

天筹求解器

API 参考

文档版本 01
发布日期 2024-03-04



版权所有 © 华为云计算技术有限公司 2024。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为云计算技术有限公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为云计算技术有限公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

华为云计算技术有限公司

地址：贵州省贵安新区黔中大道交兴功路华为云数据中心 邮编：550029

网址：<https://www.huaweicloud.com/>

目 录

1 使用前必读..... 1

1.1 概述..... 1

1.2 调用说明..... 1

1.3 终端节点..... 1

1.4 约束限制..... 2

1.5 基本概念..... 2

2 API 概览..... 4

3 如何调用 API..... 5

3.1 购买服务..... 5

3.2 构造请求..... 5

3.3 认证鉴权..... 8

3.4 返回结果..... 8

4 API..... 10

4.1 二维切割方形件..... 10

4.1.1 创建任务..... 10

4.1.2 查询单个任务..... 14

4.1.3 查询任务列表..... 18

4.1.4 删除任务..... 20

4.2 二维切割异形件..... 20

4.2.1 创建切割分析任务..... 20

4.2.2 查询任务详情..... 31

4.3 求解器..... 36

4.3.1 创建任务..... 36

4.3.2 查询任务..... 39

5 附录..... 43

5.1 状态码..... 43

5.2 错误码..... 46

1 使用前必读

- 1.1 概述
- 1.2 调用说明
- 1.3 终端节点
- 1.4 约束限制
- 1.5 基本概念

1.1 概述

天筹求解器服务(OptVerse)是一种基于华为云基础架构和平台的智能决策服务，以自研AI求解器为核心引擎，结合机器学习与深度学习技术，为企业提供生产计划与排程、切割优化、路径优化、库存优化等一系列有竞争力的行业解决方案。

OptVerse以开放API（Application Programming Interface，应用程序编程接口）的方式提供给用户，用户通过实时访问和调用API获取推理结果，帮助用户自动采集关键数据，打造智能化业务系统，提升业务效率。

您可以使用本文档提供的天筹求解器服务API的描述、语法、参数说明及样例等内容，进行相关操作，例如天筹求解器服务包含的二维切割等具体接口使用说明。支持的全部操作请参见[2 API概览](#)。

在调用天筹求解器服务API之前，请确保已经充分了解运筹优化算法相关概念，详细信息请参见“产品介绍”。

1.2 调用说明

天筹求解器服务提供了REST（Representational State Transfer）风格API，支持您通过HTTPS请求调用，调用方法请参见[3 如何调用API](#)。

1.3 终端节点

终端节点即调用API的请求地址，不同服务不同区域的终端节点不同，您可以从[地区和终端节点](#)中查询所有服务的终端节点。

运筹优化算法服务的终端节点如表1-1所示，请您根据业务需要选择对应区域的终端节点。

表 1-1 运筹优化算法服务的终端节点

区域名称	区域	终端节点 (Endpoint)	部署服务
华北-北京四	cn-north-4	optverse.cn-north-4.myhuaweicloud.com	二维切割，求解器

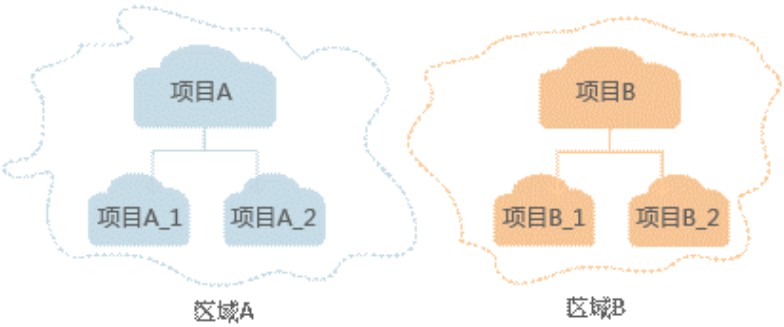
1.4 约束限制

更详细的限制请参见具体API的说明。

1.5 基本概念

- 账号
用户注册华为云时的账号，账号对其所拥有的资源及云服务具有完全的访问权限，可以重置用户密码、分配用户权限等。由于账号是付费主体，为了确保账号安全，建议您不要直接使用账号进行日常管理工作，而是创建用户并使用他们进行日常管理工作。
- 用户
由账号在OptVerse中创建的用户，是云服务的使用人员，具有身份凭证（密码和访问密钥）。
在我的凭证下，您可以查看账号ID和用户ID。通常在调用API的鉴权过程中，您需要用到账号、用户和密码等信息。
- 区域
指云资源所在的物理位置，同一区域内可用区间内网互通，不同区域间内网不互通。通过在不同地区创建云资源，可以将应用程序设计的更接近特定客户的要求，或满足不同地区的法律或其他要求。
- 可用区
一个可用区是一个或多个物理数据中心的集合，有独立的风火水电，AZ内逻辑上再将计算、网络、存储等资源划分成多个集群。一个Region中的多个AZ间通过高速光纤相连，以满足用户跨AZ构建高可用性系统的需求。
- 项目
华为云的区域默认对应一个项目，这个项目由系统预置，用来隔离物理区域间的资源（计算资源、存储资源和网络资源），以默认项目为单位进行授权，用户可以访问您账号中该区域的所有资源。如果您希望进行更加精细的权限控制，可以在区域默认的项目中创建子项目，并在子项目中购买资源，然后以子项目为单位进行授权，使得用户仅能访问特定子项目中资源，使得资源的权限控制更加精确。

图 1-1 项目隔离模型



2 API 概览

OptVerse所提供的接口均为自研接口。

通过使用OptVerse的自研接口，您可以使用OptVerse的所有功能，如[表2-1](#)所示。

表 2-1 接口说明

API	说明
二维切割方形件	满足客户多样化的约束场景和策略选择，为用户提供高利用率、高产能的切割方案
二维切割异形件	
求解器	为用户提供线性规划和混合整数线性规划问题的求解服务。

3 如何调用 API

[3.1 购买服务](#)

[3.2 构造请求](#)

[3.3 认证鉴权](#)

[3.4 返回结果](#)

3.1 购买服务

首次使用OptVerse之前，必须先购买套餐包“开通服务”。服务购买步骤如下所示。

前提条件

申请OptVerse之前，必须先申请华为云账号并进行实名认证。通过此账号，您可以使用所有华为云服务，并且只需为您所使用的服务付费。具体申请信息请见[账号注册](#)和[实名认证](#)。

使用服务

1. 进入华为云在左侧列表搜索并点击“OptVerse”服务，进入天筹求解器服务管理控制台（需要使用华为云账号登录）。
2. 选择服务所在的区域。
系统默认显示“华北-北京四”，请根据各服务的部署区域选择对应区域，具体请参见[终端节点](#)。
3. 进入套餐包管理页面点击购买套餐包。
4. 用户创建应用并授权。

3.2 构造请求

本节介绍REST API请求的组成，并说明如何调用OptVerse服务的API，该API获取用户的AppCode，AppCode可以用于调用其他API时鉴权。

请求 URI

请求URI由如下部分组成。

{URI-scheme} :// {Endpoint} / {resource-path} ? {query-string}

尽管请求URI包含在请求消息头中，但大多数语言或框架都要求您从请求消息中单独传递它，所以在此单独强调。

表 3-1 URI 中的参数说明

参数	说明
URI-scheme	表示用于传输请求的协议，当前所有API均采用HTTPS协议。
Endpoint	指定承载REST服务端点的服务器域名或IP，不同服务不同区域的Endpoint不同，您可以从 地区和终端节点 中获取。 例如OptVerse服务在“华北-北京四”区域的Endpoint为“optverse.cn-north-4.myhuaweicloud.com”。
resource-path	资源路径，也即API访问路径。 从具体API的URI模块获取，例如“获取用户Token”API的resource-path为“/v3/auth/tokens”。
query-string	查询参数，是可选部分，并不是每个API都有查询参数。查询参数前面需要带一个“？”，形式为“参数名=参数取值”，例如“limit=10”，表示查询不超过10条数据。

说明

为查看方便，在每个具体API的URI部分，只给出resource-path部分，并将请求方法写在一起。这是因为URI-scheme都是HTTPS，而Endpoint在同一个区域也相同，所以简洁起见将这两部分省略。

请求方法

HTTP请求方法（也称为操作或动词），它告诉服务你正在请求什么类型的操作。

表 3-2 HTTP 方法

方法	说明
GET	请求服务器返回指定资源。
PUT	请求服务器更新指定资源。
POST	请求服务器新增资源或执行特殊操作。
DELETE	请求服务器删除指定资源，如删除对象等。
HEAD	请求服务器资源头部。
PATCH	请求服务器更新资源的部分内容。 当资源不存在的时候，PATCH可能会去创建一个新的资源。

