ALIBABA CLOUD

# 阿里云

## 媒体处理

## 开发指南

文档版本：20220706

阿里云

# 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。

1. 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。

2. 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。

3. 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<br>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<br>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。 | 注意<br>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。 | 说明<br>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 list --instanceid` *Instance_ID* |
| [] 或者 [a\|b] | 表示可选项，至多选择一个。 | `ipconfig [-all\|-t]` |
| {} 或者 {a\|b} | 表示必选项，至多选择一个。 | `switch {active\|stand}` |

# 目录

1.开发指南概述 06
2.概念介绍 09
2.1. 作业和管道 09
2.2. 转码模板 10
2.3. 媒体工作流 11
3.前置条件 14
3.1. 前置条件概述 14
3.2. 申请RAM用户权限 14
3.3. 配置加速域名 21
4.消息通知 27
4.1. 设置转码消息通知 27
4.2. 绑定管道 29
4.3. 设置工作流消息通知 32
4.4. 接收消息通知 34
5.控制台用户路径 36
5.1. 媒资转码 36
5.1.1. 转码概述 36
5.1.2. 创建倍速转码任务 38
5.1.3. 创建窄带高清TM1.0转码任务 40
5.1.4. 创建窄带高清TM2.0转码任务 44
5.1.5. 创建分辨率倍增任务 48
5.2. 视频AI 51
5.2.1. 视频AI概述 51
5.2.2. 视频生产 54
5.2.3. 媒体审核 59
5.2.4. 视频DNA 64

5.2.5. 智能标签 69
5.3. 视频加密 71
5.3.1. 视频加密概述 72
5.3.2. 阿里云私有加密 72
5.3.3. HLS标准加密 74
6.API用户路径 78
6.1. API用户路径概述 78
6.2. 媒资转码 81
6.3. 视频拼接剪辑 83
6.4. 视频AI 86
6.5. 数字水印 88
6.6. HLS标准加密 90
6.7. 上传视频文件 98
6.7.1. 设置RAM用户和授权 98
6.7.2. 请求安全令牌 100
6.7.3. 视频播放 102
6.8. 媒体库管理 105
6.8.1. 媒体库简介 105
6.8.2. 媒体基本信息 106
6.8.3. 媒体详细信息 110
6.8.4. 标签管理 113

# 1.开发指南概述

阿里云媒体处理MPS（ApsaraVideo Media Processing，原MTS）以经济、弹性和高可扩展的转换方法，将多媒体数据转码成适合在全平台播放的格式，并基于海量数据对媒体的内容进行多模态分析，实现智能审核、智能生产、版权保护等功能。开发指南从媒体处理功能实现的角度，介绍媒体处理的功能和使用方法。

## 整体架构

媒体处理服务功能包括管理控制台、服务API和软件开发工具包。您可以通过它们管理、使用转码服务，也可以将转码功能集成到您自己的应用和服务中。

- 媒体处理：基于阿里云云计算服务构建，借助云计算的弹性伸缩特性，按需提供转码能力，最大限度满足您的转码需求。
- 视频AI：可以对音视频文件内容进行智能识别、智能分析和智能审核，准确理解文件内容，提升视频点击转化。
- 工作流：支持自定义工作流，通过控制台或API创建工作流，按需搭建云端音视频处理流程。

流程说明：

| 步骤 | 说明 |
| --- | --- |
| ① | 用户将各种形式的音视频文件上传至输入Bucket。 |
| ② | 根据用户需求对音视频文件进行相应的处理。 |
| ③ | 将处理好的文件存储至输出Bucket。 |
| ④ | 文件传输至CDN。 |
| ⑤ | CDN根据实际情况对内容进行分发播放。 |

## 全局配置

| 配置分类 | 描述 | 配置原因 |
| --- | --- | --- |
| 账号与授权 | 媒体处理支持访问控制（RAM）服务和临时安全令牌（STS）服务。您可以通过如下方式进行授权操作：<br>• 对RAM用户用系统授权策略授权。<br>• 对RAM用户自定义授权。<br>• 对角色进行STS临时授权。 | 媒体处理会根据每一次发起的请求，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。 |
| （可选）消息通知 | 媒体处理支持通过消息回调的方式获取消息通知。<br>• HTTP回调（兼容HTTPS）。<br>• 阿里云消息服务（MNS）回调。 | 媒体处理支持消息服务（MNS）的消息队列和消息通知功能。如需接收回调消息，需要先启用消息通知功能。 |

## 功能使用

| 功能 | 描述 | 触发方式 |
| --- | --- | --- |
| 媒资转码 | 将多媒体数据转码成适合在全平台播放的格式。 | 控制台、API |
| 视频截图 | 可以为输入视频文件创建指定时间节点的截图，目前仅支持生产jpg格式的图片。 | API<br>说明 媒体处理控制台只能对截图任务进行查询，不能提交截图任务。 |
| 智能封面 | 通过对视频内容的理解，选出最优的关键帧或关键片段作为视频封面，提升视频点击转化及用户体验。 | 控制台、API |
| 视频审核 | 从语音、文字、视觉等多维度精准识别媒体文件中包含的违禁内容，同时支持视频图像文件内容质量审核功能。 | 控制台、API |
| 视频DNA | 提取并比对视频中的图像、音频等指纹特征，解决重复视频查找、视频片段查源、原创识别等问题。 | 控制台、API |
| 智能标签 | 智能标签通过分析识别，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。 | 控制台、API |
| 视频加密 | 对视频中的内容进行加密，可有效防止视频泄露和盗链问题，广泛用于在线教育及财经等领域。 | 控制台、API |

| 功能 | 描述 | 触发方式 |
| --- | --- | --- |
| 媒体工作流 | 媒体工作流可以对上传的文件进行转码、审核、打包、分析等处理，并生成媒体文件。 | 控制台、API |

# 2.概念介绍

## 2.1. 作业和管道

本文为您介绍媒体处理中的几个基本概念及其关系，以便您更好的理解和使用媒体处理服务。

### 处理流程

1. 同步提交作业：用户侧同步提交待处理作业。
2. 异步提交作业：作业按提交顺序和优先级进入管道中排队。
3. 发送消息通知：若管道配置了消息服务MNS，则作业执行完毕后发送消息通知。

### 概念介绍

- 作业

  作业在媒体处理服务里面是一个抽象的概念，包含多种类型的作业，如转码作业、截图作业、媒体信息作业等。

  在一个作业中，包含3个关键信息：输入、输出和参数。输入和输出设置作业执行时的输入文件以及执行后的输出文件，参数则用来设置执行具体功能的详细配置。

- 管道

  当用户通过控制台或API接口提交作业后，作业会进入管道中进行排队，根据优先级和提交顺序依次执行。

  > **说明** 管道中的作业可以有多种优先级。通过调用API提交的作业可以自定义优先级，最高为10，最低为1。通过控制台提交的作业优先级默认为6，不能修改。相同优先级的作业之间，提交作业时间早的比晚的先执行，不同优先级的作业之间，高优先级的比低优先级的先执行。

- 参数

- 模板参数

  转码作业参数多，每次提交转码作业时需要填写的参数重复部分较多。转码模板为解决这一问题而设，其本质是将一些转码作业中常用的参数组合后供转码作业调用。可以减少提交作业需要填写的参数数量，简化提交作业的代码。模板可以在不同的作业中重复使用。

- API参数

  如果给每种不同的参数组合都创建模板，会导致模板的数量剧增，也使得模板管理变的复杂。所以不仅可以在模板中设置参数，也可以在实际调用API时设置对应的参数。在API中设置的参数只对当前作业生效。

- 覆盖顺序

  在调用提交转码作业接口时，如果既指定了转码模板，又设置了API参数，则API参数比模板中对应参数的优先级更高，会覆盖模板中对应的参数。

  例如：同一个视频可以转码输出多种清晰度（高清、标清等），不同清晰度的容器格式（MP4)、编码标准（H.264）、帧率（25帧）是相同的，只有码率和分辨率不同。就可以先创建（MP4+H.264+25 FPS+2 Mbps+1280×720）这样一个默认参数组合模板。调用API时，如果不设置API参数，则按照默认码率和分辨率（2 Mbps+1280×720）执行作业，如果设置API参数中的码率和分辨率为（4 Mbps+1920×1080），则按照API参数（4 Mbps+1920×1080）执行作业。

- 作业执行和结果

  作业类型不同，执行所需的时间也不同，一些作业能很快完成，但是大部分的作业都无法即时完成。由此将作业执行分为同步和异步两种方式。

  - 同步

    同步方式（例如：截图作业）的结果会立即返回。

  - 异步

    异步方式（例如：转码作业）的结果有2种查询方式：定时轮询和消息通知。

    - 定时轮询

      每个作业都有一个唯一作业ID来标识，在提交作业时会同步返回给调用方，之后可以在控制台或API接口返回数据中查看作业ID，通过作业ID查询作业结果。这种方式的缺点是不够实时。

    - 消息通知

      管道配置消息通知，就能及时获得作业结果。消息通知中包含了几个重要的信息：作业ID、用户数据和结果。

      - 作业ID

        提交作业时记录下作业ID，然后和消息通知的作业ID对比，就能知道是哪个作业的结果。

      - 用户数据

        提交作业时，可以填写自定义的用户数据参数（例如：商品ID），然后消息通知中会返回自定义的用户数据参数，这样可以不需要在业务系统中记录作业ID，使用自定义的用户数据来关联业务系统。

# 2.2. 转码模板

转码作业参数多，每次提交转码作业时需要填写的参数重复部分较多。转码模板为解决这一问题而设，其本质是将一些转码作业中常用的参数组合后供转码作业调用。媒体处理提供两种类型的转码模板：预置模板和自定义模板。

## 预置模板

根据一些常用的参数组合，预先设置好提供给用户使用的转码模板。预置模板又细分为几个类型：

- 预置静态模板：可以直接使用的转码模板。包含视频转码、音频转码、转封装等各种场景。例如：“MP4-高清”、“MP3-128K”。
- 窄带高清预置模板：窄带高清是媒体转码特有的技术。在相同的码率下，能带来更高的清晰度，这样可以在相同成本下提供更好的用户体验。包含预置窄带高清TM模板和预置窄带高清TM2.0模板。
- 预置智能模板：预置智能模板能自动根据输入文件内容的特点动态调整转码参数，在相同的清晰度下，能带来更低的码率，节省成本。

