

GeoREST Specification

Protocols for interacting with ArcGIS
Content



NYC Geoclient

GeoREST Specification

CHAPTER V: ADDRESS PROCESSING - FUNCTIONS 1, 1A, 1E

V.1 Introduction

This chapter and the following two chapters discuss in detail the various types of geographic data that Geosupport can accept as input, and the Geosupport functions that process them. This chapter discusses addresses and Functions 1, 1A and 1E. Chapter VI discusses properties (tax lots) and Functions 1A (covering aspects not discussed in Chapter V), BL and BN. Chapter VII discusses 'configurations' (geographic locations that are specified in terms of combinations of streets) and Functions 3, 3C and 3S.

The following topics are discussed in this chapter:

- (Section V.2) House number normalizing, and Geosupport's normalized house number (HNI (MSW only), HNS (COW only), and HND).
- (Section V.3) The two types of input data accepted by the address-processing functions

Geoclient

Geosupport
Documentation

↓ UPG PDF

↓ UPG Updates

🐱 Home

[Geoclient Version](#)

[Geosupport Version](#)

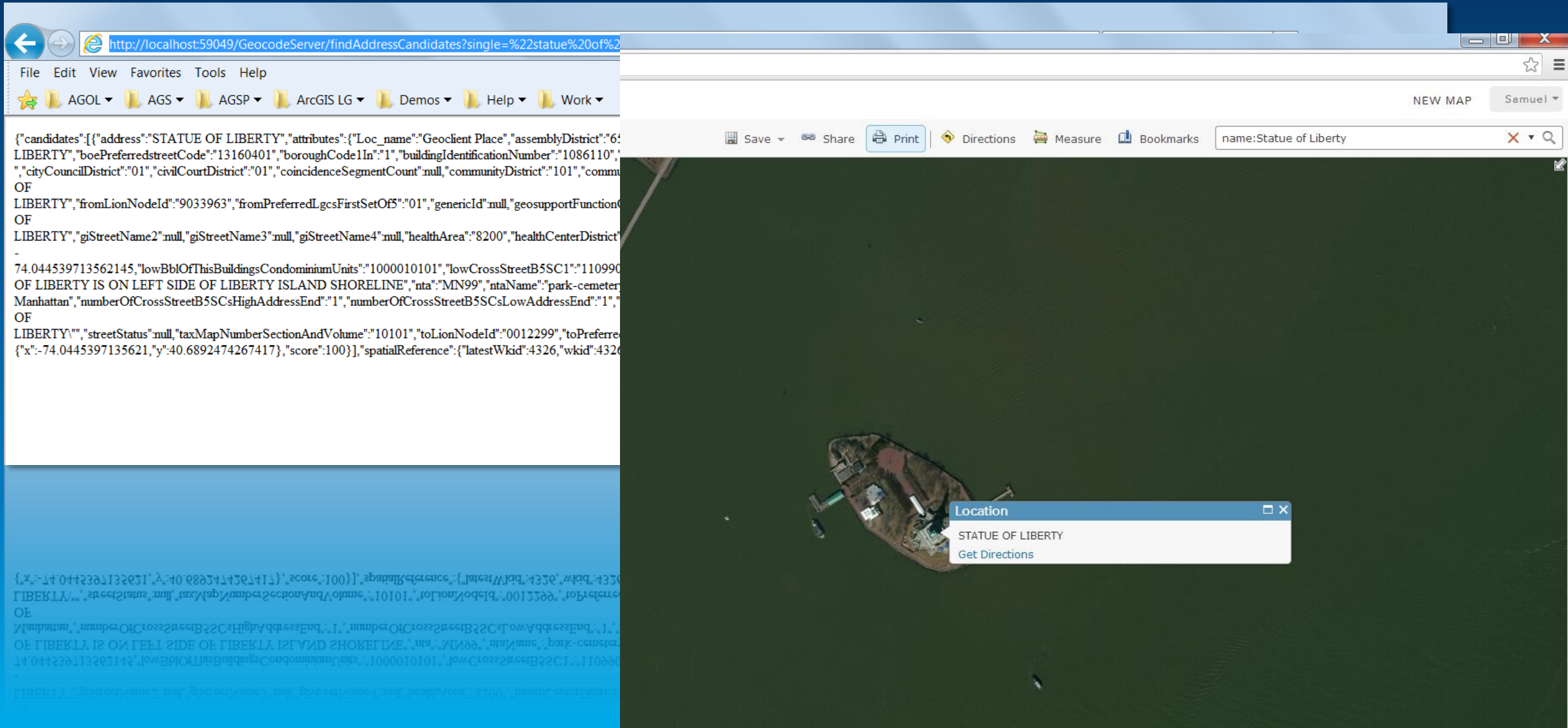
Geoclient API v1 (BETA)

