

Cesium开发培训

讲解人：火星科技 木遥

目 录

1

开发环境搭建

2

Cesium基础知识

3

数据处理及服务发布

4

MarsGIS平台概览

说明

➤ 教学视频

本PPT教程对应的录制视频：<https://ke.qq.com/course/468292>

➤ 最新版本

本PPT会不定期更新，最新发布地址为：<http://marsgis.cn/doc/study-cesium.pdf>

➤ 问题及意见反馈

如果您有任何建议意见。欢迎发送E-mail邮件至：wh@marsgis.cn

1、开发环境搭建

- Node
- VSCode
- Nginx

1.1 开发环境所需工具

以下工具为后续讲解开发中所涉及到的工具，也是我们推荐的，建议您安装使用。

当然如果您已经非常精通web前端领域，并有其他习惯的工具也可以忽略安装。

安装工具	下载地址	说明
Node	https://nodejs.org/zh-cn/download/	使用包管理器 npm
Visual Studio Code	https://code.visualstudio.com/Download	Web前端代码编辑IDE
Nginx	http://nginx.org/en/download.html	http容器，用于发布marsv和GIS静态数据

1.2 开发工具: Node安装

Node.js 是一个基于 Chrome V8 引擎的 JavaScript 运行环境。

Node.js 使用了一个事件驱动、非阻塞式 I/O 的模型，使其轻量又高效。

Node.js 的包管理器 npm，是全球最大的开源库生态系统。

简单的说 Node.js 就是运行在服务端的 JavaScript。



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [版本 10.0.17134.915]
(c) 2018 Microsoft Corporation。保留所有权利。

C:\Users\dell>node -v
v10.10.0

C:\Users\dell>npm -v
6.4.1

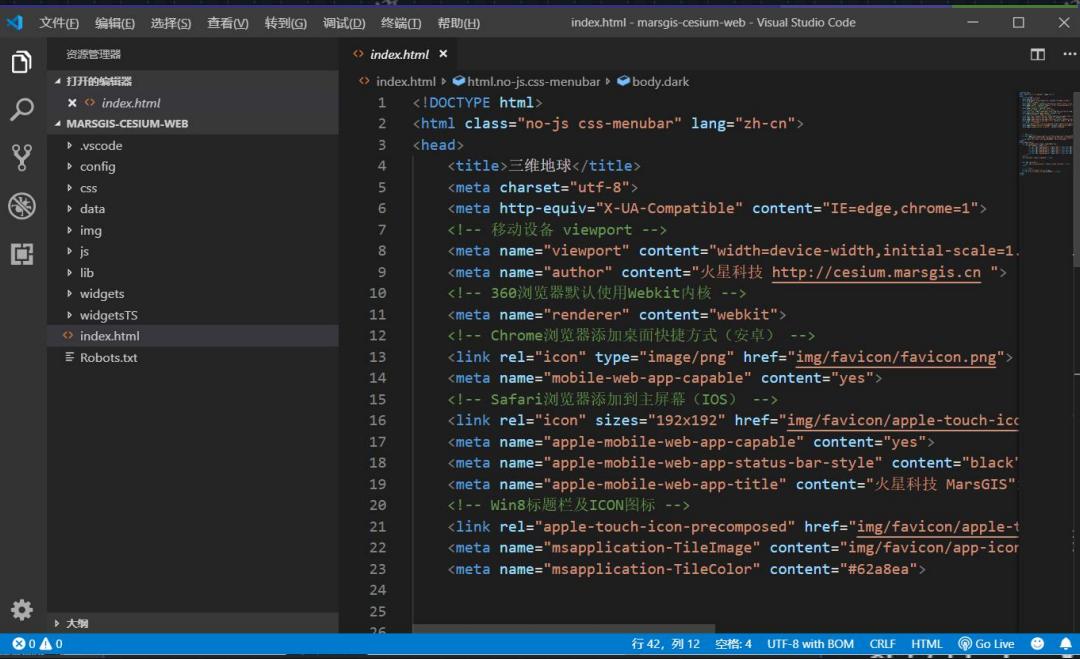
C:\Users\dell>cnpm -v
cnpm@6.0.0 (D:\webdev\npm\node_global\node_modules\cnpm\lib\
npm@6.4.1 (D:\webdev\npm\node_global\node_modules\cnpm\node_
node@10.10.0 (D:\webdev\nodejs\node.exe)
npminstall@3.11.0 (D:\webdev\npm\node_global\node_modules\cn
prefix=D:\webdev\npm\node_global
win32 x64 10.0.17134
registry=https://registry.npm.taobao.org
```

安装步骤

- 根据自己需要下载对应的版本
- 安装时一直单击Next到Finish完成
- 在终端里面输入node -v 检查是否安装成功
- 重启电脑，终端里面输入npm -v检查是否生效
- 在终端输入安装cnpm命令，使用国内镜像
`npm install -g cnpm --registry=https://registry.npm.taobao.org`

1.3 开发工具: VSCode安装

Visual Studio Code (简称 VS Code) 是一款免费开源的现代化轻量级代码编辑器，支持几乎所有主流的开发语言的语法高亮、智能代码补全、自定义热键、括号匹配、代码片段、代码对比 Diff、GIT 等特性，支持插件扩展，并针对Web网页开发和云端应用开发做了优化。



安装步骤

- 根据自己需要下载对应的版本
- 安装时把勾选项全选中，一直单击下一步到完成
- 安装插件：Live Server、Chinese~(汉化包)
- 打开任意web项目后右击html页面，选择Open with Live Server查看运行（方便之处：修改代码后自动刷新页面，无缓存，操作简单方便）

1.4 开发工具: Nginx安装

Nginx是一款轻量级的Web 服务器/反向代理服务器，在BSD-like 协议下发行。其特点是占有内存少，并发能力强，事实上nginx的并发能力确实在同类型的网页服务器中表现较好，中国大陆使用nginx网站用户有：百度、京东、新浪、网易、腾讯、淘宝等。

下载地址：<http://data.marsgis.cn/dataproce/nginx.rar>



安装步骤

- 根据自己需要下载对应的版本
- 在终端cd命令进行nginx所在目录
- 终端命令行输入命令启动服务 start nginx
- 浏览器访问 <http://localhost> 检查服务是否正常

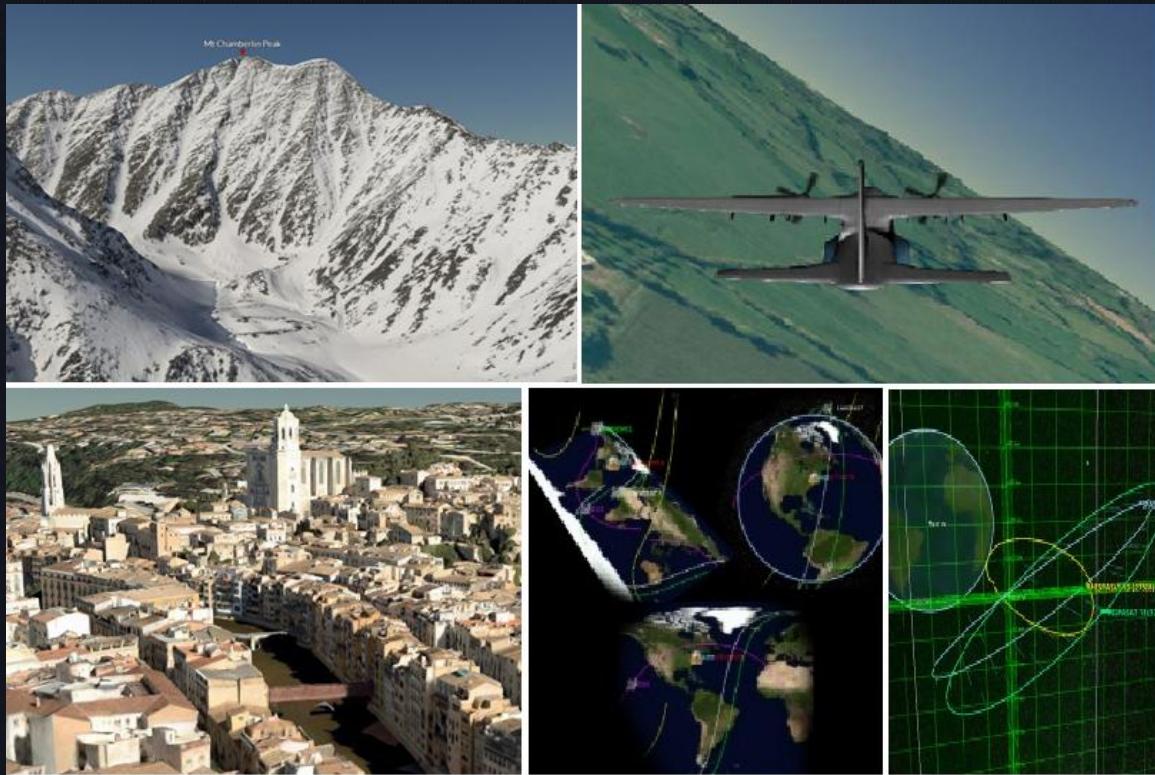
默认端口为80，如有冲突请修改 conf\nginx.conf 文件
站点资源都在 html 子目录下，可以替换进行测试访问

2、Cesium基础知识

- 基础介绍和运行Cesium
- 功能点讲解
- 常见问题
- 源码介绍和打包

2.1 Cesium 介绍

Cesium是一款Web三维地球的世界级的开源产品。基于WebGL技术，是使用JavaScript语言开发，方便用户快速搭建一款零插件跨平台跨浏览器的虚拟地球Web应用。



官网：<https://cesiumjs.org>

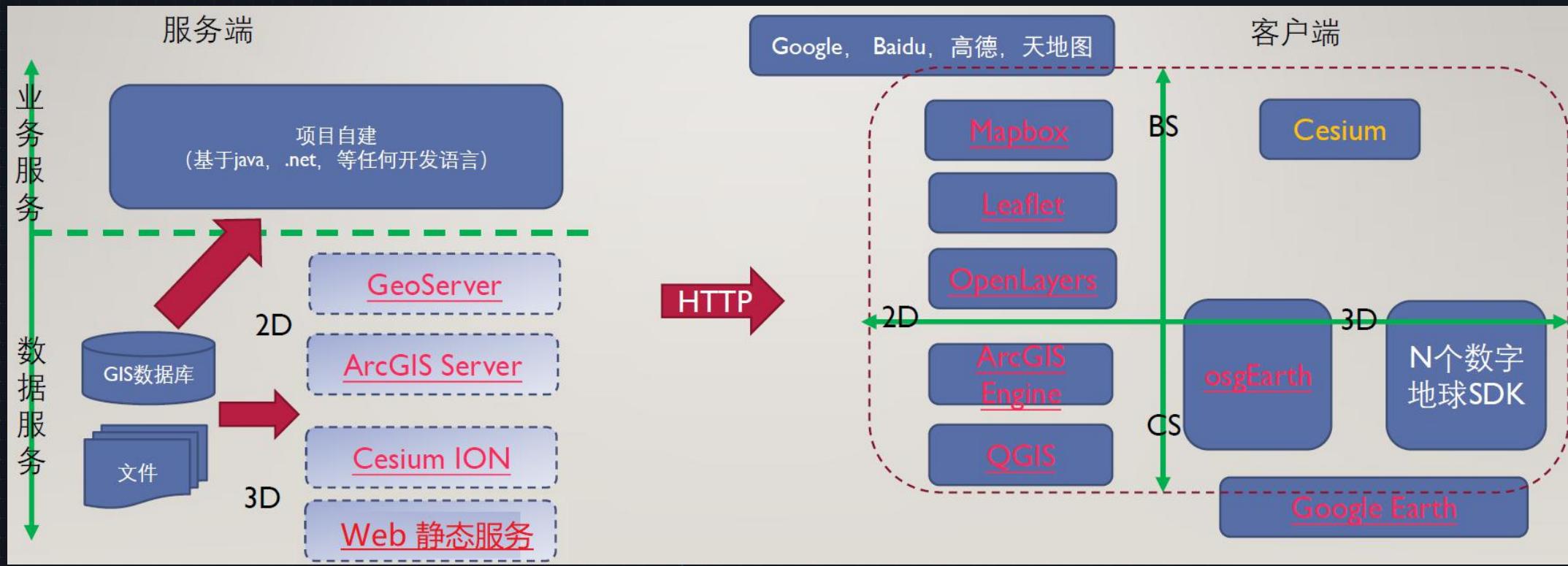
2.2 Cesium 具备的功能

通过Cesium提供的Javascript API，可以实现以下功能：

- 支持鼠标和触摸操作的三维空间 渲染、缩放、惯性平移、飞行、任意视角漫游。
- 支持各种几何体：点、线、面、走廊、管径、墙体、立方体、圆柱、球体等。
- 支持标准的矢量格式 KML、GeoJSON、TopoJSON，以及矢量的贴地效果。
- 支持多种资源的图像层，包括 WMS，TMS，WMTS以及时序图像。支持透明度叠加，亮度等参数的调整。
- 全球高精度地形数据可视化，支持夸张效果、以及可编程实现的等高线和坡度分析效果。
- 使用glTF和3DTiles格式加载各种不同的 3d 数据，包含 倾斜摄影、人工模型、BIM，点云数据等。
- 大气、雾、太阳、阳光、月亮、星星、水面。
- 支持 3d 地球、2d 地图、2.5d 哥伦布模式。3d 视图可以使用透视和正视两种投影方式。
-

2.3 Cesium在项目中的定位

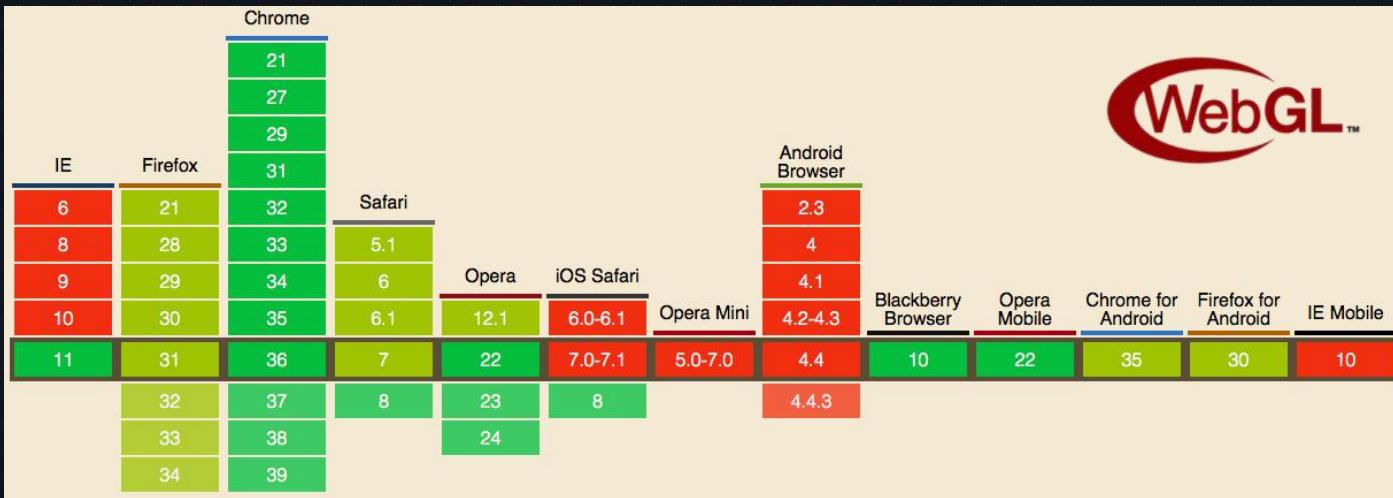
在GIS的数据、服务、应用 三层架构中，Cesium属于客户端应用层面的开发框架。



2.4 Cesium开发的依赖性

- 必须有 HTTP Server来发布后运行 (http方式访问)
- 基于WebGL技术实现三维，所以尽量用最新版本浏览器
- 想要流畅的加载三维模型等大量数据，对显卡有一定要求。

注：可通过浏览器访问[WebGL Report](#)网站，获取浏览器对WebGL的支持清单。



2.5 Cesium所涉及的知识领域

Cesium是涉及三个知识领域：Web前端、计算机图形学、地理信息系统（GIS）的跨界SDK



- Web前端：网上找学习资料
- 计算机图形学：《WebGL编程指南》书籍
- GIS基础知识：培训资料

<http://marsgis.cn/doc/study-gis.pdf>

2.6 Cesium 官方包的下载及运行



- Releases已编译版本包下载地址:

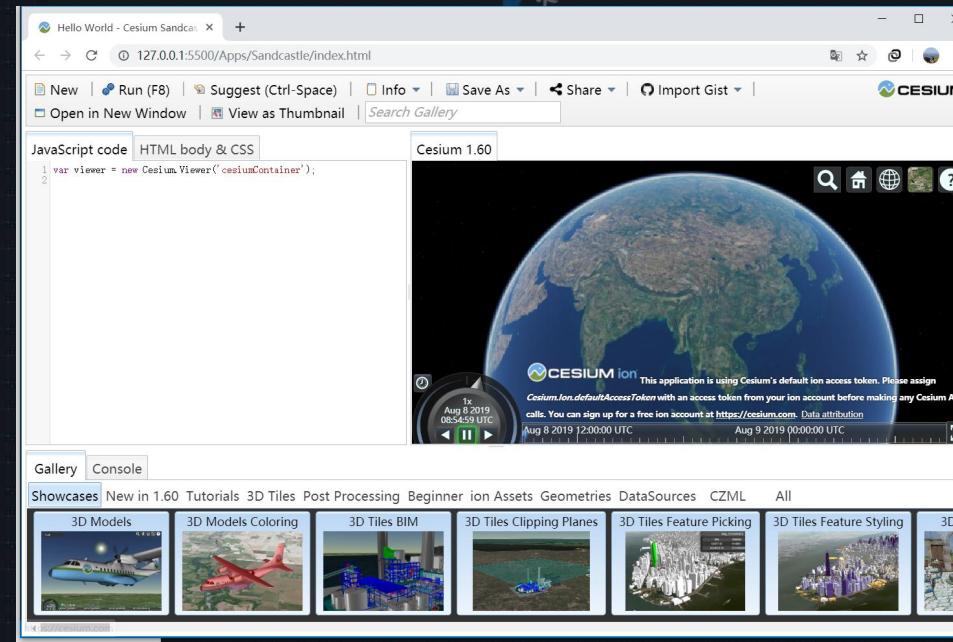
(1) 官网 <https://cesiumjs.org/downloads/>

(2) Github <https://github.com/CesiumGS/cesium/releases>

- VSCode打开Cesium包,并右击浏览示例页面 Apps/Sandcastle/index.html

The screenshot shows the Visual Studio Code interface with the 'index.html' file open. The file content includes HTML, CSS, and JavaScript code, such as the Cesium viewer initialization script:

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="utf-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <meta name="viewport" content="width=device-width, initial-scale=1, max-scale=1, user-scalable=no">
7     <title>Cesium Sandcastle</title>
8     <link rel="stylesheet" href="../../ThirdParty/codemirror-4.6/lib/codemirror.css">
9     <link rel="stylesheet" href="CesiumSandcastle.css">
10    <script>
11      if (window.location.protocol === 'file:') {
12        if (confirm('You must host this app on a web server.\nSee contribution guidelines')) {
13          window.location = 'https://github.com/AnalyticalGraphicsInc/cesium/tree/v1.60/Apps/Sandcastle';
14        }
15      }
16    </script>
17    <script src="../../ThirdParty/jshint-2.1.10/jshint.js"></script>
18
19    <script data-dojo-config="async: 1, t1mSiblingOfDojo: 0" src="../../ThirdParty/dojajs/dojo.js"></script>
20
21    <script src="jsHintOptions.js"></script>
22    <script src="gallery/gallery-index.js"></script>
23    <script type="text/javascript" src="Sandcastle-header.js"></script>
24    <script src="CesiumSandcastle.js"></script>
25    <!-- Needed for autocomplete -->
26    <link rel="stylesheet" href="../../ThirdParty/codemirror-4.6/addon/hint/codemirror-hint.css">
27    <!-- End of autocomplete -->
28
29    <meta property="og:title" content="Cesium Sandcastle" />
30    <meta property="og:description" content="The Cesium Sandcastle provides a quick way to get started with Cesium." />
31    <meta property="og:type" content="website" />
32    <meta property="og:url" content="https://cesiumjs.org/Cesium/Build/Apps/Sandcastle/index.html" />
```