例如，在创建任务的URI部分，您可以看到其请求方法为“POST”，则其请求为：

```
POST https://optverse.cn-north-4.myhuawei.com/v1/optverse/2dcut/regular-plate/tasks
Content-Type: application/json
X-Apig-AppCode: ABCDEFJ....
```

请求消息头

附加请求头字段，如指定的URI和HTTP方法所要求的字段。例如定义消息体类型的请求头“Content-Type”，请求鉴权信息等。

如下公共消息头需要添加到请求中。

表 3-3 公共请求消息头

名称	说明	是否必选	示例
Content-Type	发送的实体的MIME类型。	是	application/json; charset=utf-8
X-Apig-AppCode	用户AppCode。	是	-

说明

AK(Access Key ID)：访问密钥ID。与私有访问密钥关联的唯一标识符；访问密钥ID和私有访问密钥一起使用，对请求进行加密签名。

SK(Secret Access Key)：与访问密钥ID结合使用的密钥，对请求进行加密签名，可标识发送方，并防止请求被修改。

请求消息体

请求消息体通常以结构化格式发出，与请求消息头中Content-type对应，传递除请求消息头之外的内容。若请求消息体中参数支持中文，则中文字符必须为UTF-8编码。

每个接口的请求消息体内容不同，也并不是每个接口都需要有请求消息体（或者说消息体为空），GET、DELETE操作类型的接口就不需要消息体，消息体具体内容需要根据具体接口而定。

例如，对于创建任务接口，您可以从接口的请求部分看到所需的请求参数及参数说明。将消息体加入后的请求如下所示，加粗的斜体字段需要根据实际值填写，其中 ***username***为用户名，***domainname***为用户所属的账号名称，***********为用户登录密码，***xxxxxxxx***为project的ID。

```
POST https://optverse.cn-north-4.myhuawei.com/v1/optverse/2dcut/regular-plate/tasks
Content-Type: application/json

{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],

```

```
{
  "password": {
    "user": {
      "name": "username",
      "password": "*****#",
      "domain": {
        "name": "domainname"
      }
    }
  },
  "scope": {
    "project": {
      "id": "xxxxxxxxx"
    }
  }
}
```

到这里为止这个请求需要的内容就具备齐全了，您可以使用[curl](#)、[Postman](#)或直接编写代码等方式发送请求调用API。

3.3 认证鉴权

您当前可以选择以下方式进行认证鉴权。

- **AppCode认证**：通过AppCode认证通用请求。

AppCode 认证

AppCode认证就是在调用API的时候将AppCode加到请求消息头，从而通过身份认证，获得操作API的权限。

在应用授权后获取AppCode，再调用其他接口时，您需要在请求消息头中添加“X-APIG-AppCode”。例如AppCode值为“ABCDEFJ....”，则调用接口时将“X-APIG-AppCode: ABCDEFJ....”加到请求消息头即可，如下所示。

```
GET https://optverse.cn-north-4.myhuaweicloud.com/v3/auth/projects
Content-Type: application/json
X-APIG-AppCode: ABCDEFJ....
```

3.4 返回结果

状态码

请求发送以后，您会收到响应，包含状态码、响应消息头和消息体。

状态码是一组数字代码，状态码表示了请求响应的状态，完整的状态码列表请参见[5.1 状态码](#)状态码。

响应消息头

对应请求消息头，响应同样也有消息头，如“Content-type”。例如[表3-4](#)所示。

表 3-4 响应消息头

名称	说明
Content-Length	响应消息体的字节长度，单位为Byte。
Date	系统响应的时间。
Content-Type	响应消息体的MIME类型。

响应消息体

响应消息体通常以结构化格式返回，与响应消息头中Content-type对应，传递除响应消息头之外的内容。

为篇幅起见，这里只展示部分内容。

```
{
  "task_id": "40axxx-701a-47cf-8b56-648xxxxxd25"
}
```

当接口调用出错时，会返回错误码及错误信息说明，错误响应的Body体格式如下所示。

```
{
  "error_code": "optverse.00000006",
  "error_msg": "Input data validate error: the item material is 18MM_PLATE, item_id is 0: Length and width are larger than plate.\n"
}
```

其中，error_code表示错误码，error_msg表示错误描述信息。

4 API

- 4.1 二维切割方形件
- 4.2 二维切割异形件
- 4.3 求解器

4.1 二维切割方形件

4.1.1 创建任务

功能介绍

二维切割服务基于数学规划和元启发式算法的优化引擎，适用于家具行业板材、石材切割，机械制造中的钣金切割，建筑领域的玻璃切割，服装领域的布料排版切割，在满足客户多样化的约束场景和切割策略选择的需求下，为客户提供高利用率、高效率的切割方案。

URI

POST /v1/{project_id}/optverse/2dcut/regular-plate/tasks

请求参数

二维切割前的原材料请求参数如[表4-1](#)所示。

表 4-1 原料请求参数说明

参数	是否必须	参数类型	说明
plate_id	是	Int	原材料ID
plate_length	是	Float	原材料长
plate_width	是	Float	原材料宽
plate_thick	是	Float	原材料厚度

参数	是否必须	参数类型	说明
plate_material	是	String	原材料制作材料
plate_grain	是	Bool	原材料是否有纹路
plate_num	是	Int	原材料数量
is_offcut	否	Bool	是否为余料
plate_cost	否	Float	原材料成本

二维切割后的成品请求参数如表4-2所示。

表 4-2 成品请求参数说明

参数	是否必须	参数类型	说明
item_id	是	int	成品ID
item_length	是	Float	成品长
item_width	是	Float	成品宽
item_thick	是	Float	成品厚度
item_num	是	Int	成品数量
item_turn	是	Bool	成品是否可以旋转 (无纹路: true,有纹路: false)
item_material	是	String	成品材料
item_name	是	String	成品名称
is_substitute	否	Bool	是否为替补板
label_info	是(键值对内的信息可随意增加与删减)	map<String,String>	打印标签信息的键值对组合

二维切割条件参数的请求参数如表4-3所示。

表 4-3 切割条件参数的请求参数说明

参数	是否必须	参数类型	说明
kerf_horizontal	是	Float	横向锯刀宽度
kerf_vertical	是	Float	纵向锯刀宽度
plate_trim_horizontal	是	Float	原材料横向修边量

参数	是否必须	参数类型	说明
plate_trim_vertical	是	Float	原材料纵向修边量
item_forward_trim	是	Float	再分锯切前向修边量
item_backward_trim	是	Float	再分锯切后部修边量
first_cut_direction	是	Int	进料方向（横向0，纵向1，任意方向2）
max_stack_height	是	Float	最大叠板高度
calculate_time	是	Int	算法的最长计算时间
offcut_length	否	Float	余料长度下限
offcut_width	否	Float	余料宽度下限
offcut_area	否	Float	余料面积下限
time_cost	否	Float	每小时的锯切时间成本
max_cut_stage_num	否	Int	开始切割后，同一个方向上的切割，统计为一次，需在设置的转板次数内，将大板切割完毕
min_stack_utilization	否	Float	最小的叠板利用率
item_sort_location	否	Bool	长条板件中部件的放置顺序，true最大部件放在左侧，false最大部件放在右侧
item_num_limit	否	Int	锯切图上的小板种类数量限制

示例

- 请求示例

```
POST https://{endpoint}/v1/{project_id}/optverse/2dcut/regular-plate/tasks
Request Header:
Content-Type: application/json;charset=utf-8
Auth-Token:
Body:
{
  "items": [
    {
```

```
    "item_id": 0,
    "item_material": "18MM_PLATE",
    "item_length": 60.0,
    "item_width": 553.0,
    "item_turn": true,
    "item_num": 1,
    "item_name": "底板",
    "item_thick": 18,
    "is_substitute": false,
    "label_info": {
      "order_code": "订单号",
      "plan_code": "批次号",
      "item_code": "小板标识"
    }
  },
  {
    "item_id": 0,
    "item_material": "18MM_PLATE",
    "item_length": 60.0,
    "item_width": 553.0,
    "item_turn": true,
    "item_num": 1,
    "item_name": "底板",
    "item_thick": 18,
    "is_substitute": true,
    "label_info": {
      "order_code": "订单号",
      "plan_code": "批次号",
      "item_code": "小板标识"
    }
  }
],
"parameters": [
  {
    "calculate_time": 30,
    "first_cut_direction": 2,
    "item_backward_trim": 0.0,
    "item_forward_trim": 0.0,
    "kerf_horizontal": 4.4,
    "kerf_vertical": 4.4,
    "max_stack_height": 80,
    "plate_trim_horizontal": 10.0,
    "plate_trim_vertical": 10.0,
    "time_cost": 200,
    "max_cut_stage_num": 3,
    "min_stack_utilization": 0.8,
    "item_sort_location": false,
    "item_num_limit": 0
  }
],
"plates": [
  {
    "plate_grain": true,
    "plate_id": 0,
    "plate_length": 2440.0,
    "plate_material": "18MM_PLATE",
    "plate_num": 9999,
    "plate_thick": 18.0,
    "plate_width": 1220.0,
    "plate_cost": 100
  }
]
```

