

Cesium 概述 (一)



原创 2016年07月09日 10:59:59

标签: javascript (http://so.csdn.net/so/search/s.do?q=javascript&t=blog) /

Cesium (http://so.csdn.net/so/search/s.do?q=Cesium&t=blog) /

三维 (http://so.csdn.net/so/search/s.do?q=三维&t=blog) /

地图 (http://so.csdn.net/so/search/s.do?q=地图&t=blog) /

开源框架 (http://so.csdn.net/so/search/s.do?q=开源框架&t=blog)

14389

Cesium是一个基于JavaScript的开源框架，可用于在浏览器中绘制3D的地球，并在其上绘制地图（支持多种格式的瓦片服务），该框架不需要任何插件支持，但是浏览器必须支持WebGL。

Cesium支持多种数据可视化方式，可以绘制各种几何图形、导入图片，甚至3D模型。同时，Cesium还支持基于时间轴的动态数据展示，例如，我们可以用它绘制卫星运行轨迹。

Cesium HelloWorld

```
[html]
1. <html lang="en">
2. <head>
3. <meta charset="utf-8">
4. <meta http-equiv="X-UA-Compatible" content="IE=Edge,chrome=1">
5. <meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1, minimum-scale=1, user-scalable=no">
6. <title>Cesium Example</title>
7. <script src="Cesium-1.7.1/Build/CesiumUnminified/Cesium.js"></script>
8. <link rel="stylesheet" type="text/css" href="Cesium-1.7.1/Build/CesiumUnminified/Widgets/widgets.css">
9. <style>
10. html,body,#cesiumContainer {
11.     width: 100%;
12.     height: 100%;
13.     margin: 0;
14.     padding: 0;
15.     overflow: hidden;
16. }
17. </style>
18. </head>
19. <body>
20.     <div id="cesiumContainer"></div>
21.     <script type="text/javascript" src="index.js"></script>
22. </body>
23. </html>
```

index.js

```
[javascript]
1. var viewer = new Cesium.Viewer( 'cesiumContainer', {
2.     animation : false, //是否创建动画小器件，左下角仪表
3.     fullscreenButton : false, //是否显示全屏按钮
4. }
```

加入CSDN，享受更精准的内容推荐，与500万程序员共同成长！



GISEarth (http://blog.c...

+ 关注

(http://blog.csdn.net/happyduoduo1/article/details/78733111)

原创 79 粉丝 280 喜欢 1 码云 (https://g... utm_sour

他的最新文章

更多文章

(http://blog.csdn.net/happyduoduo1)

开源GIS介绍（openlayer,leaflet,map box,cesium）(http://blog.csdn.net/happyduoduo1/article/details/78733111)

基于3dtiles做的视域分析 (http://blog.csdn.net/happyduoduo1/article/details/72921643)

cesium模拟水系效果 (http://blog.csdn.net/happyduoduo1/article/details/62049895)

博主在北京西站地铁上被偷了一块移动硬盘 (http://blog.csdn.net/happyduoduo1/article/details/61203194)

F4map (http://blog.csdn.net/happyduoduo1/article/details/58112441)

相关推荐

cesium的学习 (http://blog.csdn.net/naqidada/article/details/41118575)

Cesium 概述 (二) 空间数据可视化 (http://blog.csdn.net/happyduoduo1/article/details/51865811)

cesiumjs学习笔记之二——空间数据可视化 (http://blog.csdn.net/Prepared/article/details/68940839)

Cesium中加载动态水 (http://blog.csdn.net/XLSMN/article/details/78752669)

