

腾讯云智能数智人 其他配套



腾讯云

【 版权声明 】

©2013–2026 腾讯云版权所有

本文档（含所有文字、数据、图片等内容）完整的著作权归腾讯云计算（北京）有限责任公司单独所有，未经腾讯云事先明确书面许可，任何主体不得以任何形式复制、修改、使用、抄袭、传播本文档全部或部分内容。前述行为构成对腾讯云著作权的侵犯，腾讯云将依法采取措施追究法律责任。

【 商标声明 】



及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。未经腾讯云及有关权利人书面许可，任何主体不得以任何方式对前述商标进行使用、复制、修改、传播、抄录等行为，否则将构成对腾讯云及有关权利人商标权的侵犯，腾讯云将依法采取措施追究法律责任。

【 服务声明 】

本文档意在向您介绍腾讯云全部或部分产品、服务的当时的相关概况，部分产品、服务的内容可能不时有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或默示的承诺或保证。

【 联系我们 】

我们致力于为您提供个性化的售前购买咨询服务，及相应的技术售后服务，任何问题请联系 4009100100或95716。

文档目录

其他配套

个人资产管理 API 文档

个人资产管理 API 概述

查询小样本形象列表

查询服务资产信息

查询声音资产信息

查询形象资产信息-查询主播

查询形象资产信息-查询主播下的所有形象

查询形象支持的动作列表

附录1-服务资产类型

附录2-情感风格

附录3-数智人类型

附录4-语种列表

接口对接常见问题

配置端 iPaaS 接入

产品介绍

基础版接入指引

高级版接入指引

API 接口主域名及鉴权

数智人 aPaas 接口调用方式

其他配套

个人资产管理 API 文档

个人资产管理 API 概述

最近更新时间：2024-11-11 15:12:11

本文档主要描述客户资产管理接口，客户可以通过该套接口查询到客户当前开通的数智人相关资产，包括**形象资产**、**声音资产**和**服务资产**，服务资产包括定制配额、播报小时报、播报并发以及交互并发等。

接口调用环境

正式环境：`https://gw.tvs.qq.com`

接口调用方式

具体内容请参考 [数智人 aPaas 接口调用方式](#)

查询小样本形象列表

最近更新时间：2026-02-10 18:50:02

接口说明

查询客户拥有的小样本形象列表。

调用协议

HTTPS + JSON

POST /v2/ivh/crmserver/customerassetsservice/describesmallsampleimage

Header Content-Type: application/json;charset=utf-8

请求参数

参数	类型	必须	说明
PageIndex	int	是	当前页
PageSize	int	是	页面大小（最大不超过100）
AnchorCodes	Array of string	否	AnchorCode 筛选条件
VirtualmanKey	string	否	数智人 VirtualmanKey

返回参数

参数	类型	必须	说明
Virtualmans	Array of [SmallSampleImage]	是	小样本形象列表
Total	int	是	总数

SmallSampleImage

参数	类型	必须	说明
VirtualmanKey	string	是	数智人 VirtualmanKey，形象唯一标识
AnchorName	string	是	数智人主播名称
AnchorCode	string	是	数智人主播 Code

BackgroundChangeable	bool	是	是否支持换背景, true: 支持, false: 不支持
ClothesName	string	是	数智人服装
PoseName	string	是	数智人姿态
Resolution	string	是	数智人分辨率
HeaderImage	string	是	数智人头像图片 URL
PoseImage	string	是	数智人姿态图片 URL
ClothesImage	string	是	数智人服装图片 URL
SupportDriverTypes	Array of [string]	是	数智人支持的驱动类型 1. Text: 文本驱动 2. OriginalVoice: 原声音频驱动 3. ModulatedVoice: 变声音频驱动
EffectiveDate	string	是	资产生效时间, 格式: yyyy-MM-dd HH:mm:ss
EffectiveTime	int	是	资产生效时间戳, 单位毫秒
ExpireDate	string	是	资产到期时间, 格式: yyyy-MM-dd HH:mm:ss
ExpireTime	int	是	资产到期时间戳, 单位毫秒
VideoDuration	int	是	数智人累计制作视频时长, 单位为毫秒
VideoNum	int	是	数智人累计制作视频个数
OriginZoom	float	是	数智人初始缩放系数
OriginZoomStr	string型浮点数	是	数智人初始缩放系数, 值精确到小数点后3位
ReferenceVideoSegmentUrl	string	是	小样本选用视频片段 URL

请求示例

```
{
  "Header": {},
  "Payload": {
    "PageIndex": 1,
  }
}
```

```
"PageSize": 10,
"AnchorCodes": ["wl_pink_skirt_suit_stand"],
"VirtualmanKey": "d3bf659f364243ceacbe026710bdc25f"
}
}
```

返回示例

```
{
  "Header": {
    "RequestID": "6dbf3272a613d3a5cafa69f8bda562ef",
    "SessionID": "gze0f59a6c17311577017347309",
    "DialogID": "",
    "Code": 0,
    "Message": "ok"
  },
  "Payload": {
    "Total": 1,
    "Virtualmans": [
      {
        "AnchorCode": "wl_pink_skirt_suit_stand",
        "AnchorName": "微澜-粉短裙套装-全身站姿",
        "BackgroundChangeable": true,
        "ClothesImage": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/virtualman-config/00e92768-3d23-40d7-aa39-94df36a44a38-Mask_group粉短裙套装.png",
        "ClothesName": "粉短裙套装",
        "EffectiveDate": "2023-10-31 14:49:44",
        "EffectiveTime": 1698734984000,
        "ExpireDate": "2023-11-14 14:49:44",
        "ExpireTime": 1699944584000,
        "HeaderImage": "https://virtualhuman-cos-prod-1251316161.cos.ap-nanjing.myqcloud.com/virtualman-config/c0948a75-b4fd-461e-88e6-c15ef7c3b8f7-weilan_xyb_fenfuanqun.png",
        "OriginZoom": 0.332,
        "OriginZoomStr": "0.332",
        "PoseImage": "https://virtualhuman-cos-prod-1251316161.cos.ap-nanjing.myqcloud.com/virtualman-config/e75180c0-393b-4fec-9895-98717ddd3db5-weilan_xyb_fenfuanqun.png",
        "PoseName": "全身站姿",

```

```
        "ReferenceVideoSegmentUrl": "https://virtualhuman-cos-  
test-1251316161.cos.ap-nanjing.myqcloud.com/prod/resource-  
manager/small/57/159/4873/model_6859_20230812184617/pv.mp4",  
        "Resolution": "1280x720",  
        "SupportDriverTypes": [  
            "Text",  
            "OriginalVoice",  
            "ModulatedVoice"  
        ],  
        "VideoDuration": 120000,  
        "VideoNum": 2,  
        "VirtualmanKey": "d3bf659f364243ceacbe026710bdc25f"  
    }  
]  
}
```

查询服务资产信息

最近更新时间：2024-11-11 15:12:11

接口说明

查询客户拥有的服务资产信息，包括播报小时包、播报并发、交互并发、形象定制配额、声音复刻定制配额和音色小时包等。

调用协议

HTTPS + JSON

POST /v2/ivh/crmserver/customerassetsservice/getserviceasset

Header Content-Type: application/json;charset=utf-8

请求参数

参数	类型	必须	说明
ServiceType	string	否	服务资产类型，具体类型见 附录1 ，不传默认查询所有类型
PageIndex	int	是	当前页
PageSize	int	是	页面大小（最大不超过100）

返回参数

参数	类型	必须	说明
ServiceAssetInfos	Array of [ServiceAssetInfo]	是	服务资产列表
Total	int	是	总数

ServiceAssetInfo

参数	类型	必须	说明
ServiceType	string	是	服务资产类型
ExpireDate	string	是	服务资产到期时间，格式：yyyy-MM-dd HH:mm:ss
UniversalAssetInfoDetail	UniversalAssetInfoDetail	是	定制配额类资产详情
UniversalConcurrencyInfoDetail	UniversalConcurrencyInfoDetail	是	并发类资产详情
UniversalHourInfoDetail	UniversalHourInfoDetail	是	小时包类资产详情
InteractConcurrencyInfoDetail	[InteractConcurrencyInfoDetail]	否	2D小样本（专属）交互并发资产详情
BroadcastHourInfoDetail	[BroadcastHourInfoDetail]	否	2D小样本（专属）播报小时包资产详情
BroadcastConcurrencyInfoDetail	[BroadcastConcurrencyInfoDetail]	否	2D小样本（专属）播报并发资产详情
ImageAssetInfoDetail	[ImageAssetInfoDetail]	否	2D小样本（专属）形象定制配额资产详情
SmallSampleGeneral2DImageAssetInfoDetail	[SmallSampleGeneral2DImageAssetInfoDetail]	否	2D小样本（通用）形象定制配额资产详情
SmallSampleGeneral2DInteractConcurrencyInfoDetail	[SmallSampleGeneral2DInteractConcurrencyInfoDetail]	否	2D小样本（通用）交互并发资产详情
SmallSampleGeneral2DBroadcastHourInfoDetail	[SmallSampleGeneral2DBroadcastHourInfoDetail]	否	2D小样本（通用）播报小时包资产详情
SmallSampleGeneral2DBroadcastConcurrencyInfoDetail	[SmallSampleGeneral2DBroadcastConcurrencyInfoDetail]	否	2D小样本（通用）播报并发资产详情
RealMan2DInteractConcurrencyInfoDetail	[RealMan2DInteractConcurrencyInfoDetail]	否	2D精品交互并发资产详情
RealMan2DBroadcastHourInfoDetail	[RealMan2DBroadcastHourInfoDetail]	否	2D精品播报小时包资产详情
RealMan2DBroadcastConcurrencyInfoDetail	[RealMan2DBroadcastConcurrencyInfoDetail]	否	2D精品播报并发资产详情
RealMan3DInteractConcurrencyInfoDetail	[RealMan3DInteractConcurrencyInfoDetail]	否	3D写实交互并发资产详情
RealMan3DBroadcastHourInfoDetail	[RealMan3DBroadcastHourInfoDetail]	否	3D写实播报小时包资产详情
RealMan3DBroadcastConcurrencyInfoDetail	[RealMan3DBroadcastConcurrencyInfoDetail]	否	3D写实播报并发资产详情
TimbreAssetInfoDetail	[TimbreAssetInfoDetail]	否	声音复刻配额资产详情
TimbreHourInfoDetail	[TimbreHourInfoDetail]	否	音色小时包资产详情

UniversalAssetInfoDetail

参数	类型	必须	说明
CustomizationNum	int	是	定制数量
RemainNum	int	是	剩余数量

UniversalConcurrencyInfoDetail

参数	类型	必须	说明
Concurrency	int	是	并发数

UniversalHourInfoDetail

参数	类型	必须	说明
----	----	----	----

OrderMilliseconds	int	是	播报小时包时长，单位为毫秒
RemainMilliseconds	int	是	播报小时包剩余时长，单位为毫秒

InteractConcurrencyInfoDetail

参数	类型	必须	说明
Concurrency	int	是	并发数

BroadcastHourInfoDetail

参数	类型	必须	说明
OrderMilliseconds	int	是	播报小时包时长，单位为毫秒
RemainMilliseconds	int	是	播报小时包剩余时长，单位为毫秒

BroadcastConcurrencyInfoDetail

参数	类型	必须	说明
Concurrency	int	是	并发数

ImageAssetInfoDetail

参数	类型	必须	说明
CustomizationNum	int	是	形象定制数量
RemainNum	int	是	形象定制剩余数量

SmallSampleGeneral2DImageAssetInfoDetail

参数	类型	必须	说明
CustomizationNum	int	是	定制数量
RemainNum	int	是	剩余数量

SmallSampleGeneral2DInteractConcurrencyInfoDetail

参数	类型	必须	说明
Concurrency	int	是	并发数

SmallSampleGeneral2DBroadcastHourInfoDetail

参数	类型	必须	说明
OrderMilliseconds	int	是	播报小时包时长，单位为毫秒
RemainMilliseconds	int	是	播报小时包剩余时长，单位为毫秒

SmallSampleGeneral2DBroadcastConcurrencyInfoDetail

参数	类型	必须	说明
Concurrency	int	是	并发数

RealMan2DInteractConcurrencyInfoDetail

参数	类型	必须	说明
Concurrency	int	是	并发数

RealMan2DBroadcastHourInfoDetail

参数	类型	必须	说明
OrderMilliseconds	int	是	播报小时包时长，单位为毫秒
RemainMillisecond s	int	是	播报小时包剩余时长，单位为毫秒

RealMan2DBroadcastConcurrencyInfoDetail

参数	类型	必须	说明
OrderMilliseconds	int	是	播报小时包时长，单位为毫秒
RemainMillisecond s	int	是	播报小时包剩余时长，单位为毫秒

RealMan3DInteractConcurrencyInfoDetail

参数	类型	必须	说明
Concurrency	int	是	并发数

RealMan3DBroadcastHourInfoDetail

参数	类型	必须	说明
----	----	----	----

OrderMilliseconds	int	是	播报小时包时长，单位为毫秒
RemainMilliseconds	int	是	播报小时包剩余时长，单位为毫秒

RealMan3DBroadcastConcurrencyInfoDetail

参数	类型	必须	说明
Concurrency	int	是	并发数

TimbreAssetInfoDetail

参数	类型	必须	说明
CustomizationNum	int	是	声音复刻数量
RemainNum	int	是	声音复刻剩余数量

TimbreHourInfoDetail

参数	类型	必须	说明
OrderMilliseconds	int	是	音色小时包时长，单位为毫秒
RemainMillisecond s	int	是	音色小时包剩余时长，单位为毫秒

请求示例

```
{
  "Header": {},
  "Payload": {
    "ServiceType": "",
    "PageIndex": 1,
    "PageSize": 10
  }
}
```

返回示例

```
{
  "Header": {
    "RequestID": "890efbe5efdcb8f75a75a47a8a19eeac",
  }
}
```

```
"SessionID": "gz92439a3317313012788482967",
"DialogID": "",
"Code": 0,
"Message": "ok"
},
"Payload": {
  "ServiceAssetInfos": [
    {
      "BroadcastHourInfoDetail": {
        "OrderMilliseconds": 999999999,
        "RemainMilliseconds": 881318391
      },
      "ExpireDate": "2024-04-05 11:31:07",
      "ServiceType": "BroadcastHour",
      "UniversalHourInfoDetail": {
        "OrderMilliseconds": 999999999,
        "RemainMilliseconds": 881318391
      }
    },
    {
      "ExpireDate": "2024-04-11 14:59:46",
      "ImageAssetInfoDetail": {
        "CustomizationNum": 5,
        "RemainNum": 0
      },
      "ServiceType": "ImageAsset",
      "UniversalAssetInfoDetail": {
        "CustomizationNum": 5,
        "RemainNum": 0
      }
    }
  ],
  "Total": 2
}
```

查询声音资产信息

最近更新时间：2025-05-14 14:08:02

接口说明

查询客户拥有的声音资产信息，包括复刻声音和公共声音。

调用协议

HTTPS + JSON

POST /v2/ivh/crmserver/customerassetsservice/describetimbre

Header Content-Type: application/json;charset=utf-8

请求参数

参数	类型	必须	说明
PageIndex	int	是	当前页
PageSize	int	是	页面大小（最大不超过100）
TimbreKey	string	否	数智人音色 code，支持精确匹配
TimbreName	string	否	数智人音色名称，支持模糊匹配
PurchaseType	string	否	购买类型，Lease:租赁，Custom:定制
TimbreKeyList	Array of string	否	数智人音色 code 数组，当前参数与 TimbreKey 同时存在时，优先取当前字段

返回参数

参数	类型	必须	说明
TimbreAssets	Array of [TimbreAssetDetail]	是	声音资产列表
Total	int	是	总数

TimbreAssetDetail

参数	类型	必须	说明
TimbreKey	string	是	数智人音色 Code

TimbreName	string	是	数智人音色名称
TimbreSample	string	是	数智人音色试听样例 URL
TimbreDesc	string	是	数智人音色描述
SupportVirtualmanTypeCodes	Array of [string]	是	音色支持的数智人类型 Code，参考 附录3-数智人类型
SupportDriverTypes	Array of [string]	是	音色支持的驱动类型 1. Text: 文本驱动 2. OriginalVoice: 原声音频驱动 3. ModulatedVoice: 变声音频驱动
ExpireDate	string	否	资产到期时间，为空表示长期有效 格式: yyyy-MM-dd HH:mm:ss
AudioDuration	int	是	音频累计调用时长，单位为毫秒
AudioNum	int	是	音频累计调用数量
EmotionalCategories	Array of [string]	否	音色支持的情感风格列表，仅多情感音色才有数据，含义见 附录2-情感风格
TimbreGender	int	是	音色性别，1:男，2:女
PurchaseType	string	是	购买类型，Lease:租赁，Custom:定制
TimbreType	string	是	音色类型 Boutique: 精品 SampleBasic: 小样本基础版 SampleFast: 小样本极速版
TimbreLanguages	Array of [string]	否	音色支持的语种列表，仅部分音色支持多语种，含义见 附录4-语种列表