2.6 Cesium 官方包目录介绍

- Apps/SampleData目录: 示例数据目录
- Apps/Sandcastle目录: 官方示例目录
- Build/Cesium目录: 压缩好的cesium库, 项目中主要使用的
- Build/CesiumUnminified目录: 未压缩的cesium库, 可用于项目调试
- Build/Documentation/目录: API文档目录
- Source目录: Cesium源码
- ThirdParty目录: 外部依赖库, 不同于Cesium的第三方库
- index.html: 导航首页
- server.js: 基于node.js的web服务, 我们直接vscode运行, 没用此功能
- LICENSE.md: Cesium的License授权说明

↳	Apps
↳	CesiumViewer
↳	SampleData
↳	Sandcastle
↳	TimelineDemo
↳	HelloWorld.html
↳	Build
↳	Cesium
↳	CesiumUnminified
↳	Documentation
↳	Source
↳	Specs
↳	ThirdParty
↳	CHANGES.md
★	favicon.ico
★	gulpfile.js
★	index.html
★	LICENSE.md
★	package.json
★	README.md
JS	server.js
□	web.config

2.6 Cesium 最简单项目的搭建运行

```

1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>你好Cesium</title>
5
6      <script src="lib/Cesium/Cesium.js"></script>
7      <link rel="stylesheet" href="lib/Cesium/Widgets/widgets.css">
8
9      <style>
10         html, body, #cesiumContainer {
11             width: 100%;
12             height: 100%;
13             margin: 0px;
14         }
15     </style>
16 </head>
17 <body>
18     <div id="cesiumContainer"></div>
19     <script>
20         var viewer = new Cesium.Viewer("cesiumContainer", {
21             animation: false,          //是否创建动画小器件，左下角仪表
22             timeline: false,          //是否显示时间线控件
23             fullscreenButton: false,   //右下角全屏按钮
24             geocoder: false,          //是否显示地名查找控件
25             baseLayerPicker: false,    //是否显示图层选择控件
26             imageryProvider: new Cesium.UrlTemplateImageryProvider({
27                 url: "http://www.google.cn/maps/vt?lyrs=s&x={x}&y={y}&z={z}"
28             })
29         });
30     </script>
31 </body>
32 </html>

```

1. 引入Cesium库的js和css

2. 建立渲染地球的div容器(含css)

3. 调用API接口进行渲染地球

- 新建一个独立的web站点，建立lib子目录和index.html页面
- 拷贝Cesium官方包的Build/Cesium目录到站点lib目录下
- 修改index.html页面,参考左图



2.7 Cesium 戏剧性开局

当你安装顺利搭建运行Cesium后，第一眼就会看到一个漂亮的地球有点沾沾自喜的时候，“领导”在背后说怎么这么丑？这时候领导会一一列举：

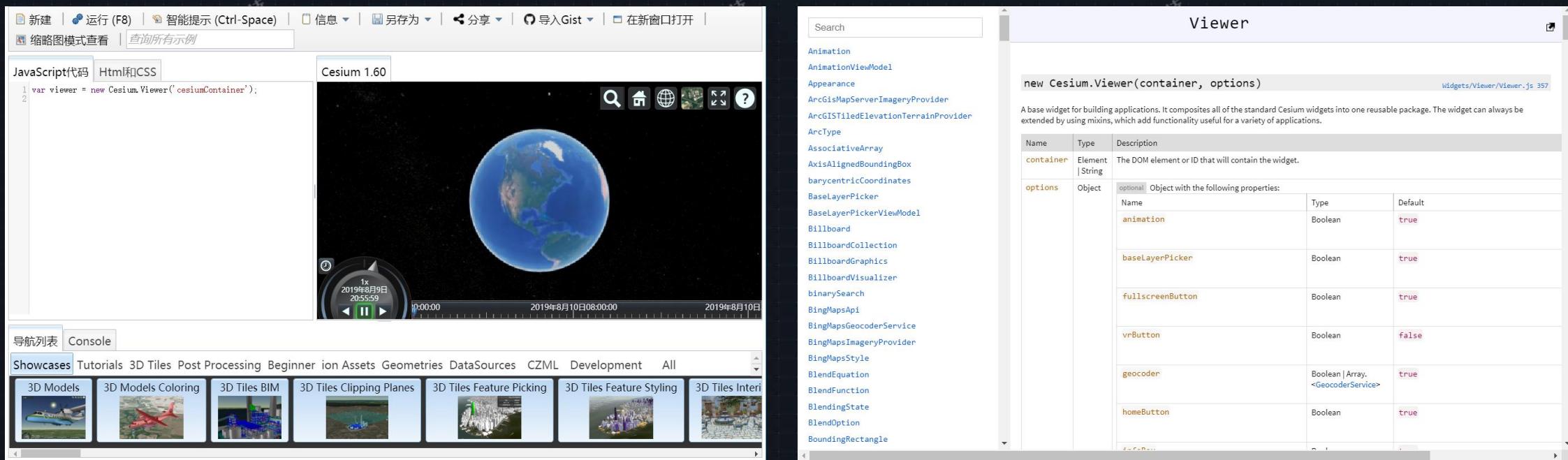


- 初始化视角不是中国
- 界面全英文提示
- 底图不是很美观，能换个高德百度底图就好看了
- 左下角Cesium还有官方Logo以及版权信息
- 加点三维模型就跟逼真了
-

既然已经入了坑，还有个往坑里倒水的“领导”，
咱只能继续填坑.....

2.7 Cesium 自学的第一步

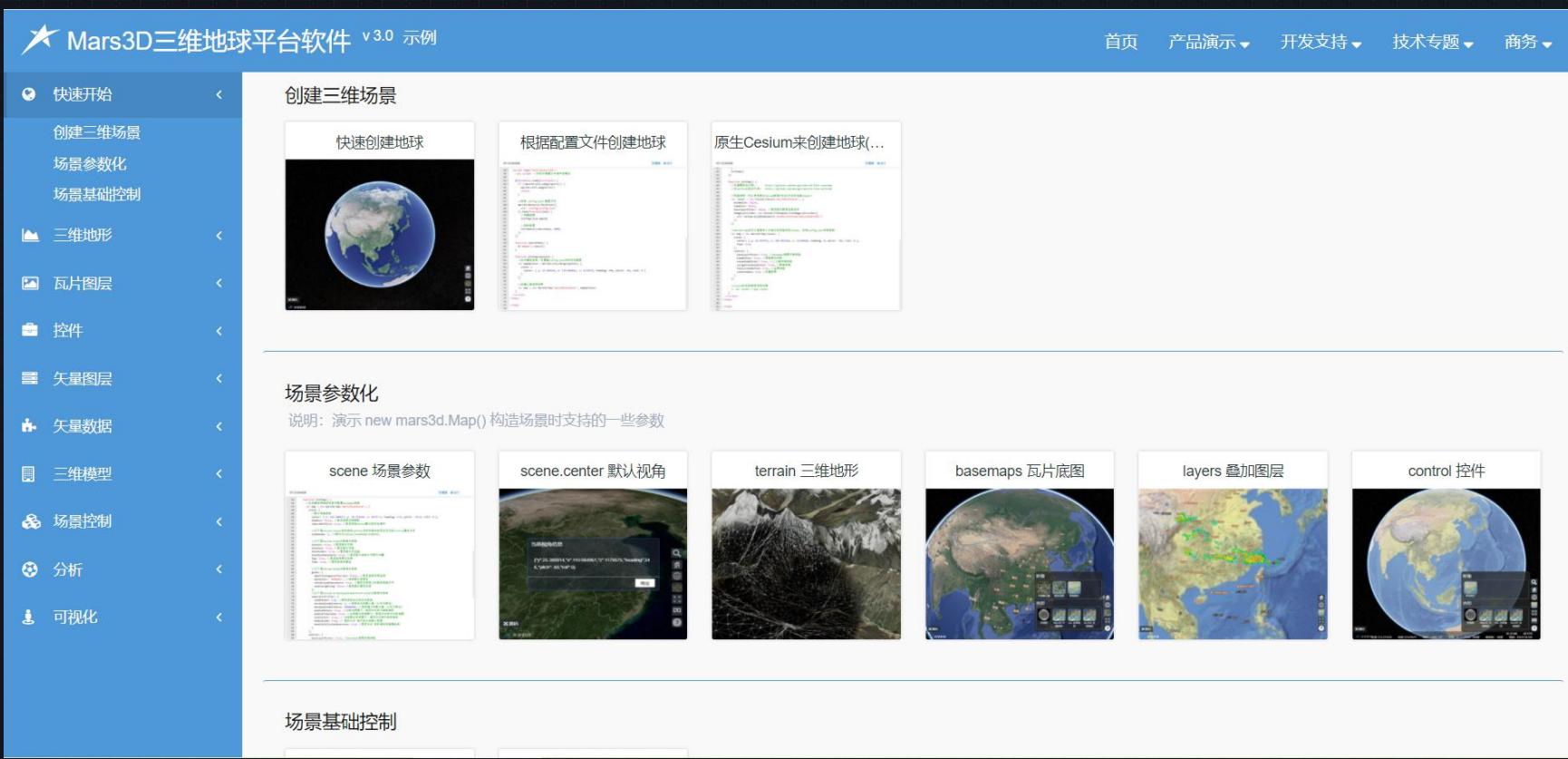
如今Cesium教程网上也有不少，不过我更建议初学者在看教程之前，先跟着Cesium自带的案例学习一遍，有个初步印象，比如Cesium都能展示什么点、线、面？渲染哪些数据类型？怎么拾取？怎么让模型动起来？过程中熟悉它的功能，并对着API文档理解类和方法的含义，并尝试修改一些代码F8运行看看修改后呈现什么效果。



2.7 Cesium 学习路线

官方示例 → 查阅API文档 →  Github → 百度谷歌 → QQ群或社区

快捷路线 → 使用 Mars3D平台 (已经为您填好了很多坑)



Mars3D三维地球平台软件 v3.0 示例

首页 产品演示 开发支持 技术专题 商务

快速开始 <

- 创建三维场景
- 场景参数化
- 场景基础控制

三维地形 <

瓦片图层 <

控件 <

矢量图层 <

矢量数据 <

三维模型 <

场景控制 <

分析 <

可视化 <

创建三维场景

- 快速创建地球
- 根据配置文件创建地球
- 原生Cesium来创建地球...