登录

注册



```

5. geocoder : false, //是否显示geocoder小器件, 右上角查询按钮
6. homeButton : false, //是否显示Home按钮
7. infoBox : false, //是否显示信息框
8. sceneModePicker : false, //是否显示3D/2D选择器
9. selectionIndicator : false, //是否显示选取指示器组件
10. timeline : false, //是否显示时间轴
11. navigationHelpButton : false, //是否显示右上角的帮助按钮
12. scene3DOnly : true, //如果设置为true, 则所有几何图形以3D模式绘制以节约GPU资源
13. clock : new Cesium.Clock(), //用于控制当前时间的时钟对象
14. selectedImageryProviderViewModel : undefined, //当前图像图层的显示模型, 仅baseLayerPicker设为true有意义
15. imageryProviderViewModels : Cesium.createDefaultImageryProviderViewModels(), //可供BaseLayerPicker选择的图像图层ProviderViewModel数组
16. selectedTerrainProviderViewModel : undefined, //当前地形图层的显示模型, 仅baseLayerPicker设为true有意义
17. terrainProviderViewModels : Cesium.createDefaultTerrainProviderViewModels(), //可供BaseLayerPicker选择的地形图层ProviderViewModel数组
18. imageryProvider : new Cesium.OpenStreetMapImageryProvider( {
19.   credit : '',
20.   url : '192.168.0.89:5539/planet-satellite/'
21. } ), //图像图层提供者, 仅baseLayerPicker设为false有意义
22. terrainProvider : new Cesium.EllipsoidTerrainProvider(), //地形图层提供者, 仅baseLayerPicker设为false有意义
23. skyBox : new Cesium.SkyBox({
24.   sources : {
25.     positiveX : 'Cesium-1.7.1/Skybox/px.jpg',
26.     negativeX : 'Cesium-1.7.1/Skybox/mx.jpg',
27.     positiveY : 'Cesium-1.7.1/Skybox/py.jpg',
28.     negativeY : 'Cesium-1.7.1/Skybox/my.jpg',
29.     positiveZ : 'Cesium-1.7.1/Skybox/pz.jpg',
30.     negativeZ : 'Cesium-1.7.1/Skybox/mz.jpg'
31.   }
32. } ), //用于渲染星空的SkyBox对象
33. fullscreenElement : document.body, //全屏时渲染的HTML元素,
34. useDefaultRenderLoop : true, //如果需要控制渲染循环, 则设为true
35. targetFrameRate : undefined, //使用默认render loop时的帧率
36. showRenderLoopErrors : false, //如果设为true, 将在一个HTML面板中显示错误信息
37. automaticallyTrackDataSourceClocks : true, //自动追踪最近添加的数据源的时钟设置
38. contextOptions : undefined, //传递给Scene对象的上下文参数 (scene.options)
39. sceneMode : Cesium.SceneMode.SCENE3D, //初始场景模式
40. mapProjection : new Cesium.WebMercatorProjection(), //地图投影体系
41. dataSources : new Cesium.DataSourceCollection()
42. //需要进行可视化的数据源的集合
43. } );
44. var scene = viewer.scene;
45. var canvas = viewer.canvas;
46. var clock = viewer.clock;
47. var camera = viewer.scene.camera;
48. var entities = viewer.entities;

```

可以加快时间的运行, 并且模拟日光照射效果:

[javascript]

```

1. //加快时钟的运行
2. clock.multiplier = 0.1 * 60 * 60;
3. //阳光照射区域高亮
4. scene.globe.enableLighting = true;

```

通过以下代码, 可以设置镜头位置与指向, Cesium的Camera对象提供了多种操控镜头的方法:
index.js

[javascript]

```

1. //设置镜头位置与方向
2. camera.setView( {
3.   //镜头的经纬度、高度。镜头默认情况下, 在指定经纬高度俯视 (pitch=-90) 地球
4.   position : Cesium.Cartesian3.fromDegrees( 116.3, 39.9, 100000000 ), //北京100000公里上空
5.   //下面的几个方向正好反映默认值
6.   heading : Cesium.Math.toRadians( 0 ),
7.   pitch : Cesium.Math.toRadians( -90 ),
8.   roll : Cesium.Math.toRadians( 0 )
9. } );
10. //让镜头飞行 (动画) 到某个地点和方向
11. setTimeout( function()
12. {
13.   camera.flyTo( {
14.     position : Cesium.Cartesian3.fromDegrees( 116, 15, 6000000 ),

```

加入CSDN, 享受更精准的内容推荐, 与500万程序员共同成长!