> **说明** 预置智能模板，目前仅支持通过API使用。使用预置智能模板时，要先调用提交模板分析作业接口（SubmitAnalysisJob），分析作业成功完成后，可以调用查询模板分析作业接口（QueryAnalysisJobList）获取该输入文件对应的有效预置智能模板列表。若提交的转码作业中指定的预置智能模板不在有效的列表中，则转码作业是无效的，会返回失败。

## 自定义模板

当预置模板不能满足您的实际转码需求时，可以使用自定义模板来自行定义转码参数（音频、视频、容器、转码等）组合。每个自定义模板有一个唯一模板ID。您可以创建普通模板或窄带高清模板，具体操作，请参见自定义转码模板、自定义窄带高清TM1.0模板、自定义窄带高清TM2.0模板。

# 2.3. 媒体工作流

媒体处理中的工作流是指通过配置多个单任务操作组合对一个输入进行自动化处理的过程。本文介绍媒体处理服务中媒体工作流的基本概念及执行规则，以便您更好的理解和使用媒体处理服务。

## 基本概念

- 媒体

  媒体包含一个输入（视频、音频多媒体文件）和相关的所有输出（例如：转码、截图、媒体信息、AI标签等）。输入和媒体是一一对应的，由媒体ID唯一标识。

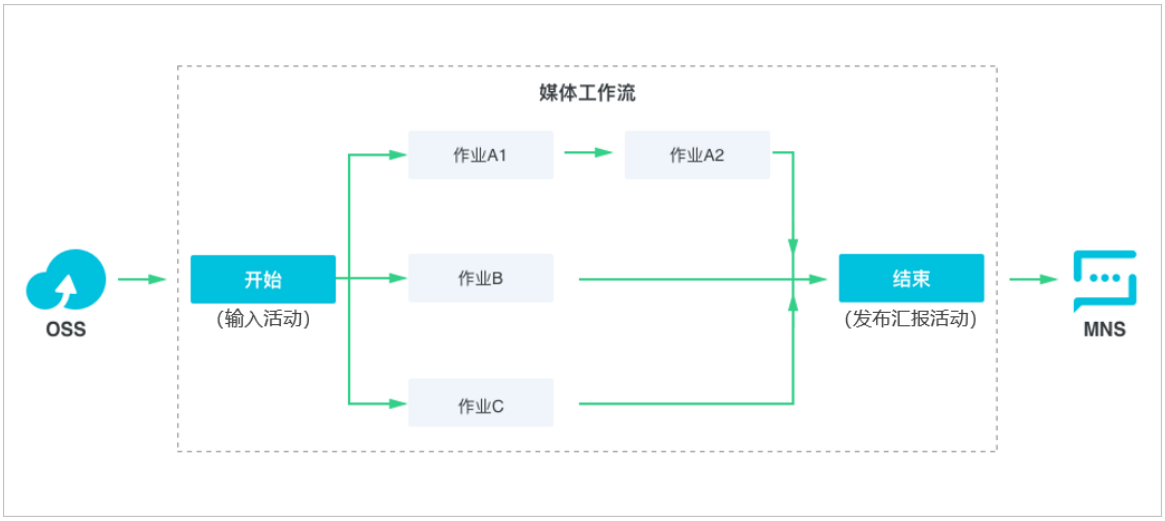
- 媒体库

  媒体库是所有媒体的集合，媒体是媒体库的最小管理单元。

- 媒体工作流

  媒体工作流是自动化生产媒体的工厂，由媒体工作流ID（MediaWorkflowId）唯一标识。

- 活动




工作流中的每个节点（即一个单任务操作）称为活动。根据实际需求，既可以并行执行（例如，示意图的作业A1、B、C之间），也可以串行执行（例如：示意图的作业A1、A2之间）。除开始的输入活动和结束的发布汇报活动外，活动也支持各种类型的作业（转码作业、截图作业等）。

> **说明** 不同地域功能支持情况不同，工作流支持的活动也不相同。功能支持情况请参见服务地域。

- 开始的输入活动

  配置工作流关联存储的触发路径，只要在对应的路径上传视频、音频多媒体文件，就会自动触发工作流执行。

- 结束的发布汇报活动

  工作流执行完成后，会向消息服务指定的队列或主题发送消息通知执行结果。执行结果包含了媒体ID和多媒体文件的绝对地址，这样就能对应具体是哪个多媒体文件执行完成。

- 作业活动

  作业支持的所有参数，都可以在作业活动中配置。

- 路径匹配规则

  当有新的媒体文件进入匹配路径时自动触发工作流。匹配使用路径前缀的规则，例如：上传的文件路径为*http://exampleBucket****.oss-cn-hangzhou.aliyuncs.com/A/B/C/video_01.flv*，配置的触发路径结果如下：

| OSS路径 | 是否匹配 |
| --- | --- |
| *http://exampleBucket****.oss-cn-hangzhou.aliyuncs.com/A/B/C/* | 是 |
| *http://exampleBucket****.oss-cn-hangzhou.aliyuncs.com/A/B/C2/* | 否 |
| *http://exampleBucket****.oss-cn-hangzhou.aliyuncs.com/A/B* | 是 |
| *http://exampleBucket****.oss-cn-hangzhou.aliyuncs.com/A/B2/* | 否 |
| *http://exampleBucket****.oss-cn-hangzhou.aliyuncs.com/A/* | 是 |
| *http://exampleBucket****.oss-cn-hangzhou.aliyuncs.com/A2/B/C/* | 否 |
| *http://exampleBucket****.oss-cn-hangzhou.aliyuncs.com/A/B/C/video* | 是 |
| *http://exampleBucket****.oss-cn-hangzhou.aliyuncs.com/A/B/C/video_01* | 否 |

- 扩展名匹配规则

  上传时的自动触发机制会检查文件的扩展名，避免产生一些无效的数据（例如pdf、word文档等）。

  > **说明** API手动触发机制不检查扩展名。

  文件没有扩展名（文件名中不包含扩展名分割符号”.”），或者扩展名符合下面的规则即为有效。

  - 视频

    .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

  - 音频

    .aac、.ac3、.acm、.amr、.ape、.caf、.flac、.m4a、.mp3、.ra、.wav、.wma、.aiff

- 工作流执行

  分为自动触发机制和API手动触发机制。

  - 自动触发机制：每次上传匹配的多媒体文件都会触发一次执行，同一个多媒体文件如果多次上传，则会触发多次执行，每次执行有唯一的RunId标识。
  - API手动触发机制：每次调用API都会触发一次执行，主要针对存储中的存量多媒体文件而设置。

- 用户数据

  每次执行时，可以填写自定义的用户数据参数（例如：商品ID），然后消息通知中会返回自定义的用户数据参数。这样使用自定义的用户数据就可以关联业务系统，不需要再在业务系统中记录媒体ID或者多媒体文件的绝对路径。

# 3.前置条件

## 3.1. 前置条件概述

在您使用媒体处理服务之前，需要进行一些设置确保所有功能能够正常使用。本文为您介绍在使用媒体处理之前需要进行的设置。

### 使用前准备

1. 开通服务。
   - 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。
   - 您已经开通媒体处理服务。具体操作，请参见开通服务。
2. 账号准备。

   > **说明** 阿里云账号AeecssKey是您访问阿里云API的密钥，具有该账户完全的权限，请您务必妥善保管！不要通过任何方式（如Github）将AccessKey公开到外部渠道，以避免被他人利用而造成安全威胁。强烈建议您遵循阿里云安全最佳实践，使用RAM用户（而不是阿里云账号）的AccessKey进行API调用。

   您已经准备好访问媒体处理服务使用的AccessKey。获取方法，请参见获取AccessKey。

   > **说明** 推荐您使用RAM用户AccessKey，为RAM用户授权请参见申请RAM用户权限。

3. （可选）如需使用媒体处理服务对指定网站上的业务进行加速，推荐您配置加速域名。配置加速域名可以对指定网站上的业务实现资源访问加速。具体操作，请参见配置加速域名。
4. （可选）如需接收任务状态回调通知，则需要在管道中配置消息服务MNS。具体操作，请参见设置转码消息通知。
5. 上传需要处理的音视频文件，具体操作，请参见上传视频。

## 3.2. 申请RAM用户权限

本文为您介绍通过阿里云账号授权RAM用户访问媒体处理服务的操作步骤。

通过阿里云账号登录RAM控制台，可以为RAM用户授予相关权限。授权不同的功能可以在控制台上使用不同的服务，以达到RAM用户在授权范围内使用阿里云其他产品服务的目的。RAM用户的权限主要包括：媒体处理MPS、对象存储OSS、内容分发CDN、消息服务MNS的产品使用权限，规划RAM用户拥有这些服务的资源实例后即可按相应的授权模板创建授权策略，并授予给RAM用户即可。

> **说明** 如果使用MPS服务时权限不足会导致报错 `User not authorized to operate on the specified resource` ，请检查您使用的RAM用户是否拥有MPS的完整控制权限，或按照下文内容申请RAM用户权限。

1. 创建子账户。
   i. 登录RAM控制台。
   ii. 在左侧导航栏，选择**身份管理 > 用户**，单击**创建用户**。

iii. 在**创建用户**页面的**用户账号信息**区域，输入**登录名称**和**显示名称**。

iv. 根据您的实际需要选择访问方式。

> **说明** 为了保障账号安全，建议仅为RAM用户选择一种登录方式，避免RAM用户离开组织后仍可以通过访问密钥访问阿里云资源。

| 访问方式 | 说明 |
| --- | --- |
| **控制台访问** | 设置控制台登录密码、重置密码策略和多因素认证策略。<br>**说明** 自定义登录密码时，密码必须满足密码复杂度规则。关于如何设置密码复杂度规则，请参见设置RAM用户密码强度。 |
| **OpenAPI调用访问** | 自动为RAM用户生成访问密钥（AccessKey），支持通过API或其他开发工具访问阿里云。 |

