

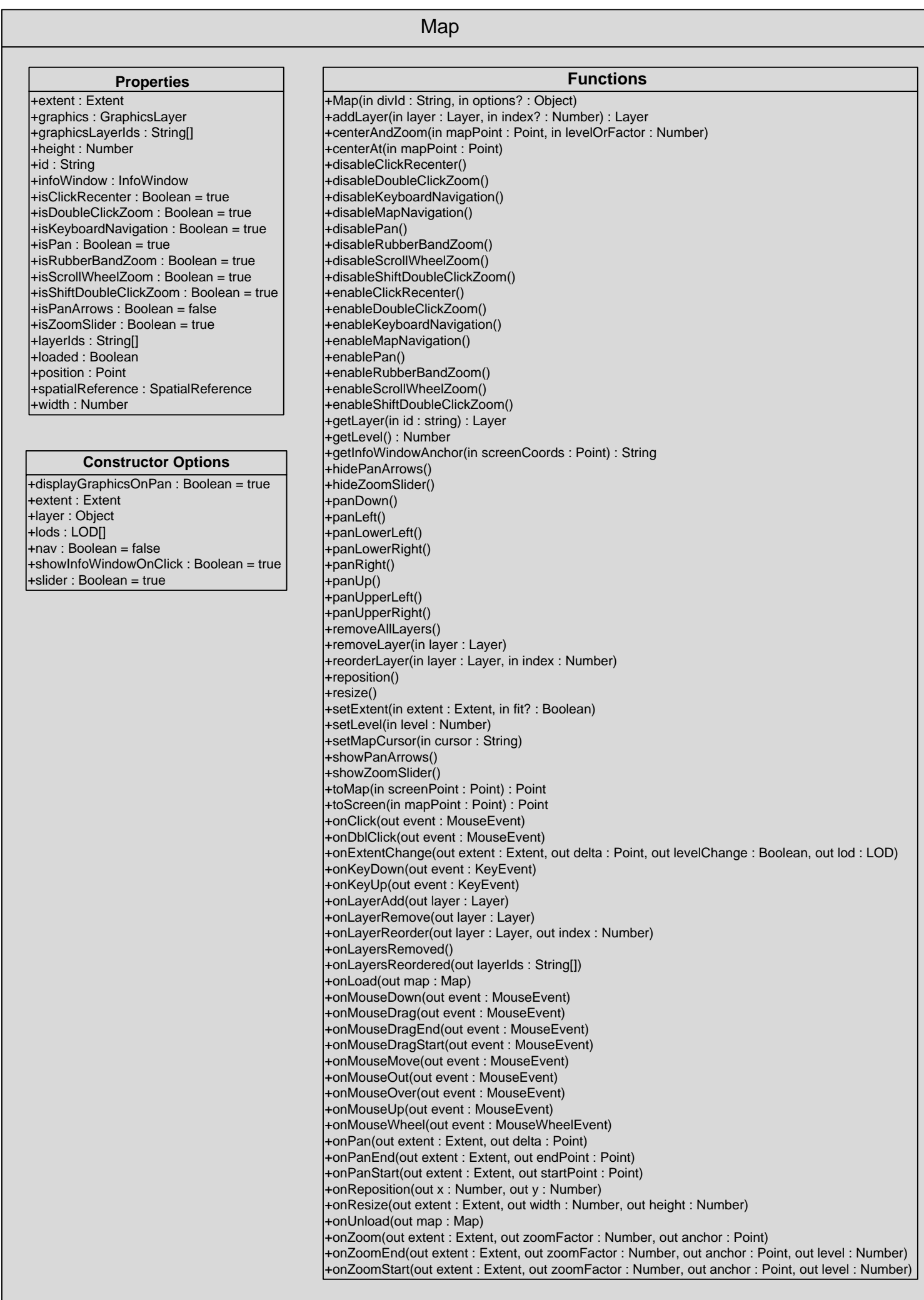
# ArcGIS JavaScript API

Version 1.5

