



视频点播 开发指南

文档版本: 20220713



法律声明

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。 如果您阅读或使用本文档,您的阅读或使用行为将被视为对本声明全部内容的认可。

- 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档,且仅能用 于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息,您应当严格 遵守保密义务;未经阿里云事先书面同意,您不得向任何第三方披露本手册内容或 提供给任何第三方使用。
- 未经阿里云事先书面许可,任何单位、公司或个人不得擅自摘抄、翻译、复制本文 档内容的部分或全部,不得以任何方式或途径进行传播和宣传。
- 由于产品版本升级、调整或其他原因,本文档内容有可能变更。阿里云保留在没有 任何通知或者提示下对本文档的内容进行修改的权利,并在阿里云授权通道中不时 发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠 道下载、获取最新版的用户文档。
- 4. 本文档仅作为用户使用阿里云产品及服务的参考性指引,阿里云以产品及服务的"现状"、"有缺陷"和"当前功能"的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引,但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的,阿里云不承担任何法律责任。在任何情况下,阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害,包括用户使用或信赖本文档而遭受的利润损失,承担责任(即使阿里云已被告知该等损失的可能性)。
- 5. 阿里云网站上所有内容,包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计,均由阿里云和/或其关联公司依法拥有其知识产权,包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意,任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外,未经阿里云事先书面同意,任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称(包括但不限于单独为或以组合形式包含"阿里云"、"Aliyun"、"万网"等阿里云和/或其关联公司品牌,上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司)。
- 6. 如若发现本文档存在任何错误,请与阿里云取得直接联系。

通用约定

格式	说明	样例
⚠ 危险	该类警示信息将导致系统重大变更甚至故 障,或者导致人身伤害等结果。	♪ 危险 重置操作将丢失用户配置数据。
▲ 警告	该类警示信息可能会导致系统重大变更甚 至故障,或者导致人身伤害等结果。	警告 重启操作将导致业务中断,恢复业务 时间约十分钟。
〔〕 注意	用于警示信息、补充说明等,是用户必须 了解的内容。	大) 注意 权重设置为0,该服务器不会再接受新 请求。
? 说明	用于补充说明、最佳实践、窍门等,不是 用户必须了解的内容。	⑦ 说明 您也可以通过按Ctrl+A选中全部文件。
>	多级菜单递进。	单击设置> 网络> 设置网络类型。
粗体	表示按键、菜单、页面名称等UI元素。	在 结果确认 页面,单击 确定 。
Courier字体	命令或代码。	执行 cd /d C:/window 命令,进入 Windows系统文件夹。
斜体	表示参数、变量。	bae log listinstanceid
[] 或者 [alb]	表示可选项,至多选择一个。	ipconfig [-all -t]
{} 或者 {alb}	表示必选项,至多选择一个。	switch {act ive st and}

目录

1.开发指南概述	80
2.点播地域标识	11
3.账号和授权	12
3.1. 概述	12
3.2. 创建RAM用户并授权	14
3.3. 创建角色并进行STS临时授权	15
3.4. 基于RAM Policy实现自定义授权	20
3.5. 凭证方式与STS方式对比	24
4.媒体上传	25
4.1. 媒体上传概述	25
4.2. 工具上传	33
4.2.1. 控制台上传	33
4.2.2. PC端上传	37
4.3. 开发上传	38
4.3.1. 如何选择上传方式	38
4.3.2. 授权访问点播服务	39
4.3.2.1. 上传地址和凭证	39
4.3.2.2. STS临时Token	42
4.3.3. 基于上传SDK上传	46
4.3.4. 基于OSS原生SDK上传	46
4.3.5. URL批量拉取上传	61
4.3.6. 基于OSS API上传	62
5.媒资管理	63
5.1. 概述	63
5.2. 获取媒资信息	65
5.3. 更新媒资信息	67

5.4. 搜索媒资信息	69
5.5. 下载媒体文件	72
5.6. 删除媒体文件	75
5.7. 媒资管理FAQ	76
.媒体处理	78
6.1. 概述	78
6.2. 音视频转码	79
6.3. 视频截图	83
6.4. 视频动图	91
6.5. 视频水印	93
6.6. 工作流	100
6.7. 推荐模板配置	104
6.8. 多码率自适应	107
6.9. 转码生成文件名称	110
6.10. 媒体处理FAQ	112
.媒体审核	113
7.1. 概述	113
7.2. 智能审核	115
7.3. 人工审核	117
7.4. 审核安全IP	118
.视频Al	120
8.1. 概述	120
8.2. 视频DNA	121
8.3. 智能标签	123
.音视频播放	125
9.1. 概述	125
9.2. 获取播放地址播放	127
9.3. 通过播放凭证播放	129

9.4. 使用阿里云播放器	130
9.5. 播放常见设置	131
10.云剪辑	134
10.1. 概述	134
10.2. 剪切合并	138
10.3. 音频处理	141
10.4. 图片叠加	149
10.5. 文字叠加	153
10.6. 特效	157
11.直播转点播	159
11.1. 概述	159
11.2. 常见问题FAQ	161
11.3. 场景实践	162
12.事件通知	168
12.1. 概述	168
12.2. HTTP回调事件通知	171
12.3. MNS可靠事件通知	173
12.4. 事件列表	175
12.4.1. 视频上传完成	175
12.4.2. 图片上传完成	175
12.4.3. 单个清晰度转码完成	176
12.4.4. 全部清晰度转码完成	178
12.4.5. 视频截图完成	181
12.4.6. 视频动图完成	185
12.4.7. 直转点视频录制完成	187
12.4.8. 直转点录制视频合成开始	188
12.4.9. URL上传视频完成	189
12.4.10. 人工审核完成	191

12.4.11. 智能审核完成	192
12.4.12. 音视频分析完成	195
12.4.13. 视频DNA完成	197
12.4.14. 智能标签作业完成	198
12.4.15. 辅助媒资上传完成	199
12.4.16. 媒体合成完成	200
12.4.17. 媒体删除完成	201
12.4.18. 媒资基础信息变更	203
12.4.19. 音视频刷新或预热完成	205
13.视频安全	208
13.1. 概述	208
13.2. 访问限制	212
13.3. URL鉴权	213
13.4. 阿里云视频加密	216
13.5. HLS标准加密	219
13.6. HLS(M3U8)标准加密改写	230

1.开发指南概述

视频点播提供一站式的音视频点播解决方案,功能涵盖视频采集、编辑、上传、自动化转码处理、媒体资源管理、高效云剪辑处理、分发加速、视频播放,开发指南为您介绍了以上功能的介绍、使用说明等。

整体流程

使用视频点播实现音视频上传、存储、处理和播放的整体流程如下:



⑦ 说明

以上是使用了转码、播放凭证播放等方式的标准流程,如没有开启转码,或开启了媒体审核等,流程会有所不同。

- 1. 用户获取上传授权。
- 2. VOD下发上传地址和凭证及VideoId。
- 3. 用户上传视频并保存视频ID(Videold)。
- 4. 用户服务端获取播放授权。
- 5. 用户客户端请求播放地址与凭证, VOD下发播放地址与带时效的播放凭证。
- 6. 用户服务端将播放凭证下发给客户端完成视频播放。

接入与存储

使用点播服务前首选需要关注服务区域、接入区域和存储区域。点播服务在全球多个区域提供就近接入,每个接入地址支持对应的存储区域,不可跨区域调用。点播中心和访问域名为您介绍了服务区域与API接入区域、支持的存储区域之间的对应关系。

全局配置

配置分类	描述	参考文档
账号与授权	视频点播支持访问控制(RAM)服务,并支持临时安全令牌(STS)服务。您可以 通过如下方式进行授权操作: • 对RAM用户用系统授权策略授权 • 对RAM用户自定义授权 • 对角色进行STS临时授权	账号与授权
多应用体系	视频点播支持将同一账号下多个使用方的资源、配置和数据进行隔离。您可以通过 管理应用和对身份实体用应用授权策略授权的方式构建多应用体系。	多应用体系
事件通知	视频点播支持通过消息回调的方式获取事件通知。 HTTP回调(兼容HTTPS) 阿里云消息服务(MNS)回调 	事件通知
视频安全	视频点播提供多种安全机制用于保障视频内容安全: 访问限制 播放中心鉴权 业务方二次鉴权 视频加密 安全下载 	视频安全

功能使用

功能分类	描述	参考文档
媒体上传	视频点播支持上传媒体文件(音频、视频、图片等)到点播存储,上传前需要从点播服务获取上传地址和凭证。支持的上传方式有: 工具上传 • 控制台 • PC客户端 开发上传 • 上传SDK(含服务端、客户端) • URL拉取上传 • 基于OSS原生SDK上传 • 基于OSS原生SDK上传	媒体上传
媒资管理	视频点播支持管理上传或媒体处理后产生的媒资(基础信息、源文件信息、播放信 息和AI数据),支持获取、更新、搜索、下载和删除。	媒资管理

功能分类	描述	参考文档
媒体处理	视频点播提供音视频转码、视频截图、视频动图、视频水印的基础媒体处理能力, 还提供了智能审核、云剪辑、AI处理等强大功能。	媒体处理 媒体审核 云剪辑 视频Al
音视频播放	视频点播支持通过控制台预览、集成阿里云播放器SDK和集成第三方播放器三种方 式进行播放音视频文件。	音视频播放
直播转点播	视频点播支持将直播流同步录制为点播视频。	直播转点播

2. 点播地域标识

点播服务在全球多个地域提供就近接入,每个接入地址有对应的存储地域。点播服务不可跨区域调用。本文介绍服务区域对应的API接入地域与标识、接入地址(访问域名)、对应的存储地域与存储地域标识。

简介

- API接入地域标识对应API/SDK的RegionId参数。更多信息请参见API概览。
- **接入地址(访问域名)**对应自行构造HTTP(HTTPS)请求访问点播API场景下的请求URL中的服务地址。更多 信息请参见<mark>请求结构</mark>。
- 存储地域标识对应上传过程中可以指定的StorageRegion。更多信息请参见媒体上传概述。
- 每个服务区域只支持该对应区域的存储,部分上海服务区域老用户已添加过北京存储的不受影响,请继续使用 上海区域服务地址进行接口调用(详情请参考控制台左上角的区域信息)。

接入区域和存储区域

服务区 域	API接入地域	API接入地域标 识 (API Region)	接入地址(访问域名)	存储地域	存储地域标识 (Storage Region)
上海	华东2(上海)	cn-shanghai	vod.cn-shanghai.aliyuncs.com	华东2(上海)	cn-shanghai
北京	华北2(北京)	cn-beijing	vod.cn-beijing.aliyuncs.com	华北2(北京)	cn-beijing
深圳	华南1(深圳)	cn-shenzhen	vod.cn-shenzhen.aliyuncs.com	华南1(深圳)	cn-shenzhen
新加坡	新加坡	ap- southeast-1	vod.ap-southeast- 1.aliyuncs.com	新加坡	ap- southeast-1
印度尼 西亚	印度尼西亚 (雅加达)	ap- southeast-5	vod.ap-southeast- 5.aliyuncs.com	印度尼西亚 (雅加达)	ap- southeast-5
印度	印度(孟买)	ap-south-1	vod.ap-south-1.aliyuncs.com	印度(孟买)	ap-south-1
德国	德国(法兰克 福)	eu-central-1	vod.eu-central-1.aliyuncs.com	德国(法兰克 福)	eu-central-1
日本	日本(东京)	ap-northeast- 1	vod.ap-northeast- 1.aliyuncs.com	日本(东京)	ap- northeast-1

3.账号和授权

3.1. 概述

视频点播会对每一次发起操作请求的用户身份进行验证,通过AccessKey验证该账号是否拥有相应的权限。视频点 播支持主账号AK、RAM用户AK和STS临时AK,本文为您介绍这三种方式和对比,以及阿里云提供的系统授权策 略。

账号权限验证

您可以使用视频点播(VOD)提供的RESTful API接口或SDK(包括接口、上传、播放和短视频SDK)开发包访问 VOD。

VOD会针对每一次发起的请求,根据当前的操作验证用户身份,验证该账号是否拥有相应的权限。验证用户身份都需要使用AccessKey。

基本概念

• 访问控制 (RAM)

RAM(Resource Access Management)是阿里云提供的用户身份管理与资源访问控制服务。更多信息,请参 见什么是访问控制。

? 说明

RAM服务提供的权限的分割和管理,而并非资源本身的隔离,也就是RAM用户从属于主账号,并且这些子账号下不能拥有实际的任何资源,所有资源都属于主账号。如果想要进行资源隔离,可配合使用点播多应用体系,更多信息,请参见多应用体系概述。

• 阿里云账号

阿里云账号是阿里云资源归属、资源使用计量计费的基本主体。阿里云账号为其名下所拥有的资源付费,并对 其名下所有资源拥有完全控制权限。

• RAM用户

根据阿里云的主账号创建的RAM用户,每个RAM用户拥有自己AccessKey,可以和阿里云主账号一样正常的完成 有权限的操作。一般来说,这里的RAM用户可以理解为具有某种权限的用户,可以被认为是一个具有某些权限 的操作发起者。

● 角色(Role)

表示某种操作权限的虚拟概念,但是没有独立的登录密码和AccessKey。RAM用户可以扮演角色,扮演角色的时候的权限是该角色自身的权限。

⑦ 说明

RAM用户和角色可以类比为个人和其身份的关系:某人在公司的角色是员工,在家里的角色是父亲,在不同的场景扮演不同的角色,但是还是同一个人。在扮演不同的角色的时候也就拥有对应角色的权限。单独的员工或者父亲概念并不能作为一个操作的实体,只有有人扮演了之后才是一个完整的概念。

这里还可以体现一个重要的概念,那就是角色可以被多个不同的个人同时扮演。完成角色扮演之后,该个人就自动拥有该角色的所有权限。

● 授权策略 (RAM Policy)

授权策略使用语法结构描述一组权限,它可以精确地描述被授权的资源集、操作集以及授权条件。通过设置权 限策略,并给用户或用户组附加授权,就可以精确控制用户访问您名下哪些资源或服务的权限,比如限制您的 用户只拥有上传、播放或审核权限。

更多关于访问控制的概念,请参见基本概念。

AccessKey(访问密钥)

AccessKey(简称AK),包括访问身份验证中用到的AccessKey ID和AccessKey Secret。VOD通过使用AccessKey ID和AccessKey Secret对称加密的方法,来验证某个请求的发送者身份。

- AccessKey ID: 用于标识用户。
- AccessKey Secret:用户用于加密签名字符串,以及VOD用来验证签名字符串的密钥,AccessKey Secret必须 保密。
- AK对:指AccessKey ID和AccessKey Secret。

目前访问VOD使用的AK有如下3种类型:

● 主账号AK

主账号AK特指VOD开通者(即阿里云网站注册的账号)的AK,每个阿里云主账号提供的AK对拥有的资源有完全的权限。每个阿里云主账号能够同时拥有不超过5个启用或者禁用 AK对。

您可以登录AccessKey管理控制台,申请新增或删除AK对。每个AK对都有启用/禁用两种状态,只有启用的AK 对才能在身份验证时使用。

△ 警告

由于主账号AK有完全的权限,一旦泄露风险巨大,不建议使用其访问VOD服务。

• RAM用户AK

RAM是阿里云提供的资源访问控制服务。RAM用户AK指的是通过RAM被授权的AK。这组AK只能按照RAM定义的规则去访问VOD的资源。通过RAM,您可以集中管理您的用户(比如员工、系统或应用程序),以及控制用户可以访问您名下哪些资源的权限。比如能够限制您的用户只拥有视频播放权限。RAM用户从属于主账号,并且这些账号下不能拥有实际的任何资源,所有资源都属于主账号。

您可以登录RAM访问控制台创建RAM用户,获取AK,并授予相应权限。

● STS临时AK

STS(Security Token Service)是阿里云提供的临时访问凭证服务。STS临时AK指的是通过STS颁发的带时效性AK。这组AK只能按照STS定义的规则去访问VOD的资源,且会定期失效。

不同验证方式的对比

验证方式	风险	权限	时效	适用场景
主账号AK	极大	管理和操作VOD所有资 源的权限	启用后一直有 效	超级管理员进行操作,不建议在程序里使用,尤其 不要放到客户端。
RAM用户AK	较大	根据授权策略获得相应 的权限	启用后一直有 效	授权进行具体的上传、播放、管理等操作,可准备 多个子账号,如遇AK泄露(人员离职等)需要更 换;建议在服务端使用。

验证方式	风险	权限	时效	适用场景
STS临时AK	安全	根据授权策略获得相应 的权限	自定义过期时 间	移动端或Web端使用,需要自己部署服务端生成 STS临时AK,要处理好临时AK失效的情况。

除了以上三种验证方式,还可以使用上传凭证、播放凭证解决媒体上传和播放过程中的授权和安全问题。凭证方 式与STS方式的对比,请参见上传(播放)凭证和STS方式对比。

系统授权策略

VOD提供了四种系统授权策略,可对RAM用户或STS账号进行方便、精确的授权。

策略名称	说明	操作权限
AliyunVODFullAccess	管理和操作VOD所有资源的权限	VOD所有API
AliyunVODReadOnlyAccess	只读访问VOD所有资源的权限	VOD所有读取类API,如以Get 、 Describe 、Search、List开头的接口
AliyunVODPlayAuth	使用VOD播放视频的权限,包括使用播 放器SDK或播放相关的API	播放API: • GetVideoPlayAuth • GetPlayInfo • GetVideoInfo
AliyunVODUploadAuth	使用VOD上传的权限,包括使用上传 SDK或上传相关的API	上传API: • CreateUploadVideo • RefreshUploadVideo • CreateUploadImage

3.2. 创建RAM用户并授权

使用RAM用户可以帮助您有效避免AccessKey或者密码泄露导致的安全问题。本文为您介绍如何创建一个RAM用 户,并设置相应的权限。

前提条件

如果您之前没有使用过RAM控制台,需要先开通服务。

操作步骤

1. 创建RAM用户。

创建RAM用户请参见创建RAM用户。

完成第5步输入登录名称后,后续步骤需要按照下文操作。

? 说明

- 建议您将登录名设置为vod,本文后续描述都以登录名vod为例。
- RAM用户创建完成后,必须对用户进行授权操作,授权方法及具体权限请参见下文。
- 2. 在访问方式区域勾选OpenAPI调用访问,单击确定。

单击确定后会弹出手机验证窗口,完成验证码验证后,自动生成该RAM用户的AccessKey。

3. 单击用户信息右侧的复制,保存用户登录名称、登录密码、AK对(AccessKey ID、AccessKey Secret)等用 户信息。

⑦ 说明 请保存用户信息并妥善保管,用于后续的访问。

- 返回用户界面,这里显示一个名为vod的账号已创建。 创建完成之后,该RAM用户还是没有任何权限。
- 5. 单击用户信息右侧的添加权限。
- 6. 在选择权限区域选择授权范围,在输入框中输入 vod ,将筛选出的权限策略都单击选择,单击确定。

* 选择权限				
系统策略 自定义策略 十	- 新建权限策略		已选择 (4)	清空
vod		G	AliyunVODFullAccess	×
权限策略名称	备注		AliyunVODReadOnlyAccess	×
AliyunVODFullAccess	管理视频点播服务(VOD)的权限		AliyunVODPlayAuth	×
AliyunVODReadOnlyAccess	只读访问视频点播服务(VOD)的权限		Aliana VODU a la a d'Auth	~
AliyunVODPlayAuth	视频点播服务(VOD)播放权限		AliyunvODopioadAuth	^
AliyunVODUploadAuth	视频点播服务(VOD)上传权限			
确定取消				

授权策略的定义,请参见系统授权策略。

7. 授权完成之后如果该账号需要控制台登录等权限,也可以单击用户名称,在认证管理页面来完成操作。

3.3. 创建角色并进行STS临时授权

使用STS临时授权可以有效避免RAM用户密码泄露导致的安全风险,本文为您如何创建RAM用户,创建角色,并进行STS临时授权。

前提条件

您已经注册了阿里云账号并完成账号实名认证。注册地址请参见<mark>阿里云官网</mark>。注册指引请参见注册阿里云账号。实 名认证指引请参见个人实名认证或企业实名认证。

背景信息

RAM用户的权限是可长期使用的,易导致安全风险。建议您生成STS临时AK,自定义过期时间,并且指定复杂的 策略来对不同的RAM用户进行限制,仅提供最小的权限。

创建RAM用户

1. 创建RAM用户。

创建RAM用户请参见创建RAM用户。

? 说明

- 建议您将登录名设置为vod,本文后续描述都以登录名vod为例。
- RAM用户创建完成后,必须对用户进行授权操作,授权方法及具体权限请参见下文。

完成第5步输入登录名称后,后续步骤需按照下文操作。

2. 在访问方式区域勾选OpenAPI调用访问,单击确定。

单击确定后会弹出手机验证窗口,完成验证码验证后,自动生成该RAM用户的AccessKey。

3. 单击用户信息右侧的复制,保存用户登录名称、登录密码、AK对(AccessKey ID、AccessKey Secret)等用 户信息。

⑦ 说明 请保存用户信息并妥善保管,用于后续的访问。

4. 返回用户界面,显示账号已创建。创建完成之后,该RAM用户还是没有任何权限,单击右边的添加权限。

RAM 访问	列控制 / 用户			
用户	1			
1 R 道	AM 用户是一个身份实体,它通常代表您的组织中需要访问云 開 <mark>號的操作步骤如下:</mark> 1. 创建用户,并为用户设置登录密码(用户登录控制台场最 2. 添加用户到用户组(需要先创建用户组并充成对用户组的	资源的人员或应用程序。) 或创建 AccessKey (应用程序调用 API 场景) 。 受仅) 。		
创建用	的 输入登录名、用户 ID 或 AccessKey ID	Q		С
	用户登录名称/显示名称	备注	创建时间	操作
	vod@ vod		2020年10月29日 20:41:01	添加到用户组添加权限删除
	添加到用户组 添加权限			

5. 在选择权限区域,在输入框中输入AliyunSTSAssumeRoleAccess并单击选择该策略,单击确定。

* 选择权限				
系统策略 自定义策略	十 新建权限策略		已选择 (1)	清空
AliyunSTSAssumeRoleAccess		G	AliyunSTSAssumeRoleAccess	×
权限策略名称	备注			
AliyunSTSAssumeRoleAccess	调用STS服务AssumeRole接口的权限			
确定取消				

创建角色

1. 创建角色。

创建角色请参见创建角色。

创建完角色后,需要对角色进行授权。具体操作请参见下文。

⑦ 说明 建议您将角色名设置为voderole,本文后续描述都以登录名voderole为例。

2. 返回角色页面,找到新创建的角色,单击角色信息右侧的添加权限。

创建角色 输入角色名称或备注	Q			Ċ
角色名称	备注	创建时间		操作
voderole		2021年8月31日 15:16:33	添加权限 精确授权	删除

3. 在选择权限区域,在输入框中输入AliyunVODFullAccess并单击选择该策略,单击确定。

* 选择权限				
系统策略 自定义策略 十	- 新建权限策略		已选择 (1)	清空
AliyunVODFullAccess		S	AliyunVODFullAccess	×
权限策略名称	备注			
AliyunVODFullAccess	管理视频点播服务(VOD)的权限			
确 定 取消				

⑦ 说明 为控制风险,建议采用最小权限。

完成授权后,会生成一条授权成功的记录。

添加权限		×
授权范围 云账号全部资源		
✓ vodrol tom		✔ 1条成功
权限策略名称 AliyunVODFullAccess	✓ 授权成功	

STS授权访问

请下载和安装对应语言版本的STS SDK或直接调用AssumeRole接口。 角色和策略等的配置请参见下文示例(以Java 为例)。

调用方法,请参见<mark>什么是STS</mark>。

• Java代码示例

完成创建RAM用户和角色后,一切准备就绪,可以正式使用STS来授权访问了。

以一个简单的STS为例。为您介绍代码中的部分参数说明和代码示例,如下所示:

- RoleArn: 需要扮演的角色ID, 角色的ID可以在角色管理>基本信息中找到。
- RoleSessionName:角色会话名称。该参数为用户自定义参数。通常设置为调用该API的用户身份,例如:用 户名。在操作审计日志中,即使是同一个RAM角色执行的操作,也可以根据不同的RoleSessionName来区分 实际操作者,以实现用户级别的访问审计。长度为2~64个字符,可包含英文字母、数字、半角句号(.)、 at(@)、短划线(-)和下划线(_)。
- Policy: 在扮演角色的时候额外加上的一个权限限制。

? 说明

- 这里需要解释一下Policy,这里传入的Policy是用来限制扮演角色之后的临时凭证的权限。最后临时凭证获得的权限是角色的权限和这里传入的Policy的交集。
- 在扮演角色的时候传入Policy的原因是为了灵活性,比如只能现在使用CreateUploadVideo接口。
- DurationSeconds: 临时凭证的有效期,单位为秒,最小为900,最大为3600。
- id和secret:需要扮演角色的RAM用户,及其AccessKey。

```
package pop;
import com.aliyuncs.DefaultAcsClient;
import com.aliyuncs.exceptions.ClientException;
import com.aliyuncs.http.MethodType;
import com.aliyuncs.profile.DefaultProfile;
import com.aliyuncs.profile.IClientProfile;
import com.aliyuncs.sts.model.v20150401.AssumeRoleRequest;
import com.aliyuncs.sts.model.v20150401.AssumeRoleResponse;
import com.aliyuncs.vod.model.v20170321.CreateUploadVideoRequest;
import com.aliyuncs.vod.model.v20170321.CreateUploadVideoResponse;
/**
* @author jack
* @date 2020/5/25
*/
public class TestStsService {
   public static void main(String[] args) {
       // 只有RAM用户(子账号)才能调用 AssumeRole 接口
       // 阿里云主账号的AccessKeys不能用于发起AssumeRole请求
       // 请首先在RAM控制台创建一个RAM用户,并为这个用户创建AccessKeys
       String accessKeyId = "<access-key-id>";
       String accessKeySecret = "<access-key-secret>";
       // AssumeRole API 请求参数: RoleArn, RoleSessionName, Policy, and DurationSeconds
       // RoleArn 需要在 RAM 控制台上获取
       String roleArn = "<role-arn>";
       // RoleSessionName 角色会话名称,自定义参数
       String roleSessionName = "session-name";// 自定义即可
       // 定制你的policy
       String policy = "{\n" +
               " \"Version\": \"1\",\n" +
                 \"Statement\": [\n" +
               ...
                   {\n" +
                     \"Action\": \"vod:*\",\n" +
                    \"Resource\": \"*\",\n" +
               ...
                    \"Effect\": \"Allow\"\n" +
               " l\n" +
```

```
" ]\n" +
               "}";
        try {
           AssumeRoleResponse response = assumeRole(accessKeyId, accessKeySecret, roleArn, ro
leSessionName, policy);
           System.out.println("Expiration: " + response.getCredentials().getExpiration());
           System.out.println("Access Key Id: " + response.getCredentials().getAccessKeyId())
;
           System.out.println("Access Key Secret: " + response.getCredentials().getAccessKeyS
ecret());
           System.out.println("Security Token: " + response.getCredentials().getSecurityToken
());
           System.out.println("RequestId: " + response.getRequestId());
           createUploadVideo(response.getCredentials().getAccessKeyId(), response.getCredenti
als().getAccessKeySecret(), response.getCredentials().getSecurityToken());
       } catch (ClientException e) {
           System.out.println("Failed to get a token.");
           System.out.println("Error code: " + e.getErrCode());
           System.out.println("Error message: " + e.getErrMsg());
       }
    }
    static AssumeRoleResponse assumeRole(String accessKeyId, String accessKeySecret, String ro
leArn, String roleSessionName, String policy) throws ClientException {
       try {
           //构造default profile (参数留空,无需添加Region ID)
           /*
           说明:当设置SysEndpoint为sts.aliyuncs.com时, regionId可填可不填;反之, regionId必填,根
据使用的服务区域填写,例如: cn-shanghai
           详情参考STS各地域的Endpoint,请参见接入地址。
            */
           IClientProfile profile = DefaultProfile.getProfile("", accessKeyId, accessKeySecre
t);
           //用profile构造client
           DefaultAcsClient client = new DefaultAcsClient(profile);
           // 创建一个 AssumeRoleRequest 并设置请求参数
           final AssumeRoleRequest request = new AssumeRoleRequest();
           request.setSysEndpoint("sts.aliyuncs.com");
           request.setSysMethod(MethodType.POST);
           request.setRoleArn(roleArn);
           request.setRoleSessionName(roleSessionName);
           request.setPolicy(policy);
           // 发起请求,并得到response
           final AssumeRoleResponse response = client.getAcsResponse(request);
           return response;
       } catch (ClientException e) {
           throw e;
       }
    }
    static void createUploadVideo(String accessKeyId, String accessKeySecret, String token) {
        // 点播服务所在的Region, 接入服务中心为上海, 则填cn-shanghai
       String regionId = "cn-shanghai";
       IClientProfile profile = DefaultProfile.getProfile(regionId, accessKeyId, accessKeySec
ret);
       DefaultAcsClient client = new DefaultAcsClient(profile);
       CreateUploadVideoRequest request = new CreateUploadVideoRequest();
       request.setSecurityToken(token);
        remuest setTitle("+5").
```

```
request.seritite( to ),
        request.setFileName("D:\\TestVideo\\t4.mp4");
        request.setFileSize(10240L);
        try {
            CreateUploadVideoResponse response = client.getAcsResponse(request);
            System.out.println("CreateUploadVideoRequest, " + request.getUrl());
            System.out.println("CreateUploadVideoRequest, requestId:" + response.getRequestId(
));
            System.out.println("UploadAddress, " + response.getUploadAddress());
            System.out.println("UploadAuth, " + response.getUploadAuth());
            System.out.println("VideoId, " + response.getVideoId());
        } catch (ClientException e) {
            System.out.println("action, error:" + e);
            e.printStackTrace();
        }
    }
}
```

- PHP示例
- Python示例
- .NET示例
- Node.js示例

请求发出后,会返回结果,更多信息,请参见返回结果。

3.4. 基于RAM Policy实现自定义授权

自定义授权可以满足用户个性化的授权策略需求,RAM控制台提供了可视化配置和脚本配置两种自定义授权配置 方式。本文为您介绍自定义授权的使用场景、操作步骤及示例。

使用场景

由于系统策略的授权粒度比较粗,很多时候并不能满足您的细粒度权限控制的需求,此时可以基于RAM Policy实现自定义授权。

使用前须知

当采用脚本配置方式自定义授权时,需要先了解权限策略的基本元素和语法结构,更多信息,请参见权限策略基本 元素和权限策略语法和结构。

以点播系统策略 AliyunVODPlayAuth 为例:

开发指南·账号和授权

```
{
    "Version": "1",
    "Statement": [
        {
            "Action": [
               "vod:GetPlayInfo",
               "vod:GetVideoPlayAuth",
               "vod:GetVideoPlayInfo",
               "vod:GetVideoInfo"
              ],
              "Resource": "*",
              "Effect": "Allow"
        }
    ]
}
```

参数解释如下:

参数	说明
Version	Version定义了Policy的版本,可以固定设置为1。
Statement	通过Statement描述授权语义,其中可以根据业务场景包含多条语义,每条包含对Action、 Effect、Resource 和 Condition的描述。每次请求,系统会逐条依次匹配检查,所有匹配成 功的Statement会根据Effect的设置不同分为通过(Allow)、禁止(Deny),其中禁止 (Deny)的优先。如果匹配成功的都为通过,该条请求即鉴权通过。如果匹配成功有一条禁 止,或者没有任何条目匹配成功,该条请求被禁止访问。
Action	点播支持的Action操作与API——对应,格式为 vod:API名称 ,如上面示例的播放操 作: vod:GetPlayInfo ,注意多个使用英文逗号分隔。通过指定授权的Action列表,就 能组合出想要的权限分组。 所有可用操作,请参见API概览。
Resource	Resource指代的是VOD上面的某个具体的资源或者某些资源(支持通配符*),Resource的 规则是 acs:vod:{region}:* 。由于点播现在没有进一步区分资源,故一般建议授权点 播时Resource填 * 。Resource也是一个列表,可以有多个Resource。其中的region字段 暂不支持,请设置为 * 。
Condition	可选。Condition代表Policy授权的一些条件,对访问来源等进行限制。详细请参见下表。

Condition	说明
acs:Sourcelp	指定IP地址或网段,取值为普通的IP,支持通配符*。
acs:SecureTransport	是否是HTTPS协议,取值:true或false。
acs:MFAPresent	用户登录时是否使用了多因素认证,取值:true或false。
acs:CurrentTime	指定合法的访问时间(云端接收到请求的时间),为ISO8601格式,例如:2012-11- 11T23:59:59Z。

下文示例表示只允许IP来源为 192.168.xx.xx 的请求者访问播放接口:

```
{
   "Version": "1",
   "Statement": [
      {
           "Action": [
               "vod:GetPlayInfo",
               "vod:GetVideoPlayAuth",
               "vod:GetVideoPlayInfo",
               "vod:GetVideoInfo"
           ],
           "Resource": "*",
           "Effect": "Allow",
           "Condition":
            {
               "IpAddress":
               {
                  "acs:SourceIp": "192.168.XX.XX"
                }
             }
      }
  ]
}
```

脚本配置

操作步骤

- 1. 使用阿里云账号登录到。
- 2. 在左侧导航栏,选择权限管理 > 权限策略。
- 3. 在权限策略页面, 单击创建权限策略。
- 4. 在新建自定义权限策略页面, 输入策略名称和备注。
- 5. 选择配置模式为脚本配置。
- 6. 在策略内容处导入系统已有策略或按需根据策略语法设置自定义策略。

以下授权示例以授予媒体审核、云剪辑权限为例,来设置自定义策略。

授权示例

根据API概览,可以得到所有对应的Action,进而分组控制权限。

• 授予使用媒体审核的权限

视频点播

开发指南·账号和授权

{
"Version": "1",
"Statement": [
{
"Action": [
"vod:SetAuditSecurityIp",
"vod:ListAuditSecurityIp",
"vod:CreateAudit",
"vod:GetAuditHistory",
"vod:SubmitAIMediaAuditJob",
"vod:GetAIMediaAuditJob",
"vod:GetMediaAuditResult",
"vod:GetMediaAuditResultDetail",
"vod:GetMediaAuditResultTimeline"
],
"Resource": "*",
"Effect": "Allow"
}
]
}

• 授予使用云剪辑的权限

```
{
   "Version": "1",
   "Statement": [
      {
           "Action": [
               "vod:ProduceEditingProjectVideo",
               "vod:AddEditingProject",
               "vod:UpdateEditingProject",
               "vod:DeleteEditingProject",
               "vod:GetEditingProject",
               "vod:SearchEditingProject",
               "vod:SetEditingProjectMaterials",
               "vod:GetEditingProjectMaterials"
           ],
           "Resource": "*",
           "Effect": "Allow"
      }
  ]
}
```

☑ 注意 以上示例中,该API分组下如果增加了新的接口,则需要更新相应 Action 部分的列表。

可视化配置

⑦ 说明 视频点播暂不支持可视化配置。

相关参考

更多关于自定义授权的内容请参见自定义策略。

3.5. 凭证方式与STS方式对比

上传凭证、播放凭证和STS方式都能解决上传和播放过程中的授权和安全问题,防止被恶意上传和播放。

• 上传凭证

上传凭证是点播服务下发的,上传媒体文件到点播存储的授权凭证,具有时效性、限制访问对象和次数等特征。更多详情,请参见上传地址和凭证。

● 播放凭证

播放凭证是点播服务下发的,授权播放器获取视频播放地址的授权凭证,也具有时效性、限制访问对象和次数 等特征。更多详情,请参见获取播放地址播放。

• STS方式

STS(Security Token Service)是为阿里云账号(或RAM用户)提供短期访问权限管理的云服务。通过STS,可以为第三方访问颁发一个自定义时效和访问权限的访问凭证。第三方用户可以使用STS短期访问凭证直接调用阿里云服务API,或登录阿里云管理控制台操作被授权访问的资源。更多详情,请参见创建角色并进行STS临时授权。

凭证的优势

上传凭证、播放凭证是视频点播推荐使用的上传、播放授权方式,相比STS方式优势如下:

对比项	凭证方式	STS方式
易用性	使用简单,准备好账号AccessKey授予点播权 限即可。	配置较为复杂,角色和授权策略的配置较为繁 琐。
安全性	上传凭证、播放凭证的授权粒度为单视频维 度,且仅能使用一次。	权限粒度较粗,在点播上是API维度,意味着端 上拿到STS授权可无限次上传N个视频或播放该 账号下的所有视频。
灵活性	上传凭证、播放凭证支持更多配置参数,比如 上传时指定消息回调地址、播放时指定域名 等。更多信息,请参见获取音视频上传地址和 凭证和获取音视频播放凭证。	需要等待客户端SDK发布新版迭代,新增功能会 有所滞后。
访问容量	默认分配较大余量,可弹性伸缩,支撑任意用 户的海量个性化授权请求。	作为中心化服务,为所有产品提供授权,有严 格的流控,不适合于高并发场景。

4.媒体上传

4.1. 媒体上传概述

媒体上传的实质是将媒体资源注册到视频点播服务,因此也是使用其它视频点播服务(媒资管理、媒体处理、分发播放等)的前提。本文描述媒体上传需要关注的媒体说明、上传方式、上传相关设置以及上传后的注意事项 等。

媒体说明

视频点播支持视频、音频、图片、辅助媒资(水印、字幕、素材文件)等媒体文件上传。文件可以是本地文件也可以是网络文件(网络文件采用URL拉取上传)。各类型文件的详细格式支持情况请参见下表。

⑦ 说明 上传时需要包含文件扩展名,大小写都支持,如MP4和mp4等同。

媒体类型	支持文件格式	
视频	 MPEG格式: MP4、TS、3GP、MPG、MPEG、MPE、DAT、VOB、ASF等。 AVI格式: AVI。 Windows Media Video格式: WMV、ASF。 Flash Video格式: FLV、F4V。 Real Video格式: RM、RMVB。 QuickTime格式: MOV。 Matroska格式: MKV。 HLS格式: M3U8。 ① 注意 控制台上传场景下, M3U8格式的文件仅支持URL拉取方式上 传, 不支持本地文件上传。 其它格式: DV、GIF、M2T、M4V、MJ2、MJPEG、MTS、OGG、QT、SWF、WEBM。 	
音频	MP3、WMA、WAV、AAC、RA、M4A、FLAC、APE、AC3、AMR、CAF、ACM。	
图片	PNG、JPG、JPEG、GIF。	
辅助媒资	 水印文件: PNG、GIF、APNG、MOV。 字幕文件: SRT、ASS、STL、TTML、VTT。 素材文件: JPG、GIF、PNG、MP4、MAT、ZIP。 	

文件相关限制

- 本地文件:最大支持48.8 TB的单个文件。
- 网络文件:最大支持5 GB的单个文件。
- 控制台上传场景下, M3U8格式的视频文件仅支持URL拉取方式上传, 不支持本地上传方式上传。

存储说明

存储地域

视频点播基于阿里云对象存储OSS(Object Storage Service)提供媒体资源存储服务。视频点播的存储节点遍布 全球,每个存储节点为一个存储地域,每个存储地域由点播系统分配一个存储地址。

存储地址

开通视频点播服务会默认启用华东2(上海)地域的存储地址(点播系统Bucket)。其它地域的存储地址可根据业务需要选择启用或不启用。如果不启用其它地域的存储地址,系统将华东2(上海)的存储地址作为默认存储地址使用。启用多个存储地域的存储地址后,可修改默认存储地址。上传文件时可以指定存储地址或不指定存储地址,不指定时点播系统将选用默认存储地址。

视频点播的存储地域及存储地址启用情况如下。

② 说明 使用视频点播控制台或PC端上传工具上传时根据登录(或选择)的地域使用存储地址。使用SDK 或API上传时,可通过参数(如URL拉取上传接口的 StorageLocation 参数)指定存储地址。

存储地域	存储地域标识	存储地址
华东2(上海)	cn-shanghai	开通视频点播服务时默认启用。如不启 用其它存储地址,本地址为默认存储地 址。
华北2(北京)	cn-beijing	
华南1(深圳)	cn-shenzhen	
日本(东京)	ap-northeast-1	默认不启用。可根据业务需要通过控制
新加坡	ap-southeast-1	后后用。后用多个仔诵地址后,款伙仔 储地址也可根据业务需要重新指定。启 用及修改默认友佬地址的操作指引 清
印度尼西亚(雅加达)	ap-southeast-5	用及修成款队行储地址的珠作指51,请 参见 <mark>存储管理</mark> 。
德国(法兰克福)	eu-central-1	
印度(孟买)	ap-south-1	

存储计费

视频点播存储服务支持按量付费(后付费)和资源包(预付费)两种付费方式。

- 按量付费详情请参见媒资存储计费。
- 资源包详情请参见视频点播存储包。

上传方式

工具上传

为满足部分用户快速上传或在较低门槛的运营场景下上传大文件等需求,视频点播提供界面化的上传工具,包括视频点播控制台和PC端上传工具。两种工具的说明如下:

上传方式	适用场景	使用说明
控制台	快速、便捷地将媒体资源上传到点播的场景。	 开通点播服务后,使用浏览器登录控制台即 可进行上传。 针对音视频媒体,控制台支持本地文件上 传,也支持URL拉取上传。

上传方式	适用场景	使用说明
	大文件、长时间上传,以及低门槛的运营场 景。	 需要下载并安装点播客户端工具,目前支持 Windows、macOS客户端。
PC端上传工具	⑦ 说明 控制台由于安全策略,长时间 上传可能会登录失效导致大文件上传失 败,PC客户端工具能有效避免此问题。	 仅支持本地文件上传。 macOS客户端仅支持上传到华东2(上海)地域。如需上传到其它地域请使用Windows端工具。

开发上传

为满足开发者用户的上传需求,视频点播提供SDK或API的上传方式,其基础流程为:

- 1. 在视频点播服务内获取上传凭证和上传地址。
- 2. 调用OSS上传能力上传媒体。

目前支持的开发者上传方式包括:

- 上传SDK(含服务端与客户端):上传SDK封装了获取上传凭证及地址的逻辑和OSS上传的逻辑。开发者仅需要 简单配置就可以实现上传功能,推荐使用。
- URL批量拉取上传:视频点播提供一个封装好的服务(URL批量拉取上传接口),帮助开发者将媒体文件通过公网访问的URL地址拉取并上传到视频点播的存储地址。这种方式可以免去先将文件下载到本地服务器或终端再使用上传SDK上传到点播的过程。
- 基于OSS原生SDK:当上传SDK无法满足业务需求(如开发语言不匹配)时,开发者可以结合点播服务端 SDK(获取上传凭证及地址)和OSS原生SDK(Base64解码获取到的上传凭证及地址,然后调用OSS接口)完成 上传。
- 基于OSS服务端API(不推荐):开发者也可以直接调用点播服务端接口获取上传地址和凭证,再调用OSS原生 API完成上传。考虑到上传效率和出错概率等因素,一般不推荐这种方式。

⑦ 说明 关于如何选择上传方式,请参见如何选择上传方式。

各开发者	上传工具的说明如下	::
------	-----------	----

上传方式	适用场景	使用说明
上传SDK(服务端)	自动化上传、大批量迁移视频上传等场景。网络媒体文件上传。	 服务端上传SDK封装了视频点播内获取上传 传证及地址的逻辑和OSS的上传能力。仅需
	⑦ 说明 推荐使用上传SDK。如果上传 SDK支持的语言不满足要求,还可以使用 OSS原生SDK完成上传。	 ● 目前支持的开发语言包括: Java、Python、 PHP、C/C++。
上传SDK(客户端)	UGC(用户生产内容)、PGC(专业生产内容) 等场景。	 客户端上传SDK封装了OSS的上传逻辑。 需要用户自行获取上传地址和凭证,推荐在应用服务器集成点播服务端SDK获取。获取后无需解码,直接下发给客户端即可。 目前支持的客户端包括:Android端、iOS端、Web端(JavaScript)、微信小程序。

上传方式	适用场景	使用说明
URL批量拉取上传	文件没有存储在本地服务器或终端,需要通 过公网访问的URL地址上传的场景。	 仅支持上传到华东2(上海)和新加坡地域。 不适用于本地文件上传。
基于OSS原生SDK上 传	 自动化上传、大批量迁移视频上传等场景。 网络媒体文件上传。 	 使用基于OSS原生SDK方式需要开发者用户 自行实现所有上传逻辑,包括在点播服务获 取上传地址和凭证、Base64解码上传凭证和 地址和调用OSS能力完成上传。 目前支持的开发语言包括:Java、Python、 PHP、.NET、Go、Node.js、C/C++。 ⑦ 说明 Java、Python、PHP、 C/C++语言推荐使用点播上传SDK。
基于OSS API上传 不推荐	 自动化上传、大批量迁移视频上传等场景。 网络媒体文件上传。	 调用视频点播相关服务端API获取上传地址和 凭证。 Base64解码上传凭证和地址后再使用OSS能 力上传文件。

上传设置

视频点播支持多种上传时的功能设置,如媒资管理、音视频转码、上传事件通知、上传控制等。

媒资管理(上传时)

- 设置媒体标题:上传时设置音视频、图片的标题。
- 指定媒体分类: 创建好分类后, 上传时指定媒体文件属于某个分类。
- 设置媒体标签:上传时设置媒体标签。
- 设置视频封面: 上传视频时附带一张图片作为该视频的封面。

各上传方式支持上传时媒资管理的详情如下:

上传方式	设置媒体标题	指定媒体分类	设置媒体标签	设置视频封面
控制台	支持	支持	不支持	不支持
PC端上传工具	支持	支持	支持	不支持
上传SDK(服务端)	支持	支持	支持	支持
上传SDK(客户端)	支持	支持	支持	支持
URL批量拉取上传	支持	支持	支持	支持
基于OSS原生SDK上 传	支持	支持	支持	支持

上传方式	设置媒体标题	指定媒体分类	设置媒体标签	设置视频封面
基于OSS API上传	支持	支持	支持	支持

音视频转码

上传音视频文件时设置转码(通过指定转码模板组或指定工作流),上传完成后自动触发转码。也可以设置不转码,但需要注意的是目前仅以下格式支持不转码直接播放:MP4、FLV、M3U8、MP3、WEBM。

- 指定转码模板组转码:指定转码模板组是点播的默认处理方式,上传时选择转码模板组处理方式。上传完成后 点播系统会自动触发转码。
- 指定工作流转码:创建好含转码任务的工作流后,上传时选择工作流处理方式。上传完成后,点播系统会自动执行指定的工作流完成转码任务。
- 不转码: 如果上传时不需要转码, 可通过选择转码模板组中的不转码模板实现。

各上传方式支持上传时转码的详情如下:

上传方式	转码模板组	工作流	不转码
控制台	支持	支持	支持
PC端上传工具	支持	不支持	支持
上传SDK(服务端)	支持	支持	支持
上传SDK(客户端)	支持	支持	支持
URL批量拉取上传	支持	支持	支持
基于OSS原生SDK上传	支持	支持	支持
基于OSS API上传	支持	支持	支持

上传事件通知

在媒体文件上传完成后,可通过事先配置的回调地址及时获取处理进展和状态。视频点播支持HTTP回调(兼容 HTTPS)、MNS回调两种方式获取事件通知。了解回调配置请参见概述。完成回调配置请参见回调设置。所有上传 方式均支持上传事件通知。上传相关的事件如下表所示。

? 说明	关于每个事件的	的内容详情,请参见 <mark>事件详</mark>	情。
事件名称		事件类型	说明

事件名称	事件类型	说明
视频上传完成	FileUploadComplete	点播服务端接收完上传的音视频文件后, 会产生 FileUploadComplete事件。 ⑦ 说明 对于不转码的音视频(或音频)接收到视频上传完成事件通知后方可获取播放地址播放。 如果上传时设置了不转码,接收到视频上传完成事件通知后,还可再调用提交媒体转码作业接口转码。
图片上传完成	ImageUploadComplete	点播服务端接收完上传的图片文件上传后,会产生 ImageUploadComplete事件。
辅助媒资上传完成	AttachedMediaUploadCo mplete	点播服务端接收完上传的辅助媒资文件后,会产生 AttachedMediaUploadComplete事件。
URL上传视频完成	UploadByURLComplete	调用 <mark>URL批量拉取上传</mark> 任务后,云端拉取视频上传完成会产 生UploadByURLComplete事件。
单个清晰度转码完成	StreamTranscodeComple te	视频的某个清晰度、某种格式的流(如:标清的MP4格式)转码完成时会产生此事件。 ⑦ 说明 对于转码的音视频,接收到单个清晰度转 码完成事件通知后方可获取播放地址播放。
视频转码完成	TranscodeComplete	视频所有流转码完成会产生TranscodeComplete事件。 ⑦ 说明 若要确保播放前所有清晰度都能获取到, 可接收到视频转码完成事件通知后再处理。

上传控制

 分片上传:将待上传的文件分成多个碎片(Part)分别上传,上传完成之后将这些Part组合成一个Object。视频 点播上传功能默认采用分片上传的方式上传文件。

⑦ 说明 使用分片上传时,若上传失败则可能会产生碎片文件,一般7天后会自动清除。开发者用户还可 以调用删除上传中的碎片文件接口手动清除碎片文件。

- 获取上传进度或状态。
- 断点续传:如果上传过程中出现网络错误导致上传失败,视频点播支持从最后一块成功上传的碎片(Part)开始继续上传,从而实现断点续传效果。

⑦ **说明** 上传SDK(客户端)内部会完成断点续传的功能,用户不需要关心任何断点和片段恢复,因为异常情况导致的视频上传失败重新上传都会从断点处开始上传。

各上传方式支持的上传控制详情如下:

上传方式	获取上传进度	断点续传	分片上传
控制台	支持	不支持	默认启用
PC端上传工具	支持 ⑦ 说明 通过正在上传或上传完成页 面查看。	不支持	默认启用
上传SDK(服务端)	支持	仅Java上传SDK支持	默认启用
上传SDK(客户端)	支持 ⑦ 说明 通过调用获取媒体上传详情接 查看。	支持	默认启用
URL批量拉取上传	支持 ⑦ 说明 通过调用获取URL上传信息接 口查看。	不支持	默认启用
基于OSS原生SDK上 传	支持	仅Java语言支持	默认启用
基于OSS API上传	支持	不支持	默认启用

上传授权

在开发上传场景下,通过上传凭证和STS方式都能解决上传中的授权和安全问题,防止媒体文件被恶意上传和播放。

- 两种方式的适用场景请参见凭证方式与STS方式对比。
- 具体使用请参见各开发者上传方式的操作文档。
 - 上传SDK(服务端)
 - 上传SDK (客户端)
 - o URL批量拉取上传
 - 。 基于OSS原生SDK上传

• 基于OSS服务端API上传

上传加速

功能说明

视频点播服务提供上传加速的功能,以提升远距离文件传输和GB、TB级大文件的上传性能。上传加速适用于跨区域(如跨海情况)上传,例如在中国大陆将视频上传到新加坡存储地域的存储地址。

开通方式

提交工单申请开通,需要提供**阿里云账号UID**和需要使用上传加速的存储地址。

计费说明

上传加速是点播的增值功能,按使用情况收费。详细价格请参见阿里云产品定价。

? 说明 只开通不使用不收费。

使用限制及支持的上传方式

上传加速仅限于上传视频。各上传方式的支持情况如下:

上传方式	上传加速
控制台	不支持
PC端上传工具	不支持
上传SDK(服务端)	支持
上传SDK(客户端)	支持
URL批量拉取上传	支持
基于OSS原生SDK上传	支持
基于OSS API上传	支持

上传之后

媒体上传完成后可使用的点播服务及后续分发播放的注意事项如下:

点播服务	描述	相关文档
媒资管理(上 传后)	上传完成后,可使用点播服务管理媒体文件,如视 频源文件、转码后的流文件、图片文件等。	媒资管理

开发指南·媒体上传

点播服务	描述	相关文档		
	点播支持媒体文件上传后的音视频转码、智能审 核、云剪辑、Al处理等。			
媒体处理	⑦ 说明 上传到点播存储的音视频文件会 默认提取Meta信息(如视频分辨率、时长、 码率等),对视频也会默认截取封面图,且这 部分预处理完全免费。	媒体处理		
音视频播放	 用户上传音视频完成后,并不代表音视频已准备就 绪,还需点播服务端确认接收完成,可根据事件通 知来确定上传的音视频何时能播放。 对于不转码的音视频(或音频)接收到视频上传 完成事件通知后即可获取播放地址播放。 对于转码的音视频,接收到单个清晰度转码完 成事件通知后即可获取播放地址播放。若要确保 所有清晰度都能获取到,可接收到视频转码完 成事件通知后再处理。 	 视频上传完成 获取音视频播放地址 单个清晰度转码完成 视频转码完成 		

4.2. 工具上传

4.2.1. 控制台上传

视频点播(VOD)控制台提供音视频、图片和短视频素材上传功能,支持批量操作,便于快速将媒体资源上传存储到VOD。本文提供控制台上传媒体的操作指引。

前提条件

上传文件扩展名需要符合以下规则。

媒资类型	支持格式				
	3GP、ASF、AVI、DAT、DV、FLV、F4V、GIF、M2T、M3U8、M4V、MJ2、 MJPEG、MKV、MOV、MP4、MPE、MPG、MPEG、MTS、OGG、QT、RM、 RMVB、SWF、TS、VOB、WMV、WEBM				
视频	⑦ 说明 控制台场景下, M3U8格式的文件仅支持URL拉取方式上传,不 支持 本地上传 方式上传。				
音频	AAC、AC3、ACM、AMR、APE、CAF、FLAC、M4A、MP3、RA、WAV、WMA				
图片 PNG、JPG、JPEG、GIF、HEIC					
短视频素材	MAT、ZIP				

