# iConnectorOpenLayers3

## 1产品概述

iConnectorOpenLayers3 是一款基于SuperMap iClient for JavaScript 和第三方地图JavaScript开发的连接器工具，此处面对的是以OpenLayers3 为基础，并且又想加入SuperMap iServer强大的功能的用户。

OpenLayers 3是一个专为Web GIS [客户端](http://baike.baidu.com/view/930.htm)开发提供的JavaScript 类库包，用于实现标准格式发布的地图数据访问。由于缺少SuperMap iServer的核心GIS功能，iConnectorOpenLayers3就是以简单的方法去衔接SuperMap iServer的核心GIS功能，将其和OpenLayers3 相结合，达到兼具强大的GIS功能以及易用性。

iConnectorOpenLayers3 主要提供了地图叠加以及Geometry的转换，可以实现使用OpenLayers3的API出的地图上面叠加SuperMap iServer中合理的地图、专题图的叠加、查询功能以及分析功能。

## 2快速入门

### 2.1申请SuperMap许可

既然我们需要用到SuperMap的功能，当然需要申请SuperMap的许可，请在SupeMap的主页<http://www.supermap.com.cn/>的下载区<http://support.supermap.com.cn/ProductCenter/DownloadCenter/ProductPlatform.aspx>下载许可配置工具以及申请许可。

### 2.2启动服务器

许可安装完毕后我们还的在下载区下载服务器产品SuperMap iServer Java ，并且安装，然后启动服务器，服务器会自带一些基础的服务，我们的示例都是以这些服务为基础的。

### 2.3创建项目

我们的项目很简单，就是普通的html，推荐大家使用WebStorm来开发，比较快捷。

首先你需要在SuperMap iServer Java 的安装文件iClient\forJavaScript\libs下拷贝一份SuperMap.js文件到你的文件夹下，此js文件是SuperMap iClient for JavaScript的产品包，详细信息可以在你本地

<http://localhost:8090/iserver/iClient/forJavaScript/index.html>下了解或官网<http://support.supermap.com.cn:8090/iserver/iClient/forJavaScript/index.html>下，其次我们还需要连接器文件，也就是我们的iConnector OpenLayers3.js文件。

首先我们创建一个html页面，需要添加四个引用：

<script src="http://openlayers.org/en/v3.10.1/build/ol.js" type="text/javascript"></script> 此为引用OpenLayers3的 API；

<link rel="stylesheet" href="http://openlayers.org/en/v3.10.1/css/ol.css" type="text/css">此为引用OpenLayers3的css;

<script type="text/javascript" src="SuperMap.js"></script>此为引用的SuperMap.js文件；

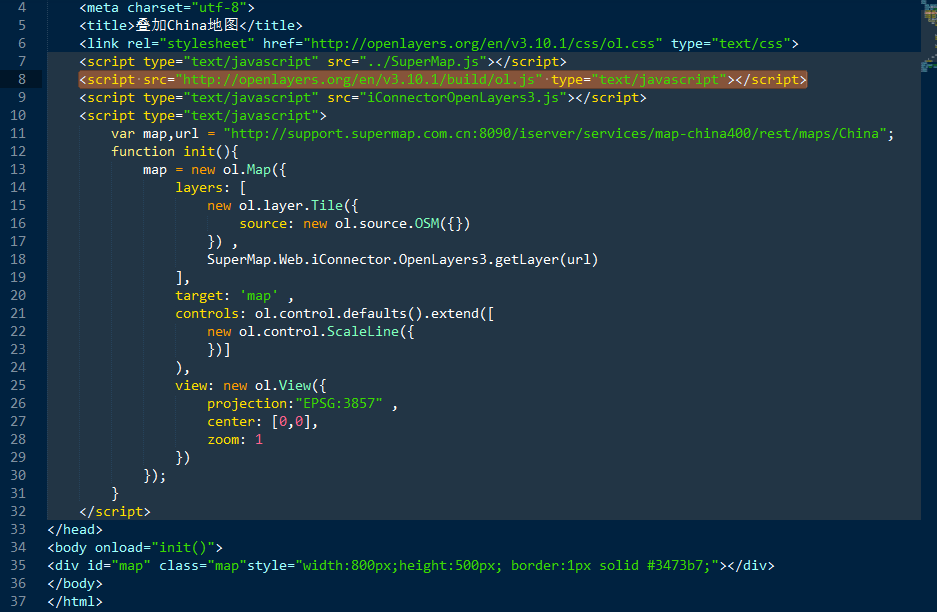
<script type="text/javascript" src="iConnectorOpenLayers3.js"></script>此为我们的连接器。注意连接器是基于前两个扩展的，所以要最后引用。