The Geoclient application is a cloud based web service which exposes select function calls to the New York City Department of City Planning's **Geosupport** application. At the current time, there are six types of location requests that the Geoclient service provides:

Type	Description	Geosupport Function
Address	Given a valid address, provides blockface-level, property-level, and political information.	1B
BBL	Given a valid borough, block, and lot provides property-level information.	BL
BIN	Given a valid building identification number provides property-level information.	BN
Blockface	Given a valid borough, "on street" and cross streets provides blockface-level information.	3
Intersection	Given a valid borough and cross streets returns information for the point defined by the two streets.	2
Place	Same as 'Address' above using well-known NYC place name for input.	1B



Starting with Geoclient service version 1.10, any of these request types can be called using the new single field search endpoint. Service calls can now be made passing in a single line



```
{
- spatialReference: {
  wkid: 102100,
  latestWkid: 3857
},
- candidates: [
  - {
    address: "Boston",
    location: {
      x: -79103,
      y: 521482
    },
    score: 100,
    attributes: {
      Match_addr: "Boston",
      StAddr: "",
      City: ""
    },
    extent: {
      xmin: -79103,
      ymin: 519,
      xmax: -78,
      ymax: 523
    }
  },
  - {
    address: "Boston",
    location: {
      x: -89675,
      y: 370987
    },
    score: 100,
    attributes: {
      Match_addr: "Boston",
      StAddr: "",
      City: "Marrakech"
    },
    extent: {
      xmin: -897312.6854788678,
      ymin: 3709221.4407674773,
      xmax: -896199.4905709352,
      ymax: 3710528.314444919
    }
  }
]
}
```

