

Application: **Solar Radiation Flex Widget**  
Version: **1.0**  
Date: **March 20, 2009**  
Author: **ESRI**



## Introduction

The ArcGIS Spatial Analyst extension provides tools to perform solar radiation analysis. An overview of these tools can be found at

[http://Webhelp.esri.com/arcgisdesktop/9.3/index.cfm?TopicName=An\\_overview\\_of\\_the\\_Solar\\_Radiation\\_tools](http://Webhelp.esri.com/arcgisdesktop/9.3/index.cfm?TopicName=An_overview_of_the_Solar_Radiation_tools).

Included with this download is a geoprocessing model in which you can point to your own data.

This document guides you through configuration of the Flex Web application with use of the geoprocessing service that is to be published in the above document.

## Requirements

To run the geoprocessing model through the Solar Radiation widget with your own data, you will need to first author and publish the model in ArcGIS Desktop with the Spatial Analyst extension. Then, to run it through the Web application, you will need ArcGIS Server with the Spatial Analyst extension. All tools assume ArcGIS 9.3 or higher.

Please see the SolarGeoprocessingWidget.pdf for instructions on configuring the geoprocessing model against your own data.

## *Directions to deploy the Solar Radiation widget in compiled/binary form*

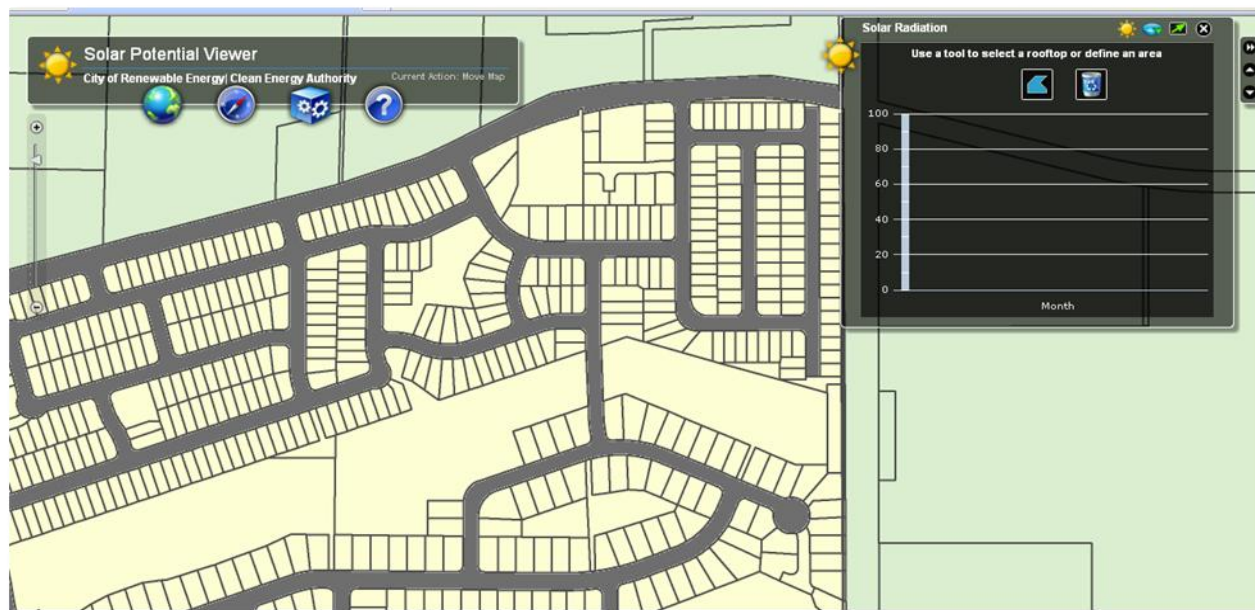
**Step 1:** Download the sample Flex viewer from the ESRI Resource Center and deploy it per the included instructions. The sample viewer and instructions can be found at <http://resources.esri.com/arcgisserver/apis/flex/index.cfm?fa=codeGalleryDetails&scriptID=15905>.

**Step 2:** Merge the contents of the bin directory with the deployed version of the sample Flex viewer. This can be done by copying the contents of SolarRadiationWidget\bin and pasting into the contents of the sample viewer from step 1, choosing to overwrite all contents.