请求示例

```

{
  "Header": {},
  "Payload": {
    "PageIndex": 1,
    "PageSize": 10
  }
}
    
```

返回示例

```
{
  "Header": {
    "RequestID": "9ab72535a33d42efea712fb19aef0971",
    "SessionID": "gze0f59a6c17313015362017196",
    "DialogID": "",
    "Code": 0,
    "Message": "ok"
  },
  "Payload": {
    "TimbreAssets": [
      {
        "AudioDuration": 221587,
        "AudioNum": 11,
        "EmotionalCategories": [
          "6",
          "0",
          "1",
          "2",
          "3",
          "4",
          "5",
          "7",
          "8",
          "9"
        ],
        "ExpireDate": "2025-09-01 18:00:00",
        "PurchaseType": "Lease",
        "SupportDriverTypes": [
          "Text",
          "OriginalVoice",
          "ModulatedVoice"
        ],
        "SupportVirtualmanTypeCodes": [
          "real_man_2d",
          "small_sample_2d",
          "real_man_3d",
          "half_realistic_3d",
          "cartoon_3d",

```

```
        "small_sample_general_2d",
        "small_sample_4k_2d",
        "small_sample_photo_2d"
    ],
    "TimbreDesc": "温柔磁性",
    "TimbreGender": 2,
    "TimbreKey": "female_1",
    "TimbreLanguages": [],
    "TimbreName": "女声1",
    "TimbreSample": "https://virtualhuman-cos-prod-1251316161.cos.ap-nanjing.myqcloud.com/virtualman-config/6bf3915e-fcc2-49b7-8fe7-650e1cbf6633-10001_2_33_2D_demo_nv1.wav",
    "TimbreType": "Boutique"
},
{
    "AudioDuration": 9389,
    "AudioNum": 2,
    "EmotionalCategories": [],
    "ExpireDate": "",
    "PurchaseType": "Lease",
    "SupportDriverTypes": [
        "Text",
        "OriginalVoice"
    ],
    "SupportVirtualmanTypeCodes": [
        "real_man_2d",
        "small_sample_2d",
        "real_man_3d",
        "half_realistic_3d",
        "cartoon_3d",
        "small_sample_general_2d",
        "small_sample_4k_2d",
        "small_sample_photo_2d"
    ],
    "TimbreDesc": "对话风格",
    "TimbreGender": 1,
    "TimbreKey": "male_11",
    "TimbreLanguages": [],
    "TimbreName": "男声11",
    "TimbreSample": "https://virtualhuman-cos-prod-1251316161.cos.ap-nanjing.myqcloud.com/virtualman-config/84fcec66-885e-
```

```
4b9d-a66e-7fe1fab3394-0c9a40260f20447ab772e2e02d46e1ae.wav",  
    "TimbreType": "Boutique"  
  }  
],  
  "Total": 2  
}  
}
```

查询形象资产信息-查询主播

最近更新时间：2025-12-10 11:09:21

接口说明

查询客户拥有的形象资产信息，查询主播列表。

调用协议

HTTPS + JSON

POST /v2/ivh/crmserver/customerassetsservice/getanchor

Header Content-Type: application/json;charset=utf-8

请求参数

参数	类型	必须	说明
VirtualmanType Code	string	是	数智人类型 Code，参考 附录3-数智人类型
AnchorCodes	Array of [string]	否	数智人主播 Code
PurchaseType	string	否	购买类型，Lease:租赁，Custom:定制
PageIndex	int	是	当前页
PageSize	int	是	页面大小（最大不超过100）

返回参数

参数	类型	必须	说明
Virtualmans	Array of [Virtualman]	是	主播列表
Total	int	是	总数

Virtualman

参数	类型	必须	说明
AnchorName	string	是	数智人主播名称
AnchorCode	string	是	数智人主播 Code

VirtualmanType	string	是	数智人类型名称
VirtualmanType Code	string	是	数智人类型 Code
PurchaseType	string	是	购买类型, Lease:租赁, Custom:定制
HeaderImage	string	是	数智人头像 URL
Tag	string	是	数智人模型tag, 分三类: 1. Basic 2. Standard 3. Advanced 目前 tag 仅影响数智人横纵向位置功能
Gender	int	否	数智人性别, 0: 空, 1: 男, 2: 女

请求示例

```
{
  "Header": {},
  "Payload": {
    "VirtualmanTypeCode": "real_man_3d",
    "AnchorCodes": [],
    "PageIndex": 1,
    "PageSize": 10
  }
}
```

返回示例

```
{
  "Header": {
    "RequestID": "4ff85e77c7c5067cb9a79802879db144",
    "SessionID": "gze0f59a6c17313022465941770",
    "DialogID": "",
    "Code": 0,
    "Message": "ok"
  },
  "Payload": {
    "Virtualmans": [
      {

```

```
        "HeaderImage": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/resource-manager/3D_Cartoon/xiaowei/202306091551/header.png",
        "VirtualmanTypeCode": "real_man_3d",
        "VirtualmanType": "3D写实",
        "Tag": "Basic",
        "PurchaseType": "Custom",
        "AnchorCode": "xiaowei",
        "AnchorName": "小微",
        "Gender": 2
    }
],
"Total": 1
}
```

查询形象资产信息-查询主播下的所有形象

最近更新时间：2026-04-23 16:09:32

接口说明

查询客户拥有的形象资产信息，查询某主播下的形象列表。

调用协议

HTTPS + JSON

POST /v2/ivh/crmserver/customerassetsservice/getimagebyanchor

Header Content-Type: application/json;charset=utf-8

请求参数

参数	类型	必须	说明
VirtualmanType Code	string	是	数智人类型 Code，参考 附录3-数智人类型
AnchorCodes	Array of [string]	否	数智人主播 Code
PageIndex	int	是	当前页
PageSize	int	是	页面大小（最大不超过100）
VirtualmanKey	string	否	数智人 VirtualmanKey
PurchaseType	string	否	购买类型，Lease：租赁，Custom：定制

返回参数

参数	类型	必须	说明
Virtualmans	Array of [ImageAssetDetail]	是	该主播下的形象列表 <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">说明： 当对应 VirtualmanTypeCode 下未根据接口参数查询到形象时，形象列表返回为空。</div>
Total	int	是	总数

ImageAssetDetail

参数	类型	必须	说明
AnchorName	string	是	数智人主播名称
AnchorCode	string	是	数智人主播 Code
BackgroundChangeable	bool	是	是否支持换背景, true: 支持, false: 不支持
VirtualmanType	string	是	数智人类型
VirtualmanTypeCode	string	是	数智人类型 Code
VirtualmanKey	string	是	数智人 VirtualmanKey
ClothesName	string	是	数智人服装名称
PoseName	string	是	数智人姿态
Resolution	string	是	数智人分辨率
HeaderImage	string	是	数智人头像图片 URL
PoseImage	string	是	数智人姿态图片 URL
ClothesImage	string	是	数智人服装图片 URL
SupportDriverTypes	Array of [string]	是	数智人支持的驱动类型: Text: 文本驱动 OriginalVoice: 原声音频驱动 ModulatedVoice: 变声音频驱动
SupportAnchorExtraParams	Array of [SupportAnchorExtraParam]	是	数智人可配置形象参数说明, 目前仅包含 3D 服装变色参数
VideoDuration	int64	是	数智人累计调用时长, 单位为毫秒
VideoNum	int64	是	数智人累计视频个数
ExpireDate	string	是	资产到期时间, 格式: yyyy-MM-dd HH:mm:ss
OriginZoom	string	是	缩放系数, 保留三位小数
ReferenceVideoSegmentUrl	string	是	选用视频片段

PurchaseType	string	是	购买类型，Lease:租赁，Custom:定制
SupportRenderTypes	Array of [string]	是	数智人支持的渲染类型： cloud: 云渲染 client: 端渲染
Gender	int	否	数智人性别，0: 空，1: 男，2: 女
ModelDownloadUrl	string	否	数智人端渲染模型资产下载链接 <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p>说明： 当入参 VirtualmanKey 不为空，对应形象支持端渲染时，在形象定制完成后30天内可获取到模型资产下载链接。</p> </div>

SupportAnchorExtraParam

参数	类型	必须	说明
Key	string	是	配置项名称
Description	string	是	配置项描述
Type	int	是	配置项的取值类型。1: 字符串，2: 数值，3: 布尔
Value	string[]	是	当前配置项，可配置的字段列表

请求示例

```

{
  "Header": {},
  "Payload": {
    "VirtualmanTypeCode": "small_sample_2d",
    "AnchorCodes": [
      "wl_pink_skirt_suit_stand"
    ],
    "PageIndex": 1,
    "PageSize": 10
  }
}

```

```
}
```

返回示例

```
{
  "Header": {
    "RequestID": "612258c65d2da9401710b06e16e0bed1",
    "SessionID": "gzfa9105ca17313024090292042",
    "DialogID": "",
    "Code": 0,
    "Message": "ok"
  },
  "Payload": {
    "Total": 2,
    "Virtualmans": [
      {
        "AnchorCode": "wl_pink_skirt_suit_stand",
        "AnchorName": "微澜-粉短裙套装-全身站姿",
        "BackgroundChangeable": true,
        "ClothesImage": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/virtualman-config/00e92768-3d23-40d7-aa39-94df36a44a38-Mask_group粉短裙套装.png",
        "ClothesName": "粉短裙套装",
        "ExpireDate": "2024-03-13 21:37:02",
        "HeaderImage": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/prod/resource-manager/small/57/159/4873/model_11131_20240429141943/header.png",
        "OriginZoom": "1.000",
        "PoseImage": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/prod/resource-manager/small/57/159/4873/model_11131_20240429141943/static_rgba.png",
        "PoseName": "全身站姿",
        "ReferenceVideoSegmentUrl": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/prod/resource-manager/small/57/159/4873/model_11131_20240429141943/pv.mp4",
        "Resolution": "1080x1920",
        "SupportAnchorExtraParams": [],
        "SupportDriverTypes": [
          "Text",
          "OriginalVoice",

```

```
        "ModulatedVoice"
    ],
    "VideoDuration": 0,
    "VideoNum": 0,
    "VirtualmanKey": "2cc66ae2950e4968880676a055013234",
    "VirtualmanType": "2D小样本(专属)",
    "VirtualmanTypeCode": "small_sample_2d",
    "PurchaseType": "Lease",
    "Gender": 2,
    "SupportRenderTypes": [
        "cloud"
    ]
},
{
    "AnchorCode": "wl_pink_skirt_suit_stand",
    "AnchorName": "微澜-粉短裙套装-全身站姿",
    "ClothesImage": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/virtualman-config/00e92768-3d23-40d7-aa39-94df36a44a38-Mask_group粉短裙套装.png",
    "ClothesName": "粉短裙套装",
    "ExpireDate": "2024-03-13 21:37:02",
    "HeaderImage": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/prod/resource-manager/small/57/159/4873/model_11131_20240429141943/header.png",
    "OriginZoom": "0.571",
    "PoseImage": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/prod/resource-manager/small/57/159/4873/model_11131_20240429141943/static_rgba.png",
    "PoseName": "全身站姿",
    "ReferenceVideoSegmentUrl": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/prod/resource-manager/small/57/159/4873/model_11131_20240429141943/pv.mp4",
    "Resolution": "1920x1080",
    "SupportAnchorExtraParams": [],
    "SupportDriverTypes": [
        "Text",
        "OriginalVoice",
        "ModulatedVoice"
    ],
    "VideoDuration": 0,
    "VideoNum": 0,
```

```
"VirtualmanKey": "fe880d9385304e688f9a8e469370fc90",
"VirtualmanType": "2D小样本(专属)",
"VirtualmanTypeCode": "small_sample_2d",
"PurchaseType": "Lease",
"Gender": 2,
"SupportRenderTypes": [
  "cloud"
]
}
]
}
}
```

查询形象支持的动作列表

最近更新时间：2024-11-19 11:30:22

接口说明

根据数智人VirtualmanKey，查询形象支持的动作列表。

⚠ 注意：

该接口只支持查询3d形象的动作。2d小样本形象不支持动作，调用该接口返回为空。

调用协议

HTTPS + JSON

POST /v2/ivh/crmserver/customerassetservice/getaction

Header Content-Type: application/json;charset=utf-8

请求参数

参数	类型	必须	说明
VirtualmanKey	string	是	已分配的数智人形象 Virtualmankey

返回参数

参数	类型	必须	说明
Actions	Array of [Action]	否	动作列表 <div style="border: 1px solid #00aaff; padding: 5px; margin-top: 10px;">ⓘ 说明： 当该形象无支持动作、或 VirtualmanKey 不存在时，动作列表返回为空。</div>

Action

参数	类型	必须	说明
ActionName	string	是	数智人动作名称
ActionCode	string	是	数智人动作 Code，对应 ssml 动作标签
ActionSample	string	是	数智人动作样例 URL

请求示例

```
{
  "Header": {},
  "Payload": {
    "VirtualmanKey": "8af9c7a960ab44c9bfe8a48deae6a250"
  }
}
```

返回示例

```
{
  "Header": {
    "RequestID": "28c713ede08f42afc058e4c57d7ecdc0",
    "SessionID": "gzfa9105ca17313027784787766",
    "DialogID": "",
    "Code": 0,
    "Message": "ok"
  },
  "Payload": {
    "Actions": [
      {
        "ActionCode": "2hands_forward1",
        "ActionSample": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/resource-manager/3d_virtualman/meta_yoyo/2023/display_videos/2hands_forward1.mp4",
        "ActionName": "双手指向前方1"
      },
      {
        "ActionCode": "admission",
        "ActionSample": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/resource-manager/3d_virtualman/meta_yoyo/2023/display_videos/admission.mp4",
        "ActionName": "入场"
      },
      {
        "ActionCode": "appearance",
        "ActionSample": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/resource-manager/3d_virtualman/meta_yoyo/2023/display_videos/appearance.mp4",

```

```
        "ActionName": "出场"
    },
    {
        "ActionCode": "bow",
        "ActionSample": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/resource-manager/3d_virtualman/meta_yoyo/2023/display_videos/bow.mp4",
        "ActionName": "弯腰鞠躬"
    },
    {
        "ActionCode": "waving_hand1",
        "ActionSample": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/resource-manager/3d_virtualman/meta_yoyo/2023/display_videos/waving_hand1.mp4",
        "ActionName": "单手挥手"
    },
    {
        "ActionCode": "welcome",
        "ActionSample": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/resource-manager/3d_virtualman/meta_yoyo/2023/display_videos/welcome.mp4",
        "ActionName": "欢迎"
    },
    {
        "ActionCode": "left_across_waist",
        "ActionSample": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/resource-manager/3d_virtualman/meta_yoyo/2023/display_videos/left_across_waist.mp4"
    },
    {
        "ActionName": "左手体侧滑动"
    },
    {
        "ActionCode": "left_side1",
        "ActionSample": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/resource-manager/3d_virtualman/meta_yoyo/2023/display_videos/left_side1.mp4",
        "ActionName": "左手指向左侧1"
    },
    {
        "ActionCode": "left_up2",
```

```
        "ActionSample": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/resource-manager/3d_virtualman/meta_yoyo/2023/display_videos/left_up2.mp4",
        "ActionName": "左手指向左上2"
    },
    {
        "ActionCode": "right_side1",
        "ActionSample": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/resource-manager/3d_virtualman/meta_yoyo/2023/display_videos/right_side1.mp4",
        "ActionName": "右手指向右侧1"
    },
    {
        "ActionCode": "right_slide_down",
        "ActionSample": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/resource-manager/3d_virtualman/meta_yoyo/2023/display_videos/right_slide_down.mp4",
        "ActionName": "右手体侧向下滑动"
    },
    {
        "ActionCode": "right_up2",
        "ActionSample": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/resource-manager/3d_virtualman/meta_yoyo/2023/display_videos/right_up2.mp4",
        "ActionName": "右手指向右上2"
    },
    {
        "ActionCode": "applaud",
        "ActionSample": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/resource-manager/3d_virtualman/meta_yoyo/2023/display_videos/applaud.mp4",
        "ActionName": "鼓掌"
    },
    {
        "ActionCode": "compliment_state3",
        "ActionSample": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/resource-manager/3d_virtualman/meta_yoyo/2023/display_videos/compliment_state3.mp4",
        "ActionName": "双手点赞"
    },
    {
```

```
{
  "ActionCode": "ok2",
  "ActionSample": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/resource-manager/3d_virtualman/meta_yoyo/2023/display_videos/ok2.mp4",
  "ActionName": "体侧比OK"
},
{
  "ActionCode": "show_love3",
  "ActionSample": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/resource-manager/3d_virtualman/meta_yoyo/2023/display_videos/show_love3.mp4",
  "ActionName": "单手比心"
},
{
  "ActionCode": "support",
  "ActionSample": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/resource-manager/3d_virtualman/meta_yoyo/2023/display_videos/support.mp4",
  "ActionName": "加油"
},
{
  "ActionCode": "turn_around",
  "ActionSample": "https://virtualhuman-cos-test-1251316161.cos.ap-nanjing.myqcloud.com/resource-manager/3d_virtualman/meta_yoyo/2023/display_videos/turn_around.mp4",
  "ActionName": "转圈"
}
]
}
```