场景参数化

说明：演示 new mars3d.Map() 构造场景时支持的一些参数

- scene 场景参数
- scene.center 默认视角
- terrain 三维地形
- basemaps 瓦片底图
- layers 叠加图层
- control 控件

场景基础控制

2.8 Cesium 功能点介绍

掌握这些，基本就已经玩转Cesium

- Viewer
- 小控件
- 场景 Scene
- 影像图层 ImageryLayer
- 地形图层 TerrainProvider
- 坐标系及坐标变换
- 相机 Camera
- 交互性(鼠标动作处理器、事件)
- 后期处理 PostProcessStage
- Entity矢量数据
- Primitive矢量数据
- Property机制
- 材质 (Material, Fabric)
- glTF小模型
- 3D Tiles三维模型 Cesium3DTileset
- CZML数据格式
- 粒子系统 ParticleSystem

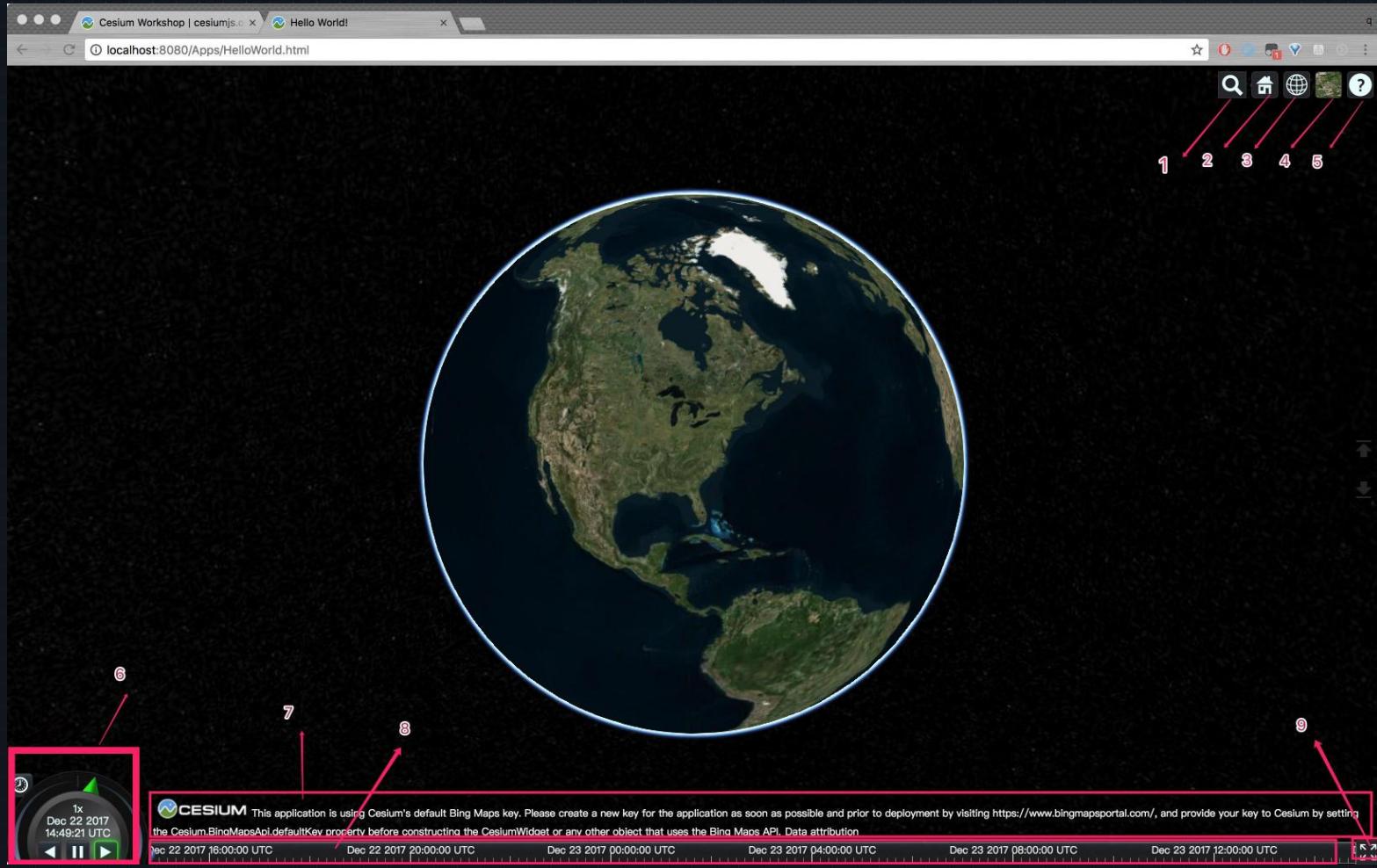
2.8 Cesium 功能点: Viewer

Viewer是一切API的
★ 开始点，new Viewer
后构造的就是我们看得
见的地球对象，Viewer
有很多属性，右侧表格
是我们常用到的。

类型	名称	类型	默认值	说明
控件 显示隐藏	homeButton	boolean	true	视角复位按钮
	sceneModePicker	boolean	true	二三维切换按钮
	navigationHelpButton	boolean	true	帮助按钮
	animation	boolean	true	左下角动画部件按钮
	timeline	boolean	true	下侧时间轴
	fullscreenButton	boolean	true	全屏按钮
	shouldAnimate	boolean	false	自动播放
	vrButton	boolean	false	vr模式按钮
	geocoder	boolean	true	查询按钮
	homeButton	boolean	true	复位按钮
参数	infoBox	boolean	true	信息框
	selectionIndicator	boolean	true	点击选中提示框
	scene3DOnly	boolean	否	3D场景模式
	showRenderLoopErrors	boolean	是	渲染出错窗
	sceneMode			初始化场景模式
图层	terrainExaggeration	number	1	地形夸张系数
	fullscreenElement	string	body	全屏时，渲染的div元素id
	baseLayerPicker	Boolean	true	底图图层控件显隐
	imageryProviderViewModels	baseLayerPicker为 true时有效		可选底图组
	terrainProviderViewModels	baseLayerPicker为 false时有效		可选地形组
	imageryProvider	baseLayerPicker为 true时有效		底图设置
	terrainProvider	baseLayerPicker为 false时有效		地形设置

2.8 Cesium 功能点：小控件

Cesium内置了一些小控件，可以在new Cesium.Viewer时传入参数进行控制。



1. **Geocoder** : 查找位置工具, 查找到之后会将镜头定位至找到的地址
2. **Home Button**: 视角返回初始位置.
3. **Scene Mode Picker** : 选择投影模式,有三种: 3D,2D,哥伦布视图
4. **Base Layer Picker**: 图层选择器, 选择要显示的地图服务和地形服务
5. **Navigation Help Button**: 导航帮助按钮, 显示默认的地图控制帮助.
6. **Animation**: 动画器件, 控制视图动画的播放速度
7. **Timeline**: 时间线, 指示当前时间, 并允许用户跳到特定的时间.
8. **Credits Display**: 版权显示
9. **Fullscreen Button** : 全屏按钮

2.8 Cesium 功能点：场景 Scene

Scene场景是所有3D图形对象的容器（HTML canvas），Scene不是由我们直接创建，它是在Viewer或CesiumWidget内部隐式创建的。

在场景对象中我们可以控制：globe 椭圆体（imageryLayers 底图、 terrainProvider 地形）、camera相机、skyBox天空盒、sun太阳、moon月亮、primitives 默认矢量数据层、postProcessStage 后处理效果等。

Scene场景渲染监听事件触发顺序：（`addEventListener` 和 `removeEventListener` 进行绑定和解绑）

- `viewer.scene.preUpdate` 在更新或呈现场景之前将引发的事件
- `viewer.scene.postUpdate` 在场景更新后以及渲染场景之前立即引发的事件
- `viewer.scene.preRender` 在场景更新后以及渲染场景之前将引发的事件
- `viewer.scene.postRender` 在渲染场景后立即引发的事件

2.8 Cesium 功能点: 影像图层 ImageryLayer

Cesium支持多种服务来源的高精度影像地图数据的加载和渲染。图层支持排序和透明混合。每个图层的亮度(brightness),对比度(contrast), gamma, hue, and saturation 都可以动态修改。

- 可以在Viewer构造时传参进行设置

baseLayerPicker	底图图层控件显隐
imageryProviderViewModels	可选底图组(baseLayerPicker:true)
terrainProviderViewModels	可选地形组(baseLayerPicker:true)
imageryProvider	底图设置(baseLayerPicker:false)
terrainProvider	地形设置(baseLayerPicker:false)

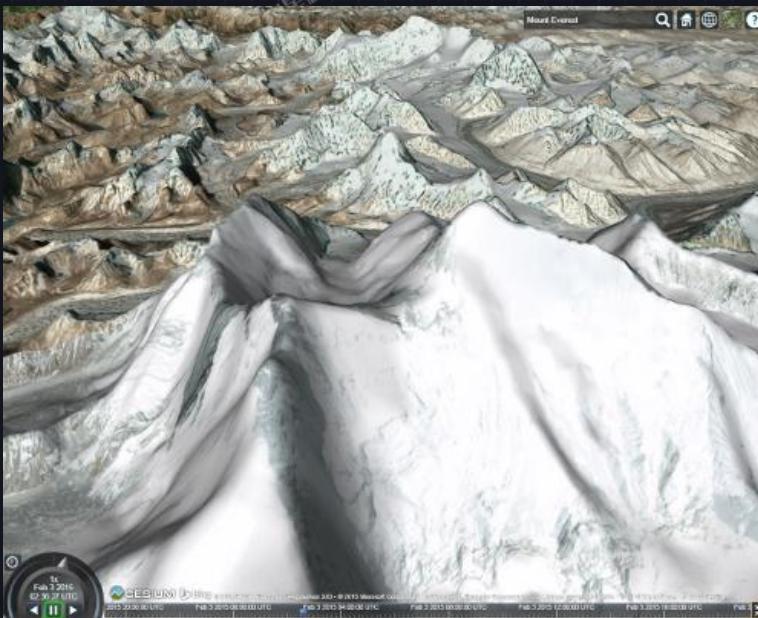
- 在地球构造后可通过 viewer.scene.imageryLayers (ImageryLayerCollection类) 来控制

更多介绍: <https://www.jianshu.com/p/98d4c0b2c618>

2.8 Cesium 功能点: 地形图层 TerrainProvider

支持渐进流式加载和渲染全球高精度地形，并且包含海、湖、河等水面效果。相对2D地图，山峰、山谷等其他地形特征的更适宜在这种3D地球中展示。

地形数据集是巨大的，通常都是GB或者TB级别。在普通3D引擎中，使用底层图形API去高效实现地形数据的可视化需要做很多事情。幸好Cesium已经完成了这个体力活，而我们只需要写几行代码。主要是修改viewer.terrainProvider属性。



- 使用Cesium在线Ion地形 `viewer.terrainProvider = Cesium.createWorldTerrain();`
- 使用自行发布的服务 `viewer.terrainProvider = new Cesium.CesiumTerrainProvider({url: "http://localhost/terrain"});`
- 不使用地形 `viewer.terrainProvider = new Cesium.EllipsoidTerrainProvider();`

2.8 Cesium 功能点：常用坐标系

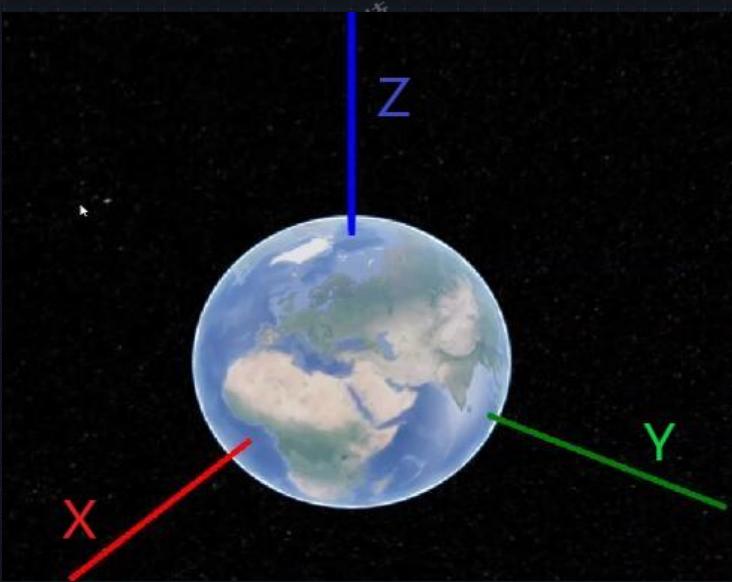
- 屏幕坐标（像素值）
- 笛卡尔平面坐标
- 笛卡尔空间直角坐标
- 地理坐标（默认为弧度值）



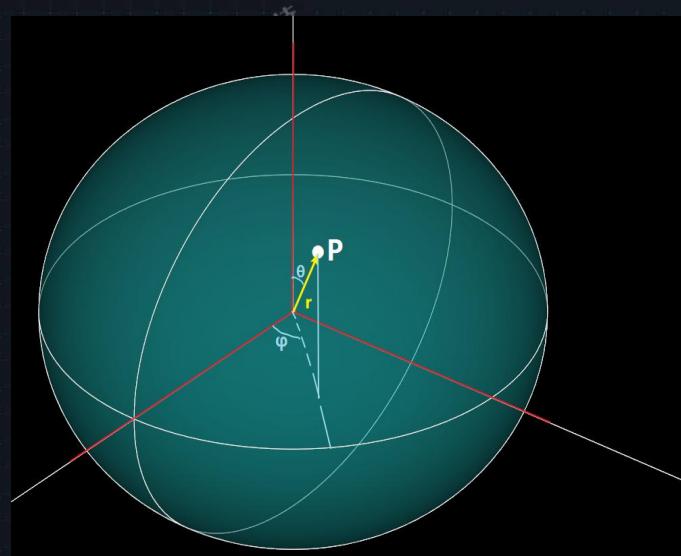
`new Cesium.Cartesian2(x, y)`

`new Cesium.Cartesian3(x, y, z)`

`new Cesium.Cartographic(longitude, latitude, height)`



笛卡尔空间直角坐标



地理坐标（经度 ϕ , 纬度 θ , 高度）

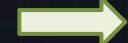
1. 笛卡尔坐标系原点是地球几何中心；
2. xz平面是中央经线和180度经线组成的平面，其中x轴正方向指向的是中央经线，x轴负方向指向180度经线；
3. y轴正方向指向东经90度经线，负方向指向西经90度经线。



弧度

2.8 Cesium 功能点：坐标变换(1)

- 获取鼠标单击后在屏幕中的坐标



```
var hander = new Cesium.ScreenSpaceEventHandler(viewer.scene.canvas);
hander.setInputAction(function (event){          // 绑定鼠标左点击事件
    var windowPosition = event.position;        // 获取鼠标点的windowPosition
}, Cesium.ScreenSpaceEventType.LEFT_CLICK);
```

- 屏幕坐标 转换为 笛卡尔空间直角坐标

//三维模式下

```
var ray = viewer.camera.getPickRay(windowPosition);
```

```
var cartesian = viewer.scene.globe.pick(ray, viewer.scene);
```

//二维模式下

```
var cartesian = scene.camera.pickEllipsoid(position, scene.globe.ellipsoid);
```

- 笛卡尔空间直角坐标 转换为 屏幕坐标

```
var pick = Cesium.SceneTransforms.wgs84ToWindowCoordinates(viewer.scene, cartesian);
```

2.8 Cesium 功能点：坐标变换(2)

- 笛卡尔空间直角坐标 转换为 地理坐标(弧度制)

```
var cartographic=Cesium.Cartographic.fromCartesian(cartesian)
```