↓ 注意 媒体文件最大支持上传48.8 TB的单个文件。如果存在不支持的格式或非媒资文件,会自动过滤不支持的文件,正常文件会在控制台上显示等待上传。

背景信息

上传文件需要注意以下相关信息。

媒资类型	说明
视频	• 媒体文件最大支持上传48.8 TB的单个文件。
音频	 上传到视频点播的媒资源文件和处理后的媒资文件按存储容量峰值与存放日期 计费。更多信息请参见计费详情。 未配置加速域名,通过控制台或存储地址播放、下载资源会产生存储流出费 用,不能使用流量包抵扣。更多信息请参见计费详情。
图片	未配置加速域名,通过控制台或存储地址查看、下载资源会产生存储流出费用, 不能使用流量包抵扣。更多信息请参见 <mark>计费详情</mark> 。
短视频素材	素材ICON必须是PNG格式,且大小不超过200 Px*200 Px。

上传前设置

● 选择存储地址

选择当前上传资源的存储地址,当您有多个存储地址时,会选中您的默认存储。配置默认存储地址,具体请参 见存储管理。

- 转码设置
 - 上传前,您可以根据实际需要,对待上传视频进行转码设置,选择相应的转码模板组,可以针对单个视频调整,也可选中多个视频后进行批量设置。
 - 转码模板组中支持配置不同清晰度,并对不同清晰度的分辨率、码率等参数,以及水印等进行自定义设置, 具体请参见普通转码模板设置。
 - 添加视频后优先会选中您设置的默认模板组,如果您是首次上传,请先到转码设置中进行配置检查和设置管理,以便在上传视频时可以按照您的业务需求进行转码策略的选择。
- 分类设置

上传前,您也可以进行媒体文件的标题和分类的修改,其中分类可以针对媒体文件进行设置或选中多个媒体文件进行批量分类设置操作。您也可以等媒体文件上传完成后进入媒体文件的详情页再作修改。配置分类模板, 具体请参见分类管理。

上传音视频

? 说明

- 大文件及批量上传,建议使用PC客户端工具。具体操作,请参见PC端上传。
- 开始上传后,离开上传页面不影响上传任务进行,但不能刷新或关闭浏览器。
- 控制台场景下,目前仅**华东2(上海)**地域支持通过URL拉取方式上传音视频,其他地域暂不支持该方式上传。
- 1. 登录视频点播控制台。
- 2. 在点播控制台左侧导航栏的媒资库区域,单击音/视频。
- 3. 单击上传音/视频。

☰ (-) 阿里云 🗠	工作台 ♥ 歩先2 (上海) ∨		Q 搜索		费用 工单 ICI	·晉案 企业 支持 Aj	∞ ⊑ û '₽	🕐 mak 🌀
点攝控制台	▲							新手引导
概范	音/视频							查看上传任务
媒资库	⑦ 帮助信息 1. 煤化学供表大支持上传48.8 TB的单个学件。帮助文料							
音/视频	 上伊到现现点播的環想預文件和处理后的媒想文件该存储容量逢僅与存放日期计量。计量详備 未配置加速域名,通过控制台或存储地址播放,下载资源会产生存储岛出费用,不能使用涡量包抵扣。 							
图片 短视频素材	上传音/视频 全部时间 > 媒体名称 > 请能入媒体名称 Q							Ŧ G
	_	类型 ♡	分类♡	状态 🖓	来源 🖓	创建时间小	操作	
制作中心	0011582 CD 9115 CD 9115	视频	未分类	✓ 正常	点播上传	2022-03-15 14:22:45	管理 媒体处理 :	更多 •

4. 单击添加音/视频。

☰ (-) 阿里云 🛛 🌣	工作台 🔚 华东2(上海)、	·			
点播控制台	▲ 点播控制台 / 音/视频 / 上传	音/视频			
概览	← 上传音/视线	顷			
媒资库	① 开始上传后, 离开此页]	面不影响上传任务进行,但请不要刷新或关闭浏览器			
育/视频	添加音/视频 清除失时	牧记录 清除成功记录			
图片	音/视频名称	存储地址	格式	大小	分类
短视频素材					没有数据

- 5. 添加上传文件,配置转码,单击开始上传。
 - 。 本地上传

	 URL拉取 选择URL拉取方式 	时,若URL中不包含	3文件扩展名,为提	高拉取成功率, 请填	号扩展名 了解URL批量	拉取上传	
ア储地址:				~			
添加音/视频	用转码模板组处理	里 🗸 不转码		✓ 未分类 ✓			
音/视频名称			格式	大小	分类	转码模板组/工作流模板	操作
		支持3GP、AS MJPEG、MKV、 RMVB、SWF、 ⁻ a	<mark>点击或者</mark> GF、AVI、DAT、I MOV、MP4、M IS、VOB、WMV cm, amr, ape, ca	文件拖拽到这里 文件拖拽到这里 DV、FLV、F4V、G IPE、MPG、MPEG MPEM等视频楷 f, flac, m4a, mp3, r	上传 IF、M2T、M4V、M 、MTS、OGG、QT. 式上传,音频支持a a, wav, wma	U2、 、RM、 ac, ac3,	
						开始上传	取消

○ URL拉取

选择URL拉取方式时,若URL中不包含文件扩展名,为提高拉取成功率,请填写扩展名。具体请参见URL拉取 上传。

添加音/视频						
上传方式	○ 本地上传					
	 URL拉取 选择URL拉取方 	试时,若URL中不包	含文件扩展名,为提高拉取成功率,	请填写扩展名 了解URL批量	量拉取上传	
字储地址:			g 🗸			
处理方式 用]转码模版组处理 、	✓ 不转码	◇ 未分类 ◇			
URL		扩展名	音/视频名称	分类	转码模板组/工作流模机	反 操作
				未分类 🗸	不转码 🛛 🗸	删除
- 添加一条						
						台上传 取消
注意						
○ 上传□	中的视频,通	道过单击 取消	可以暂停上传。			
○ 暂停_	上传或上传失	败的视频,	可以通过重新上传恢复	夏上传,且恢复.	上传的视频将重新	i开始上传。

上传图片

- 1. 登录视频点播控制台。
- 2. 在点播控制台左侧导航栏单击**媒资库 > 图片 > 上传图片**。

☰ (-)阿里云 🖙	工作論 👳 华东2(上海) 🗸	Q 搜索	费用	目 工单 ICP 偏震 企业 支持 App 🕞	û 🗑 🗑 🕷 🗿
点攝控制台	▲ 点攝控制台 / 間片				新手引导
概范	图片				查看上传任务
媒资库 窗/视频	7 帮助结息 1、上传型片支持PNG、IPG、GF、HEIC型片指式。帮助文档 2、未配型加速域条、通过控制组成存储加出重要、下数运序会产生存储流出类用。不能使用流量包延加、计数字并	ñ			
图片	上标間片 全部时间 > 圏片名称 > 資輸入圏片名称 Q				Ŧ G
短视频素材	□ 國片 ▽	分类▽	状态 🖓	创建时间 4	操作
制作中心 视频路镜	о слява	未分类	✓ 正常	2022-03-15 10:36:49	管理 動除

3. 单击添加图片,添加上传文件后,单击开始上传。

添加图片					×
添加图片 分类设	置删除				存储地址: 🗸 🗸
音/视频名称	格式	大小	分类	操作	
			点击或者又	文件拖拽到这里上传	
			支持PNG, J	PG, JPEG, GIF图片格式	
					开始上传 取消

上传短视频素材
- 1. 登录视频点播控制台。
- 2. 在点播控制台左侧导航栏单击媒资库 > 短视频素材 > 上传素材。

☰ (-) 阿里云 ♠	工作会 ② 将东2 (上海) ∨		Q 搜索		费用]	E单 ICP 借案	企业 支持	⊕ App	ΕĻ	A	() î	iāda 👩
点攝控制台	▲ 振瀾控制台 / 垣视频蒸材										新	淨引导
概范	短视频素材											
媒资库	 帮助信息 素材ICON必须是PNG格式,且大小不超过200 Px*200 Px* 											
着/视频 图片	上传素材 全部时间 > 素材名称 > 名称 Q										7	с
短视频素材	□ 素材 ▽	英型公	分类♡	上架状态	状态 🛛	(1) and	时间小	ž	発作			
制作中心	о сужев	未分类	未分类	下桌	✓ 正常	2022	-03-04 15:02:	51 1	「理 上架	删除		

3. 配置上传素材, 单击上传。

点攝控制台 / 短视线	项素材 / 上传短视频素材	
← 上传素	慰材	
存储地址		•
• 资源素材	▲ 上传素材压缩包	
• 素材ICON	+ 上传	
要材复数	pngnasu, 入小小 <u>sasu</u> 2007200PX 法位入	
2010 million	若未輸入,则自动获取素材资源文件名作为默认名称	
* 类型	请选择	·
分类	请选择	,
标签		
描述	支持中、英文及下划线,输入运号、分号,或按回车键完成单个标签输入 请输入	
上传取消	29 29	

⑦ 说明 素材ICON必须是PNG格式,且图像分辨率不超过200×200像素。

4.2.2. PC端上传

本文为您介绍PC端上传的使用场景、准备工作、使用限制、上传等内容。

使用场景

大文件,多批量的上传场景,建议使用PC客户端工具进行上传。

使用限制

- 最大支持上传48.8 TB的单个文件。
- macOS客户端仅支持上海地域,如果要上传到其他地域,请使用Windows客户端。

准备工作

- 您已经创建了RAM用户。
 创建RAM用户请参见创建RAM用户。
- 您已经为RAM用户授予了视频点播相关权限。

为RAM用户授权请参见为RAM用户授权。

 登录PC端上传工具需要使用AccessKey完成身份验证,请提前获取AccessKey。获取方法请参见获取 AccessKey。

⑦ 说明 建议在PC端通过RAM用户的AccessKey ID和AccessKey Secret 登录操作。使用RAM用户登录是为了进行权限隔离,降低在控制台对配置类功能的误操作风险。

(可选)如需对资源进行分类或对媒体资源进行转码操作,请先登录视频点播控制台添加分类、转码模板组。
 具体操作,请参见分类管理、普通转码模板设置。

Windows上传流程

1. 下载并安装PC客户端工具。

Windows下载地址: V1.1.0

○ 注意 当打开Windows版本,系统提示找不到MSVCP140.dll或VCRUNTIME140.dll时,请下载并安 装微软2017运行时库。

- 2. 使用RAM用户的AccessKey ID和AccessKey Secret登录PC客户端。
- 3. 在PC客户端顶部菜单栏单击添加上传文件,添加需要上传的音视频。
- 4. 选中上传的音视频,按需设置资源名称、分类、转码模板组、标签、资源描述等内容。

⑦ 说明 如果分类、转码模板组下拉列表中没有内容,请先在视频点播控制台设置后再进行上传操作。具体操作,请参见分类管理、普通转码模板设置。

5. 选中需要上传的音视频,单击开始上传。

开始上传后,可以通过左侧导航栏**正在上传**查看上传进度,上传完成后可以通过左侧导航栏**上传完成**查看音 视频资源信息。

macOS上传流程

1. 下载并安装PC客户端工具。

macOS下载地址: V1.3.2

- 2. 使用RAM用户的AccessKey ID和AccessKey Secret登录PC客户端。
- 3. 在PC客户端左侧导航栏选择视频。
- 4. 单击顶部菜单栏上传,选中需要上传的音视频并打开。
- 5. 在PC客户端左侧导航栏选择传输管理 > 上传编目。
- 6. 选中需要上传的音视频,单击开始上传。

执行结果

使用PC客户端上传文件后,文件会在视频点播控制台显示。

4.3. 开发上传

4.3.1. 如何选择上传方式

开发者的类型、上传的业务场景等都会影响上传方式的选择。本文对视频点播提供的开发者上传方式按照开发类型进行简单分类,描述不同开发语言和客户端类型下支持的上传方式和推荐的上传方式。目的是帮助开发者用户迅速找到适合的上传方式和参考文档。

服务端开发

在服务端,视频点播提供上传SDK,同时也支持OSS原生SDK上传。服务端SDK封装了获取上传地址和凭证的逻辑,也封装了OSS的上传能力,仅需简单配置即可上传,推荐使用。服务端语言支持不足时可选择OSS原生SDK上传。

开发语言	支持的上传方式	推荐的上传方式	文档链接
Java	点播服务端上传SDK OSS原生SDK	点播服务端上传SDK	
Python	点播服务端上传SDK OSS原生SDK	点播服务端上传SDK	请参见上传SDK(服务
C/C++	点播服务端上传SDK OSS原生SDK	点播服务端上传SDK	端)。
РНР	点播服务端上传SDK OSS原生SDK	点播服务端上传SDK	
其它语言 Go、Node.js和.NET等	OSS原生SDK	OSS原生SDK	基于OSS原生SDK上传

客户端开发

视频点播提供封装好OSS的上传能力的客户端上传SDK。请根据使用的客户端选择文档参考:

- Web端上传SDK
- Android端上传SDK
- iOS端上传SDK
- 微信小程序上传SDK

其它场景

- URL批量拉取上传:视频点播提供一个封装好的服务,帮助开发者将媒体文件通过公网访问的URL地址拉取并上 传到视频点播的存储地址。这种方式可以免去先将文件下载到本地服务器或终端再使用上传SDK上传到点播的过 程。详情请参见URL批量拉取上传。
- 基于OSS API上传(不推荐):开发者也可以直接调用点播服务端接口获取上传地址和凭证,再调用OSS原生API 完成上传。考虑到上传效率和出错概率等因素,一般不推荐这种方式。详情请参见基于OSS API上传。

4.3.2. 授权访问点播服务

4.3.2.1. 上传地址和凭证

获取上传地址和凭证的环节为点播服务的核心基础。本文介绍上传地址和凭证的工作原理、使用说明、使用限制、获取方式以及解析方式。

技术原理

上传地址是上传媒体文件到点播存储的文件地址,上传凭证是完成上传动作的授权凭证。上传地址和上传凭证均 由点播服务下发,主要解决媒体上传过程中的授权和安全问题,防止恶意上传。同时,点播服务在下发上传地址 和凭证时还会自动创建媒资信息,即媒体ID(Mediald),用于媒资生命周期管理或媒体处理。

媒体ID

媒体ID在某些场合也叫视频ID(Videold)或图片ID(Imageld),可用于以下场景:

- 上传后追踪和管理媒体的生命周期。利用上传地址和凭证完成上传后,媒体的初始状态一般为上传中。相应的 处理完成后会自动更新为下一个状态,如上传完成、转码中、正常等。
- 上传后指定媒体(用媒体ID识别)发起转码、截图、视频AI处理及视频剪辑等操作。

使用限制

- 上传地址和凭证只能用于单个媒体(单个音频、单个视频或单个图片),不能多个混用,否则会互相覆盖。
- 同一个视频, 重复请求会获取到不同的上传地址和凭证。
- 上传地址(目录)由点播服务自动分配,暂不支持自由指定。
- 上传凭证有效期统一为3000秒(50分钟)。
- 由于视频文件可能较大、上传耗时较长,凭证失效后需要刷新上传凭证,刷新后新的凭证有效期仍是3000秒, 且上传地址不变。
- 图片、辅助媒资(水印、字幕文件等)文件较小,凭证失效后只能重新获取。

使用说明

上传视频、图片和辅助媒资需要调用不同的服务端接口获取对应的上传地址和凭证。在所有面向开发者的上传方 式中, 仅部分方式需要关心如何获取上传地址和上传凭证。详情如下表所示。

? 说明

除了通过上传地址和凭证方式,部分场景还支持通过STS临时Token访问点播服务。两种方式的对比请参见<mark>凭</mark> 证方式与STS方式对比。在上传场景下,两种方式在操作上的区别如下:

- 使用上传凭证在构造上传请求时可直接传入阿里云账号AK或者RAM用户AK。
- 使用STS临时Token在构造上传请求时需要传入提前获取的STS临时Token。更多信息请参见STS临时 Token。

上传方式	是否需要自行获取上传地址和凭 证	是否需要自行解析上传地址和凭 证
------	---------------------	---------------------

上传方式	是否需要自行获取上传地址和凭 证	是否需要自行解析上传地址和凭 证
上传SDK(服务端)	否 ↓ 注意 上传SDK在服务端当前仅支持四种 语言: ↓ Java、Python、C/C++、 PHP。	否
上传SDK(客户端)	是	否
URL拉取上传	否	否
基于OSS原生SDK上传	是 ⑦ 说明 基于OSS原生SDK上传的实质是集 成点播服务端SDK获取上传地址和 凭证,使用获取到的上传地址和 凭证初始化OSS SDK完成上传逻 辑。	是
OSS文件上传	是	是

获取上传地址和凭证

视频点播提供两种方式供开发者自行获取上传地址和凭证。

? 说明

具体的代码示例请点击<mark>使用说明</mark>中对应上传方式跳转至操作步骤中查看。

● (推荐)通过视频点播服务端SDK调用API来获取上传地址和凭证。

开发语言	SDK调用示例
Java	Java媒体上传
Python	Python媒体上传
РНР	PHP媒体上传
.NET	.NET媒体上传
Node.js	Node.js媒体上传
Go	Go媒体上传
C/C++	C/C++媒体上传

通过生成HTTP/HTTPS请求获取上传地址和凭证需要自行构造签名,相对复杂。详细信息请参见请求构造指引和获取上传地址和凭证相关接口。

常用接口

- 获取视频上传地址和凭证
- 刷新视频上传凭证
- 获取图片上传地址和凭证
- 获取辅助媒资上传地址和凭证

解析上传地址和凭证

只有基于OSS原生SDK或API实现上传的方式需要关注如何解析上传地址和凭证。其它上传方式不需要解析上传地址和凭证,也不需要关注其内部逻辑。

将点播服务端返回的上传地址(UploadAddress)和上传凭证(UploadAuth)分别进行Base64解码,可得到OSS 的上传地址和授权信息。解析的具体示例请参见基于OSS原生SDK上传视频到点播。

4.3.2.2. STS临时Token

STS(Security Token Service)是为阿里云账号(或RAM用户)提供短期访问权限管理的云服务。除了通过上传 地址和凭证上传外,视频点播的部分开发者上传方式还支持使用STS临时Token访问点播服务。本文介绍STS临时 Token的技术原理、使用说明及获取方式。

技术原理

通过STS为第三方访问颁发一个自定义时效和访问权限的访问凭证(STS临时Token)。第三方用户可以使用STS临时Token直接调用点播服务端API。

使用说明

⑦ 说明 使用STS临时Token方式和使用上传地址和凭证方式各有优势。用户需要根据实际需求选择使用。 两种方式详细的对比请参见凭证方式与STS方式对比。更多关于上传地址和凭证的信息请参见上传地址和凭证。 证。

在上传场景下,使用STS临时Token和使用上传地址和凭证的操作区别是:

- 使用STS临时Token需要在构造上传请求时传入提前获取到的STS临时Token和临时AK对。
- 使用上传地址和凭证在构造上传请求时可直接传入阿里云账号或RAM用户的AK对。

不同的上传方式使用STS临时Token的描述如下:

上传方式	是否支持STS临时Token	获取/使用方式
上传SDK(服务端)	仅Java语言支持	
上传SDK(客户端)	支持	获取STS临时Token的方式
URL批量拉取上传	不涉及	请参见。 STS临时Token在上传场景
基于OSS原生SDK上传	通过OSS原生SDK调用OSS 服务上传时必须使用STS方 式。	下的具体应用请参见各上传 方式对应的操作文档。
基于OSS API上传	不涉及	

获取STS临时Token

为免去自行签名等麻烦,建议集成STS SDK并通过SDK调用AssumeRole来获取STS临时Token。集成STS SDK前必须创建RAM用户,RAM用户角色并为角色授权访问点播服务。

- 1. 创建RAM用户。操作指引请参见创建角色并进行STS临时授权。
- 2. (可选)如需自定义授权,请参见基于RAM Policy实现自定义授权。
- 3. 集成STS SDK并通过SDK调用AssumeRole来获取STS临时Token。操作步骤根据使用的语言不同而不同。

服务端语言	操作指引		
	Java示例		
Java	⑦ 说明 下文提供了Java语言的示例代码详解。		
Python	Python示例		
РНР	PHP示例		
.NET	.NET示例		
Node.js	Node.js示例		
Go	Go示例		

Java示例

Java获取STS临时Token示例代码

```
import com.aliyuncs.DefaultAcsClient;
import com.aliyuncs.exceptions.ClientException;
import com.aliyuncs.http.MethodType;
import com.aliyuncs.profile.DefaultProfile;
import com.aliyuncs.profile.IClientProfile;
import com.aliyuncs.sts.model.v20150401.AssumeRoleRequest;
import com.aliyuncs.sts.model.v20150401.AssumeRoleResponse;
/**
* @author jack
* @date 2020/5/25
*/
public class TestStsService {
   public static void main(String[] args) {
        // 只有RAM用户(子账号)才能调用 AssumeRole 接口
       // 阿里云主账号的AccessKeys不能用于发起AssumeRole请求
       // 请首先在RAM控制台创建一个RAM用户,并为这个用户创建AccessKeys
       String accessKeyId = "LTAI5tKtf6vKccbinQu****";
       String accessKeySecret = "D4711yBPgjdqe3JzVASSF9yrje****";
       // AssumeRole API 请求参数: RoleArn, RoleSessionName, Policy, and DurationSeconds
        // RoleArn 需要在 RAM 控制台上获取
       String roleArn = "acs:ram::1748098430911242:role/vodrole";
        // RoleSessionName 角色会话名称,自定义参数
       String roleSessionName = "session-name";// 自定义即可
       // 定制你的policy
       String policy = "{n" +
               " \"Version\": \"1\",\n" +
               " \"Statement\": [\n" +
               ...
                   {\n" +
               ...
                      \"Action\": \"vod:*\",\n" +
               ...
                      \"Resource\": \"*\",\n" +
                     \"Effect\": \"Allow\"\n" +
               .....
                   }\n" +
               " ]\n" +
               "}";
        try {
           AssumeRoleResponse response = assumeRole(accessKeyId, accessKeySecret, roleArn, role
SessionName, policy);
           System.out.println("Expiration: " + response.getCredentials().getExpiration());
           System.out.println("Access Key Id: " + response.getCredentials().getAccessKeyId());
           System.out.println("Access Key Secret: " + response.getCredentials().getAccessKeySec
ret();
           System.out.println("Security Token: " + response.getCredentials().getSecurityToken()
);
           System.out.println("RequestId: " + response.getRequestId());
           createUploadVideo(response.getCredentials().getAccessKeyId(), response.getCredential
s().getAccessKeySecret(), response.getCredentials().getSecurityToken());
       } catch (ClientException e) {
           System.out.println("Failed to get a token.");
           System.out.println("Error code: " + e.getErrCode());
           System.out.println("Error message: " + e.getErrMsg());
        }
    }
    static AssumeRoleResponse assumeRole(String accessKeyId, String accessKeySecret, String role
```

```
Arn, String roleSessionName, String policy) throws ClientException {
        try {
           //构造default profile (参数留空,无需添加Region ID)
           /*
           说明:当设置SysEndpoint为sts.aliyuncs.com时, regionId可填可不填;反之, regionId必填,根据
使用的服务区域填写,例如: cn-shanghai
           详情参考STS各地域的Endpoint,请参见接入地址。
            */
           IClientProfile profile = DefaultProfile.getProfile("", accessKeyId, accessKeySecret)
;
           //用profile构造client
           DefaultAcsClient client = new DefaultAcsClient(profile);
           // 创建一个 AssumeRoleRequest 并设置请求参数
           final AssumeRoleRequest request = new AssumeRoleRequest();
           request.setSysEndpoint("sts.aliyuncs.com");
           request.setSysMethod(MethodType.POST);
           request.setRoleArn(roleArn);
           request.setRoleSessionName(roleSessionName);
           request.setPolicy(policy);
           // 发起请求,并得到response
           final AssumeRoleResponse response = client.getAcsResponse(request);
           return response;
       } catch (ClientException e) {
           throw e;
       }
    }
```

参数说明

参数	描述		
RoleArn	需要扮演的角色ID。为RAM用户创建好角色后,角色的ID可以在RAM控制台的 角色 管理 > 基本信息中获取。		
RoleSessionName	角色会话名称。该参数为用户自定义参数。通常设置为调用该API的用户身份,例如:用户名。在操作审计日志中,即使是同一个RAM角色执行的操作,也可以根据不同的RoleSessionName来区分实际操作者,以实现用户级别的访问审计。长度为2~64个字符,可包含英文字母、数字、半角句号(.)、at(@)、短划线 (-)和下划线(_)。		
Policy	在扮演角色的时候额外加上的一个权限限制。 ⑦ 说明 Policy是用来限制扮演角色之后的临时凭证的权限。最后临时凭证获 得的权限是角色的权限和这里传入的Policy的交集。 在扮演角色的时候传入Policy的原因是为了灵活性,比如只能现在使 用CreateUploadVideo接口。		
DurationSeconds	临时凭证的有效期,单位为秒,最小为900,最大为3600。		
accessKeyld和accessKeySecret 需要扮演角色的RAM用户,及其AccessKey Secret。			

4.3.3. 基于上传SDK上传

本文介绍视频点播上传SDK的基础信息,包括SDK概况以及SDK使用参考等。

什么是上传SDK

上传SDK提供上传媒体文件到点播存储的开发工具包,是视频点播端到端服务的重要一环。通过上传SDK,您可以快捷上传视频、音频、图片、字幕等各种媒体文件到点播存储。上传SDK同时提供服务端和移动客户端等多种版本SDK,全面适配各个主流平台和运行环境。

端侧	支持的语言/平台
服务端上传SDK	 Java Python PHP C/C++
客户端上传SDK	 Web端 Android端 iOS端 微信小程序

上传SDK使用说明

各上传SDK的操作指引请参见下表:

端侧	上传流程	操作指引
服务端上传SDK	服务端上传SDK指引	 Java上传SDK Python上传SDK PHP上传SDK C/C++上传SDK
客户端上传SDK	客户端上传SDK指引	 Web端上传SDK Android端上传SDK iOS端上传SDK 微信小程序上传SDK

4.3.4. 基于OSS原生SDK上传

在服务端上传场景下,当推荐的上传方式(如<mark>上传SDK</mark>)无法满足业务需求时,您还可以基于OSS原生SDK上传媒体。本文提供基于OSS原生SDK上传媒体的流程指引、操作步骤和代码示例。

使用场景

视频点播面向开发者提供了丰富的上传方式,其中上传SDK(分服务端和客户端)封装了上传相关的基础逻辑,仅 需简单配置即可实现上传功能,推荐使用。但是服务端上传SDK仅支持Java、Python、C/C++和PHP四种语言。如 果有超出上述语言范围的开发需求(如使用 .NET 语言、Go语言),可以直接基于OSS原生SDK上传。更多上传 方式的介绍请参见媒体上传概述。

不同服务端语言支持的上传方式和推荐的上传方式如下表所示:

开发语言	支持的上传方式	推荐的上传方式	备注
Java	点播服务端上传SDK OSS原生SDK ⑦ 说明 本文提供上传示例代 码。	点播服务端上传SDK	
Python	点播服务端上传SDK OSS原生SDK	点播服务端上传SDK	请参见上传SDK(服务
C/C++	点播服务端上传SDK OSS原生SDK	点播服务端上传SDK	端)。
РНР	点播服务端上传SDK OSS原生SDK ⑦ 说明 本文提供上传示例代 码。	点播服务端上传SDK	
Go	OSS原生SDK ⑦ 说明 本文提供上传示例代 码。	-	需集成点播服务端SDK获取 长传地址和凭证。
.NET	OSS原生SDK ⑦ 说明 本文提供上传示例代 码。	-	本文提供上传示例代码。
Node.js	OSS原生SDK	-	需集成点播服务端SDK获取 长传地址和凭证。 暂无完整的上传示例代码, 仅提供操作参考。

上传流程

基于OSS原生SDK上传需要开发者自行实现所有上传逻辑,包括在点播服务获取上传地址和凭证、Base64解码上传地址和凭证以及调用OSS能力完成上传。

以集成点播服务端SDK获取上传地址和凭证为例,完整的上传流程如下图所示:

上传应用]服务器 视频	页点播	OSS
⚠	点擴服券 端SDK OSS SDK		^
	1. 初始化点播客户端		
	2.获取上传地址和上传凭证	-	
	3.返回上传地址和上传凭证		
	4.解析上传地址和 析后上传凭证和1	┘ 扣上传凭证,并使用解 地址初始化OSS客户端 ─────	
	5.7	开始上传	
	6.返回	! 副上传结果	

流程详解

- 1. 上传应用服务器使用RAM用户AK方式(AccessKey ID和AccessKey Secret)或STS临时AK初始化点播客户端。
- 2. 上传应用服务器使用点播服务端SDK调用上传地址和凭证相关接口获取上传地址、上传凭证及媒资信息。
 - ⑦ 说明 您也可以构造HTTP/HTTPS请求获取上传地址和凭证,但需要自行构造签名,出错概率较高。上传地址和凭证相关接口包括:
 - 获取音视频上传地址和凭证
 - 刷新视频上传凭证
 - 。 获取图片上传地址和凭证
 - 。 获取辅助媒资上传地址和凭证
- 3. 点播服务在请求结果中返回Base64加密的上传地址(UploadAddress)、上传凭证(UploadAuth)、和媒体 ID等信息。上传应用服务器可根据返回结果解析参数作为初始化OSS原生SDK的入参。

? 说明

关于解析后的上传地址和凭证说明请参见参考信息:解析上传地址和凭证说明。

点播服务在下发上传地址和凭证时还会自动创建媒资信息,即媒体ID(Mediald),用于媒资生命周期管理或媒体处理。

- 获取视频上传地址和凭证返回的 VideoⅠd 即媒体ID。
- 获取图片上传地址和凭证返回的 ImageId 即媒体Ⅳ。
- 获取辅助媒资上传地址和凭证返回的 MediaId 即媒体ⅠD。
- 请妥善保存媒体ID,作为媒资管理、音视频播放、媒体处理等的输入。
- 4. 上传应用服务器使用解析后的上传地址(UploadAddress)和授权信息(UploadAuth),通过STS方式初始 化OSS客户端。

↓ 注意 为避免AK信息泄露带来的安全风险,初始化OSS SDK必须使用STS方式。

- 5. 上传应用服务器使用OSS SDK调用OSS上传相关接口将媒体文件上传至指定的存储地址。
- 6. OSS服务返回上传结果。

⑦ 说明 上传结果也可以通过提前配置回调接收上传相关事件来监听。

前提条件

- 您已经开通了视频点播服务。开通步骤请参见开通视频点播服务。
- 您已经完成上传相关的系统配置,包括启用目标存储地域的存储地址和配置回调。操作指引请参见存储管理及回调设置。
- 您已准备好用于调用点播服务的账号。为避免阿里云账号AccessKey泄露带来的安全风险,推荐您创建RAM用户 并授予其VOD相关权限。然后使用RAM用户的AK对(AccessKey ID和AccessKey Secret)访问点播服务。操作 指引请参见创建RAM用户并授权。
- (可选)如需使用STS临时授权方式(阿里云Security Token Service)访问点播服务,请为RAM用户创建角色 并授予角色VOD相关权限。操作指引请参见创建角色并进行STS临时授权。

⑦ 说明 STS临时授权方式的适用场景请参见凭证方式与STS方式对比。

操作步骤

视频点播目前提供PHP、Go、Java和.NET语言的详细操作指引和示例代码。

- PHP上传示例
- Go上传示例
- Java上传示例
- .NET上传示例

注意 真实业务场景下,请参考接口文档将示例中的参数值替换为与业务场景匹配的真实值。获取上传地址和凭证的接口包括:

- 获取音视频上传地址和凭证
- 刷新视频上传凭证
- 获取图片上传地址和凭证
- 获取辅助媒资上传地址和凭证

示例代码均以上传音视频为例(调用点播CreateUploadVideo接口获取上传地址和凭证)。

Demo源码

针对上述示例,视频点播提供了Demo源码供开发者参考。

语言	源码	更多参考信息
PHP上传Demo	PHP版上传Demo源码	OSS-PHP-SDK上传文件
Go上传Demo	GO版上传Demo源码	OSS-Go-SDK上传文件
.NET上传Demo	.NET版上传Demo源码	OSSNET-SDK上传文件

语言	源码	更多参考信息
Java上传Demo	Java版上传Demo源码	OSS-Java-SDK上传文件

PHP上传示例

步骤一、安装点播服务端SDK

操作指引请参见点播PHP SDK安装。

步骤二、安装OSS SDK

↓ 注意 环境要求如下:

- PHP 5.3+,可在终端执行 php -v 命令查看当前的PHP版本。
- CURL扩展,可在终端执行 php -m 命令查看 curl 扩展是否已经安装好。
- 1. 在您的PHP项目中添加空文件夹aliyun-php-sdk。
- 2. 从aliyun-openapi-php-sdk下载整个源码。

解压后拷贝aliyun-php-sdk-core和aliyun-php-sdk-voo两个文件夹到aliyun-php-sdk目录下。

3. 从aliyun-oss-php-sdk下载最新的OSS PHP SDK的源码。

解压ZIP文件后添加文件夹到*aliyun-php-sdk*目录下。以下载V.2.4.3 Source code为例,解压后文件夹为*aliyun -oss-php-sdk-2.4.3*。

步骤三、上传 (示例代码)

? 说明

故障排除方法

• 如果由于SSL配置异常导致错误(如出现cURL error: SSL certificate problem),可将HTTPS替换为HTTP后再初始化OssClient。

```
$uploadAddress['Endpoint'] = str_replace("https:", "http:", $uploadAddress['Endpoint']
);
```

• 如果上传凭证过期(默认3000秒),可以刷新上传凭证再上传。刷新凭证的示例代码如下:

```
function refresh_upload_video($vodClient, $videoId) {
    $request = new vod\RefreshUploadVideoRequest();
    $request->setVideoId($videoId);
    return $vodClient->getAcsResponse($request);
}
```

<?php

```
//1、在代码中引用voD和oss的文件。
require_once './aliyun-php-sdk/aliyun-php-sdk-core/Config.php'; // 假定您的源码文件和aliyun-php-sdk处于同一目录。
require_once './aliyun-php-sdk/aliyun-oss-php-sdk-2.4.3/autoload.php';
use vod\Request\V20170321 as vod;
use OSS\OssClient;
use OSS\Core\OssException;
//2、使用AK对方式初始化voD客户端。
function init_vod_client($accessKeyId, $accessKeySecret) {
    StrationId = log characterity // 提供点班接入服务的方式的口口。
```

```
stegionia = Cn-Shanghai; // 1KI加出面(MD)/ILD)Kegion保可, MUH. 按八版力1工工序, MUHCN-Sh
anghai
   $profile = DefaultProfile::getProfile($regionId, $accessKeyId, $accessKeySecret);
   return new DefaultAcsClient($profile);
}
//3、获取上传地址和凭证
function create upload video($vodClient) {
   $request = new vod\CreateUploadVideoRequest();
   $request->setTitle("视频标题");
                                   // 视频标题(必填参数)
   $request->setFileName("文件名称.mov"); // 视频源文件名称,必须包含扩展名(必填参数)
   $request->setDescription("视频描述"); // 视频源文件描述(可选)
   //CoverURL示例: http://example.alicdn.com/tps/TB1qnJ1PVXXXXXXXXXXXXXXXXX700-****.png
   $request->setCoverURL("<your Cover URL>"); // 自定义视频封面(可选)
   $request->setTags("标签1,标签2"); // 视频标签,多个用逗号分隔(可选)
   return $vodClient->getAcsResponse($request);
}
//4、使用上传凭证和地址通过STS方式初始化OSS客户端。
function init oss client($uploadAuth, $uploadAddress) {
   $ossClient = new OssClient($uploadAuth['AccessKeyId'], $uploadAuth['AccessKeySecret'], $uplo
adAddress['Endpoint'],
       false, $uploadAuth['SecurityToken']);
   $ossClient->setTimeout(86400*7);  // 设置请求超时时间,单位秒,默认是5184000秒, 建议不要设置太小
,如果上传文件很大,消耗的时间会比较长
   $ossClient->setConnectTimeout(10); // 设置连接超时时间,单位秒,默认是10秒
   return $ossClient;
}
//5、上传。
function upload local file($ossClient, $uploadAddress, $localFile) {
   return $cssClient->uploadFile($uploadAddress['Bucket'], $uploadAddress['FileName'], $localFi
le):
}
$accessKeyId = '<Your AccessKey ID>';
                                                   // 您的AccessKeyId
$accessKeySecret = '<Your AccessKey Secret>';
                                                   // 您的AccessKeySecret
$localFile = '/Users/aligame/Downloads/****.mp4'; // 需要上传到VOD的本地视频文件的完整路径
try {
   // 初始化VOD客户端并获取上传地址和凭证
   $vodClient = init vod client($accessKeyId, $accessKeySecret);
   $createRes = create upload video($vodClient);
   // 执行成功会返回VideoId、UploadAddress和UploadAuth。Base64解码UploadAddress和UploadAuth。
   $videoId = $createRes->VideoId;
   $uploadAddress = json_decode(base64_decode($createRes->UploadAddress), true);
   $uploadAuth = json decode(base64 decode($createRes->UploadAuth), true);
   // 使用UploadAuth和UploadAddress初始化OSS客户端
   $ossClient = init oss client($uploadAuth, $uploadAddress);
   // 上传文件,注意是同步上传会阻塞等待,耗时与文件大小和网络上行带宽有关
   $result = upload local file($ossClient, $uploadAddress, $localFile);
   printf("Succeed, VideoId: %s", $videoId);
} catch (Exception $e) {
   // var_dump($e);
   printf("Failed, ErrorMessage: %s", $e->getMessage());
}
```

Go上传示例

步骤一、安装点播服务端SDK

操作指引请参见点播Go SDK安装。

步骤二、安装OSS SDK

↓ 注意 环境要求如下:

支持Go 1.7及以上版本,可到Go官网下载合适的版本安装。

1. 使用 go get 或 glide 安装阿里云 Go SDK。

○ 使用 go get 安装阿里云 Go SDK。

go get -u github.com/aliyun/alibaba-cloud-sdk-go/sdk

- ? 说明
 - 安装过程中, 界面不会打印提示, 请耐心等待。如发生超时, 请再次执行命令。
 - 当GOPATH对应的目录下出现了子目录*src/github.com/aliyun/alibaba-cloud-sdk-go/services* /vod即表示SDK安装成功。

○ 使用glide安装阿里云Go SDK。

glide get github.com/aliyun/alibaba-cloud-sdk-go

2. 安装OSS SDK。操作指引请参见OSS Go SDK安装。

步骤三、上传 (Go示例代码)

```
package main
import (
   "encoding/base64"
   "encoding/json"
   "fmt"
   "github.com/aliyun/alibaba-cloud-sdk-go/sdk"
   "github.com/aliyun/alibaba-cloud-sdk-go/sdk/auth/credentials"
   "github.com/aliyun/alibaba-cloud-sdk-go/services/vod"
   "github.com/aliyun/aliyun-oss-go-sdk/oss"
   "os"
)
func InitVodClient(accessKeyId string, accessKeySecret string) (client *vod.Client, err error) {
   // 点播服务接入区域
   regionId := "cn-shanghai"
   // 创建授权对象
   credential := &credentials.AccessKeyCredential{
       accessKeyId,
       accessKeySecret,
   }
   // 自定义config
   config := sdk.NewConfig()
                              // 失败是否自动重试
   config.AutoRetry = true
   config.MaxRetryTime = 3 // 最大重试次数
   config.Timeout = 300000000 // 连接超时,单位: 纳秒; 默认为3秒
   // 创建vodClient实例
   return vod.NewClientWithOptions(regionId, config, credential)
}
func MyCreateUploadVideo(client *vod.Client) (response *vod.CreateUploadVideoResponse, err error
) {
         st ·= wod CreateCreateIInloadVideoRemuest()
```

```
reduest ·- And orearentearenthroads incoredness ()
   request.Title = "Sample Video Title"
   request.Description = "Sample Description"
   request.FileName = "/opt/video/sample/video file.mp4"
   //request.CateId = "-1"
   //Cover URL示例: http://example.alicdn.com/tps/TB1qnJ1PVXXXXXCXXXXXXXXX-700-****.png
   request.CoverURL = "<your CoverURL>"
   request.Tags = "tag1,tag2"
   request.AcceptFormat = "JSON"
   return client.CreateUploadVideo(request)
}
func InitOssClient(uploadAuthDTO UploadAuthDTO, uploadAddressDTO UploadAddressDTO) (*oss.Client,
error) {
   client, err := oss.New(uploadAddressDTO.Endpoint,
       uploadAuthDTO.AccessKeyId,
       uploadAuthDTO.AccessKeySecret,
       oss.SecurityToken(uploadAuthDTO.SecurityToken),
       oss.Timeout(86400*7, 86400*7))
   return client, err
}
func UploadLocalFile(client *oss.Client, uploadAddressDTO UploadAddressDTO, localFile string) {
    // 获取存储空间。
   bucket, err := client.Bucket(uploadAddressDTO.Bucket)
   if err != nil {
       fmt.Println("Error:", err)
       os.Exit(-1)
    }
   // 上传本地文件。
   err = bucket.PutObjectFromFile(uploadAddressDTO.FileName, localFile)
    if err != nil {
       fmt.Println("Error:", err)
       os.Exit(-1)
   }
}
type UploadAuthDTO struct {
   AccessKeyId string
   AccessKeySecret string
   SecurityToken string
}
type UploadAddressDTO struct {
   Endpoint string
   Bucket string
   FileName string
}
func main() {
   var accessKeyId string = "<Your AccessKeyId>"
                                                         // 您的AccessKeyId
   var accessKeySecret string = "<Your AccessKeySecret>" // 您的AccessKeySecret
   var localFile string = "/opt/video/sample/video file.mp4" // 需要上传到VOD的本地视频文件的完
整路径
   // 初始化VOD客户端并获取上传地址和凭证
   var vodClient, initVodClientErr = InitVodClient(accessKeyId, accessKeySecret)
    if initVodClientErr != nil {
       fmt.Println("Error:", initVodClientErr)
       return
    }
    // 获取上传地址和凭证
   var response, createUploadVideoErr = MyCreateUploadVideo(vodClient)
   if anostollolosdWidooErr l- nil (
```

开发指南·媒体上传

```
II CLEACEOPIOAUVIUEOEII :- IIII (
      fmt.Println("Error:", createUploadVideoErr)
       return
   }
   // 执行成功会返回VideoId、UploadAddress和UploadAuth
   var videoId = response.VideoId
   var uploadAuthDTO UploadAuthDTO
   var uploadAddressDTO UploadAddressDTO
   var uploadAuthDecode, _ = base64.StdEncoding.DecodeString(response.UploadAuth)
   var uploadAddressDecode, _ = base64.StdEncoding.DecodeString(response.UploadAddress)
   json.Unmarshal(uploadAuthDecode, &uploadAuthDTO)
   json.Unmarshal(uploadAddressDecode, &uploadAddressDTO)
   // 使用UploadAuth和UploadAddress初始化OSS客户端
   var ossClient, = InitOssClient(uploadAuthDTO, uploadAddressDTO)
   // 上传文件,注意是同步上传会阻塞等待,耗时与文件大小和网络上行带宽有关
   UploadLocalFile(ossClient, uploadAddressDTO, localFile)
   //MultipartUploadFile(ossClient, uploadAddressDTO, localFile)
   fmt.Println("Succeed, VideoId:", videoId)
}
```

Java上传示例

```
↓ 注意 环境要求如下:
```

- 环境要求使用Java 1.8及以上版本。
- 可在终端执行 java -version 命令查看Java版本。

步骤一、安装点播服务端SDK

操作指引请参见点播Java SDK集成。

步骤二、安装OSS SDK

操作指引请参考OSS Java SDK安装。

步骤三、上传(Java代码示例)

```
import com.alibaba.fastjson.JSONObject;
import com.aliyun.oss.OSSClient;
import com.aliyuncs.DefaultAcsClient;
import com.aliyuncs.exceptions.ClientException;
import com.aliyuncs.profile.DefaultProfile;
import com.aliyuncs.vod.model.v20170321.CreateUploadVideoRequest;
import com.aliyuncs.vod.model.v20170321.CreateUploadVideoResponse;
import org.apache.commons.codec.binary.Base64;
import java.io.File;
/**
* descript
public class UploadDemo {
  public static DefaultAcsClient initVodClient(String accessKeyId, String accessKeySecret) thr
ows ClientException {
        // 根据点播接入服务所在的Region填写,例如: 接入服务在上海,则填cn-shanghai; 其他区域请参见存储说明
       String regionId = "cn-shanghai";
       DefaultProfile profile = DefaultProfile.getProfile(regionId, accessKeyId, accessKeySecre
t.):
```

```
DefaultAcsClient client = new DefaultAcsClient(profile);
        return client;
   public static CreateUploadVideoResponse createUploadVideo(DefaultAcsClient vodClient) throws
ClientException {
       CreateUploadVideoRequest request = new CreateUploadVideoRequest();
       request.setFileName("vod test.mp4");
       request.setTitle("this is title");
        //request.setDescription("this is desc");
        //request.setTags("tag1, tag2");
        //CoverURL示例: http://example.aliyundoc.com/test cover ****.jpg
        //request.setCoverURL("<your CoverURL>");
        //request.setCateId(-1L);
        //request.setTemplateGroupId("");
        //request.setWorkflowId("");
        //request.setStorageLocation("");
        //request.setAppId("app-1000000");
        //设置请求超时时间
       request.setSysReadTimeout(1000);
       request.setSysConnectTimeout(1000);
       return vodClient.getAcsResponse(request);
    }
   public static OSSClient initOssClient(JSONObject uploadAuth, JSONObject uploadAddress) {
       String endpoint = uploadAddress.getString("Endpoint");
       String accessKeyId = uploadAuth.getString("AccessKeyId");
       String accessKeySecret = uploadAuth.getString("AccessKeySecret");
       String securityToken = uploadAuth.getString("SecurityToken");
       return new OSSClient(endpoint, accessKeyId, accessKeySecret, securityToken);
   public static void uploadLocalFile(OSSClient ossClient, JSONObject uploadAddress, String loc
alFile) {
       String bucketName = uploadAddress.getString("Bucket");
       String objectName = uploadAddress.getString("FileName");
       File file = new File(localFile);
       ossClient.putObject(bucketName, objectName, file);
    }
   public static void main(String[] argv) {
        //您的AccessKeyId
       String accessKeyId = "<Your AccessKeyId>";
        //您的AccessKeySecret
        String accessKeySecret = "<Your AccessKeySecret>";
       //需要上传到VOD的本地视频文件的完整路径,需要包含文件扩展名
       String localFile = "/Users/yours/Video/testVideo.flv";
       try {
           // 初始化VOD客户端并获取上传地址和凭证
           DefaultAcsClient vodClient = initVodClient(accessKeyId, accessKeySecret);
           CreateUploadVideoResponse createUploadVideoResponse = createUploadVideo(vodClient);
           // 执行成功会返回VideoId、UploadAddress和UploadAuth
           String videoId = createUploadVideoResponse.getVideoId();
           JSONObject uploadAuth = JSONObject.parseObject(decodeBase64(createUploadVideoRespons
e.getUploadAuth()));
           JSONObject uploadAddress = JSONObject.parseObject(decodeBase64(createUploadVideoResp
onse.getUploadAddress()));
           // 使用UploadAuth和UploadAddress初始化OSS客户端
           OSSClient ossClient = initOssClient(uploadAuth, uploadAddress);
           // 上传文件,注意是同步上传会阻塞等待,耗时与文件大小和网络上行带宽有关
           uploadLocalFile(ossClient.uploadAddress.localFile);
```

```
System.out.println("Put local file succeed, VideoId : " + videoId);
} catch (Exception e) {
System.out.println("Put local file fail, ErrorMessage : " + e.getLocalizedMessage())
;
}
private static String decodeBase64(String data) {
return new String(Base64.decodeBase64(data));
}
```

.NET上传示例

↓ 注意 环境要求如下:

- .NET Framework 4.6.1及其以上版本。
- .NET Standard 2.0及其以上版本。

步骤一、安装点播服务端SDK

操作指引请参见点播.NET SDK集成。

步骤二、安装OSS SDK并初始化

操作指引请参见OSS .NET SDK安装。

步骤三、初始化点播客户端

示例代码

```
public static DefaultAcsClient InitVodClient(string accessKeyId, string accessKeySecret)
{
    // 根据点播接入服务所在的Region填写,例如: 接入服务在上海,则填cn-shanghai
    string regionId = "cn-shanghai";
    IClientProfile profile = DefaultProfile.GetProfile(regionId, accessKeyId, accessKeySecret);
    return new DefaultAcsClient(profile);
}
```

步骤四、完成上传

1. 获取上传地址和凭证(以上传视频为例)。

```
public static CreateUploadVideoResponse CreateUploadVideo(DefaultAcsClient vodClient)
{
   CreateUploadVideoRequest request = new CreateUploadVideoRequest();
   request.AcceptFormat = Aliyun.Acs.Core.Http.FormatType.JSON;
   request.FileName = "vod test.mp4";
   request.Title = "this is title";
   //request.Description = "<文件描述>";
   //request.Tags = "<标签,多个标签用逗号分开>";
   //CoverURL示例: http://example.aliyundoc.com/test cover ****.jpg
   //request.CoverURL = "<视频封面>";
   //媒体分类。登录点播控制台
   //request.CateId = -1;
   //转码模板组ID。登录点播控制台
   //request.TemplateGroupId = "<转码模板组ID>";
   //工作流ID。可登录点播控制台
   //request.WorkflowId = "<工作流ID>";
   //存储地址。登录点播控制台
   //request.StorageLocation = "<存储地址>";
   //AppId为固定取值。
   //request.AppId = "app-1000000";
   //设置请求超时时间
   request.SetReadTimeoutInMilliSeconds(1000);
   request.SetConnectTimeoutInMilliSeconds(1000);
   return vodClient.GetAcsResponse(request);
}
```

- 2. Base64解析获取到的UploadAddress和UploadAuth。
- 3. 使用上传凭证和地址初始化OSS客户端。

↓ 注意

- 需要先Base64解码并JSON Decode再传入。
- 必须使用STS方式初始化OSS客户端。

示例代码

```
public static OssClient InitOssClient(JObject uploadAuth, JObject uploadAddress)
{
    string endpoint = uploadAddress.GetValue("Endpoint").ToString();
    string accessKeyId = uploadAuth.GetValue("AccessKeyId").ToString();
    string securityToken = uploadAuth.GetValue("AccessKeySecret").ToString();
    return new OssClient(endpoint, accessKeyId, accessKeySecret, securityToken);
}
```

4. 上传本地文件。

示例代码

```
public static void UploadLocalFile(OssClient ossClient, JObject uploadAddress, string localF
ile)
{
    string bucketName = uploadAddress.GetValue("Bucket").ToString();
    string objectName = uploadAddress.GetValue("FileName").ToString();
    ossClient.PutObject(bucketName, objectName, localFile);
}
```

5. (可选)如果上传凭证过期,可刷新上传凭证再上传。

示例代码

```
public static RefreshUploadVideoResponse RefreshUploadVideo(DefaultAcsClient vodClient)
{
    RefreshUploadVideoRequest request = new RefreshUploadVideoRequest();
    request.AcceptFormat = Aliyun.Acs.Core.Http.FormatType.JSON;
    request.VideoId = "VideoId";
    //设置请求超时时间
    request.SetReadTimeoutInMilliSeconds(1000);
    request.SetConnectTimeoutInMilliSeconds(1000);
    return vodClient.GetAcsResponse(request);
}
```

6. 执行完整流程(注意捕获异常)。

示例代码

```
//您的AccessKeyId
string accessKeyId = "<Your AccessKeyId>";
//您的AccessKeySecret
string accessKeySecret = "<Your AccessKeySecret>";
//需要上传到VOD的本地视频文件的完整路径,需要包含文件扩展名
string localFile = "/Users/yours/Video/testVideo.flv";
trv {
   // 初始化VOD客户端并获取上传地址和凭证
   DefaultAcsClient vodClient = InitVodClient(accessKeyId, accessKeySecret);
   CreateUploadVideoResponse createUploadVideoResponse = CreateUploadVideo(vodClient);
   // 执行成功会返回VideoId、UploadAddress和UploadAuth
   String videoId = createUploadVideoResponse.VideoId;
   JObject uploadAuth = JObject.Parse(Base64Decode(createUploadVideoResponse.UploadAuth));
   JObject uploadAddress = JObject.Parse(Base64Decode(createUploadVideoResponse.UploadAddre
ss));
   // 使用UploadAuth和UploadAddress初始化OSS客户端
   OssClient ossClient = InitOssClient(uploadAuth, uploadAddress);
   // 上传文件,注意是同步上传会阻塞等待,耗时与文件大小和网络上行带宽有关
   UploadLocalFile(ossClient, uploadAddress, localFile);
   Console.WriteLine("Put local file succeed, VideoId : " + videoId);
}
catch (Exception e)
{
   Console.WriteLine("Put local file fail, ErrorMessage : " + e.Message);
}
```

