目录

简介 1.1

基本概念 1.2

开发者账号申请 1.3

GraphQL API 1.4

- 三维重建 1.4.1
- 上传图像 1.4.2
- API 文档 1.4.3
- API 更新日志 1.4.4

JavaScript SDK 1.5

- 演示应用 1.5.1
- 常见问题 1.5.2
- SDK 文档 1.5.3
- SDK 更新日志 1.5.4

Altizure 开发平台

欢迎使用 Altizure！

在 Altizure 您除了可以使用全球领先的从图像到三维模型的自动建模服务，还可以通过 Altizure 在线浏览和应用技术，轻松分享和使用您的三维模型，为您带来无限乐趣和价值。

加入 Altizure 开发平台，可以把 Altizure 提供的建模浏览服务与您的业务做更深入的集成，释放三维数据带来的无限可能。

让我们开始实景三维世界的探险吧。

- 基本概念
- 开发者账号申请
- GraphQL API
  - 三维重建
  - 上传图像
  - API 文档
  - API 更新日志
- JavaScript SDK
  - 演示应用
  - 常见问题
  - SDK 文档
  - SDK 更新日志

更多了解 Altizure 欢迎访问：

- 探索 Altizure 的三维世界：altizure.cn/explore
- 关注微信公众号：altizure_info
- 关注官方博客：blog.altizure.cn
- 离线文档：pdf, epub

联系我们 support@altizure.com

该文档最后修改于 Fri Aug 10 2018 18:30:59 GMT+0800 (HKT)

基本概念

在开始使用 Altizure 开发平台之前，您需要大概了解以下概念。您如果使用过一些其他开发平台，例如微信开放平台、支付宝开放平台，这些概念都是稀松平常的，您可以轻易上手。

1. 开发者账号

开发者账号需要您先有 Altizure 账号。

如果您还没有一个 Altizure 账号，请马上在注册页面免费申请一个。

然后您需要按照开发者账号申请指示，把它升级为开发者账号。所有开发的权限都会和这个账号关联。请确保这个账号的安全和保密。

2. 应用 (App)

应用是您在 Altizure 开发平台上创建的最基本应用管理单元。每个应用都有相应的权限级别，安全设置，唯一的 ID，令牌和鉴权密码。

一个开发者账号，可以申请多个应用，来区分不同的业务和流程。

应用令牌 (App key)

每个应用唯一的令牌，用于识别开发平台的服务调用。用户在授权相关应用访问其 Altizure 平台上的信息后，可以通过该令牌识别相关应用，终止相关授权。

应用密钥 (App secret)

用于服务器端调用时鉴别这个 API 调用是否来自相应的开发者账号。切勿在任何前端代码里泄露这个密钥。只能在服务器端程序内使用。

3. 用户令牌 (User token)

这个令牌是应用在需要获得用户授权时得到的鉴权令牌，用于访问与用户授权相关的 Altizure 上的信息。开发者可以存储这个令牌，而不应该以任何形式诱导用户提供密码或者存储用户的密码。

4. GraphQL API

GraphQL API 是一组以 GraphQL 为协议提供的在线 API。这套 API，提供了获取和修改 Altizure 上基础数据的能力。详情可以浏览：GraphQL API

5. JavaScript SDK

这是一个以 JavaScript 为开发语言的 SDK。提供了丰富的三维浏览和编辑的功能。详情可以浏览：JavaScript SDK

该文档最后修改于 Fri Aug 10 2018 18:30:59 GMT+0800 (HKT)

开发者账号申请

目前开发者账号仅对公司用户开放注册。您如果有兴趣，请前往我们的开发平台网站 developers.altizure.cn 或 developers.altizure.com 申请。

跟随以下教程，您就可以顺利完成开发者账号的申请。

1. Altizure 账号

1.1 申请 Altizure 账号

申请开发者账号需要您先拥有 Altizure 账号， Altizure 账号提供了精准快速的自动建模功能，即使是免费版 Altizure 账号也一定会让您爱不释手。只需花费几分钟，我们有信心彻底颠覆您对三维的理解。