附录1-服务资产类型

最近更新时间：2025-04-08 14:05:52

参数	说明
ImageAsset	2D小样本（专属）形象定制配额
InteractConcurrency	2D小样本（专属）交互并发
BroadcastHour	2D小样本（专属）播报小时包
BroadcastConcurrency	2D小样本（专属）播报并发
SmallSampleGeneral2DImageAsset	2D小样本（通用）形象定制配额
SmallSampleGeneral2DInteractConcurrency	2D小样本（通用）交互并发
SmallSampleGeneral2DBroadcastHour	2D小样本（通用）播报小时包
SmallSampleGeneral2DBroadcastConcurrency	2D小样本（通用）播报并发
SmallSample4K2DImageAsset	2D小样本（高精）形象定制配额
SmallSample4K2DInteractConcurrency	2D小样本（高精）交互并发
SmallSample4K2DBroadcastHour	2D小样本（高精）播报小时包
SmallSample4K2DBroadcastConcurrency	2D小样本（高精）播报并发
SmallSamplePhoto2DImageAsset	2D小样本（照片）形象定制配额
SmallSamplePhoto2DInteractConcurrency	2D小样本（照片）交互并发
SmallSamplePhoto2DBroadcastHour	2D小样本（照片）播报小时包
SmallSamplePhoto2DBroadcastConcurrency	2D小样本（照片）播报并发
NoTrainVideoBroadcastHour	2D免训练播报小时包（视频素材）
NoTrainVideoBroadcastConcurrency	2D免训练播报并发（视频素材）
RealMan2DInteractConcurrency	2D精品交互并发

RealMan2DBroadcastHour	2D精品播报小时包
RealMan2DBroadcastConcurrency	2D精品播报并发
RealMan3DInteractConcurrency	3D写实交互并发
RealMan3DBroadcastHour	3D写实播报小时包
RealMan3DBroadcastConcurrency	3D写实播报并发
HalfRealistic3DInteractConcurrency	3D半写实交互并发
HalfRealistic3DBroadcastHour	3D半写实播报小时包
HalfRealistic3DBroadcastConcurrency	3D半写实播报并发
Cartoon3DInteractConcurrency	3D卡通交互并发
Cartoon3DBroadcastHour	3D卡通播报小时包
Cartoon3DBroadcastConcurrency	3D卡通播报并发
TimbreAsset	声音复刻（基础版）定制配额
ZeroShotTimbreAsset	声音复刻（极速版）定制配额
TimbreHour	音色小时包
SmallSampleImageRenewalAsset	2D形象-专属口型在架服务配额
SmallSampleGeneralImageRenewalAsset	2D形象-通用口型在架服务配额
RealMan3DImageRenewalAsset	3D形象在架服务配额
TimbreRenewalAsset	音色定制在架服务配额

附录2-情感风格

最近更新时间：2024-08-23 10:02:15

参数	说明
0	高兴
1	新闻
2	客服
3	恐惧
4	诗歌
5	故事
6	中性
7	悲伤
8	广播
9	生气
tc_amaze	震惊
tc_angry	生气
tc_aojiao	傲娇
tc_call	客服
tc_disgusted	厌恶
tc_exciting	兴奋
tc_fear	恐惧
tc_happy	高兴
tc_jieshuo	解说
tc_neutral	中性
tc_news	新闻
tc_peaceful	平静

tc_poetry	诗歌
tc_radio	广播
tc_sad	悲伤
tc_sajiao	撒娇
tc_story	故事

附录3-数智人类型

最近更新时间：2026-04-24 09:13:58

参数	说明
real_man_2d	2D 精品
real_man_3d	3D 写实
half_realistic_3d	3D 半写实
cartoon_3d	3D 卡通
small_sample_2d	2D 小样本（专属）
small_sample_general_2d	2D 小样本（通用）
small_sample_4k_2d	2D 小样本（高精）
small_sample_photo_2d	2D 小样本（照片）
small_sample_lite_2d	2D 小样本端渲染形象（包含专属、通用、照片）

⚠ 注意：

如果需要查询端渲染相关的形象，数智人类型请选择 small_sample_lite_2d。

附录4-语种列表

最近更新时间：2025-03-12 17:03:53

参数	说明
am-ET	阿姆哈拉语(埃塞俄比亚)
ar-EG	阿拉伯语(埃及)
ar-SA	阿拉伯语(沙特阿拉伯)
bg-BG	保加利亚语(保加利亚)
bs-BA	波斯尼亚语(波斯尼亚和黑塞哥维那)
ca-ES	加泰罗尼亚语
cs-CZ	捷克语(捷克)
da-DK	丹麦语(丹麦)
de-AT	德语(奥地利)
de-CH	德语(瑞士)
de-DE	德语(德国)
el-GR	希腊语(希腊)
en-AU	英语(澳大利亚)
en-GB	英语(英国)
en-IE	英语(爱尔兰)
en-IN	英语(印度)
en-US	英语(美国)
es-ES	西班牙语(西班牙)
es-MX	西班牙语(墨西哥)
et-EE	爱沙尼亚语(爱沙尼亚)
eu-ES	巴斯克语
fi-FI	芬兰语(芬兰)

fil-PH	菲律宾语(菲律宾)
fr-BE	法语(比利时)
fr-CH	法语(瑞士)
fr-FR	法语(法国)
gl-ES	加利西亚语
he-IL	希伯来语(以色列)
hr-HR	克罗地亚语(克罗地亚)
hu-HU	匈牙利语(匈牙利)
id-ID	印度尼西亚语(印度尼西亚)
it-IT	意大利语(意大利)
ja-JP	日语(日本)
jv-ID	爪哇语(拉丁语、印度尼西亚)
ko-KR	韩语(韩国)
mk-MK	马其顿语(北马其顿)
ms-MY	马来语(马来西亚)
nb-NO	书面挪威语(挪威)
ne-NP	尼泊尔语(尼泊尔)
nl-BE	荷兰语(比利时)
nl-NL	荷兰语(荷兰)
pl-PL	波兰语(波兰)
ps-AF	普什图语(阿富汗)
pt-BR	葡萄牙语(巴西)
pt-PT	葡萄牙语(葡萄牙)
ro-RO	罗马尼亚语(罗马尼亚)
sk-SK	斯洛伐克语(斯洛伐克)

sl-SI	斯洛文尼亚语(斯洛文尼亚)
sv-SE	瑞典语(瑞典)
sw-KE	斯瓦希里语(肯尼亚)
th-TH	泰语(泰国)
tr-TR	土耳其语(土耳其)
uk-UA	乌克兰语(乌克兰)
vi-VN	越南语(越南)
yue-CN	中文(粤语, 简体)
zh-CN	中文(普通话, 简体)
zh-HK	中文(粤语, 繁体)
zh-TW	中文(台湾普通话, 繁体)
zu-ZA	祖鲁语(南非)

接口对接常见问题

最近更新时间：2024-08-23 10:29:43

问题描述	解决方式	备注
个人资产接口是否支持查询已制作视频记录明细	暂不支持，当前支持查询总视频个数和总耗时小时数。	-

配置端 iPaaS 接入 产品介绍

最近更新时间：2025-07-10 14:17:11

适用场景

iPaaS（Integration Platform as a Service）的模式，集成配置端（数智人平台）页面并进行主题色、品牌 LOGO 的配置后，即可在任一业务系统内直接享受到与原生数智人平台一致的云服务。

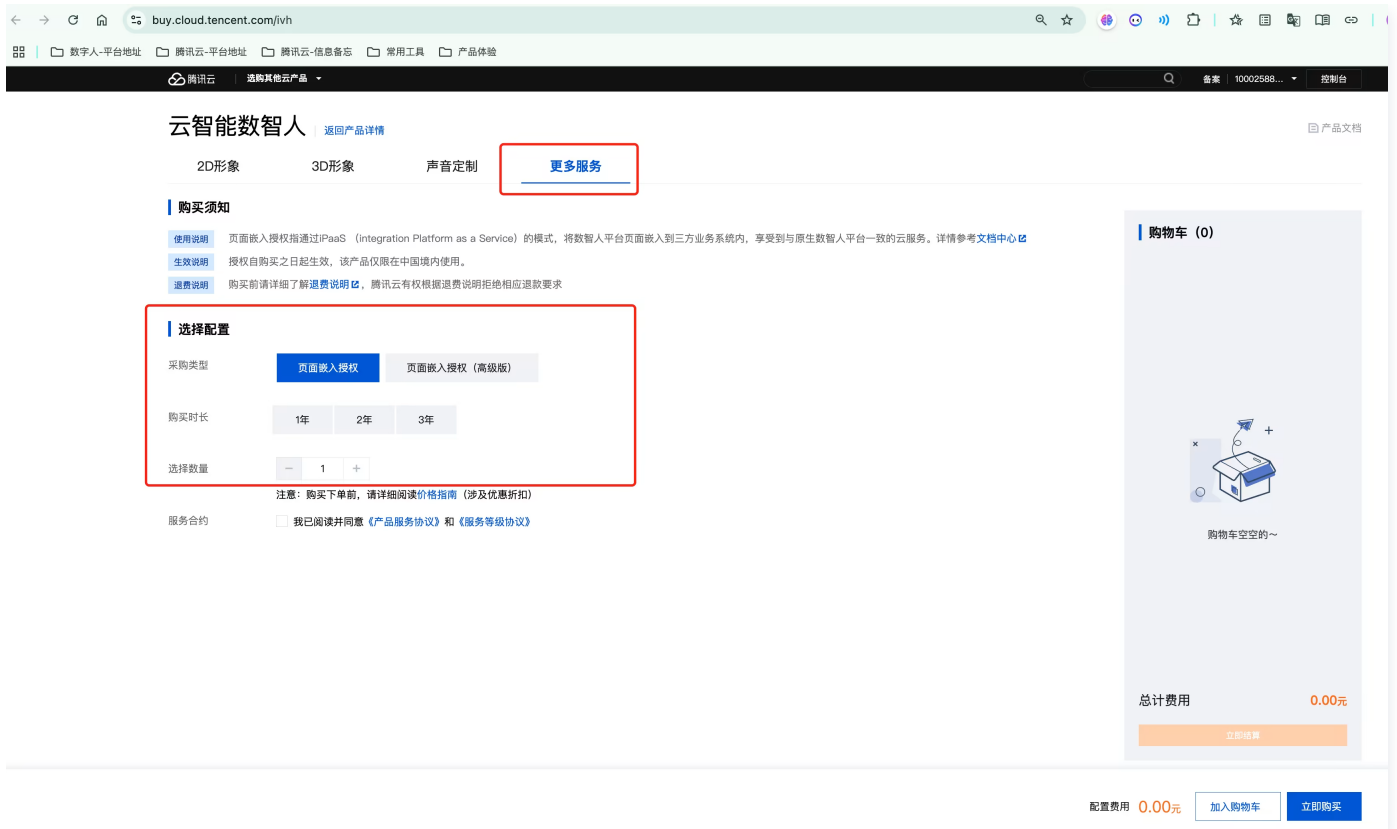
适合接入方

- 业务诉求：希望集成数智人平台核心配置能力、快速搭建业务系统的客户：
 - 例如，软件服务渠道商。通过 iPaaS 集成数智人平台模式，降低各级集成商接入的门槛，扶持更多伙伴快速起量。
 - 例如，需要基于客户内部业务平台去调用数智人服务的 ISV 集成方。客户无法接受独立系统或单点登录，期望深度集成融合到自己的管理系统中使用，则可通过 iPaaS 集成数智人平台模式，快速在客户已有系统中新增数智人服务模块。
- 研发能力：有前端集成能力。

接入流程

1. 与商务经理沟通下单 iPaaS 服务（基础版5万元/年，高级版30万元/年）。
2. 由交付架构师支持接入。提供您的业务回调信息到交付架构师处，进行接入授权服务开通。
3. 参考体验中心确认接入页面配置项。
4. 参考接入指引完成接入（一般情况“天”级别即可完成）。

下单地址及购买界面如下图所示：



基础版 VS 高级版对比

可通过以下表格快速获悉基础版、高级版的产品功能差异。如需更详细的体验了解，请参见下文 [基础版体验中心](#) 和 [高级版体验中心](#)。

产品能力		基础版	高级版
交付对象	iPaaS 用户界面	 <p>除菜单导航外页面内容的iframe 嵌入</p>	 <p>包含菜单导航在内的全页面嵌入</p>
	iPaaS 运营后台	✘	✓
页面个性化配置	更换界面主题色	✓	✓
	更换界面品牌及LOGO	✓	✓
	配置方式	代码配置	iPaaS 运营后台可视化配置

iPaaS 用户界面 > 数字人视频制作	文本驱动	✓	✓
	音频驱动	✓	✓
	多轨道编辑器	✗ 仅支持文本及 SSML 标签的方式制作一整段视频	✓ 支持插入背景、花字、贴纸、音乐、多媒体资源、转场、动效等能力，支持制作包含多段的视频
	PPT/PDF 导入	✗	✓
	数字人视频模板	✗	✓ 内置数字人视频模板，支持一键应用
iPaaS 用户界面 > 数字人交互会话	大模型对话驱动	✓ 需搭配购买腾讯智能体开发平台相关大模型对话服务	✗
	客服对话驱动	✓ 需搭配购买腾讯企点客服相关应用服务	✗
iPaaS 运营后台 > 分销管理	租户自组 SKU	✗ 仅可查询和直接使用腾讯云数字人的 PaaS 资产	✓ 支持租户自定义配置会员套餐、增值包等 SKU 及包含的用户权益项，可直接用于商业化售卖及交付请参见下文 产品特性 ，高级版运营后台功能清单。

基础版介绍

产品特性

被集成的基础版页面可直接满足以下特性：

序号	特性	具体描述
1	新页面可零/低成本嵌入	嵌入框架搭建好后，后续新增页面可以快速嵌入，无需引发增量研发工作。
2	页面内更新无感知	客户接入只需要验证 URL 跳转，每次页面内功能/UI 升级时 测试不需要全量回归验证，页面内功能升级可无感知。
3	iframe 方式嵌入	对指定的页面范围，能提供除了菜单导航外页面内容的 iframe 嵌入方式。