本田新车上市



学咖啡拉花

三室两厅装修



咬肌肥大

OA办公系统

他的热门文章

Cesium 概述 (一) (<http://blog.csdn.net/happyduoduo1/article/details/51865758>)

14382

Cesium 概述 (二) 空间数据可视化 (<http://blog.csdn.net/happyduoduo1/article/details/51865811>)

12863

开源GIS浅谈 (<http://blog.csdn.net/happyduoduo1/article/details/51773850>)

11163

GitHub完整使用教程 (<http://blog.csdn.net/happyduoduo1/article/details/51798678>)

8641

SVG基础以及使用Javascript DOM操作SVG (<http://blog.csdn.net/happyduoduo1/article/details/51789552>)

8111

登录

注册 X

```

15.         orientation : {
16.             heading : Cesium.Math.toRadians( -15 ),
17.             pitch : Cesium.Math.toRadians( -65 ),
18.             roll : Cesium.Math.toRadians( 0 )
19.         },
20.         duration : 3, //动画持续时间
21.         complete : function() //飞行完毕后执行的动作
22.         {
23.             addEntities();
24.         }
25.     } );
26. }, 1000 );
27.
28. //监听键盘事件, 用于平移或者旋转镜头
29. var ellipsoid = scene.globe.ellipsoid;
30. canvas.onclick = function()
31. {
32.     canvas.focus();
33. };
34. var flags = {
35.     looking : false,
36.     rotateLeft : false,
37.     rotateRight : false,
38.     moveUp : false,
39.     moveDown : false,
40.     moveLeft : false,
41.     moveRight : false
42. };
43. var handler = new Cesium.ScreenSpaceEventHandler( canvas );
44. function getFlagForKeyCode( keyCode )
45. {
46.     switch ( keyCode )
47.     {
48.         case 'W'.charCodeAt( 0 ) : //向下平移镜头
49.             return 'moveDown';
50.         case 'S'.charCodeAt( 0 ) : //向上平移镜头
51.             return 'moveUp';
52.         case 'A'.charCodeAt( 0 ) : //向右平移镜头
53.             return 'moveRight';
54.         case 'D'.charCodeAt( 0 ) : //向左平移镜头
55.             return 'moveLeft';
56.         case 'Q'.charCodeAt( 0 ) : //向右旋转镜头
57.             return 'rotateRight';
58.         case 'E'.charCodeAt( 0 ) : //向左旋转镜头
59.             return 'rotateLeft';
60.         default :
61.             return undefined;
62.     }
63. }
64. document.addEventListener( 'keydown', function( e )
65. {
66.     var flagName = getFlagForKeyCode( e.keyCode );
67.     if ( typeof flagName !== 'undefined' )
68.     {
69.         flags[flagName] = true;
70.     }
71. }, false );
72. document.addEventListener( 'keyup', function( e )
73. {
74.     var flagName = getFlagForKeyCode( e.keyCode );
75.     if ( typeof flagName !== 'undefined' )
76.     {
77.         flags[flagName] = false;
78.     }
79. }, false );
80. viewer.clock.onTick.addEventListener( function( clock )
81. {
82.     var cameraHeight = ellipsoid.cartesianToCartographic( camera.position ).height;
83.     var moveRate = cameraHeight / 100.0;
84.
85.     if ( flags.rotateLeft )
86.     {
87.         camera.rotateLeft( 0.01 );
88.     }
89.     if ( flags.rotateRight )
90.     {
91.         camera.rotateRight( 0.01 );
92.     }
93.     if ( flags.moveUp )
94.     {

```

```

95.         camera.moveUp( moveRate );
96.     }
97.     if ( flags.moveDown )
98.     {
99.         camera.moveDown( moveRate );
100.    }
101.    if ( flags.moveLeft )
102.    {
103.        camera.moveLeft( moveRate );
104.    }
105.    if ( flags.moveRight )
106.    {
107.        camera.moveRight( moveRate );
108.    }
109. } );