Cache
TextWizard
Tearoff
MSDN Search.
Online

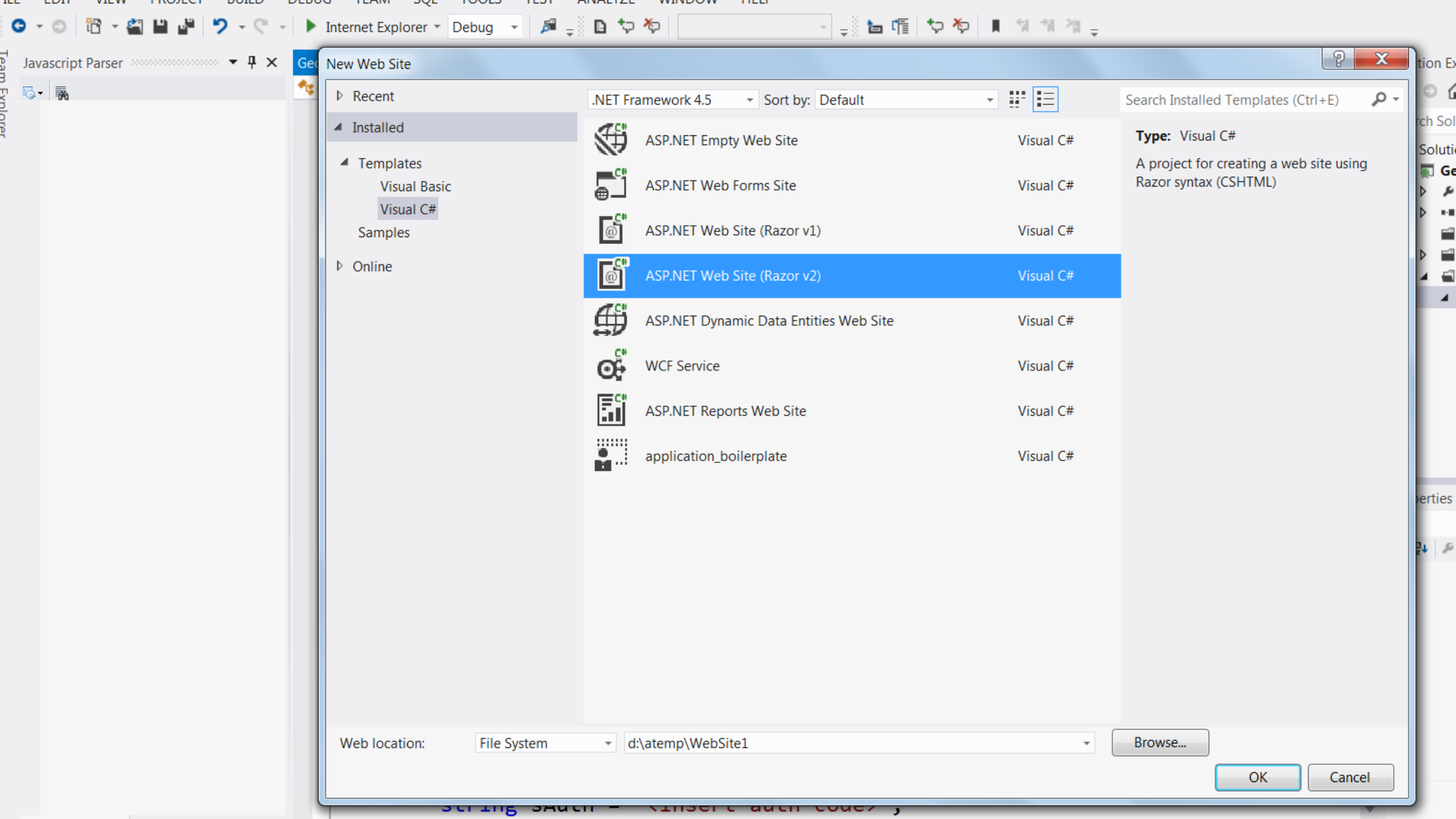
Statistics
Inspectors
AutoResponder
Composer
Filters
Log
Timeline

Headers
TextView
WebForms
HexView
Auth
Cookies
Raw
JSON
XML

Name	Value
SingleLine	boston,ma
f	json
outSR	{"wkid":102100}
outFields	Match_addr,StAddr,City
distance	50000
location	{"x":-8242600.665508177,"y":4966614.496095429,"spatialReference":{"wkid":102100}}
maxLocations	5

Response is encoded and may need to be decoded before inspection. Click here to transform.

Response is encoded and may need to be decoded before inspection. Click here to transform.



New Web Site

Recent

Installed

Templates

Visual Basic

Visual C#

Samples

Online

.NET Framework 4.5

Sort by: Default



ASP.NET Empty Web Site

Visual C#



ASP.NET Web Forms Site

Visual C#



ASP.NET Web Site (Razor v1)

Visual C#



ASP.NET Web Site (Razor v2)

Visual C#



ASP.NET Dynamic Data Entities Web Site

Visual C#



WCF Service

Visual C#



ASP.NET Reports Web Site

Visual C#



application_boilerplate

Visual C#

Search Installed Templates (Ctrl+E)

Type: Visual C#

A project for creating a web site using
Razor syntax (CSHTML)

Web location:

File System

d:\atemp\WebSite1

Browse...

OK

Cancel

```
JavaScript Parser  GeoCodeServerController.cs  GeoREST  find()

[HttpGet]
[ActionName("findAddressCandidates")]
public HttpResponseMessage find()
{

    #region parse format
    var matchesF = query.Where(kv => kv.Key.ToLower() == "f");

    this.format = "html";
    if (matchesF.Count() > 0)
    {
        this.format = matchesF.First().Value.ToLower();
        if (this.format.ToLower() != "html") result = "";
    }
    #endregion

    if (this.queryParams.SearchKey != null)
    {

        string sAuth = "<insert auth code>";

JavaScript Parser  Toolbox  100 %
Error List  Output  Find Results 1  Find Symbol Results  Immediate Window  Bookmarks  Python 64-bit 2.7 Interactive  Web Publish Activity  Python Environments  Python 2.7 Interactive  Python Debug Interactive
```

Parse format...