用过OpenLayers3的 API的用户都知道我们需要一个div来承载地图，所以在body里面创建一个<div id=" map " style="width: 800px;height: 400px"></div>，注意需要赋予id。OpenLayers3支持经纬度（3857）、4326等投影，我们叠加的图层可以是经纬度（3857）和4326两种，这里范例使用经纬度的

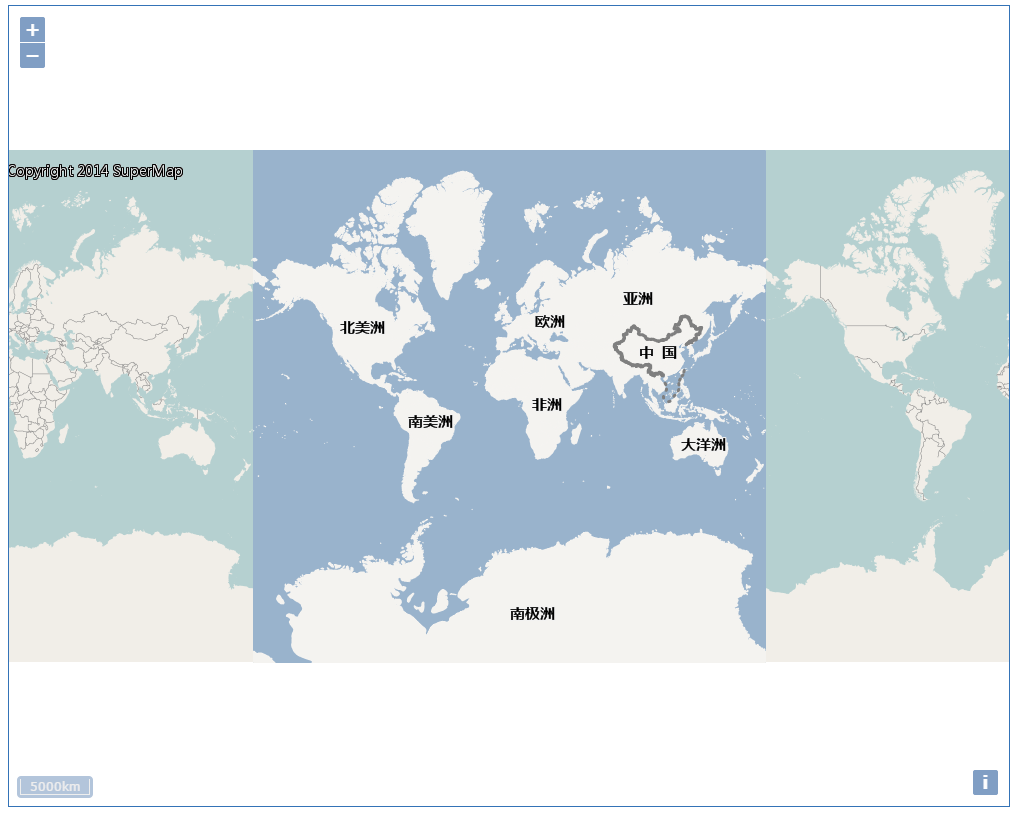
添加核心的脚本：

|  |
| --- |
| var map,layer,  url= "http://support.supermap.com.cn:8090/iserver/services/map-china400/rest/maps/China";  map =new ol.Map({ layers: [  new ol.layer.Tile({  source: new ol.source.OSM({  }) }) ,  SuperMap.Web.iConnector.OpenLayers3.getLayer(url)  ],  target: 'map' ,  controls: ol.control.defaults().extend([  new ol.control.ScaleLine({  })]  ),  view: new ol.View({  projection:"EPSG:3857" ,  center: [0,0],  zoom: 1  })  }) |