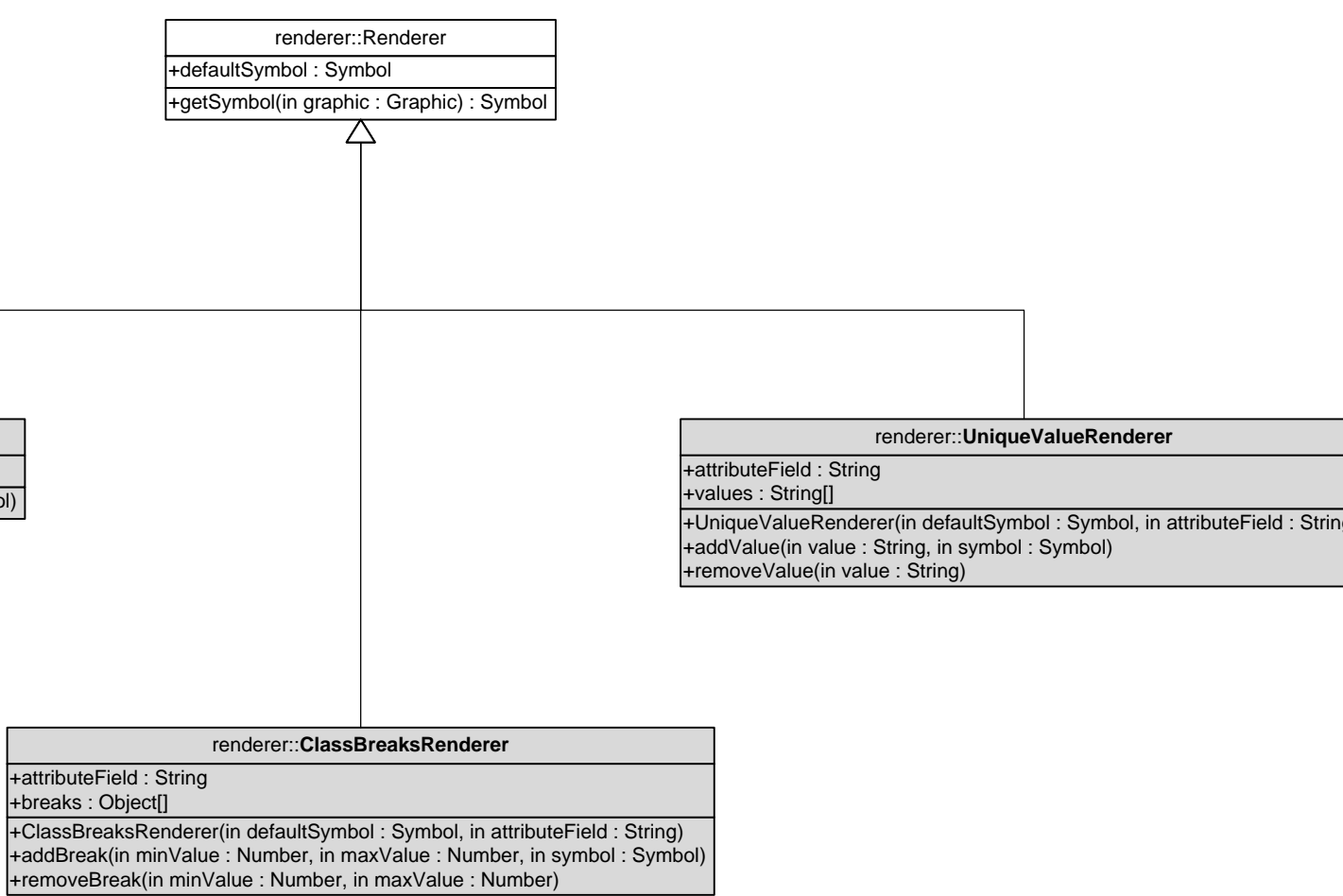
<http://serverapi.arcgisonline.com/jsapi/arcgis/?v=1.5>

