., services, value objects) here -->
8 </fx:Declarations>
9 </s:ViewNavigatorApplication>
10
```

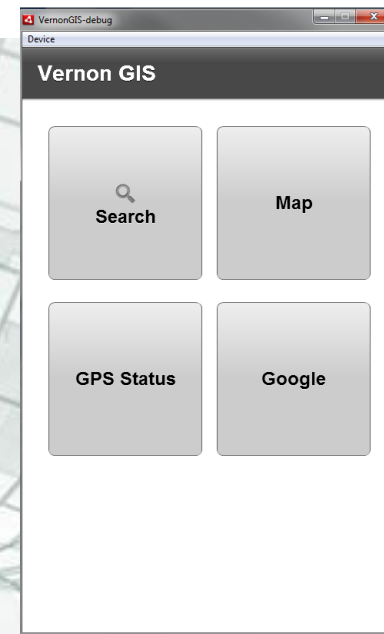
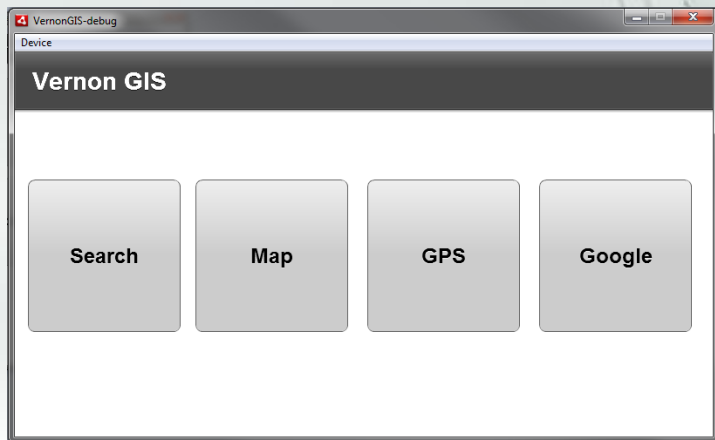


Step 2

■ Create your First View

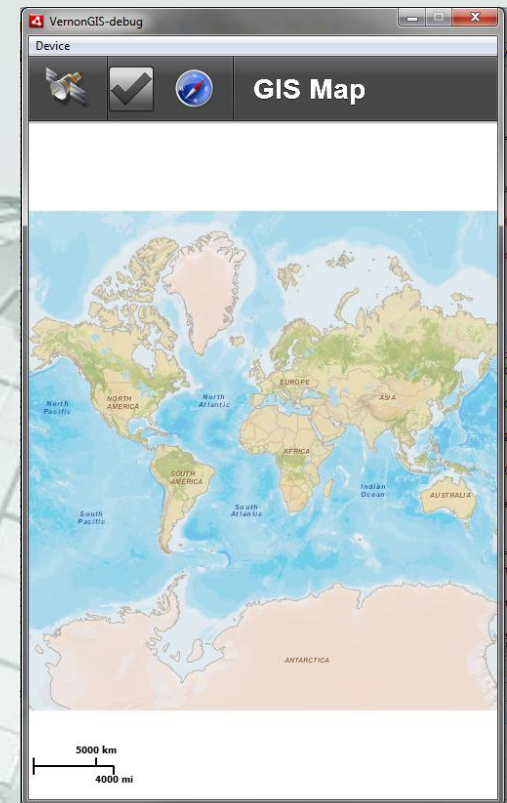
```
</fx:Declarations>
<s:states>
  <s:State name="portrait"/>
  <s:State name="landscape"/>
</s:states>
<s:Button includeIn="portrait" x="34" y="36" width="200" height="200" label="Search" click="SearchChange(event)" icon="@Embed(s
<s:Button includeIn="portrait" x="253" y="36" width="200" height="200" label="Map" click="MapChange(event)"/>
<s:Button includeIn="portrait" x="253" y="265" width="200" height="200" label="Google" click="SearchGoogle(event)"/>
<s:Button includeIn="portrait" x="34" y="265" width="200" height="200" label="GPS Status" click="GPSChange(event)"/>