如果您已经拥有了 Altizure 账号，请直接看本章第二部分 `2. Altizure 开发者账号` 。

登陆或注册您的 Altizure 账号。

如果您还没有注册 Altizure 账号，赶快点击 “建立新账号”，加入 Altizure 大家庭，享受实景三维带来的乐趣。

先登入到 Altizure 再去 Altizure Developer
电邮
密码
忘记密码?
登入
没有帐户? 建立新帐户

当然，您注册成功后还要授权给Altizure Developer，才可以进行下一步的操作。

1.2 验证注册邮箱

为了方便我们给您提供更加强大的技术支持，一个有效的邮箱也是不可或缺的。

一封验证邮件将发送到您注册时所使用的邮箱中。

小提示：如果没有收到验证邮件，请检查一下您的垃圾箱、已删除或者广告邮件。

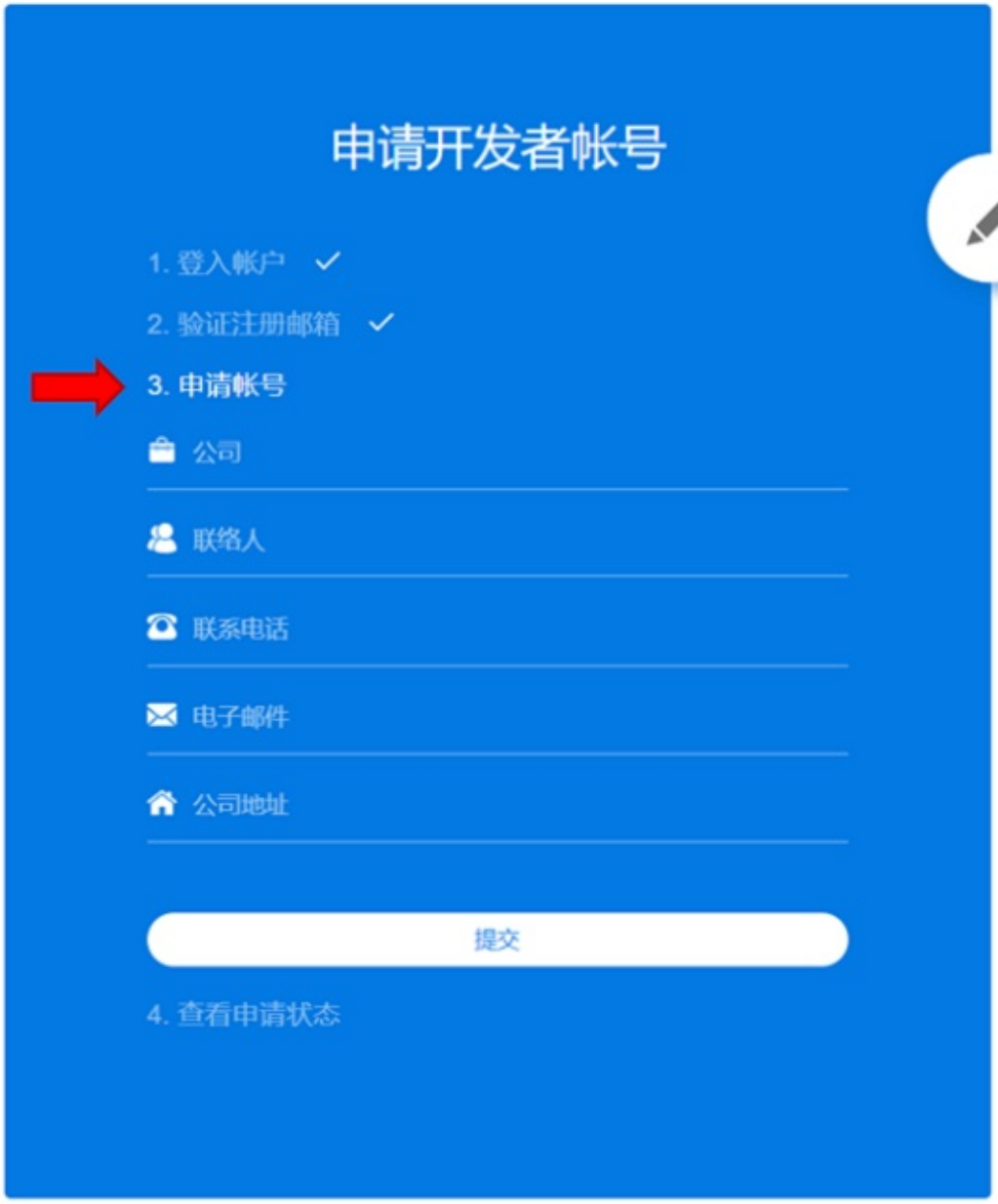
看到这个界面，说明您的验证邮箱已经通过啦。

2. Altizure 开发者账号

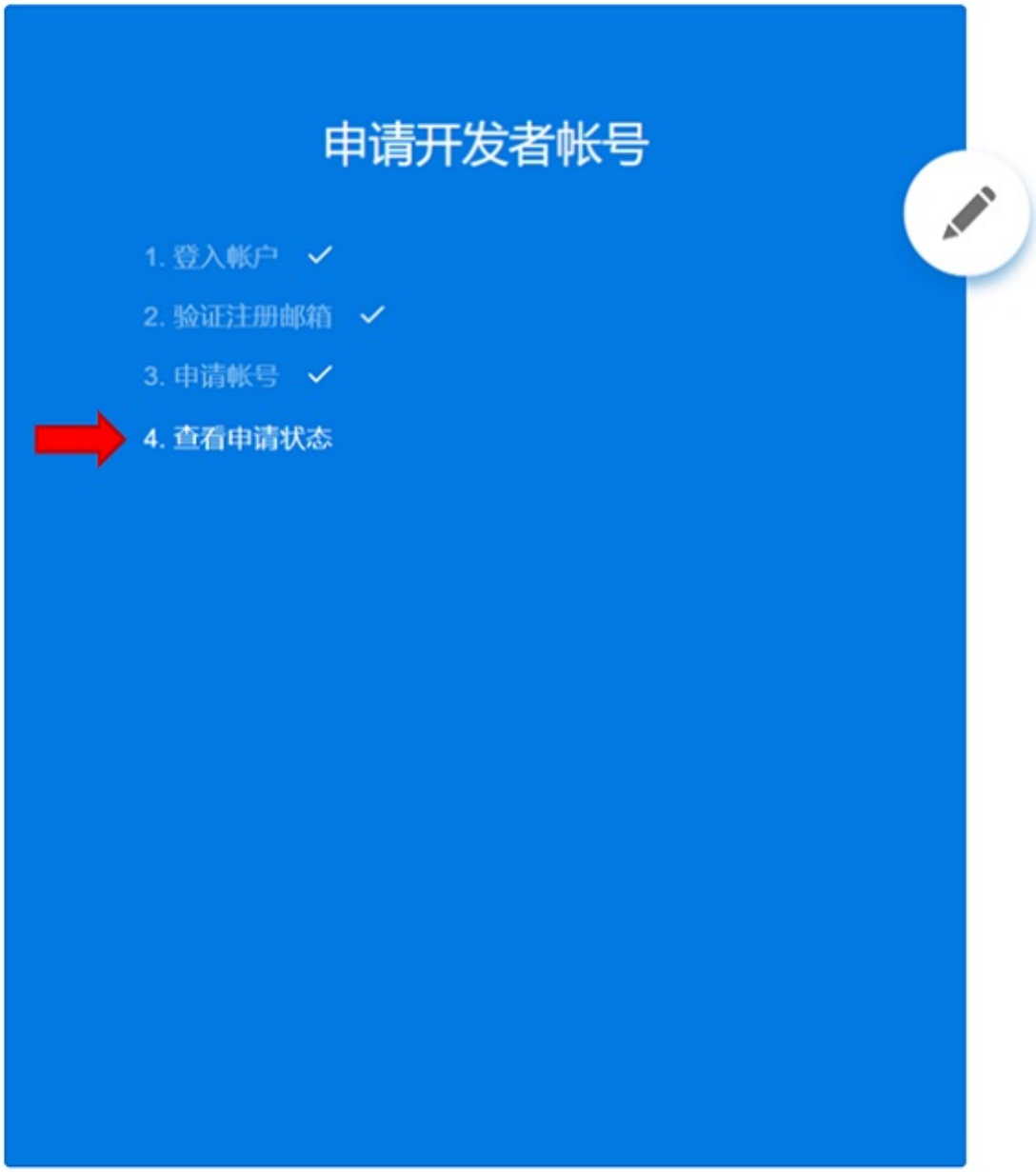
2.1 申请开发者账号

目前开发者账号只针对企业用户，如果您想在 Altizure 的基础上开发更加个性化的功能，欢迎与我们的客服进行联系 developers@altizure.com

在这里填入您的公司信息，以加快开发者账号的审核速度。


申请开发者帐号
1. 登入帐户
2. 验证注册邮箱
3. 申请帐号
公司
联络人
联系电话
电子邮件
公司地址
提交
4. 查看申请状态

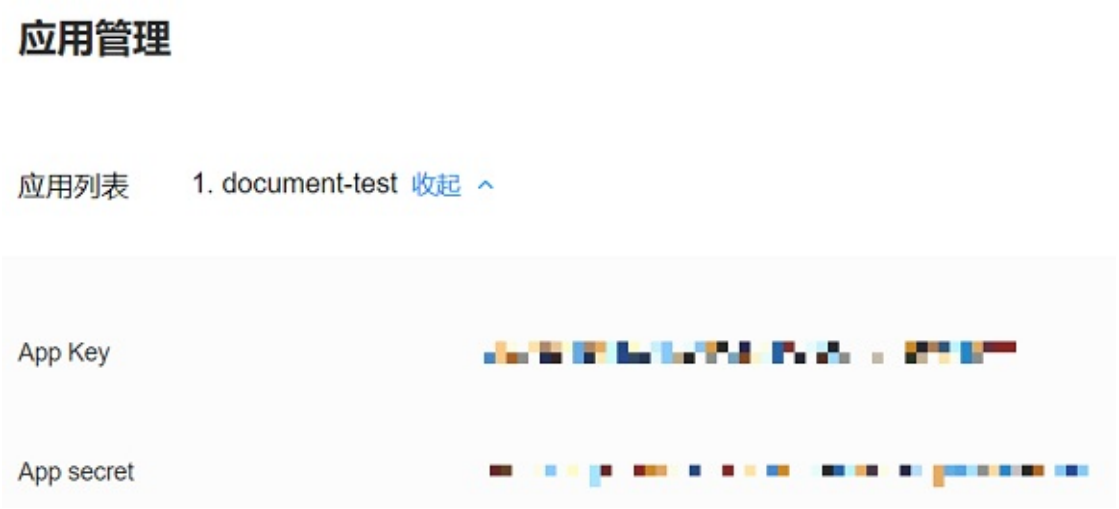
2.2 等待审核通过



提交信息后，我们将在5个工作日内回复您，并欢迎您随时与客服联系以获得最新审核进度。

当您的账号状态变为“通过审核”时，说明您已获得了开发者的所有权限，您可以添加应用并获得开发所必须的“APP Key”，在此之后，您就可以在 Altizure 的世界中尽情发挥想象力了。

在这里，您可以创建您的 APP 并得到您的 APP Key。



欢迎随时与我们的客服进行联系：developers@altizure.com

关于最新的开发者信息，请关注我们的社交媒体和官方博客：

- 微信公众号：altizure_info
- 中文官方博客：blog.altizure.cn
- Altizure 开发平台: developers.altizure.cn

该文档最后修改于 Mon Aug 20 2018 17:30:20 GMT+0800 (HKT)

Altizure Graphql API

GraphQL API 是一组以 GraphQL 为协议提供的在线 API。这套 API 提供了获取和修改 Altizure 上基础数据的能力。

您如果在寻找如何快速加载和渲染 **Altizure** 上的三维数据，请查看我们的 **JavaScript SDK** 页面，可以方便您快速上手。

1. 使用前的准备

- Altizure 开发者账号 （必须）（参考开发者账号）
- 应用令牌 (App Key) （必须）（参考开发者账号 2.2）
- 用户令牌 (User token) （部分调用需要）

API 调用效果和文档的入口，您可以在这里查询 API 的具体用法和使用效果：

- 国际站 api.altizure.com/graphql 。
- 中国站 api.altizure.cn/graphql 。

2. 在浏览器中测试 API

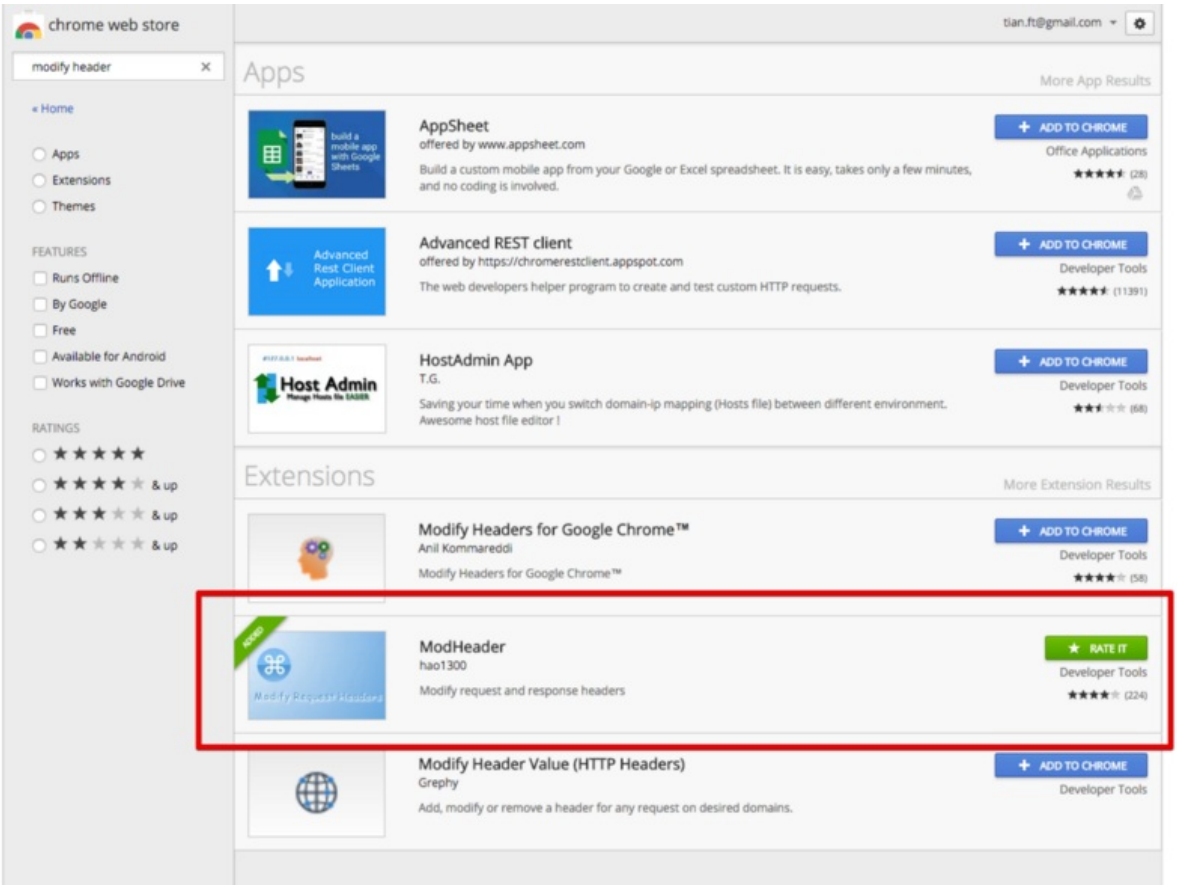
您可以在以上站点中直接执行查询语句来察看查询的效果，部分查询方法需要您使用应用令牌 (App Key)和用户令牌 (User token) ，以下将介绍如何通过修改浏览器的 http header 使用您自己的应用令牌和用户令牌来进行查询。