v. 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。

vi. 单击用户信息右侧的**复制**，保存用户登录名称、登录密码、AccessKey ID和AccessKey Secret等用户信息。

> **说明** 请保存用户信息并妥善保管，用于后续的访问。

2. 子账户授权。

i. 在左侧导航栏，选择**用户**进入用户界面，选择新建的RAM账户，并单击**操作**列**添加权限**。

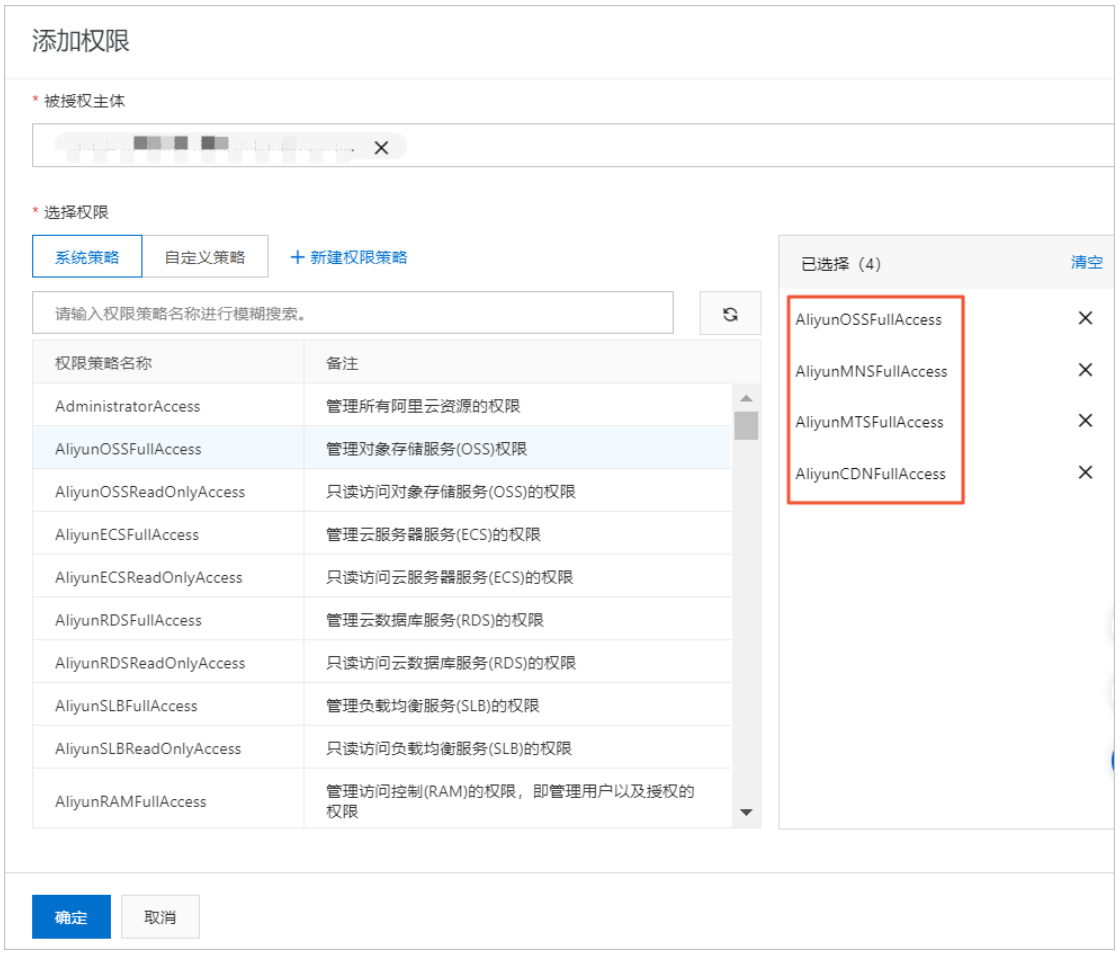
ii. 选择授权应用范围。

| 授权范围 | 说明 |
| --- | --- |
| **整个云账号** | 权限在当前阿里云账号内生效。 |
| **指定资源组** | 权限在指定资源组内生效。 |

iii. 在**授权主体**输入框中输入被授权主体。

> **说明** 被授权主体即需要授权的RAM用户，系统会自动填入当前的RAM用户，您也可以添加其他RAM用户。

iv. 选择AliyunOSSFullAccess，AliyunCDNFullAccess，AliyunMTSFullAccess，AliyunMNSFullAccess并单击**确定**，完成授权。




> **说明** 此处授权的是MPS服务中所需要的四个产品的全部权限，如果需要更细粒度的权限控制请参考下文中创建自定义授权策略进行授权。

3. （可选）创建自定义授权策略。

> **说明** 如果用户需要对各个产品进行更细权限的控制，可以创建自定义权限，将对应的自定义权限授权给子账户。

i. 在左侧导航栏，选择**权限管理 > 权限策略**。

ii. 单击**创建授权策略**，进入自定义权限页面。

iii. 设置权限策略。

> **说明** 授权策略可以参考下文中的OSS授权策略、CDN授权策略或MNS授权策略。

| 参数 | 说明 |
| --- | --- |
| **策略名称** | 在文本输入框中输入名称。 |
| **备注** | 对策略进行补充说明。 |
| **配置模式** | 选择脚本配置。 |
| **策略内容** | 选择已有策略，或将下方需要的策略代码复制到代码框中。 |

- OSS授权策略

  权限描述：

  - 对指定的输入、输出Bucket有所有操作权限。

- 查看Bucket列表权限。

```
{
"Version": "1",
"Statement": [
{
"Action": [
"oss:*"
],
"Resource": [
"acs:oss:*:*:$InputBucket",
"acs:oss:*:*:$InputBucket/*",
"acs:oss:*:*:$OutputBucket",
"acs:oss:*:*:$OutputBucket/*"
],
"Effect": "Allow"
},
{
"Action": [
"oss:ListBuckets"
],
"Resource": "*",
"Effect": "Allow"
}
]
}
```

**说明**

- $InputBucket：MPS媒体输入Bucket，填入具体工作流中要涉及到的Bucket名称。
- $OutputBucket：MPS媒体输出Bucket，填入具体工作流中要涉及到的Bucket名称。
- 这里的 `oss:ListBuckets` 权限是子账号需要通过可视化工具操作OSS必备的权限，该权限赋予后子账号登录可以查看到所有的Bucket列表，但是仅能操作前面赋权的$InputBucket和$OutputBucket这两个Bucket，并且该权限暂时仅支持赋权给所有的Bucket，不支持赋权给某个Bucket。

- MNS授权策略

  权限描述：

  - 对指定的Queue、Topic有所有权限。
  - 有查询Queue、Topic列表的权限。

```
{
"Version": "1",
"Statement": [
{
"Action": [
"mns:*"
],
"Resource": [
"acs:mns:$Region:$Uid:/queues/$QueueName",
"acs:mns:$Region:$Uid:/topics/$TopicName",
],
"Effect": "Allow"
},
{
"Action": [
"mns:Get*",
"mns:List*"
],
"Resource": "*",
"Effect": "Allow"
}
]
}
```

**说明**

- $QueueName：MNS队列名称，填入具体工作流中要涉及到的队列名称。
- $TopicName：MNS通知主题名称，填入具体工作流中要涉及到的通知名称。

- CDN授权策略

  权限描述：

  - 对指定的CDN加速域名有所有权限。
  - 有查询CDN加速域名的权限。

```
{
"Version": "1",
"Statement": [
{
"Action": "cdn:*",
"Resource": [
"acs:cdn:*:$Uid:domain/$DomainName"
],
"Effect": "Allow"
},
{
"Action": "cdn:Describe*",
"Resource": "*",
"Effect": "Allow"
}
]
}
```

> **说明** $DomainName：CDN加速域名名称，填入具体工作流中要涉及到的CDN域名名称。

- RAM授权策略

  权限描述：

  使被授权的子用户有权限查看角色的权限策略。

```
{
    "Statement": [{
        "Action": ["ram:ListPoliciesForRole"],
        "Effect": "Allow",
        "Resource": "*"
    }],
    "Version": "1"
}
```

iv. 单击**确定**完成自定义授权策略创建。

4. RAM用户登录MPS控制台。

   i. 在云账号控制台左侧导航栏，选择**概览**，单击**用户登录地址**。

ii. 输入RAM**用户名**和**用户密码**，单击**登录**，进入阿里云控制台。

> **说明** 子账户初次登录需要重置密码。

iii. 在**保有资源的云产品**下方，单击**媒体处理**，进入媒体处理控制台。

授权完成后，即可使用RAM用户访问媒体处理服务。

# 3.3. 配置加速域名

媒体处理服务支持分发加速功能，可以对指定网站上的业务实现资源访问加速。如果您有加速需求，建议您将需要加速的网站作为源站，为其创建加速域名。媒体处理添加加速域名需要在CDN中进行配置。本文为您介绍为媒体处理服务添加加速域名的方法。

