

## Chapter 7: The FeatureLayer: Graphics on Steroids!

The ArcGIS Server API for JavaScript offers a feature layer for working with client-side graphic features. You should already be familiar with the graphics layer from the previous chapter. The feature layer inherits from the graphics layer, but offers additional capabilities such as the ability to perform queries and selections. Feature layers are also used for Web editing.

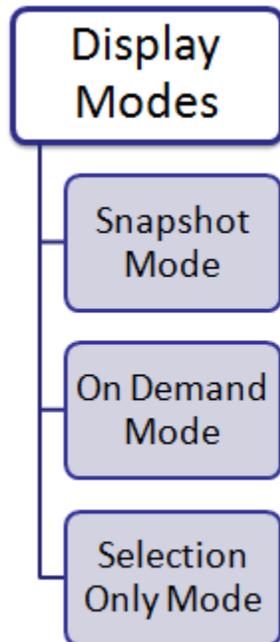
Feature layers differ from tiled and dynamic map service layers because feature layers bring geometry information down to the client computer to be drawn by the Web browser. Feature layers potentially cut down on round trips to the server. A client can request the features it needs, and perform selections and queries on those features without having to request more information from the server. Feature layers are especially appropriate for layers that respond to user interaction, such as a mouse click or hover.

A feature layer honors any definition queries, scale dependencies, and other properties configured on the layer in the map service. Using a feature layer, you can access related tables, perform queries, display time slices, work with feature attachments, and do various other useful things.



A feature layer must reference a layer from either a map service or a feature service. If the retrieval of geometries and attributes from layer is the only operation to be performed a map service will suffice. On the other hand feature editing and the use of symbols from the service's source map document require the use of a feature service. For editing, feature layers honor any feature templates that have been configured in the source map document.

The initial creation of a feature layer also requires that a display mode be selected. Display modes include snapshot mode, on-demand mode, and selection only mode. The mode specifies how and when features are transported from the server to the client browser. It is very important that the mode selected fit both the underlying data of the feature layer as well as the needs of the application. Because the mode determines when and how features are brought from the server to the client, the selection of a mode has a large affect on the speed and appearance of the application.



### **Snapshot Mode**

Snapshot mode retrieves all features from the layer when it is added to the map so you need to carefully consider the size of your layer before using this mode. Generally you will want to use this mode only with small datasets. Large datasets in snapshot mode can significantly degrade the performance of your application. The benefit of snapshot mode is that since all features from the layer are returned to the client there is no need to return to the server for additional data. This raises the potential for a significant boost in your application performance. However, it also raises the potential to significantly degrade performance if not used correctly.

ArcGIS imposes a limit of 1000 features that may be returned at any one time. This number is configurable through ArcGIS Server administration. Practically though you will want to only use this mode when you're working with small datasets of no more than a couple hundred features. Attempting to use snapshot mode with larger datasets can significantly decrease the performance and responsiveness of an application.

### **On Demand Mode**

On demand mode retrieves features only as needed. Essentially this means that only the features

in the current map extent are returned to the client. As the user navigates to other areas through zoom or pan operations a new set of features is retrieved from the server and streamed to the client. This tends to work well with large datasets that won't operate efficiently in snapshot mode. It does require a round trip to the server to fetch the features each time the extent changes, but for large datasets this is preferable.

### Selection Only Mode

Selection only mode does not initially retrieve features from the server. Instead, features are returned only when a selection is made on the client. Selected features are streamed to the client from the server when features are selected. These selected features are then held on the client.

### Creating a FeatureLayer

Below you will see a code example detailing how to create a FeatureLayer using its constructor. With tiled and dynamic layers you simply provide a pointer to the REST endpoint, but with FeatureLayer you need to point to a specific layer in the service. In this code example we are creating a FeatureLayer from the first layer. This is indicated by the number 0 which is the first layer. The constructor for FeatureLayer also accepts options such as the display mode, output fields, info template, and others. Here, the display mode is set to snapshot which would indicate that we are probably dealing with a relatively small dataset.

```
var earthquakes = new esri.layers.FeatureLayer("http://servicesbeta.esri.com/ArcGIS/rest/services/Earthquakes/Since_1970/MapServer/0",  
{  
  mode: esri.layers.FeatureLayer.MODE_SNAPSHOT, Display Mode is snapshot here  
  outFields: ["Magnitude"] Options  
}); Specific layer in a service
```

### Setting a Definition Expression

Definition expressions are used to limit the features that are displayed on a map. The FeatureLayer class contains a setDefinitionExpression() method that is used to create the expression. All features that meet the specified criteria will be returned for display on the map. Expressions are built using traditional SQL expressions. You can retrieve the currently set definition expression by using FeatureLayer.getDefinitionExpression which returns a string containing the expression.

### Feature Selection

FeatureLayer also supports feature selection which is simply a subset of features from a layer that is used for viewing, editing, analysis, or input to other operations. Features are added to or removed from a selection set using either spatial or attribute criteria and can easily be drawn with a different symbol than those used in the normal display of a layer. The selectFeatures(query) method on FeatureLayer is used to create a selection set and takes a Query object as input.

Any definition expression or time definition set on a layer either through the application or on the layer inside the map document file will be honored. Setting a symbol to use for the selected features is quite easy and simply involves creating a symbol and then using the 'setSelectionSymbol' method on FeatureLayer. Selected features will automatically be assigned

this symbol. You can opt to define a new selection set, add features to an existing selection set, or remove features from a selection set. In addition, you can define callback and errback functions to process the returned features or handle any errors.

```
var query = new esri.tasks.Query();  
query.geometry = feature.geometry;  
featureLayer.selectFeatures(query, esri.layers.FeatureLayer.SELECTION_NEW);
```