<s:Button includeIn="landscape" x="15" y="80" width="175" height="175" label="Search" click="SearchChange(event)"/>
<s:Button includeIn="landscape" x="207" y="80" width="175" height="175" label="Map" click="MapChange(event)"/>
<s:Button includeIn="landscape" x="404" y="80" width="175" height="175" label="GPS" click="GPSChange(event)"/>
<s:Button includeIn="landscape" x="601" y="80" width="175" height="175" label="Google" click="SearchGoogle(event)"/>
/s:View>
```



Map View

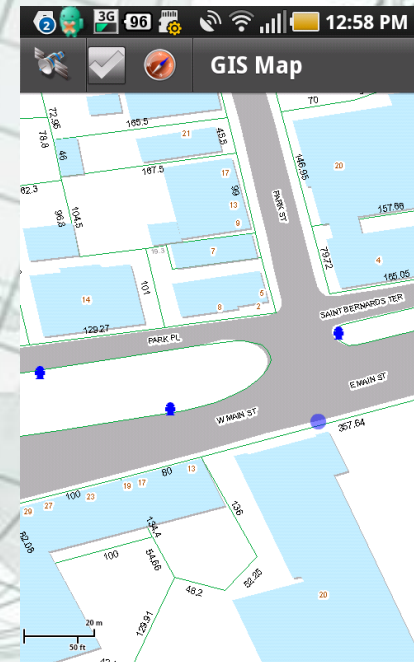
- How easy is this!!!!!!
- Same ArcServer code



```
<esri:Map id="map" logoVisible="false" navigationClass="{null}" zoomSliderVisible="false" mapClick="myClickHandler(event)">
  <!--esri:ArcGISDynamicMapServiceLayer url="http://gis.vernon-ct.gov/ArcGISweb/rest/services/Flex_Vernon/MapServer/-->
  <!--esri:ArcGISDynamicMapServiceLayer url="http://gis.vernon-ct.gov/ArcGISweb/rest/services/Fire_Hydrant/MapServer/-->
  <esri:ArcGISTiledMapServiceLayer url="http://services.arcgisonline.com/ArcGIS/rest/services/World_Street_Map/MapServer"/>
  <esri:GraphicsLayer id="myGraphicsLayer">
    <esri:symbol>
      <esri:SimpleMarkerSymbol alpha="0.5" color="0xFF0000" size="19" style="circle"/>
    </esri:symbol>
  </esri:GraphicsLayer>
  <esri:GraphicsLayer graphicProvider="{data.features}">
    <esri:symbol>
      <esri:SimpleFillSymbol color="red" alpha="0.5">
        <esri:outline>
          <esri:SimpleLineSymbol color="yellow" alpha="1.0"/>
        </esri:outline>
      </esri:SimpleFillSymbol>
    </esri:symbol>
  </esri:GraphicsLayer>
</esri:Map>
```

Who Really Needs to Know Where They are?

- I do...I may be a Geographer but I get lost and so do most of my users.
- There is a full Geolocator API available in flex
- You can access:
 - Longitude
 - Latitude
 - Accuracy
 - Altitude
 - Time
 - Speed



Here is the Code

- The first function initializes the Geolocation event and sets the update parameters
- The second function creates the variables, the map point and map graphic
- Simple...easy!!!

```
private function locate():void
{
    if (Geolocation.isSupported==true)
    {
        //Initialize the location sensor.
        geolocation = new Geolocation();
        geolocation.setRequestedUpdateInterval(10000);
        geolocation.addEventListener(GeolocationEvent.UPDATE, onTravel);
    }
    else
    {
        Alert.show("Geolocation feature not supported");
    }
}

private function onTravel(event:GeolocationEvent):void
{
    var long:String = event.longitude.toString();
    var lat:String = event.latitude.toString();
    var longnew:Number = Number(long);
    var latnew:Number = Number(lat);
    var centerPoint:MapPoint = new MapPoint(longnew, latnew);
    var graphic:Graphic = new Graphic();
    graphic.geometry = centerPoint;
    myGraphicsLayer.clear();
    myGraphicsLayer.add(graphic);
    if (GPSCheck.selected==true)
    {
        map.centerAt(centerPoint);
        map.zoom(500);
    }
    else
    {
        //geolocation.removeEventListener(GeolocationEvent.UPDATE, onTravel);
    }
}
```

Lets at Some More Functionality

- How about a Query on an ArcServer Web Service
- Again...same code from the ArcServer Flex API