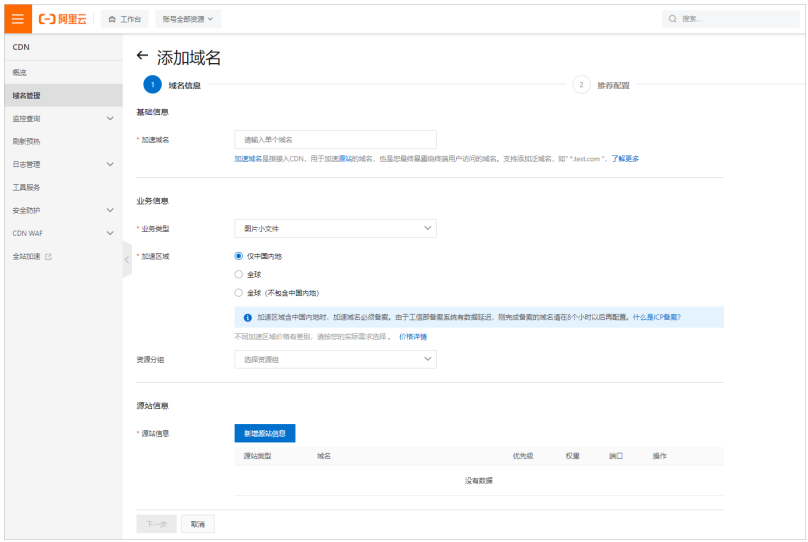
## 前提条件

您需要开通CDN服务。如未开通，请先开通CDN服务。详细步骤请参见开通CDN服务。

## 配置步骤

1. 登录CDN管理控制台。
2. 在左侧导航栏，选择**域名管理**，单击**添加域名**。

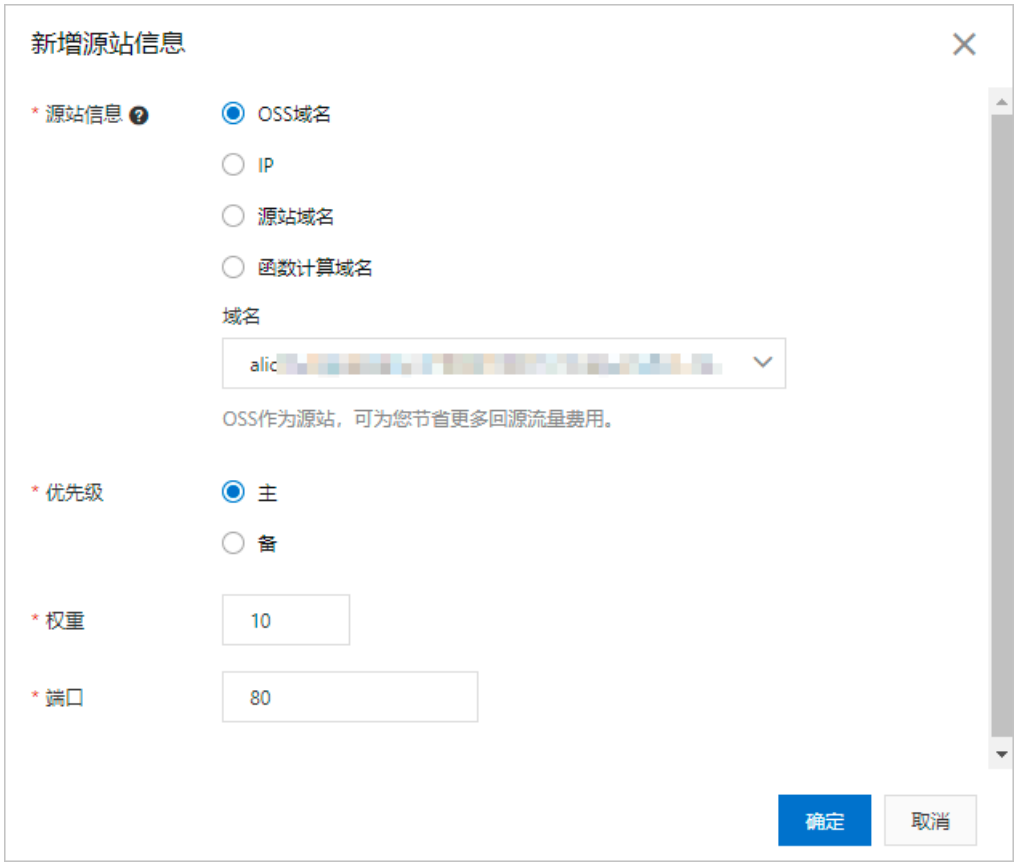
3. 配置域名信息。
   i. 配置**加速域名**、**业务类型**、**加速区域**、**资源分组**。




| 参数 | 说明 |
| --- | --- |
| **加速域名** | ■ **域名类型**：子域名（例如，`example.aliyundoc.com`）或泛域名（例如，`*.aliyundoc.com`）。<br>■ **格式**：<br>■ 支持全英文小写的域名，例如example.com，不支持含大写英文字母的域名。<br>■ 中文域名：例如：阿里云.网址，请以中文形式进行相关备案，再通过第三方工具punnycode将中文域名转换成为英文域名（例如：xn--fiq****.xn--eq****）后填入。<br>■ **泛域名要求**：<br>■ CDN支持泛域名加速，关于泛域名的层级限制，请参见CDN支持泛域名加速吗？。<br>■ 泛域名和子域名必须在同一个账号下，否则添加域名时系统会报错。如果您无法自行解决，请提交工单处理。<br>■ 如果泛域名未被添加到任何CDN账号下，则支持在多个账号下添加不同的子域名。<br>■ 单个泛域名最多匹配500个精确域名，超过500个后新增的精确域名无法同步获取泛域名配置，即无法使用CDN服务。<br>**说明** 未超出500个之前匹配到的精确域名使用不受影响。<br>■ **加速域名不允许重复添加。**<br>如果出现域名已被添加到其他云产品（例如视频点播、全站加速等）的提示，您可以提交工单。 |

| 参数 | 说明 |
| --- | --- |
| | ■ **每个阿里云账号最多可以添加50个加速域名。**<br>**说明** 如果您域名的总带宽日均峰值大于50 Mbps，且业务无风险，可提交工单申请增加域名个数。<br>■ **加速内容必须合法且符合CDN业务规范。**详细信息，请参见使用限制关于内容审核部分。<br>■ **字符串长度**：不超过67个字符。<br>■ **备案**：如果您的加速区域为**全球**或**仅中国内地**，无论源站在哪里域名都必须备案，推荐您进入阿里云ICP代备案管理系统进行备案。<br>■ **回收规则**：如果您的域名处于停用状态超过120天（包含审核未通过状态），阿里云CDN会自动删除该域名的相关记录。如果您需要继续使用，需前往阿里云CDN控制台重新添加域名。<br>■ **下线（offline）规则**：详细信息，请参见关于域名下线（Offline）规则调整的公告。 |
| **业务类型** | 在下拉列表中，选择**视音频点播**。 |
| **加速区域** | ■ **仅中国内地**：全球用户访问均会调度至中国内地加速节点进行服务（海外用户的访问流量将会被调度至华东电信的CDN节点）。<br>■ **全球**：全球用户访问将会择优调度至最近的加速节点进行服务。<br>■ **全球（不包含中国内地）**：全球用户访问均会调度至中国香港、中国澳门、中国台湾以及其他国家和地区的加速节点进行服务（中国内地用户将会被调度至日本、新加坡和中国香港的CDN节点）。 |
| **资源分组** | 在下拉列表中，选择加速域名所属的资源组。<br>**说明** 新开通CDN服务的用户，不显示资源分组。如果您需要使用资源组功能，则需要登录资源管理控制台，根据界面提示开通资源组服务，并创建资源组，操作方法请参见创建资源组。 |

ii. 单击**新增源站信息**配置源站信息。




| 参数 | 描述 |
| --- | --- |
| **源站信息** | 选择**OSS域名**，可以手动输入阿里云OSS Bucket的外网域名，如：example-bucket-****.oss-cn-hangzhou.aliyuncs.com。OSS外网域名可前往OSS控制台查看，也可直接选择同账号下的OSS Bucket。不支持OSS内网域名。 |
| **优先级** | 支持设置主备，主优先级大于备优先级。用户请求通过阿里云CDN回源时，会优先回源到优先级为主的源站地址。<br><br>例如，有A、B两个源站，A源站的优先级为主，B源站的优先级为备，则用户请求通过阿里云CDN回源时会优先回源到A源站，如果A源站出现故障，将会回源到B源站，当A源站恢复正常后会从B源站切换回A源站。 |

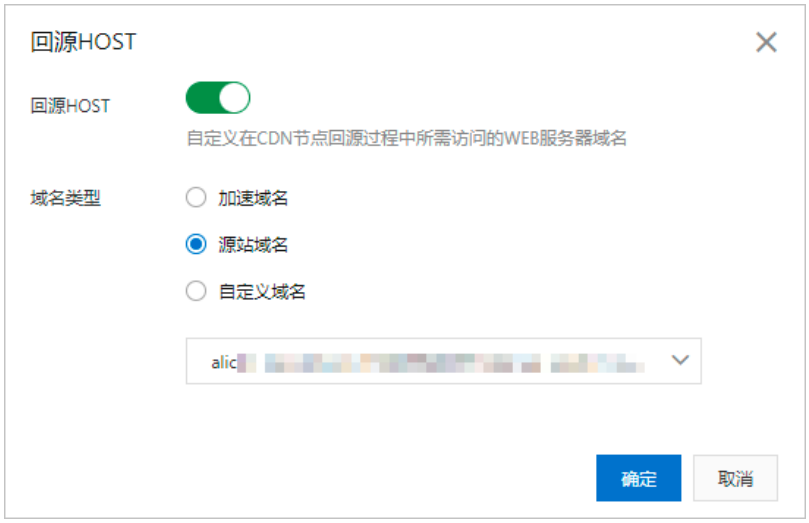
| 参数 | 描述 |
| --- | --- |
| **权重** | 当多个源站的优先级相同时，阿里云CDN会按照源站的权重分配用户请求回源到不同源站的比例，实现按权重的负载均衡。您可以根据业务需求，自行设置权限值。<br>■ 取值范围：1~100，数值越大，源站分配到的用户请求比例越高。<br>■ 默认值：10。<br>示例：有A、B两个源站，两个源站的优先级都是主，A源站的权重为80，B源站的权重为20，则用户请求将会按照8:2的比例在A、B两个源站之间分配。 |
| **端口** | CDN节点回到源站哪个端口请求资源。默认为80，根据您源站的支持情况，可自定义设置回源端口，允许设置的端口范围为1~65535。<br>■ 默认值：80。<br>■ 端口值为443时，以HTTPS协议回源；80或其他自定义端口，以HTTP协议回源。<br>说明<br>■ 如果需要以HTTPS协议回源到其他自定义端口，请提交工单申请。<br>■ 如果配置了**回源协议**功能（默认为关闭状态），这里配置的端口会失效。关闭回源协议的方法，请参见配置回源协议。<br>■ 当源站选择OSS域名时，回源端口是否支持自定义端口，取决于OSS产品。 |

iii. 单击**确定**，完成配置。

4. 单击**下一步**。您首次在CDN控制台添加一个新域名时，需要完成域名归属权验证。具体操作，请参见验证域名归属权。如果您之前已经验证通过，请忽略该步骤。
5. 等待人工审核。审核时间预计需要1~2个工作日，如果您的源站是阿里云ECS或OSS，则审核速度会加快，您也可以提交工单加急审核。加速域名审核通过后，您可以在**域名管理**页面查看，状态显示为**正常运行**，即为添加成功。
6. 配置CNAME。加速域名添加成功后，阿里云CDN会为您分配对应的CNAME地址，您需要完成CNAME配置，CDN加速服务才会生效。具体操作，请参见配置CNAME。
7. 设置回源HOST。
   i. 在左侧导航栏，单击**域名管理**。
   ii. 在域名管理页面列表中选择需要配置的域名，在**操作**列单击**管理**。
   iii. 在域名的左侧导航栏，单击**回源配置**。
   iv. 在**回源HOST**区域，单击**修改配置**。

v. 单击**回源HOST**开关，在**域名类型**中选择**源站域名**。

> **说明** 源站域名是指您的源站服务器的域名地址，即CDN回源需要访问的域名地址，例如：`example.com`。当回源HOST的域名类型选择为源站域名的时候，回源HOST将会被配置为源站域名，例如：`example.com`。




vi. 单击**确定**，完成回源HOST配置。

8. 打开拖拽播放功能。
   i. 在左侧导航栏，单击**域名管理**。
   ii. 在域名管理页面列表中选择需要配置的域名，在**操作**列单击**管理**。
   iii. 在域名的左侧导航栏，单击**视频相关**。
   iv. 在**拖拽播放**区域，单击**拖拽播放**开关。