GeoREST.Controllers.GeocodeServerController find()

```
[HttpGet]
[ActionName("findAddressCandidates")]
public HttpResponseMessage find()
{
    #region parse search type
    if (d.ContainsKey("NAME"))
    {
        this.queryParams.searchObject = new PlaceSearch();
        (this.queryParams.searchObject as PlaceSearch).name = d["NAME"];
        this.queryParams.searchFile = "place.json";

        this.queryParams.searchURL = String.Format(this.GeoClientAPIURL + "{0}", d["NAME"]);
    }
    else if (d.ContainsKey("HOUSENUMBER"))
    {
        this.queryParams.searchObject = new AddressSearch(d);
        this.queryParams.searchFile = "address.json";

        this.queryParams.searchURL = String.Format(this.GeoClientAPIURL + "{0}", d["HOUSENUMBER"]);
    }
    else if (d.ContainsKey("LOT"))
    {
        this.queryParams.searchObject = new BBLSearch(d);
        this.queryParams.searchFile = "bbl.json";
    }
    }

    if (t
    {
        string sAuth = "<insert auth code>";
    }
}
```

JavaScript Parser Toolbox 100 %

Error List Output Find Results 1 Find Symbol Results Immediate Window Bookmarks Python 64-bit 2.7 Interactive Web Publish Activity Python Environments Python 2.7 Interactive Python Debug Interactive

Search Type...

```

using (HttpWebResponse response = request.GetResponse() as HttpWebResponse)
{
    Stream responseStream = copyStream(response.GetResponseStream());

    StreamReader reader = new StreamReader(responseStream);
    string s = reader.ReadToEnd();
    result = s;

    responseStream.Position = 0;

    PlaceResult placeResult = null;
    if (this.queryParams.searchObject.GetType() == typeof(PlaceSearch))
    {
        CandidatePlace cplace = new CandidatePlace();
        Place place = null;
        DataContractJsonSerializer placeSerializer = new DataContractJsonSer
        placeResult = (PlaceResult)placeSerializer.ReadObject(responseStream);
        place = placeResult.place;

        cplace.address = place.firstStreetNameNormalized;
        if (place.latitude == 0 && place.longitude == 0) this.nullResult = 1

        string sAuth = "<insert auth code>";

```

Javascript Parser Toolbox

100 %

Error List Output Find Results 1 Find Symbol Results Immediate Window Bookmarks Python 64-bit 2.7 Interactive Web Publish Activity Python Environments Python 2.7 Interactive Python Debug Interactive

Call Geoclient API

```

HttpWebRequest request = WebRequest.CreateHttp(this.queryParams.searchUF
.....

Geometry g = ensureWebMeractor(caddress.location.x, caddress
address.location.x = g.x;
address.location.y = g.y;

address.attributes = address;// new Attributes();
address.attributes.Loc_name = "Geoclient Address";
address.score = 100;

FindResultAddress fResult = new FindResultAddress();
fResult.spatialReference = new SpatialReference();
fResult.spatialReference.wkid = 4326;
fResult.spatialReference.latestWkid = 4326;

fResult.candidates = new List<CandidateAddress>();

if (!this.nullResult)
{
    fResult.candidates.Add(caddress);
}

```

HttpWebResponse

ream());

ceSearch))

ntractJsonSer
responseStrear

ullResult = 1

Javascript Parser | To

Error List | Output | Find Results 1 | Find Symbol Results | Immediate Window | Bookmarks | Python 64-bit 2.7 Interactive | Web Publish Activity | Python Environments | Python 2.7 Interactive | Python Debug Interactive

Generate Result...