**Step 3:** Test the application in a browser by entering the URL to the index.html page. The Solar Radiation widget will appear automatically. If you close the widget (clicking x in the upper right of the widget) and want it to appear again, it will be listed under the Tools menu.

Once you access your application through a URL, the sample viewer with the Solar Radiation widget should appear automatically. The application is configured to use Portland data from a sample server provided by ArcGIS Online. **Note: The Solar Radiation widget will not work at this point.** You must configure the model and geoprocessing service using your

own data. The viewer is only using the Portland data as a visual starting point. Steps 4 and 5 will allow you to configure the widget with your services.



**Step 4:** Configure the map to use your map services. Open the config.xml file in the folder representing the Flex viewer. You will want to change the BaseMaps and LiveMaps section to utilize map services for your location. At a minimum, you will want to change the URL in the following tag to show a map service you have published:

```
<mapservice label="Street Base" type="tiled" visible="true" alpha="1"
icon="com/esri/solutions/flexviewer/assets/images/icons/i_highway.png">http://sampleserver1.arcgisonline.com/ArcGIS/rest/services/Portland/ESRI\_LandBase\_WebMercator/MapServer</mapservice>
```

You will also want to change the initial extent to use an extent that applies to your map service, or leave the values in this tag blank:

```
<map initialExtent="-13673609 5710910 -13672256 5712067">
```

If you have not closed your Web browser, do so now and open a Web browser with the URL of your application to see your map service. Note that the geoprocessing service for the Solar Radiation widget will still not work. This will just ensure that your map is working.

**Step 5:** Configure the Solar Radiation widget to use your map and geoprocessing services. Open the SolarRadiationWidget.xml file found in <FlexApplicationPath>\com\esri\solutions\flexviewer\widgets. You will need to edit the following services:

Rooftoplayer—This needs to point to a layer within a map service that has building footprints and the associated interval values burned into it. See the Preprocessing data section in the SolarGeoprocessingModel\_Configuration.pdf guide on how to obtain the interval values.

If you do not have building footprints, you can leave the tag blank, which will remove the button from the widget to select an existing building.

```
<rooftoplayer> </rooftoplayer>
```

Gpservice—This needs to point to the geoprocessing service that gets created from publishing the geoprocessing model as mentioned in the SolarGeoprocessingModel\_Configuration.pdf guide. The tag should look similar to the following:

```
<gpservice>http://yourservernamehere/ArcGIS/rest/services/  
GPService/RooftopSolarCalculator</gpservice>
```

GeometryService—Currently, this is pointing to an ArcGIS Online sample server. You should create your own geometry service on ArcGIS Server as discussed at [http://Webhelp.esri.com/arcgisserver/9.3/dotNet/index.htm#geometry\\_service.htm](http://Webhelp.esri.com/arcgisserver/9.3/dotNet/index.htm#geometry_service.htm).

Conversionfactor—Currently, this is set to 0.092903, which represents the conversion of one square foot to one square meter. The default unit is in square meters. The default value would assume the base data is in feet (typical for a State Plane projection). If your base data unit of measurement is in meters, you can set the conversion factor to 0.

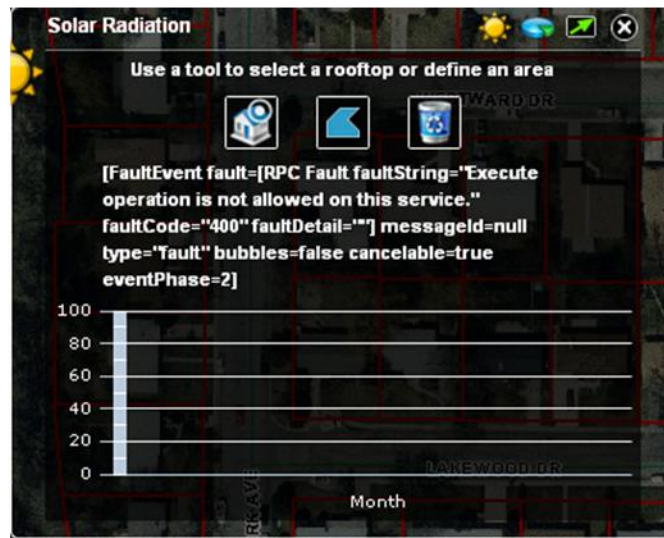
Once the gpservice has been configured, you can run the model again. You can click the button to draw a polygon, then double-click to complete the polygon; the geoprocessing service will execute, giving you back results.