```

可以添加若干实体，实体可以用于组织多个可视化对象，下面的例子模拟了卫星波束的覆盖范围：
index.js

[javascript]

```

1.  /**
2.   * 根据偏移量计算目标点经纬度
3.   * @param {} start 起始点经纬度数组，单位度
4.   * @param {} offset 东北方向的偏移量，单位米
5.   * @param {} 目标点经纬度数组，单位度
6.   */
7.  function offsetToLongLat( start, offset )
8.  {
9.      var er = 6378137;
10.     var lat = parseFloat( start[1] );
11.     var lon = parseFloat( start[0] );
12.     var dn = parseFloat( offset[1] );
13.     var de = parseFloat( offset[0] );
14.
15.     dLat = dn / er;
16.     var pi = Math.PI;
17.     var dLon = de / ( er * Math.cos( pi * lat / 180 ) )
18.     return [
19.         lon + dLon * 180 / pi, lat + dLat * 180 / pi
20.     ];
21. }
22. /**
23.  * 通过绘制三角形模拟卫星光束效果
24.  * @param {} entities 实体集
25.  * @param {} stltPos 卫星三维坐标数组
26.  * @param {} points 地面点
27.  * @param {} color CSS颜色代码，例如#FF0000
28.  */
29.  function lightShinePolygon( entities, stltPos, points, color )
30.  {
31.      for ( var i = 0; i < points.length; i += 2 )
32.      {
33.          var array = [
34.              stltPos[0], stltPos[1], stltPos[2], points[i], points[i + 1], 0
35.          ];
36.          if ( i + 2 == points.length )
37.          {
38.              array.push( points[0] );
39.              array.push( points[1] );
40.          }
41.          else
42.          {
43.              array.push( points[i + 2] );
44.              array.push( points[i + 3] );
45.          }
46.          array.push( 0 );
47.          entities.add( {
48.              polygon : {
49.                  hierarchy : Cesium.Cartesian3.fromDegreesArrayHeights( array ),
50.                  perPositionHeight : true,
51.                  outline : false,
52.                  material : Cesium.Color.fromAlpha( Cesium.Color.fromCssColorString( color ), .1 )
53.              }
54.          } );
55.      }
56.  }
57.
58.  // 添加实体

```

```

59.  */
60.  function addEntities()
61.  {
62.      //卫星一
63.      {
64.          var stltPos = [
65.              110.0, 40.0, 2500000
66.          ];
67.          entities.add( {
68.              position : Cesium.Cartesian3.fromDegrees.apply( this, stltPos ),
69.              billboard : {
70.                  image : 'images/satellite-1.png',
71.                  horizontalOrigin : Cesium.HorizontalOrigin.CENTER,
72.                  verticalOrigin : Cesium.VerticalOrigin.BOTTOM, //垂直方向位置计算基准设为底部, 默认中心
73.                  width : 92,
74.                  height : 36
75.              }
76.          } );
77.          //一个多边形覆盖范围
78.          {
79.              var color = '#FF0000';
80.              //模拟光照效果的若干多边形
81.              var points = [
82.                  100, 48, 110, 40, 115, 40, 120, 43, 120, 55
83.              ];
84.              lightShinePolygon( entities, stltPos, points, color );
85.              //地面多边形
86.              entities.add( {
87.                  polygon : {
88.                      hierarchy : Cesium.Cartesian3.fromDegreesArray( points ),
89.                      outline : true,
90.                      outlineColor : Cesium.Color.fromAlpha( Cesium.Color.fromCssColorString( color ), .4
91.                      material : Cesium.Color.fromAlpha( Cesium.Color.fromCssColorString( color ), .3 )
92.                  }
93.              } );
94.          }
95.
96.          //一个圆形覆盖范围
97.          {
98.              var r = 600000; //半径
99.              var color = '#0000FF';
100.             //圆心
101.             var ecLong = 110.0;
102.             var ecLat = 30.0;
103.             var ec = Cesium.Cartesian3.fromDegrees( ecLong, ecLat, 0 );
104.             //模拟光照效果的若干多边形
105.             var points = [];
106.             for ( var i = 0; i < 360; i += 30 )
107.             {
108.                 var coord = offsetToLongLat( [
109.                     ecLong, ecLat
110.                 ], [
111.                     Math.cos( Math.PI * i / 180 ) * r, Math.sin( Math.PI * i / 180 ) * r
112.                 ] );
113.                 points.push( coord[0] );
114.                 points.push( coord[1] );
115.             }
116.             lightShinePolygon( entities, stltPos, points, color );
117.             //圆
118.             viewer.entities.add( {
119.                 position : ec,
120.                 ellipse : {
121.                     semiMinorAxis : r,
122.                     semiMajorAxis : r,
123.                     height : 0.0,
124.                     outline : true,
125.                     outlineColor : Cesium.Color.fromAlpha( Cesium.Color.fromCssColorString( color ), .4
126.                     material : Cesium.Color.fromAlpha( Cesium.Color.fromCssColorString( color ), .3 )
127.                 }
128.             } );
129.         }
130.     }
131. }

```



10



版权声明：本文为博主原创文章，未经博主允许不得转载。



- juerjuerwer (/juerjuerwer) 2017-09-03 22:59

请问是在什么IDE上开发的呢?