- 地理坐标(弧度制) 转换为 笛卡尔空间直角坐标

```
var position = Cesium.Cartesian3.fromRadians(lng, lat, height)
```

- 笛卡尔空间直角坐标 转换为 地理坐标(经纬度)

```
var cartographic=Cesium.Cartographic.fromCartesian(cartesian)
```

```
var lat=Cesium.Math.toDegrees(cartographic.latitude);
```

```
var lng=Cesium.Math.toDegrees(cartographic.longitude);
```

```
var height=cartographic.height;
```

➤ 度数与弧度互转
Cesium.Math.toDegrees(radians)
Cesium.Math.toRadians(degrees)

- 地理坐标(经纬度) 转换为 笛卡尔空间直角坐标

```
var position = Cesium.Cartesian3.fromDegrees(longitude, latitude, height)
```

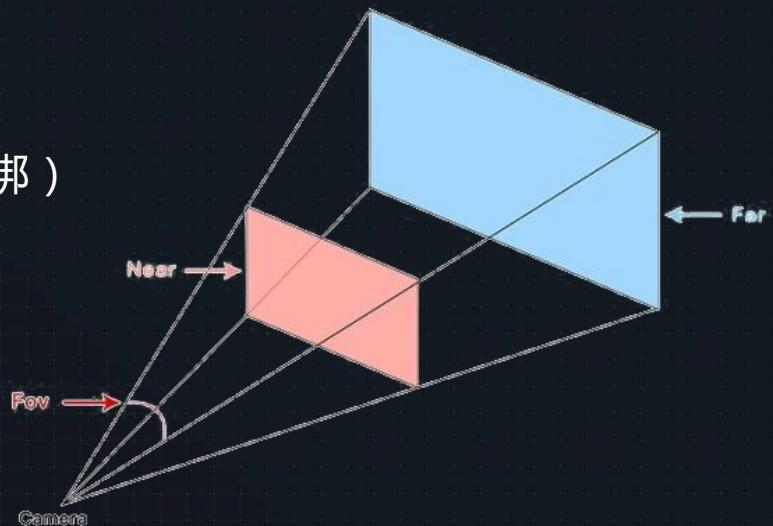
2.8 Cesium 功能点：相机 Camera

相机控制了场景的观察视角。有很多相机操控方法，比如旋转、缩放、平移以及飞行定位。其内部默认已支持使用鼠标和触摸事件控制相机。

- Camera类（相机）描述了相机的当前状态，包括：位置（position），朝向（orientation），参考空间（reference frame），视锥体（view frustum），可以调用move*、zoom*、look*、twist*、rotate*、setView、flyTo等方法进行控制相机操作
- ScreenSpaceCameraController类（屏幕控件相机控制器）把屏幕空间的用户输入（鼠标拖拽点击或者触摸事件）转换为三维世界的相机移动。它包含一些属性，可以启用/禁用某种用户输入，修改惯性、最小最大缩放距离等。

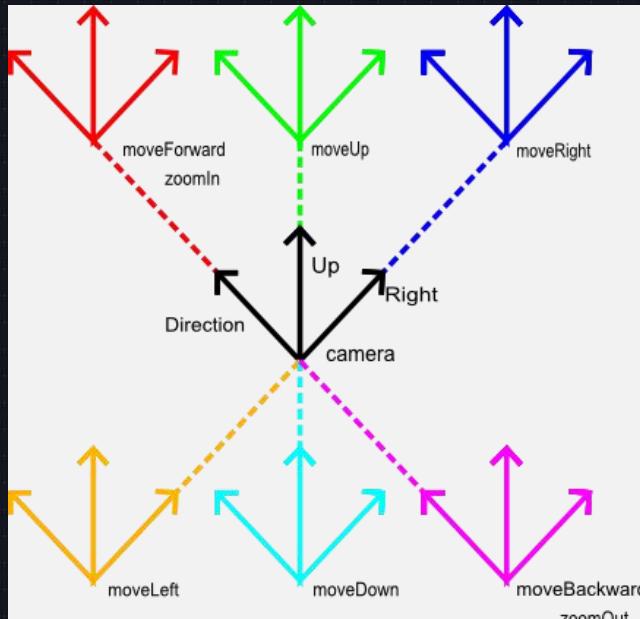
相机相关事件：（`addEventListener` 和 `removeEventListener` 进行绑定和解绑）

- `viewer.camera.moveStart` 相机开始移动时将引发的事件
- `viewer.camera.moveEnd` 相机停止移动时将引发的事件
- `viewer.camera.changed` 相机更改后将引发的事件

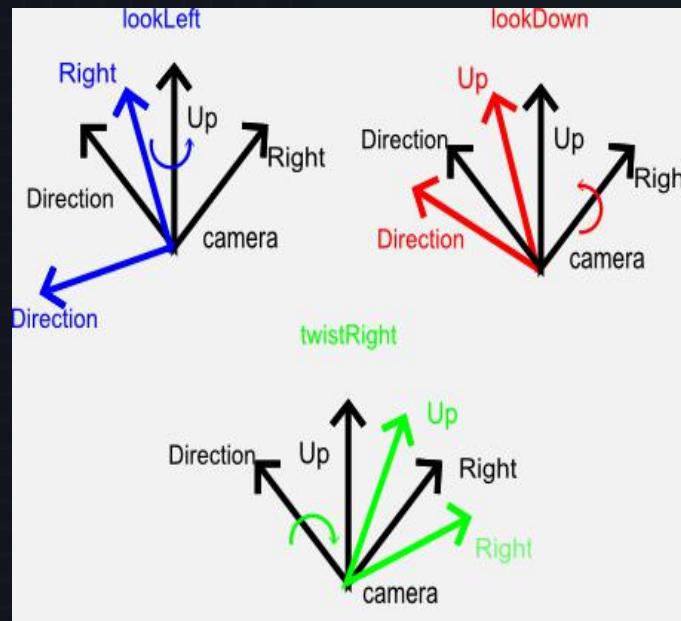


2.8 Cesium 功能点: 相机 Camera – 基本操作方法

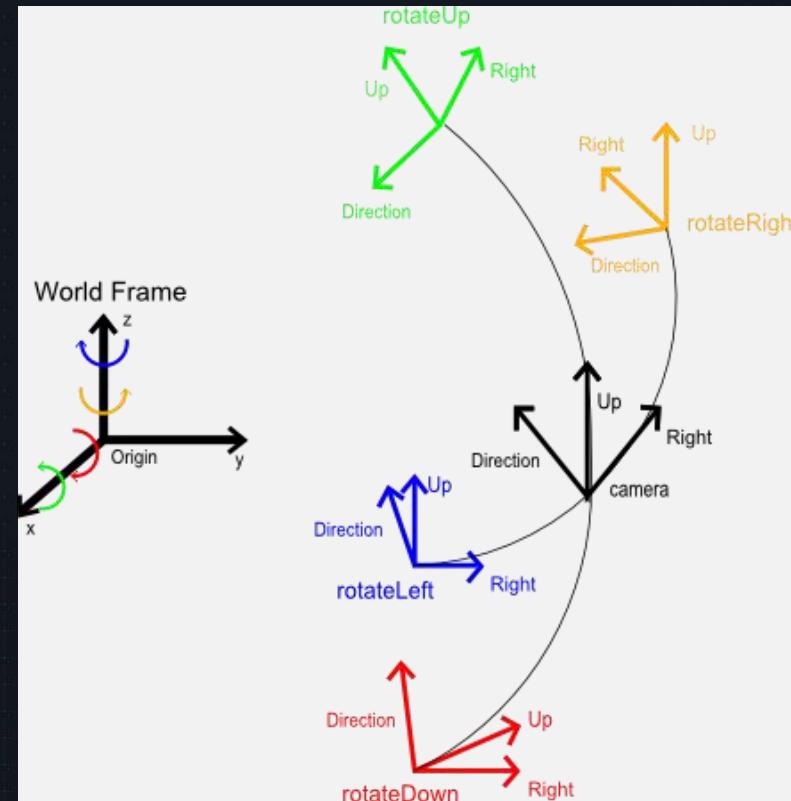
- move* 和 zoom* 方法: 沿着相机方向或者某个给定向量来平移相机的位置。 相机朝向不变。
- look* 和 twist* 方法: 旋转相机朝向, 向前向量 (direction) , 向上向量 (up) , 向右向量 (right) 都会改变。 相机位置保持不变。
- rotate*方法: 相对一个给定的向量, 旋转相机的位置和朝向。



相机平移



朝向旋转



旋转化动

2.8 Cesium 功能点：交互性(鼠标动作处理器、事件)

ScreenSpaceEventHandler类 处理用户输入事件。可以添加自定义函数，以便在用户输入时对其执行。

- setInputAction(action, type, modifier) 设置事件
- getInputAction(type, modifier) 得到交互事件
- removeInputAction(type, modifier) 移除事件
- destroy() 销毁Handle
- isDestroyed() 判断是否销毁

ScreenSpaceEventType类 为事件类型，包含鼠标单击、双击、按下、抬起、滚轮、右击等

示例：

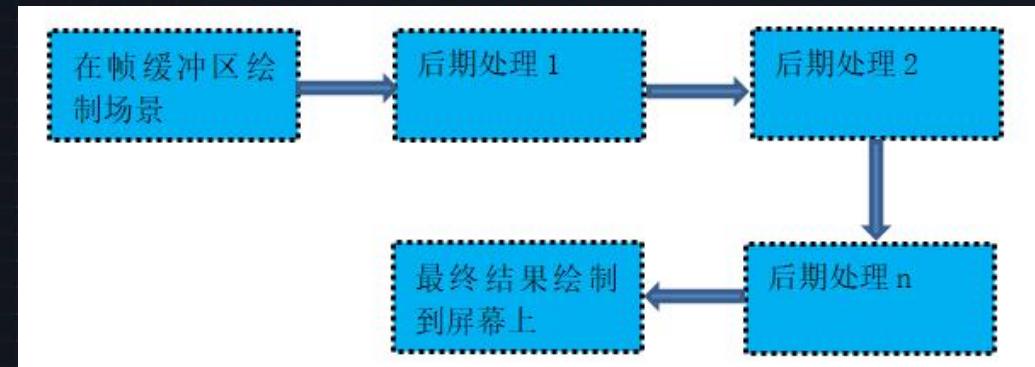
```
var hander = new Cesium.ScreenSpaceEventHandler(viewer.scene.canvas);
hander.setInputAction(function (event){    // 绑定事件
}, Cesium.ScreenSpaceEventType.LEFT_CLICK)
```

2.8 Cesium 功能点：后处理 PostProcessStage

Cesium支持对整个场景的后处理(Post Processing)功能，包括模型描边、黑白图、明亮度调整、夜视效果、环境光遮蔽等。后期处理的过程有点类似于照片的PS。

- PostProcessStage：对应于某个具体的后期处理效果，它的输入为场景渲染图或者上一个后期处理的结果图，输出结果是一张处理后的图片。
- PostProcessStageComposite：一个集合对象，存储类型为PostProcessStage或者PostProcessStageComposite的元素。
- PostProcessStageLibrary：负责创建具体的后期处理效果，包括Silhouette、Bloom、AmbientOcclusion等，创建返回的结果是PostProcessStageComposite或者PostProcessStage类型。
- PostProcessStageCollection：是一个集合类型的类，负责管理和维护放到集合中的元素，元素的类型是PostProcessStage或者PostProcessStageComposite。

更多介绍：<https://www.cnblogs.com/webgl-angela/p/9272810.html>



2.8 Cesium 功能点: Primitive 与 Entity

Cesium加载点线面矢量数据分为两部分:

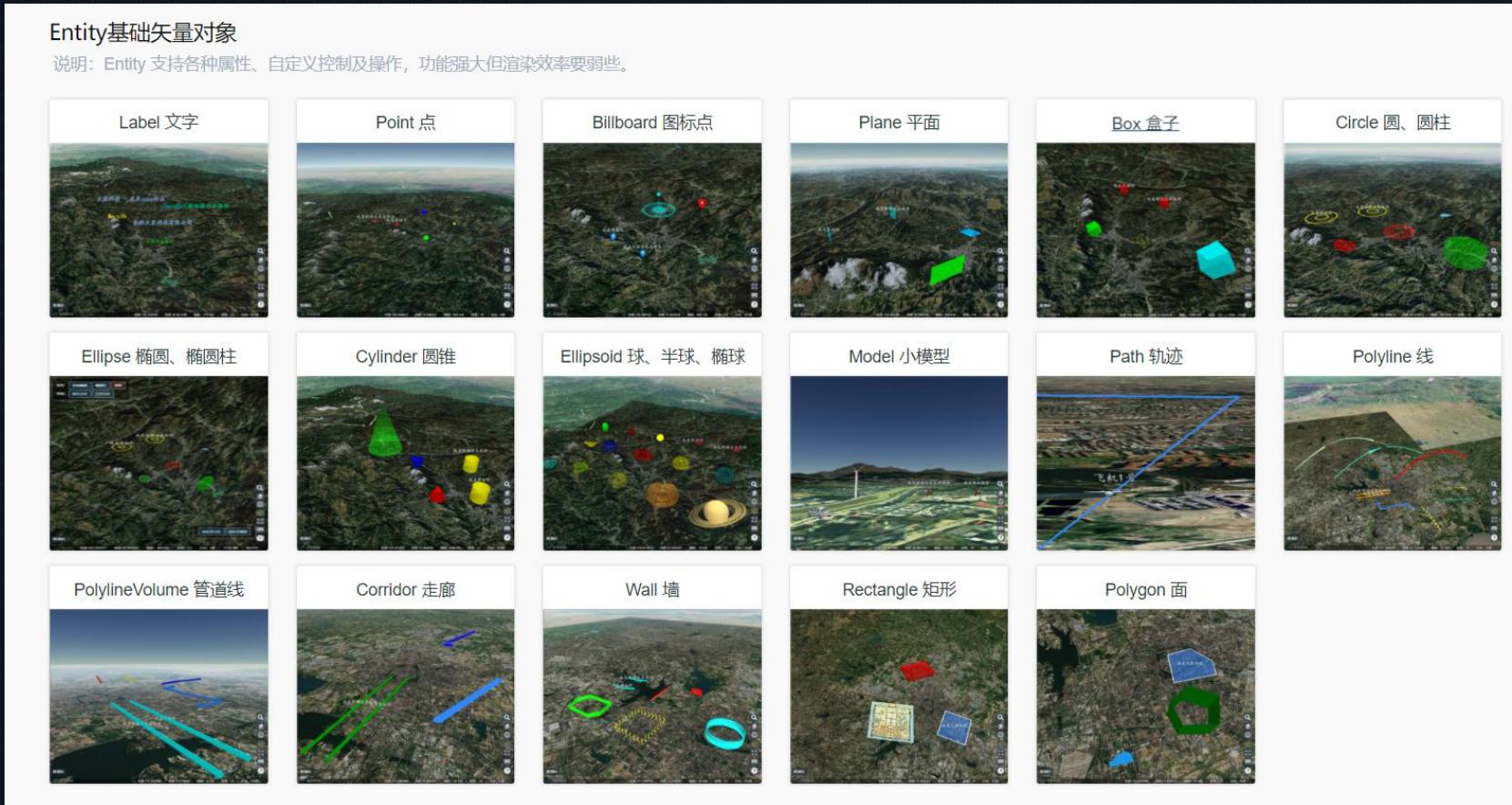
- Primitive API 面向三维图形开发者，更底层一些 【灵活、性能高、使用复杂】
- Entity API 是数据驱动更高级一些 【性能略低、接口一致、容易使用】