- **成功响应示例**

```
{
  "task_id": "40axxxx-701a-47cf-8b56-648xxxxxxd25"
}
```

参数说明如[表4-4](#)所示。

表 4-4 参数说明

参数	参数类型	说明
task_id	String	任务ID

- 失败响应示例

```
{  "error_code": "optverse.00000006",  "error_msg": "Input data validate error: the item material is 18MM_PLATE, item_id is 0: Length and width are larger than plate.\n"}

```

4.1.2 查询单个任务

功能介绍

该API用于查询并显示单个任务详情。

URI

GET /v1/{project_id}/optverse/2dcut/regular-plate/tasks/{task_id}

请求参数

表 4-5 参数说明

参数	参数类型	说明
task_id	String	任务ID

请求消息

请求样例

GET /v1/{project_id}/optverse/2dcut/regular-plate/tasks/
40a8175b-701a-47cf-8b56-64875cabcd25

响应参数

表 4-6 响应参数说明

参数	参数类型	说明
error_detail	String	如果运行出错会提示
result	Object	如果运行正常会在该属性内给出结果
status	String	任务运行状态，详情请参见 表4-7

参数	参数类型	说明
task_id	String	任务id
result_list	Array of objects	切割方案列表
plan_info	Array of objects	切割方案的统计信息列表
cut_file	String	以字符串形式保存的切割方案文件
plan_name	String	此方案序号（根据参数设置可给出多组方案）
material_info	String	统计信息，详情请参见表4-8
material_name	String	材料名称
problem_id	Int	按材料划分的问题序列号

表 4-7 任务状态说明

状态	说明
waiting	在任务队列中，等待执行, 当等待时间超过一定时间，会超时，转为abnormal状态
running	任务正在执行
finished	任务成功运行结束，通过`result`获取结果
failed	任务执行出错，通过`error_detail`获取错误信息
abnormal	任务异常，可能是排队超时，可能是重试次数超时，可能是计算时间超时，通过`error_detail`获取错误信息

表 4-8 统计信息响应参数说明

参数	单位	参数类型	说明
cut_length	米(m)	float	切割长度
cut_sum	-	Int	切割刀数
cut_time	分钟(min)	float	锯切时间
item_num	-	Int	开料件数
material	-	String	材料

参数	单位	参数类型	说明
program_num	-	Int	锯切图数量
run_program_num	-	Int	运行的锯切图数
rate	百分比(%)	float	原材料利用率
sum_area	平方米(m ²)	float	原材料使用面积
sum_cost	-	Float	锯切成本
sum_plate_num	-	Int	原材料使用数量
sum_used_area	平方米(m ²)	float	原材料利用面积