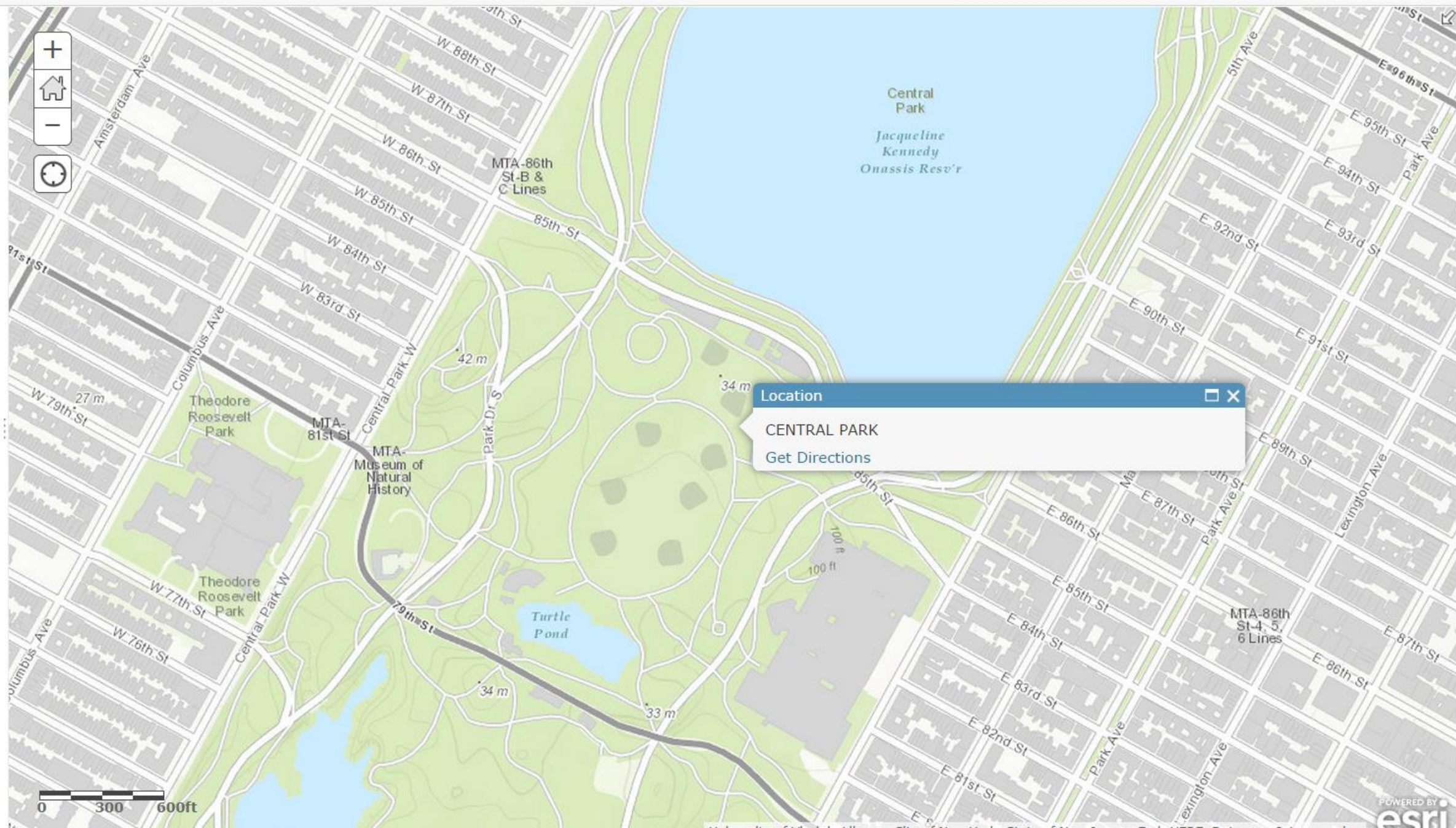
started

he map

ap

p

ri



Location

CENTRAL PARK

Get Directions

PLACE

name:statue of liberty

Anything else

ADDRESS

houseNumber: 314 street:west 100 st

(st, or street, or str, or pl, or ave, or avenue)

BIN

bin:1079043

(single number)

BBL

Block:1889 lot:1

Looking for block and lot

BLOCKFACE

onStreet:amsterdam ave crossStreetOne:w

110 st crossStreetTwo:w 111 st

(no lat lon for blockface)

INTERSECTION

crossStreetOne:broadway crossStreetT