> **说明** 打开此开关后，CDN将支持阿里云Web播放器对MP4、FLV格式文件的拖拽播放。M3U8格式文件的播放，无需开启此功能即可支持拖拽播放。

媒体处理加速域名配置完成后，后续在访问已添加加速域名网站上的业务时可以实现加速访问的效果。

# 4.消息通知

## 4.1. 设置转码消息通知

媒体处理支持消息服务MNS的消息队列和主题通知功能，如需使用转码消息通知功能，可根据您的实际需要，启用消息队列或主题通知功能。

### 设置主题消息通知

1. 创建主题消息通知。
   i. 开通消息服务MNS并授权。详细步骤请参见开通消息服务MNS并授权。
   ii. 单击**管理控制台**进入消息服务控制台。

   iii. 在左侧导航栏选择**主题列表**，在主题列表页面单击**创建主题**。
   iv. 在创建主题页面，填写如下参数，完成配置。

| 参数 | 说明 |
| --- | --- |
| **名称** | 主题名称。<br>**说明** 主题名称必须以英文字母开头，剩余名称可以是英文、数字、短划线（-），长度不超过120个字符。 |
| **消息最大长度** | 允许发送到该主题的消息体的最大长度。取值：1024~65536，默认为65536，单位为Byte。 |
| **启用日志功能** | 在选项中选择是或否。<br>**说明** 开启日志功能后，消息服务MNS将自动推送该主题的操作日志到指定的LoggingBucket中。然后您可以通过日志查看消息轨迹、消息延迟以及其他信息。更多信息，请参见日志管理概述。 |

   v. 单击**确定**，完成创建主题，跳转至主题详情页面。
   vi. 单击**创建订阅**。

vii. 在创建订阅页面，填写如下参数，完成配置。

| 参数 | 说明 |
| --- | --- |
| **名称** | 订阅名称。<br>**说明** 名称必须以英文字母开头，剩余名称可以包含英文字母、数字、短划线（-），名称不超过255个字符。 |
| **推送类型** | 在下拉列表中选择推送类型，默认为**HTTP**。 |
| **接收端地址** | 消息的**接收端地址**。<br>**说明** HTTP地址，必须以http://或者https://开头。 |
| **（可选）消息过滤标签** | 通过设置的标签对消息进行过滤。<br>**说明** 用于消息过滤，不超过16个字符。 |
| **重试策略** | 向接收端推送消息出现错误时的重试策略。<br>■ **退避重试**：重试3次，间隔时间为10~20秒之间的一个随机值。<br>■ **指数衰减重试**：重试176次，总计重试时间为1天，间隔时间单位为秒，依次为：2^0，2^1，...，512，512，...，512。 |
| **消息推送格式** | 推送给用户的消息格式。<br>■ **SIMPLIFIED**：消息体即用户发布的消息，不包含任何属性信息。<br>■ **JSON**：消息体为JSON格式，包含消息正文和消息属性。<br>■ **XML**：消息体为XML格式，包含消息正文和消息属性。 |

viii. 单击**确定**，完成创建订阅。

2. 创建完成后，可按需绑定主题通知与转码管道。具体步骤，请参见绑定管道。

> **说明** 如果您使用工作流进行转码，则无需绑定管道。媒体处理中通过工作流触发执行的作业，忽略转码管道上绑定的消息机制。如需在工作流中使用消息通知，创建工作流时**输入**节点选择所需消息通知的主题即可。具体操作，请参见创建工作流。

3. 创建转码任务时，选择对应的管道即可接收消息通知。

## 设置队列消息通知

1. 创建队列。
   i. 开通消息服务MNS并授权。详细步骤请参见开通消息服务MNS并授权。

ii. 单击**管理控制台**进入消息服务控制台。

iii. 在左侧导航栏，单击**队列列表**。

iv. 在**队列列表**页面，单击**创建队列**。

v. 在**创建队列**面板配置以下参数。

| 参数 | 描述 |
| --- | --- |
| **名称** | 队列名称。 |
| **消息最大长度** | 发送到队列的消息体的最大长度。 |
| **长轮询时间** | 当队列中没有消息时，该队列的ReceiveMessage请求的最大等待时长。 |
| **消息可见性超时时间** | 消息从队列中取出后从Active状态变成Inactive状态后的持续时间。 |
| **消息保存时长** | 消息在队列中的最长存活时间。从发送到队列开始经过此参数指定的时间后，不论消息是否被取出都将被删除。 |
| **消息定时时间** | 发送到队列的所有消息将延后此参数指定的时间后被消费。 |
| **启用日志功能** | 是否开启日志管理功能。 |

vi. 配置完成后单击**确定**完成队列创建。

2. 创建完成后，可按需绑定队列与转码管道。具体步骤，请参见绑定管道。

> **说明** 如果您使用工作流进行转码，则无需绑定管道。媒体处理中通过工作流触发执行的作业，忽略转码管道上绑定的消息机制。如需在工作流中使用消息通知，创建工作流时**输入**节点选择所需消息通知的队列即可。具体操作，请参见创建工作流。

3. 创建转码任务时，选择对应的管道即可接收消息通知。

# 4.2. 绑定管道

创建完主题通知或队列后，需要绑定主题通知（队列）与转码管道。本文为您介绍主题通知、队列通知绑定转码管道的步骤。

## 背景信息

如果您需要使用普通转码管道进行转码操作并接收消息通知，则需要为您在设置转码消息通知设置的主题通知或队列绑定转码管道，在转码时选择绑定了主题通知（队列）的管道即可正常接收消息。如果您使用工作流进行转码操作并接收消息通知，则无需为管道绑定转码消息通知，只需在设置工作流消息通知时指定您所需的主题通知或队列即可。

## 绑定转码管道与主题通知

1. 登录媒体处理控制台。
2. 在顶部菜单栏左侧选择**地域**。

3. 在左侧导航栏中，选择**全局设置 > 管道**。
4. 在管道列表中，选择需要开启消息通知的管道，单击**设置消息通知**。

5. 在**设置消息通知**页面，打开消息通知开关，**消息类型**选择**通知**，并选择所需的通知。

| 参数 | 说明 |
| --- | --- |
| **消息通知** | 打开消息通知开关。 |
| **消息类型** | 支持队列和主题通知两种消息类型。 |
| **通知名称** | 在通知名称下拉列表中选择所需的通知。 |

6. 单击**确定**，完成主题通知绑定。

## 绑定转码管道与队列

1. 登录媒体处理控制台。
2. 在顶部菜单栏左侧选择**地域**。

3. 在左侧导航栏中，选择**全局设置 > 管道**。
4. 在管道列表中，选择需要开启消息通知的管道，单击**设置消息通知**。

5. 在设置消息通知页面，打开消息通知开关，**消息类型**选择**队列**，并在下拉列表中选择所需的队列。

| 参数 | 说明 |
| --- | --- |
| **消息通知** | 打开消息通知开关。 |
| **消息类型** | 支持队列和主题通知两种消息类型。 |
| **队列名称** | 在队列名称下拉列表中选择所需的队列。 |

6. 单击**确定**，完成队列绑定。

## 参数说明

下表为转码消息参数说明。

| 参数名称 | 说明 |
| --- | --- |
| jobId | 作业ID。 |

| 参数名称 | 说明 |
| --- | --- |
| type | 作业类型。值为：<br>• Transcode：转码。<br>• Analysis：智能模板分析。<br>• Snapshot：截图。<br>• MediaInfo：媒体信息。 |
| state | 作业状态。值为：<br>• Success：成功。<br>• Fail：失败。 |
| code | 错误码。 |
| msg | 错误消息详情。 |

# 4.3. 设置工作流消息通知

开启工作流消息通知，媒体工作流开始执行和完成执行时，会向指定的队列或主题发送消息通知。

## 操作步骤

1. 根据实际需要创建队列或主题通知。具体操作请参见设置转码消息通知。

   > **说明** 目前媒体工作流触发执行的作业，忽略转码管道上绑定的消息机制，以创建工作流时实际绑定的主题通知（消息队列）为准。

2. 创建工作流，其中输入节点参数需按下表设置，其余节点具体操作请参见创建工作流。

| 参数 | 描述 |
| --- | --- |
| **输入Bucket** | 选择输入路径后自动显示。 |
| **输入路径** | 待处理文件的输入路径。 |
| **转码管道** | 选择在上一步中已经绑定的转码管道。 |
| **消息通知** | 单击开启消息通知。 |
| **消息类型** | 需要接收的消息类型。根据您管道绑定的消息类型选择。 |
| **队列（通知）名称** | 选择对应的队列（主题通知）。 |