回复

5楼
- xuening_snow (/xuening_snow) 2017-08-31 15:13

请问cesium提供地图拼接功能吗? 官网找了很久也没看到相关的说明。

回复 1条回复

4楼
- fengzhongyela (/fengzhongyela) 2016-10-27 13:13

基于Onyia的实战地理信息系统 (离线地图, 通过基站转经纬度, Quartz深入, 轨迹实战)
百度网盘地址: http://pan.baidu.com/s/1nv6PUB3 密码: yegn

回复

3楼

查看 9 条热评

相关文章推荐

cesium的学习 (<http://blog.csdn.net/nanqidada/article/details/41118575>)

一、学习资料: <http://cesiumjs.org/>

nanqidada (<http://blog.csdn.net/nanqidada>) 2014年11月14日 14:38 6867

Cesium 概述 (二) 空间数据可视化 (<http://blog.csdn.net/happyduoduo1/article/details...>)

空间数据可视化 Cesium提供Entity API来绘制空间数据, 例如点、标记、标签、线、3D模型、形状、立体形状 (volume)。 Entity API简介 Cesium提供两类...

happyduoduo1 (<http://blog.csdn.net/happyduoduo1>) 2016年07月09日 11:07 12867

加入CSDN, 享受更精准的内容推荐, 与500万程序员共同成长!



太任性！学AI的应届学弟怒拒20K Offer，他想要多少钱？

AI改变命运呀！！前段时间在我司联合举办的校招聘会上，一名刚刚毕业的学弟陆续拒绝2份Offer，企业给出18K、23K高薪，学弟拒绝后直接来了一句...

(http://www.baidu.com/cb.php?c=lgF_pyfqHmknjnvPjn0lZ0qnfK9ujYzP1f4PjDs0Aw-5Hc3rHnYnHb0TAq15HfLPWRznjb0T1YzPH03nyNbP1D1Phc4nmvY0AwY5HDdnHf1PHTLnHT0lgF_5y9YIZ0IQzq-uZR8mLPbUB48ugfEIAqspynEmybz5LNYUNq1ULNzmvRqmhkEu1Ds0ZFb5HD30AFV5H00TZcqn0KdpyfqHRLPjnvnfKEpyfqHc4rj6kP0KWpyfqP1cv

cesiumjs学习笔记之二——空间数据可视化 (http://blog.csdn.net/Prepared/article/deta...

cesiumjs学习笔记之二——空间数据 Prepared (http://blog.csdn.net/Prepared) 2017年04月18日 14:31 1547

Cesium中加载动态水 (http://blog.csdn.net/XLSMN/article/details/78752669)

关于Cesium中对全球以及自定义区域动态水的加载 XLSMN (http://blog.csdn.net/XLSMN) 2017年12月08日 16:27 86

cesiumjs开发实践 (七) (http://blog.csdn.net/happyduoduo1/article/details/51942892)

3D模型 摘要 cesium中支持载入3D模型，不过只支持gltf格式。 cesium中支持载入3D模型，不过只支持gltf格式。 gltf是khronos组织（起草Op... happyduoduo1 (http://blog.csdn.net/happyduoduo1) 2016年07月18日 16:09 2623

4



85.00/台

300M无线ap 高通 9531芯片 大功率吸顶

5



145.00/台

厂家直销 吸顶AP 300M 大功率无线AP

6



90.00/台

300M 吸顶AP 大功率 无线ap 认证路由器 网

Cesium (七) 镜头 (http://blog.csdn.net/happyduoduo1/article/details/51868100)

镜头 Cesium提供了以下默认鼠标行为：（1）单击并拖拽球体：旋转地球，镜头俯角不变（2）单击并拖拽空间：滚动rol l、俯仰pitch镜头（3）右击并拖拽、中键滚动：缩放镜... happyduoduo1 (http://blog.csdn.net/happyduoduo1) 2016年07月09日 19:18 3354



Delphi7高级应用开发随书源码 (http://download.csdn.net/download/c...

(http://download 2003年04月30日 00:00 676KB 下载

基于 Cesium 的Web三维地球客户端程序开发框架 (http://blog.csdn.net/muyao520/arti...

MarsGIS for Cesium三维地球框架 是火星科技研发的一个Web三维地图开发平台系统，是火星科技团队成员多年GIS开发和Cesium使用的技术沉淀。基于Cesium和现代Web技术栈全新构... muyao520 (http://blog.csdn.net/muyao520) 2017年11月10日 15:19 249



Delphi7高级应用开发随书源码 (http://download.csdn.net/download/c...