```
protected function doquery(event:MouseEvent):void
{
    progressBar.visible = true;
    const query:Query = new Query();
    query.returnGeometry = true;
    query.where = "PROP_STREE like '%" + textInput.text + "%'";
    query.outFields = ['*'];
    const queryTask:QueryTask = new QueryTask;
    queryTask.url = "http://gis.vernon-ct.gov/ArcGISweb/rest/services/Proval/MapServer/0";
    queryTask.showBusyCursor = true;
    queryTask.useAMF = false;
    queryTask.addEventListener(QueryEvent.EXECUTE_COMPLETE, executeCompleteHandler, false, 0);
    queryTask.execute(query);
}

private function executeCompleteHandler(event:QueryEvent):void
{
    progressBar.visible = false;
    navigator.pushView(FeatureList, event.featureSet);
}

}}>
</fx:Script>
<s:actionContent>
    <s:Button click="doquery(event)" icon="@Embed('/assets/search.png')"/>
</s:actionContent>
<s:layout>
    <s:VerticalLayout paddingLeft="5" paddingRight="5" paddingTop="10"/>
</s:layout>
<s:TextInput id="textInput" width="100%" text="Park Pl"/>
<mx:ProgressBar id="progressBar" mode="event" label="Searching database..." width="100%" indeterminate="
:View>
```

How Does it Look?

- I Created Three Separate Views



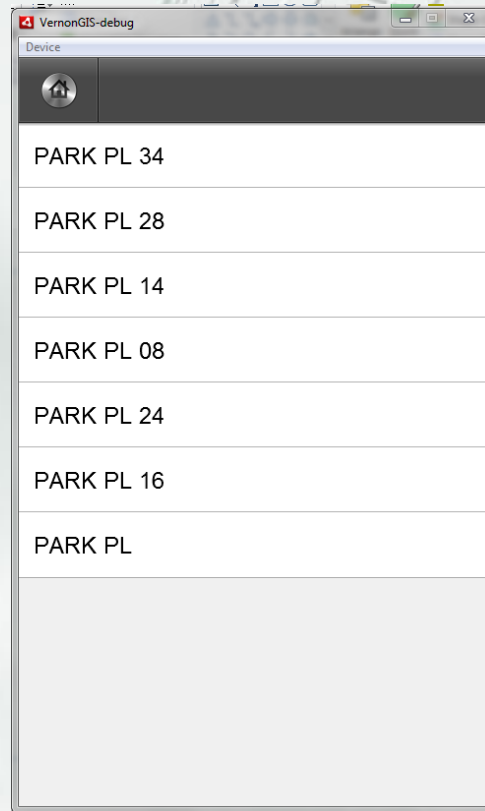
VernonGIS-debug

Device

Query

Park PI

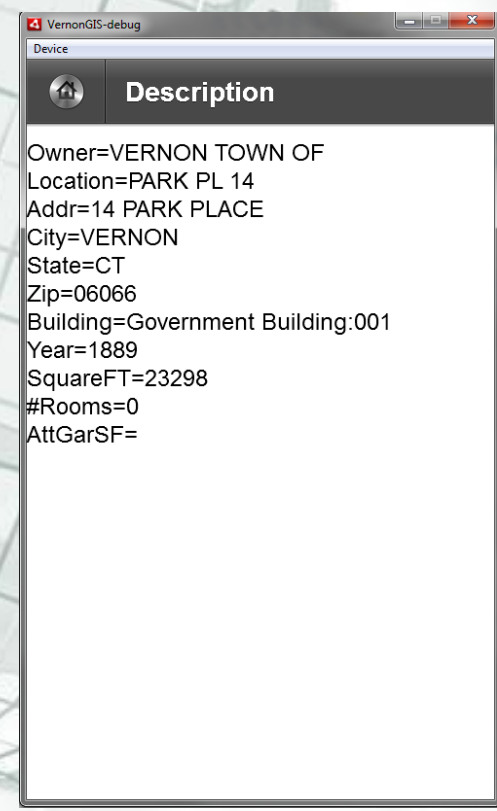
Searching database...



VernonGIS-debug

Device

PARK PL 34
PARK PL 28
PARK PL 14
PARK PL 08
PARK PL 24
PARK PL 16
PARK PL



VernonGIS-debug

Device

Description

Owner=VERNON TOWN OF
Location=PARK PL 14
Addr=14 PARK PLACE
City=VERNON
State=CT
Zip=06066
Building=Government Building:001
Year=1889
SquareFT=23298
#Rooms=0
AttGarSF=

How about Identifying a Feature

- To Difficult...or is it...?
- Same Sample Code from the ESRI resource center