在浏览器中测试 API，可以在浏览文档的同时对效果进行测试，提高开发效率。确定数据正确后，再把相关的调用复制到代码中集成，过程轻松简单。

这里的教程以谷歌 Chrome 浏览器为例子，其他支持插件的浏览器也可以用类似的方法进行测试。

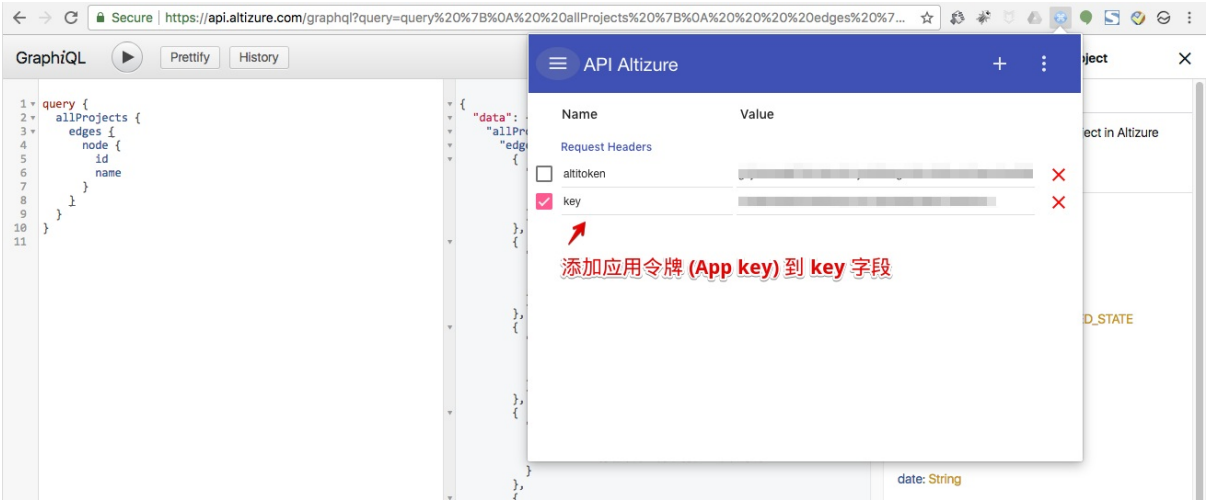
2.1 安装插件

安装一个可以修改 http 访问请求 header 的插件。这里我们以 ModHeader 为例。在谷歌 Chrome 扩展商店里面搜索 `ModHeader` ，并点击安装即可。



2.2 设置插件修改 **http header**

在 http request header 中，添加 `key` 字段，填入应用令牌 (App Key)。 有的查询和调用也需要添加 `altitoken` 字段，填入用户令牌 (User token)。



设置完成后访问 api.altizure.cn/graphql，可以看到以下界面，集成了“查询区”，“查询结果”和“文档”三个模块。

试着在查询区输入一下字段查询一个组图像的 GP 数。

```
query {
  utility {
    sizeToGigaPixel(width: 4000, height: 3000, numImg: 100)
  }
}
```

按查询按钮，便可获得查询结果。在这个样例中 API 调用会计算 100 张 4000×3000 的图像的总 GP 数。

3. 在代码中调用 API

您可以使用任何可以发起 http post request 的程序库来调用 GraphQL API。

例如使用 *JQurey in JavaScript* 及如下代码调用 API：

JQurey 的安装和在 Node.js 调用。

```
$.ajax({
    type: 'POST',
    url: 'https://api.altizure.cn/graphql',
    headers: {
      altitoken: '用户令牌',
      key: '应用令牌'
    },
    data: 'query=' + 'GraphQL的查询字符串（即上文中的 query 部分）'
  })
```

4. 获取用户令牌

您需要通过 OAuth 2.0 的标准流程来获取用户令牌。

其中调取 OAuth 窗口的链接为:

```
`https://api.altizure.cn/auth/start?client_id=${appKey}&response_type=token&redirect_uri=${redirect_uri}`
```

其中 `appKey` 是您的应用令牌，`redirect_uri` 是您注册该令牌时绑定在上面的域名。

您的应用/网页需要在客户端调出浏览器或者 WebView 来访问这个链接。接下来一个授权窗口会呈献给用户用于输入用户名和密码以便授权您的应用访问该用户的信息。您的应用的说明信息会被现实在这个窗口，便于用户识别开发者的身份。

用户授权之后，登录窗口会重新向到 `redirect_uri`。重新向的链接会包含一个用户令牌 `access_token`。您既可以在客户端页面处理这个令牌，也可以实现一个服务器端的路由在服务器端接收这个令牌。

对于移动应用，`redirect_uri` 应该填入您的应用的识别符 application bundle identifier name (iOS) 或者包名 package name (android)。这些信息都必须在申请应用令牌时提供给 Altizure 以便记录在册。

您可以访问这个网址 here 查看一个极简的登录实现。

5. 常见问题

5.1 国际站和中国站如何选择？

请选择与您链接最快的一个。可以在程序中做些自动判断，如果一个站点掉线了可以切换成另外一个。

5.2 GraphQL API 详细文档在哪儿？

请直接用浏览器参照以上教程访问 api.altizure.cn/graphql，即可直接看到所有查询接口的文档。

6. 了解更多

- 深入学习 GraphQL
- 通过 Altizure JavaScript SDK 实现丰富的三维浏览编辑功能
- 更多 GraphQL 相关工具 Awesome GraphQL

该文档最后修改于 Wed Aug 15 2018 16:30:21 GMT+0800 (HKT)

如何通过 API 调用三维重建引擎

概况

Altizure 的三维重建引擎是通过一个个项目的形式来调用的。开发者需要创建一个项目以启动相关的三维重建运算。每个用户可以创建许多个项目。但每个项目只能由一个用户来拥有。为了获得创建项目的权限，必须让用户登录后授权给开发者的 App 获取用户令牌，才能创建项目。

1. 新建项目

调用 `createProject` 这个 mutation 便可以建立一个新的项目。调用这个 mutation 的时候，必须在 http header 设定应用令牌和用户令牌 (当前操作的用户的用户令牌，不一定是开发者账号的用户令牌)。这个 mutation，接受两个参数，一个是项目名称 (同一个用户的项目可以重名)，另一个是项目类型 (免费或者专业)。无论是创建免费还是专业项目，本身是不需要使用 Alticoin 的。但是专业项目在启动三维重建的时候，必须有足够的 Alticoin 支付相关费用。

如果项目创建成功了，API 会返回一个项目类型的对象，否则返回 `null`。通过这个返回的对象，可以通过里面的 id 成员来获取项目的 ID。这个 ID 非常重要，在所有后续的项目相关的调用中都需要使用。

样例 **GraphQL** 语句

```
mutation {
  createProject(name: "sample project", type: free) {
    id
  }
}
```

2. 上传图像

创建完一个项目后，便可以让用户上传图像。请参考上传图像的教程了解如何通过 API 调用上传图像。 图像上传完成后，三维重建便可以准备开始了。

3. 开始重建

发送 `startReconstructionWithError` 这个 mutation 来启动三维重建任务。这个 GraphQL mutation 需要输入 project id (项目 ID，必填输入) 和 coupon id (优惠券代码，可选输入)。其中项目 ID 是刚才 `createProject` 的 API 调用返回的对象可以查询到，或者可以通过 `my.projects` 来查询。另外的 coupon 可以通过 `my.coupons` 来查询。如果是专业项目，这一步就需要用户账号里有足够的余额或者足够的优惠券才能成功。

如果这个调用成功了，一个新的 task object (任务对象) 会被创建并返回。否则会返回一个错误代码。关于错误代码的详细解释，请参考 API Reference 中 `startReconstructionWithError` 相关的文档。

样例 **GraphQL** 语句

```
mutation {
  startReconstructionWithError(id: "54ed8e5741fbfa3e1967fe9b") {
    error {
      code
    }
    task {
      id
      state
```

```
    }
  }
}
```

4. 查询项目状态和进度

当一个三维重建任务开始后，您便可以通过 `project` 这个查询来查看其运行状态和其他信息。

Last modified at Fri Aug 10 2018 18:30:59 GMT+0800 (HKT)

上传图像

概况

为了获得最佳的上传速度，用于三维重建的图像将直接从客户电脑传输到我们的亚马逊或者阿里云的存储空间里。 传输和存储中，您的图像将被严格保密，其他任何人都不可以读取和修改您的图像。 所以上传开始前需要开发者通过我们的 API 获取一个临时的加密 url (亚马逊 S3) 或者 STS 鉴权令牌 (阿里云 OSS)。因为在 altizure 上每个项目中不允许有同名文件，所以我们的 API 会根据您所上传文件的校验码分配一个唯一的文件名，而不是直接使用原文件的名字作为上传的名字。上传完成后便可以调用启动项目的 API，开始三维重建。