## 消息格式

媒体工作流消息通知的消息体是JSON格式，详细的字段名称、类型、描述请参见新增媒体中媒体工作流消息部分。

结构的层次定义如下：

- 开始

  `活动基本属性` 中 `活动类型` 是 `Start` 。

- 结构
  - 顶层

    JSON对象。定义： `{当前活动的基本属性,工作流执行对象}`

    - `当前活动的基本属性`

      当前活动的基本属性不是一个独立的对象，是直接属于顶层的键值属性，可以参考下面的示例。定义： `{工作流执行ID,活动名字活动类型,活动状态,错误信息}`

    - `工作流执行详情对象`

      JSON对象。定义： `{工作流执行ID,媒体工作流ID,媒体工作流名字,媒体ID,输入文件,工作流执行类型,活动对象数组,创建时间}`

  - `活动对象数组`

    JSON数组，包含执行到当前状态的所有活动。例如：开始消息中只有一个Start活动对象，完成消息则包含所有活动对象。定义： `[活动对象1,活动对象2...]`

    - `活动对象1`

      JSON对象。定义： `{活动名字,活动类型,作业ID,活动状态,开始时间,结束时间,错误信息}`

    - `活动对象2`

      结构同 `活动对象1` 。

- 完成

  `活动基本属性` 中 `活动类型` 是 `Report` 。

- 示例

```
{
      "RunId": "8f8aba5a62ab4127ae2add18da20****",
      "Name": "Act-4",
      "Type": "Report",
      "State": "Success",
      "MediaWorkflowExecution": {
        "Name": "ConcurrentSuccess",
        "RunId": "8f8aba5a62ab4127ae2add18da20****",
        "Input": {
            "InputFile": {
                "Bucket": "exampleBucket****",
                "Location": "oss-test",
               "Object": "mediaWorkflow/ConcurrentSuccess/01.wmv",
            }
        },
        "State": "Success",
        "MediaId": "2be491ab4cb6499cd0befe5fcf0c****",
        "ActivityList":
[
{
                "RunId": "8f8aba5a62ab4127ae2add18da20****",
                "Name": "Act-1",
```

```
                "Type": "Start",
                "State": "Success",
                "StartTime": "2016-03-15T02: 53: 41Z",
                "EndTime": "2016-03-15T02: 53: 41Z",
            },
            {
                "RunId": "8f8aba5a62ab4127ae2add18da20****",
                "Name": "Act-2",
                "Type": "Transcode",
                "JobId": "f34b6d1429dd491faa7a6c1c8f90****",
                "State": "Success",
                "StartTime": "2016-03-15T02: 53: 43Z",
                "EndTime": "2016-03-15T02: 53: 47Z",
            },
            {
                "RunId": "8f8aba5a62ab4127ae2add18da20****",
                "Name": "Act-3",
                "Type": "Snapshot",
                "JobId": "c14150be33304825a5d67cd5364c****",
                "State": "Success",
                "StartTime": "2016-03-15T02: 53: 44Z",
                "EndTime": "2016-03-15T02: 53: 45Z",
            },
            {
                "RunId": "8f8aba5a62ab4127ae2add18da20****",
                "Name": "Act-4",
                "Type": "Report",
                "State": "Success",
                "StartTime": "2016-03-15T02: 53: 49Z",
                "EndTime": "2016-03-15T02: 53: 49Z",
            }
        ],
        "CreationTime": "2016-03-15T02: 53: 39Z",
    }
}
```


# 4.4. 接收消息通知

通知发布成功后，您可以在消息服务MNS（Message Service）中通过队列方式或主题通知方式接收消息通知。本文为您介绍如何接收消息通知。

## 前提条件

您已设置消息通知。具体操作，请参见设置转码消息通知或设置工作流消息通知。

## 队列方式接收消息通知

- 使用Java SDK接收消息通知。

> **说明** 示例语言采用Java，其他语言使用说明，请参见SDK使用手册。

- 使用控制台接收消息通知。

## 主题通知方式接收消息通知

- 使用Java SDK接收消息通知。

> **说明** 示例语言采用Java，其他语言使用说明，请参见SDK使用手册。

- 使用控制台接收消息通知。

# 5.控制台用户路径

## 5.1. 媒资转码

### 5.1.1. 转码概述

阿里云媒体处理为您提供媒资转码功能，可以将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。本文为您介绍转码的功能详情与对比、转码前的准备以及转码的具体操作步骤。

#### 功能介绍

> **注意** 使用功能前，请先参照转码准备完成准备工作（比如：开通服务、开启管道等），否则可能导致转码失败。

视频上传完成后，进入转码处理阶段。您可以根据实际业务需要，选择使用系统预置模板或自定义模板，对媒资文件进行普通转码、窄带高清TM1.0转码、窄带高清TM2.0转码、倍速转码等操作。

- 普通转码：根据需求选择不同的封装格式和清晰度对视频进行转码播放，您可对不同清晰度的码率、分辨率、格式等进行自定义设置。
- 窄带高清TM1.0转码：窄带高清TM1.0是一项基于阿里云转码技术的媒体处理功能。窄带高清TM1.0在转码过程中对视频中的场景、动作、内容、纹理等进行智能分析，以相对较低的码率输出相同画质的视频，从而在一定程度上降低带宽成本。
- 窄带高清TM2.0转码：窄带高清TM2.0是一项基于阿里云转码技术的媒体处理功能。从人眼视觉模型出发，将编码器的优化目标从“保真度最高”调整为“主观体验最好”，在节省码率的同时，也能提供更加清晰的观看体验。
- 倍速转码：倍速转码通过对视频分片并进行转码，可将转码速度提升5倍，适用于30分钟以上的长视频。

#### 功能对比

| 对比项 | 窄带高清TM1.0转码 | 窄带高清TM2.0转码 | 倍速转码 | 分辨率倍增转码 |
| --- | --- | --- | --- | --- |
| 核心技术 | 通过传统算法做画质增强，其本质就是节省不必要的码率输出。 | 通过AI智能算法，基于主观感受最好为标准进行优化。 | 倍速转码通过对视频分片并进行转码，可将转码速度提升5倍。 | 业界独有的画质重生技术，将现存普通或受损的影视内容重制为超高清或画质修复的版本。 |

| 对比项 | 窄带高清TM1.0转码 | 窄带高清TM2.0转码 | 倍速转码 | 分辨率倍增转码 |
| --- | --- | --- | --- | --- |
| 适用场景 | • 原始画质较好时。<br>• 码流和成本均比较敏感时。<br>说明 转码时需保证源文件时间戳连续，否则可能会导致转码失败、音画不同步等问题。对于此种情况阿里云暂无处理方案，需用户自行处理源文件。 | 原始画质较差时，相对更适用。<br>说明 窄带高清TM2.0可以针对不同局部做自适应优化。 | 适用于时长大于30分钟的视频。 | • 标清版本转高清版本。<br>• 2K节目源转4K节目源。 |
| 码流节省（同等码率） | 窄带高清TM1.0较普通转码节省约20%码流，但具体也会和用户原始设置的初始码流有一定关系。 | 窄带高清TM2.0的码流节省可以较窄带高清TM1.0再多5~10%。 | 不涉及。 | 不涉及。 |

## 转码准备

在进行转码任务之前，需要进行一些设置确保功能能够正常使用，如果已设置请忽略。

- 参考前置条件完成所需设置。
- 媒体处理操作的媒体资源均存放在OSS的Bucket中，需要添加媒体Bucket才可以在任务、工作流中选择对应的Bucket作为输入和输出路径。具体操作，请参见添加媒体Bucket。
- 开启管道。如果管道列表中没有您需要的管道类型，请先提交工单开通所需类型的管道。
- （可选）如预置转码模板不能满足您的实际转码需要，您可以自定义转码模板。具体操作，请参见创建自定义模板。
- （可选）如您有添加水印的需求，请先创建水印模板。具体操作，请参见添加水印模板。
- （可选）如您需要在工作完成后接收消息通知，请先配置消息通知服务。具体操作，请参见设置转码消息通知。

## 转码操作

- 创建转码任务
- 创建倍速转码任务
- 创建窄带高清TM1.0转码任务
- 创建窄带高清TM2.0转码任务
- 创建分辨率倍增转码任务

# 5.1.2. 创建倍速转码任务

倍速转码通过对视频分片并进行转码，可将转码速度提升5倍，适用于30分钟以上的长视频。本文为您介绍控制台倍速转码任务的创建步骤、使用以及使用中的常见问题。

## 使用方法

倍速转码可以通过控制台创建转码任务或工作流触发，使用前需要先开启倍速转码管道。

> **说明** 如果您需要通过API触发倍速转码任务，请参考媒资转码进行操作。

## 转码准备

在进行转码任务之前，需要进行一些设置确保功能能够正常使用，如果已设置请忽略。

- 参考前置条件完成所需设置。
- 媒体处理操作的媒体资源均存放在OSS的Bucket中，需要添加媒体Bucket才可以在任务、工作流中选择对应的Bucket作为输入和输出路径。具体操作，请参见添加媒体Bucket。
- 开启管道。如果管道列表中没有您需要的管道类型，请先提交工单开通所需类型的管道。
- （可选）如预置转码模板不能满足您的实际转码需要，您可以自定义转码模板。具体操作，请参见创建自定义模板。
- （可选）如您有添加水印的需求，请先创建水印模板。具体操作，请参见添加水印模板。
- （可选）如您需要在工作完成后接收消息通知，请先配置消息通知服务。具体操作，请参见设置转码消息通知。

## 配置倍速转码管道

1. 登录媒体处理控制台。
2. 在顶部菜单栏左侧选择地域。

3. 在顶部菜单栏左侧选择**全局设置 > 管道**。
4. 在倍速转码管道的**操作**列单击**启用管道**。
5. 输入**管道名称**，单击**确定**，完成倍速转码管道开启。

## 单任务倍速转码

1. 登录媒体处理控制台。
2. 在顶部菜单栏左侧选择地域。

3. 在左侧导航栏，选择**任务管理**。
4. 在**转码**页签下，单击**创建任务**。
5. 设置参数，在转码管道下拉列表中选择您配置好的倍速转码管道，单击**下一步**。

> **说明** 转码任务具体参数描述请参见创建转码任务。

6. 根据您的需求，自行选择转码模板、输出格式。转码模板详细信息请参见预置模板详情。
7. 单击**提交任务**，完成创建提交倍速转码任务。
   任务提交后，可通过**任务管理**页面**转码**页签找到任务ID和文件名，在对应ID的**操作**列单击详情，可查看任务进度。

## 工作流倍速转码

1. 登录媒体处理控制台。
2. 在顶部菜单栏左侧选择地域。

3. 创建工作流，具体操作请参见创建工作流。其中，输入节点的设置需按如下步骤设置。
   i. 设置**输入路径**节点。
   ii. 选择您配置好的倍速转码管道，单击**确定**。

   按照创建工作流继续配置完**转码**、**发布**节点保存之后，完成倍速转码工作流配置。

## 倍速转码作业用时

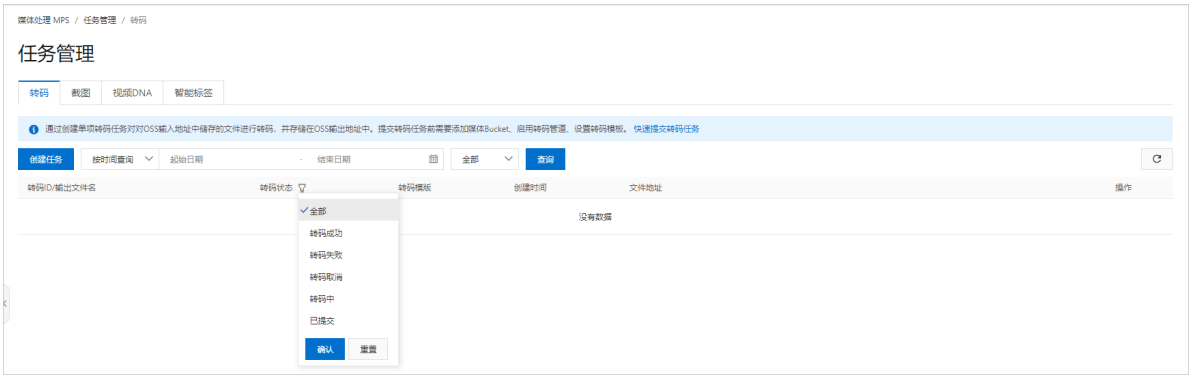
理论上，倍速转码作业用时是同分辨率普通转码作业用时的1/5，但是因为媒体文件在进行倍速转码时需要做分片并行处理，所以实际完整用时会略大于理论值。此外，如果同一时间收到大量并行作业的请求，管道将会对作业进行排序处理。