Primitive API 的主要目的是为了完成（可视化）任务的最少的抽象需求。它很强大又很灵活，要求我们以一个图形开发者的方式去思考，并且使用了一些图形学术语。它是为了最高效最灵活的实现可视化效果，忽略了API的一致性。他们每种都有自己的独特的性能提升方式，也需要遵守不同的优化原则。

Entity API 的主要目的是定义一组高级对象，它们把可视化和信息存储到统一的数据结果中，这个对象叫Entity。它让我们更加关注我们的数据展示而不是底层的可视化机制。它提供了很方便的创建复杂的与静态数据相匹配的随时间变化的可视化效果。Entity内部也是使用了Primitive，它的实现细节，我们无需关心，Entity暴露一些一致性的、容易去学习和使用的接口。

2.8 Cesium 功能点： Entity矢量数据

通常使用 viewer.entities.add 方法进行添加Entity矢量数据，或者使用CustomDataSource 对象进行管理，支持的类型见下图：



在线示例：<http://mars3d.cn/examples.html#graphic-feature-entity>

2.8 Cesium 功能点： Primitive矢量数据

Primitive方式更接近渲染引擎底层，需要理解Primitive API参数时需要您已有WebGL知识储备，建议先学习下[《WebGL编程指南》](#)， Primitive由两个部分组成：

- 几何形状 (Geometry)：定义了Primitive的结构，例如三角形、线条、点等
- 外观 (Appearance)：定义Primitive的着色 (Shading)，包括GLSL (OpenGL着色语言) 顶点着色器和片段着色器 (vertex and fragment shaders)，以及渲染状态 (render state)

使用Geometry和Appearance 具有以下优势：

- (1) 性能：绘制大量Primitive时，可以将其合并为单个Geometry以减轻CPU负担、更好的使用GPU。合并Primitive由web worker线程执行，UI保持响应性
- (2) 灵活性：Geometry与Appearance 解耦，两者可以分别进行修改
- (3) 低级别访问：易于编写GLSL 顶点、片段着色器、使用自定义的渲染状态

同时，具有以下劣势：

- (1) 需要编写更多地代码
- (2) 需要对图形编程有更多的理解，特别是OpenGL的知识

更多介绍：

- <https://www.cnblogs.com/fuckgiser/p/5706842.html>
- <https://www.jianshu.com/p/5a74c607a591>
- <http://cesium.marsgis.cn/go.html?id=12>
- <https://www.jianshu.com/p/fc0252e63204>

2.8 Cesium 功能点: Property机制

Cesium宣称自己是数据驱动和time-dynamic visualization，这些可都是仰仗Property系统来实现的。Property最大的特点是和时间相互关联，在不同的时间可以动态地返回不同的属性值。而Entity则可以感知这些Property的变化，在不同的时间驱动物体进行动态展示。



2.8 Cesium 功能点：材质（Material、Fabric）

Fabric 是Cesium中基于JSON格式来描述Material的机制。材质描述多边形、折线、椭球等对象的外观特征。材质可以简单的是覆盖一张图片，或者是条纹或者棋盘图案。使用Fabric 和GLSL，可以从零开始写脚本新建材质，也可以从现有的材质中派生。比如潮湿碎裂的砖块可以使用程序生成的纹理、凹凸贴图和反射贴图来组合。对象通过material 属性来支持材质效果。

名称	效果图	描述
颜色 Color		包含透明通道的颜色值
贴图 Image		jpg或者png格式的图片，可以带透明通道，用 <code>rgb</code> 表示颜色， <code>a</code> 表示透明度
棋盘图 Checkerboard		明暗交替组成的棋盘图。
条纹 Stripe		水平或者垂直方向明暗交替的图案
斑点 Dot		按行列排列的点组成
网格 Grid		网格边线，描述3D体的时候有用

更多介绍：

<https://www.cnblogs.com/fuckgiser/p/6171245.html>
<https://www.jianshu.com/p/f8fee864379a>

名称	效果图	描述
漫反射贴图 DiffuseMap		一张图片定义了光在所有方向上的散射颜色，一般是个三维向量 (<code>vec3</code>)
高光反射贴图 SpecularMap		一张图片，定义 (<code>scalar</code>)，通常表示反光强度
透明贴图 AlphaMap		一张图片，定义让一部分表面透明
法向贴图 NormalMap		一张图片，定义法向量 (<code>vec3</code>)，提升了表面渲染的细节
凹凸贴图 BumpMap		一张图片，定义凹凸贴图，也可提升表面渲染的细节，它改变了法向量
自发光贴图 EmissionMap		一张图片，定义发光量 (<code>vec3</code>)。比漫反射贴图更亮

名称	效果图	描述
带箭头折线 PolylineArrow		在折线尾端增加一个箭头
泛光折线 PolylineGlow		折线泛光
带边界折线 PolylineOutline		带边界折线
水面 Water		带波纹动画的水面
外轮廓高亮 RimLighting		高亮边缘或者轮廓

2.8 Cesium 功能点： glTF小模型

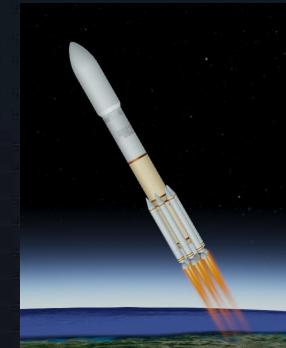
glTF小模型只是 Primitive 或Entity的其中一种类型数据，代码加载方式是一致的，这里单独拿出来讲解下他的.gltf数据特性。规范：<https://github.com/KhronosGroup/glTF>

Cesium支持包含关键帧 (key-frame) 动画、骨骼 (skinning) 动画 的glTF格式的三维模型，并且支持模型节点 (node) 的拾取。glTF是 Khronos Group 定义的一个基于web上的新兴三维模型格式行业标准。Khronos Group是WebGL和 COLLADA的背后财团。

格式转换：<https://cesiumjs.org/convertModel/>

其他资源：官方示例中 [3D models](#)，官方文档 [Model](#) 类和[ModelAnimationCollection](#)类

更多介绍：<https://www.jianshu.com/p/47cd185b58c7>



2.8 Cesium 功能点： 3D Tiles三维模型 cesium3DTileset

Cesium与开源社区合作开发了3D Tiles，这是一个开放的规范，用于传输海量的异构三维地理空间数据集。使用概念上类似于terrain和imagery的瓦片流技术，3D Tiles 使得建筑物数据集、BIM模型、点云和摄影测量模型等大模型比较流畅的在Web端进行浏览展示。

3D Tiles 还允许我们使用3D Tiles styling语言 来调整我们的样式。3D Tiles 样式定义了用于评估颜色（RGB和透明度）的表达式，并显示了**Cesium3DTileFeature**特征的属性，这是tileset的一部分。样式是用JSON定义的，而表达式是在JavaScript的小子集中编写的，用于样式化。此外，样式语言提供了一组内置函数来支持常见的数学运算。

数据规范：<https://github.com/AnalyticalGraphicsInc/3d-tiles>

更多示例：[the 3D Tiles sandcastle demos](#)

2.8 Cesium 功能点： 3D Tiles三维模型 – 代码示例

```
var tileset = viewer.scene.primitives.add(new Cesium.Cesium3DTileset({
    url: "http://localhost/3DTiles/jzw-shanghai/tileset.json", //模型服务地址
    maximumScreenSpaceError: 1
});
tileset.readyPromise.then(function (tileset) {
    var boundingSphere = tileset.boundingSphere;
    var hpr = new Cesium.HeadingPitchRange(0.0, -0.5, boundingSphere.radius * 2)
    viewer.camera.flyToBoundingSphere(boundingSphere, hpr); //视角定位至模型

    tileset.style = new Cesium.Cesium3DTileStyle({ //设置样式
        color: {
            conditions: [
                ["${floor} >= 300", "rgba(45, 0, 75, 0.5)"],
                ["${floor} >= 100", "rgb(170, 162, 204)"],
                ["${floor} >= 25", "rgb(252, 230, 200)"],
                ["${floor} >= 5", "rgb(198, 106, 11)"],
                ["true", "rgb(127, 59, 8)"]
            ]
        }
    });
}).otherwise(function (error) {
    throw (error);
});
```



运行后效果：http://mars3d.cn/example/editor.html#g15_dth_inside

2.8 Cesium 功能点：CZML数据格式

CZML是一种用来描述动态场景的JSON架构的语言，主要用于Cesium数据与程序分离，就如同Google Earth和KML的关系。采用数据驱动的方式，不用写代码即可构建出丰富的动态场景。

一个CZML文档包含一个JSON数组，数组中每一个对象都是一个CZML数据包（packet），一个packet对应一个场景中的对象，例如一个飞机。

`Cesium.CzmlDataSource.load` 可以加载czml对象或者czml文档的url

CZML比较特殊的是跟时间序列相关的属性。

格式说明：<https://github.com/AnalyticalGraphicsInc/czml-writer/wiki/CZML-Guide>

```
1  [
2  {
3      "version": "1.0",
4      "id": "document",
5      "clock": {
6          "interval": "2019-03-27T05:57:05.33Z/2019-03-27T05:57:47.33Z",
7          "currentTime": "2019-03-27T05:57:05.33Z",
8          "multiplier": 1
9      },
10     {
11         "id": "汽车漫游",
12         "availability": "2019-03-27T05:57:05.333Z/2019-03-27T05:57:47.33Z",
13         "orientation": { "velocityReference": "#position" },
14         "position": {
15             "epoch": "2019-03-27T05:57:05.33Z",
16             "cartographicDegrees": [
17                 0, 121.92203, 40.896242, 1.99,
18                 10, 121.923628, 40.898229, 6.41,
19                 20, 121.925196, 40.898912, 1.99,
20                 30, 121.926071, 40.897773, 1.98
21             ],
22         },
23         "model": {
24             "show": true,
25             "gltf": "../gltf/qiche.gltf",
26             "scale": 0.1,
27             "minimumPixelSize": 50
28         }
29     }
30 ]
```

2.8 Cesium 功能点：粒子系统 ParticleSystem

粒子系统是一种图形技术，可以模拟复杂的物理效果。粒子系统是小图像的集合，当它们一起观看时，会形成一个更复杂的“模糊”物体，如火、烟、天气或烟花。通过使用诸如初始位置、速度和寿命等属性指定单个粒子的行为，可以控制这些复杂的效果。

粒子发射器（ParticleEmitter）控制了粒子产生时候的位置以及初始速度方向，并依据emissionRate来决定每秒产生多少粒子，根据发射器类型不同决定了粒子的随机速度方向。



- CircleEmitter: 圆形发射器使用CircleEmitter类在圆形面上随机一个位置，粒子方向是发射器的向上向量。它接受一个float参数指定了圆的半径。
- BoxEmitter: 在盒子里（box）里随机一个位置，沿着盒子的6个面的法向量向外运动。它接受一个Cartesian3 参数，定义了盒子的长宽高。
- ConeEmitter: 锥形发射器类使用ConeEmitter在椎体顶点产生粒子，粒子方向在椎体内随机一个角度向外。它接受一个float参数，制定了锥角。
- SphereEmitter: 球形发射器使用SphereEmitter类在球体内随机产生粒子，初始速度是沿着球心向外。它接受一个float参数指定了球体半径。

更多介绍：

<https://www.jianshu.com/p/cb18e2c8ba72>

<https://www.jianshu.com/p/339c3685ba00>

2.8 Cesium 功能点：结语

以上的讲解只是简单的让大家对Cesium的功能体系有个大概了解，其实每一个功能点都可以作为单独课程来讲解，后面大家可以按照这个大纲进一步深入学习，可参考官方示例和互联网教程等资源进行实践开发，一步步加深印象，学以致用。

在编写本章节时，参阅了大量资料。

感谢左侧人员和组织对Cesium普及和推广所做的贡献（排名不分先后）：

Peter Lu (法克鸡丝)

- 唐晓飞 (vtxf)
- 张言哲 (cumtzheNo1)
- Kamijawa
- 伐罗密 (cesium.xin)
- 自由战士
- Cesium中文社区(cesiumcn.org)
- 北京西部世界 公司 (CesiumLab)
- 火星科技 公司 (Mars3D) 感谢公司各同事对PPT的优化

2.9 初学者常见问题

Cesium是个跨学科的产品，web前端、图形学、GIS相关问题都可能碰到。下面我们来讲解一些初学者遇到的一些毫无头绪的问题及其解决思路和方法。当然问题永远无法全面罗列，下面就几个常见问题分析下：

- 加载的模型会在地面漂移
- 加载外部资源的跨域问题
- 加载的矢量数据或倾斜模型有偏移
- 加载的gltf模型不出现
- 立体面的拉伸高度设置问题

这些问题只是抛砖引玉，讲解下解决问题的思路和方法。实际我们学习和研发就是不断的发现问题、解决问题的过程。

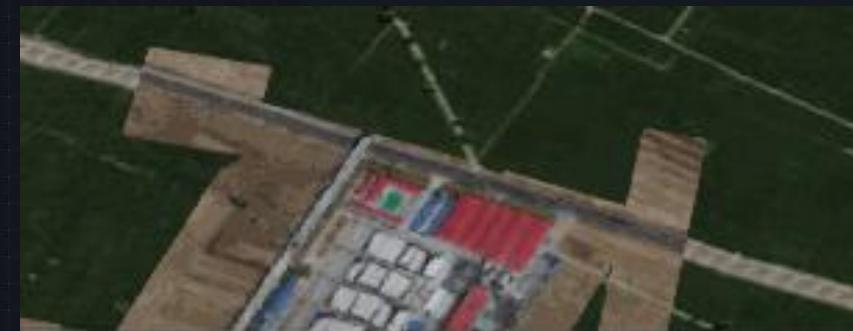
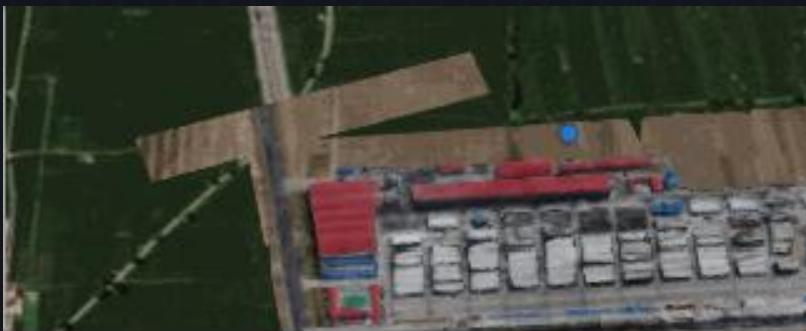
2.9 初学者常见问题：加载的模型会在地面漂移

问题：加载的矢量数据或模型，浏览时会在地面漂移

解决方法：

因为模型或数据的高度在地面以下造成的（可类比水面折射看水底物体现象）

- ✓ 开启深度检测`viewer.scene.globe.depthTestAgainstTerrain = true;`
- ✓ 如果看不到模型，说明高度确实有问题，调整模型高度
- ✓ 关闭深度检测`viewer.scene.globe.depthTestAgainstTerrain = false;`