以下我们详细介绍上传的每个步骤。

1. 选择最近的 **bucket**

开发者可以通过 `GeoIPInfo.nearestBuckets` 这个查询来获取当前用户最近的 Bucket 用于上传，以达到最快的上传速度。

2. 计算图像校验码

为了保证图像上传的稳定性和数据的一致性。在上传每张图像前开发者必须

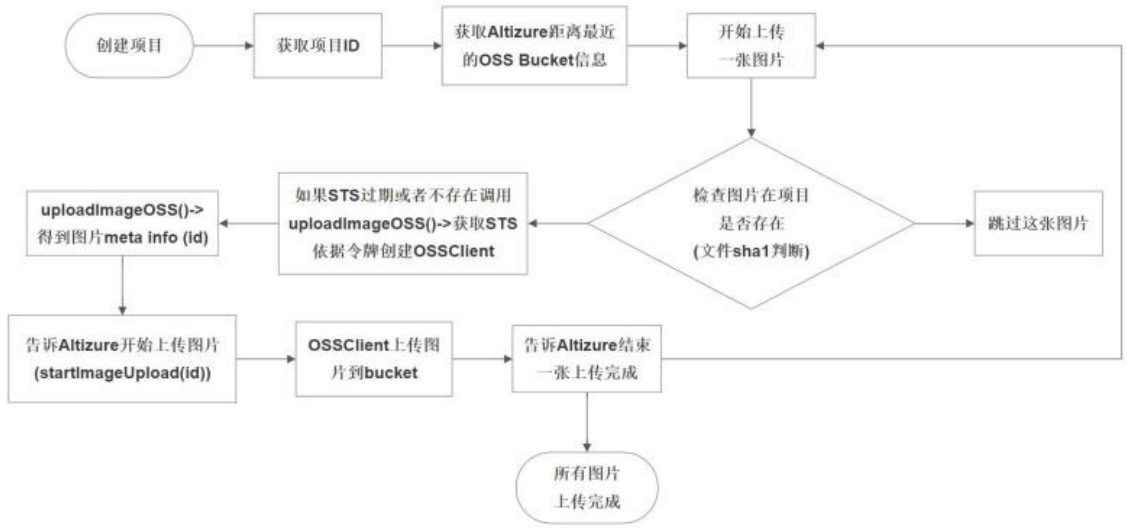
- 计算的图像的 **sha1sum** 校验码
- 调用 mutation `hasImage(pid, sha1sum)` 检查图像是否已经项目 `pid` 里存在。
- 如果图像存在则跳过该图像无需上传；如果图像不存在则上传图像。

3. 上传

3.1 上传到阿里云的 **OSS**

在选择了最近的 OSS bucket 后，我们需要按照以下流程进行上传：

- 通过 `uploadImageOSS(pid, bucket, filename, type, sha1sum)` 获取 **STS** 令牌及其他上传所需的图像 meta info，包括上传 图像的 **id** 和 哈希化的文件名。STS 令牌的有效期只有 1 小时，对相关 bucket 的 `/pid` 只有写权限。如果 STS 令牌过期了，开发者必须利用同样的 mutation `uploadImageOSS(pid, bucket, filename, type, sha1sum)` 进行更新。如果令牌还未过期，开发者在上传新的图像时无需请求新的令牌，只需用同样的 STS 请求 `uploadImageOSS` 中的 `image` 项。虽然开发者可以每上传一张图像都请求新的 STS 令牌，但是由于阿里云的签发令牌的相应很慢每次都请求新的令牌会导致上传不太稳定，所以我们推荐在 STS 令牌未过期前，不要重复申请新的令牌。如果您使用阿里云 OSS JavaScript SDK 中的 `Wrapper`，请注意在 STS 过期前不要重复创建该类型对象，只有在 STS 过期后才需要重新创建。创建过多 `Wrapper` 对象会导致 `The sockets is full` 的错误。
- 获取了 STS 令牌后和相应的 meta image info 后，调用 mutation `startImageUpload(id)` 通知服务器一张图像即将开始上传。
- 使用相应的 STS 令牌把图像上传到 OSS 的 bucket 里 `${pid}/${image.filename}` 的路径下。其中 `${image.filename}` 是上传前调用 `uploadImageOSS` 所返回的图像 meta info。
- 上传完成后调用 mutation `doneImageUpload(id)` 通知服务器，一张图像的上传完成了。



3.2 上传到亚马逊的 S3

我们未能在中国大陆区提供亚马逊 **S3** 的节点，请中国大陆的开发者使用阿里云的 **OSS** 节点。

如果最近的上传节点是在 S3，开发者可以使用 S3 上传图像。

- 在每张图像上传前，调用 mutation `uploadImageS3(pid, bucket, filename, type, checksum)` 获取一个有效期为 3 小时的加密 url 和相关的图像 meta info。
- 调用 `startImageUpload(id)` 通知服务器一张图像的上传即将开始。
- 通过标准的 HTTP PUT 命令把图像的文件上传到加密的 url 上，文件名为 `${image.filename}` 。上传时需要把传输的内容标记为 `Content-type: JPEG` 。相片上传后无需调用 mutation `doneImageUpload(id)`

4. 等待图像预处理

图像上传到 OSS 或者 S3 后，Altizure 的服务器会对图像进行验证和预处理，确保图像都是正确有效的。最终每个图像都会被标记为 `Ready` 或者 `Invalid` 。如果 `Ready` 的图像数目和您想上传的图像数目一样，就可以继续调用 mutation `startReconstructionWithError(id, options)` 来开始三维重建。如果有效图像的数目不对，开发者可以通知用户重新上传图像确保图像正确，重新上传中可以利用 mutation `hasImage` 来判断图像是否已经存在服务器上来决定是否重新上传，只上传服务器上不完整或者不存在的图像。

开发者也可以不等待 Altizure 服务器对图像进行验证，在图像上传完后，便调用 mutation `preStartReconstruction(id, options)` 。如果这样，服务器在验证完所有图像后，就会用已有的有效图像开始三维重建。

5. 了解更多

- 了解更多关于 STS 的信息
- 阿里云 OSS JavaScript SDK

Last modified at Mon Aug 20 2018 16:30:23 GMT+0800 (HKT)

Altizure GraphQL API 更新日志

2.0.0

Release data: November. 23rd, 2018

New features:

- Added mutation: makeLODImage for making LOD tiles of an user uploaded image
- Added field: lod in type: ProjectImage for showing the state and other meta info of tilings of this image

1.10.1

Release data: November. 16th, 2018

New features:

- Added super/sales query/mutation for getting/setting weekly recommended projects

Fixes:

- Caught invalid project flight id when querying Project.flight

1.10.0

Release data: November. 9th, 2018

New features:

- Added User.language for storing the preferred display language
- Added GET handler (api.altizure.com/subscribe/news?email=abc@gmail.com&locale=zh_tw) for subscribing promotional emails

Change:

- Added new args: language in mutation: updateMyProfile

Fixes:

- Fixed releasing incorrect number of membership coupons
- Fixed auto-starting stalled projects created in mobile apps

1.9.9

Release data: November. 2nd, 2018

New features:

- Added new user setting for subscribing promotional emails

Change:

- Updated the design of user verification and reset password emails

Fixes:

- Fixed infinite pending of processing imported model
- Fixed long waiting time for setting thumbnail via CN api

1.9.8

Release data: October. 19th, 2018

New features:

- Added super mutation: removeProject

Change:

- .com returns cloudfront paths of thumbnails for non-logined user, while .cn returns ali-cdn
- Lowered the minimum size of image from ~8 MegaPixel to ~2 MegaPixel for calculating the GP usage, effectively lowering the cost of projects with small images

1.9.7

Release data: October. 12th, 2018

New features:

- Added new fields: error and errorCode in Project for giving the error of reconstruction or model conversion

Improvements:

- Non China staff-picked or paid projects are available in Cloudfront

Change:

- Changed type: ProjectDiagnostic.error as a list of errors instead of a single error

Fixes:

- Fixed getting user.freeGPQuota in super query

1.9.6

Release data: October. 5th, 2018

Fixes:

- Fixed default value of `range` of geoNearProjects

1.9.5

Release data: September. 14th, 2018

New features:

- Added mutations: browseProject + unbrowseProject for tracking which projects are browsed
- Added query: my.browsingHistory for listing all the projects that have been viewed

Change:

- Hide Chinese projects if user's country is not China, Hong Kong, Taiwan, Macao and Singapore

Fixes:

- Fixed getting the correct values of Project.initialCamPos, instead of zero values
- Fixed normalizing phone number, all non-digit characters are stripped, e.g. +852-123456789 is equivalent to 852123456789

1.9.4

Release data: September. 7th, 2018

New features:

- Added new field: importedModels in Project for querying all of the imported models of a project
- Added Search.sandboxes for searching sandbox by names
- Supported signing in by verified phone number and one time password sent via SMS
- Added mutation: requestLoginCode for getting one time password over SMS
- Added mutation: getUserTokenByLoginCode for getting self-issued user token from verified phone and one time password

Change:

- Each mobile phone number could only be associated with one account
- Allowed setting logo for sandbox
- Allowed setting visibility and logo for imported projects
- Mutation: getUserToken would not throw GraphQL error if credentials are incorrect, would only return null instead

Fixes:

- Fixed type errors of Project.onEarch, SandboxEntity.pid, SandboxEntity.tid
- Fixed sandbox mutations by respecting related membership plan