更多语言操作指引

如果上传SDK和上述上传示例依然不能满足业务需求,请参考下表获取更多语言的操作参考。

⑦ 说明 点播上传SDK已覆盖的语言以及已有完整示例的语言不再赘述。

| 开发语言 | 点播服务端SDK | OSS原生SDK |
|---------|---|---|
| Node.js | 点播Node.js SDK安装 点播Node.js SDK初始
化 点播Node.js SDK获取
上传地址和凭证示例 | 1. OSS Node.js SDK安装
2. OSS Node.js SDK初始化
3. OSS Node.js SDK上传文件 |

参考信息:解析上传地址和凭证说明

Base64解析上传地址(UploadAddress)和上传凭证(UploadAuth),得到OSS的上传地址和授权信息。解析后的上传地址和授权信息可用于初始化OSS客户端。

⑦ 说明 Base64解析为较常用的开发者功能,本文不提供具体的解析工具,仅提供解析后的参数说明。

UploadAddress解析后字段

| 字段 | 描述 |
|---------------|---------------------|
| Bucket | 点播存储地址。 |
| Endpoint | 点播存储地域标识。 |
| FileName | 点播系统为上传文件分配的文件名。 |
| Object Prefix | 当且仅当上传文件是M3U8文件时返回。 |

UploadAuth解析后字段

| 字段 | 描述 |
|-----------------|-----------------------------------|
| AccessKeyld | 上传用户的AccessKey ID。 |
| AccessKeySecret | 上传用户的AccessKey Secret。 |
| SecurityToken | 上传授权安全令牌。 |
| ExpireUT CT ime | 上传地址和凭证过期时间。 |
| Expiration | 上传授权过期时间,视频为3000秒,过期需要刷新上传凭
证。 |
| Region | 上传地域标识。 |

以下是上传地址和凭证解析前后的示例:

UploadAddress及UploadAuth解析前

```
{
"VideoId": "1c1255e7e58d4d7487d0c1852954****",
"UploadAddress": "eyJFbmRwb2ludCl6Imh0dHBzOi8vb3NzLWNuLXNoYW5naGFpLmFsaX11bmNzLmNvbSIsIkJ1Y2tldC
I6InZvZC1zaGFuZ2hhaS1rYXNpIiwiRmlsZU5hbWUi0iJ2b2QtZjFlOGM0L3N2LzFhYmUyZDVjLTE3ZWU3Zjc3ZjA0LzFhYm
UyZDVjLTE3ZWU3Zjc3ZjA0Lm1w****",
"RequestId": "2975D91A-F89F-5EBF-97D5-07028C90****",
"UploadAuth": "eyJTZWN1cml0eVRva2VuIjoiQ0FJUzhRUjFxNkZ0NUIyeWZTaklyNWZNQ3Z6NWg1aFEzWVc2VVhmVDEwN
FpXUGtVaHE3Z3B6ejJJSGhFZjNWdEErOGN0LzQxbFdsVzUvWWFscklxRk1NVkhSMlVNWkFvdjqwT3ExLzdKb0hidk5ldTBic
0hoWnY50EtzbG9waThqcUhvZU96Y11JNzN2cHpQQWdtMldFUVJySkwrY1RLOUpaZ1BVL21nZ29KbWFkSTZSeFN4YVNF0GF2N
nlXTXZYdXV0dGJmamgzU2RjekJKSStheHE3dFZENG15VmtkeUJKRk5UN2g2YUZ1ZkkrNzhjS3hSaU5JazhHckpGcGVUeHNPQ
jhwdTNMbnA3cjlnNUpPdWhaTVYvWVg1dnk1ZGJBQk9LeFA4d2xTSzcvUEgzQnNMYUlMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdoYkxXUIMWUtuamhzZ1puc1hEQXBRWjdNKYY
nJXeUUwVGpiVEpwZXU0a3poWlIreUtkdUN5N3BrK3AxdzExSGw0ZGV2T2x5VFFxMlF5em9sSXBzN1prOUtSeGtOeGo3SWVLZ
2NieWxSZDA5bUg3K1BVYmhyYUJkWHBjR1l1aGJKQkFsaDFXMXlvdm5pWnV2Zm9Md2ljWUg0VjU1ZHMrOGFmNDhVbDJveFUye
itVYTYxOEZoMkVsWm9XckZMeXFMZ0ZJYWk0YnVldW9XcGJQWEJGK29NdHlkcEV3SFlwREhmVG1oTGYyK2pwSTg5YmxiZDRNV
F15cVBNK3BJbEV3STE1NWRCQ2czQWY00WcvMWM2cExDc3FFMmerdUs1WGkyaHV6Tm450VNIOGRZUTV4TTRPYTZ0aWVMSXNtU
FYrUlhQYktSbWxwaUlWalV6QnhHdkoxNEJtYS9PMWlwVzkweGF5ekR0WWt4RHNXNzhqempnSUpjcHRkV1EzVzVCRDZ0cG51R
FZkaXl0NUcwSUM2alN1K1pRQmI0NEk3b1hDS3p0aEY4NGxxWDkwelMwdXB6Y3R5bENTOEw1ZWJJN1o1SnlCQmIrOEJzNXppV
3FXRGtKR29BQmq1Z3M3Y1VOZEs2NGdVaThRM2tOdWpVZFdiZTFTb1Y1cVZJOEdmTkYvZ1q0MUNvWG1rSHhwN3FwOTBpMFBSd
1h5VUc4WEVkVGUxY3RZbGEwZG9WeGhtNThOQVdoR0pEOXQxK21iWjd4ZUFkczNqT3NpSUNPb251b3JWYmNUb0s2VGs3cjNxe
j \texttt{ZHSlgyb} \texttt{E54djVm} \texttt{WTBvUWZ} \texttt{sSTJNQWR0} \texttt{cho5WnJidnZ5} \texttt{eUFjYz0iLCJBY2Nlc3NLZXlJZCI6I1NUUy5} \texttt{OVHlBRk1qR3FqR1FTU}
WIzTkxUdjhpa0tFIiwiRXhwaXJlVVRDVGltZSI6IjIwMjItMDItMTFUMDk6NTc6NDNaIiwiQWNjZXNzS2V5U2VjcmV0Ijoi0
WZGUXQyeVJSWTZGMVBFb0hNS1dMYmFmeHlSd1BNNjFuRUhDbzZDTWlQZCIsIkV4cGlyYXRpb24i0iIzNjAwIiwiUmVnaW9uI
joiY24tc2hhbmdoYWk****"
}
```

UploadAddress解析后

```
{
"Endpoint": "https://*****.aliyuncs.com",
"Bucket": "vod-shanghai-****",
"FileName": "vod-fle8c4/sv/labe2d5c-17ee7f77f04/labe2d5c-17ee7f7****.mp4"
}
```

☑ 注意 如果上传的是M3U8文件, UploadAddress 解析后, 会有 ObjectPrefix 字段值。示例如下:

M3U8文件UploadAddress解析后

```
{
    "Endpoint": "https://*****.aliyuncs.com",
    "Bucket": "vod-shanghai-****",
    "ObjectPrefix": "vod-fle8c4/sv/4b9c68b2-17ee8156****",
    "FileName": "vod-fle8c4/sv/4b9c68b2-17ee81564e7/4b9c68b2-17ee815****.m3u8"
}
```

UploadAuth解析后

{

| "SecurityToken": "CAIS8QR1q6Ft5B2yfSjIr5fMCvz5h5hQ3YW6UXfT104ZWPkUhq7gpzz2IHhEf3VtA+8ct/411W1W5/ |
|--|
| YalrIqFMMVHR2UMZAov800q1/7JoHbvNeu0bsHhZv98Kslopi8jqHoeOzcYI73vpzPAgm2WEQRrJL+cTK9JZfPU/mggoJmad |
| ${\tt I6RxSxaSE8av5d0gplrr34Xxm0Mu22YCb3nk3aDkdjpmgajnhku5y42dG74BjTh0GYsug0vNbYOoCpacVLNpVyWMvXuuttbfrake1000000000000000000000000000000000000$ |
| jh3SdczBJI+axq7tVD4iyVkdyBJFNT7h6aFufI+78cKxRiNIk8GrJFpeTxsOB8pu3Lnp7r9g5JOuhZMV/YX5vy5dbABOKxP8 |
| wlSK7/PH3BsLaILYKnjhsgZnsXDApQZ7hbLWRrWyE0TjbTJpeu4kzhZR+yKduCy7pk+p1w11Hl4devOlyTQq2QyzolIps7Zk |
| $9 \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \$ |
| $\verb"ZoWrFLyqLgFIai4bueuoWpbPXBF+oMtydpEwHYpDHfTmhLf2+jpI89blbd4MTYyqPM+pIlEwI155dBCg3Af49g/1c6pLCsqEWHYpDHfTmhLf2+jpI89blbd4MTYyqPM+pIlEwI155dBCg3Af49g/1c6pLCsqEWHYpDHfTmhLf2+jpI89blbd4MTYyqPM+pIlEwI155dBCg3Af49g/1c6pLCsqEWHYpDHfTmhLf2+jpI89blbd4MTYyqPM+pIlEwI155dBCg3Af49g/1c6pLCsqEWHYpDHfTmhLf2+jpI89blbd4MTYyqPM+pIlEwI155dBCg3Af49g/1c6pLCsqEWHYpDHfTmhLf2+jpI89blbd4MTYyqPM+pIlEwI155dBCg3Af49g/1c6pLCsqEWHYpDHfTmhLf2+jpI89blbd4MTYyqPM+pIlEwI155dBCg3Af49g/1c6pLCsqEWHYpDHfTmhLf2+jpI89blbd4MTYyqPM+pIlEwI155dBCg3Af49g/1c6pLCsqEWHYpDHfTmhLf2+jpI89blbd4MTYyqPM+pIlEwI155dBCg3Af49g/1c6pLCsqEWHYpDHfTmhLf2+jpI89blbd4MTYyqPM+pIlEwI155dBCg3Af49g/1c6pLCsqEWHYpDHfTmhLf2+jpI89blbd4MTYyqPM+pIlEwI155dBCg3Af49g/1c6pLCsqEWHYpDHfTmhLf2+jpI89blbd4MTYyqPM+pIlEwI155dBCg3Af49g/1c6pLCsqEWHYpDHfTmhLf2+jpI89blbd4MTYyqPM+pIlEwI155dBCg3Af49g/1c6pLCsqEWHYpDHfTmhLf2+jpI89blbd4MTYyqPM+pIlEwI155dBCg3Af49g/1c6pLCsqEWHYpDHfTmhLf2+jpI89blbd4MTYyqPM+pIlEwI155dBCg3Af49g/1c6pLCsqEWHYpDHfTmhLf2+jpI89blbd4MTYyqPM+pIIEwI155dBCg3Af49g/1c6pLCsqEWHYpDHfTmhLf2+jpI89blbd4MTYyqPM+pIIEwI155dBCg3Af49g/1c6pLCsqEWHYpDHfTmhLf2+jpI89blbd4MTYyqPM+pIIEwI155dBCg3Af49g/1c6pLCsqEWHYpDHfTmhLf2+jpI89blbd4MTYyqPM+pIIEwI155dBCg3Af49g/1c6pLCsqEWHYpDHfTmhLf2+jpI89blbd4MTYyqPM+pIIEwI155dBCg3Af49g/1c6pLCsqEWHYpDHfTmhLf2+jpI89blbd4MTYyqPM+pIIEwI155dBCg3Af49g/1c6pLCsqEWHYpDHfTmhLf2+jpI89blbd4MTYyqPM+pIIEwI155dBCg3Af49g/1c6pLCsqEWHYpDHfTmhLf2+jpI89blbd4MTYyqPM+pIIEwI155dBCg3Af49g/1c6pLCsqEWHYpDHfTmhLf2+jpI89blbd4MTYyqPM+pIIEwI155dBCg3Af49g/1c6pLCsqEWHYpDHfTmhLf2+jpI89blbd4MTYqQPM+pIIEwI155dBCg3Af49g/1c6pLCsqEWHYpDHfTmhLf2+jpI89blbd4MTYqQFM+pI89blbd4MTYqQFM+pI80blbd4MTYqQFM+pI80blbd4MTYqQFM+pI80blbd4MTYqPM+pI80blbd4MTYqPM+pI80blbd4MTYqPM+pI80blbd4MTYqPM+pI80blbd4MTYqPM+pI80blbd4MTYqPM+pI80blbd4MTYqQFM+pI80blbd4MTYqQFM+pI80blbd4MTYqQFM+pI80blbd4MTYqQFM+pI80blbd4MTYqQFM+pI80blbd4MTYqqFM+qI80blbd4MTYqqFM+qI80blbd4MTYqFM+qI80blbd4MTYqFM+qI80blbd4MTYqFM+qI80blbd4MTYqFM+qI80blbd4MTYqFM+qI80blbd4MTYqFM+qI80blbd4MTYqFM+qI80blbd4MTYqFM+qI80blbd4MTYqFM+qI80blbd4MTYqFM+qI80blbd4MTYqFM+qI80blbd$ |
| 2a+uK5Xi2huzNn99SH8dYQ5xM4Oa6tieLIsmPV+SXPbKRmlpiIVjUzBxGvJ14Bma/OlipW90xayzDtYkxDsW78jzjgIJcptd |
| WQ3W5BD6tpnuDVdiyt5G0IC6jSu+ZQBb44I7oXCKzthF841qX90zS0upzctylCS8L5ebI7Z5JyBBb+8Bs5ziWqWDkJGoABh5D000000000000000000000000000000000000 |
| gs7bUNdK64gUi8Q3kNujUdWbe1SoV5qVI8GfNF/gX41CoXikHxp7qp90i0PRwXyUG8XEdTe1ctYla0doVxhm58NAWhGJD9t1 |
| +mbZ7xeAds3jOsiICOonHorVbcToK6Tk7r3qz6GJX2lNxv5fY0oQflI2MAdtpz9Zrbvvy****", |
| "AccessKeyId": "STS.NTyAFMjGqjGQSQb3NLTv8****", |
| "ExpireUTCTime": "2022-02-11T*****", |
| "AccessKeySecret": "9fFQt2yRRY6F1PEoHMKWLbafxyRwPM61nEHCo6C****", |
| "Expiration": "****", |
| "Region": "cn-shanghai" |
| } |
| |

4.3.5. URL批量拉取上传

当文件没有存储在本地服务器或终端时,需要通过公网访问URL地址上传。用户指定媒体的URL地址,视频点播通 过封装好的接口拉取上传。本文介绍URL批量拉取上传的使用说明及相关文档。

使用说明

视频点播提供一个封装好的服务,帮助开发者将媒体文件通过内网访问的URL地址拉取并上传到视频点播的存储地址。这种方式可以免去先将文件下载到本地服务器或终端后再使用上传SDK上传到点播的过程。URL批量拉取上传的时效性较低,一般提交后会在数小时、甚至数天内完成迁移上传,主要适用于离线搬站场景。

● 推荐集成点播服务端SDK后调用点播的URL拉取上传接口完成上传。

| 开发语言 | 调用参考 |
|---------|-------------|
| Java | Java媒体上传 |
| Python | Python媒体上传 |
| РНР | PHP媒体上传 |
| .NET | .NET媒体上传 |
| Node.js | Node.js媒体上传 |
| Go | Go媒体上传 |
| C/C++ | C/C++媒体上传 |

• 您也可以直接调用服务端API,但需要自行构建HTTP或HTTPS请求。点播请求构造请参考请求结构。

相关文档

- 获取URL上传信息
- 取消URL上传任务

4.3.6. 基于OSS API上传

本文为您介绍基于OSS API上传的使用场景和相关文档。

使用说明

视频点播支持直接调用点播服务端接口获取上传地址和凭证,再调用OSS原生API完成上传。适用于自动化上传、 大批量迁移视频上传、网络媒体文件上传等场景。考虑到上传效率和出错概率等因素,一般不推荐使用此方式进 行上传操作。

直接调用服务端API的方式需要用户自行构建HTTP或HTTPS请求。

- 点播请求构造请参考请求结构。
- OSS请求构造请参考公共HTTP头定义。

相关文档

- 调用视频点播相关服务端API获取上传地址和凭证,相关API信息如下:
 - 获取音视频上传地址和凭证
 - o 刷新视频上传凭证
 - 获取图片上传地址和凭证
 - 获取辅助媒资上传地址和凭证
- Base64解码上传凭证和地址后再使用OSS能力上传文件,具体操作请参见上传文件。

5.媒资管理

5.1. 概述

在完成了媒体上传和媒体处理后会产生大量的媒资,视频点播提供统一强大的媒资管理服务,支持通过控制台和 调用接口进行管理。本文为您介绍什么是媒资、媒资的类型、媒资管理功能、媒资管理方式和使用场景。

什么是媒资

内容生产过程中会产生的大量的音视频、图片、文字等业务数据, 媒资是指这些业务数据、描述这些业务数据的 元数据以及相关的一些数据信息。

媒资类型

媒资信息的分类如下图所示:



| 信息类型 | 描述 | 参考文档 |
|-------|---|---|
| 基础信息 | 音视频信息:包括视频ID、标题、描述、时长、封面URL、状态、创建时间、文件大小、截图、分类、标签、存储区域等。 图片信息:包括图片ID、标题、URL、状态、描述、文件大小、分类、宽高、创建时间、标签、存储区域等。 水印、字幕等辅助媒资:包括标题、URL、描述、文件大小、业务类型、分类、创建时间、标签、存储区域等。 ⑦ 说明 其中 媒资分类 是全局统一设置,所有媒资类型都可使用,最多支持三级,可通过登录视频点播控制台,在点播控制台左侧的导航栏选择配置管理,单击分类管理,进入分类管理页面,在分类管理页面进行设置,或者调用分类API来设置。 | 视频信息 Video 音频流信
息AudioStream 图片信息
ImageInfo 辅助媒资信息
AttachedMediaIn
fo 分类管理 分类API |
| 源文件信息 | 包括文件名称、时长、大小、状态、宽高、帧率、地址、码率、输出文件
类型、创建时间等。 | 源文件信息
Mezzanine |
| 播放信息 | 包括视频流码率、清晰度、时长、是否加密、播放地址、格式、帧率、宽
度、高度、流类型、转码作业ID、水印ID、流状态、创建时间、修改时间
等。 | 播放信息 PlayInfo 音视频播放 |
| AI数据 | 智能审核信息:智能审核状态(未审核、成功、失败、审核中、重新
审核中),审核内容(视频、封面、标题),审核来源(画面),智
能审核结果(违规、疑似、正常)。 AI结果信息:语音识别、文本识别、实体标签、人物识别等AI处理结果
信息。 | 智能审核结果
AlMediaAuditRes
ult 智能审核 视频Al参数说明 视频Al |

媒资管理使用场景

- 全方位地管理视频和图片等多种媒体类型的元数据和meta信息,降低维护大量内容的成本。
- 通过多种方式获取媒资信息,支持标题和描述的分词搜索,能够快速方便的找到媒资信息,为长期频繁的使用 媒体内容提供了便利。
- 多种类型媒资之间相互打通和关联,将音频、视频、图片、文字、流信息、AI信息等多种信息关联在一起,通过一种媒资可以方便的找到其关联的信息,比如通过视频可以方便的找到对应的截图、封面、播放信息、AI信息等。
- 为后续的内容生产提供了素材,已有的素材可以应用到剪辑、转码等多个环节的内容生产。
- 媒资管理的文件均会生成网络地址,为后续的分享和持续传播提供了便利。

媒资管理功能

视频点播提供获取媒资信息、更新媒资信息、搜索媒资信息、下载媒体文件和删除媒体文件的媒资管理功能。

| 功能 | 描述 | 参考文档 |
|--------|---|--------|
| 获取媒资信息 | 通过媒资ID获取媒资的基本信息,不同的媒资类型,需要调用不同的接口。 | 获取媒资信息 |
| 更新媒资信息 | 支持媒资标题、描述、分类、标签等信息修改,媒资文件的大小、状态、
存储区域等字段无法修改。 | 更新媒资信息 |
| 搜索媒资信息 | 支持媒资基本信息的搜索,其中标题和描述使用了分词器,支持模糊搜
索。为不同的媒资类型设计了统一的 <mark>搜索媒体信息</mark> 接口和统一的 <mark>媒资搜索</mark>
<mark>协议</mark> 。 | 搜索媒资信息 |
| 下载媒体文件 | 音视频下载: 音视频的源文件信息可以通过获取源文件信息接口来获取源文件地址,进行下载。 图片下载: 图片可以通过获取媒资信息和搜索媒资获取到URL来保存文件 客户端下载:移动端用户通过"离线下载"功能将视频缓存至本地观看,需集成官方播放器SDK。 | 下载媒体文件 |
| 删除媒体文件 | 使用点播服务产生的多种媒体文件,包括音视频源文件、转码输出的流文
件、视频截图文件、上传的图片文件等,点播提供了各种粒度的删除功
能。 | 删除媒体文件 |

媒资管理方式

视频点播支持通过控制台和调用接口两种方式进行媒资管理。

● 控制台管理

开通点播服务后,登录视频点播控制台,在点播控制台左侧的导航栏选择**媒资库**,单击**音/视频、图片**或**短视频素材**,进入音视频、图片或短视频素材页面即可对媒资进行管理。提供音频、视频、图片上传、删除、更新、查看详情及搜索等功能,删除媒资信息的同时会彻底删除源文件不可恢复。操作步骤,请参见媒资管理。

| | (上海) - | Q 搜索文档、控制台、AP | 、解决方案和资源 | 费用 工单 | 备案 企业 | 支持 官网 Ъ | Ū. Ä | O 16(4 📀 |
|-----------------------------|--|---------------|----------|-------|-------|------------------------|---------|----------|
| 点播控制台 | ▲ 点播控制台 / 音/视频 | | | | | | | 4 |
| 概览 | 音/视频 | | | | | | | 查看上传任务 |
| 媒资库 | 上传音/税績 全部时间 > 媒体名称 > 诱胞入媒体名称 | Q | | | | | | ₹ C |
| 音/视频 | 首/视频↓ | 类型⑦ | 分类公 | 状态冒 | 来源▽ | 创建时间小 | 操作 | |
| 图片
短视频奏材 | C C Market: outin- | 视频 | 未分类 | ✓ 正常 | 控制台 | 2020-09-04
16:05:25 | 管理 删除 | 更多▼ |
| 制作中心
视频剪辑 | MG 2393.MOV
ID:
Mistait: outin- | 视频 | 未分类 | ✓ 正常 | 控制台 | 2020-09-01
14:44:03 | 管理 删除 | 更多 🔹 |
| 审核管理
视频审核
审核设置 | ID CONTRACTOR OF | 视频 | 未分类 | ✓ 正常 | 控制台 | 2020-08-26
17:49:34 | 管理 删除 | US • |

● 调用接口管理

提供音视频及图片的搜索、查看详情、批量更新、删除接口等。更多信息,请参见媒资管理API。

5.2. 获取媒资信息

媒体文件上传完成或完成处理后,就可以通过控制台或者API获取到相应媒资信息。本文为您介绍通过控制台和 API/SDK获取媒资信息的方法。

简介

可获取的媒资信息包括媒体文件的基本信息、源文件信息。

- 如果文件为视频,还可以获取到视频的播放信息。
- 如果开通了AI服务,对视频文件进行了AI处理,还能获取到相应的AI数据。

控制台管理

登录视频点播控制台,在点播控制台左侧的导航栏选择**媒资库**,单击**音/视频、图片**或**短视频素材**,进入音视频、图片或短视频素材页面即可通过单击管理查看媒资详情,提供音视频及图片视频的列表。具体操作,请参见媒资管理。

| ☰ (-)阿里云 | 华东2(上海) 🔸 | Q 搜索文档、控制台。 | 、API、解决方案和资源 | 费用 工单 | 备案 企业 | 支持 官网 | d. | © # | 简体 | 0 |
|-----------------------------|--|-------------|--------------|-------|-------|------------------------|----|---------|-------|---|
| 点播控制台 | ▲
点攝控制台 / 音/视频 | | | | | | | | | 1 |
| 概览 | 音/视频 | | | | | | | 査 | 看上传任务 | 8 |
| 媒资库 | 上传音/视频 全部时间 > 媒体名称 > 请输入媒体名称 | Q | | | | | | ć | F C | |
| 音/视频 | □ 音/砚频▽ | 类型公 | 分类♡ | 状态公 | 来源公 | 创建时间小 | 操作 | | | |
| 图片
短视频康材 | D D D D D D D D D D D D D D D D D D D | 视频 | 未分类 | ✓ 正常 | 控制台 | 2020-09-04
16:05:25 | 管理 | 删除 更多 | | |
| 制作中心
视频照镜 | く
ID: 2333MOV
ID: 105
105
105
105
105
105
105
105 | 视频 | 未分类 | ✓ 正常 | 控制台 | 2020-09-01
14:44:03 | 管理 | 删除 更多 | . • | |
| 审核管理
视频审核
审核设置 | orbat | 视频 | 未分类 | ✓ 正常 | 控制台 | 2020-08-26
17:49:34 | 管理 | 删除 更多 | | |

API/SDK操作

获取媒资信息的相关API如下所示:

| 获取信息类型 | 描述 | 参考文档 |
|--------|-------------|----------------|
| 基础信息 | 获取单个音视频详情 | 获取视频信息 |
| | 批量获取音视频信息 | 批量获取视频信息 |
| | 获取图片信息 | 获取图片信息 |
| 源文件信息 | 获取音视频的源文件信息 | 获取源文件信息 |
| 播放信息 | 获取播放信息 | 获取播放地址播放 |
| | 获取智能审核结果摘要 | 获取智能审核结果摘
要 |
| | | |

| AI数据
获取信息类型 | 描述 | 参考文档 |
|----------------|----------------------------|----------------|
| | 获取智能审核结果详情 | 获取智能审核结果详
情 |
| | 语音识别、文本识别、实体标签、人物识别等Al结果信息 | 视频AI |

? 说明

也可以通过搜索接口来获取媒资信息,更多信息,请参见搜索媒资信息。

调用方式:

- 推荐使用服务端SDK调用API来获取媒资信息,更为简单、高效。更多信息,请参见使用说明, API详情可参考各 API文档的 SDK示例 。
- 您也可以参见请求结构,来生成HTTP/HTTPS请求。

5.3. 更新媒资信息

视频点播支持在上传媒体文件前先设置一部分媒资信息,也可对已经上传的部分媒体数据进行修改更新。本文为 您介绍通过控制台和API/SDK更新媒资信息的方法。

简介

您可以在上传媒体文件时就指定一部分媒资信息,也可以在完成上传后对存量的媒资数据进行更新。

- 支持修改视频的标题、分类、描述等。
- 不支持修改媒体文件本身的部分源文件信息比如文件大小、类型、存储区域等。

控制台管理

登录<mark>视频点播控制台</mark>,在点播控制台左侧的导航栏选择**媒资库**,单击**音/视频、图片**或**短视频素材**,进入音视频、图片或短视频素材页面即可通过单击**管理**查看媒资详情,并编辑视频信息。具体操作,请参见媒资管理。

| 点播控制台 | ^ | 点播控制台 / 音/树 | 点播控制台 / 音/视频 / 编辑视频信息 | | | | |
|--------|----------|-------------|---|--|--|--|--|
| 概览 | | ← 编辑视频信息 | | | | | |
| 媒资库 | | 基础信息 | 视频地址 Web播放器代码 | | | | |
| 音/视频 | | | 您可以从下图中选择一张图片作为封面 | | | | |
| 图片 | | 封面 | | | | | |
| 短视频素材 | _ | | | | | | |
| 制作中心 | | | 您也可以自主义封面,仅支持JPG / PNG格式,最大1MB,不超过1920*1080
+上传封面 | | | | |
| 视频剪辑 | | | | | | | |
| 审核管理 | - 1 | < * 名称 | | | | | |
| 视频审核 | | 描述 | 请输入 | | | | |
| 审核设置 | _ | | | | | | |
| 配置管理 | | 分类 | 未分类 🗸 🗸 🗸 去普理分类 | | | | |
| 媒资管理配置 | ^ | 标签 | | | | | |
| 存储管理 | | | 支持中,英文及下划线,输入逗号、分号,或按回车键完成单个标
签输入 | | | | |
| 分类管理 | | ID | and the second | | | | |
| 媒体处理配置 | ^ | 时长 | 00:00:04 | | | | |
| 转码模版组 | | 原片大小 | 504 KB | | | | |
| 参照特に | | 存储地址 | outin-(《集之》 | | | | |
| 截圍慢做 | | 创建时间 | 2020-11-09 16:08:31 | | | | |
| 动图模板 | | 最近更新 | 2020-11-09 16:08:35 | | | | |

API/SDK操作

更新媒资信息的相关API如下所示:

| 获取信息类型 | 描述 | 参考文档 |
|--------|---|----------|
| 基础信息 | 修改单个音视频信息,可以修改视频的标题、描述、封面图、分类ID和标
签等信息。 | 修改视频信息 |
| | 批量修改音视频信息,批量接口也支持单个视频的信息。可以修改视频的
标题、描述、封面图、分类ID和标签等信息。 | 批量修改视频信息 |
| | 图片修改图片信息,该接口也支持单个图片信息的修改。可以修改图片的
标题、描述、标签、分类ID等信息。 | 批量更新图片信息 |
| 源文件信息 | 源文件信息从文件本身提取的Meta信息,文件本身的大小、类型、宽
高、状态、存储区域等属性不支持修改。 | 获取源文件信息 |

| 获取信息类型 | 描述 | 参考文档 |
|--------|--|----------------|
| 播放信息 | 播放信息中输出的编码格式、清晰度、码率帧率等信息本身不能修改。但
您可通过转码,转出新的音视频流。 | 音视频转码 |
| AI数据 | AI元数据是AI处理产生的结果数据,不支持修改,但可以重新提交AI任务
进行覆盖。 | 获取智能审核结果摘
要 |

调用方式:

- 推荐使用服务端SDK调用API来获取媒资信息,更为简单、高效。更多信息,请参见使用说明, API详情可参考各 API文档的 SDK示例 。
- 您也可以参见请求结构,来生成HTTP/HTTPS请求。

5.4. 搜索媒资信息

搜索媒资信息,是指对视频点播媒资库(视频、音频、图片等)进行搜索、筛选、过滤和排序,以获取到符合搜 索条件的媒资信息。本文为您介绍通过控制台和API/SDK搜索媒资信息的方法、使用限制和使用API的示例。

使用方式

视频点播提供了如下两种方式来搜索媒资信息:

• 通过控制台搜索

登录控制台,进入<mark>媒资库</mark>页面,可以通过单击媒体名称或媒体ID(图片名称或图片ID)搜索媒资信息。具体操 作,请参见媒资管理。

| 点播控制台 / 音/视 | 频 | | | | | |
|-------------|------|------|------|-----|----------------------------------|---|
| 音/视频 | | | | | | |
| 上传音/视频 | 全部时间 | ~ | 媒体名称 | ^ | 请输入媒体名称 | Q |
| 音/视频♡ | | | 媒体名称 | ~ | | |
| 1.1 | | : 76 | 媒体ID | 500 | Contract of Contract of Contract | - |

● 通过API/SDK搜索

使用搜索媒体信息接口,同时配合媒资搜索协议使用。

使用限制

翻页限制

为避免深度翻页造成性能问题,媒资搜索限制只返回命中搜索条件的部分数据。如果要获取更多数据、甚至遍历 所有数据,则需要使用翻页标记(ScrollToken)、会话ID(SessionId)。

- 获取部分数据使用翻页参数(PageNo 、 PageSize),且不传翻页标识(ScrollToken 、 SessionId
 -),可逐次获取前5000条数据。

● 获取更多数据,或遍历所有数据使用翻页参数(PageNo 、 PageSize),且传入翻页标识(

ScrollToken 、 SessionId),则翻页位置与当前位置之间不能超过1200条数据;但可以在翻到最后一页 时获取到接下来的1200条,以此类推,直到取完所有数据。也就是,可以根据翻页标识,分多段拿数据,但每 段限制最多1200条。

API使用示例

下文将以搜索视频信息为例介绍搜索语句。

↓ 注意

- 发送请求前需要对请求参数做URL编码。
- 同时语句中使用的等号、双引号、单引号、括号必须为英文半角。

| 分类 | 描述 |
|------|--|
| 返回字段 | 媒资搜索接口默认返回媒资基本信息,当需要附加其他媒资信息时需要指定 Fields 字段。查询
结果中包含 Title 和 CoverURL :
Fields=Title,CoverURL
请求示例:
http://vod.cn-shanghai.aliyuncs.com?Action=SearchMedia
&Fields=Title,CoverURL |
| 精确匹配 | 査询 VideoId 値为 28ba2b26d540446c94cdd2c4c48090e5 的视频信息: VideoId='28ba2b26d540446c94cdd2c4c48090e5' 请求示例: http://vod.cn-shanghai.aliyuncs.com?Action=SearchMedia &Match=VideoId='28ba2b26d540446c94cdd2c4c48090e5' |

| 分类 | 描述 |
|------|--|
| 模糊匹配 | Title 中包含 音乐 , 可以写成:
Title='音乐'
或者写成
Title in ('音乐')
请求示例:
http://vod.cn-shanghai.aliyuncs.com?Action=SearchMedia
eMatch=Title='音乐' |
| 多值查询 | 查询 Status 値为 Normal 或 Checking
Status in ('Normal','Checking')
请求示例:
http://vod.cn-shanghai.aliyuncs.com?Action=SearchMedia
éMatch=Status in ('Normal','Checking') |

| 分类 | 描述 |
|------|--|
| 范围查询 | 使用开闭区间表示范围, CreationTime 位于 2018-01-01T00:00:00z 和
2018-02-01T00:00:00z 之间:
CreationTime=('2018-01-01T00:00:00z','2018-02-01T00:00:00z') |
| | 请求示例:
http://vod.cn-shanghai.aliyuncs.com?Action=SearchMedia
&Match=CreationTime=('2018-01-01T00:00:00Z','2018-02-01T00:00:00Z') |
| | 如果只有左边界或右边界,对应右边界或左边界值为空即可,如 CreationTime 大于
2018-01-01T00:00:00Z :
CreationTime=('2018-01-01T00:00:00Z',) |
| | 请求示例:
http://vod.cn-shanghai.aliyuncs.com?Action=SearchMedia
&Match=CreationTime=('2018-01-01T00:00:00Z',) |
| 排序字段 | 按创建时间倒排:
CreationTime:Desc |
| | 请求示例:
http://vod.cn-shanghai.aliyuncs.com?Action=SearchMedia
&SortBy=CreationTime:Desc |

5.5. 下载媒体文件

视频点播支持将存储在视频点播里的媒体文件下载到本地磁盘或其它存储上,提供了控制台、客户端和接口用于 查询各种媒体文件的下载地址。本文为您介绍媒体文件、下载地址类型、下载限制和获取方法。

媒体文件

使用视频点播服务时可能会产生多种可下载的媒体文件,包括源文件、转码输出的流文件、视频截图文件、上传的图片文件等,视频点播提供了各种媒体文件的查询和下载。媒体文件的分类如下:

● 音视频

- 源文件:上传至点播的音视频源文件。
- 转码输出文件:下载转码输出的流文件(某个清晰度)。
- 图片
 - 上传至点播的图片文件。
 - 经视频点播自动截图处理后生成的图片文件(封面截图和雪碧图)。
 - · 提交媒体截图作业后生成的图片文件(普通截图、雪碧图和组成雪碧图的原始小图)。详情请参见提交媒体截图作业。

下载地址类型

获取的媒体文件下载地址,一般为存储地址或加速地址,前者会产生媒资管理(存储)费用,即OSS外网流出流量的费用。后者会产生CDN流量费用,即视频加速(加速)的费用。

- 存储地址
 - 未配置加速域名时, 各媒体文件的下载地址统一为存储地址。更多信息, 请参见存储管理。
 - 配置加速域名时,各媒体文件的下载地址统一为加速地址。其中,源文件和转码输出文件下载地址支持强制返回源站地址。更多信息,请参见获取源文件信息和获取视频播放地址。调用以上接口时,需要设置
 OutputType 值为 oss 。当您的视频存储地址为输入bucket时(bucket名称以 in- 开头),配置加速域名的情况下,源文件下载地址仍然为存储地址。
- 加速地址

当您在点播配置了加速域名后,各媒体文件的下载地址为加速地址。具体操作,请参见域名管理。

下载限制

如果使用了防盗链配置、视频加密等功能,即便获取到媒体文件的下载地址,在下载媒体后仍可能受到同样的使 用限制。

- 配置了访问限制时,通过加速地址下载,会受到配置的Refer防盗链、IP黑名单等策略的限制,与在线播放相同; 通过存储地址下载,则不受此限制;
- 使用视频加密,不管通过哪种方式下载,得到的转码输出的文件仍然为加密文件,需要解密后进行播放。更多 信息,请参见概述-视频加密;
- OSS存储如设置私有,则地址存在时效性,下载时要注意可能会过期,此时需要重新获取,建议设置较长有效 期;
- CDN加速域名如开启了URL鉴权,则地址会在一段时间后过期,此时需要重新获取,建议设置较长有效期。

通过控制台获取下载地址

查看

登录视频点播控制台,在点播控制台左侧的导航栏选择媒资库,单击音/视频或图片,进入音/视频或图片页面, 在音视频或图片页面单击要查看的文件右侧的管理。在详情页中单击复制,即可获取下载地址。

- 导出
 - 音视频转码输出文件:

登录视频点播控制台,在点播控制台左侧的导航栏选择**媒资库**,单击音/视频,进入音视频页面,通过单击 右上角 <u>。</u>图标导出媒体地址,获取当前筛选条件下的前200条音视频对应的转码输出文件下载地址(各个 清晰度)。 ○ 上传至点播的图片文件:

登录视频点播控制台,在点播控制台左侧的导航栏选择媒资库,单击图片,进入图片页面,通过单击右上角 图标导出媒体地址,获取当前筛选条件下的前100条图片对应的文件下载地址。

通过API/SDK获取下载地址

媒体文件下载包括下载音视频源文件、下载音视频流、下载自动截图文件、下载图片文件。您可以通过以下接口 获取到相应的文件地址进行下载。

| 获取信息 | 描述 | 参考文档 |
|-------|--|----------|
| 源文件地址 | 返回结果Mezzanine对象中的 FileURL 字段为源文件地址。 | 获取源文件信息 |
| 播放信息 | 返回结果PlayInfo对象中的 PlayURL 字段为转码输出流地址。 | 获取视频播放地址 |
| 视频信息 | 返回结果Video对象中的 CoverURL 字段为封面地址, Snapshots 字段为自动截图处理后截图地址数组。 | 获取视频信息 |
| 图片信息 | 返回结果ImageInfo对象中的 URL 字段为图片地址。 | 获取图片信息 |
| 媒资信息 | 媒资类型为音视频时,返回结果Video和Audio对象中的 CoverURL 字
段为封面地址, Snapshots 字段为视频截图地址数组,
SpriteSnapshots 字段为自动截图处理后的雪碧图地址数组;媒资
类型为图片时,返回结果Image对象中的 URL 字段为图片地址。 | 搜索媒体信息 |
| 截图数据 | 返回结果MediaSnapshot对象中的 Snapshots 字段为与 SnapshotType (截图类型)对应的截图地址数组。 | 查询截图数据 |

调用方式:

- 推荐使用服务端SDK调用API来获取媒资信息,更为简单、高效。更多信息,请参见使用说明,API详情可参考各 API文档的 SDK示例 。
- 您也可以参见请求结构,来生成HTTP/HTTPS请求。

客户端下载

针对移动端的使用场景,阿里云播放器SDK提供了**离线下载**功能,您可以根据业务需要,通过启用下载功能,允 许终端用户将音视频流文件缓存至本地进行观看。离线下载提供两种下载方式——普通下载和安全下载。

下载设置

您可以登录<mark>视频点播控制台</mark>,在点播控制台左侧的导航栏选择**配置管理**,单击**分发加速配置 > 下载设置**,进入 下载设置页面,在下载设置页面中设置离线下载的方式。具体操作,请参见下载设置。

| 视频剪辑 🖸 | 点题是 () () () () () () () () () (| 新手引导 |
|------------|--|------|
| 审核管理 | 下载设置 | |
| 视频审核 | ● 帮助信息
开码后,北计移动调用户通过 南线下载 功能将视频带存至本地现着 (當集成官方屢於器SDK) 释動交档 | |
| 审核设置 | 下载设置 | |
| 配置管理 | - 下级功能 | |
| 煤资管理配置 🗸 🗸 | | |
| 媒体处理配置 🗸 🗸 | | |
| 分发加速配置へ | | |
| 域名管理 | | |
| 刷新预热 | | |
| 下载设置 | | |
| 数据中心 | | |
| 监控统计 🗸 | | B |
| 用量查询 | | |

- 普通下载的音视频文件不会被加密,可拷贝,也可使用任意播放器播放,请谨慎使用。
- 安全下载的音视频文件会使用生成的密钥文件进行加解密,保障视频安全。需要使用阿里云播放器SDK完成解密 播放,点播控制台提供了加密私钥的生成工具。

| 开启下载功能 | > | < |
|--------|--|---|
| * 下载方式 | 普通下载 离线下载的视频未加密,任何播放器均可播放,请谨慎使用 | |
| | 安全下载 离线下载的视频会使用下面生成的密钥文件加解密,下载后请安全保存在 客户端应用中 | |
| | 确定取消 | |

播放器SDK

iOS和Android播放器SDK同时支持客户端下载和缓存功能,具体操作,请参见iOS播放器<mark>功能使用</mark>和Android播放 器<mark>功能使用</mark>。

5.6. 删除媒体文件

使用点播服务可能会产生多种媒体文件,点播提供了各种粒度的删除功能,支持通过控制台和API/SDK进行删除。 本文为您介绍点播的删除功能并提供删除媒体文件的操作指引。

删除功能

- 删除完整音视频:包括上传的源文件、转码输出的流文件、封面截图、雪碧图。
- 删除音视频流: 即删除转码输出的流文件(某个清晰度)。
- 删除源文件:即删除上传的音视频源文件。
- 删除图片: 即删除上传的图片文件, 删除后, 应用到音视频的封面、截图和水印等的图片地址将不可访问。

↓ 注意

- 与在OSS中删除不同:OSS上删除文件,只删除文件,不涉及媒资信息等;删除存储在点播服务的媒体 文件,会将文件和在点播建立的媒资信息一并删除。
- 所有删除操作均为物理删除,且不可恢复。
- 删除音视频源文件时,如果文件使用了不转码即分发的方式上传,由于源文件会被当作输出流用于播放(即原画),所以默认不可删除;如确定要删除,可指定强制删除的标识。
- 删除数据的操作只能用户自行操作。

通过控制台删除

登录视频点播控制台,在点播控制台左侧的导航栏选择媒资库,单击音/视频、图片或短视频素材,进入音/视频、图片或短视频素材页面,单击要删除文件右侧的删除。更多信息,请参见媒资管理。

| 上传题片 全部时间 > 图片名称 > 请输入图片名称 | Q | | | 不 |
|----------------------------|-----|------|---------------------|----------|
| 图片公 | 分类♡ | 状态▽ | 创建时间小 | 操作 |
| | 未分类 | ✓ 正常 | 2020-11-10 10:22:11 | 管理
删除 |

通过API/SDK删除

媒体文件删除包括删除音视频、删除音视频流、删除音视频源文件、删除图片。

相关API如下:

| 删除文件 | 描述 | 参考文档 |
|------|--|---------|
| 音视频 | 删除完整视频(包括其源文件、转码后的流文件、封面截图等),支持批
量删除。 | 删除完整视频 |
| 音视频流 | 删除媒体流(视频流,音频流)信息及存储文件,并且支持批量删除。 | 删除媒体流 |
| 源文件 | 批量删除源文件信息及存储。 | 批量删除源文件 |
| 图片文件 | 删除用户上传的图片及视频自动截图。 | 删除图片 |

调用方式:

- 推荐使用服务端SDK调用API来获取媒资信息,更为简单、高效。更多信息,请参见使用说明,API详情可参考各 API文档的 SD**K示例**。
- 您也可以参见<mark>请求结构</mark>,来生成HTTP/HTTPS请求。

5.7. 媒资管理FAQ

如何删除某个清晰度的流?

调用获取视频播放地址接口获取要删除的流ID(即JobId),再调用删除媒体流接口进行删除。

重转码后如何删除旧的流?

为了新旧输出流的播放能平滑切换,重转码后会保留旧的流文件。为了保证每次播放的是最新转码输出的流,播 放接口默认每种清晰度和格式只返回一路最新转码完成的流。调用获取视频播放地址接口时,可以设置 ResultType 为Multiple来获取该音视频下的所有转码完成的流,根据创建时间可以获取到旧的流D并删除。

如何删除加密流?

为了保障流信息的安全性,播放接口默认只返回非加密流。调用获取视频播放地址接口时,可以设置 ResultType 为Multiple来获取该音视频下的所有转码完成的流,根据PlayInfo结构体中的 Encrypt 标识,获取加密流ID并删 除。

6.媒体处理

6.1. 概述

视频点播提供了多样化的媒体处理能力,包括音视频转码、视频截图、视频动图、视频水印,并且提供了智能审核、云剪辑、AI处理等强大功能。本文为您介绍媒体处理的概述,帮助您在阅读媒体处理、媒体审核、云剪辑和 视频AI等文档前,对媒体处理有大致了解。

简介

阿里云视频点播提供的多样化的媒体处理能力,以经济、弹性和高可扩展的音视频转换方法,帮助您将音视频转 码成适合在PC、TV以及移动终端上播放的格式。用户可根据需要构建丰富的媒体处理流程,来满足不同的场景。

您可以通过控制台、API/SDK或客户端进行媒体处理,也可使用控制台的工作流功能,自定义处理流程。更多信息,请参见工作流。

| 处理能力 |
|------|
|------|

| 媒体处理 | 描述 | 参考文档 |
|-------|---|-------|
| 音视频转码 | 提供丰富的转码处理功能,将一个音视频文件转换成另一个或多个音视频文件,以适应不同网络带宽、不同终端设备和不同的用户需求。 点播基于长期的数据分析,进行多种清晰度场景化的预设配置,提供从上传到转码一体化的流程处理。 提供私有加密和标准加密等功能,有效保护视频内容。 | 音视频转码 |
| 视频截图 | 在指定视频特定的位置进行图像截取,并生成图片文件。 | 视频截图 |
| 视频动图 | 在指定视频中特定的一个区间进行内容截取,并生成动图文件。 | 视频动图 |
| 视频水印 | 在视频的编解码过程中将图片、文字等信息压制到视频流中合并输出一个
带图文水印的新的视频文件。 | 视频水印 |
| 媒体审核 | 支持视频、音频、图片、文本等全方位的媒体审核,包括智能审核、人工
审核、以及安全审核配置等,通过自动化、智能审核媒资库资源,极大降
低色情、暴恐、涉政等内容违规风险,降低人工审核成本。 | 媒体审核 |
| 云剪辑 | 点播云剪辑服务是阿里云点播服务的制作中心,支持剪切合并、混音、字幕、图片叠加、遮标、转场特效等一系列功能,提供在线可视化的剪辑平
台及相应的OpenAPI。
同时,云剪辑还提供了丰富的剪辑模版,视频Al的处理结果也可作为云剪
辑的输入源,进行二次制作。 | 云剪辑 |

| 媒体处理 | 描述 | 参考文档 |
|------|--|------|
| | 点播提供了智能封面、视频DNA、智能视觉等AI处理能力,深度解读视频
的每一帧,全方位挖掘结构化信息,为零算法基础的开发者和企业提供定
制化的模型训练能力,应用与各种场景可前往视频AI线上体验馆进行体
验。
效果样例如下: | |
| AI处理 | Artist 0.8.70 0.0.70 2.8.0.0 BARA (Att 0.0.70 2.8.0.0 2.7.00 5.8.0.0 BARA (Att 0.0.70 2.8.0.0 2.7.00 5.8.0.0 BARA (Att 0.0.70 2.8.0.0 2.7.00 5.8.0.0 BARA (Att 0.0.70 5.8.0.0 2.9.00 5.8.0.0 BARA (Att 0.0.70 5.8.0.0 2.9.00 5.8.0.0 BARA (Att 0.0.70 5.8.0.0 2.9.00 5.8.0.0 | 视频AI |

使用场景

通过点播的媒体处理,可以解决如下问题:

- 多终端设备适配:适配PC、TV以及移动终端等多平台播放。
- 多种网络环境适配:不同网络带宽的用户选择最佳码率,流畅播放。
- 降低存储、分发成本:保证在相同画质质量的前提下,调整视频码率、提高视频压缩率、减小文件体积,从而减少播放卡顿并节省存储和流量费用。
- 内容保护:为内容保护提供了有效的支持,适用于在线教育、原创保护等多种场景。

6.2. 音视频转码

转码是媒体处理中重要的一步,本文为您介绍转码的简介、阿里云支持的转码技术及如何配置转码模板和发起转码。

转码简介

什么是转码

视频转码(Video Transcoding)是指将已经压缩编码的视频码流转换成另一个视频码流,以适应不同的网络带宽、不同的终端处理能力和不同的用户需求。转码本质上是一个先解码、再编码的过程,因此转换前后的码流可能遵循相同的视频编码标准,也可能不遵循相同的视频编码标准。

转码在视频生产流程中的位置如下:



点播转码功能特性

- 视频处理:完善的转码与转封装能力,进行丰富的媒体格式转换。
- 音频处理: 音频转码、音频抽取等。

- 水印:支持静态图片水印、动态图片水印(如gif、mov等)和文字水印,并且支持多水印添加。更多信息,请参见视频水印。
- 多清晰度预置:标清、高清、超高、2k、4k等多个预置清晰度,提供最佳经验值,降低接入门槛。
- ・ 音频抽取等多场景解决方案:客户可在视频中提取一路普通音质、高音质等作为音频输出,如电台等场景。
- 内容保护: 内容加密, 适用于在线教育、原创保护等多种场景。

使用场景

- 多终端设备适配:适配PC、TV以及移动终端等多平台播放。
- 多种网络环境适配:多种网络环境适配:不同网络带宽的用户选择最佳码率,流畅播放。
- 降低存储、分发成本:保证在相同画质质量的前提下,调整视频码率、提高视频压缩率、减小文件体积,从而减少播放卡顿并节省存储和流量费用。
- 内容保护:为内容保护提供了有效的支持,适用于在线教育、原创保护等多种场景。
- 水印添加:在视频上添加企业Logo、电视台台标、用户ID或昵称等标志性信息,用于品牌宣传或者宣示当前视频的版权归属。

点播基于长时间及大量的数据分析,进行多种清晰度场景化的预设配置,提供从上传到转码一体化的流程处理。

转码技术

视频点播服务支持基础的普通转码和不转码外,还支持窄带高清TM1.0和窄带高清TM2.0转码。在配置转码模板时,可按需选择需要的转码技术。

<⇒ 注意

- 视频点播的转码服务均为收费项, 收费详情请参见视频转码定价。
- 同一视频每次转码都会产生转码计费,转码后会产生新的视频文件占用存储空间,请按需创建转码模板组及发起转码,避免不必要的空间浪费或费用产生。

普通转码

满足基础的转码需求,可根据需求选择不同的封装格式和清晰度将视频转码播放,适用于长视频。

视频点播系统预置了流畅、标清、高清、超清、2K、4K、原画(转封装)共7种清晰度,以及普通音质、高音质两 档音质。您可以对不同清晰度进行码率、分辨率、格式等自定义设置,同时使用转码模板组进行不同转码模板的 分组及管理。

窄带高清[™]1.0

窄带高清TM1.0基于阿里云自有的转码技术,对视频中每个场景、动作、内容、纹理等进行智能分析。在保证相同视频画质的同时,视频码率更低,降低带宽成本。更多内容,请参见云栖社区。

窄带高清[™]2.0

窄带高清TM2.0根据清晰度预期尝试降低码率,根据预设码率选择最佳分辨率。从人眼视觉模型出发,凭借阿里云 技术算法,将编码器的优化目标从保真度更高调整为主观体验更好,突破视频编码器的能力上限,在节省码率的 同时,也能提供更加清晰的观看体验。相比窄带高清TM1.0,码率更低,分辨率更佳。更多信息,请参见^{窄带高清} TM2.0。

? 说明

目前仅上海、北京地域支持窄带高清TM2.0。

不转码

不转码即直接将原片作为视频流进行播放,适用于短视频、直播转点播录制等场景,用户可直接播放原片。

- 在短视频等场景下,视频通常已在端上进行过拍摄和压缩,已经具备多终端播放的能力。此时可以不进行转码。
- 如果用户已经自己在本地进行过转码操作,不需要依赖点播的转码进行二次处理,也可以选择不转码。
- 目前仅以下格式的视频支持不转码直接播放: MP4、FLV、M3U8、MP3、WEBM。

配置转码模板组

转码参数十分复杂,为了避免用户将过多精力集放在参数管理,视频点播系统将这一系列复杂参数保存为模板。 用户可自行定义转码模板,它是转码参数(音频、视频、容器等)的集合,可以满足用户个性化的转码需求。一 个转码模板可包含多路输出,同时,视频点播根据视频清晰度、音频音质等不同,提供了一系列的参数建议值, 方便用户低门槛、快速使用。视频点播支持通过控制台和API的方式创建转码模板组。