(http://download 2003年04月30日 00:00 676KB 下载

基于Cesium的demo赏析（持续更新） (http://blog.csdn.net/u013929284/article/detai...

更新于2017.1.14Cesium的强大不用多说，所以有很多政府、组织基于cesium做了一些应用，其中不乏有很多优秀的示例，我们大都可以从中获得对自己的项目有益的东西。另：有的网站需要翻墙。 1...

cesiumjs开发实践 (五) (http://blog.csdn.net/happyduoduo1/article/details/51942862)

坐标变换 摘要 cesium中常用的坐标有两种WGS84地理坐标系和笛卡尔空间坐标系。 cesium中常用的坐标有两种WGS84地理坐标系和笛卡尔空间坐标系。我们平时常用的以经纬...

happyduoduo1 (http://blog.csdn.net/happyduoduo1) 2016年07月18日 16:07 2143

angular集成Cesium (http://blog.csdn.net/5hongbing/article/details/78194267)

第一种方案 安装cesium \$ npm install --save cesium 将cesium的assets、script和css配置到angular-cli.json文件 "as...

5hongbing (http://blog.csdn.net/5hongbing) 2017年10月10日 16:35 261

Cesium开源三维地球离线地图发布源码示例功能 (http://blog.csdn.net/mrib/article/deta...

一、概述 Cesium开源三维地球离线地图发布源码提供了地图切换、查询定位、模型加载、专题图叠加显示和测量功能等，旨在为用户提供一个可以在Cesium快速加载离线地图或在线地图的解决方案，并提供技术...

mrib (http://blog.csdn.net/mrib) 2017年10月17日 15:23 561

关于Cesium 加载wms服务 (http://blog.csdn.net/woshirsn/article/details/52575273)

1, 在三维地球上加载wms服务 function startup(Cesium) { widget = new Cesium.CesiumWidget('cesiumContainer'); ...

woshirsn (http://blog.csdn.net/woshirsn) 2016年09月18日 16:04 2194

cesiumjs的一些简单地图应用，主要包含cesium调用arcgis服务 (http://blog.csdn.net/u0...

cesiumjs的一些简单地图应用，主要包含cesium调用arcgis服务

u010001043 (http://blog.csdn.net/u010001043) 2017年07月04日 12:33 1937

Cesium之Primitive相关探索 (http://blog.csdn.net/u013929284/article/details/52749...

Primitive: 它代表着场景（scene）中的几何体，这个几何体可以是一个几何实例，或者由一组几何实例合并而成，即便它们是不同的种类的几何体，例如一个矩形和一个椭圆。 primitive通过把...

u013929284 (http://blog.csdn.net/u013929284) 2016年10月07日 15:46 1517

Cesium学习笔记（五）几何和外观 (http://blog.csdn.net/HobHunter/article/details/75...

之前的笔记中记录了Cesium可以使用实体（如多边形和椭圆）创建不同的地质类型。例如：使用点阵图在地球上创建一个矩形 let view = new Cesium.Viewer('cesiumDemo'...

HobHunter (http://blog.csdn.net/HobHunter) 2017年07月14日 14:39 902

cesium-3d-tile/倾斜摄影加载技术 (http://blog.csdn.net/as320qqw/article/details/60...

关于3D-Tiles还是Beta阶段，其定位是Web环境下海量三维模型数据。其数据的生产目前还没有成熟的工具，不过cesium在去年9月已经开始推进3dtile数据规范的ogc进程，相信很快就会有结果...

as320qqw (http://blog.csdn.net/as320qqw) 2017年03月06日 11:46 4751

Cesium加载天地图及注记，并且去除自带控件 (http://blog.csdn.net/GISuuser/article/d...



Cesium的练手

GISuuser (http://blog.csdn.net/GISuuser) 2017年08月29日 19:29 978

Cesium应用篇：快速搭建 (<http://blog.csdn.net/yangzhai/article/details/60322234>)

(<https://passpoc.com>)

范例中所有范例可以在Github中搜索：ExamplesforCesium Cesium ['si:ziem]是一款开源的JavaScript开源库，开发者通过Cesium，实现无插件的...

 yangzhai (<http://blog.csdn.net/yangzhai>) 2017年03月04日 10:48  1416



10