1.9.3

Release data: August. 31th, 2018

New features:

- Added new type of membership plan for upgrading between plans
- Added trial state in user.developer.status
- Added query: support.developerProgram for showing the details of developer program
- Added mutation: payDeveloperBraintree for paying developer program via Braintree
- Added mutation: createAliPayDeveloperOrder for paying developer program via Alipay

Change:

- Added args: filter in query: support.membershipPlans for querying `Normal` or `Upgrade` or `All` plans
- Updated mutation: uploadImageS3 so that the `image` field no longer needed to be queried for creating the `ProjectImage`, only the url is required now
- Updated mutation: uploadMetaFileS3 so that the `file` field no longer needed to be queried for creating the `ProjectMetaFile`, only the url is required now

Fixes:

- Fixed undefined error when querying User.membershipState if no user token is provided

1.9.2

Release data: August. 24th, 2018

New features:

- Added initialCamPos in Project.modelTransform, for getting the initial position when rendering a project
- Added input argument: initialCamPos in mutation.setProjectProperties for setting initial camera position
- Added input argument: maxPolarAngle in mutation.setProjectProperties for setting maximum polar angle
- Added input argument: onLoadTrans in mutation.setProjectProperties for setting on load transform of model
- Added mutation: setProjectCropMask for setting crop mask
- Added mutation: setProjectWaterMask for setting water mask

- Added private fields: phone + phoneVerified in User object
- Added mutation: requestPhoneVerification for sending SMS to registered phone
- Added mutation: verifyPhone for verifying SMS code
- Supported sending notification alerts via SMS to verified phone

Change:

- Allowed setting user phone number + whether to receive sms alert in mutation: updateMyProfile

Fixes:

- Fixed setting membership plan name

1.9.1

Release data: August. 17th, 2018

New features:

- Added support for paying coins or membership plan via Alipay web and mobile
- Added transaction types: AlipayBuyMembership
- Added super/sales query: refundAliPay
- Added super/sales query: allGQLEvents

Change:

- Renamed mutation: createAliPayOrder to createAliPayCoinsOrder

Fixes:

- Fixed setting OAuth apps domains to not to include the protocol

1.9.0

Release data: August. 10th, 2018

New features:

- Added options: taskType in mutation: startReconstructionWithError, for starting different task types
- Automatically create thumbnail for every newly imported model
- Added new fields: assetStorage, visibility, coupon and modelPerProject etc in MembershipInfo in User object, to query current membership details
- Added query: support.membershipPlans for querying membership plans information
- Added new field: savePaymentMethod in User object, to indicate whether to store the payment method in Braintree or not
- Added new args: savePaymentMethod in mutation: updateMyProfile, to remove any existing payment methods stored
- Added query: Sandbox.collaborators relay connection
- Added super query: userTransactions
- Added new field: lastDoneGP in Project
- Added mutation: payMembershipBraintree to pay for membership plan via Braintree

Change:

- Allowed renewing membership without extending existing period in super/sales api
- Added args: sortBy + search in query my.favouriteProjects and my.favouriteSandboxes

Fixes:

- Fixed setting the correct GP as last done GP in project upgrade transaction, instead of the current GP

1.8.6

Release data: July. 27th, 2018

New features:

- Supported custom notification description by offering tokenized description arguments `descArgs` in type: Notification

Change:

- Allowed any user to set thumbnail of Sandbox

1.8.5

Release data: July. 20th, 2018

New features:

- Added query: My.allProjects, for querying native and imported model projects at the same time
- Added mutation: makeScreenshot to make a server side rendered screenshot with specific dimension and camera pose
- Added mutation: removeScreenshot to remove a screenshot of a project

Fixes:

- Fixed empty importedModelType error

1.8.4

Release data: July. 13th, 2018

New features:

- Added fields: animation + texture in type: SandboxEntity
- Added args: planet + background in SandboxMutation.create + update

Fixes:

- Fixed setting uploaded image as `Invalid` if client and server side checksums mismatch

1.8.3

Release data: July. 6th, 2018

New features:

- Added query: my.favouriteProjects / my.favouriteSandboxes for querying favourite projects with descending created time
- Added sorting options for sandbox apis
- Added super/sales query: userProjectList to output all the projects as CSV of a user

Change:

- Removed deprecated query: version, User.thumb, getGeoIPInfo.suggestedBuckets
- Removed deprecated mutation: startReconstruction, upgradeProject,

1.8.2

Release data: June. 22th, 2018

New features:

- Added query: search.user(username) for searching the account unique username

1.8.1

Release data: June. 15th, 2018

New features:

- Supported changing the imported model type via mutation: setProjectProperties(importedModelType)

Change:

- For query: project.cropMask, if crop mask does not exist, return a default mask instead of null

Fixes:

- Fixed issues of translating original image names to hashed image names in meta files if same image is uploaded multiple times
- Fixed changing project.importedState of imported-model-project to `Pending` immediately after a new upload URL is signed

1.8.0

Release data: May. 25th, 2018

New features:

- Supported Minio object storage server
- Added mutation: uploadImageMinio, to upload image to our private Minio server
- Added mutation: downgradeProject, for downgrading pro and non-done-once project to free project

Fixes:

- Fixed issue of pre-processing images indefinitely triggered from mobile apps, projects that have timeout images will be auto-started

Change:

- Changed SandboxEntity's name from required to optional.
- Changed domains of cdn of various sandbox thumbs according to the region of api server

1.7.0

Release data: May. 18th, 2018

New features:

- Added mutation: transferProject for transferring ownership of projects to other user
- Added mutation: transferCoins for transferring coins to other user
- Added mutation: transferCoupon for transferring unused and non-expired timed coupons to other user

Breaking Change:

- Allowed zero transaction for starting reconstruction / upgrading project. (Previously, if total value of timed coupon used is bigger than required, no transaction will be conducted. Now, a transaction of zero amount will be carried out with relevant information such as the coupons used in the transaction.)

1.6.2

Release data: May. 11th, 2018

New features:

- Added type: frame in sandbox's animation

Fixes:

- Fixed setting project as modified if an image is removed
- Fixed missing message of 'Authentication Error' in various mutations

1.6.1

Release data: May. 4th, 2018

New features:

- Added transaction types: [BraintreeBuyCoin, AlipayBuyCoin, WechatBuyCoin, DepositCoinCoupon, Reconstruction, ProjUpgrade, Refund, SpecialCharge] in type: Transaction
- Supported custom transaction description by offering tokenized description arguments `descArgs` in type: Transaction
- Added ids of coupons used in relevant transaction
- Added sales mutation: refundProject and specialChargeUser

Fixes:

- Fixed issue of not refunding a pro project if it is stopped within 15 minutes

1.6.0

Release data: April. 27th, 2018

New features:

- Added mutation: sandbox.storyboard.{createStory + updateStory + removeStory + createStoryScene + removeStoryScene}
- Supported image file name that contains space(s), e.g. 'abc def ghi.JPG' in metafile (despite space is being used as delimiter)

Improvements:

- Reduced the latency of all the query (especially in China) with geo-distributed replica set

Fixes:

- Fixed typo of metafile filename to accept `group.txt` instead of `groups.txt`

Breaking Change:

- All images that are not in state: `Ready(Strong)` or `Invalid` will be forcefully switched to `Invalid` if their last `modifiedDate` is older than 65 minutes

1.5.1

Release data: April. 20th, 2018

New features:

- Supported markdown format in sales query: sendEmail
- Added mutation: setSandboxThumb for setting progressive sandbox thumbnails
- Added sales mutation: addUserNote and removeUserNote
- Added new buckets: Beijing, Virginia and London in enum: BucketS3Model, used in mutation: uploadModelS3
- Added new query: getGeoIPInfo.nearestModelBuckets to get the nearest buckets for uploading obj model

1.5.0

Release data: April. 13th, 2018

New features:

- Added field: importedModelType in Project, to indicate the imported model type
- Added field: developer.level in User, to store the base level of subsequent apps
- Allowed sales api to get/set developer base level and individual apps level
- Added sales api to send email

Fixes:

- Fixed setting default materials for imported models
- Allowed sales to query private coupon fields
- Enforced expiring developer status and all of his apps
- Properly handled the case when project image is removed when it is still in excited state

Breaking Change:

- Changed the definition of User.freeGPQuota to {free global quota + staff picked bonus + membership quota}, instead of just {free global quota + staff picked bonus}
- project.allMaterials were set as public, instead of private

1.4.0

Release data: April. 6th, 2018

New features:

- Added 2 new filters: ActiveUsed + ActiveUnused in FILTER_COUPON_TIME_STATE, which is used in query.my.coupons
- Added query.sandbox.sandbox + query.sandbox.allSandboxes + query.my.sandboxes
- Added mutation.sandbox
- Supported importing CAD, photogrammetric and point-cloud models

Fixes:

- Fixed status 500 server error if no user token is provided in any query
- Fixed handling pre-starting non-existing project
- Fixed unhandled promise rejection error when registering new user with empty fields

Breaking Change:

- Mutation: createProject accepts an additional arg: modelType for indicating the imported model type.

1.3.1

Release data: March. 29th, 2018

New features:

- Added resendVerificationEmail in query.support
- Added args: type in query.my.transactions to filter credit or debit transactions
- Added new field: MembershipInfo in User object

Fixes:

- Fixed issues of translating original image names to hashed image names in meta files
- Fixed issues of user login via OAuth

1.3.0

Release data: March. 23th, 2018

New features:

- Supported accepting multiple time based coupons in a single operation
- Added sales API

Fixes:

- Fixed redeem coupon link in http server mode
- Fixed ProjectImage's 'resized' ValidatorError when calling mutation: hasImage
- Fixed mis-judging 'Invalid' image as 'ReadyStrong'

Breaking Change:

- Mutation: upgradeProjectWithError, startReconstructionWithError and preStartReconstruction accept a list of coupon ids, instead of a single coupon id.