普通转码模板组

控制台

登录视频点播控制台,在视频点播控制台左侧的导航栏选择**配置管理**,单击媒体处理配置>转码模板组,在 转码模板页面单击添加转码模板组,即可创建转码模板组。详细操作及相关参数配置解释,请参见普通转码模板 设置。

- · 视频点播系统预置了流畅、标清、高清、超清、2K、4K、原画共7种清晰度,以及普通音质、高音质两档音
 质,并且为每种清晰度提供了推荐的参数,当用户在控制台勾选所指定清晰度时,会自动填充系统推荐值,
 便于用户快速接入。
- 视频点播系统预置了一个普通转码模板组TranscodeTemplateGroup,此模板配置了常见的标清的HLS和 MP4两种视频格式,及其对应的码率、分辨率等参数,您可以快捷设置格式、水印及加密。如果您是新客 户,推荐您使用此模板组。

API

通过调用创建转码模板组接口,Definition参数指定为普通转码相关的值,可创建普通转码模板组。

窄带高清TM1.0模板组

• 控制台

与创建普通转码模板组的路径一致,在配置转码模板组时,选择**流畅-窄带高清1.0、标清-窄带高清1.0**等窄带高清1.0年常高清1.0相关的清晰度,即可创建转码模板组。详细操作及相关参数配置解释,请参见窄带高清TM1.0。

视频点播系统预置了多种窄带高清TM1.0的清晰度,支持MP4、HLS格式,具体参数如下:

| 清晰度 | 码率范围 | 分辨率-宽 |
|------------|-------|-------|
| 流畅-窄带高清1.0 | ≤400 | 640 |
| 标清-窄带高清1.0 | ≤800 | 848 |
| 高清-窄带高清1.0 | ≤1500 | 1280 |
| 超清-窄带高清1.0 | ≤3000 | 1920 |

| 清晰度 | 码率范围 | 分辨率-宽 |
|------------|-------|-------|
| 2K-窄带高清1.0 | ≤4000 | 2048 |
| 4K-窄带高清1.0 | ≤8000 | 3840 |

• API

通过调用创建转码模板组接口, Definition参数指定为窄带高清TM1.0相关的值, 可创建窄带高清TM1.0转码模板 组。

窄带高清[™]2.0模板组

需要通过提交工单来申请将普通转码模板开通为窄带高清TM2.0模板。更多信息,请参见^{窄带高清TM2.0}。

不转码模板组

点播系统内置了一个**不转码**的模板组,该模板组不支持编辑修改。点播服务开通后,默认使用**不转码**模板组。如 果您发起转码时采用了**不转码**模板组,您的视频地址将归类为**原画**。

? 说明

- 不转码时,要求用户上传的源片可以直接正常播放。目前仅以下格式支持不转码直接播放: MP4、 FLV、M3U8、MP3、WEBM。
- 上传视频时,非MP4格式的文件如果选择不转码模板组上传,上传完成后会直接成为原始文件,该文件无法通过视频点播控制台进行播放预览,也无法使用GetPlayInfo接口获取到播放地址,只能通过GetMezzanineInfo接口获取源文件地址用于播放。
- 控制台

登录<mark>视频点播控制台</mark>,在点播控制台左侧的导航栏选择**配置管理**,单击媒体处理配置 > 转码模板组,选择不 转码模板组。

| 名称 | | ID | 类型 | 操作 |
|-----|----|-------------|------|------|
| 不转码 | 默认 | 10-0-700470 | 系统内置 | 设为默认 |

• API

不转码模板组的转码模板组ID(TemplateGroupId)为VOD_NO_TRANSCODE,调用提交媒体转码作业等接口发起转码时,指定TemplateGroupId参数的值为VOD_NO_TRANSCODE。

发起转码

上传时自动触发转码

● 控制台

登录视频点播控制台,在点播控制台左侧的导航栏选择媒资库,单击音/视频,在音/视频页面可以直接进行上 传,在上传前选择转码模板。具体操作,请参见控制台上传。

| 添加音/视频 | 页 | | | | | × |
|--------|-------------------------|--------------------------|--------------|------------|-------------|----|
| 上传方式 | ◉ 本地上传 | | | | | |
| | ○ URL拉取
选择URL拉取方式时,非 | 告URL中不包含文件扩展名,为提高拉耳 | 双成功率,请填写扩) | 展名 了解URL批量 | 拉取上传 | |
| 存储地址: | | and the second | \checkmark | | | |
| 添加音/视频 | 用转码模版组处理 | TranscodeTemplateGroup | 未分类 🗸 | | | |
| 音/视频名称 | t | TranscodeTemplateGroup 🗸 | сл. | 分类 | 转码模板组/工作流模板 | 操作 |
| | | 不转码 | 4 | | | |

• API

通过调用获取音视频上传地址和凭证或URL拉取上传接口上传视频时指定所使用的转码模板组 ID(TemplateGroupId),上传完成后点播系统会自动根据所指定的模板进行转码,并且会根据用户配置的回调,将转码后的信息回调给用户。更多信息,请参见媒体上传。

上传后手动发起转码

● 控制台

登录<mark>视频点播控制台</mark>,在点播控制台左侧的导航栏选择**媒资库**,单击**音/视频**,在音/视频页面单击操作列的**媒** 体处理,对已有视频发起转码,该操作常用于对已有视频的重转码。

| / 市邇控制台 / 管/祝娟 | | | | | | | | 新手引导 |
|---|--|------------------------|------|----------------------|------|---------------------|------------------|--------|
| 音/视频 | | | | | | | | 宣誓上传任务 |
| 8 帮助信息
1、煤体文件是大支持上作45.6 TB的单个文件,將助文档
2、上传到间面的增加煤后及以中和处理后的煤后又并按序编音
3、未配置加速域名,通过控制台或存储地址增收。下板资源会 | 最终值与存放日期计器。计器详情
产生存编点出最用,不能使用完是包括归。计器详情 | | | | | | | |
| 上传音/祝新 全部时间 > 媒体名称 > 清給入 | 媒体各称 Q | | | | | | | ₹ C |
| · · · · · · · · · · · · · · · | | 20 | 分離分 | 状态 🔽 | 未運 💟 | 会议是951间 4 | 操作 | |
| 019.mp4
ID:
7798 | | 税類 | 未分类 | ✓ 正常 | 拉制台 | 2021-10-28 16:53:58 | 管理 建体处理 王多 - | |
| grev
ID:
7752 | 媒体处理 | | ALTE | × ;= | 拉制台 | 2021-09-16 1640:21 | 管理 媒体处理 更多 - | |
| non
10:
7#82 | 转码模糊泪 | TranscodeTemplateGroup | | 987. 3 96 | 拉制台 | 2021-02-26 13:51:52 | 管理 媒体处理 更多 - | |
| non
ID:
7税 | | | 織定 | NGE | 控制台 | 2020-12-18 16:28:38 | 管理 媒体处理 更多 * | |

• API

通过调用提交媒体转码作业接口的方式,指定所使用的转码模板组ID(TemplateGroupId),手动发起转码操作。该操作常用于对已有视频的重转码、标准加密、水印参数覆盖等。

场景实践

详情请参见最佳实践:如何选择转码类型。

6.3. 视频截图

视频点播支持视频截图,支持通过提前设置截图模板进行视频截图,您可通过控制台和API管理模板。本文为您介 绍视频截图、截图类型、使用说明、截图模板和截图参数,以及如何管理截图模板。

简介

视频截图,是指在指定视频特定的位置进行图像截取,并生成图片文件。为了提高易用性,视频点播提供了截图 模板,用户可先将相关参数配置成截图模板,在发起截取任务时指定截图模板ID即可。

↓ 注意

- 纯音频且无任何图像信息流的媒体文件、源文件损坏或源文件封装信息异常,则可能无法生成截图文件。
- 截图的过程是完全异步的,发出截图请求后,在接口返回结果时,任务可能还在排队中,并没有完成。可以通过接收视频截图完成事件消息获取截图结果。
- 截图耗时与文件的大小、时长以及截图采用的帧类型有关。

截图类型

● 封面截图 (CoverSnapshot)

视频点播对每一个视频源片都会进行截图,该类截图为封面截图,默认按照视频关键帧平均截取最多8张图片, 且截图时间点是从视频的第5ms开始。封面截图可在视频点播控制台视频管理详情页查看,可选择其中任意一 张作为视频封面。

? 说明

- 如果视频的关键帧数量不够8个,截图数将不足8张。
- 如果视频没有设置封面,则默认会使用封面截图的中间一张设置为视频封面。

● 普通截图 (NormalSnapshot)

通过API对指定视频截取一定数量的图片,可设置开始截图的时间点、截图的总数、截图的时间间隔以及截取的 图片宽高等。如果对视频通过API重复发起截图,视频点播只会记录最新的截图数据。更多信息,请参见提交媒 体截图作业。

• 雪碧截图 (SpriteSnapshot)

所谓雪碧图是指先进行普通截图,然后将普通截图按照一定的排列规则拼成一张大图,而这张大图即为雪碧 图,而此时的普通截图又称组成**雪碧图的原始图**。截取雪碧图的优势在于可降低图片的请求数量,从而可以通 过请求雪碧图一次获取多张截图的信息,加强了客户端的性能。 例如:按照10行、10列规则对普通图进行排列,则一张雪碧图中的小图理论数目为10×10=100张,由于受限于 普通截图的实际张数,雪碧图中小图数目可能不够100张,而如果超过100张则会再次生成第二张雪碧图,以此 类推。如下图示意:



⑦ 说明

上图示例中, 普通截图总数为50, 按照10×3排列。第一张雪碧图中小图数为30, 则第二张雪碧图中小图数为20。

• 雪碧图原始图 (SpriteOriginSnapshot)

雪碧图原始图,即用来拼接雪碧图的普通截图,对于这些原始截图,可以选择删除或者保留,如果保留可以通 过查询截图数据接口获取。更多信息,请参见查询截图数据。

● WebVTT截图

WebVTT截图即根据所有截图信息生成VTT的文件,截图的基本信息(截图时间、截图地址)会记录到VTT文件中,在使用缩略图时需要先获取VTT的内容,解析截图的信息进行展示,可用于播放器进度条缩略图展示。

WebVTT截图存储方式

• 截图单张存储

所有截图分开存储, VTT内容记录单张截图的相对位置信息、截图时间, VTT内容如下图所示:

WEBVTT

00:00:00.000 --> 00:00:00.000 snapshot-00001.jpg

00:00:00.000 --> 00:00:00.000 snapshot-00002.jpg

00:00:00.000 --> 00:00:00.000 snapshot-00003.jpg

00:00:00.000 --> 00:00:00.000 snapshot-00004.jpg

00:00:00.000 --> 00:00:01.000 snapshot-00005.jpg

00:00:01.000 --> 00:00:01.000 snapshot-00006.jpg

00:00:01.000 --> 00:00:01.000 snapshot-00007.jpg

00:00:01.000 --> 00:00:01.000 snapshot-00008.jpg

● 截图拼接存储

所有截图的图片会先拼接成一张大图统一存储,访问具体图片时需要从VTT内容中解析到图片的位置,VTT内容 如下图所示: < WEBVTT 00:00:00.000 --> 00:00:00.000 snapshot-tile-00001.jpg#xywh=0,0,200,200 00:00:00.000 --> 00:00:00.000 snapshot-tile-00001.jpg#xywh=200,0,200,200 00:00:00.000 --> 00:00:00.000 snapshot-tile-00001.jpg#xywh=400,0,200,200 00:00:00.000 --> 00:00:00.000 snapshot-tile-00001.jpg#xywh=600,0,200,200 00:00:00.000 --> 00:00:01.000 snapshot-tile-00001.jpg#xywh=800,0,200,200 00:00:01.000 --> 00:00:01.000 snapshot-tile-00001.jpg#xywh=1000,0,200,200 00:00:01.000 --> 00:00:01.000 snapshot-tile-00001.jpg#xywh=1200,0,200,200 00:00:01.000 --> 00:00:01.000

snapshot-tile-00001.jpg#xywh=1400,0,200,200

使用说明

● 封面截图

视频上传完成后,点播会默认进行封面截图,该类截图为封面截图,且该处理过程完全免费。

● API截图

通过提交截图作业接口发起对指定视频的截图任务,更多信息,请参见<mark>提交媒体截图作业</mark>。该方式可以截取**普通 截图**和雪**碧截图**。

• 如何获取截图

目前,视频点播支持通过以下三种方式获取视频的截图信息。

- 通过接收视频截图完成回调消息获取截图信息。
- 通过获取视频信息接口可获取默认截图信息。
- 通过查询截图数据接口查询截图数据。
- 如何删除截图

目前视频点播暂不支持截图单独管理,只随视频进行管理,即删除视频会将对应的截图信息以及截图文件都删 掉,且数据不可恢复。

管理截图模板

截图涉及到的参数相对较多,如果将诸多截图参数都统一在提交截图任务时传入,不仅加大了截图功能接入门 槛,更降低了易用性。因此视频点播提供了截图模板,用户可先将相关参数配置成截图模板,在发起截取任务时 指定截图模板ID即可。

视频点播支持控制台和API的方式管理截图模板。

• 通过控制台管理

通过点播控制台可以进行截图模板的添加、修改及删除操作。

| | - | | Q 搜索文档、控制台、API、解决方案和资 | 課 表用] | [单 备案 | 企业 支持 | 官网 | Þ. | ٥. | ₩ @ | 简体 | 0 |
|--------------|---------------------------------------|---------------------------------|-----------------------|--------|--------------------|---------|----|----|----|-----|----|----------|
| 图片
症视频素材 | ^{视频点播 / 載四模板} | | | | | | | | | | | Í |
| 制作中心 | | | | | | | | | | | | . 1 |
| 视频剪辑 | () 単位目的な「以人工工下がた、工165%)
活力の部分別将24万 | N、 3県145公理レイコロCH1。 」 第41688代第6日 | | | | | | | | | | 1 |
| 审核管理 | 模板名称 | ID | 截徑 | 防式 | 基本设置 | | | | | 操作 | | 11 |
| 视频审核 | | | | | 起始时间: 0 | 0:00:00 | | | | _ | _ | 1 |
| 审核设置 | 雪碧函載图模板 | 3d4871 | 雪 相 | 18 | 截图数量: 50
关键帧:关闭 |)
] | | | | 编辑 | 删除 | - 1 |
| 配置管理 | | | | | 起始时间: 0 | 0:00:01 | | | | | | - 1 |
| 媒演管理配置 > | 截图测试 | fc36ab3 | \$3 | 《截图 | 截烟数量: 3
关键帧: 关闭 |] | | | | 编辑 | 删除 | - 1 |
| 煤体处理配置 へ | | | | | | | | | | | | - 1 |
| 191-91060X43 | | | | | | | | | | | | - 1 |
| 动图模板 | | | | | | | | | | | | |
| 水印模板 | | | | | | | | | | | | |
| 工作流管理 | | | | | | | | | | | | |
| 回调设置 | | | | | | | | | | | | <u> </u> |

● 通过API管理

通过接口进行管理,更多信息,请参见截图模板。

截图参数

• 普通截图配置

? 说明

本节只介绍普通截图配置的部分参数使用细节,更多参数信息,请参见普通截图配置SnapshotConfig。

| API参数 | 控制台参数 | 描述 |
|-------------------------|-------|---|
| FrameType | 帧类型 | 截图截取的帧类型包括:关键帧(intra)、普通帧(normal)。
在同等截图规则条件下,一般关键帧截图比普通帧截图速度要快。 |
| SpecifiedOffse
tTime | 开始时间 | 截图开始的时间点。正整数类型,单位:毫秒。
如果是单帧截图,则 SpecifiedOffsetTime 为截图的时间点。 |
| Count | 截图总数 | 截图总数。 |

| API参数 | 控制台参数 | 描述 |
|----------|-------|---|
| Interval | 截图间隔 | 截取多张截图时,截图之间的时间间隔。
• Count>1:表示按照间隔时间截取到指定的Count数。
• Count>1, Interval = 0:表示在视频时长范围内按照Count值截图,如果
FrameType=intra,关键帧数小于Count数,则截图实际总数 <count。
• Count=1:截取单帧截图。</count。
 |
| Width | 截图宽 | 截图的宽。单位:像素,取值范围:[8,4096]。
⑦ 说明
Width、Height说明:
• 不设置宽和高,截图宽高与输入视频宽高相同。
• 如果只设置宽或者高,则没有设置的一边按照输入视频的画面比例
进行缩放,维持图片不变形。 |
| Height | 截图高 | 截图的高。单位:像素,取值范围:[8,4096]。 |

● WebVTT截图配置

除了普通截图需要配置的参数外,还需要配置 Format 和 SubOut 。

| API参数 | 控制台参数 | 描述 |
|--------|-------|--|
| Format | 文件格式 | 该参数说明截图需要将截图索引信息生成到VTT文件当中。
⑦ 说明
仅WebVTT截图需要设置该参数,且取值为:VTT。 |
| SubOut | | 仅WebVTT截图需要设置该参数。
示例:
{
"IsSptFrag":"true"
}
IsSptFrag:控制生成VTT文件时,截图图片的生成方式。false表示图片单张分
开存储,true表示将图片拼接成大图存储。 |

● 雪碧截图配置

? 说明

本节内容只对雪碧图的部分参数使用细节进行说明。更多信息,请参见<mark>雪碧图配置SpriteSnapshotConfig</mark>。

| API参数 | 控制台参数 | 描述 |
|-------------|---------|---|
| CellWidth | 小图宽 | 雪碧图中的小图宽和高设置参数,如果都不设置则小图宽高为普通截图的宽高, |
| CellHeight | 小图高 | 如果只设置其中一个,则另一边按照比例缩放。 |
| | | 是否保留构成雪碧图的原始图,即截图的普通截图。取值:delete(删除,即不
保留)、keep(保留)。 |
| KeepCellPic | 是否删除原始图 | ⑦ 说明
对于雪碧图截取,如无特殊需求,建议不保留构成雪碧图的原始图。 |
| | | 生成雪碧图的背景颜色。更多信息,请参见 <mark>颜色设置参数说明</mark> 。 |
| | | ⑦ 说明
暂不支持RGB参数值。 |
| Color | 雪碧图背景色 | 雪碧图参数示意如下: |
| | | |

6.4. 视频动图

视频动图服务可对指定视频中的一个区间进行内容截取,并生成动图文件,支持GIF和WebP两种格式。本文为您介绍视频动图功能的简介、使用方式、如何获取动图以及动图的模板与参数。

简介

动态图片(Dynamic Image)是指一组特定的静态图像以指定的频率切换而产生某种动态效果的图片。网络上常见的表现形式是GIF动图,其次是WebP。两者各有优缺点,用户可根据实际需要进行选择。

视频截动图,是对指定视频中的一个区间进行内容截取,并生成动图文件。

↓ 注意

- 纯音频且无任何图像信息流的媒体文件、源文件损坏或源文件封装信息异常,则可能无法生成截图文件。
- 截动图的过程是完全异步的,可通过视频动图完成接口获取视频动图完成情况。
- 截动图耗时与文件的大小、时长以及动图截取时长有关。
- 视频截动图, 会产生一定的费用, 请知晓。具体收费详情, 请参见媒资转码计费。

格式说明

视频动图目前支持GIF和WebP。

● GIF动图

GIF动图的特点:兼容性好,经常用来作网页装饰用。

● WebP动图

WebP动图的特点:文件体积相比GIF小很多,但缺点是部分浏览器不兼容。不兼容的浏览器有IE、Safari、iOS Safari3.2~13.7、KaiOS Browser等。

使用方式

● API截动图

视频截动图,通过提交截动图作业接口发起对指定视频的截动图任务。具体操作,请参见提交媒体动图作业。

● 控制台截动图

(-) 阿里云 华东2 (上海) ▼ Q、搜索文档、控制台、API、解决方案和资源 费用 工单 备 视频点播 / 工作流管理 / 添加工作流 审核设置 ← 添加工作流 配置管理 * 工作流名称 工作流 媒资管理配置 媒体处理配置 工作流拓扑结构 转码模版组 ビ 开始 0 ☆ 结束 -截图模板 媒体转码 动图模板 智能媒资审核 水印模板 智能媒资标签 工作流管理 智能媒资查重 回调设置 视频截图 视频动图 分发加速配置 数据中心 确认 取消 监控统计 用量查询

您可以通过工作流截取动图。登录<mark>视频点播控制台</mark>,在**配置管理>媒体处理配置>工作流管理**页面中,可以 添加视频动图作业。

动图获取

目前,视频点播支持通过以下两种方式获取视频的动图信息。

- 通过接收视频动图完成回调消息获取动图信息(推荐)。
- 视频动图完成后,可以通过查询动图列表接口来获取视频的动图信息。

? 说明

如果将动图设置为封面,可以通过获取视频信息接口获取封面动图链接。

动图模板

动图涉及到的参数相对较多,如果将动图参数都统一在提交动图任务时传入,不仅提高了动图功能接入门槛,更 降低了易用性。因此视频点播提供了动图模板,用户可先将相关参数配置成动图模板,在发起动图任务时指定动 图模板ID即可。

● API管理

点播提供动图模板管理接口,具体信息,请参见添加截图模板。

• 控制台管理

为方便用户管理与快速有效的实现动图任务,目前点播控制台已支持动图模板管理。你可以通过登录<mark>视频点播</mark> 控制台,在配置管理 > 媒体处理配置 > 动图模板页面中,可以进行动图模板管理。

| 审核管理 | | ^
[| 点播控制台 / 动图相 | 类板 | | | | | | | 新手引导 |
|---------|---|--------|------------------|--------------------------|------|------|-----------|--------------------------------------|---------|-------|------|
| 视频审核 | | l | 动图楔似 | | | | | | | | |
| 审核设置 | | | 1 帮助信息
动图模板可以 | L在工作流、上传媒体、媒体处理环节使用。希助文档 | | | | | | | |
| 配置管理 | | | 视频动图生产 | "按照视频转码标准计器。计器详情 | | | | | | | |
| 媒资管理配置 | ~ | | 添加动图模板 | | | | | | | | |
| 媒体处理配置 | ~ | Ь | 模板名称 | ID | 输出格式 | 顿率 | 尺寸(宽x高) | 截取方式 | 设置为默认封面 | 操作 | |
| 转码模板组 | | < | | | gif | 5fps | -рх х -рх | 載取方式: 按持续时长
开始时间: 0秒
持续时长: 15秒 | 关闭 | 编辑 删除 | |
| 截图模板 | | | | | | | | | | | |
| 动图模板 | | | | | | | | | | | |
| 水印模板 | | | | | | | | | | | |
| 工作流管理 | | | | | | | | | | | |
| 回调设置 | | | | | | | | | | | BS |
| DRM证书管理 | | | | | | | | | | | |
| 分发加速配置 | ~ | | | | | | | | | | |

动图参数

此处只介绍动图配置的部分参数使用细节,更多参数详情,请参见动图模板配置。

- 动图画面配置: Video
 - Width: 动图的宽。单位: 像素。取值范围: [128,4096]
 - Height:动图的高。单位:像素。取值范围: [128,4096]
 - Fps: 动图的帧率。取值范围: (0,60]
- 动图裁剪时间线配置: TimeSpan
 - Seek: 视频裁剪的开始时间。
 - Duration: 视频裁剪的持续时长。
 - End:视频裁剪需要丢弃的尾部时长。

? 说明

TimeSpan配置参数遵循以下格式:

- 格式1: sssss[.SSS]。取值范围: [0.000, 86399.999]示例: 12000.556
- 格式2: hh:mm:ss[.SSS]。取值范围: [00:00:00.000,23:59:59.999] 示例: 00:00:05.003
- Seek必填, Duration和End选填一个, 如果两者都填, 则以End为准。

6.5. 视频水印

视频点播提供视频添加水印功能,支持通过控制台和API管理水印,并在转码时添加水印。本文为您介绍添加水印 的使用说明、水印管理、水印类型、位置参数和使用场景。

简介

视频添加水印,即指在视频的编解码过程中将图片、文字等信息压制到视频流中合并输出一个带图文水印的新的 视频文件。通常在视频上添加企业、品牌Logo、电视台台标、用户ID或昵称等标志性信息,可用于宣示对视频的 版权归属或品牌宣传。

水印类型

视频点播当前支持三种水印类型:静态图片、动态图片和文字水印。

• 静图水印

目前仅支持PNG格式,图片位于视频特定位置从片头展示到片尾,也可设置图片在视频时长内的某段时间展示 (水印开始展示时间和水印结束展示时间)。

适合对视频版权宣示或品牌宣传。

• 动图水印

目前支持GIF动图、APNG动图和MOV格式的视频,动态图片可在视频特定的位置从片头到片尾循环播放,也可 设置动态图片在视频时长内的某段时间播放(水印开始展示时间和水印结束展示时间)。

适合对视频版权宣示或品牌宣传。

? 说明

- 目前控制台仅支持GIF格式, API支持GIF、APNG和MOV格式。
- 动态图片水印受文件扩展名限制,gif、apng和mov必须是小写,静态图片不受扩展名限制。
- 水印文件存储源站必须与需要添加水印的视频存储源站一致。例如:存在华东2源站的视频,使用的水印文件必须存储在华东2的源站上,不可跨区域且暂不支持跨存储源站,存储源站请参见存储管理。
- 文字水印

用户可将特定的一段或多段文字压制到视频画面上作为水印,视频点播当前可支持用户设置文字的字体、字 号、颜色、透明度、字体描边等效果,且支持不同视频压制不同的文字内容。动态文字水印详细请参见提交媒体 转码作业。

适合对视频版权宣示或品牌宣传,和短视频中文字动态替换。

? 说明

动图(MOV格式、APNG格式)水印和文字水印暂只支持通过水印接口进行管理,更多信息,请参见<mark>添加水</mark> 印接口。

使用说明

视频水印的使用流程如下:

- 1. 添加并管理水印信息,请参见下文水印管理。
- 2. 设置默认水印,具体操作,请参见控制台水印管理或API设置默认水印。
- 3. 创建转码模板并设置水印配置,具体操作,请参见控制台转码设置或API添加转码配置。
- 4. 使用刚创建好的水印模板上传视频,具体操作,请参见控制台媒资上传或媒体上传概述。

? 说明

- 转码模板需要关联多个水印,请参考添加转码配置接口。
- 原画转码模板不支持水印勾选,对应输出视频流不能添加水印。

水印管理

视频点播当前支持两种水印管理方式:视频点播水印管理控制台、水印管理接口。

• 控制台管理

> 文档版本: 20220713

视频点播控制台支持水印管理的相关操作,请参见水印管理。

? 说明

视频点播控制台暂不支持mov、apng格式的动图水印文件上传,可通过添加水印接口实现。

● API管理

添加水印模板可分为如下三个步骤:

- i. 调用CreateUploadAttachedMedia接口获取文件上传的路径以及临时授权。
- ii. 使用OSS上传指定的水印文件,请参见OSS文件上传。
- iii. 调用添加水印接口,添加水印信息。

水印位置参数

水印位置参数是用于控制水印在输出视频画面中显示的位置、尺寸,可通过视频点播控制台或添加水印接口设置水印参数。

⑦ 说明

以下仅介绍控制水印位置及尺寸的参数示意。更多信息,请参见水印配置 WatermarkConfig。

• 图片水印

可通过ReferPos、Dx、Dy、Width、Height控制图片在输出视频画面中的显示位置以及效果。

| API参数 | 控制台参数 | 描述 |
|----------|-------|--|
| ReferPos | 相对位置 | 水印相对输出视频画面的大概位置。 ・ 左上角: TopLeft ・ 左下角: BottomLeft ・ 右上角: TopRight ・ 右下角: BottomRight |
| Dx | 水平偏移 | 水印在输出视频画面上水平偏移距离。取值:
• 正整型,单位:像素;取值范围[8,4096]。
• 画面占比,单位:%;取值范围(0,1)。 |
| Dy | 垂直偏移 | 水印在输出视频画面上垂直偏移距离。取值:
• 正整型,单位:像素;取值范围[8,4096]。
• 画面占比,单位:%;取值范围(0,1)。 |
| Width | 水印宽 | 水印在输出视频画面中显示的宽。取值:
• 正整型,单位:像素;取值范围[8,4096]。
• 画面占比,单位:%;取值范围(0,1)。 |

| API参数 | 控制台参数 | 描述 |
|----------|-------|--|
| Height | 水印高 | 水印在输出视频画面中显示的高。取值:
• 正整型,单位:像素;取值范围[8,4096]。
• 画面占比,单位:%;取值范围(0,1)。 |
| | | 水印时间线用于控制图片水印在视频画面上的开始显示时间以及持续显示时间,
可以通过多组水印配置控制图片水印在视频画面的动态显示效果。更多信息,请
参见 <mark>水印时间线 Timeline</mark> 。 |
| Timeline | 时间线 | ↓ 注意
文字水印暂不支持该参数。 |

示意图

如需水印居中显示,则可按照画面占比设置Dx=0.5、Dy=0.5参数调整(水印的偏移位置为视频输出画面宽高的50%)。



• 文字水印

文字水印暂时只支持相对输出视频画面的左上角开始偏移,偏移参数为Top、Left,暂只支持像素取值,不支持 画面占比类参数取值。

| API参数 | 控制台参数 | 描述 |
|-------|-------|---------------------------------------|
| Тор | 上边距 | 文字水印距离输出视频画面的上边距离。单位:像素,取值范围[8,4096]。 |
| Left | 左边距 | 文字水印距离输出视频画面的左边距离。单位:像素,取值范围[8,4096]。 |

示意图

文字水印



● 计算方式

图片水印的Dx、Dy、Width、Height参数支持两种计算模式,文字水印的Top和Left只支持正整型取值,具体如下:

○ 正整型取值, 单位: 像素, 取值范围[8,4096]。

↓ 注意

如果设置的参数超过视频输出画面的大小,则可能导致水印局部显示或者不显示。

例如:视频输出画面为640×360,相对位置为TopLeft,Dx=4000,Dy=4000,则输出画面上看不到水印(文字水印亦是如此)。

○ 画面占比取值:相对输出视频分辨率宽度、高度的占比值,取值范围(0,1),精确到小数点后四位。

↓ 注意

如果输出画面比例多变不可控,建议按照画面占比设置水印位置、尺寸相关参数。

例如: 0.9999。计算示例: Width/视频宽 = 0.1; Height/视频高 = 0.06; Dx/视频宽 = 0.02; Dy/视频高 = 0.03。

- 默认值
 - 水印宽、高都不设置:水印宽=视频输出画面宽×0.12,水印高按照水印原图比例等比缩放。
 - 宽、高设置其中一个:设置的一边按照设置的参数输出,另一边按照水印原图比例等比缩放。例如: Widt h=44,原图宽高比例是2:1,则Height=22。
 - 。 宽、高都设置:则水印尺寸按照设置的参数输出。
 - Dx、Dy不设置时,则默认取值为0。

常用场景

文字水印动态替换

通常在短视频场景中,用户上传分享的视频在转码输出后,都会带有一个图片水印(通常是当前产品Logo)和文字水印(例如:视频原创者的ID或者昵称),用于宣示视频版权归属。

而对于不同用户上传分享的视频,用户的ID或者昵称并不相同,这就需要在对不同的视频转码时使用不同文字内容 作为水印,视频点播支持在转码模板设置关联多水印后,通过提交媒体作接口发起转码并进行文字水印的内容替 换,则可实现不同用户视频,文字水印内容不同的效果。更多信息,请参见提交媒体转码作业。

效果示例

图片水印 + 文字水印,不同用户文字水印的内容不同。





图片水印动态替换

在短视频场景中,为了加强视频安全,用户上传分享的视频在转码输出后,都会带有一个动态的图片水印(通常是用户的Logo),用于宣示视频版权归属。

而对于不同用户上传分享的视频,这就需要对不同的视频转码时使用不同图片作为水印,视频点播支持使用获取 辅助媒资上传地址和凭证接口,获取文件上传的路径以及临时授权。使用OSS上传指定的水印文件,获取图片水印 文件的OSS地址,通过提交媒体转码接口发起转码并进行图片水印文件的OSS地址动态替换,则可实现不同用户视 频,图片水印不同的效果。更多信息,请参见获取辅助媒资上传地址和凭证、OSS文件上传、提交媒体转码作业。

效果示例

图片水印,不同用户图片水印不同。



实现方式 Java: 提交转码作业。 Python: 提交转码作业。

PHP:提交转码作业。

.NET:提交转码作业。

Node.js: 提交转码作业。

GO: 提交转码作业。

C/C++: 提交转码作业。

6.6. 工作流

点播提供了一系列媒体处理的能力,如转码、截图、智能审核等。为提高媒体处理的易用性,视频点播提供了工作流能力,支持在控制台自定义媒体处理的流程。本文为您介绍工作流的基础架构、配置方式和使用方式。

简介

工作流旨在将大部分媒体处理功能流程化、实例化,用户可提前构建自定义处理流程,并将制定视频按照特定工作流程进行处理,避免用户重复、繁琐的调用接口,提供一站式的媒体处理服务。同时,工作流提供了丰富的条件判断机制,可自定义下一节点条件触发条件,便于灵活组建场景化处理流程。

基础架构

工作流示例

• 审核+转码串行

此流程表示视频进入到点播后,先进行审核操作,然后进行转码操作的串行化处理。具体审核完成后如何进行 转码,请参见下文<mark>审核+转码串行</mark>。

| ≌ 开始 | \longrightarrow | • 审核 | S 转码 | 0 → | ☆ 结束 |
|------|-------------------|------|------|-----|------|
| | | | | | |

• 审核+转码并行

此操作表示视频进入到点播后,同时进行审核和转码的操作。

| 区 开始 —— | S 转码 | 0 |
☆ 结束 |
|---------|------|---|----------|
| | ⑤ 审核 | 0 | |

• 分发源片+转码并行

该流程会将源文件作为一路播放流,同时会按照设置的转码节点进行其他码流输出。



处理流程

> 文档版本: 20220713

工作流的处理流程如下:



准备工作:用户在控制台配置工作流。

- 1. 用户获取上传凭证,同时指定工作流。点播服务返回上传凭证。
- 2. 用户发起视频上传。
- 3. 上传完成后自动触发工作流。
- 4. 视频点播服务的工作流引擎根据工作流配置,自动进行媒体处理、视频AI等操作。

配置方式

目前,您可以登录<mark>视频点播控制台</mark>,在点播控制台左侧的导航栏选择**配置管理**,在**媒体处理配置 > 工作流管** 理页面配置工作流。

使用方式

您可以在获取上传凭证时指定工作流。

↓ 注意

工作流方式和原模板组上传方式只能使用一种,并且使用工作流ID上传时,请注意工作流ID的合法性。更多信息,请参见获取视频上传地址和凭证接口。

审核+转码串行

以审核+转码模式(先进行审核,审核通过后,再进行转码)为例,具体操作如下:

1. 登录视频点播控制台,在点播控制台左侧的导航栏选择配置管理。

2. 单击媒体处理配置 > 工作流管理,进入工作流管理页面。

3. 单击添加工作流,输入名称。

4. 单击开始右边的+按钮,此处以审核节点为例,添加智能媒资审核节点。

| ≌ 开始 | C | ☆ 结束 |
|------|-------|------|
| 媒 | 体转码 | |
| 智 | 能媒资审核 | |
| 智 | 能媒资查重 | |
| 视 | 频截图 | |
| 视 | 频动图 | |

5. 添加完成后,单击编辑按钮,选择审核模板。

| | ≌ 开始 | | 智能媒资审核 ① ① 日 | | ☆ 结束 |
|--|------|--|---|--|------|
|--|------|--|---|--|------|

- 6. 单击审核节点右边的+按钮,添加转码节点,表示转码节点在审核节点之后。
- 7. 添加完成后,单击编辑按钮,配置执行条件。

↓ 注意

此时多了一个选项:执行条件表示父节点满足何种状态,才能触发该节点。用户可根据需要进行触发条件的选择。

| 媒体转码 | | × |
|---------|------------------------------|--------|
| * 节点名称 | Territory, etc. and an and a | |
| 执行条件 | ○ 任意条件 | |
| | ● 父节点审核通过 | |
| | ○ 父节点审核操作失败 | |
| | ○ 父节点视频违规屏蔽 | |
| * 转码模版组 | | \sim |
| | | 确定取消 |

其他执行条件介绍如下表所示:

| 情况描述 | 执行条件 | 条件描述 |
|--------|-----------|--|
| 父节点为审核 | 任意条件 | 任何条件该节点均可执行。 |
| | 父节点审核通过 | 父节点审核通过才可执行该节点。 |
| | 父节点审核操作失败 | 父节点审核失败才可执行该节点,常见于源文件存在问题等,不常
见。 |
| | 父节点视频违规屏蔽 | 父节点审核完成,并且确认视频违规,并已经自动屏蔽。常用与违规
视频只转部分低码率清晰度用于人工审核等场景。 |
| 父节点为转码 | 任意条件 | 任何条件该节点均可执行。 |
| | 父节点转码成功 | 父节点转码成功后才可执行此节点。 |
| | 父节点转码失败 | 父节点转码失败才可执行此节点,不常用。 |

8. 在完成工作流拓扑结构后,单击确认,会生成一个工作流ID,记录此工作流ID,用于后面的上传指定。

| 名称 | ID | 创建时间↓ | 操作 |
|----|--|---------------------|---------|
| - | - Anno 2011 - Conta Cont | 2020-11-10 18:11:52 | 详情 删除 |

审核+转码并行

如果需要审核和转码并行,可以在开始节点右边的按钮添加两个子节点,即审核、转码,可以构造如下工作流。



○ 注意

需要点击节点编辑,并选择所要使用的操作模板(审核模板、转码模板组等)。

6.7. 推荐模板配置

本文主要介绍一些媒体处理相关的配置经验,为用户提供参考。

转码模板配置

视频参数:

● 禁用视频

勾选此选项,则输出文件不包含视频信息,常用与生成一路纯音频,如电台等场景。

- 码率、分辨率
 - 码率(Bitrate)是指视频文件在单位时间内使用的数据流量,也叫码流或码流率,是视频编码中画面质量控制最重要的部分。量度单位为"比特每秒"(bit/s或bps),常使用Kbps(每秒多少千个比特)或Mbps。

一般来说同样分辨率下,视频文件的码率越大,压缩比就越小,画面质量就越高。码率越大,说明单位时间 内取样率越大,数据流精度就越高,处理出来的文件就越接近原始文件,图像质量越好,画质越清晰,要求 播放设备的解码能力也越高。

○ 分辨率(Resolution)用来描述视频对细节的分辨能力,通常表示为每一个方向上的像素数量,比如 1280×720等。

分辨率决定了视频画面细节的精细程度。通常情况下,视频的分辨率越高,所包含的像素就越多,画面就越 清晰。分辨率是决定码率的主要因素,不同的分辨率要采用不同的码率。

这体而言,视频的分辨率越高,所要求的码率也越大,但并不总是如此,不同分辨率都有合理的码率选择范围。所谓"合理的范围"指的是,如果低于这个范围,视频画面质量会很差(即很模糊);如果高于这个范围,画面提升有限或无提升,则没有必要,对于网络流量以及存储空间是一种浪费。

○ 对于不同的清晰度输出, 码率、分辨率有不同的推荐值:

| 清晰度 | 推荐码率 | 推荐分辨率 | 分辨率区间 |
|-----|------|-----------|---------------------|
| 流畅 | 400 | 640×360 | 128×128~640×360 |
| 标清 | 900 | 960×540 | 641×361~960×540 |
| 高清 | 1500 | 1280×720 | 961×541~1280×720 |
| 超清 | 3000 | 1920×1080 | 1281×721~1920×1080 |
| 2К | 3500 | 2560×1440 | 1920×1080~2560×1440 |
| 4К | 6000 | 2560×1440 | 2560×1440~3840×2160 |

? 说明

- 分辨率的宽和高可不设置。如果仅设置宽,高度会按照视频原始比例进行缩放。如果宽高都不设置,则会按照原始视频宽高值进行处理。
- 码率单位: Kbps。
- 分辨率 (宽×高) 单位: px。
- 帧率

帧率(Frame Rate)是单位时间内视频显示帧数的量度单位,也就是每秒钟刷新的图片的帧数,量度单位为 "每秒显示帧数"(Frame Per Second,FPS)或"赫兹"。

高的帧率可以得到更流畅、更逼真的画面效果。一般来说25~30fps就可接受,提升至60fps则可以明显提升交 互感和逼真感,但一般来说超过75fps就不容易察觉到有明显的流畅度提升了。如果帧率超过屏幕刷新率只会浪 费图形处理的能力,因为显示设备不能以这么快的速度更新,这样超过刷新率的帧率就浪费掉了。在分辨率不 变的情况下,帧率越高,则对显卡的处理能力要求越高。

点播默认推荐值: 25 FPS。

• 关键帧最大间隔(一个GOP内的帧数)

GOP(Group of Pictures)是一组以 MPEG 编码的影片或视讯串流内部的连续图像,以 I 帧开头,到下一个 I 帧 结束。一个 GOP 包含如下图像类型:

- Ⅰ帧(Intra Coded Picture):又称帧内编码帧,为关键帧,是一种自带全部信息的独立帧,无需参考其他图像便可独立进行解码,可以简单理解为一张静态画面。视频序列中的第一个帧始终都是Ⅰ帧,每个 GOP 由Ⅰ帧 开始。
- P 帧(Predictive Coded Picture): 又称帧间预测编码帧,需要参考前面的I帧才能进行编码。表示的是当前 帧画面与前一帧(前一帧可能是I帧也可能是P帧)的差别。解码时需要用之前缓存的画面叠加上本帧定义的 差别,生成最终画面。与帧相比,P帧通常占用更少的数据位,但不足是,由于P帧对前面的P和I参考帧有着 复杂的依耐性,因此对传输错误非常敏感。

B 帧(Bidirectionally Predictive Coded Pictures): 又称双向预测编码帧,也就是B帧记录的是本帧与前后帧的差别。也就是说要解码B帧,不仅要取得之前的缓存画面,还要解码之后的画面,通过前后画面的与本帧数据的叠加取得最终的画面。B帧压缩率高,但是对解码性能要求较高。

GOP值表示关键帧的间隔,也就是两个IDR帧之间的距离,一个帧组的最大帧数。一般而言,每一秒视频至少需 要使用 1 个关键帧。增加关键帧个数可改善视频质量,但会同时增加带宽和网络负载。GOP值(帧数)除以帧 率即为时间间隔。

阿里云视频点播推荐的GOP值为250帧,帧率为25 FPS,则时间间隔为250/25=10秒。

音频参数:

● 禁用音频

勾选此选项,则输出文件不包含音频,常用与需要把原视频声音禁用场景。

• 采样率

采样率(Sample Rate),也称为采样速度或者采样频率,定义了每秒从连续信号中提取并组成离散信号的采样 个数,单位为赫兹(HZ)。采样率是指将模拟信号转换成数字信号时的采样频率,也就是单位时间内采样多少 点,采样频率越高声音的还原就越真实越自然。

点播推荐值: 44100 HZ。

码率

| 音质 | 推荐码率 |
|------|------|
| 普通音质 | 128 |
| 高音质 | 320 |

? 说明

码率单位: Kbps。音频码率支持输入范围8Kbps~1000Kbps。

• 声道数

点播推荐值:2。

截图模板配置

● 普通截图

普通截图参数设置可按照需求进行设置,参数详细请参考普通截图配置SnapshotConfig。如需要保证截图宽高和 原视频保持一致,则建议不设置Width、Height参数。

● 雪碧图截图

雪碧图参数设置可按照需求进行设置,参数详细请参考<mark>雪碧图配置SpriteSnapshotConfig</mark>。其中参数KeepCellPic如 无特殊需要,建议设置为delete,即不保留生成雪碧图的原始小图文件。

水印配置

• 图片水印

如需要尽量保证水印在不同分辨率输出视频的显示效果,建议Width、Height、Dx、Dy参数按照画面占比设置。更多信息,请参见如果水印类型为Image。

• 文字水印

> 文档版本: 20220713

文字水印的各参数取值详细,请参见如果水印类型为Text,除文字水印内容外,其他参数如无特殊需求可参考默 认取值。

6.8. 多码率自适应

多码率自适应可将指定的音视频文件流统一打包生成一个自适应码流文件,让播放器可根据终端网络带宽环境。 本文为您介绍多码率自适应、使用说明、模板管理、模板参数和示例。

简介

多码率自适应即指将指定的音视频文件流统一打包生成一个自适应码流文件,该自适应码流文件包含不同音视频 文件流的码率、分辨率等信息,播放器则根据网络带宽环境自动选择最适合当前带宽环境的码流播放。

? 说明

目前使用比较广的多码率自适应输出格式为HLS和DASH,视频点播目前仅支持HLS多码率自适应生成。

优点

相比较传统的单码率播放,自适应码率文件能够让播放器可根据终端网络带宽环境,选择最适合当前带宽环境和 终端设备的码流播放,从而提升播放体验。

使用说明

 登录视频点播控制台,在点播控制台左侧的导航栏选择配置管理,单击媒体处理配置>转码模板组,在转 码模板组页面创建多码率自适应的打包模板。

⑦ 说明

视频点播支持视频多码率打包模板和字幕多轨打包生成模板,用于生成可切换不同分辨率和不同语言字幕的多码率自适应文件。

2. 使用创建好的多码率自适应打包模板上传视频。

| 添加音/视频 | | | | | | : |
|-----------|---|------------|-------------|-------------|------------------|----|
| 上传方式 | ◉ 本地上传 | | | | | |
| | URL拉取 选择URL拉取方式时,若UR | 中不包含文件扩展名, | 为提高拉取成功率,请填 | 与扩展名了解URL批量 | 拉取上传 | |
| 予储地址: | | | ~ | | | |
| 添加音/视频 | 用转码模版组处理 💙 | 不转码 | ◇ 未分类 ◇ | | | |
| 音/视频名称 | | 格式 | 大小 | 分类 | 转码模板组/工作流模板 | 操作 |
| (a) = (a) | | mp4 | 504 KB | 未分类 🗸 | 自适应1 ヘ | 删除 |
| | | | | | 自适应1 | ~ |
| | | | | | The shift of the | 肖 |
| | | | | | 不转码 | |

3. 多码率自适应码率处理完成后,查看视频播放列表。

模板管理

多码率打包模板当前支持控制台和API两种管理方式。

- 控制台管理
 视频点播控制台支持多码率打包模板管理的相关操作。具体操作,请参见转码设置。
- 见<u>然然通过的百丈的夕时中的已</u>快恢告连的伯大课作。
- API管理

多码率打包模板API管理相关操作。详细参见转码模板接口。

模板类型

视频点播目前支持多码率自适应模板分两种类型:视频多码率和字幕多轨打包模板。更多信息,请参见转码模板配置TranscodeTemplate。

⑦ 说明

字幕打包模板不能单独创建使用,必须与视频打包模板一同使用。

- 视频流打包模板
 - 视频打包模板用于将视频转码输出成HLS格式码流信息构建到自适应多码率文件当中,可供播放器进行分辨率 选择和切换。
- 字幕打包模板
 - > 字幕打包模板用于将视频多语言字幕文件的信息构建到自适应多码率文件中,可用播放器进行语言字幕选择 和切换。

? 说明

- 字幕文件仅支持VTT格式。
- 字幕打包如果需要动态替换模板中的字幕文件,可以调用SubmitTranscodeJobs接口进行参数替换。

模板参数

本文档仅列举打包模板相对特异的配置参数进行说明,更多详细参数请参见基本数据类型。

打包模板具体的参数设置区别于不同的打包模板类型,参数设置也不尽相同。

视频打包参数:

| API参数 | 控制台参数 | 描述 |
|-------------------|-------|---|
| PackageType 打包类型 | | 自适应多码率打包类型,输出HLS或DASH格式。取值:HLSPackage(HLS打
包)、DASHPackage(DASH打包)。 |
| | 打包类型 | ⑦ 说明
目前仅支持HLSPackage。 |
| API参数 | 控制台参数 | 描述 |
|-----------|-------|--|
| BandWidth | 带宽阈值 | 带宽阈值,提供给播放器根据当前网络带宽环境判断需要选择播放的码流,单
位:bps。 |
| | | ⑦ 说明
该参数为视频打包模板参数。 |
| | | |

字幕打包参数:

| 参数 | 控制台参数 | 描述 |
|-----------------|-------|--|
| Language | 语言 | 字幕语言类型,例如:ja(日文)、en-US(英文)等,需要符合RFC5646描述
规范。
⑦ 说明
该参数为字幕打包模板参数。 |
| Name | 显示名称 | 字幕语言类型在播放器上的显示名称,例如: 中文、英文 等。 |
| Format | 字幕格式 | 字幕文件的后缀名,目前HLS打包仅支持vtt格式。
示例: subtitle.vtt |
| SubtitleUrlList | 字幕地址 | 完整的OSS地址,不支持CDN域名地址和HTTPS地址,且字幕文件仅支持存储在点
播分配的系统Bucket上。
示例:
<pre>http://example-bucket-****.oss-cn-
shanghai.aliyuncs.com/subtitles/4dba87c2-a787-42cd-8328-
2369aeb8****-cn.vtt</pre> ⑦ 说明
该参数为字幕打包模板参数。 |

内容示例

| #EXTMBU |
|--|
| #EXT-X-MEDIA:TYPE=SUBTITLES,GROUP-ID="Subtitles",NAME="英文],DEFAULT=N0,AUTOSELECT=N0,FORCED=N0,LANGUAGE="en-US",URI="subtitles/en-us/b76bebcd0d5dc2c000cae32382b8d446.m3u8" |
| #EXT-X-MEDIA:TYPE=SUBTITLES,GR0 <u>UP-ID="Subtitle</u> s",NAME= <mark>"中文]</mark> ,DEFAULT=NO,AUTOSELECT=NO,FORCED=NO,LANGUAGE="cn",URI="subtitles/cn/b76bebcd0d5dc2c000cae32382b8d446.m3u8" |
| #EXT-X-STREAM-INF:PROGRAM-ID=1 BANDWIDTH=400000 CODECS="mp4a.40.2, avc1.64001e" RESOLUTION=640x360 SUBTITLES="Subtitles" |
| video/7f6b284b57f345868381effc7ddeb1fc-31bda29c75e4691da4c7c95eadb64b4d-video-f <u>d.m3u8</u> |
| #EXT-X-STREAM-INF:PROGRAM-ID=1,BANDWIDTH=900000 CODECS="mp4a.40.2,avc1.64001f" RESOLUTION=960x540 SUBTITLES="Subtitles" |
| video/7f6b284b57f345868381effc7ddeb1fc-e4d1340513727c223383a4245ecb8335-video-ld.m3u8 |
| |

6.9. 转码生成文件名称

视频上传到点播转码后生成的文件名称(即文件的相对路径),会遵循一定规则,本文档主要介绍转码生成文件 名称的默认规则,以及如何自定义。

文件名的定义

文件名由可变字符串、固定字符串组成,其中可变字符串主要是指点播提供的特定通配符,可进行动态替换,而 固定字符串不能动态替换。

? 说明

该文件名称定义规则暂不适用HLS和DASH打包的输出文件名称。

可变字符串

可变字符串即通配符,是指具体取值的替换规则,只需要设置对应的通配符,则点播会在生成转码文件名称时,获取具体的取值去替换通配符,当前点播支持的通配符如下:

- 视频ID: {MediaId} ,局部唯一性,一个视频可生成多路转码输出。
- 转码任务ID: {JobId} ,完全唯一性,可唯一标记一个转码文件。
- 播放清晰度标记: {PlayDefinition} , 不具有唯一性。
 - ? 说明

播放清晰度标记为获取视频播放地址接口中返回的 Definition 字段的取值,且使用英文小写形式。

• 固定字符串

固定字符串可代表具体的业务信息。例如:watermark(表示有水印输出)、encrypt(表示加密输出)等,点播提供的默认规则中对于转码加密输出会具备encrypt-stream关键字,也可通过自定义规则配置具体各自业务特性的固定字符串。

示例: {MediaId}/{JobId}-watermark-sd 、 {MediaId}/encrypt-hd , 具备watermark、encrypt关键可表

文件名的唯一性

为避免转码输出文件相互覆盖,点播要求配置的转码输出的文件名称必须具备唯一性。

• 唯一性示例:

{MediaId}/{JobId}: {Mediald} 仅具备局部唯一性, {JobId} 具备全局唯一性,从而确保整个文件名称的唯一 性。

• 非唯一性示例:

{MediaId}/test : {Mediald} 仅具备局部唯一性, test代表某的业务信息。当此时需要转码输出两个分辨

率、码率等转码参数不同的MP4文件,对于同一个视频两个MP4格式的转码输出文件,名称就会相同,且会出现 文件相互覆盖的情况,最终导致只有一个MP4文件。

默认规则

点播转码会对文件输出名称提供一套默认的生成规则,具体如下:

? 说明

以下随机字符串主要是为了保证转码输出文件名称的唯一性。

• 非加密输出

形式为:

{MediaId}/[**随机字符串**]-{PlayDefinition}

生成示例:

99f4f18c560c4f35459ef71tr/1b1196c767003c3dba0543055-40023f3f33e648df975f4dfc4c-sd

• 加密输出

形式为:

{MediaId}/[随机字符串]-{PlayDefinition}-encrypt-stream

生成示例:

```
99f4f18c560c4f35459ef71tr/1b1196c767003c3dba0543055-40023f3f33e648df975f4dfc4c-sd-encrypt-stre am
```

? 说明

```
encrypt-stream关键字表示当前转码输出文件为转码加密输出, 仅对HLS加密生效(私有加密/标准加密)。
```

自定义规则

对于某些特定的业务场景,如用户为了降低视频整体的播放响应时间,希望根据某些可行的规则直接拼接处播放 地址,例如:规则为 {MediaId}/{PlayDefinition} ,则可直接通过 MediaId +不同的清晰度标记 来拼接对应的 文件路径,而无需通过播放接口获取播放列表。由此,点播转码支持用户自定义转码文件名称规则,转码会根据 规则生成具体的名称。

规则限制

- 自定义转码文件名称只支持特定的通配符(可变字符串)、英文字符串、数字、连接号(-)、下划线(_), 整个自定义规则总长度不能超过128个字符。
- 自定义转码文件名称必须以通配符 {Mediald} 开始,可首先保证转码文件名称局部唯一,防止不同视频转码文件名称相互影响。
- 自定义转码文件名称不需配置文件后缀名,因为点播转码将自动补齐后缀名,例如:名称规则为 {Mediald}/{JobId}-watermark-{PlayDefinition},格式 mp4,那么点播转码会在生成实际文件名称时自动 带上mp4后缀。

```
示例: 99f4f18c560c4f35459ef71tr/1b1196c767003c3dba0543055-watermark-sd.mp4。
```

• 自定义转码文件名称规则必须确保转码输出文件名称具备唯一性,防止文件覆盖。

配置方式

当前仅支持通过API自定义转码文件名称,详细请参考添加转码配置,参数详情请参考转码模板配置

TranscodeTemplate的 TranscodeFileRegular 参数。

配置示例

```
public static JSONArray buildTranscodeTemplateList() {
    JSONObject transcodeTemplate = new JSONObject();
    //fc义转码输出文件名称
    transcodeTemplate.put("TranscodeFileRegular", "{MediaId}/{JobId}-watermark-{PlayDefiniti
on}");
    //模板名称
    transcodeTemplate.put("TemplateName", "testtemplate");
    //清晰度
    transcodeTemplate.put("Definition", "LD");
    ......
    //视频转码配置、音频转码配置、输出格式等参数配置
    ......
    JSONArray transcodeTemplateList = new JSONArray();
    transcodeTemplateList.add(transcodeTemplate);
    return transcodeTemplateList;
}
```

更多配置详情请参考转码模板中添加转码模板组。

6.10. 媒体处理FAQ

本文介绍媒体处理过程中的常见问题及处理方法。

只想上传不想转码如何操作?