> **说明** 分片并行处理指需要对原始文件分析后分割切片，并在分片转码后合并成一个文件。

## 常见问题

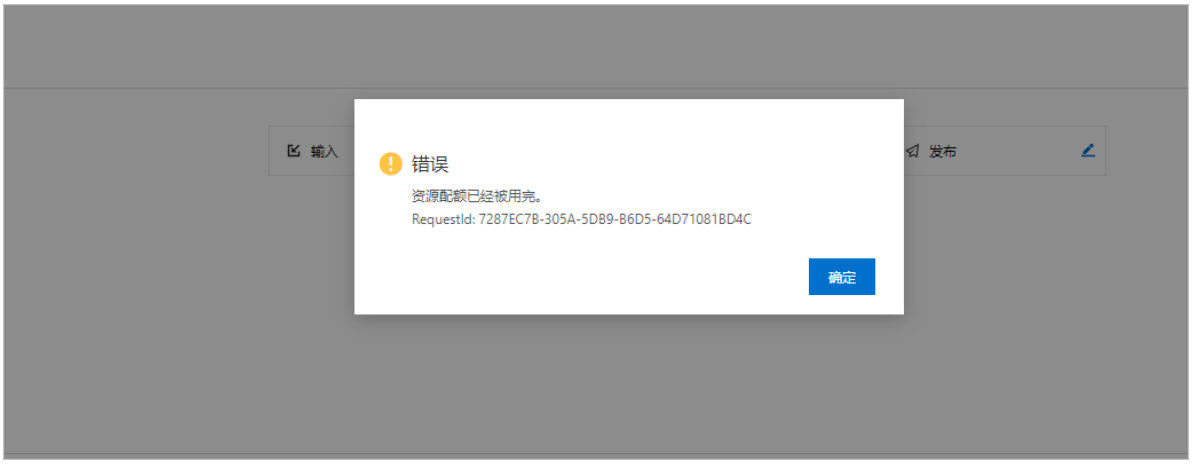
- 任务提交后找不到对应任务。

  可能是任务失败导致的，可在转码页签下筛选转码状态，查看转码失败的任务。




- 创建工作流时提示资源配额已被用完。

  每个用户在每个地域最多可创建10个媒体工作流，超过则会提示资源配额已经被用完。可以通过提交工单申请提升限额。




# 5.1.3. 创建窄带高清TM1.0转码任务

窄带高清TM1.0是一项基于阿里云转码技术的媒体处理功能。窄带高清TM1.0在转码过程中对视频中的场景、动作、内容、纹理等进行智能分析，以相对较低的码率输出相同画质的视频，从而在一定程度上降低带宽成本。

## 使用方法

窄带高清TM1.0转码可以通过创建转码任务或工作流触发，您可以使用预置窄带高清TM模板或根据您的实际转码需要自定义窄带高清TM1.0模板。

> **说明** 如您需要通过API触发窄带高清TM1.0转码任务，请参考媒资转码进行操作。

## 转码准备

在进行转码任务之前，需要进行一些设置确保功能能够正常使用，如果已设置请忽略。

- 参考前置条件完成所需设置。
- 媒体处理操作的媒体资源均存放在OSS的Bucket中，需要添加媒体Bucket才可以在任务、工作流中选择对应的Bucket作为输入和输出路径。具体操作，请参见添加媒体Bucket。
- 开启管道。如果管道列表中没有您需要的管道类型，请先提交工单开通所需类型的管道。
- （可选）如预置转码模板不能满足您的实际转码需要，您可以自定义转码模板。具体操作，请参见创建自定义模板。
- （可选）如您有添加水印的需求，请先创建水印模板。具体操作，请参见添加水印模板。
- （可选）如您需要在工作完成后接收消息通知，请先配置消息通知服务。具体操作，请参见设置转码消息通知。

## （可选）自定义窄带高清TM1.0模板

> **说明** 当预置窄带高清模板不能满足您的实际转码需求时，可以根据实际需要进行自定义模板。自定义模板开通后，可以在控制台创建单任务转码、工作流创建转码任务、API创建转码任务时使用。

1. 登录媒体处理控制台。
2. 在顶部菜单栏左侧选择地域。

3. 在左侧导航栏中，单击**全局设置 > 转码模板**。
4. 单击**添加转码模板**，添加**转码模板名称**。
5. 在**基本参数**区域的**封装格式**下拉框中选择mp4或m3u8。其他参数根据您的转码需要进行配置。参数详情，请参见转码设置。配置完成后单击**下一步**。
6. 根据实际需求配置**音频参数**，单击**下一步**。
7. 根据实际需求配置**高级参数**，单击**提交**。
8. 在**转码模板**页面，获取配置成功的转码模板ID。

> **说明** 在API触发窄带高清1.0转码时如果您使用自定义模板，则填入此步骤获取的转码模板ID即可。

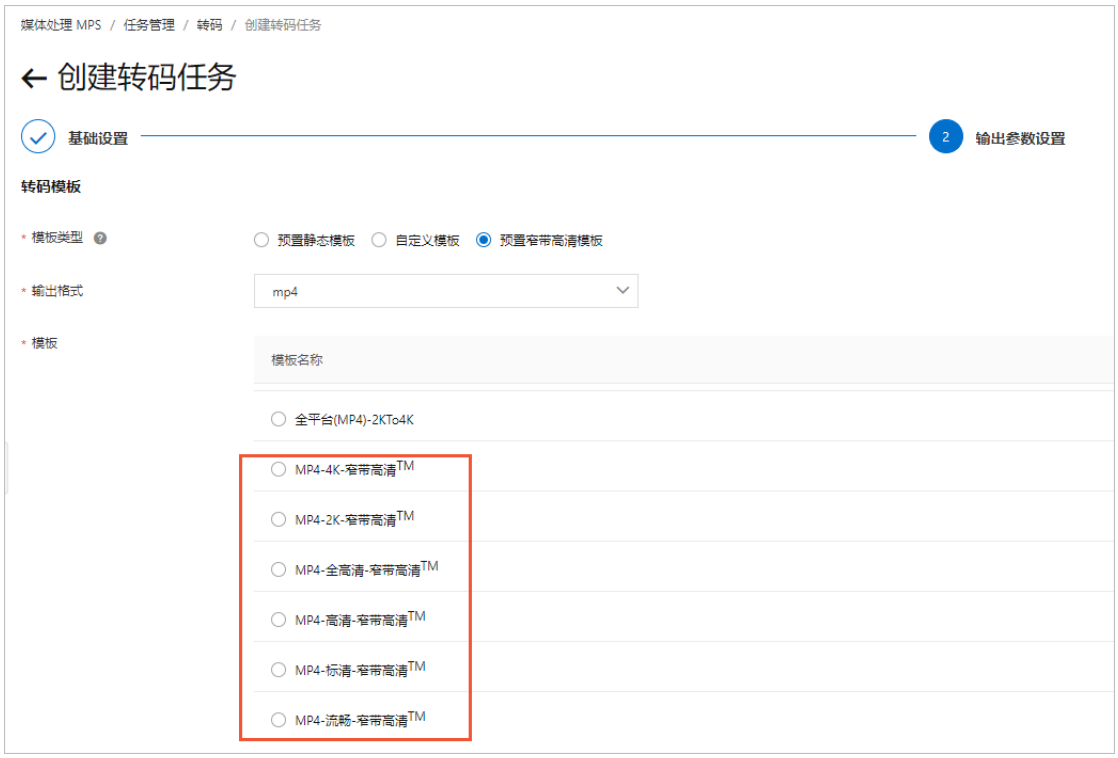
9. 提交工单并提供转码模板ID、UID、开通区域等信息开通窄带高清TM1.0模板。
   开通成功后，模板会升级为窄带高清TM1.0模板。例如：配置的普通模板的清晰度为标清，开通成功后，模板的清晰度成为标清-窄带高清TM1.0，其他配置参数设置不变。

## 创建转码任务触发窄带高清TM1.0转码

1. 登录媒体处理控制台。
2. 在顶部菜单栏左侧选择地域。

3. 添加媒体Bucket。如媒体Bucket已添加，则直接进入下一步。
4. 创建转码任务，具体操作请参见创建转码任务。其中转码模板需按照如下步骤设置。

i. 设置转码模板时，**模板类型**根据您的实际转码需要，选择使用**自定义模板**（您自定义的窄带高清TM1.0模板）或**预置窄带高清模板**（模板名以**窄带高清TM**结尾）。




ii. 单击**提交任务**，完成创建转码任务触发窄带高清TM1.0转码。

## 工作流触发窄带高清TM1.0转码

1. 登录媒体处理控制台。
2. 在顶部菜单栏左侧选择地域。

3. 添加媒体Bucket。如果媒体Bucket已添加，则直接进入下一步。
4. 创建转码工作流，具体操作请参见创建转码工作流。其中转码节点需按照如下步骤设置。
   i. 在转码节点参数配置中，单击**转码模板**右侧**选择**。
   ii. **模板类型**选择**自定义模板**（您自定义的窄带高清TM1.0模板）或**预置窄带高清模板**（模板名以**窄带高清TM**结尾），具体模板根据您的实际需要选择。

   > **说明** 更多预置模板参数信息请参见预置模板详情。

   iii. 单击**确定**，完成窄带高清TM1.0模板选择。

iv. 其余参数根据您的需要配置，配置完成后单击**确定**。

v. 所有参数配置完成后单击**保存**，完成工作流创建。

在工作流设置页面可以查看模板ID、模板详情和修改模板状态。

# 5.1.4. 创建窄带高清TM2.0转码任务

窄带高清TM2.0是一项基于阿里云转码技术的媒体处理功能。从人眼视觉模型出发，将编码器的优化目标从保真度最高调整为主观体验最好，在节省码率的同时，也能提供更加清晰的观看体验。

## 使用限制

- 窄带高清TM2.0目前仅支持杭州、上海地域。
- 窄带高清TM2.0预置模板适用于大部分互联网视频，如预置窄带高清模板不能满足您的转码需求，可以根据您的实际转码需求自定义窄带高清TM2.0模板。具体操作，请参见（可选）自定义窄带高清TM2.0模板。
- 窄带高清TM2.0转码速度较慢，建议使用3分钟以内的短视频测试 。