1.2.0

Release data: March. 16th, 2018

New features:

- User registration: send an email verification upon user registration
- Added query: project.allMaterials for getting all materials
- Added mutation: addProjectMaterial + updateProjectMaterial + removeProjectMaterial

Breaking Change:

- Mutation: PreStartProject will start the reconstruction task if all the images are in either 'Ready' or 'Invalid' state. So PreStartProject should be called after all images are uploaded. Previously, it is supposed to be called before all images are 'Ready' or 'Invalid'.

1.1.0

Release data: Feb. 12th, 2018

New features:

- Make endpoint: `/graphql` browsable without app key
- Added query: support.projectDoctor for diagnostic information
- Added super query: approveDeveloper

Improvements:

- Used provider pattern for selecting cloud providers upload services

Bug fixes:

- Fixed query: suggestedBuckets return values

Altizure JavaScript SDK

这是一个以 JavaScript 为开发语言的 SDK。提供了丰富的三维浏览和编辑的功能。SDK 的主要目的是：

- 简化加载和渲染海量三维数据的开发工作
- 简化各种数据源的数据在实景三维底图上的整合和高效显示
- 简化实景三维数据和行业应用的开发工作

使用 Altizure JavaScript 3D SDK 并结合 Electron 、 React Native 等混合开发的工具具有大量实用功能，而开发者只需掌握网页编程所需的 JavaScript 和 HTML 就可以轻松开发出高质量的实景三维桌面和移动应用，助力您的商业应用。

1. 快速开始指南

想把地球显示在屏幕上，总共分几步？

三步。

第一步：引用 SDK

在网页的 head 部分，引用我们的 SDK。

```
<!-- 设置编码, 确保 utf8 字符的正确显示 -->
<meta charset="utf-8">
<!-- 设置 viewport, 确保移动端的正确渲染 -->
<meta name="viewport" content="width=device-width, initial-scale=1.0, user-scalable=no">
<!-- 引用sdk -->
<script type="text/javascript" src="https://beta.altizure.com/sdk"></script>
```

我们提供四个版本的 SDK 引用链接供您选择：

- 最新版：`<script type="text/javascript" src="https://beta.altizure.com/sdk"></script>`
- 稳定版：`<script type="text/javascript" src="https://www.altizure.com/sdk"></script>`
- 中国最新版：`<script type="text/javascript" src="https://beta.altizure.cn/sdk"></script>`
- 中国稳定版：`<script type="text/javascript" src="https://www.altizure.cn/sdk"></script>`

第二步：创建三维显示容器

三维数据的下载和渲染会由我们的 SDK 完全接管，用户只需要创建一个 `div` 指定用于渲染的容器的位置和大小。

```
<body>
  <div id="page-content"></div>
</body>
```

第三步：创建三维引擎对象

我们的三维引擎是以最新的 Altizure 地球的引擎作为基础，新建对象时需要把它附着在一个作为显示容器的 `div` 里。

```
// 创建一个参数配置对象
let options = {
  altizureApi:{
    // 填入您的 app key
    key: 'your-app-key'
  }
}
```

```
// 创建地球渲染引擎对象，附着在 page-content 这个 div 上
let sandbox = new altizure.Sandbox('page-content', options)
```

其中 `'page-content'` 是第二步中创建三维显示容器的 `div` 的 id。 `options` 用于配置新建的引擎对象，更多参数可以参考下面的范例和详细文档。

小结

所有代码组合起来就是：

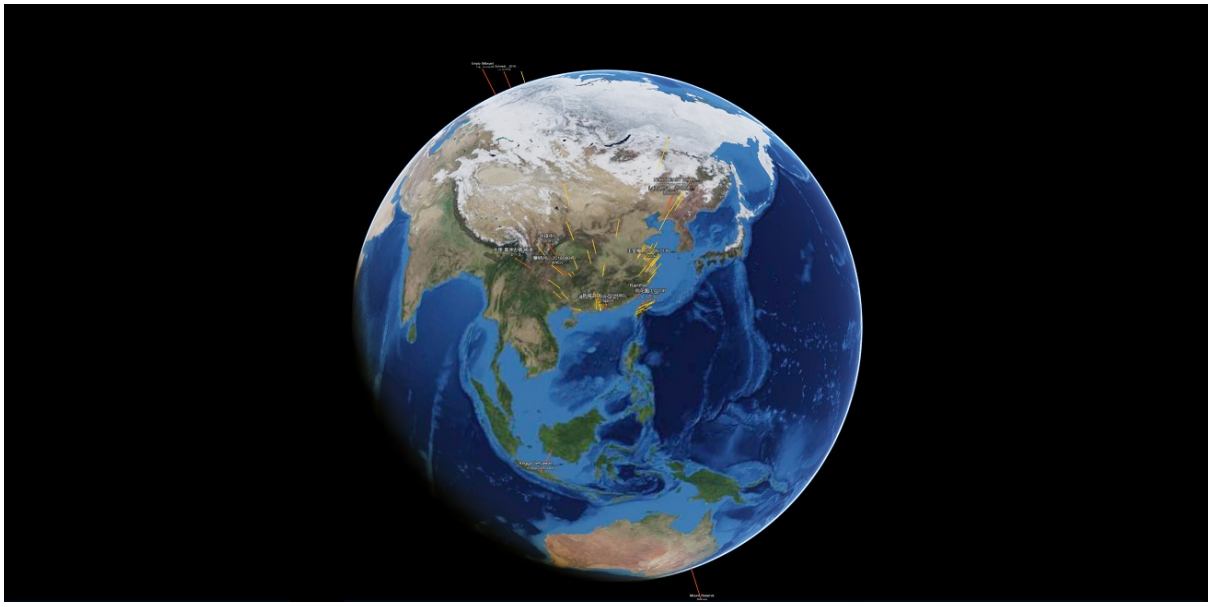
```
<html lang="en">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0, user-scalable=no">
  <script type="text/javascript" src="https://beta.altizure.com/sdk"></script>
</head>
<body>
  <div id="page-content"></div>
  <script>
    let options = {
      altizureApi:{
        // 填入您的 app key
        key: 'your-app-key'
      }
    }

    let sandbox = new altizure.Sandbox('page-content', options)
  </script>
</body>
</html>
```

把这段代码保存为一个 html 文件并放在一个文件夹如 `<path>/altizure-sdk-test/earth.html` 中，然后在控制台中键入：

```
cd <path>/altizure-sdk-test/
python -m SimpleHTTPServer
```

再通过浏览器访问 `http://127.0.0.1:8000/earth.html` 就可以加载这个 Altizure 地球了。

您也可以访问演示页面观看这段代码的效果。



只要三步，我们便创建出了一个可以加载全球实景三维模型的视图。惊不惊喜？激不激动？

相信您已经被 Altizure SDK 多彩有趣的内容深深吸引了，接下来我们将直接通过范例代码来学习 SDK 的丰富功能。

2. 范例

2.1 概念释义

我们简单解释一下出现在范例里的元素的概念

- Sandbox (沙盒)： altizure.Sandbox (沙盒) 是整个三维应用的核心，它负责管理整个三维场景的数据和绘制。通过对沙盒进行定制和添加数据，可以定制出非常强大的三维应用。这是编写三维应用的主要入口。
- Marker (标记)： 包括 OBJMarker，ProjectMarker，CanvasTagMarker 等在内的所有模型和标签都可以被称作 marker，通过将具有不同功能的 marker 组合起来，可以获得具有丰富功能的应用，比如说我们可以在一个模型中添加水流、文字标签、火焰效果等。
- Camera (相机)： altizure.Sandbox.camera(相机) 即观众以什么视角来观看 marker，Camera 的坐标和方向就是屏幕中心点在 Sandbox 空间中的位置和朝向。

2.2 范例列表

您可以参考 github.com/altizure/sdk.examples 的教程把这些范例代码下载下来，在本地建立服务器进行尝试。您只需要对其中的部分函数做些简单修改便可以和您现有的系统进行整合。

您也可以直接访问 SDK demo 来直接尝试各种范例的效果。

SDK中的控件和接口的使用方法在 SDK 文档 中有详细的解释。

对范例和使用方法有任何疑问可以在 issue page 中进行提问和交流。

3. 常见问题

可以访问常见问题了解更多常见问题解答。

演示应用页面展示了更加复杂的 Altizure JavaScript SDK 应用范例。

访问详细文档了解更多细节。

4. 了解更多

- ThreeJS
- WebGL
- OpenGL
- Vulkan
- OpenGL 坐标变换详解 OpenGL Transformation

—

该文档最后修改于 Mon Aug 27 2018 14:30:22 GMT+0800 (HKT)

Altizure JavaScript SDK 演示应用

这里我们会演示如何用 Altizure JavaScript SDK 解决一些实际应用场景的问题，让 Altizure 的实景模型可以更好地结合到您的商业场景中，发挥实景三维模型的无穷潜力。

您对这些演示有任何问题都欢迎到 issue page 留言，我们会尽力解答。

1. 时间轴

时间轴可以非常直观地展示一个场景按照时间的变化。这是工程进度管理等任何与时间相关的应用都必不可少的工具。下面我们就通过几个应用演示几种可视化时间轴的方法。

1.1 格状平铺时间轴

- 演示链接：altizure.github.io/experimental-demo/timeline.html
- 演示重点：多个 Altizure 模型加载，文字标签设置，模型位置调整
- 演示简介：范例里对一个工地的施工过程超过20个不同时间点的三维模型进行了格状的平铺展示。每个模型带有相关的时间文字标签。