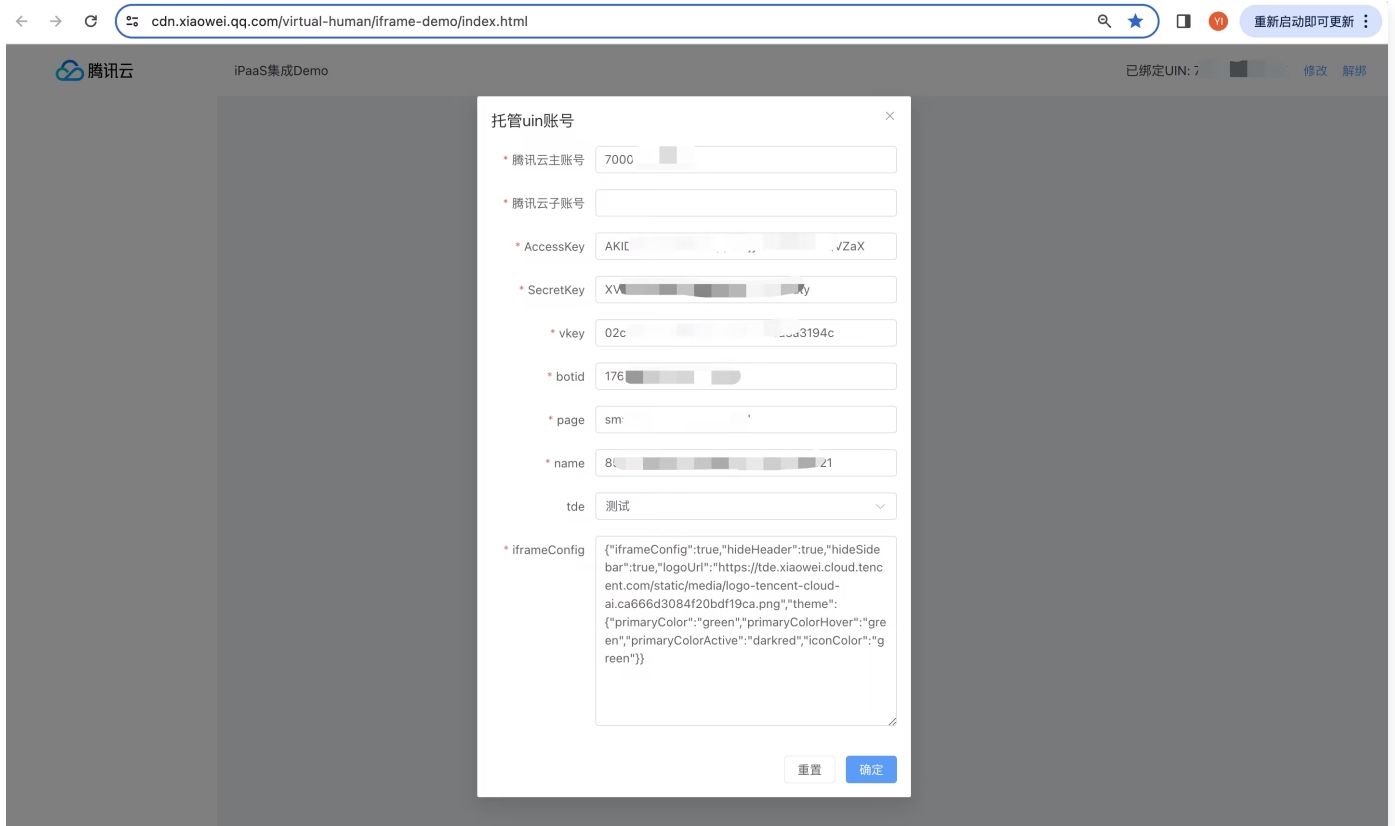
		 <p>The diagram shows a yellow rectangular area representing an iframe. On the left side, there is a dark grey vertical bar labeled 'sidebar'. The main area of the yellow rectangle is labeled 'content'. Above the yellow area, there is a blue horizontal bar labeled 'navbar'. A red arrow points from the text '整个黄色内容区域都属于iframe' to the yellow area.</p>
4	iframe 自适应	<p>可支持传入嵌入范围的宽和高，iframe 能自适应展示，且尽量保证：</p> <ul style="list-style-type: none"> • 弹窗居中。 • 嵌入部分不出现滚动条。 • 浮层弹窗等能在最上层弹出。
5	跨域接入	<p>可支持客户跨域接入，无论客户平台使用什么网关和账号体系：</p> <ul style="list-style-type: none"> • 客户账号租户：支持客户通过映射表或类似方式，将自己的账号体系转换为腾讯云 UIN，能以不同租户的身份来访问平台嵌入页面。 • 主子账号：支持客户维护腾讯云父子账号租户关系，若客户指定了主子账号关系，能实现在嵌入页面内操作主子账号的资产分配功能。
6	每客户唯一 license + 短链	<p>即 A 客户的 license 仅限 A 且仅限A唯一对接嵌入，如果 A 将接入链接给到 B，B 客户无法接入或者接入后 A 无法接入。</p>
7	页面前进后退 + 跳转	<ul style="list-style-type: none"> • 支持通过浏览器内的 go back 返回前序页面。 • 支持通过客户传入的 postMessage 跳转到指定页面。
8	主题色级别的换肤	<p>可支持一个主题对应一个配置文件，可通过客户写入配置文件的方式来实现主题风格的变化。</p>
9	LOGO 等图片资源的配置	<p>可支持一个主题对应一个配置文件，可通过客户写入配置文件的方式来实现 LOGO 图片的变化。</p>
10	嵌入页面的配置	<p>可指定嵌入页面范围。</p>

体验中心

为集成方提供了集成 Demo 参考，可直观理解集成基础版后的功能及体验，以及更改主题色、LOGO 配置后的展示效果。

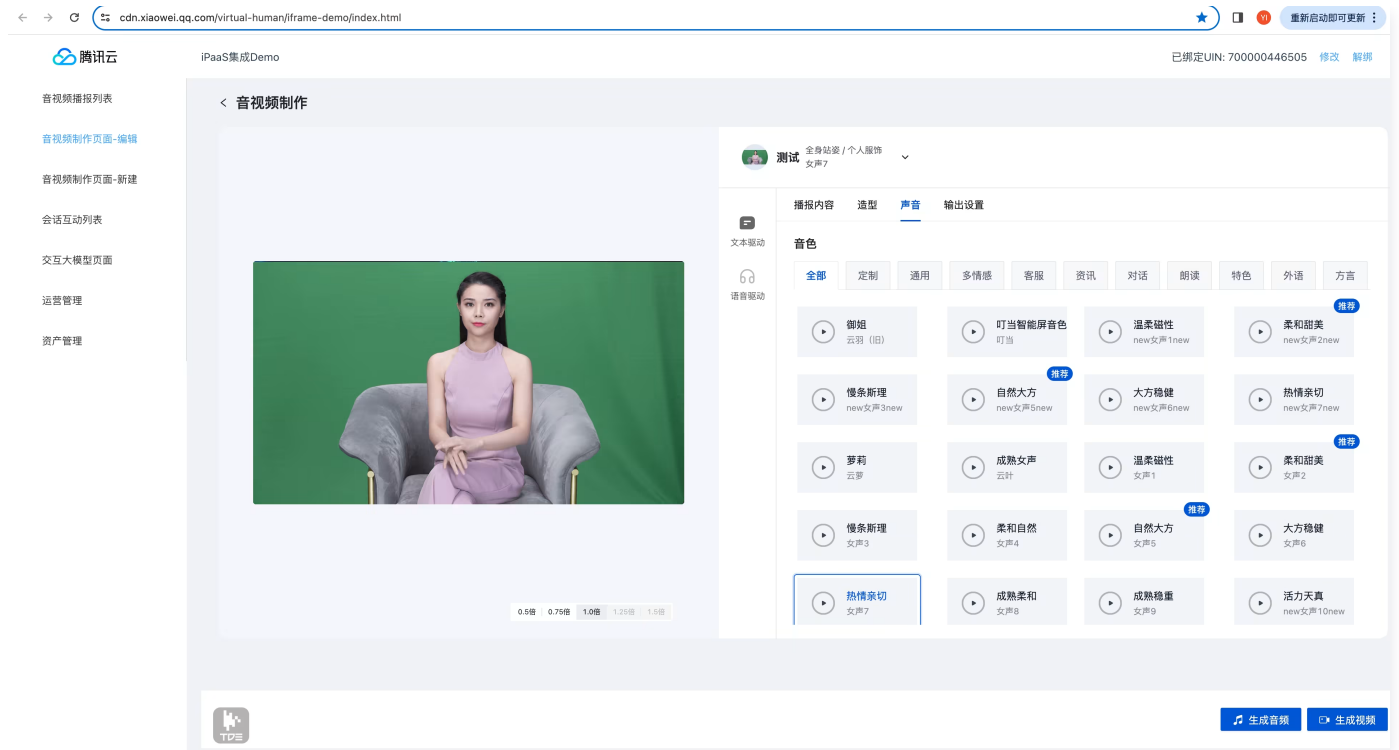
! Demo 示例:

<https://cdn.xiaowei.qq.com/virtual-human/iframe-demo/index.html>



参数说明

名称	说明
vkey	数智人项目的 vkey。
botid	客户托管账号绑定的腾讯云智能体开发平台机器人 ID。
page	播报页面类型。
name	播报数智人 name 参数。



高级版介绍

产品特性

高级版在基础版的特性之外，还搭配了运营后台，集成商可通过运营后台自主控制：

- iframe 的换肤和站点对接
- 货架运营
- SKU 管理
- 用户权益管理

其中用户端功能清单如下：

iPaaS 高级版用户界面功能清单			
一级页面	二级页面	功能	描述简述
首页	/	首页	呈现清晰的用户场景指引和功能菜单指引。
作品	/	草稿管理	支持查看历史编辑的稿件。
		视频管理	管理已生成的视频。
模板	/	模板管理	支持按用户场景筛选和按名称搜索数字人视频模板。
		通用模板	内置常用应用场景的数字人视频模板50个，区分电商、营销、商务等常用行业场景。

		租户自营模板	支持根据租户的市场需求，自行建设、上架数字人视频模板库。
创建视频	数字人视频多轨编辑器	剪辑编辑页面	撤销，重做，清空，画面比例切换，剪辑。
		模板	横版，竖版，搜索。
		PPT 模式	支持按幻灯片模式剪辑视频，支持新增页面。
		数字人	支持选择数字人形象，展示全部公开形象、收藏形象、定制形象。
			播报内容创作，多种语种 TTS 生成。
		背景	支持图片背景，纯色背景，自定义上传。
		资源库	支持我的资源中的视频、音频、图片选择。
		贴纸	支持在视频中添加各种贴图和动画。
		音乐	支持在视频中添加背景音乐。
		花字	支持在视频中设置和添加花字。
字幕	支持开关播报内容字幕，并设置字体、样式。		
数字人	共享数字人	数字人列表	提供多种数字人形象可供选择，预置135个，形象名称、标签等不可变更。
		租户自营数字人	支持根据租户的市场需求自行建设、上架数字人，形象名称、标签等可以租户自行维护。
	我的数字人	数字人列表	查看自己定制的数字人形象。
		创建数字人	用户上传视频，定制自己的视频数字人，包含通用口型、专属口型2种类型。
声音	共享音色	声音列表	提供多种 TTS 音色可供选择，预置88个，普通话82个，中文方言东北话、粤语、四川话3个，外语包含英语、日语3个。
		租户自营声音	根据租户的市场需求自行建设、上架音色，音色名称、标签等可以租户自行维护。
	我的音色	我的声音列表	查看自己定制的音色。
		声音复刻	用户上传音频，复刻自己的音色用于多语种 TTS，包含基

			基础版、极速版2种类型。
资源库	/	个人资源管理	支持上传视频、音频、图片素材，可分类查看。
AI 工具	智能抹除	智能抹除	添加视频，选择区域，自动去除水印或字幕。
	智能转比例	智能转比例	上传视频实现智能横竖屏转换，保持人物始终在视频中心。
我的	会员权益	我的权益	我的会员资产用量及总量，增值权益用量及总量。
		权益清单	用户可查看各会员套餐包含的权益内容。需要租户在租户运营后台预先进行 SKU 配置。
		购买会员	用户可通过租户自建的购买页购买会员及增值服务。需要租户接入 iPaaS 提供的下单发货、查库存、退费 API。
	账号及空间	个人账号	查看当前登录的用户账号、头像信息。
		团队空间设置	查看已经创建好的团队空间，支持邀请成员加入。仅限企业客户且购买会员权益里包含团队空间权益项的用户。
		切换空间	支持用户在个人空间、团队空间中切换。仅限企业客户或购买会员权益里包含团队空间权益项的个人用户。
	帮助中心	新手指引	新手指引视频，帮助用户创建第一支数字人视频。由租户负责制作，并在租户运营后台进行配置视频地址。
		帮助中心	查看帮助文档。由租户负责制作，并在租户运营后台进行配置跳转链接。

其中运营后台功能清单如下：

iPaaS 高级版运营后台功能清单			
模块	功能项		功能简述
运营权限管理	登录访问	登录页	有独立域名，租户可自助访问。
		账号鉴权	用户名密码访问，滑块验证码验证。
	权限管理	用户管理	新增/删除运营后台用户，按用户查看和编辑其角色、权限有效期。
	租户空间	/	为集成商分配一个独立的租户空间，用于存储租户及其用户的成片数据、用户数据等资产，默认上限20TB。支持租户增购。

页面个性化	换肤	LOGO 替换	涉及位置：登录页、首页 titlebar、视频水印、页签、分享卡片。
		主题色更换	涉及位置：按钮、banner & 登录页 & 权益页氛围背景、主要 icon SVG。
	运营位	外链配置	权益页购买按钮 URL、咨询销售 URL、用户手册 URL。
		教学视频	指定视频文件。
	嵌入对接	Token 回调检查	用于数字人后台校验集成商登录态的合法性以及获取用户资料。
运营管理	货架运营	数字人货架	独立控制 iPaaS 空间的形象上下架。
		音色货架	独立控制 iPaaS 空间的音色上下架。
		剪辑模板	包含模板分类、官方模板管理、模板制作 & 投稿权限管理。
		背景素材管理	包含背景图片管理。
		音频管理	包含背景音乐的管理。
	用户运营	用户列表	查询个人用户的 ID、昵称、注册时间等用户基础信息。
		企业列表	录入企业，并查询和编辑企业的名称、成员、头像等企业基础信息。
商业化管理	商品管理	SKU 管理	定义会员套餐包权益，定义增值购买项权益。
		促销活动	定义 SKU 划线价及生效期。
	商务管理	用户订单	录入个人和企业客户订单，用于导出对账。
		库存水位	查看集成商的资产消耗情况。
		用户会员	查询个人和企业客户的会员套餐权益发放和生效情况。
		用户权益	查询个人和企业客户的增值权益发放和生效情况。
		核销记录	查询个人和企业客户在用户端的权益核销记录。

体验中心

为集成方提供了集成高级版后的体验中心供参考，可直观理解集成高级版后的功能及体验，并对照高级版功能清单直观查看可由租户配置的内容。

 Demo 示例：

<https://xiaowei.cloud.tencent.com/ivh/#/ipaas>



当前仅限邀请体验，如需体验请通过您的客户经理申请，或通过邮件提供您的 UIN 给数字人业务组开通体验入口：

注意：

体验授权有效时间为申请完成之日起60天。

邮件格式参考如下：

体验中心访问权限申请流程（邮件）

- 邮箱地址：txc_avatar@tencent.com
- 邮件主题：iPaaS 高级版申请（体验）
- 邮件内容：
 - 腾讯云主账号（UIN）：
 - 申请人/客户名称：

基础版接入指引

最近更新时间：2026-01-21 17:30:24

一、集成方工作清单

1. 云账号托管能力建设：

○ 第一步：准备工作

- 创建腾讯云账号（即 UIN 以及子账号），客户自行在腾讯云注册创建即可。
- 获取“客户标识”值（即 merchantID 值，后面提到的 from 参数也用该值代替），由数智人管理后台配置后提供，可邮件申请，也可联系商务同学获取。

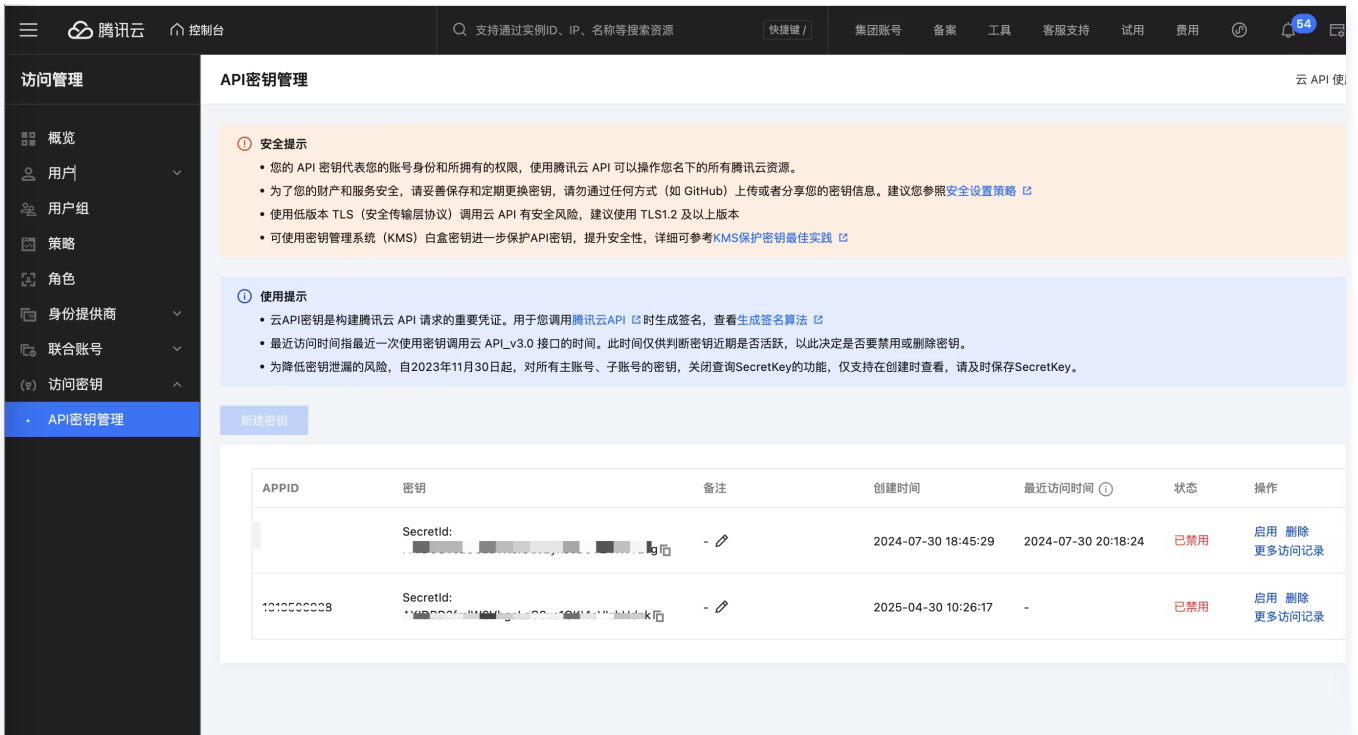
○ 第二步：用注册的腾讯云账号生成 aksk（特指 secretId 和 secretKey），aksk 有临时和永久两种，二选一即可，建议用临时 aksk。