响应示例

成功响应示例

```
{
  "result": {
    "result_list": [
      {
        "cut_file": "<?xml version='1.0' encoding='UTF-8' standalone='yes'?'><AutoCut ver=
'100'><Project Name='订单' CutTime='999.99' Selected='-1' AppliionVer='1.0.0' Update=
'2020-4-22 11:14:00'><Data Class='3' TotalUnit='1000000' Type='18.0_18MM_PLATE' Ply='18.0'
Grain='True'><Objective Type='Shape' Count='1'><Shape X='60.0' Y='553.0' Turn='True'
Count='1' Name='底板' TLType='18MM_PLATE_18.0mm' FH='18.0' item_code='小板标识'
order_code='订单号' plan_code='批次号'></Objective><Substitute Type='Shape' Count=
'1'><Shape X='60.0' Y='553.0' Turn='True' Count='1' Name='底板' TLType=
'18MM_PLATE_18.0mm' FH='18.0' item_code='小板标识' order_code='订单号' plan_code='批次号
'></Substitute><Condition Type='Shape' Count='1'><Shape Name='18MM_PLATE' X='2440.0' Y=
'1220.0' Turn='True' Sx='10.0' Sy='10.0' Count='9999'></Condition><Solutions Selected=
'0'><Solution SolverName='排料名称' SolveGap='4.4' SheetCounts='1' SumUnitArea='66360.0'
SumSheetArea='2976800.0' SumSpareArea='2910440.0'><Layouts><Layout ID='0' SheetID='0'
Count='1' Cut='False' X='2440.0' Y='1220.0' UnitArea='66360.0' SheetArea='2976800.0'
SpareArea='2910440.0'><Part X='1212.2' Y='2432.2' Px='7.8' Py='7.8'><Part X='2432.2' Y=
'60.0' Px='7.8' Py='7.8'><Part X='60.0' Y='553.0' Px='7.8' Py='7.8' Turn='False' UID='0'
BID='0' Angle='90'><Substitute X='60.0' Y='553.0' Px='565.2' Py='7.8' Turn='False' SID=
'0' BID='0' Angle='90'></Part></Part></Layout></Layouts></Solution></Solutions></Data></
Project></AutoCut>",
        "plan_info": {
          {
            "material_info": {
              "cut_length": 5.02,
              "cut_sum": 5,
              "cut_time": 1.1,
              "item_num": 2,
              "material": "18MM_PLATE",
              "program_num": 1,
              "rate": 0.02229,
              "run_program_num": 1,
              "sum_area": 2.98,
              "sum_cost": 103.67,
              "sum_plate_num": 1,
              "sum_used_area": 0.07
            },
            "material_name": "18MM_PLATE",
            "problem_id": 1
          }
        ],
        "plan_name": "方案1"
      },
    ]
  },
}
```

```
"cut_file": "<?xml version='1.0' encoding='UTF-8' standalone='yes'?'><AutoCut ver=
'100'><Project Name='订单' CutTime='999.99' Selected='-1' ApplicationVer='1.0.0' Update=
'2020-4-22 11:14:00'><Data Class='3' TotalUnit='1000000' Type='18.0_18MM_PLATE' Ply='18.0'
Grain='True'><Objective Type='Shape' Count='1'><Shape X='60.0' Y='553.0' Turn='True'
Count='1' Name='底板' TLType='18MM_PLATE_18.0mm' FH='18.0' item_code='小板标识'
order_code='订单号' plan_code='批次号'></Objective><Substitute Type='Shape' Count=
'1'><Shape X='60.0' Y='553.0' Turn='True' Count='1' Name='底板' TLType=
'18MM_PLATE_18.0mm' FH='18.0' item_code='小板标识' order_code='订单号' plan_code='批次号
'></Substitute><Condition Type='Shape' Count='1'><Shape Name='18MM_PLATE' X='2440.0' Y=
'1220.0' Turn='True' Sx='10.0' Sy='10.0' Count='9999'></Condition><Solutions Selected=
'0'><Solution SolverName='排料名称' SolveGap='4.4' SheetCounts='1' SumUnitArea='66360.0'
SumSheetArea='2976800.0' SumSpareArea='2910440.0'><Layouts><Layout ID='0' SheetID='0'
Count='1' Cut='False' X='2440.0' Y='1220.0' UnitArea='66360.0' SheetArea='2976800.0'
SpareArea='2910440.0'><Part X='1212.2' Y='2432.2' Px='7.8' Py='7.8'><Part X='2432.2' Y=
'60.0' Px='7.8' Py='7.8'><Part X='60.0' Y='553.0' Px='7.8' Py='7.8' Turn='False' UID='0'
BID='0' Angle='90'><Substitute X='60.0' Y='553.0' Px='565.2' Py='7.8' Turn='False' SID=
'0' BID='0' Angle='90'></Part></Part></Layout></Layouts></Solution></Solutions></Data></
Project></AutoCut>",
"plan_info": [
{
"material_info": {
"cut_length": 5.02,
"cut_sum": 5,
"cut_time": 1.1,
"item_num": 2,
"material": "18MM_PLATE",
"program_num": 1,
"rate": 0.02229,
"run_program_num": 1,
"sum_area": 2.98,
"sum_cost": 103.67,
"sum_plate_num": 1,
"sum_used_area": 0.07
},
"material_name": "18MM_PLATE",
"problem_id": 1
},
],
"plan_name": "方案2"
},
{
"cut_file": "<?xml version='1.0' encoding='UTF-8' standalone='yes'?'><AutoCut ver=
'100'><Project Name='订单' CutTime='999.99' Selected='-1' ApplicationVer='1.0.0' Update=
'2020-4-22 11:14:00'><Data Class='3' TotalUnit='1000000' Type='18.0_18MM_PLATE' Ply='18.0'
Grain='True'><Objective Type='Shape' Count='1'><Shape X='60.0' Y='553.0' Turn='True'
Count='1' Name='底板' TLType='18MM_PLATE_18.0mm' FH='18.0' item_code='小板标识'
order_code='订单号' plan_code='批次号'></Objective><Substitute Type='Shape' Count=
'1'><Shape X='60.0' Y='553.0' Turn='True' Count='1' Name='底板' TLType=
'18MM_PLATE_18.0mm' FH='18.0' item_code='小板标识' order_code='订单号' plan_code='批次号
'></Substitute><Condition Type='Shape' Count='1'><Shape Name='18MM_PLATE' X='2440.0' Y=
'1220.0' Turn='True' Sx='10.0' Sy='10.0' Count='9999'></Condition><Solutions Selected=
'0'><Solution SolverName='排料名称' SolveGap='4.4' SheetCounts='1' SumUnitArea='66360.0'
SumSheetArea='2976800.0' SumSpareArea='2910440.0'><Layouts><Layout ID='0' SheetID='0'
Count='1' Cut='False' X='2440.0' Y='1220.0' UnitArea='66360.0' SheetArea='2976800.0'
SpareArea='2910440.0'><Part X='1212.2' Y='2432.2' Px='7.8' Py='7.8'><Part X='2432.2' Y=
'60.0' Px='7.8' Py='7.8'><Part X='60.0' Y='553.0' Px='7.8' Py='7.8' Turn='False' UID='0'
BID='0' Angle='90'><Substitute X='60.0' Y='553.0' Px='565.2' Py='7.8' Turn='False' SID=
'0' BID='0' Angle='90'></Part></Part></Layout></Layouts></Solution></Solutions></Data></
Project></AutoCut>",
"plan_info": [
{
"material_info": {
"cut_length": 5.02,
"cut_sum": 5,
"cut_time": 1.1,
"item_num": 2,
"material": "18MM_PLATE",
"program_num": 1,
"rate": 0.02229,
```

```
        "run_program_num": 1,
        "sum_area": 2.98,
        "sum_cost": 103.67,
        "sum_plate_num": 1,
        "sum_used_area": 0.07
    },
    "material_name": "18MM_PLATE",
    "problem_id": 1
}
},
"plan_name": "方案3"
},
{
    "cut_file": "<?xml version='1.0' encoding='UTF-8' standalone='yes'?><AutoCut ver=
\\'100\\'><Project Name='\\'订单\\' CutTime='\\'999.99\\' Selected='\\'-1\\' ApplicationVer='\\'1.0.0\\' Update=
\\'2020-4-22 11:14:00\\'><Data Class='\\'3\\' TotalUnit='\\'1000000\\' Type='\\'18.0_18MM_PLATE\\' Ply='\\'18.0\\'
Grain='\\'True\\'><Objective Type='\\'Shape\\' Count='\\'1\\'><Shape X='\\'60.0\\' Y='\\'553.0\\' Turn='\\'True\\'
Count='\\'1\\' Name='\\'底板\\' TLType='\\'18MM_PLATE_18.0mm\\' FH='\\'18.0\\' item_code='\\'小板标识\\'
order_code='\\'订单号\\' plan_code='\\'批次号\\'></Objective><Substitute Type='\\'Shape\\' Count=
\\'1\\'><Shape X='\\'60.0\\' Y='\\'553.0\\' Turn='\\'True\\' Count='\\'1\\' Name='\\'底板\\' TLType=
\\'18MM_PLATE_18.0mm\\' FH='\\'18.0\\' item_code='\\'小板标识\\' order_code='\\'订单号\\' plan_code='\\'批次号
\\'></Substitute><Condition Type='\\'Shape\\' Count='\\'1\\'><Shape Name='\\'18MM_PLATE\\' X='\\'2440.0\\' Y=
\\'1220.0\\' Turn='\\'True\\' Sx='\\'10.0\\' Sy='\\'10.0\\' Count='\\'9999\\'></Condition><Solutions Selected=
\\'0\\'><Solution SolverName='\\'排料名称\\' SolveGap='\\'4.4\\' SheetCounts='\\'1\\' SumUnitArea='\\'66360.0\\'
SumSheetArea='\\'2976800.0\\' SumSpareArea='\\'2910440.0\\'><Layouts><Layout ID='\\'0\\' SheetID='\\'0\\'
Count='\\'1\\' Cut='\\'False\\' X='\\'2440.0\\' Y='\\'1220.0\\' UnitArea='\\'66360.0\\' SheetArea='\\'2976800.0\\'
SpareArea='\\'2910440.0\\'><Part X='\\'1212.2\\' Y='\\'2432.2\\' Px='\\'7.8\\' Py='\\'7.8\\'><Part X='\\'2432.2\\' Y=
\\'60.0\\' Px='\\'7.8\\' Py='\\'7.8\\'><Part X='\\'60.0\\' Y='\\'553.0\\' Px='\\'7.8\\' Py='\\'7.8\\' Turn='\\'False\\' UID='\\'0\\'
BID='\\'0\\' Angle='\\'90\\'><Substitute X='\\'60.0\\' Y='\\'553.0\\' Px='\\'565.2\\' Py='\\'7.8\\' Turn='\\'False\\' SID=
\\'0\\' BID='\\'0\\' Angle='\\'90\\'></Part></Part></Layout></Layouts></Solution></Solutions></Data></
Project></AutoCut>",
    "plan_info": [
        {
            "material_info": {
                "cut_length": 5.02,
                "cut_sum": 5,
                "cut_time": 1.1,
                "item_num": 2,
                "material": "18MM_PLATE",
                "program_num": 1,
                "rate": 0.02229,
                "run_program_num": 1,
                "sum_area": 2.98,
                "sum_cost": 103.67,
                "sum_plate_num": 1,
                "sum_used_area": 0.07
            },
            "material_name": "18MM_PLATE",
            "problem_id": 1
        }
    ],
    "plan_name": "方案4"
}
]
},
"status": "finished",
"task_id": "40axxxx-701a-47cf-8b56-648xxxxxx25"
}
```

4.1.3 查询任务列表

功能介绍

该API用于查询并显示完整的任务列表。（注：任务列表中的任务没有任何顺序关系）

URI

GET /v1/{project_id}/optverse/2dcut/regular-plate/tasks/?offset=0&size=1

参数说明如表4-9所示。

表 4-9 参数说明

参数	参数类型	说明
size	String	分页大小
offset	String	页序号

请求消息

请求样例

GET /v1/{project_id}/optverse/2dcut/regular-plate/tasks/?offset=0&size=1

响应参数

二维切割响应参数如下所示。

表 4-10 响应参数说明

参数	参数类型	说明
total	Int	总的任务数
count	Int	本次查询的任务数
tasks	Array of objects	任务列表

响应示例

成功响应示例

```
{
  "total": 1,
  "count": 1,
  "tasks": [
    {
      "task_id": "40axxxx-701a-47cf-8b56-648xxxxxx25",
      "status": "running",
      "result": {},
      "error_detail": null
    }
  ]
}
```

4.1.4 删除任务

功能介绍

该API用于查询并显示完整的任务列表。（注：任务列表中的任务没有任何顺序关系）

URI

DELETE /v1/{project_id}/optverse/2dcut/regular-plate/tasks/{taskId}

参数说明如表4-11所示。

表 4-11 参数说明

参数	参数类型	说明
task_id	String	任务ID

请求消息

请求样例

DELETE /v1/{project_id}/optverse/2dcut/regular-plate/tasks/
40a8175b-701a-47cf-8b56-64875cabcd25

响应示例

失败响应示例

```
{
  "error_code": "ModelArts.3801",
  "error_msg": "任务不存在"
}
```

4.2 二维切割异形件

4.2.1 创建切割分析任务

功能介绍

创建二维切割任务，该接口为异步操作，返回任务ID，任务的状态及结果请通过查询任务详情接口获取。

URI

POST /v1/{project_id}/optverse/2dcut/irregular-textile/tasks

表 4-12 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	用户项目ID，获取方法请参见获取项目ID。