2.9 初学者常见问题：加载外部资源的跨域问题

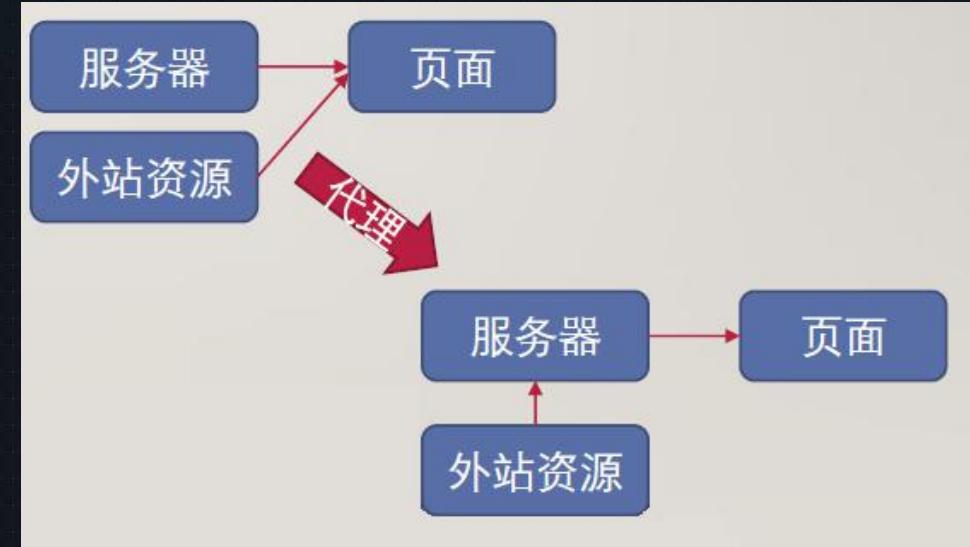
问题：浏览器提示CORS policy错误，所加载的底图、地形、模型等外部数据没有显示。

解决方法：

因为浏览器和WebGL的安全性要求，cesium加载的所有外部资源，都需要服务端允许跨域访问，跨域问题的终极解决方法在服务端：

- 若服务端可控：添加跨域头
- 若服务端不可控：添加代理服务

跨域解决的具体步骤可以参看下一章“服务发布”



2.9 初学者常见问题：加载的矢量数据或倾斜模型位置不正确



问题：使用国内公开访问的底图服务时（如高德百度、国内谷歌等）都是国测局偏移坐标系，如果数据是标准无偏的，叠加后肯定会有偏移。

问题原因：

坐标系问题，使用与数据相同坐标系的底图，或者对矢量数据做加偏处理。



发生了偏移



无偏移

（GIS坐标系的理解可以参看我们另外的ppt教程《GIS基础知识培训》中有详细讲解）

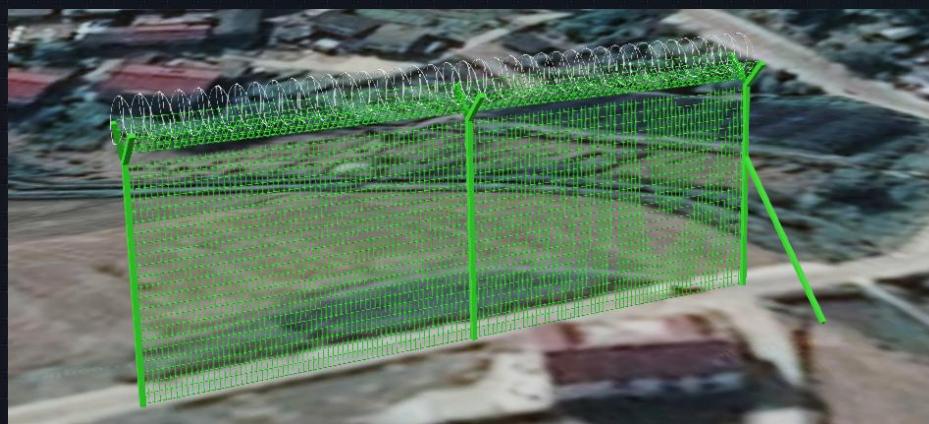
另：MarsGIS产品内已有算法对矢量数据和倾斜模型进行加偏处理。

2.9 初学者常见问题：加载的glTF模型不出现

问题：按示例代码加载glTF模型后在地球上看不到

解决思路：

- 把URL地址拷贝出来在浏览器栏里访试下,保证network成功请求了模型
- 模型比例大小问题，修改模型缩放scale参数或缩放下视角进行浏览
- 模型位置在地面以下，可以尝试关闭深度检测
(`viewer.scene.globe.depthTestAgainstTerrain = false`)



2.9 初学者常见问题：立体面的拉伸高度设置问题

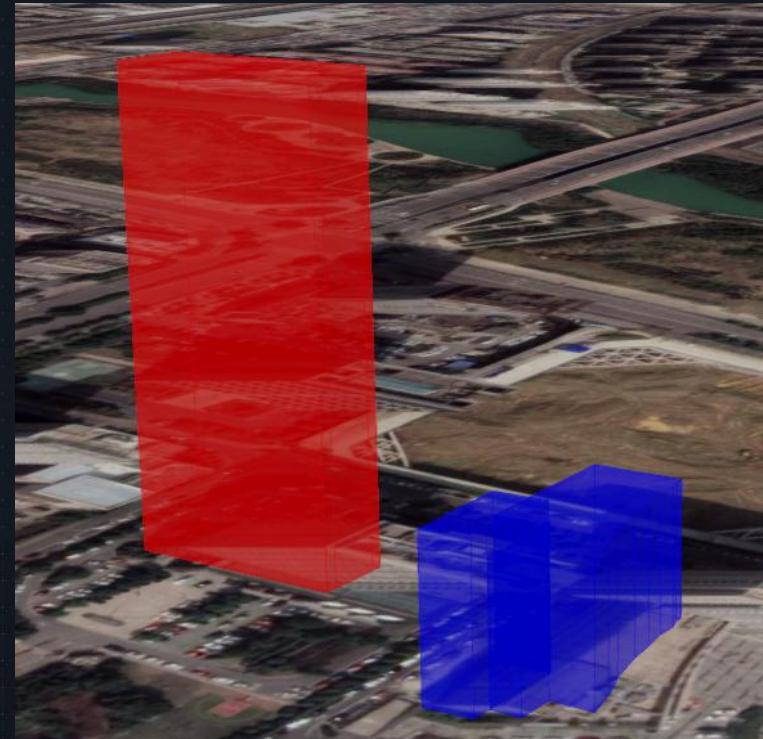
问题： polygon entity对象的高度与拉伸高度无论怎么设置好像都不是想要的值。

问题原因：

polygon的与高度2个参数具体含义是：

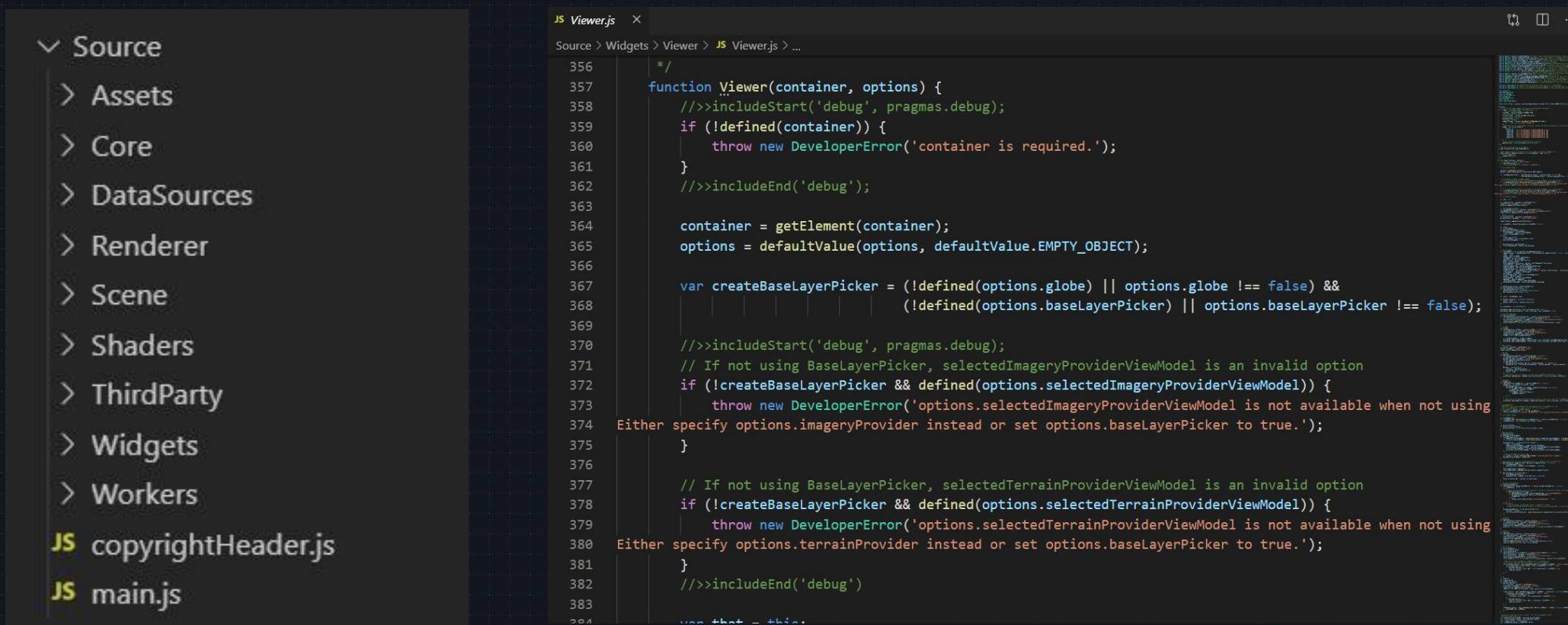
- **height**是指polygon距离地面的高度
- **extrudedHeight**是指polygon拉伸后的面距离地面的拉伸高度

只有当**extrudedHeight**大于**height**时才会呈现挤出高度的效果，且polygon的厚度就是两者的差值。



2.10 Cesium 底层源码的修改和打包

Cesium学习到了一定程度后，肯定是要进入下一阶段，读Cesium底层源码和修改编源码并重新打包Cesium。最后在此之前先掌握：git、npm、gulp等现代web前端知识。



The screenshot shows a code editor interface with a sidebar on the left containing a tree view of Cesium source files. The main area displays the `Viewer.js` file. The code is written in JavaScript and defines the `Viewer` constructor function. The function takes a `container` and `options` parameter. It includes logic to handle `options.globe`, `options.baseLayerPicker`, and `options.selectedImageryProviderViewModel`. It also handles `options.selectedTerrainProviderViewModel`. The code uses comments to indicate when certain options are invalid if not used with specific挑子.

```
JS Viewer.js ×
Source > Widgets > Viewer > JS Viewer.js > ...
356     */
357     function Viewer(container, options) {
358         //>>includeStart('debug', pragmas.debug);
359         if (!defined(container)) {
360             throw new DeveloperError('container is required.');
361         }
362         //>>includeEnd('debug');
363
364         container = getElement(container);
365         options = defaultValue(options, defaultValue.EMPTY_OBJECT);
366
367         var createBaseLayerPicker = (!defined(options.globe) || options.globe !== false) &&
368             (!defined(options.baseLayerPicker) || options.baseLayerPicker !== false);
369
370         //>>includeStart('debug', pragmas.debug);
371         // If not using BaseLayerPicker, selectedImageryProviderViewModel is an invalid option
372         if (!createBaseLayerPicker && defined(options.selectedImageryProviderViewModel)) {
373             throw new DeveloperError('options.selectedImageryProviderViewModel is not available when not using
374 Either specify options.imageryProvider instead or set options.baseLayerPicker to true.');
375         }
376
377         // If not using BaseLayerPicker, selectedTerrainProviderViewModel is an invalid option
378         if (!createBaseLayerPicker && defined(options.selectedTerrainProviderViewModel)) {
379             throw new DeveloperError('options.selectedTerrainProviderViewModel is not available when not using
380 Either specify options.terrainProvider instead or set options.baseLayerPicker to true.');
381         }
382         //>>includeEnd('debug')
383
384         var that = this;
```

2.10 Cesium Git仓库最新代码下载

除了已经发布好的Releases版本包外，当我们有修改Cesium源码的需求时，就需要从github直接下载最新源码后进行自行编译打包（您也可以下载对应发布版本的源码包）

Github仓库地址：<https://github.com/CesiumGS/cesium>

如果您电脑有git客户端可直接输入git命令clone代码；如果没可以直接按下图所示下载zip包；

The image displays two screenshots illustrating how to download the latest Cesium code. The left screenshot shows a terminal window on a Windows system (MINGW64) executing a git pull command against the 'cesium' repository. The output of the command shows the cloning process, including object enumeration, counting, compressing, and unpacking. The right screenshot shows the GitHub repository page for 'cesium'. It features standard GitHub navigation like 'Explore', repository statistics (96 releases, 184 contributors, Apache-2.0 license), and a 'Clone or download' button. A red arrow points to the 'Clone or download' button, and another red arrow points to the 'Download ZIP' button below it, indicating the two primary ways to obtain the code.

2.10 Cesium 源代码修改及编译



我们以汉化帮助按钮提示信息为例，来解说本节。

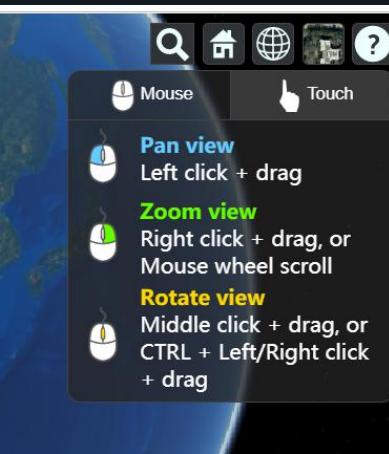
```
Source\Widgets\NavigationHelpButton\NavigationHelpButton.js
```

```
108    options.className = 'cesium-click-navigation-help cesium-navigation-help' + id;
109    options.setAttribute('data-bind', 'css: { "cesium-click-navigation-help": true }');
110    options.innerHTML = '\
111      \
112    > \
113      <td> \
114      <td>\n        <div class="cesium-navigation-help-pan">平移视图</div>\n        <div class="cesium-navigation-help-details">鼠标左键拖动</div>\n      </td>\n    > \
118  > \
119  > \
120  <td> \
121  <td>\n        <div class="cesium-navigation-help-zoom">缩放视图</div>\n        <div class="cesium-navigation-help-details">右键拖动或鼠标中键滚轮</div>\n      </td>\n    > \
124  </td>\n  > \
125  >'
```

1: node

```
[21:08:14] Starting 'release'...
[21:08:14] Starting 'generateStubs'...
[21:08:14] Starting 'build'...
[21:08:17] Finished 'build' after 2.35 s
[21:08:17] Starting 'generateStubs'...
[21:08:17] Finished 'generateStubs' after 214 ms
[21:08:17] Starting 'combine'...
[21:08:17] Finished 'combine' after 2.56 s
[21:10:00] Starting 'minifyRelease'...
[21:10:00] Finished 'minifyRelease' after 1.72 min
```

行 1, 列 1 空格: 4 UTF-8 with BOM CRLF JavaScript Go Live



- 打开文件
Source\Widgets\NavigationHelpButton\NavigationHelpButton.js
- 找到代码中英文提示信息改为对应的中文后保存，另存为utf-8格式覆盖原文件
- 运行npm run release
- 编译完成后拷贝编译好的Build/Cesium目录到项目中使用

3. 数据处理及服务发布

- 地图底图瓦片数据
- terrain地形数据
- glTF 小模型处理
- 3D Tiles模型处理
- 服务发布(nginx)