```
private function mapClickHandler(event:MapMouseEvent):void
{
    clickGraphicsLayer.clear();

    var identifyParams:IdentifyParameters = new IdentifyParameters();
    identifyParams.returnGeometry = true;
    identifyParams.tolerance = 3;
    identifyParams.width = map.width;
    identifyParams.height = map.height;
    identifyParams.geometry = event.mapPoint;
    identifyParams.mapExtent = map.extent;
    identifyParams.spatialReference = map.spatialReference;
    var clickGraphic:Graphic = new Graphic(event.mapPoint, clickPtSym);
    clickGraphicsLayer.add(clickGraphic);
    identifyTask.execute(identifyParams, new AsyncResponder(myResultFunction, myFaultFunction));
}

private function myResultFunction(results:Array, clickGraphic:Graphic = null):void
{
    if (results && results.length > 0)
    {
        var result:IdentifyResult = results[0];
        var resultGraphic:Graphic = result.feature;
        switch (resultGraphic.geometry.type)
        {
            case Geometry.MAPPOINT:
            {
                resultGraphic.symbol = smsIdentify;
                break;
            }
            case Geometry.POLYLINE:
            {
                resultGraphic.symbol = sfsIdentify;
                break;
            }
            case Geometry.POLYGON:
            {
                resultGraphic.symbol = sfsIdentify;
                break;
            }
        }
        lastIdentifyResultGraphic = resultGraphic;
        // update clickGraphic (from mouse click to returned feature)
        clickGraphic.symbol = new InfoSymbol(); // use default renderer
    }
}
```


Additional Features

- Plans to add
 - Building photos
 - Additional Assessment data
- Use a Geometry Service to Reproject the Geolocation Event
- For the Iphone lovers
 - Port this code over to the IOS