(Page 1

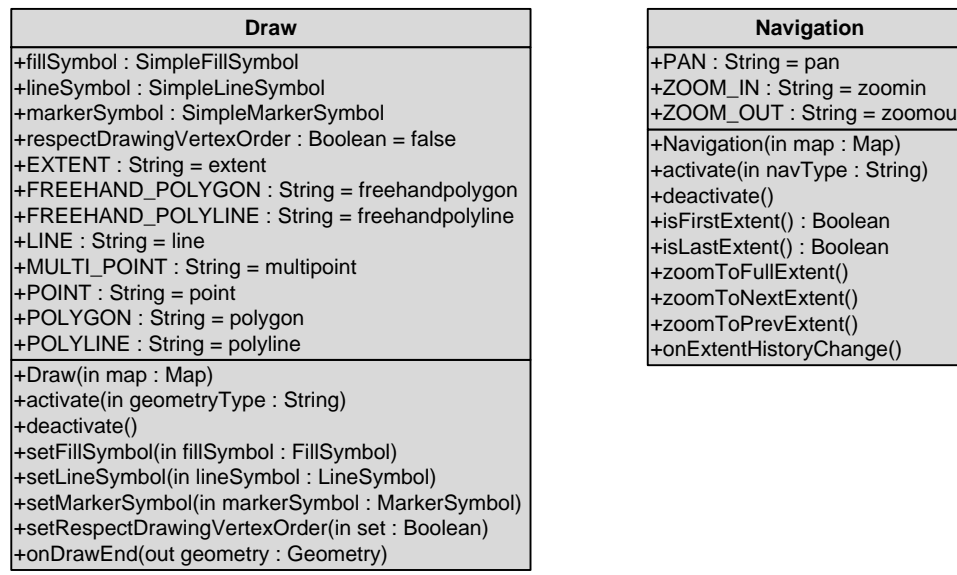
## Map



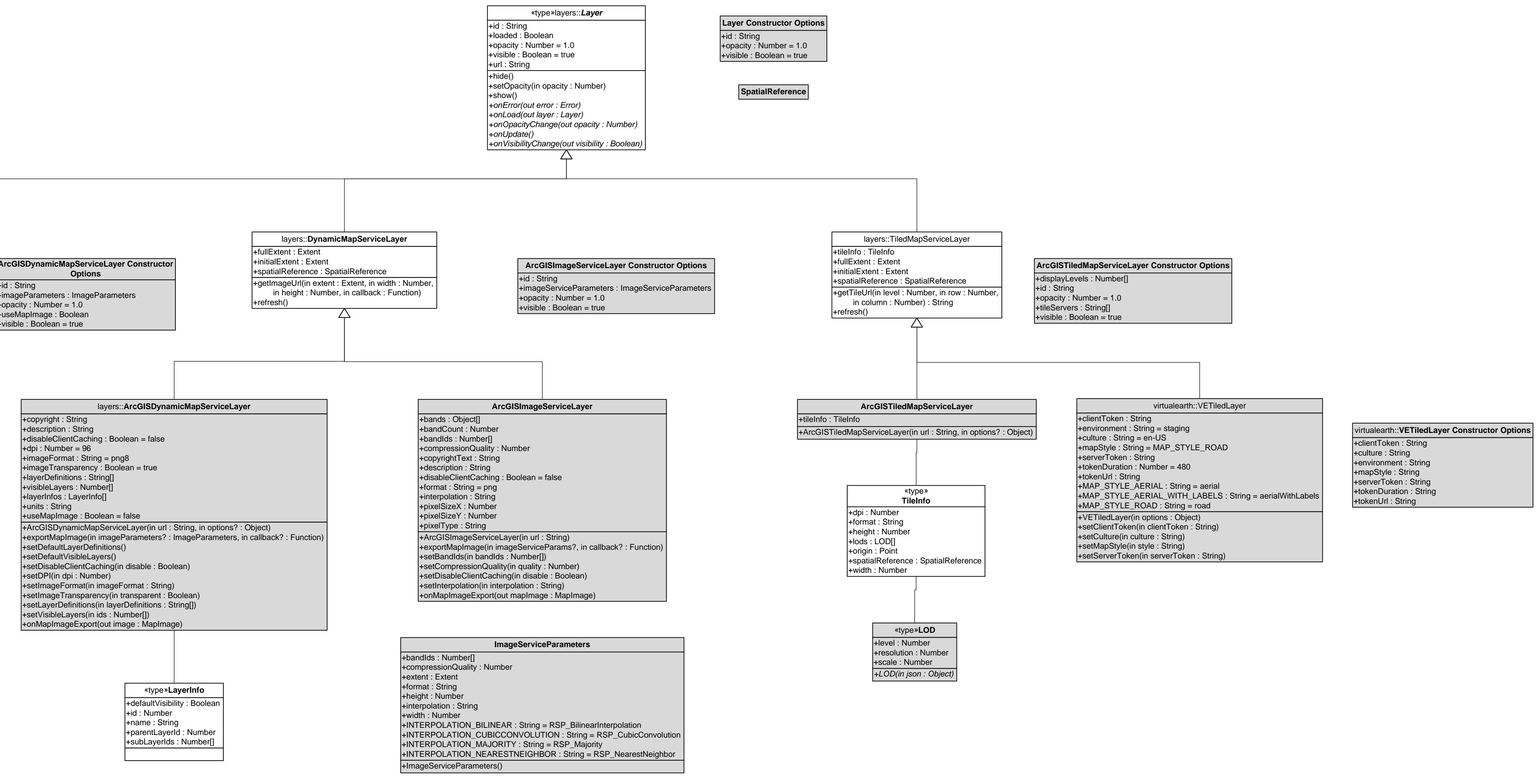
## Render



## Toolbar



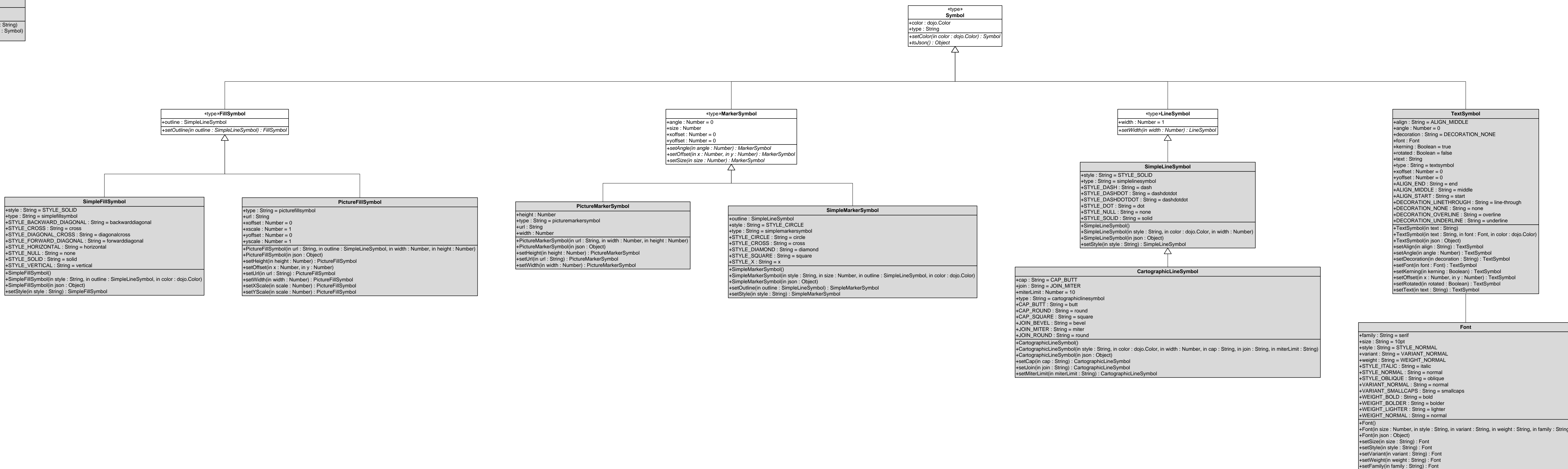
## Layer



## Dijit



## Symbol



# ArcGIS JavaScript API

Version 1.5

<http://serverapi.arcgisonline.com/jsapi/arcgis/?v=1.5>

(Page 2)

## Geometry Service

[illegible][illegible]