○ 临时 aksk 获取流程

- 按照 [获取联合身份临时访问凭证](#) 说明构建请求（临时 aksk 只能通过 API 获取），其中 policy 参数指定为：`%7B%22version%22%3A%222.0%22%2C%22statement%22%3A%5B%7B%22effect%22%3A%22allow%22%2C%22action%22%3A%5B%22ivh%3A*%22%5D%2C%22resource%22%3A%5B%22*%22%5D%7D%5D%7D`
- 动态存储返回的 Token 和 aksk 信息，做好和腾讯云账号的关联映射关系，方便“Token 校验回调接口”使用。Token 和 aksk 过期后要及时更新。

○ 永久 aksk 获取流程

- 主账号 aksk：可通过“腾讯云控制台 > 访问管理 > 访问密钥 > API 密钥管理 > 新建密钥”获取。
- 子账号 aksk：可通过“腾讯云控制台 > 访问管理 > 用户 > 用户列表 > 选择用户 > 单击用户名 > API 密钥 > 新建密钥”获取，若必须用永久 aksk，优先建议使用子账号 aksk。提示：永久 aksk 模式下，Token 由开发者自行实现。



主账号 aksk 获取路径截图



子账号 aksk 获取路径截图

- 第三步：集成方按照接口规范提供一个公网 API，作为数智人校验 Token（即上一步中提到的 Token）的回调接口。接口规范请参见 [Token 校验回调接口规范](#)，可邮件发给我们，也可通过商务同学传递。提示：merchantID（即客户标识）在未获取的情况下，可以先自定义一个值，等数智人方提供后，替换即可。

说明：

第一步和第三方操作，可发邮件至：`txc_avatar@tencent.com`，邮件内容如下：

- 邮件主题：iPaaS 接入申请
- UIN: xxxxxx
- Token 校验回调接口：`https://xxxxx`

我们会在邮件回复中提供“客户标识”值。

- 第四步：参照 [产品介绍](#) 中的 Demo 来构建数据的“信息录入交互页面”。数据信息包括：UIN、vkey、botid、page、name 等等。

⚠ 注意：

Demo 中的 AccessKey 和 SecretKey 为永久密钥，可根据实际情况调整为不需要。

2. 数智人平台菜单框架开发及集成：

开发者通过 iframe 集成，使用下面各个菜单页的 URL 在客户业务系统中嵌入数智人平台页面。

- 菜单各个页面的拼接规则为：**主域名 + 路由地址**
- 主域名为：`https://xiaowei.cloud.tencent.com`，**路由地址**详见下文中的表格清单。
- 示例：
 - 音视频播报列表页面，其路由地址为：`/ivh/#/application?iframe=true&from=xxx&token=xxx`
 - 则该页面的访问地址 URL：`https://xiaowei.cloud.tencent.com/ivh/#/application?iframe=true&from=xxx&token=xxx`
- 统一参数：
- from 指客户标识，由数智人侧分配，请参见 [第一步：准备工作](#)。
- token 由集成商生成，背后对应客户的 uin、aksk，当访问页面时数智人侧会回调接口通过 token 换取 uin 和 aksk，详见上文。

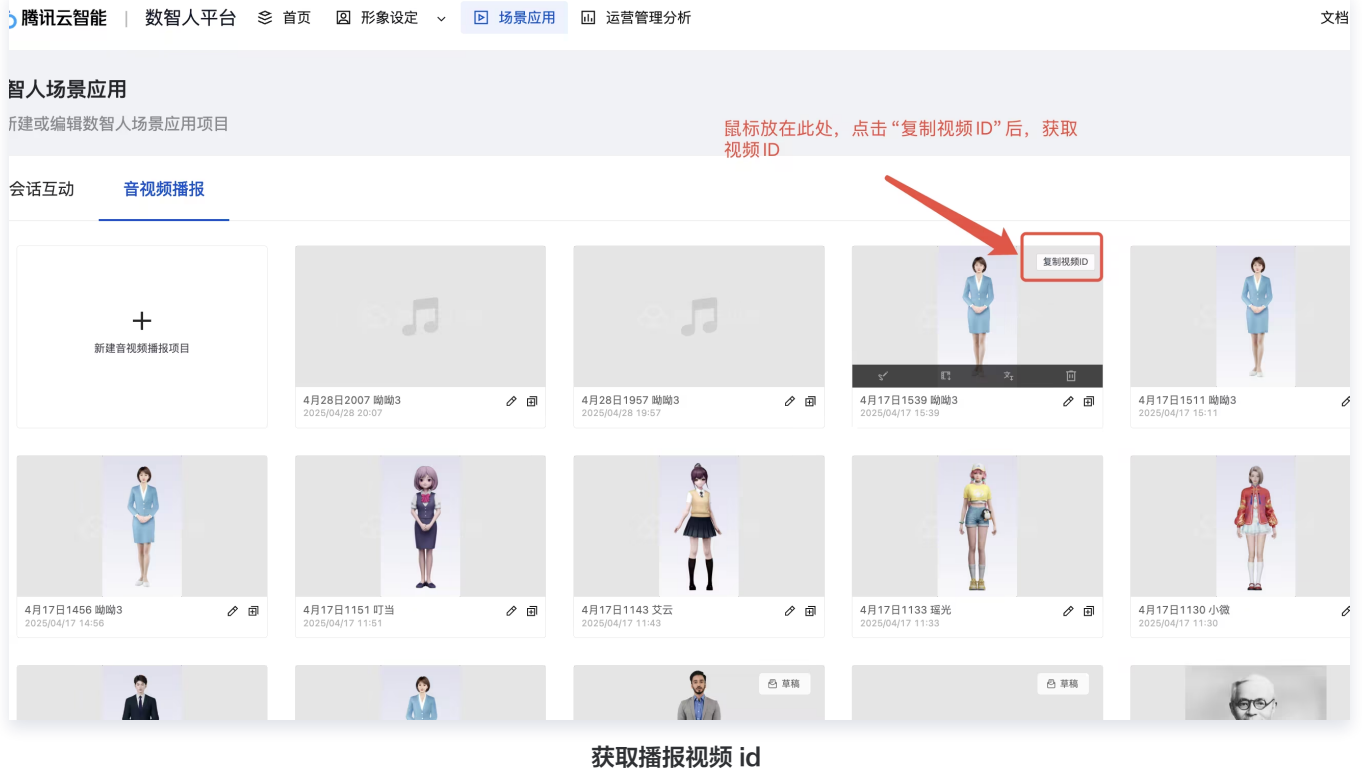
目前数智人平台支持以下菜单页面的嵌入（持续更新中），完整平台功能范围请参见 [数智人平台操作指引](#)。

菜单页面名称	菜单页面路由地址	备注
音视频播报列表	<code>/ivh/#/application?iframe=true&from=xxx&token=xxx</code>	-
音视频制作页面-编辑	<code>/ivh/#/bvh_page?page=xxx&name=xxx&iframe=true&from=xxx&token=xxx</code>	<ul style="list-style-type: none"> • page 为制作视频所选用的数智人类型，详见下文的数智人类型。 • name 为播报视频 ID，详见下文图示

音视频制作页面-新建	<code>/ivh/#/bvh_page?page=xxx&iframe=true&from=xxx&token=xxx</code>	page 为制作视频所选用的数智人类型，详见下文的数智人类型。
会话互动列表	<code>/ivh/#/application?iframe=true&from=xxx&token=xxx</code>	-
运营管理	<code>/ivh/#/analysis?iframe=true&from=xxx&token=xxx</code>	-
资产管理	<code>/ivh/#/assets_management?iframe=true&from=xxx&token=xxx</code>	-

数智人类型代号

数智人类型	
real_man_2d	2D 精品
small_sample_2d	2D 小样本 (专属)
half_realistic_3d	3D 半写实
cartoon_3d	3D 卡通
real_man_3d	3D 写实
small_sample_general_2d	2D 小样本 (通用)
small_sample_photo_2d	2D 小样本 (照片)
noTrain	免训练视频
noTrainPhoto	免训练照片



3. Token 回调检查接口:

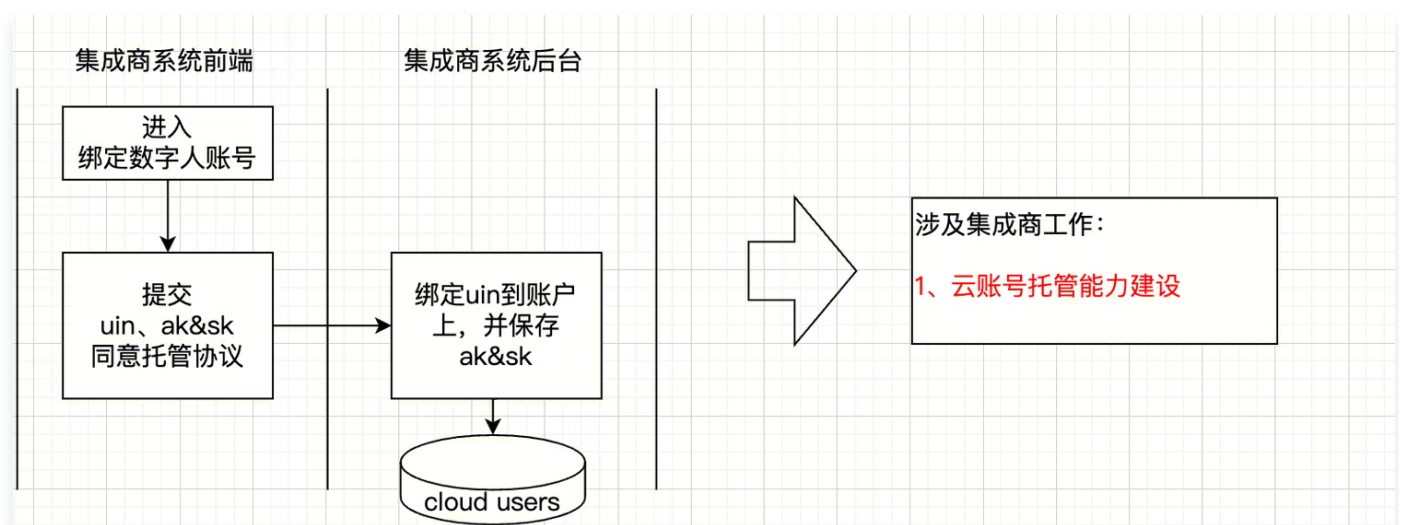
数字人进行 Token 回调，返回客户托管的 aksk 信息。

当数智人页面打开时，会拿到 URL 上的带参 Token，回调客户系统检查 Token 合法有效性，并希望客户返回该 Token 对应的客户已托管的 uin、ak、sk 信息，以便后续能访问内嵌的数智人平台页面 xxxx。

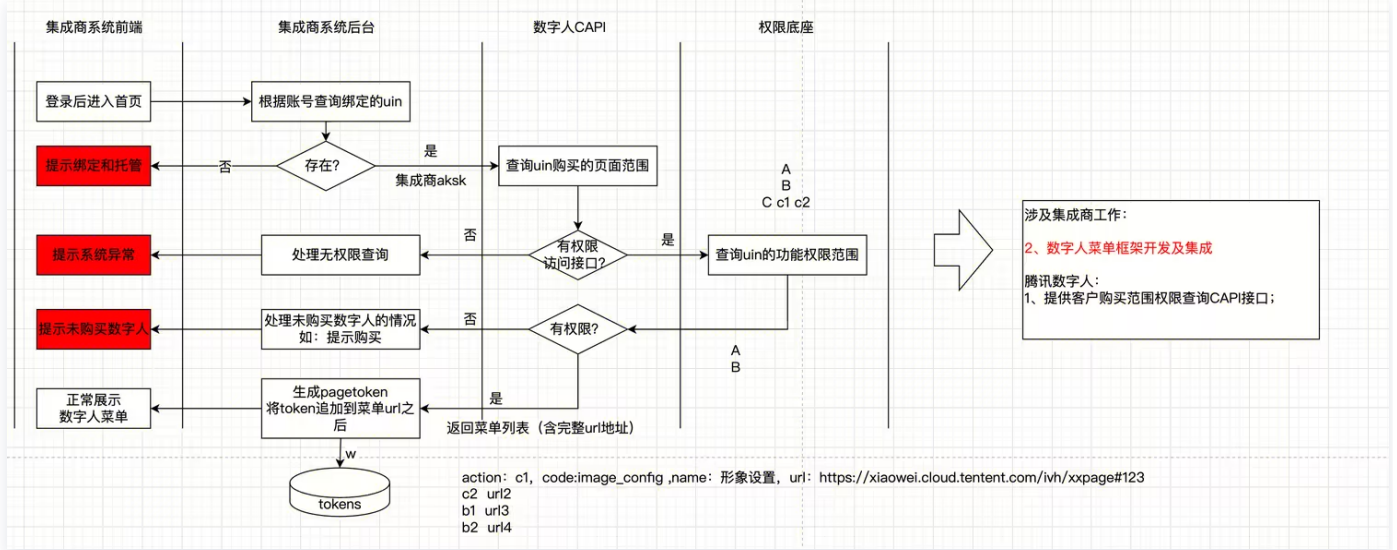
若上文第一部分的第四步已经完成，此处可忽略。

二、集成过程说明

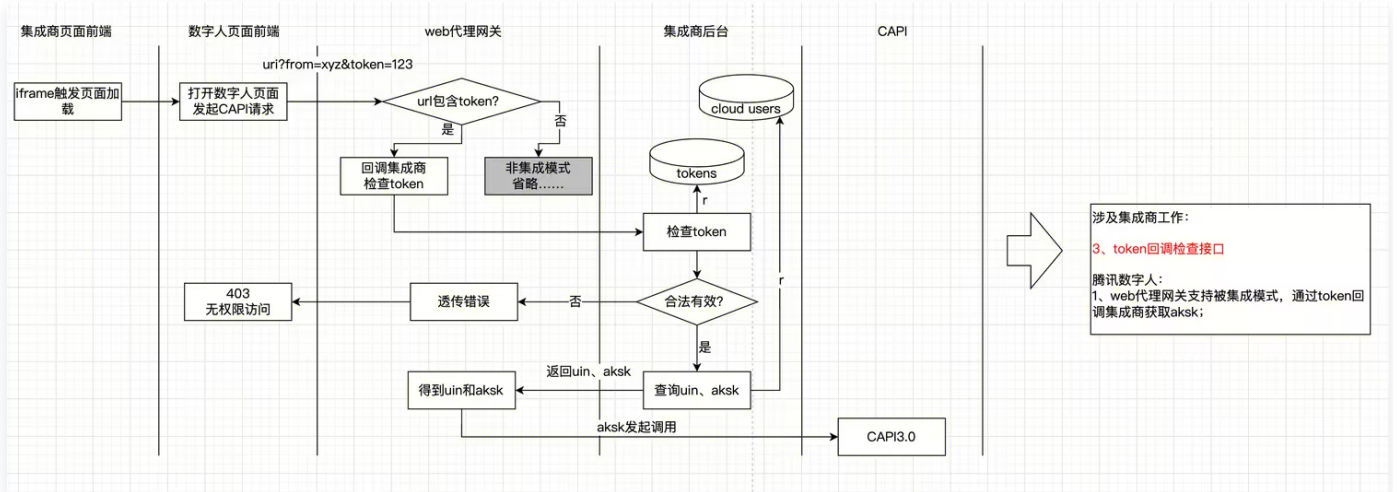
1. 客户 uin、aksk 托管。



2. 集成商系统主框架加载流程图简化，只区分腾讯 vs 客户的鉴权和查询。



3. 集成商系统内打开数字人页面。



三、Token 校验回调接口规范

- 协议: https
- 请求 method: POST
- 请求 content-type: application/json
- 请求 body:

```

{
  "token": "token", // 集成商生成，在页面嵌入时在url中追加的token参数。
}
    
```