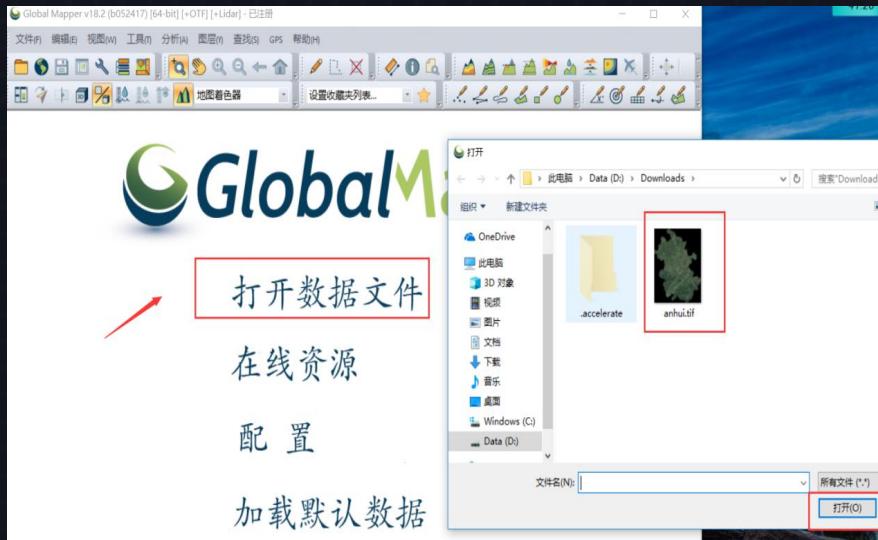
3.1 数据处理解决方案

我们可提供数据处理相关的第三方处理工具（免费或开源）及操作手册。

数据处理解决方案		源格式	Cesium格式	备注
地图底图	卫星影像图处理及加载	tif,img,jpg	瓦片(jpg或png)	免费
	在线地图的下载和内网部署	瓦片		免费
地形	地形数据处理	tif,dem	terrain	免费
glTF 小模型	小模型转换glTF处理	obj、dae	glTF	开源
3DTiles 模型	人工模型数据处理	obj、dae	3D Tiles	免费
	倾斜摄影模型数据处理	osgb		免费
	城市建筑物白膜数据处理	shp		免费
	点云数据处理	las,ply		免费
	BIM模型数据处理	rvt、dgn		收费

3.2 地图底图：卫星影像图处理及加载

当前有很多软件都可以做切片处理，比如ArcGIS、GeoServer、Global Mapper等，此次我们以开源免费的Global Mapper工具软件做介绍，如果您熟悉其他软件平台，也可以使用您说熟悉的工具进行处理。



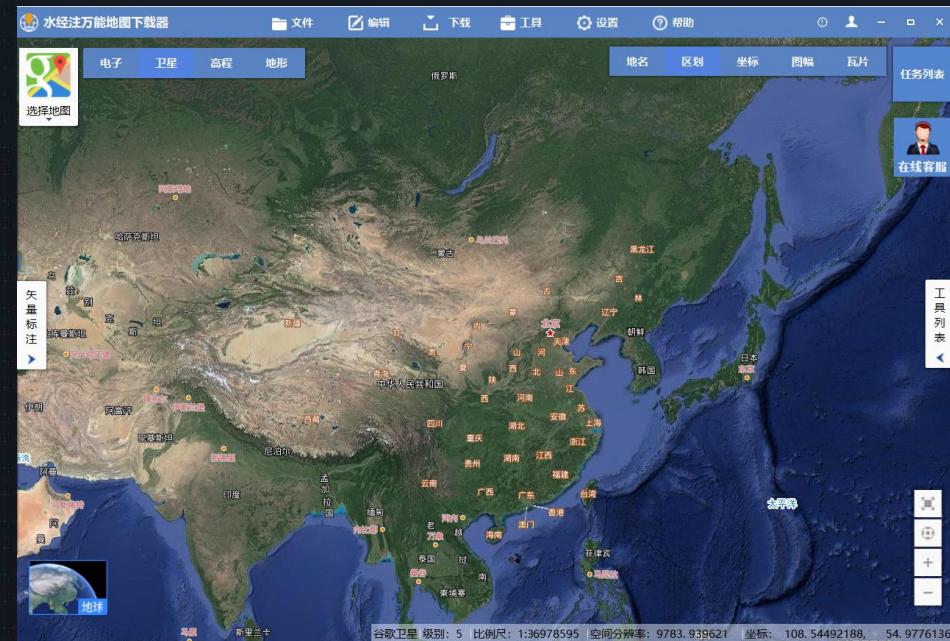
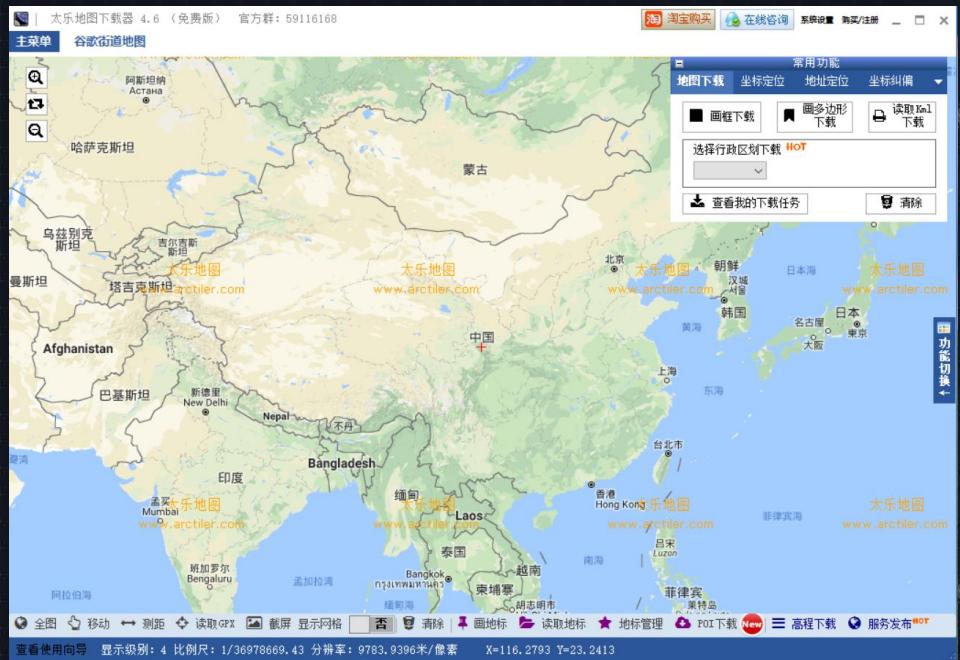
处理及使用流程：

- 导入tif影像图源文件
- 生成xyz金字塔瓦片
- 瓦片移动至mapdata目录下（已nginx发布）
- 客户端调用加载

相关软件和详细操作步骤文档见附件《A1-卫星影像图处理及加载》

3.2 地图底图：在线地图的下载和内网部署

当前互联网地图基本都是xyz的金字塔瓦片数据，所以很方便的进行爬虫抓取下载，目前市场上有很多相关下载工具：Mobile Atlas Creator、太乐地图下载器、水经注万能地图下载器等。按各工具平台使用说明下载导出xyz标准金字塔格式瓦片即可使用。



相关软件和详细操作步骤文档见附件《A2-在线地图的下载和内网部署》

3.3 地形数据处理

根据tif格式的DEM（数字高程模型）地形文件，生成Cesium可用的离线terrain格式地形瓦片数据，目前可以通过3个工具进行处理。

- CTB地形处理工具
- GDAL地形处理工具
- Cesiumlab处理工具

还有一种方式是类似上一节在线地图下载，通过“在线地形下载器”来下载在线的服务数据。

相关软件和详细操作步骤文档见附件《B1-地形数据处理》

3.4 glTF小模型转换处理



需要将3dmax、Maya等建模软件建好的三维模型导出为dae或obj数据，再通过dae和obj格式转换为gltf，转换方式有以下方式：

- 在线转换：<https://cesiumjs.org/convertModel/>
- dae转gltf转换：<https://github.com/KhronosGroup/COLLADA2GLTF>
- obj转gltf转换：<https://github.com/AnalyticalGraphicsInc/obj2gltf>
或 <https://github.com/PrincessGod/objTo3d-tiles>
或 <https://github.com/fanvanzh/3DTiles>
- gltf内部优化处理：<https://github.com/AnalyticalGraphicsInc/gltf-pipeline>



相关软件和详细操作步骤文档见附件《C1-小模型转换gltf处理》

3.5 3DTiles三维模型数据处理（1）

Cesium对数据处理工具没特别要求，生产的数据格式满足Cesium要求的3DTiles格式即可。

3DTiles格式规范：<https://github.com/AnalyticalGraphicsInc/3d-tiles>



- **城市建筑白膜：**在拥有如shp、kml等格式的建筑物二维面边界坐标数据，和高度或层数属性信息，再通过工具转换为三维立体的白膜建筑物3DTiles模型。
- **倾斜摄影：**一般是无人机拍摄，拍摄的数据通过建模工具ContextCapture Cente软件可以直接导出cesium 3DTiles格式。也可以通过osgb通用格式转3DTiles格式后在平台中使用。
- **点云数据：**一般是激光扫描后产生的数据，有las、pts、ply等格式。
- **人工建模：**数据来源于3dmax、Maya等建模软件建模，建好的三维模型导出为dae和obj数据后，再转换为3DTiles数据格式。
- **BIM模型：**数据来源于专业的BIM软件，常见的有rvt和dgn格式，目前市面没有免费软件，都是收费为主。

3.5 3DTiles三维模型数据处理 (2)



3.6 数据处理总结

目前市场的数据处理工具或解决方案主要来源于以下：

- Github 开源的小工具
- Cesium 官方推出的 Cesium Ion 在线平台（对国内企业来说有点鸡肋）
- 一些商家推出的工具集（如cesiumlab）

3.7 三维数据的服务发布

前面我们讲解的几类数据都是静态资源数据，可以直接通过http服务的方式进行发布，无需专业的GIS服务器，实际就算是GIS服务的底层最终也是静态资源来访问。

比如我们将相关所有数据放在mapdata目录，并通过我们推荐的nginx来发布为服务，

分类	目录名称	说明
三维模型	3dtiles	3DTiles格式三维模型
地形数据	terrain	地形数据
瓦片底图	wgs3857imgD	卫星影像(无注记) 瓦片数据
	wgs3857imgZ	卫星注记 瓦片数据

3.7 服务发布：nginx发布（推荐）

- 进行已经安装好的nginx目录（参看前面开发环境搭建章节讲解）
- 下载地址：<http://data.marsgis.cn/dataproce/nginx.rar>
- 参考我们提供的 **nginx.conf** 配置文件进行修改或直接覆盖配置（有允许跨域配置）
- 终端输入命令 `nginx.exe -s reload` 使修改配置生效
- 浏览器输入相关URL来校验服务是否正常

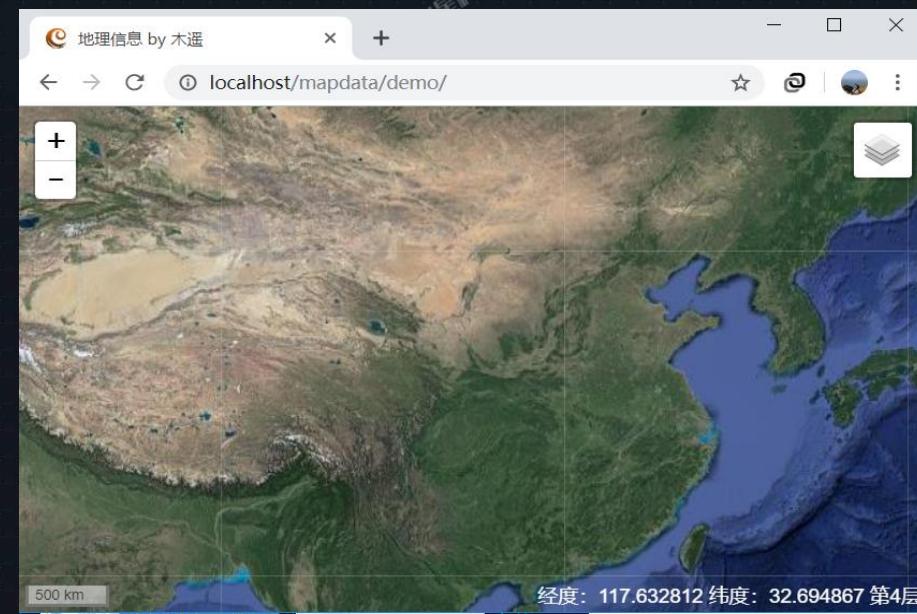
```
root    html;
index   index.html index.htm;

增加mapdata跳转节点配置

#地图数据
location /mapdata {
    include nginx_cors; ←此行是允许跨域
    root D:\\;
    index  index.html index.htm;

    expires 30d;
}
```

提供文件：
`nginx.conf`
`nginx_cors`



3.7 服务发布：非nginx发布时的常见问题

如果您未使用我们推荐的nginx发布服务，使用IIS、Tomcat等其他http服务容器发布时，要注意下面2个问题。

- 1、需要开启允许跨域配置（Access-Control-Allow-Origin *）
- 2、相关文件类型的mimeMap配置（.terrain .b3dm .gltf）