## Identify task

```

IdentifyParameters
+dpi : Number = 96
+geometry : Geometry
+height : Number
+layersId : Number[]
+layerOption : String = LAYER_OPTION_TOP
+mapExtent : Extent
+returnGeometry : Boolean = false
+spatialReference : SpatialReference
+tolerance : Number
+width : Number
+LAYER_OPTION_ALL : String = all
+LAYER_OPTION_TOP : String = top
+LAYER_OPTION_VISIBLE : String = visible
+IdentifyParameters()

```

## Find task

FindParameters
+contains: Boolean = true
+layerIds: Number[]
+outSpatialReference: SpatialReference
+returnGeometry: Boolean = false
+searchFields: String[]
+searchText: String
+FindParameters()

tasks::FindTask
+url : String
+FindTask(in url : String)
+execute(in findParameters : FindParameters, in callback? : Function, in errorCallback? : Function) : dojo.Deferred
+onComplete(out findResults : FindResult[])
+onError(out error : Error)

FindResult
+displayFieldName : String
+feature : Graphic
+foundFieldName : String
+layerId : Number
+layerName : String

## Query task

```

tasks::IdentifyTask
ng
Task(in url : String)
(in identifyParameters : IdentifyParameters, in callback? : Function, in errorCallback? : Function) : dojo.Deferred
  complete(out identifyResults : IdentifyResult[])
  (out error : Error)

```

IdentifyResult
+displayName : String
+feature : Graphic
+layerId : Number
+layerName : String

## Geoprocessor

[illegible]

## Virtual Earth Geocoding

```

esri.geometry

4from ison (in json : Object) : Geometry
5    * geographicToWebMercator() : Geometry
6    * geoToUtm (in projName : Geometry) : String
7    * getLength (in point : Point, in point2 : Point) : Number
8    * getLineIntersect (in line1 start : Point, in line1 end : Point, in line2 start : Point, in line2 end : Point) : Point
9    * isArea (in mapGeometry : Number, in mapGeometry2 : Number) : Boolean
10   * isClockwise (in ring : Array) : Boolean
11   * MapGeometry (in width : Number, in height : Number, in screenGeometry : Geometry) : Geometry
12   * MapPointInExtent (Extent : Extent, in width : Number, in height : Number, in screenPoint : Point) : Point
13   * MapScreenGeometry (in extent : Extent, in width : Number, in height : Number, in mapGeometry : Geometry) : Geometry
14   * MapScreenPointInExtent (Extent : Extent, in width : Number, in height : Number, in mapGeometry : Point) : Point
15   * toWebMercator (in geometry : Geometry) : Geometry

```

<b>esri.symbol</b>
<b>+fromJson(in json : Object) : Symbol</b>

```
virtualearth:VEGeocoder Constructor Options
+culture : String
+environment : String
+serverToken : String
+tokenDuration : String
+tokenUri : String
```

```
virtualearth:VEGeocodeResult
+address : VEAddress
+bestView : Extent
+calculationMethod : String
+confidence : String
+displayName : String
+entityType : String
+location : Point
+matchCodes : String
```

```

virtualEarth:VEAddress
addressLine : String
adminDistrict : String
countryRegion : String
district : String
formattedAddress : String
locality : String
postalCode : String
postalTown : String

```

## Route task

tasks: <b>RouteTask</b>
+url: String
+RouteTask(in url: String)
+solve(in params: RouteParameters, in callback?: Function, in errback?: Function): dojo.Deferred
+onError(out error: Error)
+onSolveComplete(out routeResults: RouteResult[], out barriers: Graphic[], out messages: GPMessages[])

tasks::FeatureSet

```

s: DirectionsFeatureSet
ent : Extent
argedGeometry : Polyline
eld : String
teName : String
allLength : Number
alTime : Number
alDriveTime : Number

```

tasks::FeatureSet
+displayName: String
+features: Graphic[]
+fieldAliases: Object
+geometryType: String
+spatialReference: SpatialReference
+FeatureSet()
+FeatureSet(in json: Object)