- 返回结构要求

```
{
  "request_id": "request_id", // 唯一请求id
  "code": 0,
  "message": "ok",
  "merchantID": "merchantID", // 集成商id, 腾讯分配, 在页面嵌入时在url
  中追加的from参数。
  "merchantUin": "merchantUin", // 托管的客户uin
  "MerchantSubAccountUin": "MerchantSubAccountUin", // 托管的客户子账
  号uin
  "sessionToken": {
    "secretId": "secretId", // 密钥id, 建议使用临时密钥返回
    "secretKey": "secretKey", // 密钥key, 建议使用临时密钥返回
  }
}
```

接口开发完成后, 提交给厂商的产品运营人员, 通过上文中的第四步所开发的“信息录入交互页面”进行相关配置。

四、页面样式规范

可配置样式范围

针对 iframe 嵌入的页面, 目前已支持一键更换主题色、logo 配置的功能。

配置方式

具体能力详见下面的配置内容:

```
{
  "iframeConfig": true, // 表示当前是iframe潜入, 并且支持换肤能力
  "hideHeader": true, // 隐藏数智人平台的头部内容
  "hideSidebar": true, // 隐藏数智人平台的左侧菜单内容
  "logoUrl": "/logo.png", // 更换数智人头部的logo信息
  // 页面相关的tab可以通过权限控制隐藏显示。
  // 主题色配置
  "theme": {
    "primaryColor": "green",
    "primaryColorHover": "green",
    "primaryColorActive": "darkred",
    "iconColor": "green"
    // ....其他antd支持的主题样式配置。
  }
}
```

```
}
```

高级版接入指引

最近更新时间：2025-07-10 14:17:11

一、角色说明

整个业务模式涉及四种角色，关系如下：

业务人员	系统角色	角色权限概述
腾讯数字人运营	管理员	集成商录入 + 集成商权益管理。
集成商	租户	将自有系统集成数字人的页面，打通用户登录态，拥有独立的租户空间 + 独立的运营后台，可自主控制： <ul style="list-style-type: none">• iframe 的换肤和站点对接• 货架运营• SKU 管理• 用户权益管理
集成商的 B 端客户	企业	由集成商录入企业 + 管理企业级权益，并添加企业成员。
集成商的 C&B 终端用户	用户	用户自主在集成商侧注册，集成商登录数字人运营后台，管理用户级权益。

二、接入前准备

2.1 服务购买开通

- **商务流程**（集成商向腾讯商务发起）：联系腾讯商务签署合约或通过腾讯云官网下单 iPaaS 服务。
- **服务开通**（腾讯交付人员操作）：腾讯交付人员通过管理员身份，在腾讯数字人 SaaS 后台管理系统 > 租户管理模块，录入集成商的 iPaaS 服务资产及服务有效期，开通日即视为服务开始日。服务开通后，会获得后台管理系统的租户工作台管理账号。

2.2 初始化设置与集成对接

了解页面个性化设置，准备相关素材和文案。

被集成页面支持个性化设置顶部 LOGO、视频成片水印文件、按钮主题色、外链配置、品宣标签、教学视频等。

页面个性化
✕

▼ 图片配置
保存

* 顶部logo 👁

* 视频水印

📁 上传文件

P9kGD...

* fav和分享logo 👁

▼ 主题配置
保存

* 渐变色范围

#3a92e3de ~#0844f5ff

▼ 外链配置
保存

* 购买页

请输入

0/200

* 联系商务

请输入

0/200

* 用户手册

请输入

▼ 品宣文案
保存

* 标签

请输入

▼ 教学视频
保存

* 场景

advertisement

* 讲解视频

* 标题

请输入

0/100

* 副标题

请输入

0/200

📁 上传文件

▶ 00:00 / 00:33 倍速 🔊 📺

对接服务，集成页面

提前阅读下方架构和接口文档，做好集成环境准备。

2.3 用户会员套餐 SKU 设计

在接入服务前，集成商可先参考以下权益项，自主设计租户层面的会员套餐 SKU 定价及包含的权益项范围。

📌 说明：

- 全部权益项可配置在权益清单页不可见，即对用户不主动感知。

- 全部用量权益（即 count 值型）将在用户核销时，扣减租户的 iPaaS 资源包。

用户会员套餐 SKU 权益控制元数据项

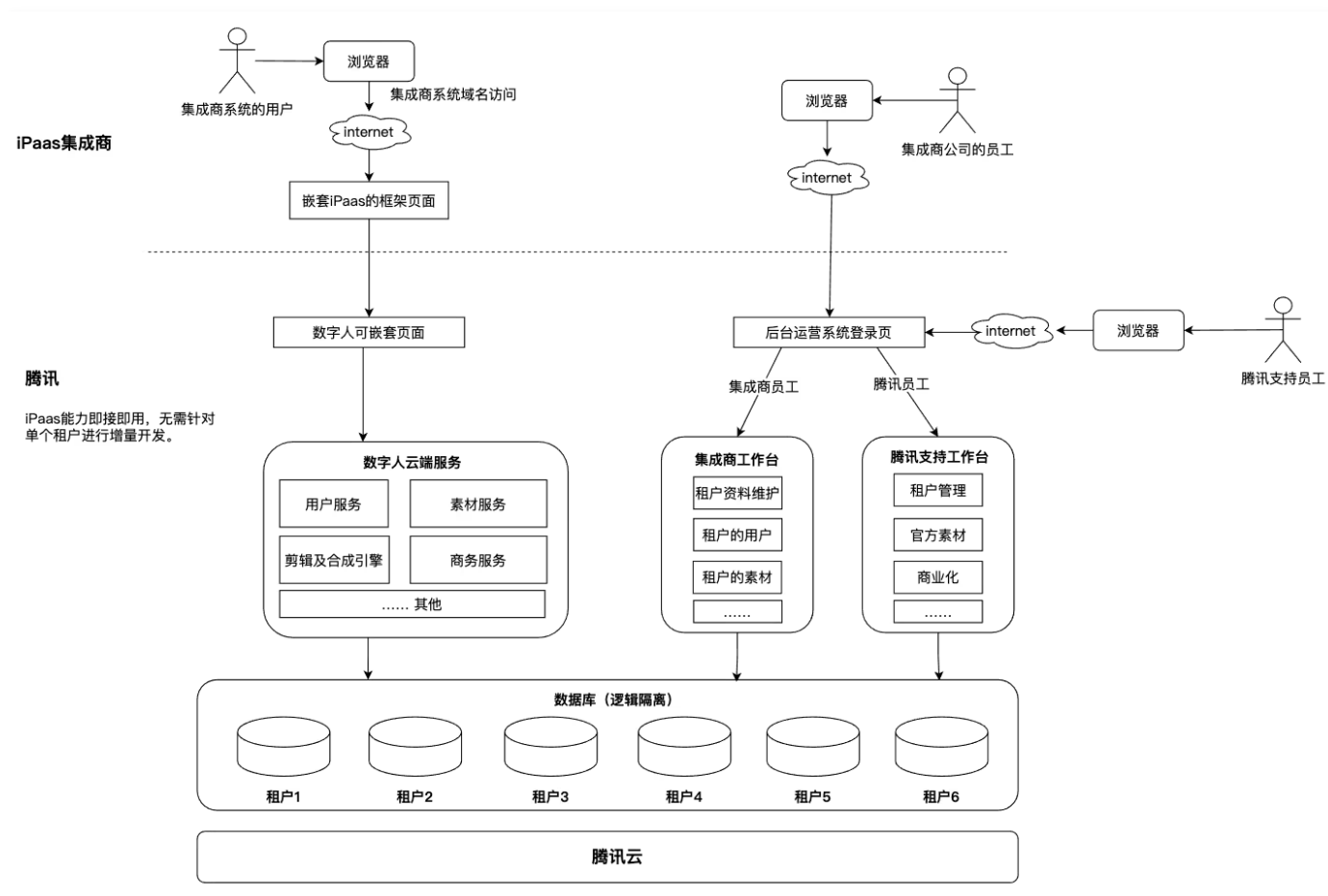
权益分类	权益项	权益说明	权益项类型	用户校验配置值	权益清单展示值（示意）
账号数	账号数量上限	控制可享受该档位会员权益的账号数量	用量权益	x个	1个/套
	允许账号增购	是否允许该档位会员增购账号	功能权限	true false	√ ×
数字人视频	视频清晰度	控制用户导出视频时的可选清晰度下拉选项	功能权限	720p 1080p 2k 4k	最高4K
	成片无水印	控制用户导出视频时的水印开关选项	功能权限	true false	√ ×
	视频合成速度	当前仅用于权益页展示，后续预留关联控制并发数	展示值	按需配置	标准 较快 最快
	单条视频最大时长	控制用户导出视频的最大单条视频时长	用量权益	x分钟/条，-1代表无限制	5分钟
	合成视频合成使用量	控制用户导出视频的当月累计视频时长	用量权益	x分钟/月，-1代表无限制	30分钟/月
	可用公版数字人数量	仅用于权益页提示用户该档位会员可使用的公版数字人数量，实际需通过数字人货架配置付费类型控制	展示值	按需配置	50+
	可用视频模板数量	仅用于权益页提示用户该档位会员可使用的视频模板数量，实际需通过官方模板配置付费类型控制	展示值	按需配置	60个公共模板 + 专用模板
TTS	可用公版音色	仅用于权益页提示用户该档位会员可使用的公版音色数量，实际需通过声音货架配置付费类型控制	展示值	按需配置	50+

	每日试听上限	控制用户制作音频中通过试听合成的 TTS 时长上限	用量权益	x分钟/日，-1代表无限制	20分钟/日
	每日合成 TTS 总次数	控制用户制作音频中正式合成的 TTS 时长上限	用量权益	x分钟/月，-1代表无限制	50分钟/月
	字幕下载	是否允许该档位会员点击下载字幕文件	功能权限	true false	√ ×
数字人定制	专属口型数字人定制额度	控制用户定制专属口型数字人时的可用配额数量	用量权益	x个，-1代表无限制	1个/套
	通用口型数字人定制额度	控制用户定制通用口型数字人时的可用配额数量	用量权益	x个，-1代表无限制	1个/套
	照片数字人定制额度	控制用户定制照片数字人时的可用配额数量	用量权益	x个，-1代表无限制	1个/套
	可用高精数字人	是否允许该档位会员增购及使用高精数字人（4K数字人）	功能权限	true false	√ ×
声音复刻	极速版声音复刻额度	控制用户定制极速版声音复刻时的可用配额数量	功能权限	true false	√ ×
	基础版声音复刻额度	控制用户定制基础版声音复刻时的可用配额数量	功能权限	true false	√ ×
AI 小工具	智能抹除	控制用户导出智能抹除视频的当月累计时长上限	用量权益	x分钟/月，-1代表无限制	300+分钟/月
	智能转比例	控制用户导出智能转比例视频的当月累计时长上限	用量权益	x分钟/月，-1代表无限制	300+分钟/月
团队协作	团队成员管理	是否允许该档位会员创建和管理团队空间	功能权限	true false	√ ×
	团队成员共享	是否允许该档位会员在团队空间内共享资	功能权限	true false	√ ×

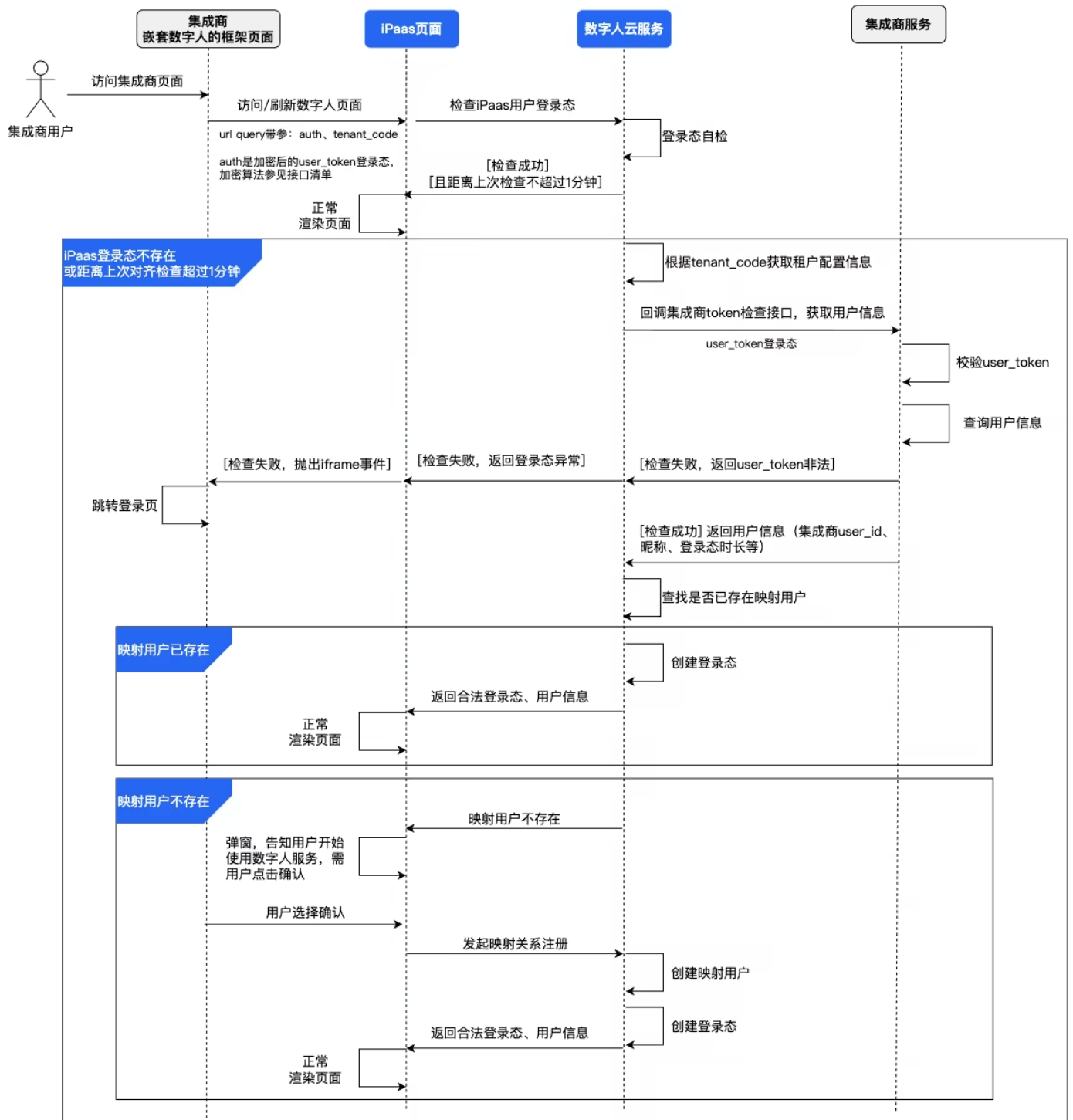
		源和成片			
	协作编辑与评论	是否允许该档位会员在团队空间内协作编辑和评论视频	功能权限	true false	√ ×
	可商用	当前仅用于权益页展示	功能权限	true false	√ ×
其他	客户服务	当前仅用于权益页展示	展示值	按需配置	客户服务：电话、邮件支持 对接帮助：专用的企微客服群频道

