# Chapter 15: Turning Addresses into Points with Geocoding

In ArcGIS Server, a Locator task performs geocoding and reverse geocoding. Geocoding is the process of assigning a coordinate to an address while reverse geocoding assigns an address to a coordinate. The figure below illustrates this process. An address, defined by an object in JavaScript, is input to a Locator object which geocodes the address and returns the results in an AddressCandidate object which can then be displayed as a point with a SimpleMarkerSymbol on your map.


A Locator service can accept either a point or an object. An object is simply a generic term that indicates an address that is formatted in the form of an object as seen in the code example below. The address is defined as a series of name/value pairs defined within brackets. Individual name/value pairs are defined for the street, city, state, and zipcode in this case, but the name/value pairs will vary depending upon the type of geocoding service you've defined in your locator. For reverse geocoding the input to a Locator takes the form of a esri.geometry.point object which is often defined through a user click on the map or perhaps through application logic.

```
var address = {
    street: "380 New York",
    city: "Redlands",
    state: "CA",
    zip: "92373"
};
```
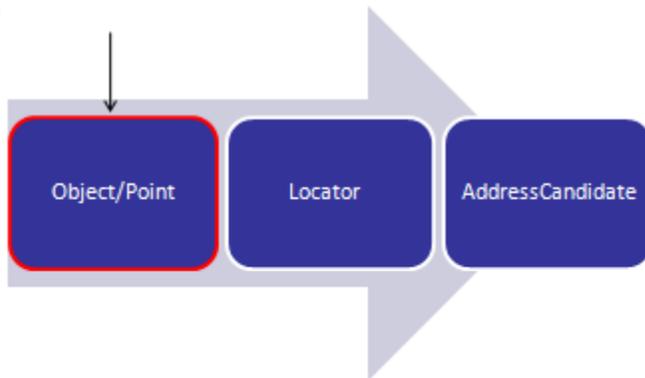

The Locator class contains methods and events that can be used to execute a geocode or reverse geocode operation using the input point or address object. Locator needs a URL pointer to your geocoding service as defined within ArcGIS Server. Once a new instance of a Locator has been created you can call the addressToLocations( ) method to geocode an address or the locationToAddress( ) method to perform a reverse geocode. These methods result in an event that is fired at the completion of the operation. In the case of an address geocode, the onAddressToLocationsComplete( ) event fires, and the onLocationToAddressComplete( ) event fires at the completion of a reverse geocode operation. In either case, an AddressCandidate object is then returned to the event.
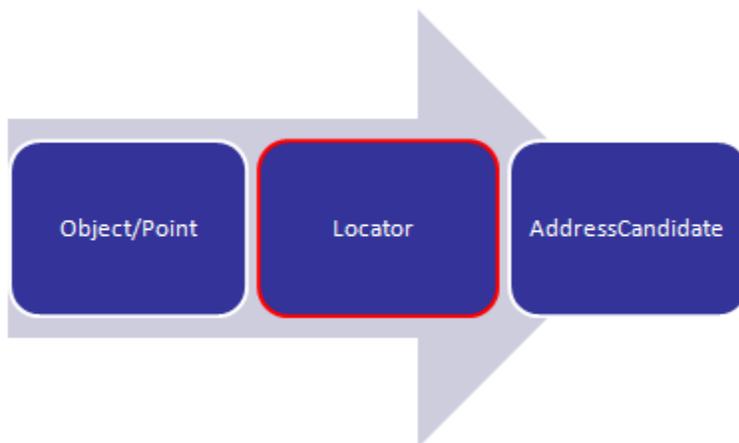

An AddressCandidate object is returned as a result of a Locator operation. Various properties are stored in this object including the address, attributes, location, and score. The attributes property contains name/value pairs of field names and values. The location is as the name would suggest the x,y coordinate of the candidate address, and the score property is a numeric

value between 0-100 that indicates the quality of address returned with a higher score representing a better match. Multiple addresses can be stored in this object as an array of candidates.
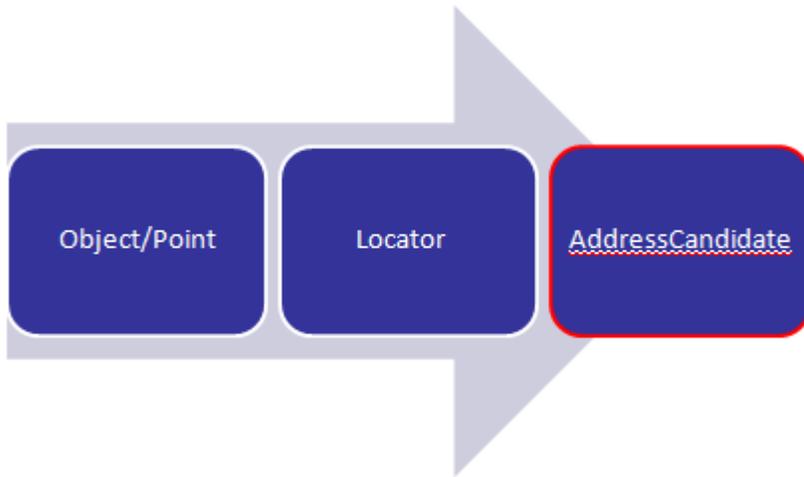

Let's take a little closer look at the Locator methods used to submit addresses and points. The Locator.addressToLocations method sends a request to geocode a single address. As you can see from the figure below, an input address object is created and used as a parameter in the addressToLocations method found on a Locator object. The results of the geocoding operation are returned in an AddressCandidate object. The address can then be plotted on the map as a graphic.
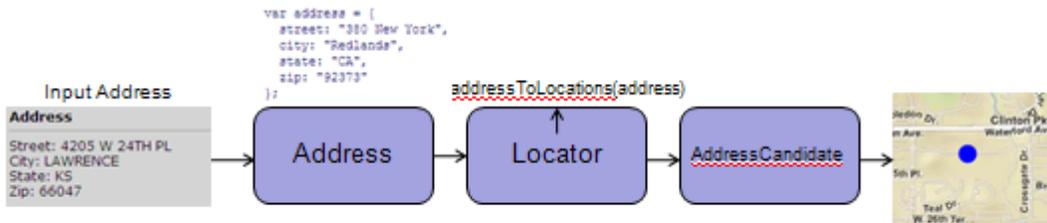

Reverse geocoding can also be performed by a Locator object through the locationToAddress method. A point object, created either by an end user click on the map or through application logic, is created and passed as a parameter into the locationToAddress method. A second parameter is also passed into the method indicating a distance in meters from the point within which a matching address should be searched. As with the addressToLocations method, an AddressCandidate object is returned from the locator and contains an address if one was found.

User Clicks a Point
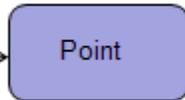
locationToAddress(thePt, 150)

Point → Locator → AddressCandidate
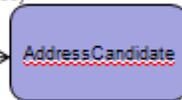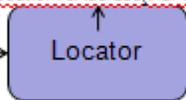
Address
Street: 4205 W 24TH PL
City: LAWRENCE
State: KS
Zip: 66047