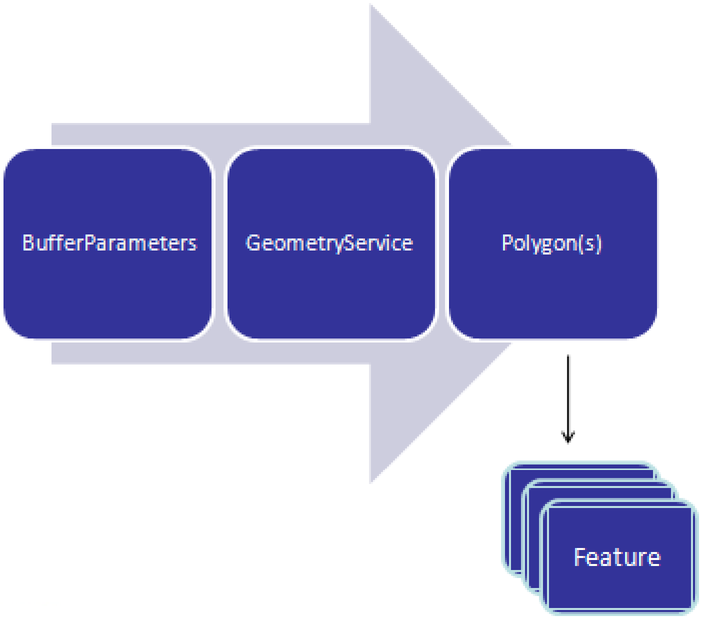
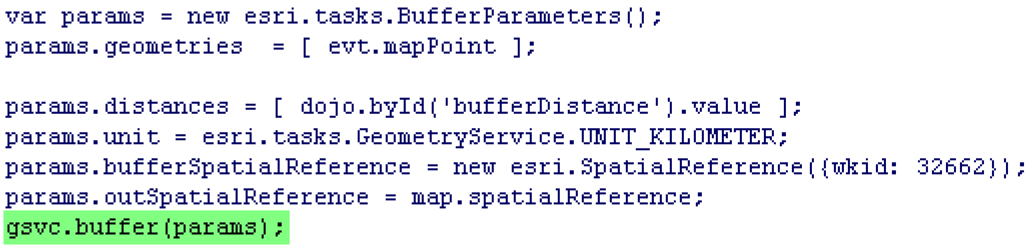
Chapter 17: Geometry Tasks

A GeometryService is used to perform various geometric operations including buffering of features, simplification of a geometry, projection to a new spatial reference, and the calculation of length and area. Each geometric operation fires a corresponding event upon completion and a graphic object is passed into this event. For example, when a buffer operation completes, the onBufferComplete(graphic) event is fired.

Buffering features is one of the more common functions in a GIS, and the ArcGIS Server API for JavaScript provides this capability through the GeometryService. The GeoemtryService.buffer( ) method accepts buffer parameters as an input and returns an output containing an array of graphic polygons which can then be plotted on your map.



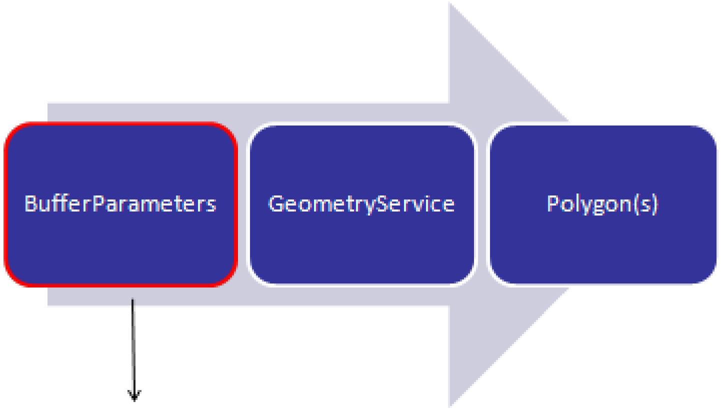


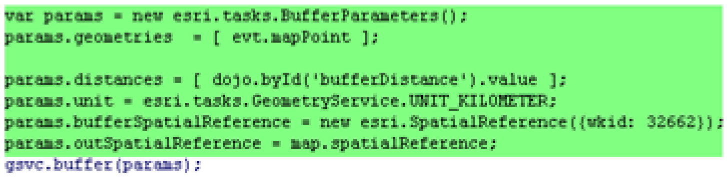
```
var params = new esri.tasks.BufferParameters();
params.geometries  = [ evt.mapPoint ];

params.distances = [ dojo.byId('bufferDistance').value ];
params.unit = esri.tasks.GeometryService.UNIT_KILOMETER;
params.bufferSpatialReference = new esri.SpatialReference({wkid: 32662});
params.outSpatialReference = map.spatialReference;
gsvc.buffer(params);
```

Buffering Features

Before a feature can be buffered you need to set various properties through the BufferParameters object. Properties include distances and units of measure. The distance property allows you to set multiple distances for your buffer operations using an array of numeric values. This is handy for creating multi-buffer polygons of various distances. The units property also needs to be set and can be set to feet, meters, kilometers, and miles. Other important parameters include features property which specifies the input feature to be buffered. UnionResults is a true/false value that indicates whether you should union all buffers into a single output polygon. Finally, the outSpatialReference property is used to specify an output spatial reference for the returned graphics.