如下为完整的代码截图：



然后运行网页，可以看到使用OpenLayers3的API出China的地图：



## 3API

整个iConnectorOpenLayers3所有接口都以静态方法开设，共开设了如下几个接口：

### 3.1地图叠加

地图叠加接口为SuperMap.Web. iConnector. OpenLayers3.getLayer(url,options);此方法第一个参数url为其他地图服务的基础地址，必设参数，此地图现支持3857和4326投影系；第二个参数options为可选参数，options为一个object，如：options={transparent:true; cacheEnabled:false}，里面的参数包括：

1. transparent：{Boolean}设置每一张图片是否背景透明，默认为true
2. cacheEnabled：{Boolean}设置访问服务器图片的时候是否使用缓存，默认为false
3. layersID：{String}当需要将服务器出的专题图叠加到其他地图上时需要将专题的id号设置到这个属性上，这样在访问图片的时候才能正确的找到专题图的图片

此接口最终返回一个OpenLayers3的图层扩展类型的对象。

### 3.2Geometry转换

几何图形的转换现在只支持三种：点、折线和多边形。由于地图的特殊性，几何图形其实最终都是分解为点后再转换成OpenLayers3的坐标，再生成OpenLayers3的几何对象，所以理论上我们支持大部分常用投影系的几何图形的转换。

#### 3.2.1点

点的转换接口为SuperMap.Web. iConnector. OpenLayers3.transferPoint (array,projection);需要将点坐标转换成为4326的标准点再生成OpenLayers3的点。

此方法的第一个参数array为点的数组，不限制数量，array里面支持的点的形式有四种：

1. 自定义对象，必须有x,y，

|  |
| --- |
| var points = [  {x:116.1,y:38.9},  {x:114.1,y:34.1}  ]; |

1. SuperMap.Geometry.Point对象，如：

|  |
| --- |
| var points = [  new SuperMap.Geometry.Point(116.1,38.9),  new SuperMap.Geometry.Point(116.1,38.9)  ]; |

1. SuperMap.LonLat对象，如：

|  |
| --- |
| var points = [  new SuperMap.LonLat(116.1,38.9),  new SuperMap.LonLat(116.1,38.4)]; |

1. ol.geom.Point对象，如：

|  |
| --- |
| var points = [  new ol.geom.Point([116.3 ,39.9]),  new ol.geom.Point([116.38,39.6])  ] |

第二个参数projection为SuperMap.Projection对象。代表第一个参数里面的点是什么坐标系下的，这样内部才方便进行转换。

最后返回ol.geom.Point对象的数组。

#### 3.2.2折线

折线的转换接口为SuperMap.Web.iConnector.OpenLayers.transferLine (array,projection);此方法同样支持任意多条折线的转换。

第一个参数array也是数组，不过是折线数组，里面的折线支持两种形式：

1. SuperMap.Geometry.LineString的对象，如：

|  |
| --- |
| var lines = [ new SuperMap.Geometry.LineString(  new SuperMap.Geometry.Point(116.1,38.9),  new SuperMap.Geometry.Point(116.1,38.9)  )]; |

1. ol.geom.LineString的对象，如：

|  |
| --- |
| var lines = [ new ol.geom.LineString([116.1,38.9],[0,0] )]; |

第二个参数projection为SuperMap.Projection对象。代表第一个参数里面的线是什么坐标系下的。最后返回ol.geom.LineString对象的数组

#### 3.2.3多边形

多边形的转换接口为SuperMap.Web. iConnector. OpenLayers3.transferPolygon (array,projection);多边形的转换最终也是分解为点转换完成后再组合成为多边形。

第一个参数array也是数组，这里是多边形数组，里面的多边形支持两种形式：

1. SuperMap.Geometry.Polygon的对象，如：

|  |
| --- |
| var polygons = [new SuperMap.Geometry.Polygon(  [new SuperMap.Geometry.LinearRing(  new SuperMap.Geometry.Point(116.37,39.90),  new SuperMap.Geometry.Point(116.38,39.90),  new SuperMap.Geometry.Point(116.38,39.89),  new SuperMap.Geometry.Point(116.37,39.89)  )  ]  )]; |

1. ol.geom.Polygon的对象，如：

|  |
| --- |
| var polygons = [ new ol.geom.Polygon([  [116.378,39.907],[116.386,39.907],[116.385,39.897]])  ]; |

第二个参数projection为SuperMap.Projection对象。代表第一个参数里面的多边形是什么坐标系下的。最后返回ol.geom.Polygon对象的数组。