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <configuration>
3   <system.webServer>
4     <!-- 跨域配置 -->
5     <httpProtocol>
6       <customHeaders>
7         <add name="Access-Control-Allow-Origin" value="*"/>
8         <add name="Access-Control-Allow-Headers" value="Content-Type"/>
9         <add name="Access-Control-Allow-Methods" value="GET, POST, PUT, DELETE, OPTIONS"/>
10    </customHeaders>
11  </httpProtocol>
12
13  <!-- 文件类型识别配置 -->
14  <staticContent>
15    <mimeMap fileExtension=".topojson" mimeType="text/json" />
16    <mimeMap fileExtension=".terrain" mimeType="application/octet-stream" />
17    <mimeMap fileExtension=".gltf" mimeType="application/octet-stream" />
18    <mimeMap fileExtension=".glsl" mimeType="application/octet-stream" />
19    <mimeMap fileExtension=".b3dm" mimeType="application/octet-stream" />
20    <mimeMap fileExtension=".s3m" mimeType="application/octet-stream" />
21    <mimeMap fileExtension=".glb" mimeType="application/octet-stream" />
22    <mimeMap fileExtension=".kml" mimeType="text/xml" />
23    <mimeMap fileExtension=".czml" mimeType="text/json" />
24
25
26    <mimeMap fileExtension=".shp" mimeType="application/octet-stream" />
27    <mimeMap fileExtension=".dbf" mimeType="application/octet-stream" />
28    <mimeMap fileExtension=".prj" mimeType="application/octet-stream" />
29  </staticContent>
30
31 </system.webServer>
32 </configuration>
```

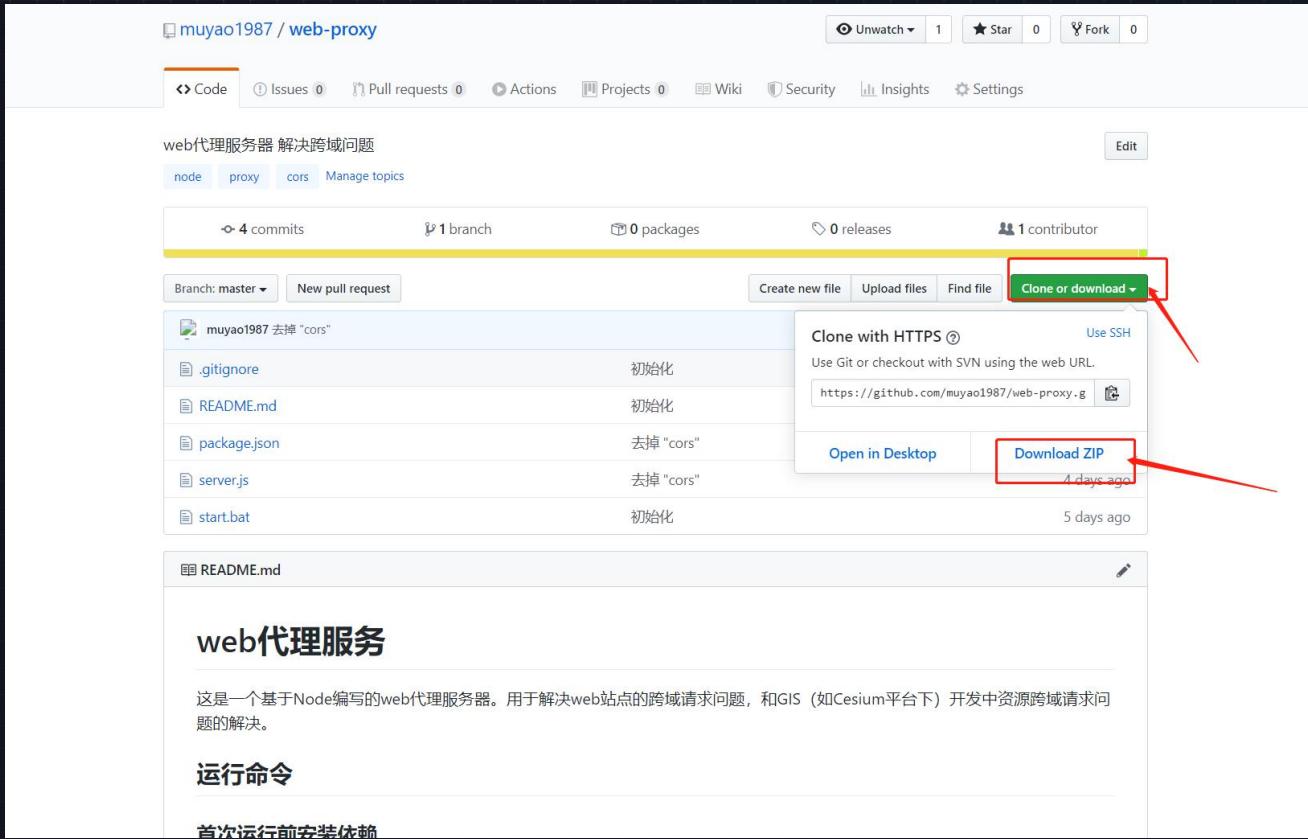
IIS 的web.config配置

```
Java: Tomcat的web.xml配置
<mime-mapping>
  <extension>json</extension>
  <mime-type>application/json</mime-type>
</mime-mapping>
<mime-mapping>
  <extension>terrain</extension>
  <mime-type>application/octet-stream</mime-type>
</mime-mapping>
<mime-mapping>
  <extension>b3dm</extension>
  <mime-type>application/octet-stream</mime-type>
</mime-mapping>
<mime-mapping>
  <extension>gltf</extension>
  <mime-type>application/octet-stream</mime-type>
</mime-mapping>
<mime-mapping>
  <extension>glb</extension>
  <mime-type>application/octet-stream</mime-type>
</mime-mapping>
<mime-mapping>
  <extension>s3m</extension>
  <mime-type>application/octet-stream</mime-type>
</mime-mapping>
```

3.8 第三方服务的跨域问题解决

如果存在一些服务不允许跨域访问，并且服务不是我们可以控制的第3方发布的。那么我们可以使用代理站点进行中转访问。

下载地址：<https://github.com/muyao1987/web-proxy>



3.8 GeoServer服务：发布矢量数据等



通常我们利用GeoServer 可以方便的发布一些矢量数据或卫星底图数据。发布的服务类型有:WMS、WMPS、WFS服务等，Cesium支持加载这些OGC标准的服务。

The screenshot shows the GeoServer interface with the 'Layer Preview' page open. The left sidebar has links for '关于和状态' (About and Status), '关于 GeoServer' (About GeoServer), '数据' (Data), 'Layer Preview' (selected), and '演示' (Demonstrations). The main content area is titled 'Layer Preview' and displays a table of configured layers. The table has columns for Type, Title, Name, Common Formats (OpenLayers, KML, GML), and All Formats (a dropdown menu). There are 7 items listed:

Type	Title	Name	Common Formats	All Formats
线	hfbj	mars:hf bj	OpenLayers KML GML	Select one ▾
线	hfdl	mars:hfdl	OpenLayers KML GML	Select one ▾
面	hfgh	mars:hf gh	OpenLayers KML GML	Select one ▾
点	hfjy	mars:hf jy	OpenLayers KML GML	Select one ▾
面	hfjzw	mars:hf jzw	OpenLayers KML GML	Select one ▾
点	hfty	mars:hfty	OpenLayers KML GML	Select one ▾
面	stqw	mars:stqw	OpenLayers KML GML	Select one ▾

4、Mars3D概览及架构

- Mars3D是什么
- Mars3D产品特点
- Mars3D的优势

4.1 Mars3D是什么?

Mars3D三维地球平台软件 是火星科技研发的一款基于 WebGL 技术实现的三维客户端开发平台，基于Cesium 优化提升与B/S架构设计，支持多行业扩展的轻量级高效能GIS开发平台，能够免安装、无插件地在浏览器中高效运行，并可快速接入与使用多种GIS数据和三维模型，呈现三维空间的可视化，完成平台在不同行业的灵活应用。



The diagram illustrates five key features of the Mars3D platform:

- 开源产品最佳实践**: Integrated Cesium, ArcGIS Server, Truf, ECharts, MapV, bootstrap map libraries.
- 设计合理简单易用**: Flexible internal architecture, reuse of components, and support for third-party services.
- 敏捷开发可复用**: JSON configuration, component-based development, and support for various map services.
- 全面完整的解决方案**: Comprehensive learning resources, API documentation, and developer guides.
- Morden WebGIS**: Based on modern web technologies like H5, WebSocket, ES6, NodeJS.

4.2 Mars3D 产品特点

1. 源码可控+平台级应用

提供所有无加密混淆的源代码，安全全程可控，可放心使用。并且完善优化了Cesium等开源库的基础功能，解决了开源产品不完善的通用弊端，填了各种“坑”，达到平台级应用。

2. 设计合理+API文档全

API 接口采用一致性的风格习惯精细化设计，API文档丰富完整，使用一目了然，容易学习，优秀的设计架构，可节省开发人员的大量学习时间和精力；提供的教程都具有很强的可读性、自学习性。平台每个功能都提供了示例教程和源码，只需要初级水平人员即可上手使用。

3. “零”客户端+多端支持

Mars3D是针对Web端的轻量级3D GIS展示端，用户可以不加载任何插件，在浏览器中对高精度的可交互的三维地形场景进行浏览、分析等操作。支持跨平台和跨浏览器，实现了三维展示、交互的“零”客户端。

4. 场景参数化+可复用

平台对地图的初始化进行了简化封装，通过传入配置参数即可快捷的进行地图三维场景的初始化，场景参数和平台内数据均可导出json数据，不同项目中我们只需加载不同场景参数即可快速创建三维地球场景。

5. 功能多 + 完整解决方案

包含300+的功能示例，并可扩展；提供不同技术栈的项目脚手架和可直接项目应用的多个项目模板，模块化设计可复用，并支持各种配置，可以快速搭建和敏捷开发，修改后可快速项目交付；提供全面的平台配套资料，包括功能示例API文档、开发教程、培训手册和视频。我们提供的是一整套的从0到1的三维WebGIS开发的完整解决方案。

4.3 Mars3D 的优势

1. 功能丰富+简单易用

提供300+功能案例，功能上其他厂商拥有的功能我们均具备，并且我们示例更加全面。此外，我们也可以为客户量身定制，结合客户的行业特点和实际需求，提供二三维地图和大数据可视化等领域全方位的行业解决方案；功能示例和API 接口采用一致性的风格习惯精细化设计，文档丰富全面，使用一目了然，简单易用；

2. 开放开源+无限制使用

平台已在Github开源，可以免费无限制使用，没有任何限制、可多次复用；平台也支持付费提供所有源代码，目前在全国来说也是唯一的本着如此开放开源态度的GIS厂商。我们的报价也是明码标价、官网在线公开透明的方式。

3. 成熟稳定+实力强

我们是全国第一批最早使用Cesium的团队，在全国走在最前列，技术圈内知名度高，模仿者众多；我们平台应用时间久，客户已超过300多家、应用的项目广，产品成熟。火星科技是一家国家级高新技术企业，研发团队成员多为中科大、合工大、安大的高技术人才，多数成员拥有十年以上研发经验、拥有军工、公安、政府从业背景，开发经验丰富；

4. 服务完善+完整方案

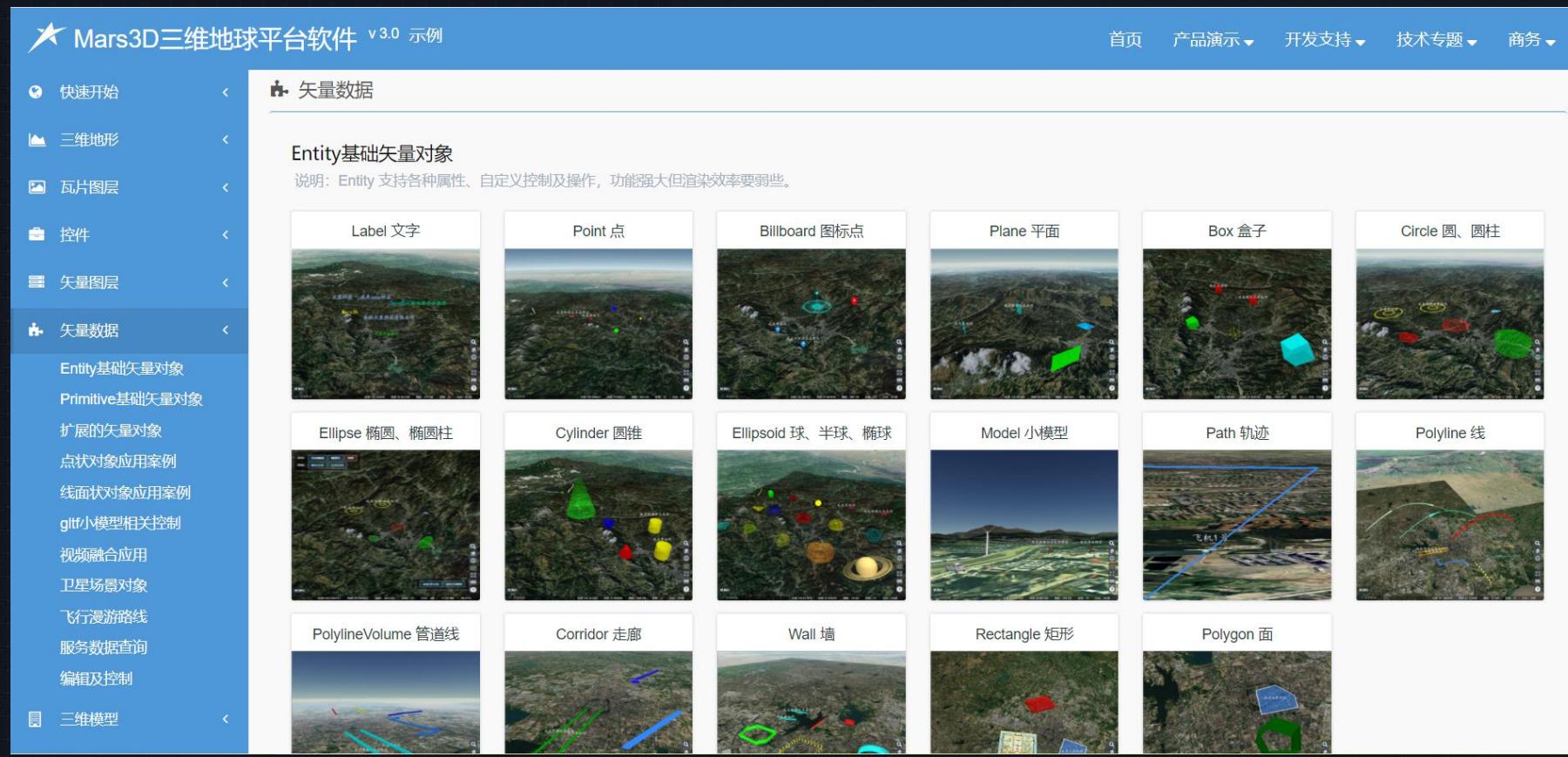
提供完善的售后技术服务和指导咨询服务，我们不仅仅只提供源代码，我们还拥有资深技术人员为您保驾护航。公司重视客户服务，本着服务至上的原则，提供线上、电话等多种方式的技术支持、咨询服务。

我们为您提供的是一个完整的三维地球开发解决方案，让贵公司能节省时间、精力，快速掌握和应用三维可视化。

4.4 Mars3D 平台在线体验

Mars3D官网: <http://mars3d.cn> (含功能示例源码、API文档)

Github地址: <https://github.com/marsgis/mars3d>



The screenshot shows the Mars3D 3D Earth Platform Software v3.0 Examples page. The left sidebar contains a navigation menu with categories like '快速开始', '三维地形', '瓦片图层', '控件', '矢量图层', '矢量数据', 'Entity基础矢量对象', 'Primitive基础矢量对象', '扩展的矢量对象', '点状对象应用案例', '线面状对象应用案例', 'glTF小模型相关控制', '视频融合应用', '卫星场景对象', '飞行漫游路线', '服务数据查询', '编辑及控制', and '三维模型'. The main content area is titled '矢量数据' and displays various examples of vector data entities. Each example includes a title and a preview image showing the entity in a 3D terrain environment. The examples are categorized into two sections: 'Entity基础矢量对象' (Label 文字, Point 点, Billboard 图标点, Plane 平面, Box 盒子, Circle 圆、圆柱) and 'Primitive基础矢量对象' (Ellipse 椭圆、椭圆柱, Cylinder 圆锥, Ellipsoid 球、半球、椭球, Model 小模型, Path 轨迹, Polyline 线). Below these are other examples: PolylineVolume 管道线, Corridor 走廊, Wall 墙, Rectangle 矩形, and Polygon 面.



THAKY U



我 们 的 征 途 在 星 辰 大 海

关于我们



火星科技创建于2017年8月，立足于科教之城、创新高地安徽省合肥市，公司致力于GIS地理信息产业和Web三维可视化技术发展和技术普惠，为客户提供地图、三维地球、全景漫游、大数据可视化等领域全方位的行业解决方案。

公司以二三维一体化GIS技术为基础框架，进一步拓展二三维一体化数据模型，融合倾斜摄影、BIM、激光点云等多源异构数据，推动三维GIS实现室外室内一体化、宏观微观一体化与空天/地表/地下一体化，赋能全空间的三维GIS应用。

公司目前已有“MarsGIS for Leaflet 地图平台软件”和“MarsGIS for Cesium三维地球平台软件”等GIS开发框架软件产品。

 0551-65883001

 <http://marsgis.cn>

 合肥市高新区望江西路539号
鲲鹏产业园6号楼