三、技术架构

3.1 iPaaS 模式整体架构



3.2 登录及用户映射



四、接口清单

4.1 页面集成

接口说明

表述数字人 iPaaS 可被集成的页面，以及页面集成相关协议参数及接口约定。

可集成页面

iPaaS 首页。

页面集成方法

1. 做一个 iframe 页面，引入 iPaaS 的 js、添加一个 iframe 元素，并使用 init 函数初始化 iPaaS 设置。

```
<html>
<head>
  <title>集成商iframe页面示例</title>
  <script src="https://static-avataarch.ivh.qq.com/avataripaassdk.js"/>
  <script>
    AVATARIPAAS.init({
      tenant_code: "apple", //租户唯一代码
      auth: "nldaso0n1nc01n13MABc==", //用户登录态加密串，拼接到iframe
      的query时会进行urlencode
      iframeid: "main", // 框架页iframe元素的id
      access_path: "https://www.mywebsite.com/hello", // iframe页
      面的url路径。
      on_goback: function() { // 当用户首次使用时，在弹窗页面选择了拒绝
      使用时回调
        // do something, 比如使用浏览器 go history -1
      },
      on_authexpired: function() { // 当腾讯检测到用户登录态失效时回调
        // do something, 比如跳转到登录页
      },
    });
  </script>
</head>
<body>
  <!-- 当引入avataripaassdk.js并init之后，会自动设置iframe的src嵌套ipaas页
  面 -->
  <iframe id="main"/>
</body>
</html>
```

2. 在 nginx 中，对 iframe 页面进行代理设置。

- 将 iframe 访问 URL 统一代理给自身（实现 iframe 内切换页面后，再刷新 iframe 页时实现当前页刷新）。
- 假设您的网站"/hello"是 iframe 页面的路径，进行如下设置。

```
server {
  listen 80;
```

```
server_name https://www.mywebsite.com; // 代理的网站域名
location /hello/ { // 将hello下的子路径全部代理回hello，此处末尾一定要加/，否则会导致代理死循环
    rewrite ^(.*)$ /hello break; // 此处路径w末尾不加/
    proxy_pass https://www.mywebsite.com;
}
}
```

附加说明：

参数	类型	必填	说明
auth	String	是	用户登录态加密串，加密算法见 auth 参数加密方法 。
tenant_code	String	是	租户唯一代码，开通租户后在管理后台获取。
iframeid	String	是	框架页 iframe 元素的 ID。
access_path	String	是	iframe 页面的 URL 路径，在 iframe 内部切换 iPaaS 页面时，SDK 会自动在此 URL 后面追加子路径，并同步到浏览器 Location，实现 iframe 页面的刷新。
on_goback	function	是	当用户首次使用时，在弹窗页面选择了拒绝使用时回调。
on_authexpired	function	是	当腾讯检测到用户登录态失效时回调。

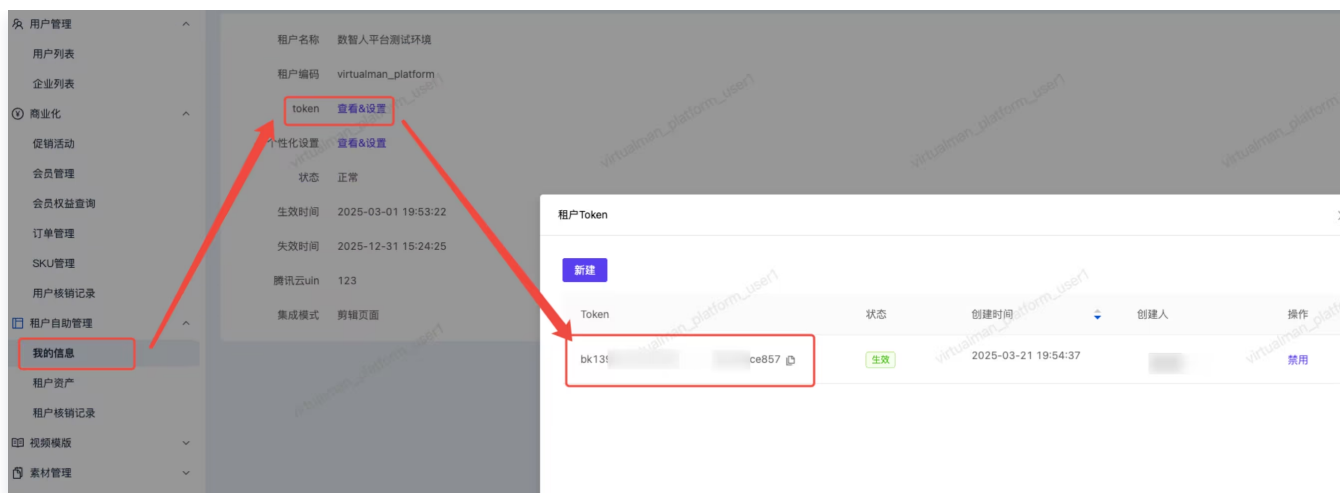
auth 参数加密方法

1. 构建明文字符串

`user_token=${user_token}&ts=${timestamp}`，其中 `user_token` 是用户登录态，能够唯一标识一个用户的登录态，可直接传入集成商登录态或者再临时生成一个和登录态一一对应的临时 Token 都可以，由集成商来决定，因是 URL query 带参，不建议直接将 jwt token 写入，会导致 query 过长；`timestamp` 是生成的当前系统时间戳（Unix 时间戳精确到秒），腾讯侧会对时间戳做校验，有效性一分钟。

2. 将明文字符串进行加密

- 使用 AES 加密算法，采用 GCM 模式，将租户级 Token 作为密钥对明文字符串进行加密，加密后转 Base64 字符串。
- 租户级 Token 获取方法如下：



3. 将步骤2得到的结果字符串作为 auth 的值。
4. 可使用后台管理控制台的调试工具，进行加解密测试。



4.2 登录态检查

接口说明

该接口用于打通数字人 iPaaS 和集成商系统的登录态以及完成两侧用户资料的映射。

⚠ 注意：

集成商需按照本接口协议实现此接口，并在管理系统将此接口地址注册到系统中。

登录集成

保存

SSO token检查回调地址

您选用了页面级集成，那么登录及认证系统将由您提供，需要按要求对接SSO登录相关流程，请在下方填入腾讯云回调您方系统的 token 检查接口来确认用户登录态的合法性。

token回调检查地址

调用协议

- 协议: HTTPS
- 方法: POST
- Header Content-Type: application/json;charset=utf-8

请求 Header

参数	类型	必填	说明
auth	String	是	腾讯侧的身份加密串，集成商可解密 auth 参数，以确认请求来源的合法性。 <ul style="list-style-type: none"> • 加密明文: tenant_code=\$tenant_code&ts=\$timestamp 其中 tenant_code 为授权给集成商的唯一标识。 <ul style="list-style-type: none"> • 加密方式: 使用租户 Token 对原文做 AES 加密，采用 GCM 模式，并对密文做 Base64 编码。

请求参数

参数	类型	必填	说明
user_token	String	是	腾讯云服务内解析出的用户登录态（明文）。
request_id	String	是	每一次请求的唯一标识，可用 UUID 算法生成。

```
{
  "user_token": "jnnuqnegwbunqoncv121nnqonca1561a",
  "request_id": "hbkoqnzlobnqbdso1bn231ncdo1oqa",
}
```

返回参数

参数	类型	必填	说明
code	int	是	结果码 <ul style="list-style-type: none"> • -1: 发生系统异常。 • 0: Token 有效。 • 900001: Token 不合法。 • 900002: Token 已过期或不存在。
message	String	是	有异常时填写异常信息。

request_id	String	是	填入请求包体中的 request_id。
data	JSON Object	否	返回包体中的数据，其中包含 user_info 用户信息。
data.user_info	JSON Object	否	当 Token 有效时，必须返回。
data.user_info.token_expire_at	int	否	用户登录态过期时间（单位秒）。
data.user_info.user_id	String	是	集成商侧用户 ID。
data.user_info.username	String	是	集成商侧用户名，如果此字段不唯一，需要至少返回手机号或者邮箱其一，否则在后台管理系统某些功能页面中下拉选择用户时将无法显性精准选中某一用户。
data.user_info.mobile	String	否	集成商侧用户手机号，可用*隐去中间四位。 <ul style="list-style-type: none"> • 例如，139****4452。 • 脱敏后，登录数字人 iPaaS 后台管理搜索用户时，也需使用脱敏后的值进行搜索。
data.user_info.email	String	否	集成商侧用户邮箱，可用*隐去中间字符只保留前4个字符和@之后的字符。 <ul style="list-style-type: none"> • 例如，liut****@163.com • 脱敏后，登录数字人 iPaaS 后台管理搜索用户时，也需使用脱敏后的值进行搜索。

```

{
  "code": 0,
  "message": "paas",
  "request_id": "hbkoqnzlobnqbdsao1bn231ncdo1oqa",
  "data": {
    "user_info": {
      "token_expire_at": 1747368027,
      "user_id": "5612910",
      "nickname": "魔礼青",
      "mobile": "139****4452",
      "email": "liut****@163.com",
    }
  }
}
    
```


API 接口主域名及鉴权

数智人 aPaaS 接口调用方式

最近更新时间：2026-03-13 12:07:42

一、接口调用环境

正式环境域名：

`https://gw.tv.s.qq.com`

`wss://gw.tv.s.qq.com`

本文主要提供生成腾讯云智能数智人 aPaaS 平台接口调用 URL 的方式，以及常见编程语言 demo，语言类型包括 Golang、Java、C++、Python、JavaScript。

二、必要参数获取

AppKey、AccessToken 请在 [数智人平台资源管理中心](#) 页面获取：

腾讯云智能 | 数智人平台 标准版 | 首页 | 形象设定 | 音视频播报 | 会话互动 | **资源管理中心** | 文档

免费体验 获取更多资源，可前往[购买页](#) [购买](#)

音视频播报PaaS/SaaS

形象资产：多种2D小样本、3D形象
服务资产：播报时长60mins、免训练（视频素材）15mins、免训练（照片素材）15mins

[去体验](#) [领用](#)

基本信息

客户名称 `levli` UIN `81554` 账号类型 `账号`

AppKey `6c***1f` AccessToken `75***81`

可用定制配额 (个)
37 2D小样本 25 声音复刻

在用形象资产 (个) 75个即将失效
154 形象定制 412 形象租赁

在用声音资产 (个)
368

在用服务资产 (套) 2026/03/13 21:37:09 即将失效
5 路并发 574 小时播报时长

三、公共参数

在调用 aPaaS 任一接口时，都需在 URL 中以 QueryString 形式携带下面的公共参数。

参数	类型	必须	说明
appkey	string	是	根据各接口文档提供的方式获取到的 appkey。
timesta	string	是	请求时间戳，单位：秒。时间戳需要和当前时间的差异不能超过五分钟，

mp			否则会鉴权失败。
requestId	string	否	只有 建立会话长链接 接口需要携带该参数, 其他接口不需要携带。获取方式请参考对应接口文档。
signature	string	是	请求签名 (参考 签名方法)。

携带参数示例:

示例1: https 协议, 注意每个参数的值替换成实际的值

```
https://gw.tvs.qq.com/v2/ivh/sessionmanager/sessionmanagerservice/createsession?appkey={{appkey}}&timestamp={{timestamp}}&signature={{signature}}
```

示例2: 建立会话长链接

```
wss://gw.tvs.qq.com/v2/ws/ivh/interactdriver/interactdriverservice/commandchannel?appkey={{appkey}}&requestid={{requestid}}&timestamp={{timestamp}}&signature={{signature}}
```

⚠ 注意:

[建立会话长链接](#) 接口的 url 会有 appkey、requestId、timestamp 三个参数, 注意都要拼接。

四、签名方法

请求参数签名步骤如下:

1. signature 签名规则如下, 以下为示例说明 (示例为伪代码仅供参考):

```
appkey = example_appkey
accesstoken = example_accesstoken
域名路由 = https://api.example.com/v2/ivh/example_uri
```

2. 根据对应接口文档, 对 query 参数中除 signature 之外的所有需要的参数按字典序进行排序, 作为签名原文, 用于生成签名。示例中仅使用 appkey 和 timestamp 两个参数 (各个接口需要的参数请在相关文档中确认), 排序拼接后的字符串示例为:

```
appkey=example_appkey&timestamp=1717639699
```

3. 对签名原文使用 accessToken 进行 HmacSha256 加密, 之后再 base64 编码:

```
hashBytes =
HmacSha256 ("appkey=example_appkey&timestamp=1717639699", "example_accessToken")
signature = Base64Encode(hashBytes)
```

4. 得到 signature 签名值为:

```
aCNWYzZdplxWVo+JsqzZc9+J9XrwWWITfX3eQpsLVno=
```

5. 将 signature 值进行 urlencode (必须进行 URL 编码, 否则将导致鉴权失败) 后拼接得到最终请求 URL 为:

```
https://api.example.com/v2/ivh/example_uri?
appkey=example_appkey&timestamp=1717639699&signature=aCNWYzZdplxWVo%2BJs
qzZc9%2BJ9XrwWWITfX3eQpsLVno%3D
```

五、签名生成 Demo

用户只需要将访问对应接口需要的除 signature 以外所有公共参数填入 demo 中相应位置, 即可生成请求签名并拼接成访问接口的完整 URL 。

Golang

```
package main

import (
    "crypto/hmac"
    "crypto/sha256"
    "encoding/base64"
    "fmt"
    "net/url"
    "sort"
    "strconv"
    "time"
)

func GenSignature(signingContent string, accessToken string) string {

    // 计算 HMAC-SHA256 值
```

```
h := hmac.New(sha256.New, []byte(accessToken))
h.Write([]byte(signingContent))

// 将 HMAC-SHA256 值进行 Base64 编码
hashInBase64 := base64.StdEncoding.EncodeToString(h.Sum(nil))

// URL encode
encodeSign := url.QueryEscape(hashInBase64)

// 拼接签名
signature := "&signature=" + encodeSign

return signature
}

func GenReqURL(parameter map[string]string, accessToken string,
baseURL string) string {
    // 按字典序拼接待计算签名的字符串
    signingContent := ""
    pathKey := make([]string, 0, len(parameter))
    for k := range parameter {
        pathKey = append(pathKey, k)
    }
    sort.Strings(pathKey)
    for _, k := range pathKey {
        if signingContent != "" {
            signingContent += "&"
        }
        signingContent += k + "=" + parameter[k]
    }

    // 计算签名
    signature := GenSignature(signingContent, accessToken)

    // 拼接访问接口的完整URL
    return baseURL + "?" + signingContent + signature
}

func main() {
    baseUrl := "https://api.example.com/v2/ivh/example_uri"
    accesstoken := "example_accesstoken"
```

```
wssUrl := "wss://api.example.com/v2/ws/ivh/example_uri"
// 示例一 (访问需要提供 appkey 和 timestamp 参数的接口):
// 用户按需填入生成签名所需的公共参数
parameter := map[string]string{
    "appkey": "example_appkey",
    // 用户提供的时间戳应保证单位为秒且与当前时间相差不得超过五分钟
    // 示例时间戳:
    // "timestamp": "1717639699",
    // 推荐使用下面的语句生成当前时间戳:
    "timestamp": strconv.FormatInt(time.Now().Unix(), 10),
}
url1 := GenReqURL(parameter, accesstoken, baseUrl)
// 使用示例时间戳输出应当如下:
// Example 1:https://api.example.com/v2/ivh/example_uri?
appkey=example_appkey&timestamp=1717639699&signature=aCNWYzZdplxWVo%2B
JsqzZc9%2BJ9XrwWWITfX3eQpsLVno%3D
fmt.Println("Example 1:" + url1)

// 示例二 (访问需要提供 appkey, requestID 和 timestamp 参数的接口):
// 用户按需填入生成签名所需的公共参数
parameter = map[string]string{
    "appkey": "example_appkey",
    "requestid": "example_requestid",
    // 用户提供的时间戳应保证单位为秒且与当前时间相差不得超过五分钟
    // 示例时间戳:
    // "timestamp": "1717639699",
    // 用户若自己提供时间戳应保证单位为秒且与当前时间相差不得超过五分钟
    "timestamp": strconv.FormatInt(time.Now().Unix(), 10),
}
url2 := GenReqURL(parameter, accesstoken, wssUrl)
// 使用示例时间戳输出应当如下:
// Example 2:wss://api.example.com/v2/ws/ivh/example_uri?
appkey=example_appkey&requestid=example_requestid&timestamp=1717639699
&signature=QVenICk0VhtHGYZKXM6IC%2BW1CjZC1joSr%2Fx0gfKKYT4%3D
fmt.Println("Example 2:" + url2)
}
```