"不转码即分发"是一个特殊的转码模板,该模板不会进行实际转码,会将用户上传的视频原片信息写入播放列 表,用户在调用获取播放信息GetPlayInfo接口时,可以获取到原片的播放地址,常用与短视频、上传后需要快速 播放等场景,客户在上传时选择此模板组即可。

目前仅以下格式支持不转码直接播放: MP4、FLV、M3U8、MP3、WEBM。

如果想让视频自动按照原始比例缩放如何设置?

设置模板输出时, 仅设置宽或者高, 则另一边会自动按照原始比例进行适应。

需要在转码同时抽取一路音频如何设置?

- 方法1: 新建一路转码输出, 封装格式为mp4或hls, 勾选"禁用视频"选项。
- 方法2: 新建一路转码输出, 封装格式为mp3。

条件转码通常如何使用?

条件转码多用于高清晰度设置,如用户设置了一个4K清晰度,但是此时用户上传的视频没有达到转码模板所设置的码率(或分辨率),此时用户可进行选择:

- 不进行该规格的视频转码。
- 按照输入视频码率(或分辨率)进行转码。

7.媒体审核

7.1. 概述

视频点播提供媒体审核服务,包括智能审核、人工审核、以及安全审核配置。本文为您介绍点播审核的优势、功能、应用场景和准备工作。

简介

上传的媒体资源可能有内容违规风险,点播服务提供了支持视频、音频、图片、文本等全方位的媒体审核,包 括智能审核、人工审核、以及安全审核配置等,通过自动化、智能审核媒资库资源,极大降低色情、暴恐、涉政 等内容违规风险,降低人工审核成本。

点播审核的优势

- 基本审核能力与绿网内容安全保持一致,提供图片、视频、文本、语音等形式内容违规检测,覆盖暴恐、涉政、涉黄、广告、辱骂、不良场景等风险检测能力。
- 点播媒体审核以媒体资源的维度将视频、音频、图片(截图、封面等)、文本(标题、简介、标签等)的单项 审核做了进一步聚合处理,只需一次调用就可完整处理所有音视图文资源,使用更加简捷、高效。
- 点播支持工作流、按照自定义配置模板自动审核,媒体上传完成可自动触发审核,并可实时回调审核结果。

产品功能

• 智能审核

智能审核服务基于海量标注数据和深度学习算法实现,从语音、文字、视觉等多维度精准识别视频、封面、标题等违禁内容,包括色情、暴恐、政治敏感等多个功能模块,可应用于短视频平台、传媒审核等场景,大大提 高视频的审核效率。

智能审核开发指南详情,请参见:智能审核。

• 人工审核

视频点播服务提供人工审核服务,用户可以提前或及时下线不符合条件的视频,避免或减少对外发布后造成的不良影响。

人工审核开发指南详情,请参见:人工审核。

● 安全审核IP

视频云点播服务提供安全审核IP服务,对于审核中(Checking)或屏蔽(Blocked)状态的视频,保证只有来自 审核安全IP的请求才可以播放。

安全审核IP开发指南详情,请参见:审核安全IP。

应用场景

- 短视频、海量UGC上传的视频,通过视频智能审核技术,快速检测违规内容,对于疑似违规视频进行人工复 审,可大大提高审核效率。
- 点播平台和传媒行业,通过视频智能审核技术可以高效识别点播视频中的涉暴、涉政类镜头,紧跟时政最新动态,严控违规内容。

使用方式

- 1. 您已经注册了阿里云账号并完成账号实名认证。注册地址请单击阿里云官网。注册指引请参见注册阿里云账 号。实名认证指引请参见个人实名认证或企业实名认证。
- 2. 调用服务端接口需要使用AccessKey完成身份验证,请提前获取AccessKey。获取方法请参见获取AccessKey。

? 说明

您也可以在RAM访问控制台创建RAM用户,并授予VOD权限(如 AliyunVODFullAccess)后进行相应 操作。具体操作,请参见RAM用户访问。

- 3. 登录视频点播控制台,在点播控制台左侧的导航栏选择审核管理,单击审核设置>审核配置,进入审核配置页面。
 - i. 设置审核流程,分为先审后发和先发后审两种。默认为先发后审,您可以根据自己的业务需要,来切换 至先审后发,即上传后的视频经过转码、分发加速等处理还是无法访问的,需要审核通过之后才可以播放。

| 点播控制台 | 点攝控制台 / 审核设置 |
|-------|--|
| 概览 | 审核设置 |
| 媒资库 | 审核配置 审核模板 |
| 音/视频 | 审核流程 ∠ 修改配置 |
| 图片 | 审核流程 先发后审 |
| 短视频素材 | 视频转码成功后默认标记为"正常"状态,可直接播放,在视频审核中进行"屏蔽"操作后不能播放 |
| 制作中心 | 智能审核 🛛 🖉 修改配置 |
| 视频剪辑 | 机审模板 预置模版 |
| 审核管理 | 机审模板ID VOD-0003-000001 |
| 视频审核 | |
| 审核设置 | |

ii. 设置智能审核,可以设置或添加智能审核默认模版。

| 智能审核 | ∠ 修改配置 | 智能审核配置 | | | × |
|--------|-----------------|-------------|---------|----------|--------------|
| 机审模板 | 预置模版 | * 机审模板 | 预置模版 | ✓ +添加新模板 | R |
| 机审模板ID | VOD-0003-000001 | D STT DE LA | INE DOM | | |
| | | | | | 确定 取消 |

 在点播控制台左侧的导航栏选择配置管理,单击媒体处理配置>回调设置,进入回调设置页面,设置回调 地址,勾选回调事件。

| 点播控制台 / 回调设置 | | |
|----------------------------------|--------|---|
| 回调设置 | | |
| ① 视频处理 (如上传、转码) 每个环节完成后,通过配置的回调地 | 回调设置 | × |
| 回调设置 🗾 🖉 修改设置 | * 回调方式 | ○ HTTP请求 |
| 回调方式 MNS队列 | | ○ MNS\$A\$PJ |
| 队列名称 | 回调URL | 请输入回调URL |
| Endpoint | | 当产生回调事件时,点播服务端会向该URL发起HTTP POST请求 |
| | 区域 | 请选择队列所在的区域 💙 |
| 回调事件 视频上传完成 | 队列 | 请输入回调URL V |
| 视频和处理完成 | 回调事件 | 视频上传完成 |
| □ 人工审核完成 | HIGTL | 全部清晰度转码完成 |
| 剪辑合成完成 | | □ 视频AI处理完成 |
| 图片上传完成
超休 <u>制</u> 除完成 | | □ 人工审核完成 □ URL上传视频完成 |
| 媒资基础信息变更完成 | | □ 剪辑合成完成 □ 录制视频合成开始 |
| 您选择的回调事件处理完成后,点播服务会进行事 | | □ 图片上传完成 |
| | | 媒体删除完成 |
| | | ◎ 媒资基础信息变更完成 |
| | | 認这裡的回调事件处理完成后,点播服务会进行事件通知,事件通知的实现方式请参考 事件通知
开发指南 |
| | | 湖 定 取消 |

- 对于人工审核需要开启人工审核完成,接收人工审核完成事件通知回调。
- 对于智能审核需要开启**音视频分析完成**,接收音视频分析完成事件通知回调后,才能提交智能审核作业。
- 对于智能审核需要开启视频AI处理完成,所有AI处理都共用此开关,接收智能审核完成事件通知回调。

7.2. 智能审核

智能审核服务可对视频、音频、图片、封面、标题等媒体文件内容进行多维度精准识别,帮助您检测媒体文件中的风险或违规内容。智能审核服务可支持配置审核功能模板、配置自定义机审标准,以适应不同业务场景下的审核要求。本文为您介绍智能审核的功能、使用方式和应用场景。

功能介绍

视频智能审核服务基于海量标注数据和深度学习算法实现,从语音、文字、视觉等多维度精准识别视频、封面、标题等媒体信息中包含的违禁内容,支持涉黄识别、暴恐涉政识别、广告及二维码识别、Logo识别、不良场景识别、语音垃圾识别等多个功能模块。

具体功能如下:

| 功能名称 | 描述 | 检测结果分类 |
|------|-------------------------|----------|
| 涉黄识别 | 检测媒体文件中是否包含色情、性感内
容。 | 正常、色情、性感 |

| 功能名称 | 描述 | 检测结果分类 |
|--------|--|--|
| 暴恐涉政识别 | 检测媒体文件中是否包含暴恐或涉政类
内容。 | 正常、血腥、爆炸烟光、特殊装束、特
殊标识、武器、涉政、打斗、聚众、游
行、车祸现场、旗帜、地标 |
| 图文违规识别 | 检测视频、图片中是否包含广告和文字
违规信息。 | 正常、文字含涉政内容、文字含涉黄内
容、文字含辱骂内容、文字含暴恐内
容、文字含违禁内容、文字含其他垃圾
内容、牛皮癣广告、含二维码、含小程
序码、其他广告 |
| 不良场景识别 | 检测视频、图片中是否包含黑屏、黑
边、昏暗画面、画中画、吸烟、车内直
播等不良场景。 | 正常、图片中无内容(例如黑屏、白
屏)、画中画、吸烟、车内直播 |
| logo识别 | 检测视频、图片中是否包含logo信
息,例如台标,商标等。 | 正常、含受管控的logo、含商标 |
| 语音垃圾识别 | 检测视频、音频中是否包含语音垃圾信
息,如广告、涉政、暴恐等。 | 正常、含垃圾信息、广告、涉政、暴
恐、辱骂、色情、灌水、违禁、无意
义、自定义(例如命中自定义关键词) |

应用场景

● UGC内容审核

海量UGC(用户生成内容)上传的视频,通过视频智能审核技术,快速检测违规内容,识别媒体质量问题,有 效减少人工审核的工作量。

● 点播内容审核

高效识别点播视频中的涉暴、涉政类镜头,紧跟时政动态,严控违规内容。

使用说明

前提条件,请参见视频AI。

• 提交智能审核任务

对于已成功上传到点播的视频、音频、图片文件,可以通过API/SDK方式提交智能审核作业。视频、音频文件调用提交智能审核作业接口进行提交,图片文件调用提交图片智能审核作业接口进行提交。

? 说明

使用提交智能审核作业接口进行图片审核时,使用限制如下:

- 。单个图片大小不超过20 MB, 高度或宽度不超过30000px, 且图片总像素不超过2.5亿px。
- (建议)图片像素不低于256×256px,像素过低可能会影响审核效果。

• 管理审核模板

提交智能审核任务时,可以指定任务使用的审核模板,模板用于配置审核功能。

- i. 审核模板的添加、删除、修改、查询可调用接口添加AI模板、删除AI模板、修改AI模板、查询AI模板、查询AI模板 列表。
- ii. 提交审核任务时,若不指定使用的审核模板,会使用默认模板。您可以调用接口设置默认AI模板,设置一个 自定义审核模板为默认模板。调用接口查询默认AI模板可查询当前的默认模板。
- 接收回调信息

智能审核作业完成后, 会通过消息回调方式通知回调URL, 具体参见: 智能审核完成。

- 支持API查询结果
 - i. 智能审核处理完成后,可以通过提交智能审核作业接口查询审核作业结果。
 - ii. 智能审核处理完成后,可以通过获取智能审核结果摘要接口查询审核结果摘要。
 - iii. 智能审核处理完成后,可以通过获取智能审核结果详情接口查询审核结果详情。
 - iv. 智能审核处理完成后,可以通过获取智能审核结果时间线接口查询审核结果时间线。
 - v. 音频审核处理完成后,可以通过获取智能音频审核结果详情接口查询音频审核结果详情。

○ 注意

审核结果的图片资源仅在点播提供的免费存储上保留2周时间,之后会自动删除。

7.3. 人工审核

视频点播提供人工审核服务,针对短视频、传媒和视频平台,建议先通过智能审核进行初审,对于疑似违规视频 结合人工复审。本文为您介绍人工审核的使用方式,如何查看审核历史。

使用方式

视频点播提供控制台和API两种方式对视频进行人工审核。

```
↓ 注意
```

使用前请确保已完成媒体审核准备工作。

- 控制台操作
 - 登录视频点播控制台,在点播控制台左侧的导航栏选择审核管理,单击视频审核,在视频审核页面进行人工 审核,具体操作,请参见视频审核。
 - 支持对单个视频进行审核,同时也支持对视频进行批量审核提交。
 - 。 审核结果会通过消息回调的方式通知回调URL, 更多信息, 请参见人工审核完成。

| 金融町町 マ 煤体名称 マ 法総入税税名称 Q | | | | | | | G |
|-------------------------|-----------|-----------|-----------|----------|-----------|-----------|-----------------------------|
| ■ 視频▽ | 机审状态
▽ | 审核模板
▽ | 机审结果
▽ | 违规项
☑ | 人审状态
□ | 终审结果
▽ | 攝作 |
| | ᇢ 未开启 | | | | 未审核 | | 洋情
开始审核

人工审核 ~ |
| 暫元裁測 | | | | | | 通过 | |
| | ᇦ 未开启 | | | | 未审核 | | 洋情
开始审核

人工审核 > |
| 通过屏蔽 | | | | | | | |

- API/SDK触发
 - 支持用户通过人工审核接口进行人工审核。
 - 审核结果会通过消息回调的方式通知回调URL,更多信息,请参见人工审核完成。

查看审核历史

 对于人工审核状态为审核完成的视频,可以通过单击视频审核页面的详情,在详情页面单击人工审核详情即可 查看该视频的人工审核历史。

| 点现控制台 / 视频审核 | | | | | | | |
|--|-------|----------------------------|-------|------|-------|-------|-----------------------------|
| 视频审核 | | | | | | | |
| 金飾対局 >> 媒体名称 >> 清编入现录名称 Q | | | | | | | С |
| | 机审状态公 | 审核構板♡ | 机审结果♡ | 违规项♡ | 人审状态♡ | 终审结果♡ | 操作 |
| 0000 CON | ● 未开启 | | | | 审核完成 | 通过 | 详情
开始审核
人工审核 > |
| | ᇢ 未开启 | | | | 未审核 | | 详情
开始审核

人工审核 > |
| | | | | | | | _ |
| 品/通型利旨 / 视频单核 / 视频单核;平衡 | | 人工审核 | 记录 | | | | × |
| ← 视频审核详情 | | | | | | | |
| and the second se | | 审核时间 | 审核人 | 5 | 审核结果 | 审核原因 | |
| And in case of the local division of the loc | | 2020-11-
12
14:19:25 | 17 |)4 | 通过 | | |
| | | | | | | | |
| Contraction of the local distance of the loc | | | | | | | |
| the second se | | | | | | | |
| | | | | | | | |
| 上一会 下一条 通过 | | | | | | | |
| □ 完成人工审核后自动切换到下一条 | | | | | | | |
| | | | | | | | |
| D | | | | | | | |
| | | | | | | | |
| 人审结果 | | | | | | | |
| 通过人工审核详情 | | | | | | | |

• 也可以通过获取人工审核历史接口追溯人工审核历史。

7.4. 审核安全IP

视频云点播服务提供安全审核IP服务,对于审核中或屏蔽状态的视频,保证只有来自审核安全IP的请求才能获取到播放地址。

使用说明

- 视频云点播服务提供安全审核IP服务,是通过IP白名单的方式,来限制用户获取视频的播放链接。如果用户将处于审核中或屏蔽状态的播放链接泄漏,拿到播放链接的第三方同样可以播放。
- 无论是否设置安全审核IP,均可通过点播控制台播放审核中或屏蔽状态的视频。

使用方式

● 设置审核安全IP

当视频状态为审核中或屏蔽状态时,只有来自审核安全IP的请求才可以获得播放地址。具体参见:设置审核安全IP。

设置安全IP支持两种模式:

- 精确IP: 10.23.12.24
- CIDR模式: 10.23.12.24/24 (无类域间路由, /24 表示了地址中前缀的长度,范围[1,32])
- 获取审核安全IP列表

对于已经设置的安全审核IP,用户可以通过获取审核安全IP列表接口查询审核安全IP列表。

8.视频AI

8.1. 概述

视频点播提供智能审核、视频DNA、智能标签、智能封面的视频AI服务,本文为您简要介绍这些功能、前提条件 与调用方式。

简介

阿里云视频AI服务可以对音视频内容进行识别、分析和理解。通过使用视频AI服务,您可以:

- 检测违规的视频内容;
- 识别并搜索重复或相似的音视频片段;
- 识别视频中的人物、文字、实体、场景和动作;
- 分析理解视频用以智能生成视频标签、推荐封面、动图及视频摘要;
- 支持将语音识别为文本等。

主要功能

| 功能 | 描述 | 参考文档 |
|-------|---|---|
| 智能审核 | 智能审核服务支持对点播视频资源的视频文件、封面图片及标题文本中涉
黄、性感、暴恐、特殊装束、特殊标识、武器、涉政等内容进行识别,并
给出建议结果。更多信息,请参见 <mark>智能审核</mark> 。 | 开发指南 API概览 |
| 媒体DNA | 媒体DNA可以标记唯一的一个视频、音频或图片,具有稳定性,不会随音
视频文件的格式转换、剪辑拼接、压缩旋转等变换而发生变化。媒体DNA
服务实现对视频中的图像、音频等指纹特征的提取和比对,解决重复视频
查找、视频片段查源、原创识别等问题。更多信息,请参见 <mark>视频DNA</mark> 。 | 开发指南API概览 |
| 智能标签 | 智能标签服务通过分析视频中视觉、文字、语音、行为等信息,结合多模
态信息融合及对齐技术,实现高准确率内容识别,自动输出视频的多维度
内容标签,将非结构化信息转化为结构化信息。更多信息,请参见 <mark>智能标</mark>
<mark>签</mark> 。 | 开发指南API概览 |
| 智能封面 | 智能封面服务支持对视频内容进行分析和理解,提取最能代表视频内容的
5张截图作为封面备选图;也支持将视频内容的关键画面进行抽取,自动
合成Gif作为动图封面。详情请参见 <mark>智能封面</mark> 。 | API概览 |

前提条件

- 您已经注册了阿里云账号并完成账号实名认证。注册地址请单击阿里云官网。注册指引请参见注册阿里云账号。
 实名认证指引请参见个人实名认证或企业实名认证。
- 您已经开通了视频点播服务。开通步骤请参见开通视频点播。
- 调用服务端接口需要使用AccessKey完成身份验证,请提前获取AccessKey。获取方法请参见获取AccessKey。
- 登录控制台,在回调配置中,开启视频AI处理完成。具体操作,请参见回调设置。
 所有AI处理都共用此开关,接收相应AI作业完成的事件通知回调。

视频AI处理的音视频文件需要存到点播媒资库。具体操作,请参见媒资管理。
 音视频文件的格式要求与点播上传格式要求一致。具体操作,请参见媒体上传。
 音视频分析已完成,接收到音视频分析完成事件通知。

调用方式

视频AI服务需要调用提交AI作业API接口,API请参见<mark>SubmitAIJob</mark>。您可通过三种方式获取AI结果数据。流程图如下 所示:



8.2. 视频DNA

视频DNA具有唯一性和稳定性,支持自动触发和通过API接口触发。本文为您介绍视频DNA的特性、应用场景、准备工作和使用说明。

简介

视频DNA通常是一个二进制串,可以用来唯一标记一个视频。您可以使用视频DNA服务实现对视频中的图像、音频等指纹特征的提取和比对,解决重复视频查找、视频片段查源、原创识别等问题。

特性

- 唯一性:不同视频的视频DNA不同。
- 稳定性:不会随音视频文件的格式转换、剪辑拼接、压缩旋转等变换而发生变化。

应用场景

● 视频查重

建立独立视频DNA库,在该库中对样本视频完成快速比对,实时召回重复视频或重复片段。防止内容分发和个性化推荐时,出现过多相同或相似视频而影响用户体验。

• 原创识别

准确识别是否为原创视频、编辑变换后视频、自媒体再创作视频。解决短视频领域维权问题,为视频素材交易 生态提供技术基础,防止侵权视频侵害原创者及视频平台利益。

快速审核

通过维护违规视频库或自定义黑库,对增量视频,快速进行DNA比对并判断是否有违规内容。相比传统审核, 提高大量视频过审的效率,保障准确性,降低审核的成本。

• 广告分成

通过视频DNA技术,检索分成广告库,对特定广告进行监播和识别。有利于支撑广告动态分成业务生态,轻松 把控广告投放的时间、次数等,同时保障了广告版权方和投放平台的利益。另外,基于此,可以进行广告的高 效识别、替换及广告位竞拍。

视频溯源

对于新增视频,在庞大的DNA库中进行快速检索,以定位一个视频的传播路径。有效解决视频溯源、传播路径 分析或挖掘媒体资源之间联系的问题。

准备工作

- 完成视频A概述中的前提条件后,还需进入视频DNA产品页单击申请开通,填写相关信息提交后等待通知。
- 初次使用视频DNA时,需提供uid及地域信息,通过提交工单申请免费开通DNA指纹库,否则将影响后续视频 DNA的功能使用。

使用说明

? 说明

使用视频DNA会产生计费,计费详情请参见视频Al计费。只要调用提交视频DNA作业接口并成功处理任务,就会 产生视频DNA计费,比如视频使用设置了智能媒资查重的工作流发起处理时,实际上会调用提交视频DNA作 业接口,从而产生视频DNA计费。

• 自动触发

开通视频DNA服务后,上传到点播的视频在转码完成后会自动触发视频DNA处理,处理结果会通过消息回调的 方式通知,具体回调消息格式请参见视频DNA完成。

● API触发

- 对于已经上传到点播的视频,支持通过调用提交视频DNA作业接口,提交视频DNA作业。
- · 视频DNA作业提交后,会进入后台异步处理,最终的结果会通过消息回调方式通知客户,具体回调消息格式 请参见视频DNA完成。同时也支持通过调用查询视频DNA作业接口,查询作业结果。

? 说明

- 对于自动触发方式,必须配置转码,只需要输出一路MP4即可(如流畅或原画);视频DNA处理 会在一路转码完成后开启。
- API方式可不配置转码,直接对源文件进行处理。

● 结果查询

- 视频DNA作业完成后,可以通过调用获取视频DNA结果接口,查询DNA结果。
- 视频DNA结果说明。

视频DNA处理结果中包含以下主要信息:

- 该作业是否有匹配到重复的视频。
- 如果匹配到了, 输出重复视频的信息。
- 视频之间重复的时码区间。
- 视频重复的程度(重复程度为介于0~1的数字)。

● 存量视频

为了确保DNA的比对效果足够好,往往需要导入存量视频来构建完整的特征库,为此我们提供了两种方式供您参考:

? 说明

视频点播不支持新建独立的DNA库(特征库),只能使用默认DNA库。如需新建DNA库,可使用媒体处理服务。详细信息,请参见提交新建DNA库。

- 如果您的视频存量少于100w分钟,可以直接通过调用提交视频DNA作业接口的方式进行存量视频处理。
- 如果您的视频存量大于100w分钟,请提交工单联系我们为您处理,并请详细说明存量视频的数量、时长,以及存储的位置。

8.3. 智能标签

智能标签功能是通过分析视频中多维度的信息,自动能输出内容标签的功能,能将非结构化信息转化为结构化信息。支持通过API调用,本文为您介绍智能标签的功能特点、应用场景、使用前提和使用说明。

功能特点

• 多维度信息识别

从视觉、文字、语音、运行行为等多个维度对视频信息进行分析,识别视频中的人物、物体场景、地标、文字、语音等内容。

• 多模态融合理解

采用多模态融合对齐、跨模态映射等技术,对媒体内容进行多模态融合识别,高效、精准理解视频内容。

● 丰富灵活的标签

结合成熟完善的媒体标签体系,根据多模态融合理解结果对媒体文件进行智能打标,输出多维度视频标签信息,同时提供灵活的标签自定义功能。

应用场景

• 准确高效的媒体资源检索

通过准确丰富的视频标签体系,对视频、图像、文本等媒体文件进行内容编目,支持通过关键词或标签文本, 对素材库中的内容进行快速检索,提升视频资源检索效率与准确性。

• 个性化推荐

精准、个性化的内容推荐。根据媒体文件内容标签,结合用户信息、行为数据及用户画像分析,进行内容精准 匹配,实现视频个性化推荐,解决推荐系统冷启动问题,提升推荐准确率。

• 智能广告投放

智能化、场景化的广告投放。根据媒体文件多模态内容分析及视频标签结果,自动识别广告位,与广告资源库 内容进行匹配,实现广告智能投放,精准触达受众人群,提升广告转化率。

使用前提

前提条件,请参见视频AI概述。

使用说明

● 提交智能标签任务(提交API作业)

对于已成功上传到点播的视频,可以通过调用提交智能标签任务接口,提交智能标签作业。

● 接收回调信息

智能标签任务完成后,如果开启了**视频AI处理完成**事件通知,会通过消息回调方式通知回调URL,具体回调消 息格式请参见智能标签作业完成。

• 查询作业结果

支持API或SDK方式主动查询作业结果,可通过作业ID或视频ID进行查询:

- 通过作业ⅠD查询任务信息及结果,请参见查询智能标签任务。
- 通过视频ID查询结果,请参见获取智能标签结果。

9. 音视频播放

9.1. 概述

视频点播支持播放音频和视频文件,您可以通过控制台预览、集成阿里云播放器SDK和集成第三方播放器进行播放。本文为您介绍音视频播放的整体流程、准备工作、前提条件、播放方式、播放地址和播放安全。

简介

视频点播(VOD)支持音频和视频文件的播放,文件来源包括:

- 选择转码方式上传的文件经转码处理后产生的输出流文件;
- 选择不转码方式上传的文件作为原画用于播放,目前仅以下格式支持不转码直接播放:MP4、FLV、M3U8、 MP3、WEBM。

您可以通过以下三种方式对音视频文件进行播放:

- 点播控制台预览,具体操作,请参见媒资管理;
- 集成阿里云播放器SDK,更多信息,请参见播放器SDK产品说明;
- 集成第三方播放器。

您可以通过以下两种方式获取播放地址:

- 直接获取:通过查看转码完成后的事件通知,或调用获取视频播放地址接口。
- 通过播放凭证自动换取:使用阿里云播放器SDK,可调用获取视频播放凭证接口来获取播放凭证,阿里云播放器 SDK通过播放凭证自动换取播放地址进行播放。

准备工作

- 配置加速域名:请确保启用的点播域名进行过域名备案。更多信息,请参见域名准入标准。
- 域名CNAME解析:请确保域名完成了CNAME绑定,否则无法播放。CNAME绑定请参见 阿里云(原万网)解析配置 CNAME流程,同时也支持新网和DNSPod的域名接入。
- 确认转码配置:视频点播支持对上传至点播的媒体文件做转码和不转码处理,请参见音视频转码。
- 确认安全配置:视频点播支持多种对视频内容的保护机制,包括访问限制、播放中心鉴权、业务方二次鉴权、视频加密及安全下载,视频是否可播放受限制于以上几种安全配置,请参见视频安全。

播放前提

● 视频状态

音视频在生产、处理和分发等流程中,状态会发生迁移和变更,其状态也是决定视频播放的关键条件。视频状态定义请参见视频状态 Status,详细变更规则如下:

。 有转码



○ 不转码



- 判断条件
 - 只有状态为正常(Status 字段值为Normal)的视频才能正常播放,即通过点播API/SDK能获取到播放地 址。
 - 当状态为审核中或屏蔽(Status 字段值分别为Checking、Blocked)时,视频只能在控制台播放,或是
 通过安全ⅠP进行播放。您可以通过点播API/SDK来设置审核安全ⅠP。
- 判断方式

用户上传视频完成后,并不代表视频已准备就绪,还需点播服务端确认接收完成,可根据**事件通知**来确定上传 的视频何时能播放。

- 对于不转码的视频(或音频)接收到视频上传完成事件通知后即可进行播放,此时可通过调用获取视频播放地 址接口获取到播放地址,目前仅以下格式支持不转码直接播放:MP4、FLV、M3U8、MP3、WEBM。
- 对于转码的视频,接收到单个清晰度转码完成事件通知后即可进行播放,若要确保所有清晰度都能获取到,可 接收到视频转码完成再处理。

播放方式

- 点播控制台预览
 - 在控制台的音/视频页面选择视频进行预览,优先播放加密流。
 - 在控制台的音/视频>管理>视频地址页面选择某路流进行预览。为了保障您的视频安全,目前仅支持非加密 流预览。
- 集成阿里云播放器SDK
 - 您可以从服务端获取播放凭证,回传给客户端进行播放,好处在于安全性较高。更多信息,请参见通过播放凭 证播放。
 - 目前阿里云播放器支持直接使用播放地址进行播放,可以将获取到的播放地址直接传递给播放器进行播放。

 更多信息,请参见获取播放地址播放。
- 集成第三方播放器
 - 您可以集成第三方播放器获取播放地址播放。
 - · 获取到播放地址后,传递给自主播放器进行播放,方式比较灵活,但需要自己实现清晰度切换、异常处理等
 开发工作。

播放地址

• 配置加速域名

在点播控制台配置加速域名后,视频播放地址为CDN文件地址,可在点播控制台的**音/视频>管理>视频地址**页面查看,分为固定和动态两种,取决于是否在域名管理里开启了URL鉴权开关,开启和配置,请参见URL鉴权。

• 固定地址

适用于安全性要求低的场景,地址永久有效,即关闭鉴权开关后不含鉴权信息的地址(URL中的 auth_key 参数值即为鉴权信息)。目前控制台添加域名后鉴权开关默认为关闭。

○ 动态地址

适用于安全性要求高的场景,地址动态生成,一定时间后过期。动态地址的默认有效时间为您在URL鉴权里配置的 默认有效时长 ,同时您也可以在生成播放地址或获取视频播放地址时设置过期时间;若地址过期则访问时

CDN会返回 HTTP 403 。

动态地址示例:

http://example.aliyundoc.com/video/aliyun-sample.mp4?auth_key=1500523200-0-0-80cd3862d699b71
18eed99103f2a****

⑦ 说明

示例中的auth_key参数值前缀1500523200(时间为2017-07-2012:00:00), 且默认有效时长设置为 60分钟,则表示该地址会在2017-07-2013:00:00过期。

- 未配置加速域名
 - 未配置加速域名时,返回的播放地址为OSS文件地址,此时无法使用URL鉴权,但会默认生成OSS鉴权信息, 请参见OSS-在URL中包含签名。此时,通过播放调用获取视频播放地址接口获取播放地址时,仍可使用
 AuthTimeout 参数指定视频OSS地址的过期时间,但无法根据AccessKey自定义鉴权信息。
 - 如您在控制台的存储管理页面设置了存储 Bucket 为公共读,则可忽略OSS鉴权信息(具体操作,请参 见存储管理),此时地址为永久有效,但有被盗链、非法下载的风险,故存储 Bucket 应尽量设置为私有。

更多播放地址的常见设置,请参见播放常见设置。

播放安全(限制播放和下载)

• 视频安全

为了保护您的视频内容,视频点播提供了多种安全保障方式,包括:访问限制、URL鉴权、视频加密(阿里云视频 加密和HLS标准加密)。更多信息,请参见视频安全概述。

• 账号安全

为了保证安全,请不要将主账号或RAM用户AK放在客户端(特别是Web端)来访问点播服务。更多信息,请参 见<mark>账号和授权概述</mark>。

9.2. 获取播放地址播放

视频点播支持通过播放地址播放音视频文件。您可以通过开启转码完成消息通知和API/SDK两种方式获取到播放地址。本文为您介绍播放的流程、获取播放地址的方式和播放说明。

播放流程



○ 注意

服务端需要持有RAM用户AK,并授予播放权限。更多信息,请参见账号与授权概述。

如果使用阿里云播放器,请参见使用阿里云播放器。

获取播放地址

视频点播支持通过事件通知和API/SDK两种方式获取播放地址。

• 事件通知

通过接收<u>单个清晰度转码完成或视频转码完成的事件通知获取到播放地址</u>,保存到用户自己的服务端,播放时访问 自己的服务端获取到存储的播放地址。

使用说明:

- 消息回调获取到的播放地址为 固定地址 ,若开启了URL鉴权则需要自己生成带鉴权的播放地址,否则无法播放。
- 如果您禁用或删除了旧的域名,保存的播放地址取出时请替换成新的域名,否则无法播放。
- 如果您配置了视频加密,返回的播放地址需要经过播放器解密后才可进行播放。
- API/SDK

通过调用获取视频播放地址接口SDK/API实时获取播放地址,需要在上传音视频文件时保存好视频ID。

调用方式:

- 推荐使用服务端SDK调用API来获取媒资信息,更为简单、高效。服务端SDK请参见使用说用,API详情可参见 各API文档的SDK示例。
- 您也可以参见公共参数和调用示例,来生成HTTP/HTTPS请求。

使用限制:

- 为了保证每次播放的是最新转码输出的流,播放接口默认每种清晰度和格式只返回一路最新转码完成的流。
 调用获取视频播放地址接口时,可以设置 ResultType 为 Multiple 来获取该音视频下的所有转码完成的
 流。
- 如果您选择了阿里云视频加密的方式,只有使用阿里云播放器SDK才可正常播放。为了保障您的视频安全,直接调用获取播放地址接口默认不返回私有加密流,您可以通过设置接口参数 ResultType 值为 Multiple 来获取所有转码完成的播放地址。
- 如果您选择了HLS标准加密的方式,默认会优先返回标准加密的流用于播放。

播放说明

您可以获取到播放地址后,传递给播放器进行播放。

- 阿里云播放器支持直接使用播放地址进行播放,请参见使用阿里云播放器。
- 您也可以使用系统原生播放器、开源播放器或自研播放器等。

9.3. 通过播放凭证播放

阿里云播放器支持通过播放凭证自动换取播放地址进行播放,接入方式更为简单,且安全性更高。您可以通过API获取播放凭证并使用播放器SDK进行播放。本文为您介绍通过播放凭证播放的播放流程、使用说明和使用方法。

播放流程



使用说明

- 播放凭证默认时效为100秒(最大为3000秒),只能用于获取指定视频的播放地址,不能混用或重复使用。如果凭证过期则无法获取播放地址,需要重新获取凭证。
- 播放凭证的时效并不是播放地址的时效(若开启了URL鉴权),后者可自定义过期时间,且无上限。
- 获取的播放凭证PlayAuth,并非真正的播放地址。阿里云播放器SDK会根据播放凭证自动获取到播放地址进行播放,但需处理好播放凭证过期后的刷新问题。
- 如果您选择了阿里云视频加密的方式,只有使用阿里云播放器SDK才可正常播放。

获取播放凭证

可调用获取视频播放凭证接口来获取播放凭证。

调用方式:

- 推荐使用服务端SDK调用API来获取媒资信息,更为简单、高效。服务端SDK请参见使用说用,API详情可参见各API文档的SDK示例。
- 您也可以参见公共参数和调用示例,来生成HTTP/HTTPS请求。

使用播放器SDK播放

阿里云播放器提供Web端、Android、iOS三种播放器SDK,且均支持使用播放凭证播放的方式,您可根据使用的播放器类型参见以下文档:

- Web端 (H5/Flash): 产品概述
- Android: 功能使用
- iOS: 功能使用

○ 注意

播放凭证具有时效性,如果凭证过期则无法获取播放地址,需要重新获取播放凭证。

9.4. 使用阿里云播放器

视频点播提供了阿里云播放器SDK,支持直接通过播放地址播放,也支持通过播放凭证自动换取播放地址播放。

阿里云播放器SDK (ApsaraVideo for Player SDK) 是阿里视频云端到云到端服务的重要一环。除了支持点播和直播的基础播放功能外,还深度融合视频云业务,如支持视频的加密播放、安全下载、清晰度切换等业务场景。为用户提供简单、快速、安全、稳定的视频播放服务。更多阿里云播放器的特色功能、核心优势、不同版本及使用场景等介绍,请参见SDK简介。

使用阿里云播放器播放视频发生异常时,可借助单点探查功能快速定位问题,详细内容,请参见单点探查。

现提供五种终端的播放器SDK:

| 终端 | 简介 | 说明文档 | 下载地址 |
|---------|--|------|-------|
| Web | 包括Flash和HTML5,可以
分别集成到PC的浏览器和移
动端的浏览器中。开发语言
JavaScript。 | 快速接入 | SDK下载 |
| iOS | 提供了SDK及Demo,可以
集成到iOS系统的App中。
开发语言Object-C。 | 快速集成 | SDK下载 |
| Android | 提供了SDK及Demo,可以
集成到Android系统的App
中。开发语言Java。 | 快速集成 | SDK下载 |
| Windows | 提供了SDK及Demo,可以
集成到Windows系统的程
序中。开发语言C++。 | 快速集成 | SDK下载 |

| 终端 | 简介 | 说明文档 | 下载地址 |
|---------|--|------|-------|
| Flutter | 提供了源码以及Demo,可
以快速在Flutter项目中集成
并使用 Android、iOS端播
放器。开发语言Dart、
Java、Object-C。 | 快速集成 | SDK下载 |

9.5. 播放常见设置

在使用视频点播播放服务时,可以根据不同的播放场景选择播放地址进行播放,您可以通过域名配置、转码配置 和播放设置来控制获取到的播放地址类型。本文为您介绍一些常用的设置。

域名配置

- 添加域名
 - 配置加速域名:请确保启用的点播域名进行过域名备案。更多信息,请参见域名准入标准。
 - 域名CNAME解析:请确保域名完成了CNAME绑定,否则无法播放。CNAME绑定请参见阿里云(原万网)解析配置CNAME流程,同时也支持新网和DNSPod的域名接入。
 - 配置CDN加速域名:具体操作,请参见添加加速域名。
- URL鉴权配置
 - 地址过期时间: 您可以设置CDN加速地址过期时间 AuthTimeout , URL失效时间计算规则,请参见URL鉴权-URL失效时间。
 - 二次鉴权:
 - 配置CDN二次鉴权时,设置 ReAuthInfo 参数可以在CDN的鉴权参数中附加uid和rand参数。更多信息, 请参见CDN二次鉴权设置 ReAuthInfo。
 - 未配置CDN二次鉴权时, ReAuthInfo 中的 rand 参数可用于确保每次生成的URL不同,如使用UUID等 做随机数。更多信息,请参见CDN二次鉴权设置 ReAuthInfo。
- 多域名选择

当您的源站配置了多个域名时,设置 PlayConfig 参数的 PlayDomain 用于返回指定CDN域名的播放地址。 更多信息,请参见媒体播放自定义设置 PlayConfig。

● HTTPS安全加速

播放接口返回的图片地址和播放地址的网络协议默认为HTTP,当需要获取HTTPS网络协议的地址时,视频点播 支持开启对CDN域名的HTTPS安全加速。具体操作,请参见HTTPS安全加速设置。

转码配置

视频点播支持您对上传至视频点播的媒体文件做转码或不转码处理。

- 转码后,播放服务获取的播放地址是经过转码处理后生成的,播放地址的格式、清晰度、是否加密及水印ID均 与您的转码配置有关。
- 不转码,播放服务获取的播放地址是您上传至视频点播的原始文件,作为原画用于播放。目前仅以下格式支持 不转码直接播放:MP4、FLV、M3U8、MP3、WEBM。

更多信息,请参见转码模板。

播放设置

播放服务支持如下三种播放方式:

- 控制台预览:无需配置。
- 播放地址: 在调用获取视频播放地址接口时, 通过不同的播放设置过滤播放地址用于播放。
- 播放器:对阿里云播放器SDK进行播放设置,播放器自动过滤播放地址进行播放。

播放接口设置:

视频点播获取视频播放地址接口,支持对如下基本播放信息做筛选,以满足不同的播放场景。

清晰度

Definition ,播放接口支持您一次选择多个清晰度,可以在不同网络情况下做清晰度的切换;当您未设置时,播放接口会返回指定视频存在的所有清晰度的播放地址。

• 输出文件格式

Formats ,转码处理后的文件封装格式,目前仅支持 MP4 、 MP3 、 M3U8 ,根据不同的播放场景您可以选择获取不同格式的输出流文件进行播放,支持传入多个值。

• 输出流类型

```
StreamType , 分为视频(video)和音频(audio), 不同的播放场景可以选择不同类型的输出流地址进行播放。
```

• 输出文件类型

OutputType :

- 未配置CDN加速域名时,播放接口默认返回的播放地址为OSS回源地址。
- 配置CDN加速域名后,播放地址默认返回的是CDN加速地址。
- 您也可以通过设置 OutputType ,获取OSS回源地址或CDN加速地址。
- 输出结果类型

ResultType , 视频点播默认每种清晰度和格式只对应一个播放地址,如果您对已经转码完成的视频提交媒体转码作业处理后,每种清晰度和格式会对应多个播放地址。为了保证每次播放的是最新转码输出的流,播放接口默认每种清晰度和格式只返回一路最新转码完成的流。当您需要获取所有转码完成的流地址时,可以设置 ResultType 值为 Multiple 。

↓ 注意

```
当一个视频对应的输出流地址超过100条时, ResultType 值为 Multiple 最多只能获取100条输出流地
址。
```

播放器设置:

阿里云播放器支持通过播放凭证自动换取播放地址进行播放,目前支持 Web(H5、Flash)、iOS和Android,以 下将介绍每个端支持的播放设置。

● Web播放器SDK

- 目前H5播放器支持设置输出流类型 mediaType 、输出文件格式 format 及清晰度 definition 。更多信

 息,请参见功能使用说明。
- o Flash播放器仅支持视频类型的媒体文件播放,支持设置输出流类型 mediaType 和输出文件格式 format 。
- iOS播放器SDK

目前支持设置输出文件格式 format 及期望清晰度 quality 。

● Android播放器SDK

目前支持设置输出文件格式 format 及期望清晰度 quality 。

10.云剪辑

10.1. 概述

视频剪辑(云剪辑)服务是视频点播的制作中心,提供在线可视化的剪辑平台(控制台)及相应的OpenAPI。本 文为您介绍云剪辑的整体工作流程、核心链路以及API的用法及示例。

简介

视频制作在整个视频链路(采集、制作、播放/互动、媒资管理)中具有非常重要的地位,将前期的采集拍摄和最终的分发播放连成一体,起着承前启后的作用。云剪辑服务就是视频点播的制作中心,支持剪切拼接、混音、字幕、图片叠加、遮标、转场特效等一系列功能。

云剪辑是一个云端一体的产品,包括前端组件/页面与后端服务两大部分。

• 云剪辑前端组件/页面

包括剪切、拼接、遮标、文字、调整顺序等组件或页面功能。具体操作,请参见视频剪辑。

● 后端服务

分为两部分内容:媒体合成服务和云剪辑工程管理服务。

• 您可以利用在线的云剪辑工具,对上传到点播媒资库的云端素材进行剪辑,提交合成;

• 也可以自行组织时间线数据,直接提交合成。

合成的新视频将存储到点播媒资库,您可以对它们快速进行分发播放,无需再进行下载和上传的操作。 云剪辑系统架构图如下:



| 名词 | 说明 |
|-------|--|
| 素材 | 用于进行剪辑制作的原材料。
可用于剪辑的媒体资源需要存到点播媒资库,支持类型包括视频、音频、图片和文本。 |
| 云剪辑工程 | 用于描述一次视频创作过程中涉及的数据。一般包括基本元数据(如标题、创建时间等),本次视
频创作所需的素材,以及描述实际视频创作内容的时间线。 |
| 时间线 | 将素材按照视频创意进行编排和特效设计的产物。 |
| 媒体合成 | 当视频创作完成后,可将最终制作好的时间线提交合成,生成新的媒体资源。媒体资源的合成是一个异步的过程,合成任务则描述了任务从开始创建到最终完成的过程。 |
| 成片 | 云剪辑的最终产物,也会存储到点播媒资库。 |

使用前提

- 输入: 可用于剪辑的媒体资源需要存到点播媒资库(多种途径: 上传、直播录制)。
- 输出:剪辑合成的媒体资源也将存到点播媒资库。

核心链路

云剪辑的核心链路分为输入、处理和输出三部分。

输入

云剪辑素材库,用来进行剪辑制作的原材料。

按素材类型,素材库可分为视频、音频、图片。各类素材文件支持的格式如下:

| 素材 | 支持格式 |
|----|--|
| 视频 | 封装格式:3GP、AVI、FLV、MP4、M3U8、MPG、ASF、WMV、MKV、MOV、TS、WebM、
MXF等。
编码格式:H.264/AVC、H.263、 H.263+、MPEG-1、MPEG-2、MPEG-4、MJPEG、VP8、VP9、
Quicktime、RealVideo、Windows Media Video等。 |
| 音频 | 封装格式: 3GP、AVI、FLV、MP4、M3U8、MPG、ASF、WMV、MKV、MOV、TS、WebM、
MXF等。
编码格式: AAC、AC-3、ADPCM、AMR、DSD、MP1、MP2、MP3、PCM、RealAudio、
Windows Media Audio。 |
| 图片 | JPG、JPEG、PNG、GIF、APNG |

? 说明

目前素材库的来源为点播媒资库资源。点播媒资库资源,一般包括源文件与转码输出文件。为了保证得到 最好质量的成片,云剪辑合成服务使用点播资源的源文件进行剪辑与合成。

处理

剪辑合成处理过程中最核心的对象是:时间线(Timeline)。Timeline包含了多个轨道(又叫多层),每个 轨道又可以包含多个素材片段(clip)。您可以为这些clip指定入出点、顺序以及相应的effect等参数。

Timeline来源如下:

- 。 使用前端剪辑组件进行剪辑操作,并进行保存,可生成Timeline数据。
- 可以通过调用视频合成服务的OpenAPI或SDK, 直接传入Timeline数据。

媒体合成服务是云剪辑的核心服务,负责最终成片的生成。该服务的处理模式为异步处理。用户调用视频合 成接口,该服务会同步返回合成的新资源D,并根据参数的情况启动异步的合成任务。

 ・ 用户可以通过接收媒体合成完成事件的通知,来获取合成完成的状态,当接收到该通知时,标志着新资源的源 文件已经合成完毕。



 ・ 也可以主动根据新资源ID查询资源状态,当资源状态在上传完成状态之后,标志着新资源的源文件已经合成 完毕。

? 说明

这种方式效率相对较低,建议通过事件回调方式接收合成完成通知。

云剪辑工程管理服务包括云剪辑工程的创建、更新、删除、获取,查询,以及为云剪辑工程设置素材等服务。 对云剪辑服务有更深入管理需求的用户,可以使用云剪辑工程管理服务对视频创作过程的工程进行管理。

● 输出

云剪辑合成的目标成片。目前,云剪辑成片均输出到点播媒资库,支持三种输出资源类型:视频、纯音频和 GIF。

云剪辑合成输出最核心设计原则是: 源文件输入输出。



云剪辑服务用于合成的是素材的原始文件,而合成的成片文件也作为点播资源的原始文件存储到媒资库。云剪 辑合成的目标媒体资源(Media),还有以下几个原则:

- 目标媒体资源分辨率与Timeline所使用的素材源文件分辨率有关:取其最大宽高。例如: Timeline使用了3个 素材,其源文件分辨率分别为:1280×720,1920×1080,720×1280,则目标成片的分辨率为1920×1280。
- 目标媒体资源码率与Timeline所使用的素材源文件码率有关: 取其最大码率。
- 。 云剪辑默认的合成目标类型为视频,编码格式是H.264,封装格式为MP4;也支持用户配置和请求级别的合成 模版指定,对输出目标资源类型(纯音频、GIF等)以及合成模版的详细参数进行设置。

合成目标文件的输出格式如下:

| 素材 | 支持格式 |
|----|---|
| 视频 | 封装格式: FLV、MP4、TS、M3U8、MPD。
编码格式: H.264/AVC、H.265/HEVC。 |
| 音频 | 封装格式: FLV、MP4、TS、M3U8、MPD。
编码格式: MP3、AAC、VORBIS、FLAC。 |
| 图片 | GIF |

云剪辑合成完成,即目标资源的源片生成完毕之后,可根据需要自动发起针对该源片的转码和截图流程,方便 客户快速分发。

- 可通过 ProduceConfig 中的 TemplateGroupId 来设置源片合成之后启动的转码流程,不设置该参数时, 合成之后根据配置的默认模版组启动转码流程。转码模版组的具体配置参考转码设置。
- 转码和截图流程是在媒体合成的源片生成之后才发起的,属于媒体合成的后处理流程。

媒体合成API用法

• 通过时间线合成

直接指定时间线,即Timeline数据,调用ProduceEditingProjectVideo接口,绝大多数场景下,使用这种方式即 可,见下图:



• 通过云剪辑工程合成

通过云剪辑工程ID进行媒体合成的方式,适用于对云剪辑工程进行深入管理,并经由云剪辑工程来提交合成任 务的场景。这种方式又可细分为两种:

- 。 方式1
 - a. 创建云剪辑工程, 创建时直接指定工程的Timeline数据, 对应AddEditingProject接口。
 - b. 根据工程ID合成,对应ProduceEditingProjectVideo接口。



。 方式2

- a. 创建云剪辑工程, 创建时根据需要指定工程的Timeline数据(也可以在创建的时候不指定), 对应 AddEdit ingProject接口。
- b. 更新云剪辑工程的Timeline数据,对应UpdateEditingProject接口。
- c. 根据工程ID合成,对应ProduceEditingProjectVideo接口。



使用示例

视频剪辑提供了丰富的使用示例,特别是通过时间线(Timeline)进行剪辑合成,覆盖如下方面:

- 剪切合并:实现整段合成、保留片头剪切、保留片尾剪切、掐头去尾和多段视频截取任意区间合并等。
- 音频处理:实现音频处理的各种场景,如静音、混音、音频提取、音量调节、配音等。
- 图片叠加:实现全程叠加、指定区域叠加图片等,包括单视频叠加和多视频叠加。
- 文字叠加: 实现全程叠加、指定区域叠加文字等。

10.2. 剪切合并

本文为您介绍整段合成、保留片头剪切、保留片尾剪切、掐头去尾和多段视频截取任意区间合并的示例,以视频 剪辑基本的剪切合并场景Timeline的各种数据组织举例。

简介

视频剪辑的简介、总体流程和媒体合成服务的两种使用方式已在概述中介绍过,媒体合成最终都是通过视频合成接口发起并实现的,而Timeline是该服务重点处理的数据,也是视频剪辑中最核心的对象。Timeline内部参数可以组织成多种形式,以应对不同的业务场景。

整段合成示例

整段合成指多段视频直接合成。将两个视频完整顺序相接,整段合成,此时不用指定视频的入出点,示例如下:

保留片头剪切示例

保留片头剪切指仅设置出点,不设置入点,即入点从0开始。截取视频的前5秒,示例如下:

保留片尾剪切示例

保留片尾剪切指仅设置入点,不设置出点,即出点为片尾。截取视频的后10秒,示例如下:

掐头去尾示例

掐头去尾指同时设置入出点,保留入出点之间的视频。示例如下:

多段视频截取任意区间合并示例

可以指定多个视频的多段区间进行剪切合并。截取视频 4bcf9b4d7cf14dc7b83b0e801cbe**** 的2个片段和视频 789f9b4d7cf14dc7b83b0e801cbe**** 的1个片段进行合并,片段合并的顺序按照数组的先后顺序首尾相接,示例如下:

```
{
   "VideoTracks": [
       {
            "VideoTrackClips": [
                {
                   "MediaId": "4bcf9b4d7cf14dc7b83b0e801cbe****",
                   "In": 12,
                   "Out": 16
               },{
                   "MediaId": "4bcf9b4d7cf14dc7b83b0e801cbe****",
                   "In": 4,
                   "Out": 7
               },{
                   "MediaId": "789f9b4d7cf14dc7b83b0e801cbe****",
                   "In": 12,
                   "Out": 20
               }
           ]
      }
  ]
}
```

10.3. 音频处理

本文为您介绍整体静音、指定区间静音、提取音频、视频静音 + 完整音频混音、视频静音 + 音频指定区间混音、 视频调节音量 + 音频调节音量、音频拼接、多轨音频混音、综合配音的示例,以音频处理场景Timeline的各种数据 组织举例。

简介

视频剪辑的简介、总体流程和媒体合成服务的两种使用方式已在概述中介绍过,媒体合成最终都是通过视频合成接口发起并实现的,而Timeline是该服务重点处理的数据,也是视频剪辑中最核心的对象。Timeline内部参数可以组织成多种形式,以应对不同的业务场景。

整体静音示例

整体静音指将视频资源中的声音进行整体消除,即静音操作。如下示例中使用了针对音频的Volume Effect。 Gain 的含义是音量增益,值为0表示静音。

```
{
   "VideoTracks": [
      {
           "VideoTrackClips": [
              {
                  "MediaId": "3f7e62d41a334dec9ac802b0f165****",
                   "Effects": [
                      {
                          "Type": "Volume",
                          "Gain": "0"
                      }
                  ]
              }
          ]
      }
  ]
}
```

指定区间静音示例

指定区间静音与视频整体静音的参数相比,增加了针对Volume Effect进行的入出点参数的设置。对视频的第8秒 到第60秒的区间进行静音,视频的剩余部分维持原始音量不变,示例如下:



提取音频示例

提取音频指在某些场景中,需要将视频资源中的音频提取为单独的音频资源,您可以参考下面的示例,创建 AudioTracks,将视频资源传入作为AudioTrackClips传入即可,b3f37e05512043f49f697f7425b9****表示一个 带声音的视频资源ID。

视频静音 + 完整音频混音示例

视频静音 + 完整音频混音指将原视频静音后重新配音, 合成新的视频, 是一个典型的配音场景。示例如下:

- 所有与音频处理相关的Clip都放入AudioTrack。
- TimelineIn 设为5,表明该音频在输出视频的第5秒开始叠加。
- TimelineOut 不设,则默认为整段音频都将进行混流,如果音频的尾部超过了视频轨的总时长,则将对音频进行截断,播放到视频轨的尾部。

⑦ 说明 AudioTrackClip可以是纯音频资源,也可以是带声音的视频资源。

```
{
   "VideoTracks": [
       {
           "VideoTrackClips": [
               {
                  "MediaId": "3f7e62d41a334dec9ac802b0f165****",
                   "Effects": [
                      {
                           "Type": "Volume",
                          "Gain": "0"
                       }
                  ]
              }
          ]
      }
   ],
   "AudioTracks": [
      {
           "AudioTrackClips": [
              {
                  "MediaId": "4a71744998414cbe8ea1976435a7****",
                  "TimelineIn":5
               }
           ]
       }
  ]
}
```

视频静音+音频指定区间混音示例

与视频静音 + 完整音频混音相比,视频静音 + 音频指定区间混音增加了对音频Clip入出点的设置。截取音频第10秒 到第20秒的区间(即10秒的音频片段),从输出视频的第5秒开始叠加,示例如下:

⑦ 说明 AudioTrackClip可以是纯音频资源,也可以是带声音的视频资源。

```
{
   "VideoTracks": [
       {
            "VideoTrackClips": [
               {
                   "MediaId": "3f7e62d41a334dec9ac802b0f165****",
                   "Effects": [
                        {
                           "Type": "Volume",
                           "Gain": "0"
                        }
                   ]
               }
           ]
       }
   ],
   "AudioTracks": [
       {
            "AudioTrackClips": [
               {
                   "MediaId": "4a71744998414cbe8ea1976435a7***",
                   "In":10,
                   "Out":20,
                   "TimelineIn":5
               }
           ]
       }
   ]
}
```

视频调节音量 + 音频调节音量示例

您可以通过Volume Effect设置音量, Gain 表示音量大小。 Gain参数说明如下:

- 0为静音
- 1为原始音量
- 0~1之间的值表示小于原始音量,值越小则音量越小
- 大于1的值表示大于原始音量, 值越大则音量越大

⑦ 说明 AudioTrackClip可以是纯音频资源,也可以是带声音的视频资源。
```
{
   "VideoTracks": [
      {
           "VideoTrackClips": [
              {
                  "MediaId": "3f7e62d41a334dec9ac802b0f165****",
                  "Effects": [
                      {
                          "Type": "Volume",
                          "Gain": "0.5"
                      }
                  ]
              }
          ]
      }
   ],
   "AudioTracks": [
      {
           "AudioTrackClips": [
              {
                  "MediaId": "4a71744998414cbe8ea1976435a7****",
                  "In":10,
                  "Out":20,
                  "TimelineIn":5,
                  "Effects": [
                     {
                          "Type": "Volume",
                          "Gain": "2"
                      }
                  ]
              }
         ]
      }
 ]
}
```

音频拼接示例

以上示例都存在视频轨,云剪辑还支持对纯音频轨的处理。在同一个音轨,将两个音频资源截取指定的区间顺序 相接,合成总时长为30秒的音频资源,示例如下:

⑦ 说明 AudioTrackClip可以是纯音频资源,也可以是带声音的视频资源。

```
{
    "AudioTracks": [
       {
           "AudioTrackClips": [
              {
                  "MediaId": "b3f37e05512043f49f697f7425b9****",
                  "In": 100,
                  "Out": 120
               },
               {
                  "MediaId": "ab654a04ce554e4f806b5f9e5a34****",
                  "In": 50,
                  "Out": 60
               }
          ]
      }
 ]
}
```

多轨音频混音示例

除同轨音频拼接之外,云剪辑还支持将多个音轨的音频资源进行混音。合成一个总时长为20秒的音频,示例如下:

⑦ 说明 AudioTrackClip可以是纯音频资源,也可以是带声音的视频资源。

```
{
   "AudioTracks": [
       {
            "AudioTrackClips": [
                {
                   "MediaId": "b3f37e05512043f49f697f7425b9****",
                   "In": 100,
                   "Out": 120,
                   "Effects": [
                       {
                           "Type": "Volume",
                           "Gain": "2"
                       }
                   ]
               }
           ]
       },
       {
            "AudioTrackClips": [
               {
                   "MediaId": "ab654a04ce554e4f806b5f9e5a34***",
                   "In": 50,
                   "Out": 60,
                    "Effects": [
                       {
                           "Type": "Volume",
                            "Gain": "1"
                        }
                   ]
               }
           ]
      }
  ]
}
```

综合配音示例

综合配音指视频 + 多轨音频混音,下面是一个综合应用音频处理的例子,包括视频指定区间静音,指定区间调节 音量,外加多个音频轨混音等功能。具体操作:

- 视频3f7e62d41a334dec9ac802b0f165****的第50秒~75秒静音。同时在这段时间内,启用两个音轨进行混 音,第一个音轨截取了音频素材b3f37e05512043f49f697f7425b9****的第100秒~120秒区间,从50秒处开 始插入,到70秒处结束,音量增大到2;第二个音轨截取了音频素材ab654a04ce554e4f806b5f9e5a34**** 的第150秒~160秒区间,从65秒处开始插入,到75秒处结束。
- 2. 视频3f7e62d41a334dec9ac802b0f165****的第50秒~75秒、第120秒~125秒之外的区间,音量保持正常。

⑦ 说明 AudioTrackClip可以是纯音频资源,也可以是带声音的视频资源。

示例如下:

```
{
    "VideoTracks": [
        {
            "VideoTrackClips": [
               {
                "WediaId"• "3f7e62d41a334dec9ac802b0f165****"
                "MediaId"• "3f7e62d41a334dec9ac802b0f165****"
```

```
JT / COEd ITUSS IGCOJUCOVEDOTTOS
                     icururu .
                                                               ,
                   "Effects": [
                      {
                           "Type": "Volume",
                           "Gain": "0",
                           "In": 50,
                           "Out": "75"
                       },
                       {
                           "Type": "Volume",
                           "Gain": "0.8",
                           "In": 120,
                           "Out": "125"
                      },
                  ]
              }
         ]
     }
   ],
   "AudioTracks": [
      {
           "AudioTrackClips": [
              {
                   "MediaId": "b3f37e05512043f49f697f7425b9****",
                   "In": 100,
                   "Out": 120,
                   "TimelineIn":50
                   "Effects": [
                      {
                           "Type": "Volume",
                           "Gain": "2"
                      }
                  ]
              }
           ]
       },
       {
           "AudioTrackClips": [
               {
                   "MediaId": "ab654a04ce554e4f806b5f9e5a34****",
                   "In": 150,
                   "Out": 160,
                   "TimelineIn":65
                   "Effects": [
                     {
                          "Type": "Volume",
                          "Gain": "1"
                      }
                  ]
              }
         ]
      }
 ]
}
```

10.4. 图片叠加

本文为您介绍图片叠加的参数说明和全程叠加、指定区域叠加的示例,以图片叠加场景Timeline的各种数据组织举例。

简介

视频剪辑的简介、总体流程和媒体合成服务的两种使用方式已在概述中介绍过,媒体合成最终都是通过视频合成接口发起并实现的,而Timeline是该服务重点处理的数据,也是视频剪辑中最核心的对象。Timeline内部参数可以组织成多种形式,以应对不同的业务场景。

参数说明

| 参数 | 说明 |
|--------------------|--|
| 图片出现的坐标 | X:图片左上角距离输出视频左上角的横向距离。 Y:图片左上角距离输出视频左上角的纵向距离。 支持百分比和像素两种形式: 当取值范围是0~0.9999时,表示相对输出视频宽/高的占比(X相对宽,Y相对高)。 当取值为≥8的整数时,表示绝对像素。 |
| 图片在输出视频中所占的大
小 | Width: 图片在输出视频中的宽度。 Height: 图片在输出视频中的高度。 支持百分比和像素两种形式: 当取值范围是0~0.9999时,表示相对输出视频宽/高的占比(Width相对宽, Height相对高)。 当取值为≥8的整数时,表示绝对像素。 |
| 图片在输出视频叠加的时间
区间 | TimelineIn:图片相对于时间线的入点。 TimelineOut:图片相对于时间线的出点。 |

全程叠加示例

全程叠加指从片头到片尾叠加图片,即视频全程叠加图片不需要指定 TimelineIn 和 TimelineOut 。图片在输出视频中出现的位置由 x 和 y 设定,在输出视频中的宽高由 Width 和 Height 设定。示例如下:

● 单视频叠加

```
{
  "VideoTracks": [
     {
         "VideoTrackClips": [
           {
               "MediaId": "222d9296e8864746a0b6f32dad6e****"
            }
         ]
     }
  ],
  "ImageTracks": [
     {
         "ImageTrackClips": [
            {
                "ImageId": "001d9296e8864746a0b6f32dad6e****",
                "Width": 0.1345,
                 "Height": 0.1678,
                 "X": 0.1234,
                 "Y": 0.1234
             },
             {
                "ImageId": "002d9296e8864746a0b6f32dad6e****",
                 "Width": 0.1345,
                 "Height": 0.1678,
                 "X": 0.7234,
                 "Y": 0.7234
             }
        ]
    }
 ]
}
```

● 多视频叠加

```
{
  "VideoTracks": [
     {
          "VideoTrackClips": [
              {
                  "MediaId": "222d9296e8864746a0b6f32dad6e****"
              },
              {
                  "MediaId": "333d9296e8864746a0b6f32dad6e****"
              }
         ]
     }
 ],
  "ImageTracks": [
     {
          "ImageTrackClips": [
             {
                 "ImageId": "001d9296e8864746a0b6f32dad6e****",
                 "Width": 0.1345,
                 "Height": 0.1678,
                  "X": 0.1234,
                  "Y": 0.1234
              },
              {
                  "ImageId": "002d9296e8864746a0b6f32dad6e****",
                 "Width": 0.1345,
                 "Height": 0.1678,
                 "X": 0.7234,
                 "Y": 0.7234
             }
         ]
     }
 ]
}
```

指定区域叠加示例

指定区域叠加指图片叠加在视频的指定时间区间。图片在输出视频中出现的位置由 x 和 y 设定,在输出视频中的宽高由 Width 和 Height 设定。在一个视频的第2秒~100秒叠加2张图片,示例如下:

⑦ 说明 输出视频指最终合成的成片视频。

● 单视频叠加

- 当 TimelineIn 不设置时,默认从0开始。
- 当 TimelineOut 不设置时,默认到视频的尾部。
- 当 TimelineOut 超过视频尾部(对单视频来说,即视频时长)时,超过的时间区间自动被忽略,仍以视频 轨的尾部为准。

```
{
  "VideoTracks": [
     {
         "VideoTrackClips": [
            {
                 "MediaId": "222d9296e8864746a0b6f32dad6e****"
              }
          ]
     }
 ],
  "ImageTracks": [
     {
          "ImageTrackClips": [
              {
                  "ImageId": "001d9296e8864746a0b6f32dad6e****",
                  "Width": 0.1345,
                  "Height": 0.1678,
                 "X": 0.1234,
                 "Y": 0.1234,
                  "TimelineIn":2,
                  "TimelineOut":100
              },
              {
                  "ImageId": "002d9296e8864746a0b6f32dad6e****",
                  "Width": 0.1345,
                 "Height": 0.1678,
                 "X": 0.7234,
                  "Y": 0.7234,
                  "TimelineIn":2,
                  "TimelineOut":100
             }
         ]
    }
 ]
}
```

• 多视频叠加图片

- 当 TimelineIn 不设置时,默认从0开始。
- 当 TimelineOut 不设置时,默认到视频的尾部。
- 当 TimelineOut 超过视频尾部(对多个视频来说,即视频轨总体时长)时,超过的时间区间,视频画面为 黑场。

如果视频轨有多个不同分辨率的源视频,那么:

- 输出视频的宽取多个源视频宽的最大值。
- 输出视频的高取多个源视频高的最大值。
- 画面采用pad+scale的方式,先加黑边保证实际画面比例不变形,再等比例放大到输出视频的宽高。
- 输出视频的时长为视频轨的时长。

```
{
  "VideoTracks": [
     {
          "VideoTrackClips": [
              {
                  "MediaId": "222d9296e8864746a0b6f32dad6e****"
              },
              {
                  "MediaId": "333d9296e8864746a0b6f32dad6e****"
              }
          ]
     }
 ],
  "ImageTracks": [
     {
          "ImageTrackClips": [
              {
                  "ImageId": "001d9296e8864746a0b6f32dad6e****",
                 "Width": 0.1345,
                 "Height": 0.1678,
                  "X": 0.1234,
                  "Y": 0.1234,
                  "TimelineIn":2,
                  "TimelineOut":100
              },
              {
                  "ImageId": "002d9296e8864746a0b6f32dad6e****",
                  "Width": 0.1345,
                  "Height": 0.1678,
                  "X": 0.7234,
                  "Y": 0.7234,
                  "TimelineIn":2,
                  "TimelineOut":100
              }
         ]
     }
 ]
}
```

10.5. 文字叠加

本文为您介绍文字叠加相关参数说明和全程叠加、指定区域叠加的示例,以文字叠加场景Timeline的各种数据组织 举例。

简介

视频剪辑的简介、总体流程和媒体合成服务的两种使用方式已在概述中介绍过,媒体合成最终都是通过视频合成接口发起并实现的,而Timeline是该服务重点处理的数据,也是视频剪辑中最核心的对象。Timeline内部参数可以组织成多种形式,以应对不同的业务场景。

参数说明

参数

说明

| 参数 | 说明 |
|--------------------|---|
| 文字出现的坐标 | X:文字左上角距离输出视频左上角的横向距离。 Y:文字左上角距离输出视频左上角的纵向距离。 支持百分比和像素两种形式: 当取值范围为0~0.9999时,表示相对输出视频宽/高的占比(X相对宽,Y相对高)。 当取值为≥8的整数时,表示绝对像素。 |
| 文字本身属性 | Content: 文本内容。 Font: 字体,该字体不填则默认为宋体。 目前支持的字体种类有: SimSun: 宋体。 WenQuanYi Zen Hei: 文泉驿正黑。 WenQuanYi Zen Hei Mono: 文泉驿等宽正黑。 WenQuanYi Zen Hei Sharp: 文泉驿点阵正黑。 Yuanti SC Bold: 圆体-简,粗体。 Yuanti SC Light: 圆体-简, 粗体。 Yuanti SC Regular: 圆体-简, 细体。 Yuanti SC Regular: 圆体-简,常规体。 FontSize: 字体大小,以像素为单位。该字段不填则默认为20。 FontColor: 字体颜色,形式为16进制颜色值,以#开头,如: #FFFFFF。该字段不填则默认为#FFFFFFFont。 ColorOpacity: 字体颜色的透明度,取值范围: 0~1, 1为完全不透明, 0为完全透明。该字段不填则默认为1。 FontFace: 文字的样式: Bold: 是否加粗。该字段不填则默认为false。 Italic: 是否斜体。该字段不填则默认为false。 Underline: 是否加下划线。该字段不填则默认为false。 |
| 文字在输出视频叠加的时间
区间 | TimelineIn:文字相对于时间线的入点。TimelineOut:文字相对于时间线的出点。 |

全程叠加示例

全程叠加指从片头到片尾叠加,即视频全程叠加文字不需要指定 TimelineIn 和 TimelineOut ,文字在输出视频中出现的位置由 x 和 y 设定。示例如下:

```
"Content": "测试文字",
                         "FontSize": 26,
                         "FontColorOpacity": 0.2,
                         "FontColor": "#000000",
                         "FontFace": {
                             "Bold": true,
                             "Italic": false,
                             "Underline": false
                         }
                     },
                      {
                         "Type": "Text",
                         "X": 30,
                         "Y": 92,
                         "Font": "WenQuanYi Zen Hei Mono",
                         "Content": "测试文字",
                         "FontSize": 26,
                         "FontColorOpacity": 1,
                         "FontColor": "#FFFFFF"
                     },
                      {
                         "Type": "Text",
                         "X": 0.8123,
                         "Y": 0.7896,
                         "Font": "WenQuanYi Zen Hei Mono",
                         "Content": "测试文字",
                         "FontSize": 26,
                         "FontColorOpacity": 0.2,
                         "FontColor": "#000000"
                     },
                      {
                         "Type": "Text",
                         "X": 0.8223,
                         "Y": 0.7796,
                         "Font": "WenQuanYi Zen Hei Mono",
                         "Content": "测试文字",
                         "FontSize": 26,
                         "FontColorOpacity": 1,
                         "FontColor": "#FFFFFF"
                     }
                ]
            }
         ]
     }
]
```

指定区域叠加

}

指定区域叠加指文字叠加在视频的指定时间区间,文字在输出视频中出现的位置由 x 和 y 设定。在一个视频 的第0~5、5~10秒叠加文字,示例如下:

- 当 TimelineIn 不设置时,默认从0开始。
- 当 TimelineOut 不设置时,默认到视频的尾部。
- 当 TimelineOut 超过视频尾部(对单视频来说,即视频时长)时,超过的时间区间自动被忽略,仍以视频轨

的尾部为准。

```
{
    "VideoTracks": [
       {
            "VideoTrackClips": [
                {
                    "MediaId": "ea9a6f9bdb68419abfd36a7113cf****",
                    "Effects": [
                       {
                            "Type": "Text",
                            "X": 31,
                            "Y": 93,
                            "TimelineIn": 0,
                            "TimelineOut": 5,
                            "Font": "WenQuanYi Zen Hei Mono",
                            "Content": "测试文字",
                            "FontSize": 26,
                            "FontColorOpacity": 0.2,
                            "FontColor": "#000000",
                            "FontFace": {
                                "Bold": true,
                                "Italic": false,
                                "Underline": false
                            }
                        },
                        {
                            "Type": "Text",
                            "X": 30,
                            "Y": 92,
                            "TimelineIn": 0,
                            "TimelineOut": 5,
                            "Font": "WenQuanYi Zen Hei Mono",
                            "Content": "测试文字",
                            "FontSize": 26,
                            "FontColorOpacity": 1,
                            "FontColor": "#FFFFFF"
                        },
                        {
                            "Type": "Text",
                            "X": 1124,
                            "Y": 516,
                            "TimelineIn": 5,
                            "TimelineOut": 10,
                            "Font": "WenQuanYi Zen Hei Mono",
                            "Content": "测试文字",
                            "FontSize": 26,
                            "FontColorOpacity": 0.2,
                            "FontColor": "#000000"
                        },
                        {
                            "Type": "Text",
                            "X": 1123,
                            "Y": 515,
                            "TimelineIn": 5,
                            "TimelineOut": 10,
                            "Font": "WenQuanYi Zen Hei Mono",
```



10.6. 特效

本文档为您介绍配置横幅文字和遮标的特效支持的配置参数,及特效的适用范围。

| 类型 | 名称 | 参数 | 适用范围 |
|------|------|---|------|
| Text | 横幅文字 | In: 文字相对于素材片段的入点。 Out: 文字相对于素材片段的出点。 TimelineIn: 文字相对于时间线的入点。 TimelineOut: 文字相对于时间线的出点。 X: 文字左上角距离输出视频左上角的横向
距离。 Y: 文字左上角距离输出视频左上角的纵向
距离。 Y: 文字左上角距离输出视频左上角的纵向
距离。 Y: 文字左h角距离输出视频本上角的纵向
距离。 Yi 文字在输出视频中的宽度。 Height: 文字在输出视频中的宽度。 Height: 文字在输出视频中的宽度。 FEHeight (可选): 剪辑时, 视频画面的实
际显示高。 FEWidth (可选): 剪辑时, 视频画面的实
际显示宽。 FEWidth (可选): 剪辑时, 视频画面的实
际显示宽。 FEWidth (可选): 剪辑时, 视频画面的实
际显示宽。 FEWidth (可选): 剪辑时, 视频画面的实
下显示宽。 FEWidth (可选): 剪辑时, 视频画面的实 FEMith (可选): 剪辑时, 视频画面的实 FEMith (可选): 京朝報时, 视频画面的实 FEMith (可选): 京和内容。 FontFace: 文字的样式: Bold: 是否加粗。 Italic: 是否加和 Italic: 是否如相。 Italic: 是否加下划线。 FontSize: 字体大小。 FontColorOpacity: 字体颜色的透明度 Italic: 特效信息。 Ying 参数详细描述, 请参 REffect: 特效信息。 | 视频 |

> 文档版本: 20220713

| 类型 | 名称 | 参数 | 适用范围 |
|-------------|----|--|------|
| DeWatermark | 遮标 | In: 遮标相对于素材片段的入点。 Out: 遮标相对于素材片段的出点。 TimelineIn: 遮标相对于时间线的入点。 TimelineOut: 遮标相对于时间线的出点。 X: 遮标左上角距离输出视频左上角的横向距离。 Y: 遮标左上角距离输出视频左上角的纵向距离。 Y: 遮标左h和出视频中的宽度。 Height: 遮标在输出视频中的宽度。 FEHeight(可选):剪辑时,视频画面的实际显示高。 FEWidth(可选):剪辑时,视频画面的实际显示宽。 SubType: 特效子类型。 | 视频 |

11.直播转点播 11.1. 概述

视频点播支持将直播流同步录制为点播视频,并支持媒资管理、媒体处理(转码、内容审核、智能封面等AI处理)、内容制作(云剪辑)、CDN分发加速等一系列操作。您可通过控制台配置工作流自动处理,也可通过 API/SDK灵活触发。本文为您介绍直播转点播的整体架构、优势、接入方式和查询方式。

整体架构

直播转点播的整体架构如下:



直转点优势

比起传统的录制到用户存储,录制到点播系统具有以下优势:

- 一站式的视频生产:录制到点播同时支持自动转码+CDN分发等,提供一站式视频生产流程。
- 丰富的媒资管理: 点播以视频维度进行录制视频管理, 提供丰富的媒资管理方式, 便于扩展。
- 强大的内容制作能力:利用点播的云剪辑等能力,可对录制的视频进行在线云剪辑,免去传统的制作流程。
- 视频AI等媒体处理:基于强大的AI能力,对视频进行智能审核、智能封面、视频DNA等处理。

接入方式

视频直播服务将直播内容进行录制存储,并转入点播系统中进行管理。

在一个直播加速域名下,直播录制配置按直播推流的 AppName 和 StreamName 进行区分,即相同 AppName

和 StreamName 下的视频流 (Stream) 都按此 AppName 和 StreamName 下的录制配置进行录制。

- 1. 开通视频点播服务。具体操作,请参见开通视频点播。
- 2. 登录视频点播控制台,在点播控制台左侧的导航栏选择配置管理,单击媒体处理配置>转码模板组>添加 转码模板组,在添加转码模板组页面创建所需要的转码模板,用于直播录制后的视频内容处理。更多信息, 请参见转码模板。

- 3. 开通视频直播服务。具体操作,请参见开通服务。
- 4. 登录视频云直播控制台,在直播控制台左侧的导航栏选择域名管理,在域名管理页面添加域名配置,在直播 控制台左侧的导航栏选择直播管理,单击录制文件管理>存储至VOD,在录制转码模板中选择刚才创建的 模板进行关联。具体操作,请参见直播录制存储至VOD。也可通过调用添加直播录制转点播配置接口,添加录制配 置。

```
? 说明
```

合成模板相关问题,请参见常见问题FAQ。

5. 完成上述配置工作后,配置好的直播流在接入时便会按照模板进行录制。

? 说明

为了避免录制时,因网络抖动或临时断流而导致录制文件被异常截断,系统会延迟断流180s。即如果断流之后在180s内重新推流,系统会默认是同一路录制流;超过180s则认为是两路录制流。

查询方式

直播录制到点播后,点播会将内容以视频维度进行管理,并且提供丰富的媒资处理能力。用户可以通过如下方式 获取到录制的视频列表:

- 控制台查看
 - 在点播控制台左侧的导航栏选择媒资库,单击音/视频,进入音/视频页面,视频来源选择直播录制进行筛选 查询。

| 音/视频♡ | | 类型♡ | 分类♡ | 状态♡ | 来源☑ 创建时间↓ 掛 |
|-------|--|-----|-----|------|--------------------------------------|
| C | AND AND CONTRACTORS AND | 视频 | 未分类 | ✓ 正常 | 全部
控制 点播上传 ^曾
短视频SDK |
| C 🕑 = | Constant interactional descention of the second sec | 视频 | 未分类 | ✔ 正常 | 剪辑合成
控制 ✔ 直播录制
控制台 |
| | | | | | 确认 重置 |

○ 在直播控制台左侧的导航栏选择直播管理,单击录制文件管理,进入录制文件管理页面,根据AppName与 StreamName进行查询。

| 2020-09-14 | - 2020-09-14 | İ | 全部AppName | 全部StreamName | |
|------------|--------------|--------|-----------|--------------|----|
| AppName | StreamName | 录制起止时间 | | 录制时长 | 操作 |
| | | | 没有数据 | | |

• 录制回调

点播系统会在直播录制时候进行回调通知,用户可以通过接收<mark>直转点视频录制完成</mark>事件回调信息,获取新增的录 制视频。

● 接口查询

点播提供了查询接口,用户可调用获取直转点视频列表接口,根据域名、 AppName 和 StreamName 来获取所需要的视频列表。

11.2. 常见问题FAQ

本文介绍直播转点播过程中的常见问题及处理方法。

录制时会生成多少点播视频?

在直播进行时,每到达一个录制周期,都会生成一个点播视频。同时,直播默认断流超过3分钟,会认为本次直播 结束,也会生成一个点播视频。

如:录制周期设置为30分钟,直播进行了38分钟后断开,那么会在点播生成两个视频,一个30分钟,一个8分钟。

点播转码模板做什么用?

每录制一个点播视频,点播服务可以自动对该视频进行转码处理,点播转码模板就是指定使用哪个转码配置来处 理录制的视频。

如:模板包含标清+高清两路流,则每录制一个视频,自动将该视频转出指定两路流。

○ 注意

转码模板需要提前在点播控制台进行创建。

可以设置不转码么?

使用点播的**不转码**模板作为转码模板即可,此时获取播放信息时使用的是原片播放。目前仅以下格式支持不转码 直接播放: MP4、FLV、M3U8、MP3、WEBM。

< ↓ 注意

通常不建议设置不转码。因为直播码率较高的情况下,卡顿会比较明显。如果选择了不转码,后续可以手动 提交媒体处理任务进行转码。媒体处理相关操作请参见<mark>音视频转码</mark>。

自动合成是什么意思?

由于每到达一个录制周期, 点播会自动生成一个视频。

如:一次直播1个小时,录制周期设置为30分钟,则会生成2个视频。如果开启了自动合成,则在本次直播结束 后,点播会自动将2个录制的视频进行合成,同时会将合成后的视频ID进行回调。即用户最后一共可以得到2+1=3 个视频。

自动合成有什么用?

通常为了快速回看,用户可能会将录制周期设置较短。

如:设置成10分钟,那么每过10分钟,则会生成一个点播视频,即10分钟后就可以回看前10分钟的点播内容。同时,整个直播结束后,用户希望得到一个完整的视频,则开启自动合成,可以帮用户自动进行视频合成(用户也可以手动调用云剪辑接口来实现)。

只有一个视频, 会合成吗?

开启自动合成后,不管多少个视频,都会进行合成。

如:用户设置录制周期为30分钟,实际直播时间为20分钟,则会在点播生成一个20分钟的录制视频。同时,会合成一个20分钟的视频(原视频只有一个)。

合成是异步的吗?

合成是异步的,需要一定的时间,可以捕捉响应的回调来判断合成状态,详情请参见场景实践。

合成转码模板ID是什么?

是开启自动合成后进行转码的模板ID。和录制设置的转码模板类似,开启自动合成后,点播可以自动对合成后的视频进行转码。

? 说明

此模板ID可以与录制模板ID不同。

如:为了快速回看,录制时使用的转码模板中,只有一路标清输出,但是合成的转码模板中,可以包含高清、超 清、2K等清晰度。即优先使用一个低码率用来回看视频,待直播结束后,再对完整视频进行多码率转码处理。

可以仅合成不转码吗?

使用点播的不转码模板作为转码模板即可。

自动合成最多支持多少视频?

目前自动合成最多支持40个视频。即如果录制周期设置为1小时,则最多支持40个小时的录制自动合成,超出会导 致本次自动合成失败。

录制周期设置为多少合适?

每到达一个录制周期,则会在点播生成一个视频。

如果不需要快速将前面的视频进行回看,可以使用默认推荐配置,即1小时。如果需要,视情况而定,通常不建议 低于20分钟。

11.3. 场景实践

直播转点播,可结合视频点播的转码、云剪辑、A处理、事件通知等功能,适应多种业务场景的需要。本文为您介绍3个典型搭配的实践方法。

名词解释

- 录制转码模板组: 直播录制到点播时, 点播会使用该模板组对视频进行转码操作。
- 合成转码模板组:多个录制视频进行自动合成时,点播会使用该模板组对视频进行合成+转码操作。
- 仅存储: 对直播内容进行录制后,不进行任何后续操作(不转码即分发模板组)。
- 仅合成:对直播内容进行合成后,不进行任何后续操作(不转码即分发模板组)。
- 直播录制周期:直播录制到点播的录制周期。例如:需要在一场3小时的直播过程中提供点播服务,则可设置录制周期为1小时;即直播1小时后,可提供第1小时内已录制内容的点播观看服务。

实践1:直播录制+自动转码+CDN加速

直播录制后快速将录制文件进行转码和CDN加速,供用户进行点播播放。

适用场景:适用于大部分直播场景(不需要对内容进行二次加工)。

流程如下:



- 1. 客户进行直播推流。
- 2. 推流达到一个录制周期,则会自动将录制文件添加到点播系统。
- 点播记录完成后,会生成点播系统的唯一视频ID,并将该视频信息回调给用户,即 AddLiveRecordVideoComplete通知,并附带直播相关的DomainName、AppName、StreamName信息。客 户收到回调后,需记录该视频信息,并以该VideoId作为索引进行后续视频状态更新。
- 4. 点播系统检测用户录制配置中的录制转码组ID(该转码组中含有具体码流转码任务),进行对应的转码操作。
- 5. 截图完成、单个码流转码完成、全部码流转码完成时,会给用户进行回调(回调顺序无时序),用户需根据 回调信息中的Videold进行视频状态更新。转码完成后,即可进行后续的播放操作(转码回调信息中含播放地 址或后续通过Get PlayInf o接口根据Videold进行播放地址获取,该播放地址已经经过CDN加速)。

实践2:直播录制+仅存储到点播+手动发起转码+CDN加速

部分用户希望将直播录制的视频仅先存储到点播,暂缓后续的转码操作,则可在新建录制配置时,选择仅存储模 板组。如果后续希望对视频进行转码,则可进行手动触发转码操作。同时,可配合点播云剪辑功能进行使用,效 果更佳。

? 说明

仅存储模板组需联系点播进行激活。

适用场景:直播完成后,需要对内容进行二次加工,如体育赛事、游戏直播剪辑等,后续由用户主动发起转码和 CDN加速全流程(转码完成后点播会自动对输出文件进行CDN加速)。

流程如下:



- 1. 客户进行直播推流。
- 2. 推流达到一个录制周期,则会自动将录制文件添加到点播系统。
- 点播记录完成后,会生成点播系统的唯一视频ID,并将该视频信息回调给用户,即 AddLiveRecordVideoComplete通知,并附带直播相关的DomainName、AppName、StreamName信息。客 户收到回调后,需记录该视频信息,并以该Videold作为索引进行后续视频状态更新。
- 4. 用户对该视频手动触发转码操作(调用转码任务API),在此之前可进行云剪辑等操作。
- 5. 截图完成、单个码流转码完成、全部码流转码完成时,会给用户进行回调(回调顺序无时序),用户需根据 回调信息中的Videold进行视频状态更新。转码完成后,即可进行后续的播放操作(转码回调信息中含播放地 址或后续通过Get PlayInf o接口根据Videold进行播放地址获取,该播放地址已经经过CDN加速)。

实践3:直播录制+多周期视频自动合并

部分用户希望将自己录制周期生成的多个文件(如录制周期为20分钟,一次直播1个小时,则会生成3个视频)进 行视频合成,再将合成后的视频进行处理,点播提供了自动合成的功能。可在新建直播录制配置时,将自动合成 开关打开,并且配置进行合成时所使用的合成转码模板组(也可配置为仅合成,后续由用户触发转码,与前两节 所述区别相同)。点播会在用户断流超过指定时间(可由直播进行配置断流超时时间)后,进行本次直播的视频 自动合成并根据转码配置进行后续操作。

• 合成+自动转码

适用场景:一场直播完成后,需要对所有录制周期内的分段进行自动合并,同时发起转码等全流程。如体育赛 事、教育多节授课合并等。

流程如下:



- i. 客户进行直播推流。
- ii. 推流达到一个录制周期,则会自动将录制文件添加到点播系统。
- iii. 点播记录完成后,会生成点播系统的唯一视频ID,并将该视频信息回调给用户,即 AddLiveRecordVideoComplete通知,并附带直播相关的DomainName、AppName、StreamName信息。 客户收到回调后,需记录该视频信息,并以该VideoId作为索引进行后续视频状态更新。
- iv. 客户断流超时或主动触发断流。
- v. 点播收到直播发送的本次直播结束消息。
- vi. 点播系统检测用户录制配置中的合成配置,判断是否需要发起自动合成。如果需要,则按照录制配置中的 合成转码组进行合成和转码。
- vii. 视频开始合成,点播会生成一个合成后视频的唯一视频ID,并将该视频信息回调给用户,即 LiveRecordVideoComposeStart通知,并附带直播相关的DomainName、AppName、StreamName信息。 客户收到回调后,需记录该视频信息,并以该Videold作为索引进行后续视频状态更新。
- viii. 视频源文件合成完成后,点播会将该状态回调给用户,即FileUploadComplete通知。
- ix. 截图完成、单个码流转码完成、全部码流转码完成时,会给用户进行回调(回调顺序无时序),用户需根据回调信息中的Videold进行视频状态更新。转码完成后,即可进行后续的播放操作(转码回调信息中含播放地址或后续通过Get PlayInf o接口根据Videold进行播放地址获取,该播放地址已经经过CDN加速)。
- 合成+手动发起转码

可在新建直播录制配置时,将自动合成开关打开,并且配置进行合成时所使用的合成转码模板组(本情况配置 成仅合成,后续由用户触发转码)。

适用场景:一场直播完成后,需要对所有录制周期内的分段进行自动合并,合并后文件不做任何处理,由用户进行二次加工,如云剪辑等,后续再进行手动转码,如视频内嵌广告、体育赛事空挡部分内容剪切等。 流程如下:



- i. 客户进行直播推流。
- ii. 推流达到一个录制周期,则会自动将录制文件添加到点播系统。
- iii. 点播记录完成后,会生成点播系统的唯一视频ID,并将该视频信息回调给用户,即 AddLiveRecordVideoComplete通知,并附带直播相关的DomainName、AppName、StreamName信息。 客户收到回调后,需记录该视频信息,并以该Videold作为索引进行后续视频状态更新。
- iv. 客户断流超时或主动触发断流。
- v. 点播收到直播发送的本次直播结束消息。
- vi. 点播系统检测用户录制配置中的合成配置,判断是否需要发起自动合成。如果需要,则按照录制配置中的 合成转码组进行合成和转码。
- vii. 视频开始合成,点播会生成一个合成后视频的唯一视频ID,并将该视频信息回调给用户,即 LiveRecordVideoComposeStart通知,并附带直播相关的DomainName、AppName、StreamName信息。 客户收到回调后,需记录该视频信息,并以该Videold作为索引进行后续视频状态更新。

- viii. 视频源文件合成完成后,点播会将该状态回调给用户,即FileUploadComplete通知。此时,代表本次录制 合成的源文件已经处于正常就绪状态,用户可对该视频进行后续的转码触发等操作。
- ix. 用户对该视频手动触发转码操作(调用转码任务API),在此之前可进行云剪辑等操作。
- x. 截图完成、单个码流转码完成、全部码流转码完成时,会给用户进行回调(回调顺序无时序),用户需根据回调信息中的Videold进行视频状态更新。转码完成后,即可进行后续的播放操作(转码回调信息中含播放地址或后续通过GetPlayInfo接口根据Videold进行播放地址获取,该播放地址已经经过CDN加速)。

12.事件通知

12.1. 概述

视频点播支持HTTP回调和MNS回调两种方式,本文为您介绍事件通知的事件类型、回调方式、回调配置、回调协议、回调内容公共参数等基本信息,以及回调判断与重试和常见问题。

事件通知简介

点播服务支持多个存储区域,每个区域可以单独配置事件通知的回调方式和回调地址。用户可以上传视频到不同 区域的存储,视频处理完成后,点播服务会根据存储区域配置的回调方式和回调地址及时通知用户。

事件类型

目前支持的事件通知类型参考如下:

| 事件类型 | 参考文档 |
|-----------------------------|-------------|
| FileUploadComplete | 视频上传完成 |
| ImageUploadComplete | 图片上传完成 |
| StreamTranscodeComplete | 单个清晰度转码完成 |
| TranscodeComplete | 视频转码完成 |
| SnapshotComplete | 视频截图完成 |
| DynamicImageComplete | 视频动图完成 |
| AddLiveRecordVideoComplete | 直转点视频录制完成 |
| LiveRecordVideoComposeStart | 直转点录制视频合成开始 |
| UploadByURLComplete | URL上传视频完成 |
| CreateAuditComplete | 人工审核完成 |
| AlMediaAuditComplete | 智能审核完成 |
| VideoAnalysisComplete | 音视频分析完成 |

| 事件类型 | 参考文档 |
|-----------------------------|----------|
| AIMediaDNAComplete | 视频DNA完成 |
| AlVideoTagComplete | 智能标签作业完成 |
| AttachedMediaUploadComplete | 辅助媒资上传完成 |
| ProduceMediaComplete | 媒体合成完成 |
| DeleteMediaComplete | 媒体删除完成 |
| MediaBaseChangeComplete | 媒资基础信息变更 |

回调方式

点播服务目前支持HTTP回调(兼容HTTPS)、MNS回调两种方式获取事件通知:

● HTTP回调(兼容HTTPS):

您需要部署一个HTTP服务来接收回调消息,并在点播控制台上配置回调URL;当事件产生时,点播服务端会向该URL发起HTTP POST请求,具体内容将通过HTTP Body送达。

● MNS回调:

您需要授权<mark>点播服务访问MNS</mark>,然后登录<mark>消息服务控制台</mark>,创建队列或使用已有队列,通过点播控制台或调用 接口完成事件通知配置;当事件产生时,点播服务会将回调内容写入该队列,具体接收消息参见队列控制台操作 帮助。更多信息,请参见MNS可靠事件通知。

两种方式对比

| 对比项 | HTTP回调 | MNS回调 |
|-----|---|-------------------------------------|
| 可靠性 | 若您的HTTP消息接收服务存在异常(崩溃或重
启),则消息可能会丢失。 | 更为可靠 ,只要配置正常,消息回调几乎都能
成功。 |
| 安全性 | HTTP方式都可回调该地址,但可配合回调鉴权
过滤非法请求,以增强安全性。
如何配置,请参见HTTP回调鉴权。 | 由于仅授权方能读写消息队列 <i>,</i> 安全性更高。 |
| 便捷性 | 接入更为方便 ,只需简单部署消息接收服务。 | 需要额外开通和配置MNS,并且需要开发和部
署消费消息的程序。 |

选择哪种回调方式,取决于您的使用场景,综合考虑。

回调配置

登录视频点播控制台,在左侧导航栏选择**配置管理>媒体处理配置>回调设置**,进入**回调设置**页面,配置回调 URL、队列及选择需要回调的事件。具体操作,请参见回调设置。

◯ 注意

勾选视频AI处理完成后,所有的AI事件如AIMediaAuditComplete、AIMediaDNAComplete、AIVideoTagComplete等都会触发通知。

| 审核设置 | | • | 视频点播 / 回调设 | 8 | |
|--------|--------|----|------------|---------------------|--|
| 配置管理 | | | 回调设置 | 2
L | |
| 媒资管理配置 | \sim | | () 视频处理 (如 | 11上传、转码)每个环节完成后,通过酢 | 出置的回调地址及时获取处理进展和状态,以便进行其他业务操作。 了解回调配置 |
| 媒体处理配置 | ^ | | 回调设置 🗾 💆 | 修改设置 | |
| 转码设置 | | | 回调方式 | HTTP请求 | |
| 截图模板 | | | 回调URL | 当产生回调事件时,点播服务端会向 | httpCallBack/httpCallback/
i该URL发起HTTP POST请求 |
| 动图模板 | | < | 回调事件 | ✓ 视频上传完成 | 单个清晰度转码完成 |
| 水印管理 | | Ľ. | | ✓ 全部清晰度转码完成 | ✓ 视频截图完成 |
| 丁作流管理 | | | | ✓ 视频AI处理完成 | 直播录制完成 |
| THMEN | | | | 人工审核完成 | ✓ URL上传视频完成 |
| 回调设置 | | | | 剪辑合成完成 | 录制视频合成开始 |
| 分发加速配置 | ~ | | | ✓ 图片上传完成 | 音视频分析完成 |
| | | | | 媒体删除完成 | 辅助媒资上传完成 |
| 数据中心 | | | | 媒资基础信息变更完成 | ✓ 视频动图完成 |
| | | | | 您选择的回调事件处理完成后,点播 | 服务会进行事件通知,事件通知的实现方式请参考事件通知开发指南 |
| 监控统计 | \sim | | | | |
| 用量查询 | | | 回调鉴权 | | |
| 日志管理 | | | 回调鉴权 | | |
| 媒资数据导出 | | | | 在HTTP回调时增加特定签名头,供回 | 调调消息接收服务端进行签名认证,防止非法或无效请求,详情参考 回调鉴权文档 |

回调协议

- HTTP回调
 - 。请求:HTTP POST请求,包体内容为JSON,每一种回调的具体包体内容参见各自文档。
 - 应答: 点播服务会忽略应答包内容。
- MNS回调

接收消息内容为JSON,每一种回调的具体包体内容参见各自文档。

回调内容公共参数

| 参数名称 | 类型 | 描述 |
|-----------|--------|---|
| EventTime | String | 事件产生时间,为UTC时间: yyyy-MM-
ddTHH:mm:ssZ。 |
| EventType | String | 事件类型。 |
| Videold | String | 视频ID。 |

| 参数名称 | 类型 | 描述 |
|--------|--------|--|
| Status | String | 处理状态,取值:
• success(成功)。
• fail(失败)。 |
| Extend | String | 在上传或提交作业接口中,指定UserData;并且若里面包含
Extend字段,会在事件完成回调时透传返回用户自定义数
据,最大长度512字节。 |

回调判断与重试

- HTTP回调
 - 点播服务发起HTTP回调时,若回调接收服务响应的HTTP状态码为200即视为回调成功;响应状态码不为200,或是响应时间超过5秒出现超时,都视为回调失败。点播服务会忽略响应的包体内容,仅以HTTP状态码为准。
 - 若由于配置异常(比如您的回调地址错误、消息接收服务异常等),导致消息回调失败,点播服务会继续重 试回调2次,即总共最多回调3次;超过后会丢弃。
- MNS回调
 - 点播服务发起MNS回调时,只要消息写入MNS队列成功即视为回调成功,否则为失败。
 - 若由于配置异常(比如未授权点播服务访问、队列地址不是公网或队列名称不对等),导致消息写入MNS失败,点播服务会继续重试2次,即总共最多回调3次;超过后会丢弃。

? 说明

点播服务在回调失败后会间隔1秒再次发起重试,若总共3次回调仍失败则会丢弃该消息;如要确保可靠通知,建议使用MNS回调,其可靠性更高,只要配置正常,几乎能确保消息回调成功。

常见问题

使用过程中遇到问题,请参见事件通知。

12.2. HTTP回调事件通知

本文为您介绍HTTP回调事件通知的配置方法及鉴权原理。

HTTP回调鉴权原理

点播服务支持在HTTP(含HTTPS)回调时增加特定签名头,供回调消息接收服务端进行签名认证,以防止非法或 无效请求。

鉴权参数

在回调HTTP头部增加的具体鉴权参数如下:

| 字段 | 描述 |
|-----------------|--|
| X-VOD-TIMESTAMP | UNIX时间戳,整形正数,固定长度10,1970年1月1日以来的秒数,表示回调请求
发起时间。 |

| 字段 | 描述 |
|------------------|-----------------------------|
| X-VOD-SIGNAT URE | 签名字符串,为32位MD5值,详细说明见下文签名算法。 |

签名算法

X-VOD-SIGNATURE的计算依赖如下字段:

| 字段 | 示例 | 描述 |
|-----------------|---------------------------------------|--|
| 回调URL | https://www.example.com/your/callback | 用户设置的回调地址。 |
| X-VOD-TIMESTAMP | 1519375990 | UNIX时间戳,整形正数,固定长度10,1970年
1月1日以来的秒数,表示回调请求发起时间。 |
| PrivateKey | test123 | 用户预设的签名Key。 |

将上述三个字段进行拼接,字段中间以竖线())分割,后计算MD5值,即:

MD5Content = 回调URL|X-VOD-TIMESTAMP|PrivateKey X-VOD-SIGNATURE = md5sum(MD5Content)

X-VOD-SIGNATURE字段计算方法示例如下:

X-VOD-SIGNATURE = md5sum(https://www.example.com/your/callback|1519375990|test123) = c72b6089414 0fa98920f1279219b****

回调消息接收服务端校验规则

- 回调消息接收端将回调所设置的回调URI、X-VOD-TIMESTAMP取值、PrivateKey字符串拼接后,进行MD5值计算,得到加密串,再将加密串与X-VOD-SIGNATURE字段进行对比,如果不一致,则请求非法。
- 回调消息接收端获取当前时间,与回调请求所带的X-VOD-TIMESTAMP字段时间相减,如果超过服务端所设定的 指定时间(如5分钟,由服务端自行定义),则认为该请求无效。

⑦ 说明 由于时间设置等问题,时间差值可能会有误差,服务端可自行决定是否进行该校验。

PrivateKey切换

客户在切换PrivateKey时,为保证回调功能不受影响,消息接收服务端需要兼容新旧两个PrivateKey的平滑切换,即在一段时间内兼容新旧两个PrivateKey的鉴权,由服务方完成。

建议操作顺序如下:

- 1. 用户定义新的PrivateKey。
- 2. 用户升级回调消息接收服务端,兼容新、旧两个PrivateKey的鉴权。
- 3. 在点播控制台将PrivateKey更新成最新。
- 4. 观察一段时间后,回调消息接收服务端去掉对原来PrivateKey的兼容。
- 5. 切换完成。

注意事项

 回调鉴权是否开启由用户决定(建议开启)。一旦设置了PrivateKey,则回调时会携带所有鉴权相关内容,供回 调消息接收服务端进行鉴权使用,即设置PrivateKey不会影响原有功能,具体是否校验由用户决定。 • 未设置PrivateKey的用户不会受任何影响。

配置HTTP回调

支持通过点播控制台和API两种方式配置HTTP回调,具体如下:

控制台

登录<mark>视频点播控制台,选择配置管理 > 媒体处理配置 > 回调设置</mark>。详细设置请参见<mark>回调设</mark>置。

API

调用设置事件通知配置接口完成事件通知配置。

12.3. MNS可靠事件通知

阿里云消息服务MNS(Message Service)是一种高效、可靠、安全、便捷、可弹性扩展的分布式消息服务。本文 为您介绍MNS可靠事件通知的注意事项、配置流程和消费消息。

背景信息

- MNS提供了队列模型,支持多个生产者和消费者并发访问同一个队列,并能确保某条消息在取出之后的特定时间段内,无法被其他消费者获得。消息被消费后一段时间内不可见,需要用户主动删除,否则该消息会被再次消费。具体介绍,请参见什么是消息服务MNS。
- 点播服务支持多个存储地域(中国内地、新加坡等),每个区域可以单独配置事件通知的回调方式和回调地址。用户可以上传视频到不同区域的存储,视频处理(如上传、转码)完成后,点播服务会根据存储区域配置的回调及时通知用户,并将内容推送到用户MNS服务的队列中。

注意事项

- 点播服务发起MNS回调时,若未授权点播服务访问、Endpoint不是公网或队列名称不对导致消息写入失败,点播服务会继续重试回调2次,即总共最多回调3次,3次回调都失败则会丢弃该消息。
- 消息队列推荐区域:
 - 如果视频保存在中国内地区域(如华北2、华东2等),建议使用华东2(上海)区域的队列,推送消息到
 非华东2(上海)区域的队列会存在较短时间的延迟。
 - 如果视频保存在其他区域(如新加坡、印度尼西亚等),建议创建或使用相应区域的消息队列。

如:视频存储区域为印度(孟买),则应创建或使用印度(孟买)区域的消息队列。

配置MNS可靠事件通知

1. 授权视频点播服务VOD访问消息服务MNS。

可通过以下两种方式进行授权:

○ 方式一、直接授予VOD对您云资源相应的访问权限,包含OSS、MNS、CDN以及KMS的部分权限。

登录阿里云控制台后,访问云资源访问授权页面,单击同意授权,直接进行授权。

| • 6 | 云资源访问授权
如需修改角色权限,请前往 RAM 控制台 <mark>角色管理</mark> 中设置,需要注意的是,错误的配置可能导致 CloudMonitor 无法获取到必要的权
限。 | | |
|--|--|--|--|
| VOD 请求获取 下方是系统创建 | 访问您云资源的权限。
的可供 VOD 使用的角色,授权后、VOD 拥有对您云资源相应的访问权限。 | | |
| ト方是系統创建的可供 VOD 使用的角色,授权后, VOD 拥有对您云资源相应的访问权限。
AliyunVODDefaultRole 展开详情 | | | |
| 同意授权 取消 | | | |

- 方式二、给VOD服务的用户、用户组或角色授于消息服务(MNS)的访问权限,即将系统策略中的AliyunMNSFullAccess(管理消息服务(MNS)的权限)或AliyunMNSReadOnlyAccess(管理消息服务(MNS)的权限)添加至授权策略中,详细操作请参见账号和授权。
- 2. 在消息服务 (MNS) 控制台中创建队列。

登录消息服务控制台创建队列或使用已有队列。具体操作,请参见队列控制台操作帮助。

⑦ 说明 建议遵循注意事项中的推荐区域创建或使用队列。

3. 配置事件通知。

可通过点播控制台或API两种方式配置,具体如下:

○ 控制台

·登录视频点播控制台,选择配置管理 > 媒体处理配置 > 回调设置。详细设置请参见回调设置。

API

调用<mark>设置事件通知配置</mark>接口完成事件通知配置。

4. 触发回调事件。

完成事件通知配置后,登录视频点播控制台,通过上传媒资(音/视频或图片)来触发需要回调的事件。

5. 在消息服务控制台查看消息。

当满足产生回调事件条件时,点播服务端会将回调内容写入您提供的队列,登录消息服务控制台,单击对应 队列的**接收消息**按钮,从**接收消息**窗口中查看消息内容。

消费消息

配置完成后,可以通过如下代码消费消息:

- 如何使用Java代码消费消息,请参见队列使用手册。
- 如何使用Python代码消费消息,请参见队列使用手册。
- 如何使用C#代码消费消息,请参见队列使用手册。
- 如何使用PHP代码消费消息,请参见队列使用手册。
- 其他语言可以通过调用接口来获取消息内容,首先掌握调用方式后,调用消费消息接口获取消息内容,调用删 除消息接口删除消息。

队列支持多个生产者和消费者并发访问同一个队列,并能确保某条消息在取出之后的特定时间段内,无法被其他 消费者获得。消息被消费后一段时间内不可见,需要用户主动删除,否则该消息会被再次消费。

12.4. 事件列表

12.4.1. 视频上传完成

本文为您介绍视频上传完成事件、事件通知的内容和回调示例。

事件类型

FileUploadComplete

事件说明

点播服务端接收完上传的视频文件后,会产生FileUploadComplete事件。

事件通知内容

| 参数名称 | 类型 | 必备项 | 描述 |
|-----------|--------|-----|---|
| EventTime | String | 是 | 事件产生时间,为UTC时间:yyyy-MM-ddTHH:mm:ssZ。 |
| EventType | String | 是 | 事件类型,系统规定参数。固定
为:FileUploadComplete。 |
| Videold | String | 是 | 视频ID。 |
| Size | Long | 否 | 上传的文件大小,单位:Byte(字节)。 |
| FileUrl | String | 否 | 上传文件的URL地址。 |

⑦ 说明 由于要支持断点续传,点播服务端无法确定您是暂时中断上传还是上传失败,故没有上传失败的事件通知。

回调示例

回调示例说明:

- 对于HTTP回调,以下内容为HTTP Post Body。
- 对于MNS回调,以下内容为消息体。

```
{
    "EventTime": "2017-03-20T07:49:17Z",
    "EventType": "FileUploadComplete",
    "VideoId": "43q91jdh7df****",
    "Size": 1439213,
    //FileURL示例: http://example-bucket-****.oss-cn-shanghai.aliyuncs.com/27ffc438-164h67f57ef-0
005-6884-51a-1****.mp4
    "FileUrl":"<your File URL>"
}
```

12.4.2. 图片上传完成

本文为您介绍图片上传完成事件、事件通知的内容和回调示例。

事件类型

ImageUploadComplete

事件说明

图片文件上传成功后, 会产生ImageUploadComplete事件。

事件内容

| 参数名称 | 类型 | 必备项 | 描述 |
|-----------|--------|-----|--|
| EventTime | String | 是 | 事件产生时间,为UTC时间: yyyy-MM-ddTHH:mm:ssZ。 |
| EventType | String | 是 | 事件类型,系统规定参数。固定
为:ImageUploadComplete。 |
| ImageId | String | 是 | 图片ID。 |
| Size | Long | 否 | 上传的文件大小,单位:Byte(字节)。 |
| FileURL | String | 否 | 上传图片文件的URL地址。 |
| Status | String | 是 | 是否上传成功。
• success:成功。
• fail:失败。 |

⑦ 说明 由于要支持断点续传,点播服务端无法确定您是暂时中断上传还是上传失败,故没有上传失败的事件通知。

回调示例

回调示例说明:

- 对于HTTP回调,以下内容为HTTP Post Body。
- 对于MNS回调,以下内容为消息体。

```
{
    "Status": "success",
    //FileURL示例: https://example-bucket-****.oss-cn-shanghai.aliyuncs.com/image/default/AF8018C
2908A434HD73JA678D08B0E****.jpg
    "FileURL": "<your File URL>",
    "AppId": "app-1000000",
    "EventType": "ImageUploadComplete",
    "EventTime": "2017-03-20T07:49:17Z",
    "Size": 105520,
    "ImageId": "59bdbf10df76a8d8b6137a5ad12****"
}
```

12.4.3. 单个清晰度转码完成

本文为您介绍单个清晰度转码完成事件、事件通知的内容和回调示例。

事件类型

StreamTranscodeComplete

事件说明

视频的某个清晰度、某种格式的流(如:标清的MP4格式)转码完成时会产生此事件。目前单个清晰度至少1种格式的转码完成即可播放。

⑦ 说明 如果开启了URL鉴权,则需要自己生成auth_key才能访问播放地址,否则会返回 HTTP 403 。关于URL鉴权的信息,请参见URL鉴权。

事件内容

| 参数名称 | 类型 | 必备项 | 描述 |
|--------------|---------|-----|--|
| EventTime | String | 是 | 事件产生时间,为UTC时间: yyyy-MM-ddTHH:mm:ssZ。 |
| EventType | String | 是 | 事件类型,系统规定参数。固定
为:StreamTranscodeComplete。 |
| Videold | String | 是 | 视频ID。 |
| Status | String | 是 | 视频流转码状态。
● success:成功。
● fail:失败。 |
| Bitrate | String | 否 | 视频流码率,单位:Kbps。 |
| Definition | String | 否 | 视频流清晰度定义。
• FD: 流畅。
• LD: 标清。
• SD: 高清。
• HD: 超清。
• OD: 原画。
• 2K: 2K。
• 4K: 4K。
• AUTO: 自适应码流。 |
| Duration | Float | 否 | 视频流长度,单位:秒。 |
| Encrypt | Boolean | 否 | 视频流是否为加密流。 |
| ErrorCode | String | 否 | 作业错误码,视频流转码出错时,会有该字段。 |
| ErrorMessage | String | 否 | 作业错误信息,视频流转码出错时,会有该字段。 |
| FileUrl | String | 否 | 视频流的播放地址。不带鉴权的auth_key,如果开启了URL
鉴权,则需要自己生成auth_key才能访问。 |
| Format | String | 否 | 视频流格式。
● mp4。
● m3u8。 |

| 参数名称 | 类型 | 必备项 | 描述 |
|--------|--------|-----|---|
| Fps | String | 否 | 视频流帧率,每秒多少帧。 |
| Height | Long | 否 | 视频流高度,单位:px。 |
| Size | Long | 否 | 视频流大小,单位: Byte。 |
| Width | Long | 否 | 视频流宽度,单位:px。 |
| Jobid | String | 否 | 转码作业ID。 |
| Extend | String | 否 | 用户自定义回调透传数据,更多信息,请参见 <mark>请求参数示</mark>
<mark>例</mark> 。 |

回调示例

回调示例说明:

- 对于HTTP回调,以下内容为HTTP Post Body。
- 对于MNS回调,以下内容为消息体。

```
{
   "EventTime": "2017-03-20T07:49:17Z",
   "EventType": "StreamTranscodeComplete",
   "VideoId": "43q9fj74hdf****",
  "Status": "success",
   "Bitrate": "925",
   "Definition": "LD",
   "Duration": 15.0,
  "Encrypt": false,
  //FileURL示例: http://example.aliyundoc.com/DBEBDEAJS73J79BE4D****/52a53151eba5js73ke2da3b55bc
5****.mp4
  "FileUrl": "<your File URL>",
   "Format": "mp4",
   "Fps": "30",
   "Height": 960,
   "Size": 1815321,
  "Width": 540,
   "JobId":"dddddddddd",
   "Extend":"test data"
}
```

12.4.4. 全部清晰度转码完成

本文为您介绍全部清晰度转码完成事件、事件通知的内容和回调示例。

事件类型

TranscodeComplete

事件说明

视频所有流转码完成会产生TranscodeComplete事件。

⑦ 说明 如果开启了URL鉴权,则需要自己生成auth_key才能访问播放地址,否则会返回 HTTP 403 。 URL鉴权的信息,请参见URL鉴权。

事件内容

| 参数名称 | 类型 | 必备项 | 描述 |
|-------------|--------|-----|--|
| EventTime | String | 是 | 事件产生时间,为UTC时间:yyyy-MM-ddTHH:mm:ssZ。 |
| EventType | String | 是 | 事件类型,系统规定参数。固定
为:TranscodeComplete。 |
| Videold | String | 是 | 视频ID。 |
| Status | String | 是 | 视频转码状态,只要有一路流转码成功即为成功。 success:成功。 fail:失败。 |
| Extend | String | 否 | 用户自定义回调透传数据。更多信息,请参见 <mark>请求参数说</mark>
<mark>明</mark> 。 |
| StreamInfos | Array | 否 | 请参见下文视频流信息表。 |

StreamInfos为数组,每个流信息的字段如下:

视频流信息表

| 名称 | 类型 | 必选项 | 描述 |
|------------|---------|-----|--|
| Status | String | 否 | 视频流转码状态。
● success:成功。
● fail:失败。 |
| Bitrate | String | 否 | 视频流码率,单位:Kbps。 |
| Definition | String | 否 | 视频流清晰度定义。
• FD: 流畅。
• LD: 标清。
• SD: 高清。
• HD: 超清。
• OD: 原画。
• 2K: 2K。
• 4K: 4K。
• AUTO: 自适应码流。 |
| Duration | Float | 否 | 视频流长度,单位:秒。 |
| Encrypt | Boolean | 否 | 视频流是否为加密流。 |
| ErrorCode | String | 否 | 作业错误码,视频流转码出错时,会有该字段。 |

| 名称 | 类型 | 必选项 | 描述 |
|--------------|--------|-----|--|
| ErrorMessage | String | 否 | 作业错误信息,视频流转码出错时,会有该字段。 |
| FileUrl | String | 否 | 视频流的播放地址。不带鉴权的auth_key,如果开启了URL
鉴权,则需要自己生成auth_key才能访问。 |
| Format | String | 否 | 视频流格式。
● mp4
● m3u8 |
| Fps | String | 否 | 视频流帧率,每秒多少帧。 |
| Height | Long | 否 | 视频流高度,单位:px。 |
| Size | Long | 否 | 视频流大小,单位: Byte。 |
| Width | Long | 否 | 视频流宽度,单位:px。 |
| Jobid | String | 否 | 转码作业ID。 |

回调示例

回调示例说明:

- 对于HTTP回调,以下内容为HTTP Post Body。
- 对于MNS回调,以下内容为消息体。
```
{
 "EventTime": "2017-03-20T07:49:17Z",
 "EventType": "TranscodeComplete",
 "VideoId": "43q9fjsh73f****",
 "Status": "success",
 "Extend":"test data",
  "StreamInfos":
  [
  {
     "Status": "success",
    "Bitrate": "925",
    "Definition": "LD",
    "Duration": 15,
    "Encrypt": false,
    "FileUrl": "http://example.aliyundoc.com/ABEBDE1JSU79FD4D1329/62cb3151eba52js82j2da3b55bc5*
***.mp4",
     "Format": "mp4",
    "Fps": "30",
    "Height": 960,
    "Size": 1815321,
    "Width": 540,
    "JobId":"fffffffff
  },
  {
    "Status": "success",
    "Bitrate": "1575",
    "Definition": "SD",
    "Duration": 15,
    "Encrypt": false,
    "FileUrl": "http://example.aliyundoc.com/ABEBDE1JSU79FD4D1329/62cb3151eba52js82j2da3b55bc5*
***.mp4",
    "Format": "mp4",
    "Fps": "30",
    "Height": 960,
    "Size": 3090951,
    "Width": 540,
    "JobId":"dddddddddd"
  }
 1
}
```

12.4.5. 视频截图完成

本文为您介绍视频截图完成事件、事件通知的内容和回调示例。

事件类型

Snapshot Complete

事件说明

视频截图完成后会产生SnapshotComplete事件。

• 截图和转码是并行处理,二者无法确定先后顺序。

- 如果是封面截图类型,并且没有设置视频封面(CoverUrl),则默认取截图的中间一张为视频封面。关于封面截 图,更多信息,请参见视频截图。
- 您也可以通过获取视频信息接口,获取视频封面和CoverSnapshot类型的截图地址。
- 您还可以通过查询截图数据接口,获取指定视频最新一次截图的截图地址。

? 说明

如果开启了URL鉴权,则需要自己生成auth_key才能访问图片地址,否则会返回HTTP 403。URL鉴权的信息, 请参见URL鉴权。

| 参数名称 | 类型 | 必备项 | 描述 |
|---------------|----------------|-----|--|
| EventTime | String | 是 | 事件产生时间,为UTC时间:yyyy-MM-ddTHH:mm:ssZ。 |
| EventType | String | 是 | 事件类型,系统规定参数。固定
为:SnapshotComplete。 |
| Videold | String | 是 | 视频ID。 |
| Status | String | 是 | 视频截图状态。
• success:成功。
• fail:失败。 |
| SubType | String | 否 | 截图子类型。取值: SpecifiedTime。
⑦ 说明
SubType=SpecifiedTime,表示通过提交媒体截图作
业接口发起的截图。 |
| ErrorCode | String | 否 | 作业错误码,视频截图失败时,会有该字字段。 |
| ErrorMessage | String | 否 | 作业错误信息,视频截图失败时,会有该字段。 |
| CoverUrl | String | 否 | 封面图片地址。截图失败不会有该字段。 |
| SnapshotInfos | SnapshotInfo[] | 否 | 截图列表。截图失败不会有该字段,具体结构请参见下
文SnapshotInfos 截图数据 。 |

| 参数名称 | 类型 | 必备项 | 描述 |
|--------|--------|-----|--|
| Extend | String | 否 | 用户自定义回调透传数据。更多信息,请参见 <mark>媒体上传自定</mark>
<mark>义设置</mark> UserData。 |

Snapshot Infos 截图数据

| 参数名称 | 类型 | 必填项 | 描述 |
|----------------------|--------|-----|---|
| Status | String | 是 | 视频截图任务状态。
• success:成功。
• fail:失败。 |
| SnapshotType | String | 是 | 截图类型。
• CoverSnapshot: 封面截图。
• NormalSnapshot: 普通截图。
• SpriteSnapshot: 雪碧截图。
更多信息,请参见视频截图。 |
| SnapshotCount | Long | 是 | 截图总数。 |
| SnapshotForma
t | String | 是 | 截图名称格式。可使用OSS存储地址或CDN域名和该字段信
息生成截图地址。
⑦ 说明
如果用户存在域名且域名可能变动频繁,可根据字段信
息生成截图具体地址。 |
| Snapshot Regul
ar | String | 是 | 截图地址规则。可根据规则生成截图地址 (推荐使用该字
段生成截图地址), 生成说明请参见下文截图地址生成。
⑦ 说明
截图地址规则, 有域名则返回CDN地址规则, 否则返回
OSS地址规则, 暂时不支持HTTPS地址规则返回。 |
| Jobid | String | 是 | 截图任务ID。 |

⑦ 说明

最新上传的视频,截图OSS存储地址与视频OSS存储地址一致。更多信息,请参见存储管理。

截图地址生成

提供两种生成地址的方法:

- 根据SnapshotFormat生成截图地址
 - 地址规则: http(s)://{CDN域名或OSS存储地址}/SnapshotFormat。
 - 生成说明: {SnapshotCount}为对应每张截图的序列号(固定5位宽度,不足以0补齐)。
 - 。 示例:
 - 第一张截图序列号为: 00001, 则地址为:

```
http://example.com/2327a6ec24b44844b3a5e2c1b691****/covers/990f3820db2948b5b4a13d65d9a4****-00001.jpg
```

■ 第二张截图序列号为: 00002, 则地址为:

```
http://example.com/2327a6ec24b44844b3a5e2c1b691****/covers/990f3820db2948b5b4a13d65d9a4***-00002.jpg
```

以此类推。

- 根据SnapshotRegular生成截图地址
 - 地址规则: Snapshot Regular已经为完整的地址规则。
 - 生成说明: 同SnapshotFormat截图具体地址的生成说明。

回调示例

回调示例说明:

- 对于HTTP回调,以下内容为HTTP Post Body。
- 对于MNS回调,以下内容为消息体。

```
{
  "EventType": "SnapshotComplete",
  "EventTime": "2018-07-31T10:07:31Z",
  //Cover URL示例: http://example.com/covers/990f3820db2948b5b4a13d65d9a4****-00002.jpg
  "CoverUrl": "<your CoverURL>",
  "Extend":"test data",
  "SnapshotInfos": [
      "Status": "success",
      "SnapshotType": "CoverSnapshot",
      "SnapshotCount": 2,
      "SnapshotFormat": "2327a6ec24b44844b3a5e2c1b691****/covers/990f3820db2948b5b4a13d65d9a4***
*-{SnapshotCount}.jpg",
     //SnapshotRegular示例: http://example.com/covers/990f3820db2948b5b4a13d65d9a4***-{Snapshot
Count }.jpg
      "SnapshotRegular": "<your SnapshotRegular>",
      "JobId": "ee16d4bbf3f7****e094bcb8cf8ddde"
    },
    {
      "Status": "success",
      "SnapshotType": "SpriteSnapshot",
      "SnapshotCount": 1,
      "SnapshotFormat": "2327a6ec24b44844b3a5e2c1b691****/covers/sprite/990f3820db2948b5b4a13d65
d9a4****-{SnapshotCount}.jpg",
     //SnapshotRegular示例: http://example.com/covers/990f3820db2948b5b4a13d65d9a4***-{Snapshot
Count }.jpg
      "SnapshotRegular": "<your SnapshotRegular>",
    }
 ]
}
```

12.4.6. 视频动图完成

本文为您介绍视频动图完成事件、事件通知的内容和回调示例。

事件类型

DynamicImageComplete

事件说明

视频动图完成后会产生DynamicImageComplete事件。

? 说明

- 若配置了CDN加速域名,并开启了URL鉴权,则需要自己生成auth_key才能访问动图地址,否则会返回
 HTTP 403。关于URL鉴权的信息,请参见URL鉴权。
- 若未配置CDN加速域名,将返回OSS源站地址,该地址仅当bucket设置为公共读时才可访问,为了保 障数据安全,建议您先配置CDN加速域名。具体操作,请参见域名管理。

| 参数名称 | 类型 | 必备项 | 描述 |
|-------------------------|---------------------------|-----|---|
| EventTime | String | 是 | 事件产生时间,为UTC时间: yyyy-MM-ddTHH:mm:ssZ。 |
| EventType | String | 是 | 事件类型,系统规定参数。固定
为DynamicImageComplete。 |
| Videold | String | 是 | 视频ID。 |
| Status | String | 是 | 动图任务状态。
• success:成功。
• fail:失败。 |
| Jobid | String | 是 | 动图作业ID。 |
| ErrorCode | String | 否 | 失败错误码,动图任务出错时,会有该字段。 |
| ErrorMessage | String | 否 | 失败错误信息,动图任务出错时,会有该字段。 |
| DynamiclmageJ
oblnfo | DynamicImageJ
obInfo[] | 否 | 动图详细信息,为JSON字符串,动图失败时不返回该字段。
详细内容请参见下表 DynamicImageJobInfo 动图数据 。 |

DynamicImageJobInfo 动图数据

| 参数名称 | 类型 | 必备项 | 描述 |
|----------|--------|-----|--|
| Jobid | String | 是 | 动图任务ID。 |
| Status | String | 是 | 动图任务状态。
• success:成功。
• fail:失败。 |
| FileURL | String | 是 | 动图文件地址。 |
| FileSize | String | 否 | 动图文件大小,单位:Byte。 |
| Fps | String | 否 | 视频流帧率,每秒多少帧。 |

| 参数名称 | 类型 | 必备项 | 描述 |
|--------|--------|-----|------------|
| Height | String | 否 | 动图高,单位:px。 |
| Width | String | 否 | 动图宽,单位:px。 |

回调示例

回调示例说明:

● 对于HTTP回调,以下内容为HTTP Post Body。

```
• 对于MNS回调,以下内容为消息体。
```

```
{
    "Status":"success",
    "VideoId":"6a45f222e08fhdjdn2d144503213****",
    "EventType":"DynamicImageComplete",
    "EventTime":"2019-11-05T09:12:58Z",
    "DynamicImageJobInfo":"{
         \"FileSize\":\"1834831\",
         //FileURL示例: http://example.aliyundoc.test/6a45f222e08fhdjdn2d144503213****/image/dyna
mic/3afle9705cb24321d4c3719f0d41****.gif
         \"FileURL\":\"<your File URL>",
         \"Fps\":\"25\",
         \"Height\":\"200\",
         \"JobId\":\"3af7e9705cb24js83j231d4c3719****\",
         \"Status\":\"success\",
         \"Width\":\"200\"
    }",
    "JobId":"3af7e9705cb24js83j231d4c3719****"
}
```

12.4.7. 直转点视频录制完成

本文为您介绍直转点视频录制完成事件、事件通知的内容和回调示例。

事件类型

AddLiveRecordVideoComplete

事件说明

开启直播录制到点播后,当完成一个录制周期或直播流结束时会生成一个点播视频,此时产生AddLiveRecordVideoComplete事件。

⑦ 说明 直播推流断开3分钟以上视为本次直播结束。

| 参数名称 | 类型 | 必备项 | 描述 |
|---------------------|--------|-----|---|
| EventTime | String | 是 | 事件产生时间,为UTC时间:yyyy-MM-ddTHH:mm:ssZ。 |
| EventType | String | 是 | 事件类型,系统规定参数。固定
为:AddLiveRecordVideoComplete。 |
| Videold | String | 是 | 视频ID。 |
| Status | String | 是 | 录制完成状态。
• success:成功。
• fail:失败。 |
| StreamName | String | 否 | 直播流名称。 |
| DomainName | String | 否 | 域名。 |
| AppName | String | 否 | App名称。 |
| RecordStartTim
e | String | 否 | 录制开始时间,为UTC时间:yyyy-MM-ddTHH:mm:ssZ。 |
| RecordEndTime | String | 否 | 录制结束时间,为UTC时间: yyyy-MM-ddTHH:mm:ssZ。 |

回调示例

回调示例说明:

- 对于HTTP回调,以下内容为HTTP Post Body。
- 对于MNS回调,以下内容为消息体。

```
{
   "EventTime": "2017-12-08T09:26:17Z",
   "EventType": "AddLiveRecordVideoComplete",
   "VideoId": "43q9fjdhef****",
   "Status": "success",
   "StreamName": "xxx",
   "DomainName": "xxx",
   "AppName": "xxx",
   "RecordStartTime":"2017-12-08T07:40:56Z",
   "RecordEndTime":"2017-12-08T09:26:17Z",
}
```

12.4.8. 直转点录制视频合成开始

本文为您介绍直转点录制视频合成开始事件、事件通知的内容和回调示例。

事件类型

LiveRecordVideoComposeStart

事件说明

直播录制视频转成点播视频的过程中,如果开启了录制视频自动合成功能,每次直播结束(推流结束或超时) 后,会自动开始点播视频合成。这时就会产生LiveRecordVideoComposeStart事件,事件内容包括将合成后的视频ID。

? 说明

- 直播默认推流断开3分钟以上,视为本次直播结束。
- 本事件仅为直转点录制视频合成开始,不包含后续合成状态,用户应当记录该ID,并跟进该ID后续状态。

事件内容

| 参数名称 | 类型 | 必备项 | 描述 |
|------------|--------|-----|--|
| EventTime | String | 是 | 事件产生时间,为UTC时间:yyyy-MM-ddTHH:mm:ssZ。 |
| EventType | String | 是 | 事件类型,系统规定参数。固定
为:LiveRecordVideoComposeStart。 |
| Videold | String | 是 | 视频ID。 |
| Status | String | 是 | 录制完成状态。
• success:成功。
• fail:失败。 |
| StreamName | String | 否 | 直播流名称。 |
| DomainName | String | 否 | 域名。 |
| AppName | String | 否 | App名称。 |

回调示例

回调示例说明:

- 对于HTTP回调,以下内容为HTTP Post Body。
- 对于MNS回调,以下内容为消息体。

```
{
   "EventTime": "2017-12-08T09:26:17Z",
   "EventType": "LiveRecordVideoComposeStart",
   "VideoId": "43q9fsjnehdf****",
   "Status": "success",
   "StreamName": "streamname_****",
   "DomainName": "domainname_****",
   "AppName": "appname_****"
}
```

12.4.9. URL上传视频完成

本文为您介绍URL上传视频完成事件、事件通知的内容和回调示例。

事件类型

UploadByURLComplete

事件说明

提交URL拉取上传任务后,云端拉取视频上传完成会产生UploadByURLComplete事件。

事件内容

| 参数名称 | 类型 | 必备项 | 描述 |
|--------------|--------|-----|--|
| EventTime | String | 是 | 事件产生时间,为UTC时间:yyyy-MM-ddTHH:mm:ssZ。 |
| EventType | String | 是 | 事件类型,系统规定参数。固定
为:UploadByURLComplete。 |
| Videold | String | 是 | 视频ID。当视频拉取成功后会有该字段。 |
| Jobid | String | 是 | 上传任务ID。 |
| SourceURL | String | 否 | 源文件URL地址。 |
| Status | String | 否 | 上传结果。
● success:成功。
● fail:失败。 |
| ErrorCode | String | 否 | 作业错误码,上传出错时,会有该字段。 |
| ErrorMessage | String | 否 | 作业错误信息,上传出错时,会有该字段。 |

回调示例

回调示例说明:

- 对于HTTP回调,以下内容为HTTP Post Body。
- 对于MNS回调,以下内容为消息体。
- 上传成功

```
{
   "Status": "success",
   "EventTime": "2017-03-20T07:49:17Z",
   "EventType": "UploadByURLComplete",
   "VideoId": "43q9fjdun3f****",
   "JobId": "4c815bjs83j1****",
   //SourceURL示例: http://example.oss-cn-shanghai.aliyuncs.com/27ffc438-164d55217ef-0005-6884-
51a-1****.mp4
   "SourceURL ": "<your Source URL>",
   "Size":"123456"
}
```

• 上传失败

```
{
   "Status": "fail",
   "EventTime": "2017-03-20T07:49:17Z",
   "EventType": "UploadByURLComplete",
   "ErrorCode ": "URLInvalidError ",
   "ErrorMessage ": "download video failed by the url, please check it",
   "JobId": "4c815bjsued19c8e",
   //SourceURL示例: http://example.oss-cn-shanghai.aliyuncs.com/27ffc438-164d55217ef-0005-6884-5
la-1****.mp4
   "SourceURL ": "<your Source URL>",
}
```

12.4.10. 人工审核完成

本文为您介绍人工审核完成事件、事件通知的内容和回调示例。

事件类型

CreateAudit Complete

事件说明

调用人工审核接口时, 会产生CreateAuditComplete事件。

事件内容

| 参数名称 | 类型 | 必备项 | 描述 |
|--------------|--------|-----|--|
| EventType | String | 是 | 事件类型,系统规定参数。固定
为:CreateAuditComplete。 |
| EventTime | String | 是 | 事件产生时间,为UTC时间:yyyy-MM-ddTHH:mm:ssZ。 |
| Status | String | 是 | 消息通知状态。
● success:成功。
● fail:失败。 |
| Mediald | String | 是 | 媒体ID。 |
| SubType | String | 是 | 审核的媒体类型。
video:音视频。 |
| AuditStatus | String | 是 | 审核结果。
• Normal: 通过。
• Blocked: 屏蔽。 |
| CreationTime | String | 是 | 审核时间,为UTC时间: yyyy-MM-ddTHH:mm:ssZ。 |
| Auditor | String | 否 | 审核人员。 |
| Reason | String | 否 | 审核理由。 |

回调示例

回调示例说明:

- 对于HTTP回调,以下内容为HTTP Post Body。
- 对于MNS回调,以下内容为消息体。

```
{
    "EventType": "CreateAuditComplete",
    "EventTime": "2017-03-20T07:49:17Z",
    "MediaId": "<Example ID>",
    "Status": "success",
    "CreationTime": "2017-03-20T07:49:17Z",
    "SubType": "video",
    "AuditStatus": "Blocked",
    "Auditor":"author",
    "Reason":"含有网络低俗内容"
}
```

12.4.11. 智能审核完成

本文为您介绍智能审核完成事件、事件通知的内容和回调示例。

事件类型

AIMediaAudit Complete

事件说明

当智能审核完成时, 会产生AIMediaAuditComplete事件。

⑦ 说明 审核结果的图片资源仅在点播提供的免费存储上保留2周时间,之后会自动删除。

| 名称 | 类型 | 必备项 | 描述 |
|-----------|--------|-----|---|
| EventTime | String | 是 | 事件产生时间,为UTC时间: yyyy-MM-ddTHH:mm:ssZ。 |
| EventType | String | 是 | 事件类型,系统规定参数。固定
为:AIMediaAuditComplete。 |
| Jobid | String | 是 | 作业ID,与提交智能审核作业接口返回的JobId一致。 |
| Mediald | String | 是 | 视频ID。 |
| Status | String | 是 | 作业状态。
● success:成功。
● fail:失败。 |
| Code | String | 否 | 作业错误码,当审核出错时,会有该字段。 |
| Message | String | 否 | 作业错误信息,当审核出错时,会有该字段。 |
| Data | String | 是 | 作业结果数据,JSON对象。具体结构,请参见具体结构 <mark>智能</mark>
<mark>审核结果AlMediaAuditResult</mark> 。 |

回调示例

回调示例说明:

- 对于HTTP回调,以下内容为HTTP Post Body。
- 对于MNS回调,以下内容为消息体。

```
{
    "EventTime": "2017-03-20T07:49:17Z",
    "EventType": "AIMediaAuditComplete",
    "JobId": "43q91jdh7df****",
    "MediaId": "SHEN38505NDF9****",
    "Status": "success",
    "Code": "0",
    "Message": "OK",
    "Data": {
           "AbnormalModules":"video",
            "Label":"porn",
            "Suggestion":"review",
            "VideoResult":{
                "Suggestion":"review",
                "TerrorismResult":{
                    "TopList":[
                        {
                            "Score":"100.000000000",
                            "Label":"normal",
                            "Timestamp":"3005",
                            "Url":"http://example-bucket-****.oss-cn-shanghai.aliyuncs.com/aivid
eocensor/1****.jpg"
                        },
                        {
                            "Score":"100.000000000",
                            "Label":"normal",
                            "Timestamp":"15005",
                            "Url":"http://example-bucket-****.oss-cn-shanghai.aliyuncs.com/aivid
eocensor/1****.jpg"
                        }
                    ],
                    "Suggestion":"pass",
                    "MaxScore":"100.000000000",
                    "AverageScore":"100.000000000",
                    "Label":"normal",
                    "CounterList":[
                        {
                            "Label":"terrorism",
                            "Count":0
                        },
                        {
                            "Label":"outfit",
                            "Count":0
                        },
                        {
                            "Label":"logo",
                            "Count":0
                        },
                        {
                            "Label":"weapon",
                                 . . .
```

```
"Count":0
                         },
                         {
                             "Label":"politics",
                             "Count":0
                         },
                         {
                             "Label":"others",
                             "Count":0
                         },
                         {
                             "Label":"normal",
                             "Count":16
                         }
                    ]
                },
                "Label":"porn",
                "PornResult":{
                    "TopList":[
                        {
                             "Score":"92.480000000",
                             "Label":"sexy",
                             "Timestamp":"1005",
                             "Url":"http://example.oss-cn-shanghai.aliyuncs.com/aivideocensor/***
*.jpg"
                         },
                         {
                             "Score":"91.820000000",
                             "Label":"sexy",
                             "Timestamp":"9005",
                             "Url": "http://example.oss-cn-shanghai.aliyuncs.com/aivideocensor/***
*.jpg"
                         }
                    ],
                    "Suggestion": "review",
                    "MaxScore":"92.480000000",
                    "AverageScore":"81.7066666667",
                    "Label":"sexy",
                    "CounterList":[
                         {
                             "Label":"porn",
                             "Count":0
                        },
                         {
                            "Label":"sexy",
                             "Count":6
                         },
                         {
                             "Label":"normal",
                             "Count":10
                         }
                    ]
                }
            },
            ...
                 CoverResult":[
                {
                    "Suggestion":"pass",
```

```
"Type":"cover",
                    "Label": "normal",
                    //URL示例: http://www.aliyundoc.com/43q91jdh7df****.jpg
                    "Url":"<your image URL>",
                    "Result":[
                        {
                             "Suggestion":"pass",
                             "Score":"65.25",
                             "Label": "normal",
                             "Scene":"porn"
                        },
                         {
                             "Suggestion":"pass",
                             "Score":"100.0",
                             "Label":"normal",
                             "Scene":"terrorism"
                         }
                    ]
                }
            ],
            "TitleResult":[
                {
                    "Suggestion":"pass",
                    "Type":"title",
                    "Score":"99.91",
                    "Content":"1111",
                    "Label":"normal",
                    "Scene":"antispam"
                }
            ]
       }
}
```

12.4.12. 音视频分析完成

本文为您介绍音视频分析完成事件、事件通知的内容和回调示例。

事件类型

VideoAnalysisComplete

事件说明

视频点播会对上传完成的音视频源文件进行分析,完成后会产生此事件。

? 说明

- 所有上传的音视频文件,默认都会进行源文件分析,且整个过程完全免费。
- 分析过程会提取源文件的时长、分辨率(宽高)、码率、帧率等基本的Meta信息。
- 分析完成后的Meta信息会录入媒资库,可通过获取源文件信息接口,获取更多的源文件Meta信息。

视频点播

| 参数名称 | 类型 | 必备项 | 描述 |
|--------------|--------|-----|--|
| EventTime | String | 是 | 事件产生时间,为UTC时间: yyyy-MM-ddTHH:mm:ssZ。 |
| EventType | String | 是 | 事件类型,系统规定参数。固定
为:VideoAnalysisComplete。 |
| Videold | String | 是 | 音视频ID。 |
| Status | String | 是 | 分析源文件结果状态。
• success:成功。
• fail:失败。 |
| Width | Long | 否 | 源文件宽 , 源片为纯音频不会该字段。 |
| Height | Long | 否 | 源文件高 , 源片为纯音频不会有该字段。 |
| Duration | Float | 否 | 源文件时长 , 单位: 秒。 |
| Bitrate | String | 否 | 源文件码率 / 单位:Kbps。 |
| Fps | String | 否 | 源文件帧率 , 每秒多少帧 , 源片为纯音频不会有该字段。 |
| Size | Long | 否 | 源文件大小,单位:Byte(字节)。 |
| ErrorCode | String | 否 | 作业错误码,分析源文件信息出错时,会有该字段。 |
| ErrorMessage | String | 否 | 作业错误信息,分析源文件信息出错时,会有该字段。 |

⑦ 说明 Status=fail, 表示分析不成功,则不会有源文件的基本信息,通常失败的原因是源文件封装信息 有异常导致。

回调示例

回调示例说明:

- 对于HTTP回调,以下内容为HTTP Post Body。
- 对于MNS回调,以下内容为消息体。

分析成功回调消息,示例如下:

```
{
    "VideoId":"84bd5b0566ddj39549986befd0e80****",
    "Duration":"12",
    "Height":"360",
    "Width":"630",
    "Fps":"30",
    "Bitrate":"499",
    "Size":"1234568",
    "EventTime":"2018-11-28T10:12:48Z",
    "EventType":"VideoAnalysisComplete",
    "Status":"success"
}
```

分析失败回调消息,示例如下:

```
{
    "VideoId":"84bd5b0566ddj39549986befd0e80****",
    "EventTime":"2018-11-28T10:12:48Z",
    "EventType":"VideoAnalysisComplete",
    "Status":"fail",
    "ErrorCode":"InvalidParameter.ResourceContentBad",
    "ErrorMessage":"The resource operated InputFile is bad"
}
```

12.4.13. 视频DNA完成

本文为您介绍视频DNA完成事件、事件通知的内容和回调示例。

事件类型

AIMediaDNAComplete

事件说明

当视频DNA作业完成时, 会产生AIMediaDNAComplete事件。

事件内容

| 参数名称 | 类型 | 必备项 | 描述 |
|-----------|--------|-----|---|
| EventTime | String | 是 | 事件产生时间,为UTC时间:yyyy-MM-ddTHH:mm:ssZ。 |
| EventType | String | 是 | 事件类型,系统规定参数。固定
为:AIMediaDNAComplete。 |
| Jobid | String | 是 | 作业ID,与提交DNA作业接口返回的JobId一致。 |
| Mediald | String | 是 | 视频ID。 |
| Status | String | 是 | 作业状态。
• success:成功。
• fail:失败。 |
| Code | String | 否 | 作业错误码,当作业失败时,会有该字段。 |
| Message | String | 否 | 作业错误信息,当作业失败时,会有该字段。 |
| Data | String | 是 | DNA处理结果数据,JSON对象。具体结构,请参见 <mark>视频DNA</mark>
结果。 |

回调示例

回调示例说明:

- 对于HTTP回调,以下内容为HTTP Post Body。
- 对于MNS回调,以下内容为消息体。

```
{
    "EventTime": "2017-03-20T07:49:17Z",
    "EventType": "AIMediaDNAComplete",
    "JobId": "88c6ca184c0edj384a5b665e2a12****",
    "MediaId": "3D12340d92cjsw83k1fab46a0b847fd****",
    "Status": "success",
    "Code": "0",
    "Message": "0K",
    "Data": "{\"VideoDNA\":[{\"Similarity\":\"0.99\",\"PrimaryKey\":\"7a1d275cc8da4jw93mkde0e182
e80****\",\"Detail\":[{\"Input\":{\"Start\":\"0.0\",\"Duration\":\"14.0\"},\"Duplication\":{\"St
art\":\"0.5\",\"Duration\":\"14.4\"}]}]"
}
```

12.4.14. 智能标签作业完成

本文为您介绍智能标签作业完成事件、事件通知的内容和回调示例。

事件类型

AIVideoTagComplete

事件说明

当智能标签作业完成时, 会产生AlVideoTagComplete事件。

事件内容

| 参数名称 | 类型 | 必备项 | 描述 |
|-----------|--------|-----|---|
| EventTime | String | 是 | 事件产生时间,为UTC时间:yyyy-MM-ddTHH:mm:ssZ。 |
| EventType | String | 是 | 事件类型,系统规定参数。固定
为:AlVideoTagComplete。 |
| Jobid | String | 是 | 作业ID,与提交智能标签作业接口返回的JobId一致。 |
| Mediald | String | 是 | 视频ID。 |
| Status | String | 是 | 作业状态。
● success:成功。
● fail:失败。 |
| Code | String | 否 | 作业错误码。当作业出错时,会有该字段。 |
| Message | String | 否 | 作业错误信息。当作业出错时,会有该字段。 |
| Data | String | 是 | 智能标签作业结果数据,JSON对象。具体结构,请参见 <mark>智能</mark>
标签。 |

回调示例

回调示例说明:

- 对于HTTP回调,以下内容为HTTP Post Body。
- 对于MNS回调,以下内容为消息体。

| { |
|--|
| "EventTime": "2017-03-20T07:49:17Z", |
| "EventType": "AIVideoTagComplete", |
| "JobId": "jd83k4184c0e47098a5b665e2aj3****", |
| "MediaId": "sj83k40d92c641401fab46a0b842e9****", |
| "Status": "success", |
| "Code": "0", |
| "Message": "OK", |
| "Data": "{\"Keyword\":[{\"Times\":[\"3260\",\"4000\",\"5000\",\"8410\",\"10000\",\"12000\",\ |
| "13000\",\"14000\",\"15000\"],\"Tag\":\" 气垫 \"},{\"Times\":[\"2000\",\"4000\",\"5000\",\"7000\",\ |
| "8000\",\"10000\",\"12000\",\"13000\",\"14000\"],\"Tag\":\"lancome\"},{\"Times\":[\"3260\",\"400 |
| 0\",\"5000\",\"8410\",\"12000\",\"13000\",\"14000\"],\"Tag\":\" 兰蔻 \"},{\"Times\":[\"4000\",\"500 |
| 0\",\"12000\",\"13000\",\"14000\",\"15000\"],\"Tag\":\"CC 霜 \"},{\"Times\":[\"8000\",\"8410\",\"1 |
| 0000\"],\"Tag\":\" 轻盈 \"},{\"Times\":[\"3260\",\"8410\"],\"Tag\":\"cc 霜 \"},{\"Times\":[\"3260\"], |
| \"Tag\":\"花瓣\"},{\"Times\":[\"8410\"],\"Tag\":\"空气\"},{\"Times\":[\"900\"],\"Tag\":\"亮白\"},{ |
| \"Times\":[\"900\"],\"Tag\":\" 清透 \"},{\"Times\":[\"900\"],\"Tag\":\" 肌肤 \"},{\"Times\":[\"8410\" |
|],\"Tag\":\" 法国兰蔻 \"}],\"Person\":[{\"Times\":[\"10760\"],\"Tag\":\" 张三 \",\"FaceUrl\":\"http:// |
| example.com/aivideotag/11111B65-1955-4B84-823D-7921A742****/Index_00016.jpg\"}],\"Location\":[{\ |
| "Times\":[\"8410\"],\"Tag\":\" 亚洲 \"}]}" |
| } |

12.4.15. 辅助媒资上传完成

本文为您介绍辅助媒资上传完成事件、事件通知的内容和回调示例。

事件类型

AttachedMediaUploadComplete

事件说明

辅助媒资文件上传成功后, 会产生AttachedMediaUploadComplete事件。

| 参数名称 | 类型 | 必备项 | 描述 |
|---------------------|--------|-----|--|
| EventTime | String | 是 | 事件产生时间,为UTC时间: yyyy-MM-ddTHH:mm:ssZ。 |
| EventType | String | 是 | 事件类型,系统规定参数。固定
为:AttachedMediaUploadComplete。 |
| AttachedMedial
d | String | 是 | 辅助媒资ID。 |
| Size | Long | 是 | 上传的文件大小,单位:Byte(字节)。 |
| FileURL | String | 是 | 上传辅助媒资文件的URL地址。 |
| Status | String | 是 | 辅助媒资上传是否成功。
• success:成功。
• fail:失败。 |

⑦ 说明 由于要支持断点续传,点播服务端无法确定您是暂时中断上传还是上传失败,故没有上传失败的事件通知。

回调示例

回调示例说明:

- 对于HTTP回调,以下内容为HTTP Post Body。
- 对于MNS回调,以下内容为消息体。

```
{
    "EventTime": "2019-01-20T07:49:17Z",
    "EventType": "AttachedMediaUploadComplete",
    "AttachedMediaId": "62cb3151eba52js82j2da3b55bc5****",
    "Size": 14333,
    //FileURL示例: https://example.aliyundoc.com/ABEBDE1JSU79FD4D****/ba52js82j2da3b55bc5****.png
    "FileURL": "<your File URL>",
    "Status":"success"
}
```

12.4.16. 媒体合成完成

本文为您介绍媒体合成完成事件、事件通知的内容和回调示例。

事件类型

ProduceMediaComplete

事件说明

当媒体合成完成时,会产生此事件。

? 说明

- 媒体合成可以在点播控制台的视频剪辑页面发起,也可以通过调用视频合成接口发起。
- 媒体合成的目标文件为媒体资源的源文件。

| 参数名称 | 类型 | 必备项 | 描述 |
|-----------|--------|-----|--|
| EventTime | String | 是 | 事件产生时间,为UTC时间:yyyy-MM-ddTHH:mm:ssZ。 |
| EventType | String | 是 | 事件类型,系统规定参数。固定
为: ProduceMediaComplete。 |

| 参数名称 | 类型 | 必备项 | 描述 |
|--------------|-----------------|--|--|
| | Status String 是 | | 媒体合成状态。
● success:成功。
● fail:失败。 |
| Status | | ⑦ 说明 Status=fail,表示媒体合成不成功,则合成目标资源对应的源文件未生成。 | |
| Mediald | String | 是 | 合成资源ID。 |
| ProjectId | String | 是 | 本次合成所对应的云剪辑工程ID。 |
| ErrorCode | String | 否 | 作业错误码,媒体合成出错时,会有该字段。 |
| ErrorMessage | String | 否 | 作业错误信息,媒体合成出错时,会有该字段。 |

回调示例

回调示例说明:

- 对于HTTP回调,以下内容为HTTP Post Body。
- 对于MNS回调,以下内容为消息体。

合成成功回调消息,示例如下:

```
{
    "MediaId": "1234343b689*****a5422dfe1e472a41",
    "ProjectId": "987624bab6*****d9fcdeefdd974c7aa",
    "Status": "success",
    "EventType": "ProduceMediaComplete",
    "EventTime": "2018-12-26T14:34:09Z"
}
```

合成失败回调消息,示例如下:

```
{
   "MediaId": "1234343b689*****a5422dfe1e472a41",
   "ProjectId": "987624bab6*****d9fcdeefdd974c7aa",
   "Status": "fail",
   "EventType": "ProduceMediaComplete",
   "EventTime": "2018-12-26T14:34:09Z",
   "ErrorCode": "InvalidParameter.ResourceContentBad",
   "ErrorMessage": "The resource operated InputFile is bad"
}
```

12.4.17. 媒体删除完成

本文为您介绍媒体删除完成事件、事件通知的内容和回调示例。

事件类型

DeleteMediaComplete

事件说明

目前仅调用删除完整视频接口时, 会产生DeleteMediaComplete事件。

事件内容

| 参数名称 | 类型 | 必备项 | 描述 |
|--------------|--------|-----|---|
| EventType | String | 是 | 事件类型,系统规定参数。固定
为:DeleteMediaComplete。 |
| EventTime | String | 是 | 事件产生时间,为UTC时间: yyyy-MM-ddTHH:mm:ssZ。 |
| Status | String | 是 | 媒体删除状态。
● success:成功。
● fail:失败。 |
| MediaType | String | 是 | 媒体类型。
• video: 音视频。
• image: 图片。
• attached: 辅助媒资。 |
| DeleteType | String | 是 | 删除资源类型。 all:完整媒体信息。 mezzanine:源文件信息。 stream:转码输出流信息。 |
| Mediald | String | 是 | 媒体ID。 |
| Jobids | String | 否 | 仅当DeleteType为stream时,才会返回。 |
| ErrorCode | String | 否 | 作业错误码,删除操作出错时,会有该字段 |
| ErrorMessage | String | 否 | 作业错误信息,删除操作出错时,会有该字段。 |

回调示例

回调示例说明:

- 对于HTTP回调,以下内容为HTTP Post Body。
- 对于MNS回调,以下内容为消息体。

```
媒体删除成功回调消息,示例如下:
```

```
{
    "EventType": "DeleteMediaComplete",
    "EventTime": "2017-03-20T07:49:17Z",
    "Status": "success",
    "MediaType": "video",
    "DeleteType": "all",
    "MediaId": "1234343b689jsi3ka5422dfele47****"
}
```

媒体删除失败回调消息,示例如下:

```
{
    "EventType": "DeleteMediaComplete",
    "EventTime": "2017-03-20T07:49:17Z",
    "Status": "fail",
    "MediaType": "video",
    "DeleteType": "all",
    "MediaId": "1234343b689jsi3ka5422dfele47****",
    "ErrorCode": "InvalidVideo.NotFound",
    "ErrorMessage": "The video does not exist."
}
```

12.4.18. 媒资基础信息变更

本文为您介绍媒资基础信息变更事件、事件通知的内容和回调示例。

事件类型

MediaBaseChangeComplete

事件说明

当调用获取视频上传地址和凭证、修改视频信息、批量修改视频信息、删除完整视频接口时, 会产生 MediaBaseChangeComplete事件。

| 参数名称 | 类型 | 必备项 | 描述 |
|------------|--------|-----|---|
| EventType | String | 是 | 事件类型,固定为MediaBaseChangeComplete。 |
| EventTime | String | 是 | 事件产生时间,为UTC时间:yyyy-MM-ddTHH:mm:ssZ。 |
| Status | String | 是 | 消息通知状态。
• success:成功。
• fail:失败。 |
| MediaT ype | String | 是 | 媒体类型。
• video: 音视频。
• image: 图片。
• attached: 辅助媒资。 |
| Mediald | String | 是 | 媒体ID。 |

| 参数名称 | 类型 | 必备项 | 描述 |
|--------------|--------|-----|--|
| OperateMode | String | 是 | 操作方式。
• create:新增。
• update:更新。
• delete:删除。 |
| MediaContent | String | 是 | 媒资修改内容,JSON字符串。目前支持音视频基础信息,更
多信息,请参见下文 音视频基础信息 。 |

音视频基础信息

| 参数名称 | 类型 | 必备项 | 描述 |
|-------------|--------|-----|--|
| Title | String | 否 | 标题, JSON字符串, 取值示例:
{"OldValue":"OldTitle",
"NewValue":"NewTitle"} |
| Description | String | 否 | 描述, JSON字符串, 取值示例:
{"OldValue":"OldDesc",
"NewValue":"NewDesc"} |
| CoverURL | String | 否 | 封面图片, JSON这符串, 取值示例:
{"OldValue":"OldCoverURL", "NewValue":
"NewCoverURL"} |
| Cateld | String | 否 | 分类ID, JSON字符串, 取值示例:
{"OldValue":123, "NewValue":456} |
| Tags | String | 否 | 标签, JSON字符串, 取值示例:
{"OldValue":"OldTag", "NewValue" :"NewTag"} |

? 说明

- OldValue: 字段修改前的值。
- NewValue: 字段修改后的值。
- 字段新增时, OldValue为空; 媒资删除时, NewValue为空。

回调示例

开发指南·事件通知

回调示例说明:

- 对于HTTP回调,以下内容为HTTP Post Body。
- 对于MNS回调,以下内容为消息体。

```
{
    "EventType":"MediaBaseChangeComplete",
    "EventTime":"2019-06-20T02:18:58Z",
    "Status":"success",
    "MediaId":"3b46b391419aj294m83b459f7435****",
    "MediaIType":"video",
    "OperateMode":"update",
    "MediaContent":
    {
        "Description":"{\"OldValue\":\"OldDesc\", \"NewValue\":\"NewDesc\"}",
        "CoverURL":"{\"NewValue\":\"http://example.aliyundoc.com/image/cover/8C46D968F6954348AFC7A8
8****-6-2.png\"}",
        "CateId":"{\"NewValue\":100002****}"
    }
}
```

12.4.19. 音视频刷新或预热完成

本文为您介绍音视频刷新或预热完成事件、事件通知的内容和回调示例。

事件类型

Submit MediaRef reshComplet e

事件说明

<mark>提交音视频刷新或预热任务</mark>后,每个音/视频会单独提交播放URL刷新或预热请求,会得到多个taskld,当每个音/视频 所有播放URL提交完刷新或预热请求后,会产生Submit MediaRefreshComplete事件。

② 说明 暂不支持通过视频点播控制台配置该回调事件,需要调用Set MessageCallback接口配置该回调事件。

| 参数名称 | 类型 | 必备项 | 描述 |
|-----------|--------|-----|---|
| EventTime | String | 是 | 事件产生时间,为UTC时
间:yyyy-MM-
ddTHH:mm:ssZ。 |
| EventType | String | 是 | 事件类型,系统规定参数。
固定
为:SubmitMediaRefre
shComplete。 |
| Status | String | 是 | 提交刷新预热是否成功。取
值:
• success:成功
• fail:失败 |

视频点播

| 参数名称 | 类型 | 必备项 | 描述 |
|-------------------|--------|-----|---|
| MediaRefreshJobId | String | 是 | 刷新任务ID。 |
| Mediald | String | 是 | 媒体ID, 即音/视频ID。 |
| TaskType | String | 是 | 任务类型。取值:
• Refresh: 刷新
• Preload: 预热 |
| SuccessPlayUrls | String | 是 | 成功提交刷新或预热的播放
URL。多个URL之间用半角
逗号(,)分隔。 |
| Taskids | String | 是 | 播放URL的刷新或预热的任
务ID,一个URL对应一个任
务ID。该任务ID用于查询刷
新和预热状态接口查询每个
播放URL的刷新或预热状
态。 |
| FilterPolicy | String | 是 | 播放流的筛选策略。JSON格
式,内容为 <mark>提交音视频刷新</mark>
或预热任务接口的请求参
数。 |
| Extend | String | 否 | 用户自定义透传参数。 |
| ErrorCode | String | 否 | 错误码。当提交刷新或预热
任务失败时会有该字段。 |
| ErrorMessage | String | 否 | 错误信息。当提交刷新或预
热任务失败时会有该字段。 |

回调示例

回调示例说明:

- 对于HTTP回调,以下内容为HTTP Post Body。
- 对于MNS回调,以下内容为消息体。
- 任务成功

```
{
    "SuccessPlayUrls":"https://shenzhen.****.aliyuncdn.com/2defb8b2cb85b87206646055c951f9c9/629487
66/sv/4841bb0f-1810a5fc460/4841bb0f-1810a5****.mp4",
    "Status":"success",
    "MediaId":"affab1a4c6ed4408aead501f32b5****",
    "FilterPolicy":"{"SliceFlag":false,"ResultType":"Single"}",
    "TaskIds":"1460435****",
    "EventType":"SubmitMediaRefreshComplete",
    "EventType":"SubmitMediaRefreshComplete",
    "EventTime":"2022-05-30T08:59:21Z",
    "MediaRefreshJobId":"c5ae61bf9af1****",
    "TaskType":"refresh"
}
```