1.2 滑动条时间轴

- 演示链接：altizure.github.io/experimental-demo/timeline-slider.html
- 演示重点：模型可见性修改，实景三维内容与 HTML 界面交互，相机环绕动画，模型动态加载和销毁
- 演示简介：范例里通过一个滑动条让用户可以任意选取一个工地的施工过程一个时间点的三维模型进行浏览和检视。用户还可以按时间先后循序播放这个工地不同时间段的施工状况。范例里也提供了一个自动环绕飞行的功能，方便用户在展示时自动环绕兴趣点循环播放。

2. 传感器

现代的手持设备里有着各种各样的传感器，测量用户的位置、设备姿态、运动速度等等。这个章节的演示会展示如何结合这些传感器信息让您的三维模型更加具有互动性。

2.1 GPS 轨迹记录

- 演示链接：altizure.github.io/experimental-demo/gps-tracker.html
- 演示重点：图片标签加载，标签位置移动，相机跟随动画，地理坐标与屏幕坐标转换，Canvas 二维元素与实景三维元素结合，获取模型表面高程。
- 演示简介：该演示推荐用手机打开。在演示中，您可以打开轨迹记录，这样您的位置就会以演示中的模型的中心点为基准点，实时以一个 Altizure 图标的形式显示在实景三维模型中。同时您的移动轨迹也会用折线显示出来。移动轨迹的折线的渲染模式可以在三维渲染或者 canvas 二维渲染模式之间切换。您还可以打开相机跟踪模式，让相机在您移动的过程中使用跟随您移动。

3. 性能测试

3.1 文字标签渲染

- 演示链接：altizure.github.io/experimental-demo/performance-test.html
- 演示重点：文字标签加载和删除，测试三维渲染性能，获取模型表面高程

- 演示简介：演示默认加载 1000 个文字标签。可以通过修改标签数目，并点击 `Generate` 生成不同数目的标签。可以同时显示的标签数目越多，说明您的电脑的渲染能力越强。

3.2 线渲染

- 演示链接：altizure.github.io/experimental-demo/polycylinder-limittest.html
- 演示重点：管线加载和删除，测试三维渲染性能
- 演示简介：演示默认加载 100 个有 7 个位置随机的节点的管线。可以通过修改管线数目和节点数目，并点击 `Generate` 生成不同节点数的管线。可以同时显示的有更多节点数的管线数目越多，说明您的电脑的渲染能力越强。

4. 导入GeoJSON矢量地图

- 演示链接：altizure.github.io/experimental-demo/import-geojson.html
- 演示重点：加载GeoJson类型的矢量数据
- 演示简介：演示加载香港科技大学的三维模型以及矢量数据。点击 `Visualize GeoJSON` 加载矢量数据，每个矢量单体可相应交互并显示预设信息。

5. 如何获取这些演示的代码

这些演示应用都是开源的，您可以通过以下的命令把代码复制到您的电脑查看并运行。

```
git clone https://github.com/altizure/experimental-demo.git
cd experimental-demo
python -m SimpleHTTPServer
```

打开您的浏览器，访问 `http://127.0.0.1:8000/` 便可运行和查看这些演示应用。

6. 了解更多

- Altizure JavaScript SDK
- Altizure JavaScript SDK 常见问题
- Altizure JavaScript SDK 详细文档

—

该文档最后修改于 Fri Aug 10 2018 18:30:59 GMT+0800 (HKT)

常见问题

1. 使用相关

1.1 能否提供测试？

您可以直接访问 altizure.github.io/sdk.examples/examples.sdk.html 来直接尝试各种范例的效果。把代码仓库 clone 到本地后即可修改其中的代码进行测试。

1.2 本地测试时无法访问相关模型？

请检查以下配置是否正确：

- 是否已经启动一个 http 服务器？最方便的方法是参考 altizure.github.io/sdk.examples/ 教程中用 python 建立 http 服务器的方法。
- 本地测试请用 `127.0.0.1` 作为地址来访问，而不要使用 `localhost`。

1.3 无法访问 SDK script 链接？

如果无法访问 `https://www.altizure.com/sdk` 或 `https://beta.altizure.com/sdk`，请使用中国站点 `https://www.altizure.cn/sdk` 或 `https://beta.altizure.cn/sdk`。

1.4 详细文档在哪儿？

请直接参考范例的代码，里面有详细注释。我们会持续更新相关范例，展示最新功能。

1.5 如何获取用户令牌？

请直接参考 GraphQL API 的 `5. 获取用户令牌` 章节。

1.6 Altizure 的三维引擎是用什么语言开发的？

开发语言是 JavaScript，最底层三维的接口是 WebGL。但是我们并不经常直接访问 WebGL 底层接口，我们更多的模块是搭建在 ThreeJS 之上。

1.7 能否自定义一些 three.js 的对象然后插入到 Altizure JavaScript SDK 的场景里？

暂时不能。因为实景模型的数据量非常巨大，我们的 SDK 对数据的加载和内存管理做了许多优化，暂时未能允许插入任意的 three.js 对象。我们会逐步实现这一点，但现在并无具体上线这个功能的时间表。

1.8 我是否需要学习图形学的知识才能用 Altizure JavaScript SDK 编程？

基本不需要，我们尝试把这些复杂的技术实现都进行了封装。您只需要懂得网页编程所需要的 JavaScript 和 HTML 就可以进行开发。更重要的是要清晰知道实景三维模型和您的业务是如何结合的。

2. 常见用例

2.1 点击模型中某个建筑物，怎么来获取对应的信息，如建筑物的名称？

可以通过鼠标事件获得经纬度，再去相关数据库中查建筑物名称。相关建筑物数据库由开发者自行管理。

2.2 能否直接插入 OBJ 模型？

可以。如果网络下载带宽不高，过大的 OBJ 文件可能导致下载时间过长。而且因为受限于浏览器的性能，过大的 obj 也会导致浏览器崩溃。我们一般推荐 obj 模型连同纹理文件不要超过 2 MB。

OBJ 模型需要满足以下要求：

- OBJ 模型需要由开发者自行找网络空间存储并获取直接访问的 https 链接。
- 其中 OBJ 对应的 mtl 文件内的纹理路径需要是相对路径。
- OBJ 只有三角形面。
- OBJ 模型里没有非2-流形的点和边，并且没有面积为 0 的面。

2.3 如果需要更换一个 marker 的 click 事件，应该怎么操作？需要把前一个绑定事件解绑吗？

最好解绑，用 `.off(event, func)`

2.4 如何实现分图层显示不同的 marker 及其他的模型？

可以在打开和关闭图层的时候从图层数据库中读取相关信息，然后将相关 marker 插入到 Sandbox 或者从中删除。如果不想每次开关图层反复插入和删除相关 marker ，可以用 `marker.visible=true` 或者 `marker.visible=false` 控制每个 marker 显示与否。

2.5 如何实现测量距离？

通过鼠标事件分别获取两次点击的坐标点，然后求两点距离。

2.6 如何加入一个带格式的文本标签

`TextTagMarker` 是一个带简单格式的文字标签，参考 范例2.6。

如果您需要更复杂的格式定义，您可以用 js 创建一个 canvas, 写入带格式的文本，把 canvas 转为 image, 传给 Tag。(可以把文字和图标都画到这张canvas上。)

2.7 如何获得三维场景拖动事件

响应 Sandbox 对象的 `cameraChange` 事件，`sandbox.on('cameraChange', callback)`。

2.8 地图框选的功能，希望能够在地图上画个方形或圆形，然后可以获取到画的图形的范围值

pick 鼠标 down 和 up 的点,

- pick 到的两个点作为矩形的对角两个端点，矩形的边平行于经纬方向, 使用 polygon marker 绘制矩形
- 以 pick 到的两个点作为直径，中点做圆心，还是用 polygon 就可以，比如画一个 360 边形。已知圆心经纬度，和直径，顶点位置可以很容易算出来。

2.9 如何删除一个 marker ?