Java

```
import java.util.*;
import java.util.stream.Collectors;
import java.time.Instant;
import java.net.URLEncoder;
import java.nio.charset.StandardCharsets;
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import java.util.Base64;
import java.security.InvalidKeyException;
import java.security.NoSuchAlgorithmException;
import java.io.UnsupportedEncodingException;

public class Presigned {

    public static String GenSignature(String signingContent, String
accessToken) {
        try {
            // 计算 HMAC-SHA256 值
            Mac sha256_HMAC = Mac.getInstance("HmacSHA256");
            SecretKeySpec secret_key = new
SecretKeySpec(accessToken.getBytes(StandardCharsets.UTF_8),
"HmacSHA256");
            sha256_HMAC.init(secret_key);

            String hashInBase64 =
Base64.getEncoder().encodeToString(sha256_HMAC.doFinal(signingContent.
getBytes(StandardCharsets.UTF_8)));

            // URL encode
            String encodeSign = URLEncoder.encode(hashInBase64,
StandardCharsets.UTF_8.toString());

            // 拼接签名
            String signature = "&signature=" + encodeSign;

            return signature;
        } catch (NoSuchAlgorithmException | InvalidKeyException |
```

```
UnsupportedEncodingException e) {
    e.printStackTrace();
    return null;
}

}

    public static String GenReqURL(Map<String, String> parameter,
String accessToken, String baseUrl) {
    // 按字典序拼接待计算签名的字符串
    String signingContent = parameter.entrySet().stream()
        .sorted(Map.Entry.comparingByKey())
        .map(entry -> entry.getKey() + "=" + entry.getValue())
        .collect(Collectors.joining("&"));

    // 计算签名
    String signature = GenSignature(signingContent, accessToken);

    // 拼接访问接口的完整URL
    return baseUrl + "?" + signingContent + signature;
}

public static void main(String[] args) {
    String baseUrl = "https://api.example.com/v2/ivh/example_uri";
    String accesstoken = "example_accesstoken";
    String wssUrl = "wss://api.example.com/v2/ws/ivh/example_uri";

    // 示例一 (访问需要提供 appkey 和 timestamp 参数的接口):
    // 用户按需填入生成签名所需的公共参数
    Map<String, String> parameter = new TreeMap<>();
    parameter.put("appkey", "example_appkey");
    // 用户提供的时间戳应保证单位为秒且与当前时间相差不得超过五分钟
    // 示例时间戳:
    // parameter.put("timestamp", "1717639699");
    // 推荐使用下面的语句生成当前时间戳:
    parameter.put("timestamp",
String.valueOf(Instant.now().getEpochSecond()));
    String url = GenReqURL(parameter, accesstoken, baseUrl);
    // 使用示例时间戳输出应当如下:
    // Example 1:https://api.example.com/v2/ivh/example_uri?
appkey=example_appkey&timestamp=1717639699&signature=aCNWYzZdplxWVo%2B
JsqzZc9%2BJ9XrwWWITfX3eQpsLVno%3D
```

```
System.out.println("Example 1:" + url);

// 示例二 (访问需要提供 appkey, requestID 和 timestamp 参数的接口):
parameter.clear();
parameter.put("appkey", "example_appkey");
parameter.put("requestid", "example_requestid");
// 用户提供的戳应保证单位为秒且与当前时间相差不超过五分钟
// 示例时间戳:
// parameter.put("timestamp", "1717639699");
// 推荐使用下面的语句生成当前时间戳:
parameter.put("timestamp",
String.valueOf(Instant.now().getEpochSecond()));
url = GenReqURL(parameter, accesstoken, wssUrl);
// 使用示例时间戳输出应当如下:
// Example 2:wss://api.example.com/v2/ws/ivh/example_uri?
appkey=example_appkey&requestid=example_requestid&timestamp=1717639699
&signature=QVenICk0VHtHGYZKXM6IC%2BW1CjZC1joSr%2Fx0gfKKYT4%3D
System.out.println("Example 2:" + url);
}
}
```

C++

```
#include<iostream>
#include <map>
#include<string>
#include<vector>
#include<algorithm>
#include <ctime>
#include <iomanip>
#include <sstream>
#include<openssl/hmac.h>
#include<openssl/sha.h>
```

```
using namespace std;

static const string base64_chars =
    "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
    "abcdefghijklmnopqrstuvwxyz"
    "0123456789+/";

string url_encode(const string &value) {
    ostringstream escaped;
    escaped.fill('0');
    escaped<< hex;

    for (size_t i = 0; i < value.length(); ++i) {
        char c = value[i];
        if (isalnum(c) || c == '-' || c == '_' || c == '.' || c ==
'~') {
            escaped << c;
        } else {
            escaped<< uppercase;
            escaped << '%'<< setw(2)<< int((unsigned char) c);
            escaped<< nouppercase;
        }
    }

    return escaped.str();
}

string base64_encode(unsigned char const* bytes_to_encode, unsigned
int in_len) {
    string ret;
    int i = 0;
    int j = 0;
    unsigned char char_array_3[3];
    unsigned char char_array_4[4];
    while (in_len--) {
        char_array_3[i++] = *(bytes_to_encode++);
```

```
    if (i == 3) {
        char_array_4[0] = (char_array_3[0] & 0xfc) >> 2;
        char_array_4[1] = ((char_array_3[0] & 0x03) << 4) +
        ((char_array_3[1] & 0xf0) >> 4);
        char_array_4[2] = ((char_array_3[1] & 0x0f) << 2) +
        ((char_array_3[2] & 0xc0) >> 6);
        char_array_4[3] = char_array_3[2] & 0x3f;

        for(i = 0; (i <4) ; i++)
            ret += base64_chars[char_array_4[i]];
        i = 0;
    }
}

if (i)
{
    for(j = i; j < 3; j++)
        char_array_3[j] = '\\0';
    char_array_4[0] = (char_array_3[0] & 0xfc) >> 2;
    char_array_4[1] = ((char_array_3[0] & 0x03) << 4) +
    ((char_array_3[1] & 0xf0) >> 4);
    char_array_4[2] = ((char_array_3[1] & 0x0f) << 2) +
    ((char_array_3[2] & 0xc0) >> 6);
    char_array_4[3] = char_array_3[2] & 0x3f;
    for (j = 0; (j < i + 1); j++)
        ret += base64_chars[char_array_4[j]];
    while((i++ < 3))
        ret += '=';
}
return ret;
}

string GenSignature(const string& signingContent, const string&
accessToken) {

    // 计算 HMAC-SHA256 值
    unsigned char hash[EVP_MAX_MD_SIZE];
    unsigned int hashLength;
```

```
HMAC(EVP_sha256(), accessToken.c_str(), accessToken.length(),
(unsigned char *) signingContent.c_str(), signingContent.length(),
hash, &hashLength);

// 将 HMAC-SHA256 值进行 Base64 编码
string hashInBase64 = base64_encode(hash, hashLength);

// URL encode
string encodeSign = url_encode(hashInBase64);

// 拼接签名
string signature = "&signature=" + encodeSign;

return signature;
}

string GenReqURL(const map<string, string>& parameter, const string&
accessToken, const string& baseURL) {
    // 按字典序拼接待计算签名的字符串
    string signingContent;
    vector<string> pathKey;
    for (const auto& p : parameter) {
        pathKey.push_back(p.first);
    }
    sort(pathKey.begin(), pathKey.end());
    for (const auto& k : pathKey) {
        if (!signingContent.empty()) {
            signingContent += "&";
        }
        signingContent += k + "=" + parameter.at(k);
    }

    // 计算签名
    string signature = GenSignature(signingContent, accessToken);
}
```

```
// 拼接访问接口的完整URL
return baseUrl + "?" + signingContent + signature;
}

// 编译时需要链接到OpenSSL库, 可以使用下面的命令:
// g++ presigned.cpp -o presigned -lcrypto

int main() {
    string baseUrl = "https://api.example.com/v2/ivh/example_uri";
    string accesstoken = "example_accesstoken";
    string wssUrl = "wss://api.example.com/v2/ws/ivh/example_uri";

    // 示例一 (访问需要提供 appkey 和 timestamp 参数的接口):
    // 用户按需填入生成签名所需的公共参数
    map<string, string> parameter1 = {
        {"appkey", "example_appkey"},
        // 用户提供的戳应保证单位为秒且与当前时间相差不得超过五分钟
        // 示例时间戳:
        // {"timestamp", "1717639699"},
        // 推荐使用下面的语句生成当前时间戳:
        {"timestamp", to_string(time(NULL))}
    };
    string url = GenReqURL(parameter1, accesstoken, baseUrl);
    // 使用示例时间戳输出应当如下:
    // Example 1:https://api.example.com/v2/ivh/example_uri?
    appkey=example_appkey&timestamp=1717639699&signature=aCNWYzZdplxWVo%2B
    JsqzZc9%2BJ9XrwWWITfX3eQpsLVno%3D
    cout << "Example 1:"<< url<< endl;

    // 示例二 (访问需要提供 appkey, requestId 和 timestamp 参数的接口):
    // 用户按需填入生成签名所需的公共参数
    map<string, string> parameter2 = {
        {"appkey", "example_appkey"},
        {"requestid", "example_requestid"},
        // 用户提供的戳应保证单位为秒且与当前时间相差不得超过五分钟
    };
}
```

```
// 示例时间戳:
// {"timestamp" , "1717639699"},
// 推荐使用下面的语句生成当前时间戳:
{"timestamp", to_string(time(NULL))}
};
url = GenReqURL(parameter2, accesstoken, wssUrl);
// 使用示例时间戳输出应当如下:
// Example 2:wss://api.example.com/v2/ws/ivh/example_uri?
appkey=example_appkey&requestid=example_requestid&timestamp=1717639699
&signature=QVenICk0VHtHGYZKXM6IC%2BW1CjZC1joSr%2Fx0gfKKYT4%3D
cout << "Example 2:"<< url<< endl;

return 0;
}
```

Python

```
import hmac
import hashlib
import time
import base64
from urllib.parse import quote

# 用户可以在函数内部生成时间戳, 只需要传入appkey和accessToken即可获取访问接口所
# 需的公共参数和签名
def GenSignature(signing_content, access_token):

    # 计算 HMAC-SHA256 值
    h = hmac.new(access_token.encode(), signing_content.encode(),
hashlib.sha256)
```

```
# 将 HMAC-SHA256 值进行 Base64 编码
hash_in_base64 = base64.b64encode(h.digest()).decode()

# URL encode
encode_sign = quote(hash_in_base64)

# 拼接签名
signature = f"&signature={encode_sign}"

return signature

def GenReqURL(parameter, access_token, base_url):
    # 按字典序拼接待计算签名的字符串
    signing_content = '&'.join(f'{k}={parameter[k]}' for k in
sorted(parameter.keys()))

    # 计算签名
    signature = GenSignature(signing_content, access_token)

    # 拼接访问接口的完整 URL
    return f'{base_url}?{signing_content}{signature}'

def main():
    base_url = 'https://api.example.com/v2/ivh/example_uri'
    access_token = 'example_accesstoken'
    wss_url = 'wss://api.example.com/v2/ws/ivh/example_uri'

    # 示例一 (访问需要提供 appkey 和 timestamp 参数的接口):
    # 用户按需填入生成签名所需的公共参数
```

```
parameter1 = {
    'appkey': 'example_appkey',
    # 用户提供的时间戳应保证单位为秒且与当前时间相差不得超过五分钟
    # 示例时间戳:
    # 'timestamp': '1717639699'
    # 推荐使用下面的语句生成当前时间戳:
    'timestamp': int(time.time()) # 使用当前时间戳 (单位: 秒)
}

url1 = GenReqURL(parameter1, access_token, base_url)
# 使用示例时间戳输出应当如下:
# Example 1:https://api.example.com/v2/ivh/example_uri?
appkey=example_appkey&timestamp=1717639699&signature=aCNWYzZdplxWVo%2B
JsqzZc9%2BJ9XrwWWITfx3eQpsLVno%3D
print('Example 1:', url1)

# 示例二 (访问需要提供 appkey, requestID 和 timestamp 参数的接口):
# 用户按需填入生成签名所需的公共参数
parameter2 = {
    'appkey': 'example_appkey',
    'requestid': 'example_requestid',
    # 用户提供的时间戳应保证单位为秒且与当前时间相差不得超过五分钟
    # 示例时间戳:
    # 'timestamp': '1717639699'
    # 推荐使用下面的语句生成当前时间戳:
    'timestamp': int(time.time()) # 使用当前时间戳 (单位: 秒)
}

url2 = GenReqURL(parameter2, access_token, wss_url)
# 使用示例时间戳输出应当如下:
# Example 2:wss://api.example.com/v2/ws/ivh/example_uri?
appkey=example_appkey&requestid=example_requestid&timestamp=1717639699
&signature=QVenICk0VHtHGYZKXM6IC%2BW1CjZC1joSr%2Fx0gfKKYT4%3D
print('Example 2:', url2)

if __name__ == '__main__':
    main()
```

JavaScript

```
const crypto = require('crypto');

// 用户可以在函数内部生成时间戳，只需要传入appkey和accessToken即可获取访问接口所需的公共参数和签名
function GenSignature(signingContent, accessToken) {
  // 计算 HMAC-SHA256 值
  const hmac = crypto.createHmac('sha256', accessToken);
  hmac.update(signingContent);

  // 将 HMAC-SHA256 值进行 Base64 编码
  const hashInBase64 = hmac.digest('base64');

  // URL encode
  const encodeSign = encodeURIComponent(hashInBase64);

  // 拼接签名
  const signature = `&signature=${encodeSign}`;

  return signature;
}

function GenReqURL(parameter, accessToken, baseUrl) {
  // 按字典序拼接待计算签名的字符串
  let signingContent = '';
  const pathKey = Object.keys(parameter).sort();
  for (const k of pathKey) {
    if (signingContent !== '') {
      signingContent += '&';
    }
    signingContent += `${k}=${parameter[k]}`;
  }
}
```

```
// 计算签名
const signature = GenSignature(signingContent, accessToken);

// 拼接访问接口的完整URL
return `${baseUrl}?${signingContent}${signature}`;
}

function main() {
  const baseUrl = 'https://api.example.com/v2/ivh/example_uri';
  const accesstoken = 'example_accesstoken';
  const wssUrl = 'wss://api.example.com/v2/ws/ivh/example_uri';

  // 示例一 (访问需要提供 appkey 和 timestamp 参数的接口):
  // 用户按需填入生成签名所需的公共参数
  const parameter1 = {
    appkey: 'example_appkey',
    // 用户提供的时间戳应保证单位为秒且与当前时间相差不得超过五分钟
    // 示例时间戳:
    // timestamp: '1717639699',
    // 推荐使用下面的语句生成当前时间戳:
    timestamp: Math.floor(Date.now() / 1000), // 使用当前时间戳 (单位:
秒)
  };
  const url1 = GenReqURL(parameter1, accesstoken, baseUrl);
  // 使用示例时间戳输出应当如下:
  // Example 1:https://api.example.com/v2/ivh/example_uri?
appkey=example_appkey&timestamp=1717639699&signature=aCNWYzZdplxWVo%2B
JsqzZc9%2BJ9XrwWWITfX3eQpsLVno%3D
  console.log('Example 1:', url1);

  // 示例二 (访问需要提供 appkey, requestId 和 timestamp 参数的接口):
  const parameter2 = {
    appkey: 'example_appkey',
    requestid: 'example_requestid',
    // 用户提供的时间戳应保证单位为秒且与当前时间相差不得超过五分钟
    // 示例时间戳:

```

```
// timestamp: '1717639699',  
// 推荐使用下面的语句生成当前时间戳:  
timestamp: Math.floor(Date.now() / 1000), // 使用当前时间戳 (单位:  
秒)  
};  
const url2 = GenReqURL(parameter2, accesstoken, wssUrl);  
// 使用示例时间戳输出应当如下:  
// Example 2:wss://api.example.com/v2/ws/ivh/example_uri?  
appkey=example_appkey&requestid=example_requestid&timestamp=1717639699  
&signature=QVenICk0VhtHGYZKXM6IC%2BW1CjZC1joSr%2Fx0gfKKYT4%3D  
console.log('Example 2:', url2);  
}  
  
main();
```

六、接口调用注意事项

发送请求调用数智人相关接口时，请求体都要求有 header 和 payload 参数，请注意不要把这里的 header 理解为请求头，它也是请求体的一个参数。

以下为请求体的正确写法示例：

```
{  
  "Header": {  
  
  },  
  "Payload": {  
    "ReqId": "d7aa08da33dd4a662ad5be508c5b77cf",  
    "ParamKey1": "paramValue1",  
    "ParamKey2": "paramValue2"  
  }  
}
```

以下为请求体的错误写法示例：

```
{  
  "ReqId": "d7aa08da33dd4a662ad5be508c5b77cf",  
  "ParamKey1": "paramValue1",  
}
```

```
"ParamKey2": "paramValue2"  
}
```