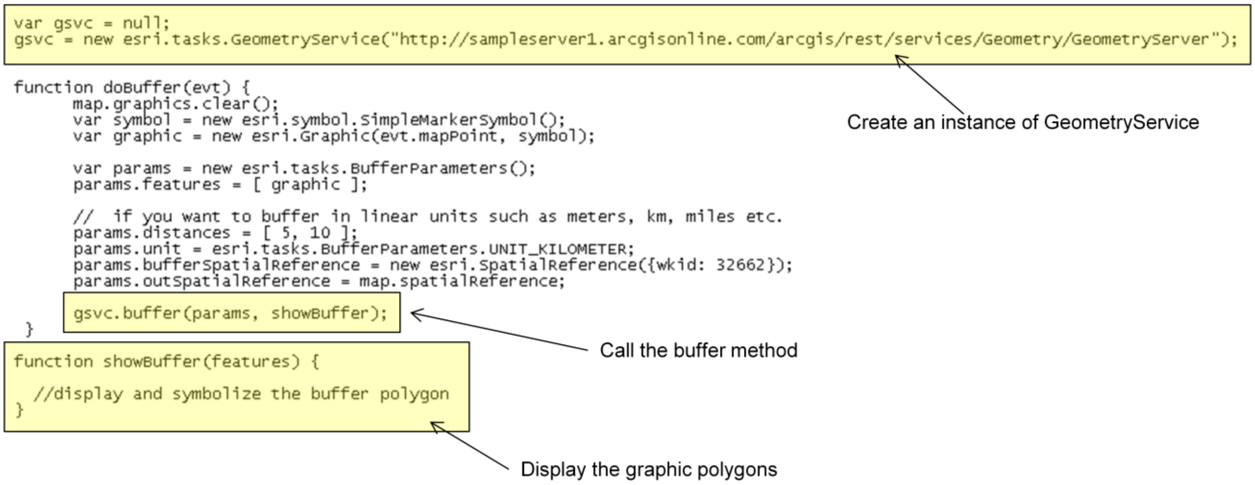




```
var params = new esri.tasks.BufferParameters();
params.geometries  = [ evt.mapPoint ];

params.distances = [ dojo.byId('bufferDistance').value ];
params.unit = esri.tasks.GeometryService.UNIT_KILOMETER;
params.bufferSpatialReference = new esri.SpatialReference({wkid: 32662});
params.outSpatialReference = map.spatialReference;
gsvc.buffer(params);
```

The buffer( ) method on GeometryService is used to perform the buffer using input parameters specified in a BufferParameters object. As you can see from the code example on this slide, when a new instance of a GeometryService is created you need to point to a URL containing a valid map service. A callback function is executed upon return from the buffer( ) method and an array of graphic polygons is passed into the callback function for processing.



```
var gsvc = null;
gsvc = new esri.tasks.GeometryService("http://sampleserver1.arcgisonline.com/arcgis/rest/services/Geometry/GeometryServer");

function doBuffer(evt) {
      map.graphics.clear();
      var symbol = new esri.symbol.SimpleMarkerSymbol();
      var graphic = new esri.Graphic(evt.mapPoint, symbol);

      var params = new esri.tasks.BufferParameters();
      params.features = [ graphic ];

      //  if you want to buffer in linear units such as meters, km, miles etc.
      params.distances = [ 5, 10 ];
      params.unit = esri.tasks.BufferParameters.UNIT_KILOMETER;
      params.bufferSpatialReference = new esri.SpatialReference({wkid: 32662});
      params.outSpatialReference = map.spatialReference;
      gsvc.buffer(params, showBuffer);
 }
function showBuffer(features) {

  //display and symbolize the buffer polygon
}
```

Create an instance of GeometryService

Call the buffer method

Display the graphic polygons

Calculating Areas and Lengths

Calculation of areas and lengths of polygons and polylines can be accomplished through two methods on GeometryService. The areasAndLengths( ) method can be used against polygons while the lengths( ) method is used against polylines. Both methods accept an array of Graphics as input and return length and area objects as numeric arrays.

Data Projection

The GeometryService also provides the capability of projecting your data to a new spatial reference through the project( ) method. This method accepts an array of graphics as input along with an output spatial reference specified in well known id format or WKID. The full list of projected and geographic coordinate systems WKIDS can be found at the link found on this slide. An array of graphics containing the newly projected data is returned.