`marker.destruct()` 可以完全销毁一个 marker，这个 marker 所用的资源将被释放。如果您想反复显示和隐藏这个 marker，可以设定这个参数 `marker.visible=true`。请参考 2.4 的解答。

2.10 平台的坐标系是什么？

WGS84，

如果您在使用过程中位置出现了数百米的偏差，那很大概率是由于坐标系的问题，详细的解释请参考 `3. 关于坐标` 章节。

2.11 如果我需要通过方法调用实现放大，或缩小，接口是什么？

更改相机高度 `alt` ，参考范例第五章。

2.12 如果我需要通过方法调用旋转、平移，接口是什么？

更改相机经纬度 `lat/lng` 实现平移， `tilt` 更改俯角（向下的与水平方向的夹角）， `north` 更改方位角（与正北方向顺时针的夹角）。请参考范例第五章。

2.13 flyTo方法，是否有飞行速度参数？

有。 `flyTo: function (position, speed)` 。 `speed` 是一个速度的相对倍率，例如设做 2 的话，就是默认速度的 2 倍。

2.14 水面高度设置？

水面高度是在 Altizure 主站上定义的, SDK 目前不允许修改。

2.14 如何加载矢量图层数据？

您需要读取、解析您所使用的矢量数据，转化为 SDK 提供的点线面，加入场景中。

2.15 如何实现局部区域透明显示设置？

您可以使用 Altizure 主站的 裁剪 功能。

2.16 如何实现消息窗口功能？（类似点击弹出信息窗口）

使用 JavaScript 和 HTML 写好您想显示的弹窗，再和 marker 的鼠标事件绑定，比如 mouseover 或 click 时显示。

2.17 如何移动 marker，或者绑定 GPS 设备的坐标？

只需要修改 marker 的坐标即可移动 marker。需要开发者通过客户端或服务器和不同设备通信获取设备的 GPS 坐标，便可直接把该坐标设做相关 marker 的坐标，这样就确保 marker 反映了设备的 GPS 位置。可以参考演示应用中的演示 2.1。

这里所说的坐标是 WGS84 坐标系下的，因此在将使用这一坐标的 marker 与百度、高德等基于国测局坐标的国内地图厂商的底图进行结合时会产生偏差，谷歌则不受此影响。详情请参考本章 `3. 关于坐标` 部分。

2.18 Sandbox 里的 pid 是什么？

pid 是 project id 的简写。 浏览 Altizure 的 url (例如，https://www.altizure.com/project/59fe68cebb619d03c446fe85/model) 里面的 `59fe68cebb619d03c446fe85` 就是 pid。

2.19 如何初始化 Sandbox 的相机 (camera) 参数？

这里的 camera 参数，是指 Sandbox 初始化的时候的相机位置。它可以通过 SDK 得到，参考 范例 5.1。camera 的参数是以经纬度，高度定义的。经纬度可以从谷歌（要和您坐标系一致，请注意国内地图厂商所使用的国测局坐标与 GPS 所用的 WGS84 坐标之间有数百米的偏差，谷歌不受此影响）拾取，或者更简单的，从范例 4.1拾取，点击想要的地方，就会显示了。console 里有日志, 要离地面近一点也能看到文字标签。 如果您也在使用 Altizure GraphQL API, project 函数中有存地理位置的参数。在 geoInfo 里的 centerLat, centerLng， 可以分别赋值给 lat, lng。高度您可以自己定义，比如 1000(米)。

2.20 平台支持 BIM 格式的文件吗？

目前不支持，有指定需要再开发。

2.21 有没有方法修改 marker 的 position 坐标？同一个 marker，不销毁重建的情况下，如何修改其位置/通过鼠标点击地图来设置 marker 的位置？

```
marker.setPose (position, orientation, scale)
```

这里 `position: {lng, lat, alt}`，`orientation: {x, y, z, w}`，`scale: number`。 如果只改 `position`，另外两个可以设为 `undefined`，如 `setPose(position, undefined, undefined)`。

2.22 将 OBJ 转成类似 Altizure 原生项目那样的金字塔分层（LOD）数据，有什么要求？

obj 需要符合一些数学规则，才可以做 LOD。包括：OBJ 只有三角形面。OBJ 模型里没有非2-流形的点和边，并且没有面积为0的面。详情需联系技术人员测试。

2.23 如何给一个已有的 `PolyLineMarker` 添加更多的点？

使用 `altizure.PolyLineMarker::addPointMarker` 如：

```
polylineMarker.addPointMarker(new altizure.LngLatAlt(
  lng, lat, alt
))
```

2.24 实景模型/OBJ 模型加载至地球指定位置，指定朝向，指定比例尺？

实景模型/OBJ 模型 使用的是同样的接口

marker.position 位置

marker.euler 欧拉方向

marker.orientation 四元数方向

marker.scale 比例尺

- 参考範例 2.1， 2.5.1
- 参考文档

2.25 制作OBJ模型的一些建议(非必须)

- 使用建模软件时，用 1m 作为单位长度，导入 SDK 时就不需要重新调整 scale 比例尺。
- （如果是对实景建筑物的建模）OBJ 模型和实景模型的形状需要尽量相近。
- 使用尽量少的面片数可以提升加载效率。
- 可将光照效果烘焙至纹理中，可以有更真实的渲染效果。

- 贴图尺寸采用 2^N 作为边长，比如 64×64, 128×256。

3. 关于坐标

3.1 我们使用什么坐标？

本平台所使用的坐标系为 WGS84，通常谷歌、GPS（包括北斗）所采集到的经纬度均是在这一坐标系下的。

3.2 WGS84 坐标、GCJ02 坐标、BD09 坐标

如果您在使用过程中不涉及国内地图厂商的产品的话，那您不需要花费过多的时间阅读本条目。

- WGS84 坐标即各种国际地图厂商和通用 GPS 设备所采用的坐标系。
- GCJ02 坐标（又称国测局坐标、火星坐标）是在 WGS84 基础上对大中华地区（有的地图产品不包含台湾）所有坐标进行加密，如果把采用 WGS84 坐标的物体放到 GCJ02 底图上，会产生100-700米不等的偏差。
- BD09 坐标是百度地图在 GCJ02 坐标的基础上再次加密所获得的坐标。

最简单的解决办法就是不论模型或底图均使用同一坐标系，如果使用了不同的坐标系，则需要进行坐标转换。

请注意：

国内主流地图厂商均提供了 API 方便用户从 WGS84 进行加密，但是反向解密（即将其他坐标系转为 WGS84）没有任何官方的解算方法，非官方解算方法均有误差且在大规模使用时有可能触犯法律，请遵守当地法律法规。Altizure 不提供任何可以反向解算坐标的接口。

3.3 坐标相关资料

- 百度地图相关解释及转为BD09的API
- 高德地图相关解释及坐标转换API
- 维基百科相关内容

4. 阅读更多

- 基础范例列表
- 演示应用 中展示了更加复杂的应用范例
- Altizure JavaScript SDK 详细文档

—

该文档最后修改于 Fri Aug 10 2018 18:30:59 GMT+0800 (HKT)

Altizure Sandbox 3D SDK 更新日志

0.2.24

发布时间: 2018年6月20日

新功能:

- 新版的 `sandbox.camera.lookAt` ，可以调整camera使得marker在视野中心。增加了範例5.10。

改进:

- 优化了 `PolyFenceLineMarker` 的贴图逻辑，避免贴图错误。

缺陷修复:

- 修复了IE11对 `Math.log2` 未定义的报错。

0.2.22

发布时间: 2018年6月11日

新功能:

- `PolyLineMarker` 增加了动画，以及可选的纹理。可以通过调整 `marker.style{animation, textture}` 设置不同效果。範例2.4

```
/**
  * set the style of the fence
  * @public
  * @param {object} _style
  * @memberof PolyFenceLineMarker
  * @example <caption> Set the style </caption>
  *  polyFenceLineMarker.style = {
  *    texture: 'fence', // 'fence', 'arrow' or an image file url
  *    color: 0xffffff,
  *    animation: 'horizontal', // or 'verticle', 'none'. default 'hortizontal'
  *    opacity: 1.0
  *  }
  */
  set style (_style){}
```

- 根据选取的(不少于)三个点作为水平面，调整marker的的向上方向， `Marker::orientationByHorizontalPoints`

```
/**
  * get the realigned orientation from horizontal points
  * @param {LngLatAlt[]} pts
  * @param {bool} flip
  * @public
  * @returns {Quaternion}
  * @memberof Marker
  */
  orientationByHorizontalPoints (pts, flip = false)
```

- 高速测量平面面积，在不需要体积测量时使用 `GeometryUtils::areaByLngLatAlts`

```
/**
  * @param {LngLatAlt[] | PolygonBaseMarker} pts - boundary points
  * @returns {Number} - squared meters
  */
```

```
    function areaByLngLatAlts (pts) {}
```

- 判断一个点（投影到地表后）是否在polygon范围内 `GeometryUtils::checkPtsInPoly`

```
/**
  * @param {LngLatAlts} pts
  * @param {LngLatAlts} poly - conter clock wise boundary
  * @returns { bool[] } - whether the points are in the poly
  */
  function checkPtsInPoly (pts, poly) {}
```

改进:

- 增加autoScale範例 范例 2.1.2

缺陷修复:

0.2.20

发布时间: 2018年5月24日

新功能:

- Tag 类 Marker 增加 `isSprite` 属性，区分是否总面向镜头。更新 范例 2.2*。
- Tag 类增加 `pivot` -- 面向镜头时的旋转中心，详见文档。
- `AltizureProjectMarker` 初始化选项 `autoscale: {number | bool}` 自动缩放CAD类模型。范例 2.1.2
- 根据天际线的区域限高。范例 6.6

改进:

- 保留主站裁剪效果。
- 加入cdn自动发布流程。
- 集中处理离线渲染。
- 提高quality上限。
- 添加瓦片bbox显示接口 showTileBBox。
- CameraMatrix重构及用例测试。

缺陷修复:

- 当有重叠出现，positionsAltitude 返回错误高度。
- GeoSystem 对齐后， Marker::position 未更新。
- CAD模型的材质显示错误。
- Marterial array 导致的 marker::destruct 报错。
- 销毁项目模型时释放裁剪Mask内存。

0.2.5

发布时间: 2018年4月9日

新功能:

- 区域标注 Zone Marker。范例 2.11
- 建筑对象化。 範例 6.4
- 标注聚合。範例 6.5

- 质量-效率平衡选项 sandbox.quality。
- 高进度投影坐标系 GeoSystem。
- CAD转换
- 交互式可视域分析。範例5.9

改进:

- threejs 更新至0.91.0
- 提高渲染效率。

缺陷修复:

- LightBeam color 接口，接受string类型。

0.1.62

发布时间: 2018年2月8日

新功能:

- Shadow Map 和通视分析。范例 5.8

改进:

- 提高渲染效率。

缺陷修复:

- 修复触碰手势和鼠标点击事件无响应的bug。