Note that in the screen shot above, you have buttons to either select a rooftop or draw a polygon that represents one. You may not have the button to select an existing rooftop if you leave the <rooftoplayer> tag blank.

The geoprocessing service must be set up for synchronous execution. Refer to the SolarGeoprocessingModel\_Configuration.pdf guide for publishing your model to run as Synchronous. If it is not, you may see the error below:





**Step 6:** Configure the Renewable Resource Projects widgets. The Tools drop-down menu in the viewer has options to show existing renewable resource projects.



Open the widgets representing the renewable resources found in `<FlexApplicationPath>\com\esri\solutions\flexviewer\widgets`. The following .xml files correspond to the renewable projects under the Tools menu:

LiveLayer\_Solar.xml  
 LiveLayer\_Wind.xml  
 LiveLayer\_Biomass.xml  
 LiveLayer\_Hydro.xml

Each .xml file is a configuration where you can set the URL for the layer coming from a map service. This assumes that you have a map service with a point feature class for existing feature classes. You can also set the query and return fields via the .xml file.

If you do not have a feature class showing the existing renewable resource projects in your area, you can remove these widgets from the config.xml file in the root directory of the Flex

application.

### *Directions to import the Solar Radiation widget in source code*

**Step 1:** Download the sample Flex viewer from the ESRI Resource Center and create a new Flex Builder project per the included instructions.

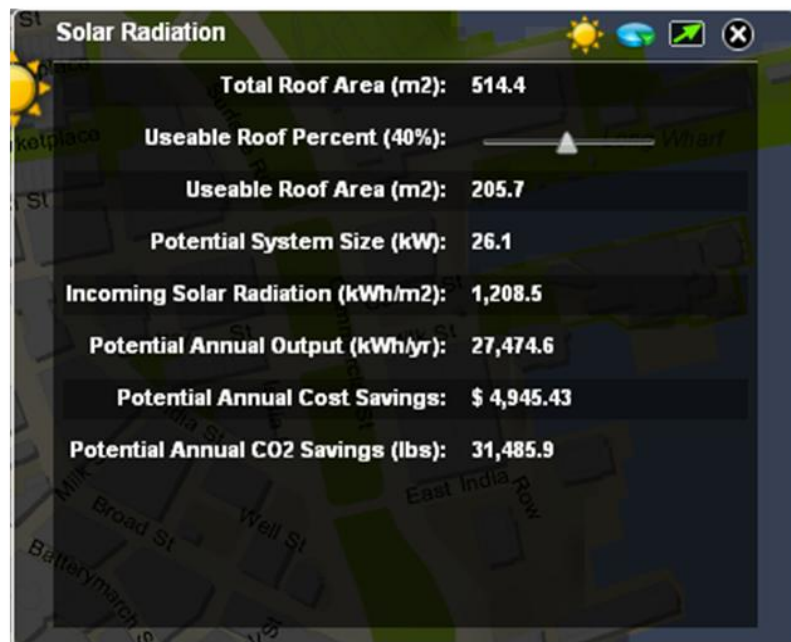
**Step 2:** Merge the contents of the src directory with the source code of the sample Flex viewer.

**Step 3:** In Flex Builder, from the Project menu, select Properties. In the Flex Modules Properties section, add SolarRadiationWidget as a module so that it will be compiled separately as SWF. See the *Developers Guide* for additional information.

**Step 4:** Test the application by running the project. The Solar Radiation widget should be listed under the Tools menu with the label Solar Radiation, as specified in the configuration file.

### *Modifying the widget*

You may want to modify the source code of the widget as it applies to your city. If you click on the Chart icon for the widget, a summary of the building or polygon is shown. **You may want to remove or modify some of these values, depending on your location.**



For instance, the widget provides values for Potential Annual Cost Savings. If your local city/county/utilities offer rebates for solar use, these values will differ. The default Cost Savings values are calculated at \$0.18 per kWh. The default Potential Annual CO2 Savings values are calculated at 1.146 per kWh.