请求参数

表 4-13 请求 Header 参数

参数	是否必选	参数类型	描述
X-Apig-AppCode	是	String	用户凭证。

Array of 表10 objects

表 4-14 请求 Body 参数

参数	是否必选	参数类型	描述
input_json	是	object	任务输入信息，json格式，具体数据结构参见IrregularTextileInputJson。

表 4-15 IrregularTextileInputJson

参数	是否必选	参数类型	描述
id	是	String	任务id，用于取回排料结果时使用。
order	是	String	订单名。
name	是	String	此床的名字。
time	是	Integer	运行时间，单位：分钟，取值范围[1,30]
gap_mode	否	Integer，枚举	间隙处理模式。备用，当前不支持。[0,1]
overlap	否	Double	两列之间交错的程度，0表示整齐 的边界；[0, 100000]
fabric	是	Array of 表5 objects	面料属性。

参数	是否必选	参数类型	描述
polygon	是	Array of 表6 objects	简单多边形定义。
piece	是	Array of 表7 objects	裁片定义。
hole	否	Array of 表9 objects	面料瑕疵定义，详细定义见下文。
constraints	否	Array of 表10 objects	其他约束定义。

表 4-16 fabric

参数	是否必选	参数类型	描述
width	是	Double	面料宽度。保留1位小数，若超过将直接进行截断。(0, 100000]
grid_x	否	Double	面料x轴防线格子宽度，若无设置为0；保留1位小数，若超过将直接进行截断。备用，当前不支持。[0, 100000]
grid_y	否	Double	面料y轴方向格子宽度，若无设置为0；保留1位小数，若超过将直接进行截断。备用，当前不支持。[0, 100000]

表 4-17 Polygon

参数	是否必选	参数类型	描述
index	是	Integer	多边形id，从0开始，依次递增。
profile	是	Array of double	简单多边形轮廓点，逆时针排序，x、y轴坐标依次排序展开，元素取值范围[-100000, 100000]，保留1位小数，超过将直接进行截断；要求元素个数至少为6，且为偶数。[-100000, 100000]

表 4-18 Piece

参数	是否必选	参数类型	描述
index	是	Integer	裁片id, 从0开始, 依次递增, 用于在输出数据gap_index中使用。(>=0)
polygon_index	是	Integer	实体多边形(裁片轮廓)在”polygon”向量中的index。(>=0)
gap_index	否	Integer	间隔多边形(加上最小间隔的裁片轮廓)在”polygon”向量中的index, -1表示没有间隔。备用, 当前不支持。(>=-1)
hole_index	否	Integer	裁片内部孔洞在在”polygon”向量中的index, -1表示没有孔洞。备用, 当前不支持。(>=-1)
tilted	否	Double	微小旋转角度, 原旋转角度(由属性rot获得)+-此数值后的获得旋转角度也作为裁片可旋转角度。通过此方法获得旋转角度不考虑裁片翻转。0表示禁止旋转; 保留1位小数, 若超过将直接进行截断。[0,15]
gap	否	Double	裁片之间的间隔, 对于不支持间隔多边形的算法有效; 保留1位小数, 若超过将直接进行截断。备用, 当前不支持。[0, 100000]
pose_group	否	Integer	姿态Y约束分组标记, -1表示没有姿态Y约束。若有分组, 同一组的裁片要求其旋转和翻转集合相同。当前不支持部分裁片分组, 部分裁片不分组(>=-1)
column_group	否	Integer	列分组标记, -1表示不分行排版。当前不支持部分裁片分组, 部分裁片不分组。(>=-1)
rot	是	Double 枚举	裁片可旋转角度倍数。(0,90,180)
flip_x	是	Integer 枚举	1表示允许裁片沿x轴翻转, 0表示禁止。(0,1)
flip_y	是	Integer枚举	1表示允许裁片沿y轴翻转, 0表示禁止。(0,1)
remark	否	String	备注信息

参数	是否必选	参数类型	描述
y	否	Array of 表8 object	不同翻转情形的裁片Y约束。若某翻转情形没有定义，采用不发生翻转的情况，即 flip_x=0,flip_y=0。元素个数最多为4；y约束具体场景要和 piece中的rot、flip_x、flip_y属性有相关约束，详细要求见备注4；

表 4-19 y

参数	是否必选	参数类型	描述
flip_x	是	Integer 枚举	裁片是否沿x轴翻转，0表示不翻转，1表示翻转。（0,1）
flip_y	是	Integer 枚举	裁片是否沿y轴翻转，0表示不翻转，1表示翻转。（0,1）
min_y	是	Double	裁片的包络矩形下边界在布料中的最小y坐标；保留1位小数，若超过将直接进行截断；该值要小于布料宽度。[0, 100000]
max_y	是	Double	片的包络矩形上边界在布料中的最大y坐标；保留1位小数，若超过将直接进行截断；该值要大于 min_y，并且小于等于布料宽度。（0, 100000]

表 4-20 Hole

参数	是否必选	参数类型	描述
polygon_index	是	Integer	实体多边形（裁片轮廓）在”polygon”向量中的index。（>=0）
gap_index	否	Integer	间隔多边形（加上最小间隔的裁片轮廓）在”polygon”向量中的index，-1表示没有间隔。备用，当前不支持。（>=-1）
id	是	Integer	-1表示真的瑕疵，>=0表示实际上被锁定的裁片；对于锁定裁片，id从0开始，注意，此时 piece->id值需要顺延Hole->id值。（>=-1）

表 4-21 Constraints

参数	是否必选	参数类型	描述
grid	否	Object	定义单一裁片自身的对格约束。 备用，当前不支持。
grid2	否	Object	定义两个裁片之间的对格约束。 备用，当前不支持。
position	否	表3-6 object	相邻约束，表征多个裁片的相对 位置关系近似不变，使得多个裁 片的参考点在指定的矩形区域 内。

表 4-22 Position

参数	是否必选	参数类型	描述
x	是	Double	相邻约束中矩形区域的x轴长 度；保留1位小数，若超过将直 接进行截断。
y	是	Double	定义两个裁片之间的对格约束。 备用，当前不支持。
piece	否	Array of 表7 objects	相邻约束，表征多个裁片的相对 位置关系近似不变，使得多个裁 片的参考点在指定的矩形区域 内。

表 4-23 Piece

参数	是否必选	参数类型	描述
index	是	Integer	裁片id。
x	是	Double	裁片参考点x坐标；保留1位小 数，若超过将直接进行截断。 [-100000, 100000]
y	是	Double	裁片参考点y坐标；保留1位小 数，若超过将直接进行截断。 [-100000, 100000]