- Somehow get an Android Tablet



Applications



- How Might Vernon use this App?
 - Fire Hydrant location
 - Zoning and Permit violations
 - Police Sidewalk Citations
 - Underground Utility Locations
 - Spring NEARC GPS Egg Hunt
- Will this Application go Public?
 - A versions will be published to the web and the Android Market for public use.

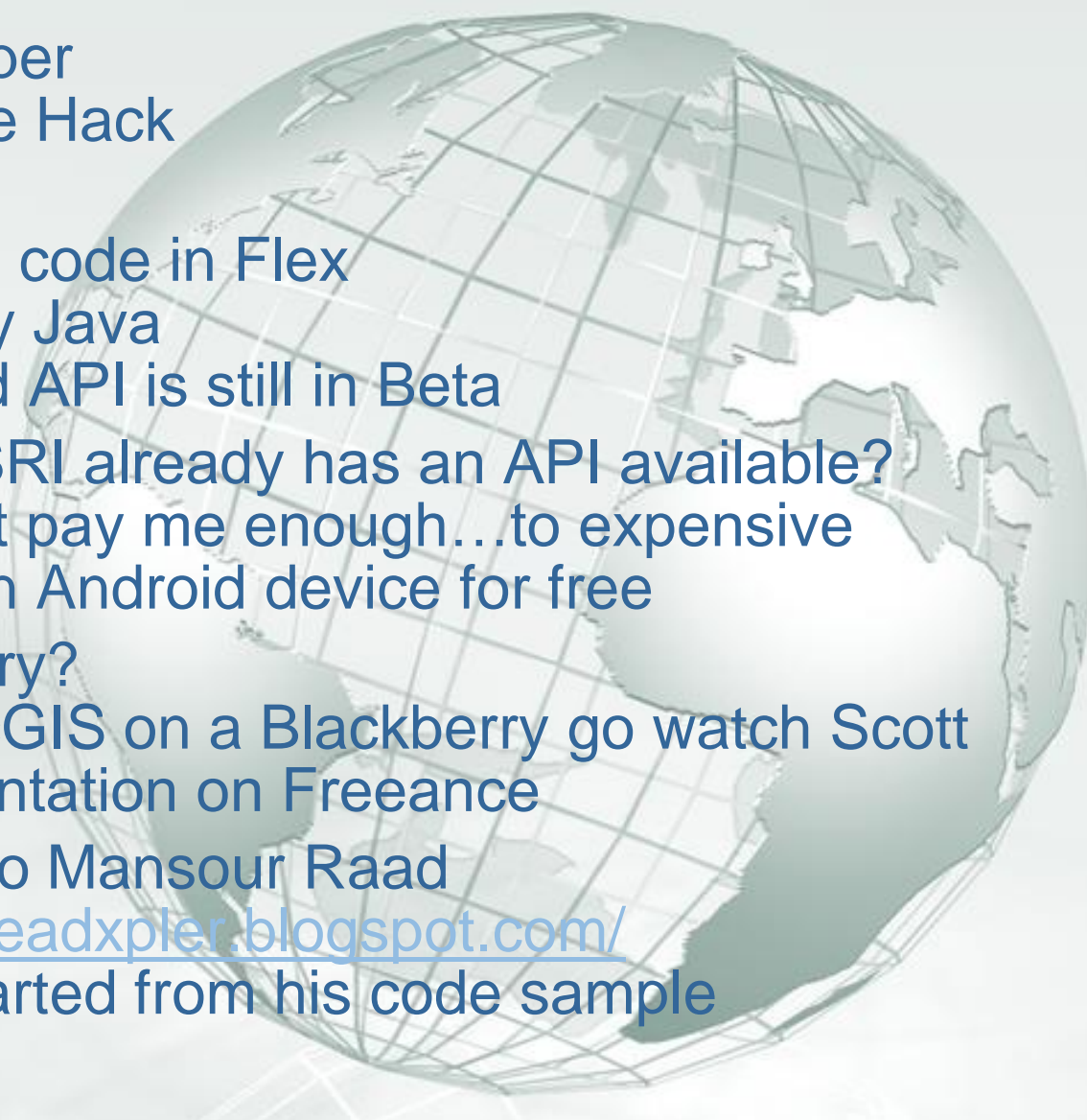


How Does This Compare to the ESRI Android API?

- Uses More Memory
- Dynamic Web Service
 - Slower
- Pinch Zoom not as responsive.
- Android API Specifically Designed for Mobile use



Closing

- I am not a developer
 - More of a Code Hack
 - Why not Java?
 - Already written code in Flex
 - Don't know any Java
 - ESRI's Android API is still in Beta
 - Why not IOS...ESRI already has an API available?
 - Vernon doesn't pay me enough...to expensive
 - Upgraded to an Android device for free
 - Why not Blackberry?
 - If you want ArcGIS on a Blackberry go watch Scott Robert's presentation on Freeance
 - A special thanks to Mansour Raad
 - <http://thunderheadxpler.blogspot.com/>
 - This project started from his code sample
- 

Closing

- I am not a developer
 - More of a Code Hack
- Why not Java?
 - Already written code in Flex
 - Don't know any Java
 - ESRI's Android API is still in Beta
- Why not IOS...ESRI already has an API available?
 - Vernon doesn't pay me enough...to expensive
 - Upgraded to an Android device for free
- Why not Blackberry?
 - If you want ArcGIS on a Blackberry go watch Scott Robert's presentation on Freeance
- A special thanks to Mansour Raad
 - <http://thunderheadxpler.blogspot.com/>
 - This project started from his code sample





ArcServer Flex API on an Android Device

Aaron Nash



Connecticut GIS User to User Network
A voluntary association of GIS Users