## Locator

```

tasks:Locator
url: String
+Locator(url: String)
+Locator(url: String, address: Object, in ourFields?: String[], in callback?: Function, in errorCallback?: Function): dojo.Deferred
+Location(address: String, location: Point, in distance: Number, in callback?: Function, in errorCallback?: Function): dojo.Deferred
+onAddressToLocationsComplete(out addressCandidates: AddressCandidate[])
+onError(out error: Error)
+onLocationToAddressComplete(out addressCandidate: AddressCandidate)

```

AddressCandidate
+address : Object
+attributes : Object
+location : Point
+score : Number

DataFile
+url : String
+DataFile()

LinearUnit
+distance : Number
+units : String
+LinearUnit()

RasterData
+format : String
+url : String
+RasterData()

Date
+date : Date
+format : String = EEE MMM dd HH:mm:ss zzz yyyy
+Date()

```
+type = JobInfo
+jobId : String
+jobStatus : String
+messages : GPMessage[]
+STATUS_CANCELLED : String = esiJobCancelled
+STATUS_CANCELLING : String = esiJobCancelling
+STATUS_DELETED : String = esiJobDeleted
+STATUS_DELETING : String = esiJobDeleting
+STATUS_EXECUTING : String = esiJobExecuting
+STATUS_FAILED : String = esiJobFailed
+STATUS_NEW : String = esiJobNew
+STATUS_SUBMITTED : String = esiJobSubmitted
+STATUS_SUCCEEDED : String = esiJobSucceeded
+STATUS_TIMED_OUT : String = esiJobTimedOut
+STATUS_WAITING : String = esiJobWaiting
```

«type» <b>GPMMessage</b>
+description : String +type : String
+type : String

ParameterValue
+dataType : String
+paramName : String
+value : Object

# Geometry

«type» geometry: <b>Geometry</b> +spatialReference: SpatialReference +type: String +setSpatialReference(in sr: SpatialReference) +toJson(): Object +getExtent(): Extent
---

SpatialReference
+wkid : Number
+SpatialReference(in json : Object)

Extent	
xmin : Number	
ymax : Number	
xmax : Number	
ymax : Number	
type : String = extent	
extent(xmin : Number, in ymin : Number, in xmax : Number, in ymax : Number, in spatialReference : SpatialReference)	
Extent(in point : Object)	
extentMin(point : Point) : Extent	
extentMax(point : Point) : Boolean	
extent(in factor : Number) : Extent	
getCenter() : Point	
getHeight() : Number	
getWidth() : Number	
intersect(in extent : Extent) : Extent	
isIn(x : Number, in y : Number) : Extent	
isWithin(in extent : Extent) : Extent	
update(in xmin : Number, in ymin : Number, in xmax : Number, in ymax : Number, in spatialReference : SpatialReference)	

geometry: <b>Multipoint</b>
+points : Number[]
+type : String = multipoint
+Multipoint(in spatialReference : SpatialReference)
+Multipoint(in json : Object)
+addPoint(in point : Point)
+addPoint(in point : Number[])
+addPoint(in json : Object)
+removePoint(in index : Number) : Point

Point
+x : Number
+y : Number
+type : String = point
+Point(x : Number, in y : Number, in spatialReference : SpatialReference)
+Point(x : y) : Array, in spatialReference : SpatialReference)
+Point(json : Object)
+offset(in x : Number, in y : Number) : Point
+setX(in x : Number)
+setY(in y : Number)

```

geometry:Polygon
+@rings: Number[]
+@type: String = polygon
+@Polygon(in SpatialReference: SpatialReference)
+@Polygon(in json : Object)
+@addRing(in points : Point[])
+@addRing(in ring : Number[])
+@addRing(in point:json : Object[])
+@contains(in point: Point): Boolean
+@getPoint(in ringIndex: Number, in pointIndex: Number): Point
+@removeRing(in ringIndex: Number): Point[]

```

```

geometry: Polyline
+paths: Number[][][]
+type: String = polyline
+Polyline(in spatialReference: SpatialReference)
+Polyline(in json: Object)
+addPath(in points: Point[])
+addPath(in path: Number[][][])
+addPath(in pointJson: Object[])
+getPoint(in pathIndex: Number, in pointIndex: Number): Point
+getPaths(in pathIndex: Number): Number[][][]

```