响应参数

状态码： 201

表 4-24 响应 Body 参数

参数	参数类型	描述
task_id	String	任务编号。

请求示例

```
{
  "input_json": {
    "id": "fu",
    "order": "fu",
    "name": "fu",
    "time": 1,
    "gap_mode": 0,
    "overlap": 0,
    "fabric": {
      "width": 38,
      "grid_x": 0,
      "grid_y": 0
    },
    "polygon": [
      {
        "index": 0,
        "profile": [
          0.0,
          0.0,
          10.0,
          0.0,
          10.0,
          10.0,
          0.0,
          10.0
        ]
      },
      {
        "index": 1,
        "profile": [
          0.0,
          0.0,
          10.0,
          0.0,
          10.0,
          10.0,
          0.0,
          10.0
        ]
      },
      {
        "index": 2,
        "profile": [
          0.0,
          0.0,
          14.0,
          0.0,
          14.0,
          9.0,
          0.0,
          9.0
        ]
      },
      {
        "index": 3,
        "profile": [
          0.0,
          0.0,
```

```
        14.0,  
        0.0,  
        7.0,  
        7.0  
    ]  
  },  
  {  
    "index": 4,  
    "profile": [  
      0.0,  
      9.0,  
      0.0,  
      0.0,  
      14.0,  
      9.0  
    ]  
  },  
  {  
    "index": 5,  
    "profile": [  
      0.0,  
      0.0,  
      14.0,  
      0.0,  
      14.0,  
      14.0,  
      0.0,  
      14.0  
    ]  
  },  
  {  
    "index": 6,  
    "profile": [  
      0.0,  
      0.0,  
      10.0,  
      4.0,  
      10.0,  
      9.0,  
      0.0,  
      9.0  
    ]  
  },  
  {  
    "index": 7,  
    "profile": [  
      0.0,  
      0.0,  
      5.0,  
      0.0,  
      5.0,  
      9.0,  
      0.0,  
      9.0  
    ]  
  },  
  {  
    "index": 8,  
    "profile": [  
      0.0,  
      0.0,  
      14.0,  
      0.0,  
      14.0,  
      14.0  
    ]  
  },  
  {  
    "index": 9,
```

```
    "profile": [
      0.0,
      0.0,
      10.0,
      0.0,
      10.0,
      10.0,
      0.0,
      14.0
    ]
  },
  {
    "index": 10,
    "profile": [
      0.0,
      8.0,
      4.0,
      0.0,
      8.0,
      8.0
    ]
  },
  {
    "index": 11,
    "profile": [
      0.0,
      0.0,
      14.0,
      0.0,
      7.0,
      12.0
    ]
  }
],
"piece": [
  {
    "id": 0,
    "polygon_index": 0,
    "gap_index": -1,
    "hole_index": -1,
    "tilted": 0,
    "gap": 0,
    "pose_group": -1,
    "column_group": -1,
    "rot": 90,
    "flip_x": 0,
    "flip_y": 0,
    "remark": "no Y constraints",
    "y": []
  },
  {
    "id": 1,
    "polygon_index": 1,
    "gap_index": -1,
    "hole_index": -1,
    "tilted": 0,
    "gap": 0,
    "pose_group": -1,
    "column_group": -1,
    "rot": 90,
    "flip_x": 0,
    "flip_y": 0,
    "remark": "no Y constraints",
    "y": []
  },
  {
    "id": 2,
    "polygon_index": 2,
    "gap_index": -1,
```

```
"hole_index": -1,
"tilted": 0,
"gap": 0,
"pose_group": -1,
"column_group": -1,
"rot": 90,
"flip_x": 0,
"flip_y": 0,
"remark": "no Y constraints",
"y": []
},
{
  "id": 3,
  "polygon_index": 3,
  "gap_index": -1,
  "hole_index": -1,
  "tilted": 0,
  "gap": 0,
  "pose_group": -1,
  "column_group": -1,
  "rot": 90,
  "flip_x": 0,
  "flip_y": 0,
  "remark": "no Y constraints",
  "y": []
},
{
  "id": 4,
  "polygon_index": 4,
  "gap_index": -1,
  "hole_index": -1,
  "tilted": 0,
  "gap": 0,
  "pose_group": -1,
  "column_group": -1,
  "rot": 90,
  "flip_x": 0,
  "flip_y": 0,
  "remark": "no Y constraints",
  "y": []
},
{
  "id": 5,
  "polygon_index": 5,
  "gap_index": -1,
  "hole_index": -1,
  "tilted": 0,
  "gap": 0,
  "pose_group": -1,
  "column_group": -1,
  "rot": 90,
  "flip_x": 0,
  "flip_y": 0,
  "remark": "no Y constraints",
  "y": []
},
{
  "id": 6,
  "polygon_index": 6,
  "gap_index": -1,
  "hole_index": -1,
  "tilted": 0,
  "gap": 0,
  "pose_group": -1,
  "column_group": -1,
  "rot": 90,
  "flip_x": 0,
  "flip_y": 0,
  "remark": "no Y constraints",
```



```
    "y": []
  },
  {
    "id": 7,
    "polygon_index": 7,
    "gap_index": -1,
    "hole_index": -1,
    "tilted": 0,
    "gap": 0,
    "pose_group": -1,
    "column_group": -1,
    "rot": 90,
    "flip_x": 0,
    "flip_y": 0,
    "remark": "no Y constraints",
    "y": []
  },
  {
    "id": 8,
    "polygon_index": 8,
    "gap_index": -1,
    "hole_index": -1,
    "tilted": 0,
    "gap": 0,
    "pose_group": -1,
    "column_group": -1,
    "rot": 90,
    "flip_x": 0,
    "flip_y": 0,
    "remark": "no Y constraints",
    "y": []
  },
  {
    "id": 9,
    "polygon_index": 9,
    "gap_index": -1,
    "hole_index": -1,
    "tilted": 0,
    "gap": 0,
    "pose_group": -1,
    "column_group": -1,
    "rot": 90,
    "flip_x": 0,
    "flip_y": 0,
    "remark": "no Y constraints",
    "y": []
  },
  {
    "id": 10,
    "polygon_index": 10,
    "gap_index": -1,
    "hole_index": -1,
    "tilted": 0,
    "gap": 0,
    "pose_group": -1,
    "column_group": -1,
    "rot": 90,
    "flip_x": 0,
    "flip_y": 0,
    "remark": "no Y constraints",
    "y": []
  },
  {
    "id": 11,
    "polygon_index": 11,
    "gap_index": -1,
    "hole_index": -1,
    "tilted": 0,
    "gap": 0,
```

```
        "pose_group": -1,  
        "column_group": -1,  
        "rot": 90,  
        "flip_x": 0,  
        "flip_y": 0,  
        "remark": "no Y constraints",  
        "y": []  
    }  
],  
"hole": [],  
"constraints": {}  
}
```

响应示例

状态码： 201

OK

```
{  
  "task_id": "ee47xxxxxc211edba900255acxxxxxx"  
}
```

4.2.2 查询任务详情

功能介绍

获取指定任务的结果详情。

URI

GET /v1/{project_id}/optverse/2dcut/irregular-textile/tasks/{task_id}

表 4-25 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	用户项目ID，获取方法请参见获取项目ID。
task_id	是	String	任务编号

请求参数

表 4-26 请求 Header 参数

参数	是否必选	参数类型	描述
X-Apig-AppCode	是	String	用户凭证。

响应参数

状态码： 200

表 4-27 响应 Body 参数

参数	参数类型	描述
task_id	String	任务编号。
input_json	String	输入数据，当前为null
obs_input_url	String	Obs桶中输入文件地址，当前为null
obs_output_url	String	Obs桶中输出文件地址，当前为null
status	String	任务运行状态。（ Pending,Running, Failed, Succeeded ）
progress	String	任务进度，当前为null
output_json	Array of 表4 objects	任务处理结果，json格式，任务状态为finished时携带。具体数据结构参见IrregularTextileResultDetail
output_file_base64	String	当前为null
start_time	String	任务开始时间
end_time	String	任务结束时间
create_time	String	任务创建时间

表 4-28 IrregularTextileResultDetail

参数	参数类型	描述
id	String	任务编号。
time	Double	任务运行时长，单位秒。
ratio	Double	排版利用率
order	String	订单名。
length	Double	使用面料长度。
pieces	Array of 表5 objects	裁片排版位置定义。

表 4-29 Piece

参数	参数类型	描述
id	String	裁片id。
angle	Double	裁片旋转角度。
flip_x	Integer	裁片是否沿x轴翻转。
flip_y	Integer	裁片是否沿y轴翻转。
box	Array of 表6 objects	裁片先旋转再翻转后放入指定位置形成的包络矩形，由矩形左底点和右上点表示。

表 4-30 box

参数	参数类型	描述
x	Double	二维点x轴坐标。
y	Double	二维点y轴坐标。

响应示例

状态码： 200

OK

```
{
  "task_id": "37210xxxxxx511edb57c0255acxxxxxx",
  "input_json": null,
  "obs_input_url": null,
  "obs_output_url": null,
  "status": "Succeeded",
  "progress": null,
  "output_json": {
    "gap_mode": 0,
    "id": "fu",
    "length": 14.0,
    "name": "fu",
    "order": "fu",
    "pieces": [
      {
        "angle": 180.0,
        "box": [
          {
            "x": 1.110223024560533e-16,
            "y": 188.54000844217342
          },
          {
            "x": 10.0,
            "y": 202.54000844217342
          }
        ],
        "flip_x": 0,
        "flip_y": 0,
        "id": 9
      }
    ]
  }
}
```

```

    "angle": 90.0,
    "box": [
      {
        "x": 1.110223024560533e-16,
        "y": 180.54018862422214
      },
      {
        "x": 9.0,
        "y": 190.54018862422214
      }
    ],
    "flip_x": 0,
    "flip_y": 0,
    "id": 6
  },
  {
    "angle": 90.0,
    "box": [
      {
        "x": 1.1102230246251565e-16,
        "y": 251.65141900241923
      },
      {
        "x": 9.0,
        "y": 265.65141900241923
      }
    ],
    "flip_x": 0,
    "flip_y": 0,
    "id": 2
  },
  {
    "angle": 0.0,
    "box": [
      {
        "x": 1.110223024560533e-16,
        "y": 216.54000844217342
      },
      {
        "x": 5.0,
        "y": 225.54000844217342
      }
    ],
    "flip_x": 0,
    "flip_y": 0,
    "id": 7
  },
  {
    "angle": 180.0,
    "box": [
      {
        "x": 1.1101552619893762e-16,
        "y": 24.0
      },
      {
        "x": 8.0,
        "y": 32.0
      }
    ],
    "flip_x": 0,
    "flip_y": 0,
    "id": 10
  },
  {
    "angle": 270.0,
    "box": [
      {
        "x": 5.551115123125783e-17,
        "y": 0.0
      }
    ]
  }