● 任务失败

{
"Status":"fail",
"MediaId":"e8a73a514fb74fd79ff77c26dbfb****",
"FilterPolicy":"{"SliceFlag":false,"ResultType":"Single"}",
"EventType":"SubmitMediaRefreshComplete",
"EventTime":"2022-05-30T08:56:14Z",
"MediaRefreshJobId":"aa23298375bd****",
"TaskType":"refresh",
"ErrorCode":"InvalidDomain.NotFound",
"ErrorMessage":"Can't find domain."
}

13.视频安全

13.1. 概述

视频点播提供了完善的内容安全保护机制,您可以根据不同业务场景的安全需求,选择访问限制、播放中心鉴权、业务方二次鉴权、视频加密和安全下载等服务。

简介

完善的内容安全保护机制,可用于保障视频内容不被盗链、非法下载和传播。可以满足不同业务场景的安全需求,特别是在线版权问题尤为突出的独播剧、在线教育、财经金融、行业培训等领域。

视频点播提供如下安全机制:

| 安全机
制 | 安全手段 | 特点 | 安全等
级 | 使用门槛 |
|-------------|----------------------------------|--|----------|-------------------------------------|
| 访问限制 | Referer黑白名单 | 基于HTTP Header跟踪来源,但极易
伪造。 | 低 | 低,仅需云端配置。 |
| | User-Agent黑白名单 | 基于HTTP Header跟踪来源,但极易
伪造。 | 低 | 低,仅需云端配置。 |
| | IP黑白名单 | 拒绝或只允许特定IP访问,不适合大量
C端用户的分发。 | 较低 | 低,仅需云端配置。 |
| | 访问次数和独立IP数限
制 | 设置访问上限,不适合大量分发,也无
法识别非法访问。 | 较低 | 低,仅需云端配置。 |
| 播放中心
鉴权 | URL鉴权 | 生成动态变化的加密URL,可进行URL
时效控制,自定义过期时间。 | 中 | 较低,通过API获取动态
地址,或根据Key自助生
成。 |
| 业务方二
次鉴权 | 透传业务请求信息给客
户自定义鉴权中心来判
断合法性 | 客户添加自定义的业务请求信息,通过
自建鉴权中心,更加精准识别合法请
求。 | 较高 | 较高,需部署鉴权中
心,并确保高可用。 |
| | 阿里云视频加密(私有
加密) | 云端一体的视频加密解决方案,采用私
有加密算法,并确保链路的安全传输。 | 高 | 较低,简单配置并集成
阿里云播放器即可。 |
| | HLS标准加密 | HLS通用加密方案,使用AES_128进行
内容加密,适配所有HLS播放器,但密
钥容易被窃取。 | 较高 | 高,需自建密钥管理和
令牌颁发服务,并确保
链路传输安全。 |
| 视频加密 | | | | |

| 安全机
制 | 安全手段 | 特点 | 安全等
级 | 使用门槛 |
|--------------|---------------------------------------|---|----------|---|
| | 商业DRM | 苹果Fairplay、谷歌Widevine的原生支
持,安全级别很高,满足大的版权内容
提供商的要求。 | 高 | 高,按调用License次数
进行收费,只需集成阿
里云播放器SDK。 |
| 安全下载
(缓存) | 将下载到本地的视频文
件通过私钥进行二次加
密,可离线解密播放 | 多种机制确保视频只能由唯一应用离线
解密播放。每个视频文件拥有独立的私
钥,且加密后存储,有效防止被窃取。 | 言同 | 较低,简单配置并集成
阿里云播放器即可。 |

访问限制

介绍: 在云端配置视频资源的访问策略, 可以达到基本保护。

主要手段有:

• Referer访问限制

基于HTTP协议支持的Referer机制,通过Referer跟踪来源。可配置拒绝访问的Referer黑名单,或仅允许访问的 白名单。

● User-Agent访问限制

基于HTTP协议支持的User-Agent请求头跟踪来源,可配置拒绝访问的User-Agent黑名单,或仅允许访问的白 名单。

• IP访问限制

根据用户请求的x-forwarded-for或真实建立连接的IP,可配置拒绝访问的IP黑名单,或只允许访问的白名单。 支持IP列表和掩码方式。

• 访问数限制

一定周期内,视频URL的访问次数限制、独立IP数限制。

更多信息,请参见访问限制。

播放中心鉴权

背景:固定不变的播放地址会导致视频内容被持久的非法扩散传播,且无法有效遏制。

介绍:URL鉴权通过生成动态的加密URL(包含权限验证、过期时效等信息)来区分合法请求,以达到保护视频资源的目的。

开启URL鉴权后:

- 所有媒体资源,包括视频、音频、封面、截图等地址都会进行鉴权。
- 点播的播放器SDK、获取播放地址的API/SDK都会自动生成带时效的播放URL。如需要自己生成鉴权的动态URL, 则可参见URL鉴权中的鉴权方法。

更多信息,请参见URL鉴权。

业务方二次鉴权

背景:弥补播放中心鉴权对盗链等非法请求判断较为单一,业务方二次鉴权可以引入客户业务请求的问题,鉴权 更加精准。 **介绍:**二次鉴权是指点播CDN将用户的请求透传到客户的鉴权中心,由客户自己判定该请求是否合法,CDN根据 客户的判断结果执行相应动作:允许或拒绝访问。

二次鉴权需要客户自己开发和部署鉴权中心,该鉴权中心的域名如果同时在CDN上加速,可以按照一定规则缓存客户的鉴权结果,以减轻客户鉴权中心的压力。

? 说明

如果客户的鉴权中心服务也使用CDN进行缓存加速,此时二次鉴权的请求先经过CDN,再到达客户的服务器,返回的结果被CDN缓存,无需客户的服务器应答。

- 点播CDN会默认把用户请求的headers和request_uri透传到客户自定义的鉴权中心,并根据鉴权中心返回的结果执行相应的动作。
- 如业务方可将其用户的登录Cookie或UUID等信息隐藏于播放请求中,进而透传到自己的鉴权中心以判定是否为 合法用户。

? 说明

二次鉴权使用门槛较高,需要客户自己开发和部署鉴权中心,如有需求可提交工单、联系售后进行开通和配置。

视频加密

背景:防盗链安全机制能有效保障用户的合法访问,但对于付费观看视频的场景,用户只需通过一次付费行为便 可拿到视频合法的防盗链播放URL,将视频下载到本地,进而实现二次分发。因此,防盗链方案对于视频版权保护 是远远不够的。视频文件一旦泄露,会给付费观看模式造成十分严重的经济损失。

介绍:阿里云视频加密是对视频数据加密,即使下载到本地,视频本身也是被加密的,无法恶意二次分发,可有 效防止视频泄露和盗链问题。

• 阿里云视频加密

阿里云视频加密采用私有的加密算法和安全传输机制,提供云端一体的视频安全方案,核心部分包括**加密转** 码和解密播放。

核心优势:

- 每个媒体文件拥有独立的加密钥匙,能有效避免采用单一密钥时,一个密钥的泄露引起大范围的安全问题。
- 提供信封加密机制密文Key+明文Key, 仅密文Key入库,明文Key不落存储,所有过程只在内存中,用完即 销毁。
- 提供安全的播放器内核SDK, 涵盖iOS、Android、H5、Flash等多平台, 自动对加密内容进行解密播放。
- 播放器和云端使用私有加密协议进行密文传输,不传输明文Key,有效防止密钥被窃取。
- 提供安全下载,缓存到本地的视频会再次加密,在确保无网离线播放前提下,防止视频被拷贝窃取。

↓ 注意

阿里云视频加密使用限制如下:

- 仅支持输出HLS格式。
- 只能使用阿里云播放器。
- 。 暂不支持iOS系统的网页端播放。

更多信息,请参见阿里云视频加密。

• HLS标准加密

HLS标准加密支持HTTP Live Streaming中规定的通用加密方案,使用AES_128对视频内容本身进行加密,同时 能支持所有的HLS播放器,用户可选择使用自研或开源的播放器。相比阿里云视频加密(私有加密)方案,灵活 性更好,但使用门槛更高、安全性更低。

- ・ 用户需搭建密钥管理服务,提供密钥生成(用于转码时对视频内容进行加密)和解密服务(用于播放时获取 解密密钥),也可基于阿里云KMS进行封装。
- 用户需提供令牌颁发服务,用于验证播放端的身份,避免解密密钥被非法获取。
- 播放器和云端传输明文Key, 容易被窃取。

更多信息,请参见HLS标准加密。

● 商业DRM

高端的视频节目,需要满足内容版权和提供商的安全要求,如好莱坞、华纳等。阿里视频云推出的云端DRM解 决方案,支持Fairplay和Widevine加密,提供了一站式的视频加密、license下发、播放等能力。

更多内容,请参见DRM介绍。

? 说明

视频加密方案各有优劣。一般来说,越是标准、通用,灵活性越高,但安全性越低。您需要根据业务场景, 有所取舍,选择适合的方案。

特性比较:

• 安全等级: 商业DRM>阿里云视频加密>HLS标准加密。

阿里云视频加密安全性接近商业DRM, 二者都明显高于HLS标准加密。

- 易用性: 商业DRM=阿里云视频加密>HLS标准加密。
 - 阿里云视频加密提供云端一体解决方案,只需简单配置并接入阿里云播放器即可无缝集成加密能力。
 - HLS标准加密需自建密钥管理和令牌颁发服务。
 - 。 商业DRM需购买License, 集成特定SDK等。
- 通用性: HLS标准加密>商业DRM>阿里云视频加密。
 - HLS标准加密可适配所有HLS播放场景。
 - 商业DRM只支持授权平台(如Chrome、Safari、IE、Edge浏览器, Android、iOS等)。
 - 阿里云视频加密仅支持阿里云播放器(Android、iOS、H5、Flash)。
- 使用费用: 阿里云视频加密=HLS标准加密<<商业DRM。

阿里云视频加密和HLS标准加密都可免费使用,商业DRM需要支付额外的License费用。

安全下载(缓存)

背景:对于视频类应用,特别是在移动端(Android、iOS),一般会有缓存、下载到本地的需求,并同时希望能进行安全保护防止被拷贝后恶意播放或传播。阿里云播放器推出的安全下载功能可以有效保护下载到本地的视频内容。

介绍:安全下载是指将视频文件通过私钥进行二次加密,下载后在播放器SDK内部完成视频解密,保障离线视频 仅能通过唯一应用(安全下载中设定的bundlelD或keystore)进行安全播放的一种下载方式。

核心优势:

- 下载后视频再次播放不需要联网,可离线解密、播放,且只能由该应用播放。
- 私钥文件加密后存储,有效防止被窃取。

• 每个视频文件在每个应用上都有独立的私钥,即便单个泄露也不会影响其它视频。

使用时请先在点播控制台开启安全下载,具体操作,请参见下载设置。暂时只支持Android端播放器和iOS端播放器。

13.2. 访问限制

视频点播支持通过设置Referer黑白名单、User-Agent黑白名单、IP黑白名单进行访问限制,还支持URL访问次数和独立IP数限制。本文为您详细介绍以上访问限制。

简介

访问限制是在云端配置视频资源的访问策略,达到基本的保护目的,具有使用门槛低(仅云端配置不需要额外开发)、快速生效等优点,主要手段有:

- Referer访问限制: Referer黑白名单。
- User-Agent访问限制: User-Agent黑白名单。
- IP访问限制: IP黑白名单。
- 访问数限制:一定周期内,视频URL的访问次数和独立IP数限制。

? 说明

User-Agent访问限制和**访问数限制**,由于配置繁琐且存在误操作风险,暂未开放控制台设置,如有需求可 提交<mark>工单</mark>或联系阿里云售后处理。

Referer黑白名单

- 介绍
 - 基于HTTP协议支持的Referer机制,通过Referer跟踪来源,对来源进行识别和判断,用户可配置访问的 Referer黑、白名单(二者互斥)来限制视频资源被访问的情况。
 - 支持黑名单和白名单两种模式,访客对资源发起请求后,请求到达CDN 节点,节点会根据用户预设的防盗链
 黑名单或白名单进行过滤,符合规则可顺利请求到视频数据;若不符合,请求会被拒绝,并返回403响应码。
 - 配置后会自动添加泛域名支持,例如填写example.com,最终配置生效的是*.example.com,所有子级域名都会生效。
 - 。由于移动端一般拿不到Referer,当前默认支持空Referer访问,可选择关闭。

具体操作,请参见防盗链。

• 示例

设置点播域名example.cn-shanghai.aliyuncs.com的Referer白名单为aliyundoc.com,且不允许空Referer访问,请求数据:

```
curl -i 'http://example.cn-shanghai.aliyuncs.com/sv/5101d1f8-1643f9a****/5101d1f8-1643f9a****.
mp4'
```

返回

```
ali-186590dffig9: Lafan$ curl -i 'http://www.internetil.cn-shanghai.aliyuncs.com/sv/5101d1f8-1643f9ahit/5101d1f8-1643f9ahit.mp4'
HTTP/1.1 403 Forbidden
Server: Tengine
Date: Wed, 27 Jun 2018 04:58:15 GMT
Content-Type: text/html
Content-Length: 254
Connection: keep-alive
X-Tengine-Error: denied by Referer ACL
Access-control-Allow-Origin: *
Via: cache2.cn491[,403003]
Timing-Allow-Origin: *
EagleId: 7b7d12ca153007549506
```

当请求带上允许的Referer后即正常返回:

```
curl -i 'http://example.cn-shanghai.aliyuncs.com/sv/5101d1f8-1643f9a****/5101d1f8-1643f9a****.
mp4' \
-H 'Referer: http://www.aliyundoc.com'
```

User-Agent黑白名单

介绍

User-Agent是一个特殊字符串头,帮助服务端识别用户使用的操作系统及版本、CPU类型、浏览器及版本、浏览器渲染引擎&语言和插件等。可通过User-Agent黑白名单来限制特定浏览器或终端的访问。

- 示例
 - 如PC端IE9浏览器:

User-Agent:Mozilla/5.0(compatible;MSIE9.0;WindowsNT6.1;Trident/5.0;

○ 可模拟HTTP请求验证:

```
curl -i 'http://example.cn-shanghai.aliyuncs.com/sv/5101dlf8-1643f9a****.mp4' \
-H 'User-Agent: iPhone OS;MI 5'
```

IP黑白名单

视频点播支持配置IP黑名单或白名单,拒绝或只允许特定IP的访问。

● 支持IP列表添加,并支持添加IP网段。

例如: 网段 <u>172.16.0.1/24</u>, 24表示采用子网掩码中的前24位为有效位,即用32-24=8bit来表示主机号,该 子网可以容纳2⁸-2=254台主机,故可表示IP网段范围: 172.16.0.1~172.16.0.255。

• 可选择优先使用remote_addr或X-Forwarded-For(XFF)作为请求端IP,或者同时匹配。

具体操作,请参见IP黑白名单。

访问次数和独立IP数限制

点播服务可限制媒体资源在一定时间周期内(如1天)的最大访问次数和最大独立IP数,核心原理是所有请求先到 中心化的访问计数服务进行验证,判断是否超过预设的阈值,若超过则拒绝服务,返回HTTP 403。

- 该限制为URL(包括文件地址和签名信息)维度而非文件维度,但可配置忽略URL中的部分参数;限制的阈值到 域名粒度,可单独设置。
- 访问计数服务是多区域部署的中心化服务,一个URL只能被调度到其中一个区域,以保证中心化计数。
- CDN边缘节点收到请求后会访问中心化计数服务进行计数和验证。

小结

- 访问限制的使用门槛很低,只需简单配置即可使用,能起到一定的保护作用,特别是Web端。
- Referer和User-Agent都是基于HTTP Header,容易被伪造,安全性低。
- IP黑白名单机制和访问数限制,无法实现内容分发给大量C端用户,不适合广泛的内容消费场景,且后者在阈值 范围内也可能被非法访问。

13.3. URL鉴权

通过控制台配置URL鉴权,可以保护用户上传到视频点播的内容不被非法站点下载盗用。本文为您简要介绍了URL 鉴权的原理和鉴权A方法。

背景

一般情况下,通过配置访问Referer黑名单和白名单来实现对访客身份的识别和过滤,保护站点资源,可以解决部 分盗链问题。但由于Referer内容可以伪造,采用URL鉴权方式更为安全有效,视频点播仅支持使用鉴权A的方式进 行鉴权加密。

实现原理

URL鉴权功能通过阿里云CDN加速节点与客户资源站点配合,形成了更为安全可靠的源站资源防盗方法。

- 1. 由使用视频点播鉴权URL的客户为其用户提供加密URL(包含权限验证信息, URL组成请参见鉴权A方法说明)。
- 2. 用户使用加密后的URL向加速节点发起请求。
- 3. 加速节点对加密URL中的权限信息进行验证以判断请求的合法性,对合法请求给予正常响应,并拒绝非法请求,从而有效保护CDN客户站点资源。鉴权判断逻辑,请参见鉴权逻辑说明。

鉴权A方法说明

● 鉴权URL构成

鉴权URL由播放文件地址加上验证串(auth_key)构成。验证串是根据鉴权Key+过期时间通过MD5算法计算得 出,且具有时效性。示例如下:

http://DomainName/Filename?auth_key=timestamp-rand-uid-md5hash

| 字段 | 描述 |
|------------|--|
| DomainName | 视频点播的域名。 |
| Filename | 实际回源访问的URL,鉴权时Filename需以正斜线(/)开头。 |
| timestamp | 服务器端返回数据的时间,即应用服务端生成URL链接的时间。Unix时间戳,时间点取的是服务
器端的时间(UTC时间),需要转换为十进制整型正数,固定长度为10,用来控制鉴权URL的失
效时间。鉴权URL的失效时间=timestamp+用户设置的有效时间。有效时间由用户在视频点播控
制台上的 域名管理 > 访问控制 > URL鉴权 > 生成鉴权URL > 有效时间 设置,如果设置为
1800秒,您访问的时间超过timestamp+1800秒后,该鉴权URL > 有效时间 设置,如果设置为
1800秒,您访问的时间超过timestamp+1800秒后,该鉴权URL失效,会返回HTTP 403。
• 若通过在视频点播控制台的 域名管理 > 访问控制 > URL鉴权 > 修改配置 > 默认有效时
长生成: timestamp=生成链接的时间+默认有效时长。
• 若使用代码拼接:您可以自己定义timestamp。
例如,鉴权URL的最终的生效时间为2022年07月07日00:30:00,计算出的时间戳:2022年07月
07日00:30:00-设定的有效时间1800秒=2022年07月07日00:00:00=1657125000。 |
| rand | 随机数,一般取值为0。如果要确保每次生成的URL不同,建议使用UUID,不能包含中划线(-),例如:477b3bbc253f467b8def6711128c****。 |
| uid | 用户ID, 暂未使用(设置成0即可)。 |

鉴权字段描述如下表所示。

| 字段 | 描述 |
|---------|--|
| | 通过MD5算法计算出的验证串。数字和小写英文字母混合0-9、a-z,固定长度32。 |
| | <pre>sstring = "URI-timestamp-rand-uid-PrivateKey" md5hash = md5sum(sstring)</pre> |
| md5hash | URI:请求文件的相对地址,不包含参数,如:/Filename。 PrivateKey:控制台配置的主Key或备Key,二者皆可。 主KEY和备KEY拥有同样的效力,备Key主要用于平滑更换。若主KEY执行更换,所有使用主KEY生成的播放地址会立即失效。备KEY作为主KEY更换时,使用主KEY的播放地址不会马上中断,备KEY可以继续替代主KEY提供服务。 |
| | ◎ md5sum:表示计算MD5值,请使用开发语言对应的函数。 |

• 鉴权逻辑说明

视频点播服务器接到资源访问请求后,判断最终生成鉴权URL请求中的 timestamp 加上控制台URL鉴权中配置 的 默认有效时长 是否小于当前时间。

- o 如果 timestamp 加上 默认有效时长 小于当前时间,服务器判定过期失效,并返回HTTP 403错误。
- 如果 timestamp 加上 默认有效时长 大于当前时间,构造出一个同样的字符串,然后使用MD5算法算出 н ashValue 的值,再与请求中 md5hash 的值进行比对。
 - 结果一致, 鉴权通过, 返回资源请求。
 - 结果不一致, 鉴权失败, 返回HTTP 403错误。

使用方法

您需要在控制台的URL鉴权页面配置并开启URL鉴权,具体操作,请参见URL鉴权。

鉴权URL的获取方式

开启URL鉴权后,如果您的资源都在视频点播控制台中,控制台会自动生成带时效的鉴权URL,您也可以通过调用获取音视频播放地址接口获取鉴权URL。

⑦ 说明 开启URL鉴权后,视频、音频、封面、截图等地址都会进行鉴权。

开启URL鉴权后,如果您的资源不在视频点播控制台中,您可以使用鉴权A方法进行鉴权URL的拼接,生成动态的鉴权URL。更多信息,请参见鉴权A方法说明、鉴权URL拼接示例说明。

鉴权URL拼接示例说明

示例条件

• 回源请求对象。

http://example.aliyundoc.com/video/standard/test-***.mp4

↓ 注意 如果您的回源请求对象中有中文汉字,请先对其进行URL转码(即encode),再进行鉴权URL的拼接。

- 设置密钥为: aliyunvodexp****。
- URL设置的有效时间为2021-08-01 00:00:00,则timestamp为1627747200。

拼接流程

1. 构造签名字符串。

构造一个用于计算 Hashvalue 的签名字符串。

/video/standard/test-***.mp4-1627747200-0-0-aliyunvodexp****

2. 计算MD5值。

根据该签名字符串计算出 Hashvalue 。

HashValue = md5sum("/video/standard/test-****.mp4-1627747200-0-0-aliyunvodexp****") = 0e9048 c8c7de46b6015618f42de7****

3. 生成鉴权URL。

http://example.aliyundoc.com/video/standard/test-****.mp4?auth_key=1627747200-0-0-0e9048c8c7 de46b6015618f42de7****

当用户使用鉴权URL进行访问时,如果视频点播服务器计算出来的 HashValue 值与访问请求中带 的 md5hash 值相同,都为 0e9048c8c7de46b6015618f42de7**** ,则鉴权通过,反之鉴权失败。

13.4. 阿里云视频加密

阿里云视频加密可针对视频数据进行加密,本文为您介绍优势、整体架构和接入方法,并提供方案扩展方法。

背景

防盗链URL不能防止用户通过一次付费行为拿到付费视频合法的防盗链URL,将视频下载到本地,进而实现二次分发。因此,防盗链方案对于视频版权保护是远远不够的。

优势介绍

阿里云视频加密是**对视频数据加密**,即使下载到本地,视频本身也是被加密的,无法恶意二次分发。视频加密可 有效防止视频泄露和盗链问题,广泛用于**在线教育、财经金融、行业培训、独播剧**等在线版权视频领域。

采用阿里云私有加密算法,安全级别高,能够便捷、高效、安全地保护视频资源。

- 每个媒体文件拥有独立的加密钥匙,能有效避免采用单一密钥时,一个密钥的泄露引起大范围的安全问题。
- 视频点播提供完善的权限管理机制"子账号+播放凭证"。
- 视频点播提供信封加密机制 "密文Key+明文Key",明文Key不存储,所有过程只在内存中。
- 视频点播提供安全的播放内核SDK。

整体架构
阿里云视频加密方案包含两部分:加密转码+解密播放。



- 加密转码
 - i. App后台发起视频加密

业务方提交需要数据加密的转码作业。(上图中流程①)

ii. 视频点播获取加密密钥

视频点播服务负责通过密钥管理服务KMS生成明文Key和密文Key。(上图中流程②)

iii. 视频加密转码

使用明文Key来加密视频文件,转码完成后,明文Key丢弃,不存储。(上图中流程③)

iv. 转码完成+消息通知

保存加密后的视频文件,并给业务方发送消息通知。(上图中流程④)

- 解密播放
 - i. 业务授权

移动端的App或者Web页面访问视频时,先访问业务方自己的API或后端页面,业务方可以在这里加上自己的权限控制(例如需要登录才能播放,建议使用HTTPS)。如果业务上允许播放,则通过业务方子账号的AK访问STS,获取播放凭证,并返回给App/Web端。

ii. 获取播放地址

移动端的App或者Web页面把播放凭证和媒体ID参数传给阿里云播放服务,播放器SDK会负责剩下的播放流 程:

- 根据媒体ID从视频点播服务获取对应的多格式、多清晰的播放地址。
- 加密视频获取对应的加密钥匙。
- iii. 解密播放

视频云提供了安全的播放内核SDK,使用加密钥匙对内容解密,然后进行视频播放。

◯ 注意

- 视频加密只支持输出HLS格式。
- 需使用视频点播提供的iOS、Android、H5、Flash播放器对加密内容进行解密播放。

接入方法

前提条件:

- 1. 开通视频点播服务。
- 2. 添加加速域名。
- 3. 完成点播加速域名CNAME绑定。

接入流程:

- 登录视频点播控制台,在点播控制台左侧的导航栏选择配置管理,单击媒体处理配置 > 转码模板组,在转 码模板组页面单击添加转码模板组。具体操作,请参见普通转码模板设置。
 - 在普通转码模板区域单击添加模板,在添加模板页面将基本参数区域中的封装格式设置为hls。
 - 在视频打包模板区域单击添加模板,在视频打包模板页面打开高级参数区域的视频加密开关,使用数据加密。
- 2. 视频上传

可以通过SDK上传、OpenAPI上传、控制台上传、OSS第三方工具上传等多种方式来将已有视频文件上传到视频点播。更多信息,请参见媒体上传概述。

3. 视频转码

视频上传完成后会自动触发转码,转码完成后视频状态标记为正常可播放。更多信息,请参见<mark>普通转码模板设</mark> 置。

4. 视频播放

视频点播提供iOS、Android、H5、Flash多平台的播放SDK供集成,以便您在自己的应用或网站中完成视频播放。

⑦ 说明 加密视频,需使用播放凭证方式进行播放,播放器需要的参数playauth,可调用API接口或 SDK获取。更多信息,请参见获取音视频播放凭证。

- Web播放器(H5/Flash):通过集成在点播控制台的媒资库>音/视频>管理页面的Web播放器代码页
 签,用户可以在自己的Web页面中嵌入播放器并实现播放。更多信息,请参见快速接入。
- 移动端播放器(iOS/Android):用户可以通过SDK快速集成到自己的应用中,Android请参见快速集成,iOS请参见快速集成。
- 5. 视频流管理

加密转码输出的内容,流信息会标示"加密",更多信息,请参见播放信息 PlayInfo。在控制台也会展示**私有** 加密标示,便于进行多种方式的内容管理。

| 清晰度
↓ | 格式 | 分辨率 | 码率 👔 🕴 |
|----------|--------------|---------|---|
| 原始文件 | mp4 | 544*960 | 视频码率:
865.288Kbps
音频码率:
64.984Kbps |
| 流畅 | m3u8
私有加密 | 360*640 | 475.435Kbps |

方案扩展

如果终端用户有离线下载需求,为保证视频安全,建议开启安全下载。具体操作,请参见下载设置。将视频文件通 过密钥进行二次加密,下载后在SDK内部完成视频解密,保障离线视频仅能通过唯一应用进行安全播放,让离线视 频更加安全。



13.5. HLS标准加密

HLS标准加密需要配合密钥管理服务和令牌服务使用,本文为您介绍HLS标准加密的相关概念、准备工作和接入流程。

HLS加密解密流程

上传加密流程图



播放解密流程图



? 说明

- 使用密钥管理服务会产生费用, 计费详情请参见KMS计费说明。
- 令牌服务和解密服务需要自行搭建。

相关概念

• 密钥管理服务

<mark>密钥管理服务</mark>(Key Management Service,简称KMS),是一种安全管理服务,主要负责数据密钥的生产、加密、解密等工作。

● 访问控制

访问控制(Resource Access Management ,简称RAM)是阿里云为客户提供的用户身份管理与资源访问控制服 务。

数据密钥

数据密钥(Data Key,简称DK)也称明文密钥。DK为加密数据使用的明文数据密钥。

• 信封数据密钥

信封数据密钥(Enveloped Dat a Key,简称EDK)也称密文密钥。EDK为通过信封加密技术保密后的密文数据密 钥。

准备工作

- 开通视频点播服务并登录视频点播控制台,开启对应服务区域的存储空间,具体操作,请参见开通存储管理。
- 2. 在视频点播控制台上配置加速域名,并开启该域名视频相关中的HLS标准加密参数透传,开启后 MtsHlsUriToken参数可以重写。具体操作,请参见添加加速域名、HLS标准加密参数透传。
- 3. 登录RAM访问控制,获取并保存AccessKey ID和AccessKey Secret。
- 4. 开通密钥管理服务并获取Service Key。

⑦ 说明 Service Key是密钥管理服务的一种加密主Key, 接入标准加密的密钥必须要使用该Service Key 生成。视频点播控制台暂不支持用户自主创建Service Key。

- i. 提交工单申请创建Service Key。Service Key与视频存储的源站区域必须一致,例如:视频存储在华东2,则Service Key必须是华东2。
- ii. 申请成功后,需调用GenerateDataKey接口,请求参数 KeyId 传入别名alias/acs/vod,请求后的返回 参数 KeyId 将用于后续的转码处理。
- 5. 已搭建服务端SDK,具体操作,请参见服务端SDK安装。

接入流程

1. 添加加密模板和不转码模板。

HLS标准加密转码需要创建两个转码模板:加密模板和不转码模板。

不转码模板在开启对应服务区域的存储空间后会自动生成。

○ 注意 目前点播上传视频默认都会自动触发转码(自动触发暂不支持HLS标准加密)),因此对于标准加密为防止自动触发转码,需要先使用不转码模板上传视频(该类模板不会自动触发转码),然后再调用提交媒体转码作业接口发起标准加密转码。

添加加密模板并保存加密模板ID,操作流程如下:

- i. 登录视频点播控制台。
- ii. 在配置管理区域,选择媒体处理配置 > 转码模板组 > 添加转码模板组。
- iii. 在添加转码模板组页面, 输入模板组名称。
- iv. 在普通转码模板区域单击添加模板开始创建转码模板。
- v. 在基本参数区域,选择封装格式为hls。

- vi. 在**视频参数**区域、**音频参数**区域及**条件转码参数**区域,可根据实际需要配置相关参数,各参数的含义 及配置限制请参见音视频转码。
- vii. 在高级参数区域,开启视频加密,并勾选私有加密选项(系统默认勾选,否则不加密)。

| 普通转码模板 | 基本参数 | |
|------------------------|------------------|---|
| | 封装格式 | his 🗸 |
| +添加模板 | 清晰度 🛛 | 滴畅 |
| 视频打包模板
+添加模板 | 视频参数 | |
| 字篇打包模板
→添加微版 | 编码格式 | H.264 V |
| | 码率(Kbps) 🕜 | 400 |
| | 分辨率(宽x高) 🔮 | 640 × 360 |
| | 較率(fps) 🚱 | 25 |
| | 关键帧最大间隔
(帧) 🕑 | 250 |
| | 水印 🚱 | |
| | 音频参数 | |
| | ▼ 高级参数 | |
| | 分片时长(s) 🖉 | 10 |
| | 视频加密 🖉 | |
| | 加密方式 | 私有加密 ORM加密 |
| | | HLS加密方式,默认开启阿里云私有加密(阿里云私有加密暂不支持OS系统的网页端播放)。如素使用标准加密,详见HLS标准加密 |

⑦ 说明 该模板在调用提交媒体转码作业接口时,通过TemplateGroupId参数传递,如此视频点播将按照设置的模板和传递的密钥信息进行标准加密转码。

viii. 单击保存,自动返回转码模板组页面,获取并自行保存加密模板ID。

2. RAM授权。

使用RAM服务给视频点播授权访问业务方密钥管理服务的权限,进入RAM授权页面,单击同意授权。

| | 云资源访问授权
如需修改角色权限,请前往 RAM 控制台 <mark>角色管理</mark> 中设置,需要注意的是,错误的配置可能导致云产品无法获取到必要的权限。 |
|---|---|
| | |
| VOD 请求获取访问下方是系统创建的可 AliyunVODDefau | 问您云资源的权限。
「供 VOD 使用的角色,授权后,VOD 拥有对您云资源相应的访问权限。
l itRole 展开详情 |
| | |
| 同意授权取消 | |

3. 搭建密钥管理服务, 封装阿里云密钥管理服务(KMS)。

调用GenerateDataKey接口生成一个AES_128密钥,该接口只需要传Keyld(Service Key)和KeySpec(固定 为:AES_128)即可,其他参数不用传,否则可能加密失败。

调用成功后保存返回参数 CiphertextBlob (密文密钥)的值。

⑦ 说明 使用密钥会产生费用,具体费用说明,请参见API调用费用。

4. 搭建令牌颁发服务, 生成MtsHlsUriToken。

Java示例代码以及示例代码需要手动变更的地方如下所示:

- ENCRYPT_KEY:加密字符串,长度为16,用户自行定义。
- INIT_VECTOR: 自定义字符串, 长度为16, 不能含有特殊字符。
- playToken.generateToken(""): 自定义字符串,长度为16。

最终代码所生成的Token即是MtsHlsUriToken。

```
import com.sun.deploy.util.StringUtils;
import org.apache.commons.codec.binary.Base64;
import javax.crypto.Cipher;
import javax.crypto.spec.IvParameterSpec;
import javax.crypto.spec.SecretKeySpec;
import java.util.Arrays;
public class PlayToken {
   //非AES生成方式,无需以下参数
   private static String ENCRYPT KEY = ""; //加密字符串,用户自行定义
   private static String INIT_VECTOR = ""; //长度为16的自定义字符串,不能有特殊字符
   public static void main(String[] args) throws Exception {
      PlayToken playToken = new PlayToken();
      playToken.generateToken("");
   }
   /**
    * 根据传递的参数生成令牌
    * 说明:
    * 1、参数可以是业务方的用户ID、播放终端类型等信息
    * 2、调用令牌接口时生成令牌Token
    * @param args
    * @return
    */
   public String generateToken(String... args) throws Exception {
      if (null == args || args.length <= 0) {
          return null;
      String base = StringUtils.join(Arrays.asList(args), " ");
       //设置30s后,该token过期,过期时间可以自行调整
      long expire = System.currentTimeMillis() + 30000L;
      base += " " + expire; //自定义字符串,base的最终长度为16位字符(此例中,时间戳占13位,下
划线 ( ) 占1位,则还需传入2位字符。实际配置时也可按需全部更改,最终保证base为16位字符串即可。)
      //生成token
      String token = encrypt(base, ENCRYPT_KEY);
       System.out.println(token);
       //保存token,用于解密时校验token的有效性,例如: 过期时间、token的使用次数
      saveToken(token);
      return token;
   }
    * 验证token的有效性
```

```
* 1、解密接口在返回播放密钥前,需要先校验Token的合法性和有效性
* 2、强烈建议同时校验Token的过期时间以及Token的有效使用次数
 * @param token
 * @return
 * @throws Exception
*/
public boolean validateToken(String token) throws Exception {
   if (null == token || "".equals(token)) {
      return false;
   }
   String base = decrypt(token, ENCRYPT_KEY);
   //先校验token的有效时间
   Long expireTime = Long.valueOf(base.substring(base.lastIndexOf("_") + 1));
   if (System.currentTimeMillis() > expireTime) {
      return false;
   }
   //从DB获取token信息,判断token的有效性,业务方可自行实现
   Token dbToken = getToken(token);
   //判断是否已经使用过该token
   if (dbToken == null || dbToken.useCount > 0) {
       return false;
   }
   //获取到业务属性信息,用于校验
   String businessInfo = base.substring(0, base.lastIndexOf(" "));
   String[] items = businessInfo.split(" ");
   //校验业务信息的合法性,业务方实现
   return validateInfo(items);
}
/**
* 保存Token到DB
* 业务方自行实现
* @param token
*/
public void saveToken(String token) {
   //TODO 存储Token
}
/**
* 查询Token
* 业务方自行实现
* @param token
*/
public Token getToken(String token) {
  //TODO 从DB 获取Token信息,用于校验有效性和合法性
   return null;
}
/**
* 校验业务信息的有效性,业务方可自行实现
* @param infos
* @return
*/
public boolean validateInfo(String... infos) {
   //TODO 校验信息的有效性,例如UID是否有效等
   return true;
```

```
}
/**
* AES加密生成Token
* @param key
 * @param value
 * @return
 * @throws Exception
*/
public String encrypt(String value, String key) throws Exception {
    IvParameterSpec e = new IvParameterSpec(INIT VECTOR.getBytes("UTF-8"));
    SecretKeySpec skeySpec = new SecretKeySpec(key.getBytes("UTF-8"), "AES");
    Cipher cipher = Cipher.getInstance("AES/CBC/PKCS5PADDING");
    cipher.init(Cipher.ENCRYPT MODE, skeySpec, e);
   byte[] encrypted = cipher.doFinal(value.getBytes());
   return Base64.encodeBase64String(encrypted);
}
/**
* AES解密token
 * @param key
 * @param encrypted
 * @return
 * @throws Exception
 */
public String decrypt(String encrypted, String key) throws Exception {
   IvParameterSpec e = new IvParameterSpec(INIT VECTOR.getBytes("UTF-8"));
    SecretKeySpec skeySpec = new SecretKeySpec(key.getBytes("UTF-8"), "AES");
   Cipher cipher = Cipher.getInstance("AES/CBC/PKCS5PADDING");
   cipher.init(Cipher.DECRYPT_MODE, skeySpec, e);
   byte[] original = cipher.doFinal(Base64.decodeBase64(encrypted));
   return new String(original);
}
/**
 * Token信息,业务方可提供更多信息,这里仅仅给出示例
*/
class Token {
   //Token的有效使用次数,分布式环境需要注意同步修改问题
   int useCount;
   //token内容
    String token;
} }
```

5. 搭建解密服务。

注意 解密服务在播放视频前就需要启动,否则视频无法正常解密。

解密密钥EDK(密文密钥),调用Decrypt接口进行解密。如果业务方需要对解密接口进行安全验证,则需要提供令牌生成服务,生成的令牌能够在解密服务中被解析验证。

解密接口返回的数据,是GenerateDataKey生成的两种密钥中的明文密钥(PlainText)经过base64decode之后的数据。

Java示例代码以及示例代码需要手动变更的地方如下所示:

• region: 填写地域,例如华东2(上海),填写 cn-shanghai 。

。 AccessKey: 填写对应账号的AccessKey ID和AccessKey Secret。

○ httpserver: 根据需求选择服务启动的端口号。

```
import com.aliyuncs.DefaultAcsClient;
import com.aliyuncs.exceptions.ClientException;
import com.aliyuncs.http.ProtocolType;
import com.aliyuncs.kms.model.v20160120.DecryptRequest;
import com.aliyuncs.kms.model.v20160120.DecryptResponse;
import com.aliyuncs.profile.DefaultProfile;
import com.sun.net.httpserver.Headers;
import com.sun.net.httpserver.HttpExchange;
import com.sun.net.httpserver.HttpHandler;
import com.sun.net.httpserver.HttpServer;
import com.sun.net.httpserver.spi.HttpServerProvider;
import org.apache.commons.codec.binary.Base64;
import java.io.IOException;
import java.io.OutputStream;
import java.net.HttpURLConnection;
import java.net.InetSocketAddress;
import java.net.URI;import java.util.regex.Matcher;
import java.util.regex.Pattern;
public class HlsDecryptServer {
   private static DefaultAcsClient client;
   static {
       //KMS的区域,必须与视频对应区域
       String region = "";
       //访问KMS的授权AccessKey信息
       String accessKeyId = "";
       String accessKeySecret = "";
       client = new DefaultAcsClient(DefaultProfile.getProfile(region, accessKeyId, accessK
eySecret));
   }
    /**
    * 说明:
    * 1、接收解密请求,获取密文密钥和令牌Token
     * 2、调用KMS decrypt接口获取明文密钥
     * 3、将明文密钥base64decode返回
    * /
   public class HlsDecryptHandler implements HttpHandler {
       /**
        * 处理解密请求
        * @param httpExchange
        * @throws IOException
        */
       public void handle(HttpExchange httpExchange) throws IOException {
           String requestMethod = httpExchange.getRequestMethod();
           if ("GET".equalsIgnoreCase(requestMethod)) {
               //校验token的有效性
               String token = getMtsHlsUriToken(httpExchange);
               boolean validRe = validateToken(token);
               if (!validRe) {
                   return;
               }
               //从URL中取得密文密钥
               String ciphertext = getCiphertext(httpExchange);
               if (null == ciphertext)
                   return;
               //从KMS中解密出来,并Base64 decode
```

```
byte[] key = decrypt(ciphertext);
       //设置header
       setHeader(httpExchange, key);
       //返回base64decode之后的密钥
       OutputStream responseBody = httpExchange.getResponseBody();
       responseBody.write(key);
       responseBody.close();
    }
}
private void setHeader(HttpExchange httpExchange, byte[] key) throws IOException {
   Headers responseHeaders = httpExchange.getResponseHeaders();
   responseHeaders.set("Access-Control-Allow-Origin", "*");
   httpExchange.sendResponseHeaders(HttpURLConnection.HTTP OK, key.length);
}
/**
* 调用KMS decrypt接口解密,并将明文base64decode
* @param ciphertext
* @return
*/
private byte[] decrypt(String ciphertext) {
   DecryptRequest request = new DecryptRequest();
   request.setCiphertextBlob(ciphertext);
   request.setProtocol(ProtocolType.HTTPS);
   trv {
       DecryptResponse response = client.getAcsResponse(request);
       String plaintext = response.getPlaintext();
       //注意: 需要base64 decode
       return Base64.decodeBase64(plaintext);
    } catch (ClientException e) {
       e.printStackTrace();
       return null;
   }
}
/**
* 校验令牌有效性
* @param token
* @return
*/
private boolean validateToken(String token) {
   if (null == token || "".equals(token)) {
       return false;
   }
   //TODO 业务方实现令牌有效性校验
   return true;
}
/**
* 从URL中获取密文密钥参数
* @param httpExchange
* @return
*/
private String getCiphertext(HttpExchange httpExchange) {
   URI uri = httpExchange.getRequestURI();
   String queryString = uri.getQuery();
   String pattern = "CipherText=(\\w*)";
   Pattern r = Pattern.compile(pattern);
   Matcher m = r.matcher(queryString);
   if (m.find())
```

```
return m.group(1);
        else {
           System.out.println("Not Found CipherText Param");
           return null;
        }
    }
    /**
     * 获取Token参数
    * @param httpExchange
     * @return
     */
    private String getMtsHlsUriToken(HttpExchange httpExchange) {
        URI uri = httpExchange.getRequestURI();
        String queryString = uri.getQuery();
       String pattern = "MtsHlsUriToken=(\\w*)";
       Pattern r = Pattern.compile(pattern);
        Matcher m = r.matcher(queryString);
        if (m.find())
           return m.group(1);
        else {
           System.out.println("Not Found MtsHlsUriToken Param");
           return null:
        }
    }
}
/**
 * 服务启动
 * @throws IOException
*/
private void serviceBootStrap() throws IOException {
   HttpServerProvider provider = HttpServerProvider.provider();
    //监听端口可以自定义,能同时接受最多30个请求
   HttpServer httpserver = provider.createHttpServer(new InetSocketAddress(8099), 30);
    httpserver.createContext("/", new HlsDecryptHandler());
    httpserver.start();
   System.out.println("hls decrypt server started");
}
public static void main(String[] args) throws IOException {
   HlsDecryptServer server = new HlsDecryptServer();
    server.serviceBootStrap();
} }
```

6. 上传视频。

使用**不转码**模板创建视频上传凭证和地址。控制台具体操作,请参见<mark>控制台上传</mark>;服务端接口上传,请参见<mark>获</mark> 取音视频上传地址和凭证。

7. 接收上传完成回调消息。

通过<mark>设置事件通知配置</mark>设置回调,通过<mark>查询事件通知配置</mark>查询回调消息,当接收到<mark>视频上传完成</mark>的回调消 息,则表明文件已经上传到视频点播。

8. 发起标准加密转码。

调用提交媒体转码作业接口发起标准加密转码。

Java示例代码以及示例代码需要手动变更的地方如下所示:

- request.setTemplateGroupId(""): 传入加密模板Ⅳ。
- ∘ request.setVideold(""): 传入视频ID。
- encryptConfig.put("CipherText",""): 传入步骤三获取的CiphertextBlob值。

```
• encryptConfig.put("DecryptKeyUri","): 传入播放地址、 CiphertextBlob 值以及 MtsHlsUriToken 。
    以在本地的8099端口为例,播放地址为: http://172.16.0.1:8099?CipherText=CiphertextBlob值&MtsH
    lsUriToken=MtsHlsUriToken值 。
    import com.alibaba.fastjson.JSON;
    import com.alibaba.fastjson.JSONObject;
    import com.aliyuncs.DefaultAcsClient;
    import com.aliyuncs.exceptions.ClientException;
    import com.aliyuncs.profile.DefaultProfile;
    import com.aliyuncs.vod.model.v20170321.SubmitTranscodeJobsRequest;
    import com.aliyuncs.vod.model.v20170321.SubmitTranscodeJobsResponse;
    public class SubmitTranscodeJobs {
       private static String accessKeyId = "accessKeyId";
       private static String accessKeySecret = "accessKeySecret";
       public static SubmitTranscodeJobsResponse submitTranscodeJobs(DefaultAcsClient client) t
    hrows Exception{
            SubmitTranscodeJobsRequest request = new SubmitTranscodeJobsRequest();
            request.setTemplateGroupId("");
           request.setVideoId("");
            JSONObject encryptConfig = new JSONObject();
            encryptConfig.put("CipherText", "");
            encryptConfig.put("DecryptKeyUri","");
            encryptConfig.put("KeyServiceType", "KMS");
            request.setEncryptConfig(encryptConfig.toJSONString());
            return client.getAcsResponse(request);
        1
        public static void main(String[] args) throws ClientException {
            String regionId = "cn-shanghai"; // 点播服务接入区域
            DefaultProfile profile = DefaultProfile.getProfile(regionId, accessKeyId, accessKeyS
    ecret);
            DefaultAcsClient client = new DefaultAcsClient(profile);
            SubmitTranscodeJobsResponse response;
            try {
               response = submitTranscodeJobs(client);
               System.out.println("RequestId is:"+response.getRequestId());
               System.out.println("TranscodeTaskId is:"+response.getTranscodeTaskId());
               System.out.println("TranscodeJobs is:"+ JSON.toJSON(response.getTranscodeJobs())
    );
            } catch (Exception e) {
               e.printStackTrace();
            }
        }
    }
9. 验证加密转码是否成功。
```

9. 验证加密转码是否成功。

您可以登录视频点播控制台查看该视频的视频地址,通过以下三种方式来逐步判断标准加密是否成功。

- 当视频加密转码后,如果该视频的视频地址只有一个M3U8格式的视频地址,那么该视频状态为转码失败。
- 当视频加密转码后,如果视频不只有M3U8格式的输出(例如还存在格式为MP4的原始文件),只需查看
 M3U8格式后是否带有标准加密,一般情况下,如果存在则表明标准加密已成功。
- 如果以上两种方式都不能判断,那么可以将带有加密标志的M3U8文件的地址拷贝出来,使用 curl -v "M

3U8**文件地址**" , 查看获取到的M3U8内容是否存在 URI="<业务方在发起标准加密时传递的解密地址,即加密配置 EncryptConfig中的DecryptKeyUri参数值>" 关键信息, 有则表明为标准加密且加密成功。

播放流程

1. 获取视频的播放地址和凭证。

调用获取音视频播放地址和获取音视频播放凭证接口获取视频的播放地址和凭证。

2. 传入认证信息。

获取M3U8文件地址后,播放器会解析M3U8文件中的EXT-X-KEY标签中的URI并访问,从而获取到带密文密钥的解密接口URI,此URI为您发起标准加密时传递的加密配置EncryptConfig中的DecryptKeyUri参数值。

若只允许合法用户才可以访问,那么需要播放器在获取解密密钥时携带您承认的认证信息,认证信息可以通过MtsHlsUriToken参数传入。

示例:

- · 视频的播放地址为: https://demo.aliyundoc.com/encrypt-stream****-hd.m3u8
 /,则请求时需要携带
 MtsHlsUriToken 参数传入。
- o 最终请求地址为: https://demo.aliyundoc.com/encrypt-stream****-hd.m3u8?MtsHlsUriToken=<令牌>
- o 解密地址为: https://demo.aliyundoc.com?Ciphertext=ZjJmZGViNzUtZWY1Mi00Y2RlLTk3MTMtOT****
- 最终解密请求地址为: https://demo.aliyundoc.com?Ciphertext=ZjJmZGViNzUtZWY1Mi00Y2RlLTk3MTMt0
 T****&MtsHlsUriToken=<颁发的令牌>
 。
- 3. 播放。

播放器在解析到解密地址URI时会自动请求解密接口获取解密密钥,拿到解密密钥去解密加密过的ts文件进行播放。

相关文档

• HLS标准加密常见问题

13.6. HLS(M3U8)标准加密改写

开启M3U8标准加密改写功能后,可以改写HLS(HTTP Live Streaming)协议的M3U8文件(Media Playlist,媒体播放列表)。改写成功后会在M3U8文件内 #EXT-X-KEY 标签后面增加加密参数(包括加密算法、密钥URI地址和鉴权参数),客户端收到被改写的M3U8文件以后,将会使用带鉴权参数的密钥URI来发起请求,从CDN节点获取到密钥以后将会使用对应的加密算法和密钥来解密TS文件。即通过配置M3U8标准加密改写功能,可以实现对HLS数据访问过程的加密保护。

您可以参考以下内容,详细了解HLS(M3U8)标准加密改写:

- •
- •
- •
- 操作步骤

背景信息

HLS(HTTP Live Streaming的缩写)是一个由苹果公司提出的基于HTTP的流媒体网络传输协议。HLS协议基于 HTTP协议,客户端按照顺序使用HTTP协议下载存储在服务器上的文件。HLS协议规定,视频的封装格式是 TS(Transport Stream),除了TS视频文件本身,还定义了用来控制播放的M3U8文件(文本文件)。HLS协议的 工作原理是把整个视频流分割成一个个小的TS格式视频文件来传输,在开始一个流媒体会话时,客户端会先下载 一个包含TS文件URL地址的M3U8文件(相当于一个播放列表),给客户端用于下载TS文件。

HLS基本字段:

- #EXTM3U : M3U8文件头,必须放在第一行。
- EXT-X-MEDIA-SEQUENCE : 第一个TS分片的序列号,一般情况下是0,但是在直播场景下,这个序列号标识 直播段的起始位置; #EXT-X-MEDIA-SEQUENCE:0 。
- #EXT-X-TARGETDURATION: 1 每个分片TS的最大的时长; #EXT-X-TARGETDURATION: 1 0 , 表示每个分片的最大时长是10秒。
- #EXT-X-ALLOW-CACHE : 是否允许cache, #EXT-X-ALLOW-CACHE:YES 、 #EXT-X-ALLOW-CACHE:NO , 默认 情况下是YES。
- #EXT-X-ENDLIST : M3U8文件结束符。
- #EXTINF : extra info, 分片TS的信息,如时长,带宽等;一般情况下是 后面可以跟其他的信息,逗号之前是当前分片的TS时长。分片时长要小于
 #EXT-X-TARGETDURATION 定义的 值。
- #EXT-X-VERSION : M3U8版本号。
- #EXT-X-DISCONTINUITY : 该标签表明其前一个切片与下一个切片之间存在中断。
- #EXT-X-PLAYLIST-TYPE : 表明流媒体类型。
- #EXT-X-KEY: 是否加密解析。例如: #EXT-X-KEY:METHOD=AES-128,URI="https://example.com/video.key?token=xxx" 加密算法是AES-128,密钥通过请求 https://example.com/video.key?token=xxx 来获取, 密钥请求回来以后存储在本地,并用于解密后续下载的TS视频文件。

技术原理

- 1. 客户端向CDN节点发起对M3U8文件的访问请求,例如: http://example.com/media/index.m3u8?MtsHlsUri Token=xxx 。
- 2. CDN节点对客户端的访问请求进行校验,校验通过。
- 3. CDN节点从源站下载原始M3U8文件,并缓存原始M3U8文件。
- 4. CDN节点对原始M3U8文件的 #EXT-X-KEY 标签进行改写,增加加密方式、密钥URI和鉴权参数,例如: #EX T-X-KEY:METHOD=AES-128,URI="https://example.com/video.key?MtsHlsUriToken=xxx"。
- 5. CDN节点将改写后的M3U8文件返回给客户端。
- 6. 客户端解析改写后的M3U8文件,拿到密钥URI地址 https://example.com/video.key? MtsHlsUriToken=xxx ,并发起访问请求。
- 7. CDN节点收到客户端请求,鉴权通过之后,将key文件返回给客户端。
- 8. 客户端继续解析改写后的M3U8文件,从CDN节点下载其中的TS视频文件。
- 9. 客户端使用key文件内的密钥和前面 #EXT-X-KEY 标签内定义的加密算法来解密TS视频文件。

适用场景

HLS协议采用M3U8文件来告知客户端视频文件播放列表,客户端拿到M3U8文件以后就可以直接播放视频,为了避免源站的视频文件被非授权客户端访问,需要对HLS协议使用的TS视频文件做加密,对TS视频文件做了加密以后,还需要告知客户端解密方法,这里就可以通过配置M3U8标准加密改写功能,通过 #EXT-X-KEY 标签来告知客户端加密算法、密钥URI和鉴权key。

操作步骤

HLS(M3U8)标准加密改写的操作步骤,请参见HLS标准加密参数透传。