```

```
    },
    {
      "x": 10.0,
      "y": 10.0
    }
  ],
  "flip_x": 0,
  "flip_y": 0,
  "id": 1
},
{
  "angle": 90.0,
  "box": [
    {
      "x": 0.0,
      "y": 10.0
    },
    {
      "x": 12.0,
      "y": 24.0
    }
  ],
  "flip_x": 0,
  "flip_y": 0,
  "id": 11
},
{
  "angle": 90.0,
  "box": [
    {
      "x": 1.1102230246251565e-16,
      "y": 237.65161696798242
    },
    {
      "x": 14.0,
      "y": 251.65161696798242
    }
  ],
  "flip_x": 0,
  "flip_y": 0,
  "id": 5
},
{
  "angle": 180.0,
  "box": [
    {
      "x": 1.1102230246251565e-16,
      "y": 49.51013170928303
    },
    {
      "x": 10.0,
      "y": 59.51013170928303
    }
  ],
  "flip_x": 0,
  "flip_y": 0,
  "id": 0
},
{
  "angle": 270.0,
  "box": [
    {
      "x": 1.110223024560533e-16,
      "y": 202.54000844217342
    },
    {
      "x": 7.0,
      "y": 216.54000844217342
    }
  ]
}
```

```
    ],
    "flip_x": 0,
    "flip_y": 0,
    "id": 3
  },
  {
    "angle": 270.0,
    "box": [
      {
        "x": 1.1102230163533504e-16,
        "y": 25.020263418566074
      },
      {
        "x": 9.0,
        "y": 39.020263418566074
      }
    ],
    "flip_x": 0,
    "flip_y": 0,
    "id": 4
  },
  {
    "angle": 0.0,
    "box": [
      {
        "x": 1.110214554295684e-16,
        "y": 166.54018862422214
      },
      {
        "x": 14.0,
        "y": 180.54018862422214
      }
    ],
    "flip_x": 0,
    "flip_y": 0,
    "id": 8
  }
],
"ratio": 0.75,
"time": 1
}
"output_file_base64": null,
"start_time": "2023-10-16T13:24:14Z",
"end_time": "2023-10-16T13:25:18Z",
"create_time": "2023-10-16T13:24:13.858Z"
}
```

4.3 求解器

4.3.1 创建任务

功能介绍

创建求解任务

URI

POST /v1/{project_id}/optverse/solver/oroas-optvsolver/tasks

备注：公测阶段服务URL由华为云提供

请求参数

备注：公测阶段以下信息由华为云提供

表 4-31 请求参数

名称	类型	IN	必选	描述
X-Apig-AppCode	string	header	是	用户凭证
project_id	string	path	是	项目ID
service_group	string	path	是	服务类别,针对不同服务类场景, 用户侧不需单独赋值
service_type	string	path	是	子服务类型,针对不同服务, 用户侧不需单独赋值

备注：以下信息由用户提供

表 4-32 请求 Body 参数

名称	类型	必选	描述
input_json	表3	是	任务输入信息,json格式;每个子服务该对象结构不同
obs_input_url	表6	是	任务输入信息为文件格式,传入值为租户OBS对应的文件绝对路径，用户需保证求解器服务有该路径的读写权限。输入内容为标准数学模型文件，支持.lp、.mps、.mps.gz格式
obs_output_url	表7	否	任务输出信息为文件格式,传入值为租户OBS对应的待存储路径前缀(和input_url成对使用),文件名服务端固定用task_id命名。输出格式为.sol格式

备注：以下信息由用户提供

表 4-33 input_json

名称	类型	必选	描述
parameters	表4	否	任务输入信息,json格式;每个子服务该对象结构不同。置空则代表使用默认参数

备注：以下信息由用户提供

表 4-34 parameters

名称	类型	必选	描述
name	string	是	求解器参数名称
value	int/ double	是	求解器参数值

公测阶段求解器开放参数如下：

表 4-35 parameters

名称	类型	必选	描述
method	int	否	求解算法。[0,2] <ul style="list-style-type: none">0：自动选择，默认值1：原始单纯形法2：对偶单纯形法
timeLimit	double	否	求解时间上限，单位为秒。取值范围[0.0, 1e+20] (1e+20)
gap	double	否	仅对MIP问题有效。[0, 1]，默认值为0.0

表 4-36 obs_input_url

名称	类型	必选	描述
bucket	string	是	输入数据obs桶名
object	string	是	输入数据绝对路径

表 4-37 obs_output_url

名称	类型	必选	描述
bucket	string	是	输出数据obs桶名
object	string	是	输出数据绝对路径

请求 Body 体示例

```
{
  "input_json": {
    "parameters": [
      {
        "name": "timeLimit",
        "value": 60.0
      }
    ]
  }
}
```

```
    },
    {
      "name": "gap",
      "value": 0.0
    }
  ]
},
"obs_input_url": {
  "bucket": "example_bucket",
  "object": "path/to/input/example.mps"
},
"obs_output_url": {
  "bucket": "example_bucket ",
  "object": "path/to/output"
}
}
```

响应参数

状态码： 201

创建成功响应信息

表 4-38 响应 Header 参数

参数	参数类型	描述
task_id	String	任务编号

状态码： 400

客户端错误响应信息

表 4-39 响应 Header 参数

参数	参数类型	描述
无	String	响应错误信息，详见错误码列表

4.3.2 查询任务

功能介绍

查询求解任务

URI

GET /v1/{project_id}/optverse/solver/oroas-optvsolver/tasks/{task_id}

备注: task_id 由用户提供

请求参数

备注：公测阶段以下信息由华为云提供

表 4-40 请求参数

名称	类型	IN	必选	描述
X-Apig-AppCode	string	header	是	用户凭证
project_id	string	path	是	项目ID
service_group	string	path	是	服务类别,针对不同服务类场景, 用户侧不需单独赋值
service_type	string	path	是	子服务类型,针对不同服务, 用户侧不需单独赋值
task_id	string	path	是	待查询任务id

响应参数

状态码： 200

创建成功响应信息

参数	参数类型	是否必选	描述
task_id	sring	是	任务id
input_json	string	是	求解器不涉及
obs_input_url	表6	是	求解器输入文件路径
obs_output_url	表7	是	求解器输出文件路径
status	string	是	任务状态，包括： <ul style="list-style-type: none">● Pending：任务等待处理中（排队中）● Running：任务正在运行● Failed：任务失败● Succeeded：任务成功

参数	参数类型	是否必选	描述
progress	string	是	返回求解器日志。求解状态在日志最后显示，可能的状态包括： - Unknown: 未知状态，可能处于问题读取过程中，或其他未定义状态 - Time limit reached: 达到时间上限 - Memory limit reached: 达到宿主机内存上限 - Optimal solution found: 找到最优解（或达到目标gap） - Problem is infeasible: 问题不可行 - Problem is unbounded: 问题无边界 - Problem is infeasible or unbounded: 问题不可行或无边界
output_json	string	是	求解器服务不涉及
output_file_base64	string	是	求解器服务不涉及
start_time	String	是	任务开始执行时间
end_time	String	是	任务结束时间
create_time	String	是	任务创建时间

状态码： 400

任务不存在

参数	参数类型	是否必选	描述
-	-	是	任务{0}不存在

响应结果示例

```
{
  "task_id": "344fxxxx-0xxx-4xxx-a248-ce26cff7xxxx",
  "input_json": null,
  "obs_input_url": {
    "bucket": "example_bucket",
    "object": " path/to/input/example.mps "
  },
  "obs_output_url": {
    "bucket": "example_bucket",
    "object": "path/to/output/344fd304-0937-4d25-a248-ce26cff78388_completeResult.sol"
  }
}
```

```
},  
  "status": "Running",  
  "progress": null,  
  "output_json": null,  
  "output_file_base64": null,  
  "start_time": "2023-00-00T07:19:35Z",  
  "end_time": null,  
  "create_time": "2023-00-00T07:19:33.607Z"  
}
```

5 附录

- 5.1 状态码
- 5.2 错误码

5.1 状态码

状态码如下所示。

状态码	编码	状态说明
100	Continue	继续请求。 这个临时响应用来通知客户端，它的部分请求已经被服务器接收，且仍未被拒绝。
101	Switching Protocols	切换协议。只能切换到更高级的协议。 例如，切换到HTTPS的新版本协议。
200	OK	服务器已成功处理了请求。
201	Created	创建类的请求完全成功。
202	Accepted	已经接受请求，但未处理完成。
203	Non-Authoritative Information	非授权信息，请求成功。
204	No Content	请求完全成功，同时HTTP响应不包含响应体。 在响应OPTIONS方法的HTTP请求时返回此状态码。
205	Reset Content	重置内容，服务器处理成功。
206	Partial Content	服务器成功处理了部分GET请求。
300	Multiple Choices	多种选择。请求的资源可包括多个位置，相应可返回一个资源特征与地址的列表用于用户终端（例如：浏览器）选择。

状态码	编码	状态说明
301	Moved Permanently	永久移动，请求的资源已被永久的移动到新的 URI，返回信息会包括新的 URI。
302	Found	资源被临时移动。
303	See Other	查看其它地址，使用 GET 和 POST 请求查看。
304	Not Modified	所请求的资源未修改，服务器返回此状态码时，不会返回任何资源。
305	Use Proxy	所请求的资源必须通过代理访问。
306	Unused	已经被废弃的 HTTP 状态码。
400	Bad Request	非法请求。 建议直接修改该请求，不要重试该请求。
401	Unauthorized	在客户端提供认证信息后，返回该状态码，表明服务端指出客户端所提供的认证信息不正确或非法。
402	Payment Required	保留请求。
403	Forbidden	请求被拒绝访问。 返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。
404	Not Found	所请求的资源不存在。 建议直接修改该请求，不要重试该请求。
405	Method Not Allowed	请求中带有该资源不支持的方法。 建议直接修改该请求，不要重试该请求。
406	Not Acceptable	服务器无法根据客户端请求的内容特性完成请求。
407	Proxy Authentication Required	请求要求代理的身份认证，与 401 类似，但请求者应当使用代理进行授权。
408	Request Timeout	服务器等候请求时发生超时。 客户端可以随时再次提交该请求而无需进行任何更改。
409	Conflict	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
410	Gone	客户端请求的资源已经不存在。 返回该状态码，表明请求的资源已被永久删除。

状态码	编码	状态说明
411	Length Required	服务器无法处理客户端发送的不带Content-Length的请求信息。
412	Precondition Failed	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。
413	Request Entity Too Large	由于请求的实体过大，服务器无法处理，因此拒绝请求。为防止客户端的连续请求，服务器可能会关闭连接。如果只是服务器暂时无法处理，则会包含一个Retry-After的响应信息。
414	Request URI Too Long	请求的URI过长（URI通常为网址），服务器无法处理。
415	Unsupported Media Type	服务器无法处理请求附带的媒体格式。
416	Requested Range Not Satisfiable	客户端请求的范围无效。
417	Expectation Failed	服务器无法满足Expect的请求头信息。
422	Unprocessable Entity	请求格式正确，但是由于含有语义错误，无法响应。
429	Too Many Requests	表明请求超出了客户端访问频率的限制或者服务端接收到多于它能处理的请求。建议客户端读取相应的Retry-After首部，然后等待该首部指出的时间后再重试。
500	Internal Server Error	表明服务端能被请求访问到，但是不能理解用户的请求。
501	Not Implemented	服务器不支持请求的功能，无法完成请求。
502	Bad Gateway	充当网关或代理的服务器，从远端服务器接收到了一个无效的请求。
503	Service Unavailable	被请求的服务无效。 建议直接修改该请求，不要重试该请求。
504	Gateway Timeout	请求在给定的时间内无法完成。客户端仅在为请求指定超时（Timeout）参数时会得到该响应。
505	HTTP Version Not Supported	服务器不支持请求的HTTPS协议的版本，无法完成处理。

5.2 错误码

当调用出错时，HTTP请求返回一个4xx或5xx的HTTP状态码。返回的消息体中是具体的错误代码及错误信息。在调用方找不到错误原因时，可以联系企业技术人员，并提供错误码，以便我们尽快帮您解决问题。

返回体格式

当接口调用出错时，会返回错误码及错误信息说明，错误响应的Body体格式如下所示

```
{
  "error_msg": "xxx",
  "error_code": "JOB_MANAGER.12050009"
}
```

其中，error_code表示错误码，error_msg表示错误描述信息。

表 5-1 异常响应说明

名称	参数类型	说明
error_code	String	错误码。
error_msg	String	错误详细信息。

错误码说明

当您调用API时，如果遇到“APIGW”开头的错误码，请参见[API网关错误码](#)进行处理。

表 5-2 错误码

错误码	参数类型	错误信息
JOB_MANAGER.12010001	基础错误	创建应用失败: {0}
JOB_MANAGER.12010002	基础错误	校验Token失败
JOB_MANAGER.12010003	基础错误	应用{0}不存在
JOB_MANAGER.12010004	基础错误	获取app详情失败: {0}
JOB_MANAGER.12010005	基础错误	已授权App: {0}
JOB_MANAGER.12010006	基础错误	删除App失败: {0}
JOB_MANAGER.12010007	基础错误	App数量超过限制: [{0}]
JOB_MANAGER.12010008	基础错误	App已经存在, name: [{0}]
JOB_MANAGER.12010009	基础错误	当App下还存在Pending状态任务时，并发数不能设置为0, 算法授权不能被取消

错误码	参数类型	错误信息
JOB_MANAGER.12010010	基础错误	应用{0}认证失败
JOB_MANAGER.12020001	基础错误	任务{0}不存在
JOB_MANAGER.12020002	基础错误	任务{0}已经结束
JOB_MANAGER.12020003	基础错误	任务类型{0}不存在
JOB_MANAGER.12020004	基础错误	服务类型{0}未授权
JOB_MANAGER.12020005	基础错误	App[{0}]的服务[{1}]并发数超出限额[{2}]
JOB_MANAGER.12020006	基础错误	App[{0}]的服务[{1}]并发数为0,不能创建任务
JOB_MANAGER.12020007	基础错误	服务[{0}]总并发数超出限额[{1}]
JOB_MANAGER.12020008	基础错误	资源不足
JOB_MANAGER.12020009	基础错误	任务输入校验失败, {0}
JOB_MANAGER.12020010	基础错误	鉴权失败: {0}
JOB_MANAGER.12020011	基础错误	任务[{0}]正在被占用, 请稍后再试
JOB_MANAGER.12050001	基础错误	用户{0}没有访问权
JOB_MANAGER.12050002	基础错误	用户{0}不能给其他用户添加访问权限
JOB_MANAGER.12050003	基础错误	存在重复的项目ID: {0}
JOB_MANAGER.12050004	基础错误	用户{0}不能查询用户访问列表
JOB_MANAGER.12050005	基础错误	用户{0}未委托
JOB_MANAGER.12050006	基础错误	用户{0}委托的角色不正确
JOB_MANAGER.12050007	基础错误	用户{0}的没有{1}角色权限
JOB_MANAGER.12050008	基础错误	CBC回调接口鉴权失败
JOB_MANAGER.12050009	基础错误	用户{0}不是IAM主账号
JOB_MANAGER.12060001	基础错误	OBS桶: {0}不存在, 用户: {1}
JOB_MANAGER.12060002	基础错误	OBS对象: {0}不存在, 桶: {1}, 用户: {2}
JOB_MANAGER.12060003	基础错误	OBS访问失败, 桶: {0}, 对象: {1}, projectId: {2}

错误码	参数类型	错误信息
JOB_MANAGER.12060004	基础错误	创建obs客户端失败, 用户: {0}
JOB_MANAGER.12060005	基础错误	obs上传文件失败, 桶: {0}, 对象: {1}, 文件路径: {2}
JOB_MANAGER.12060006	基础错误	obs创建文件夹失败, 文件夹: {0}
JOB_MANAGER.12060007	基础错误	obs下载对象失败, 桶: {0}, 对象: {1}, 用户: {2}
JOB_MANAGER.12080001	基础错误	文件 {0} 压缩发生异常
JOB_MANAGER.12080002	基础错误	文件 {0} 解压发生异常
JOB_MANAGER.12080003	基础错误	压缩文件 {0} 中存在目录结构, 请提供单一模型文件压缩得到的.tar.gz文件
JOB_MANAGER.12080004	基础错误	文件路径仅支持大小写字母、数字、中划线(-)、下划线(_)、英文句号(.)、斜杠(/). 不符合路径{0},{1}
JOB_MANAGER.12090001	基础错误	该租户 {0} 禁止访问本服务