## 使用方法

窄带高清TM2.0转码可以通过控制台创建转码任务或工作流触发，您可以使用预置窄带高清TM2.0模板或根据您的实际转码需要自定义窄带高清TM2.0模板。

> **说明** 如您需要通过API触发窄带高清2.0转码任务，请参考媒资转码进行操作。

## 转码准备

在进行转码任务之前，需要进行一些设置确保功能能够正常使用，如果已设置请忽略。

- 参考前置条件完成所需设置。
- 媒体处理操作的媒体资源均存放在OSS的Bucket中，需要添加媒体Bucket才可以在任务、工作流中选择对应的Bucket作为输入和输出路径。具体操作，请参见添加媒体Bucket。
- 开启管道。如果管道列表中没有您需要的管道类型，请先提交工单开通所需类型的管道。
- （可选）如预置转码模板不能满足您的实际转码需要，您可以自定义转码模板。具体操作，请参见创建自定义模板。
- （可选）如您有添加水印的需求，请先创建水印模板。具体操作，请参见添加水印模板。
- （可选）如您需要在工作完成后接收消息通知，请先配置消息通知服务。具体操作，请参见设置转码消息通知。

## （可选）自定义窄带高清TM2.0模板

> **说明** 当预置窄带高清TM2.0模板不能满足您的实际转码需求时，可以根据需要进行自定义模板。自定义模板开通成功后，可以在控制台创建单任务转码、工作流创建转码任务、API创建转码任务时使用。

1. 登录媒体处理控制台。
2. 在顶部菜单栏左侧选择地域。

3. 在左侧导航栏中，单击**全局设置 > 转码模板**。
4. 单击**添加转码模板**，添加**转码模板名称**。
5. 在**基本参数**区域的**封装格式**下拉框中选择**mp4**或**m3u8**。其他参数根据您的转码需要进行配置。参数详情，请参见转码设置。配置完成后单击**下一步**。
6. 根据实际需求配置**音频参数**，单击**下一步**。
7. 根据实际需求配置**高级参数**，单击**提交**。
8. 在**转码模板**页面，获取配置成功的转码模板ID。

> **说明** 在API触发窄带高清TM2.0转码时如果您使用自定义模板，则填入此步骤获取的转码模板ID即可。

9. 提交工单并提供转码模板ID、UID、开通区域等信息开通窄带高清TM2.0模板。
   开通成功后，模板会升级为窄带高清TM2.0模板。例如：配置的普通模板的清晰度为标清，开通成功后，模板的清晰度成为标清-窄带高清TM2.0，其他配置参数设置不变。

## 创建转码任务触发窄带高清TM2.0转码

1. 登录媒体处理控制台。
2. 在顶部菜单栏左侧选择地域。

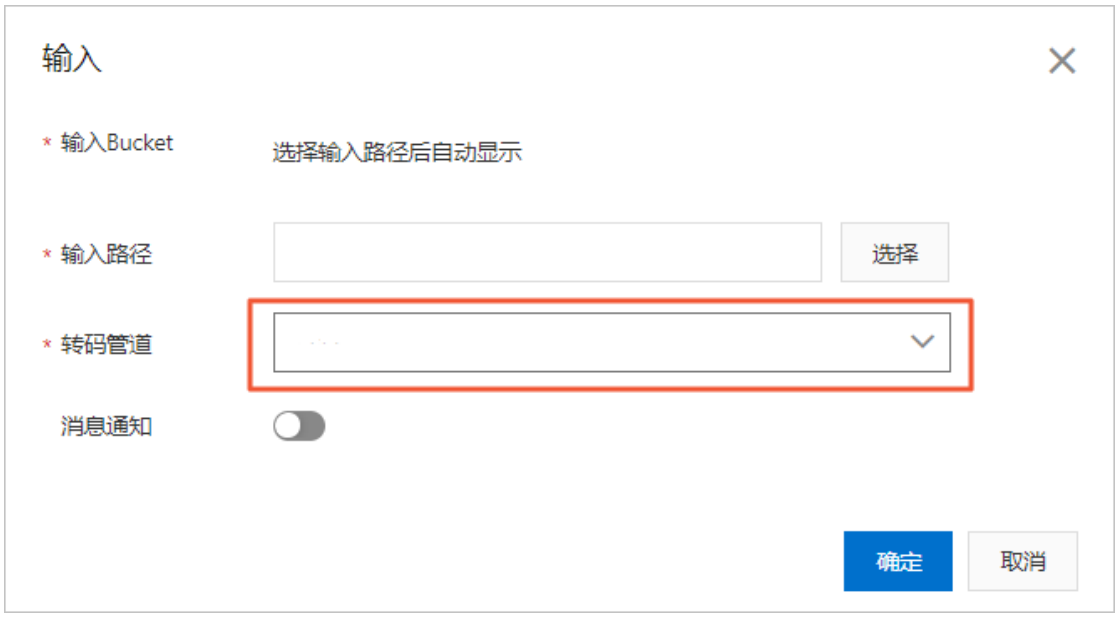
3. 添加媒体Bucket。如媒体Bucket已添加，则直接进入下一步。
4. 开启窄带高清TM2.0管道，具体操作，请参见开启管道。如管道已经开启，则直接进入下一步。
5. 创建转码任务。具体操作，请参见创建转码任务。其中转码模板需按照如下步骤设置。
   i. 设置转码模板时，**模板类型**根据您的实际转码需要，选择使用自定义模板（您自定义的窄带高清TM2.0模板）或预置窄带高清TM2.0模板（以窄带高清TM2.0结尾）。

   ii. 单击**提交任务**，完成通过创建转码任务触发窄带高清TM2.0转码操作。

## 工作流触发窄带高清TM2.0转码

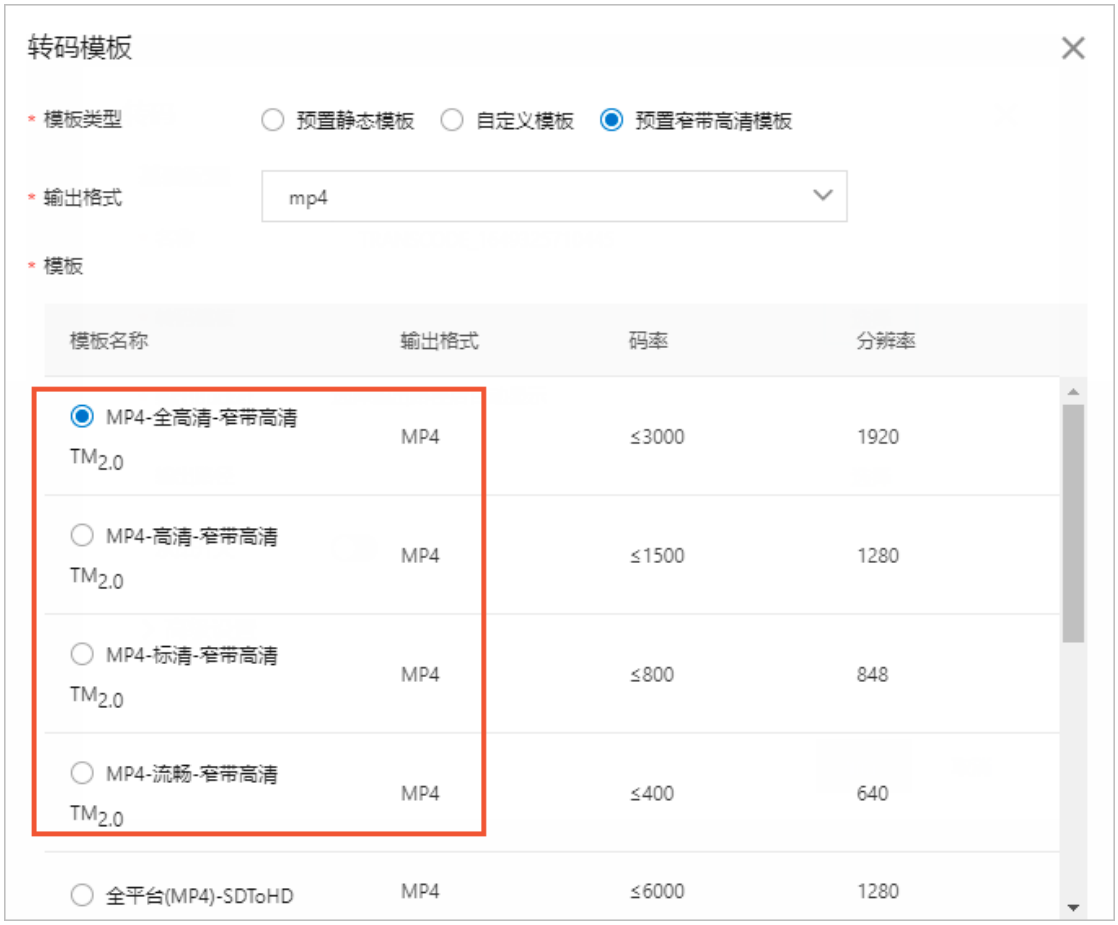
1. 登录媒体处理控制台。
2. 在顶部菜单栏左侧选择地域。

3. 添加媒体Bucket。如媒体Bucket已经添加，则直接进入下一步。
4. 开启窄带高清TM2.0管道，具体操作，请参见开启管道。如管道已经开启，则直接进入下一步。
5. 创建转码工作流，具体操作，请参见创建转码工作流。其中**输入**节点需按照如下步骤配置。
   i. 在输入节点参数配置中，单击转码管道下拉列表，选择已开启的窄带高清2.0管道。




   ii. 在转码节点参数配置中，单击转码模板右侧**选择**。

iii. **模板类型**选择**自定义模板**（您自定义的窄带高清TM2.0模板）或预置窄带高清TM2.0模板（模板名以窄带高清TM2.0结尾），具体模板根据您的实际需要选择。




说明 更多预置模板参数信息请参见预置模板详情。

iv. 单击**确定**，完成选择窄带高清TM2.0模板。

v. 其余参数根据您的需要配置，配置完成后单击**确定**。

vi. 所有参数配置完成后单击**保存**，完成工作流创建。

在工作流设置页面可以查看模板ID、模板详情和修改模板状态。

# 5.1.5. 创建分辨率倍增任务

本文介绍了如何使用分辨率倍增对视频画质进行标清版本转高清版本，或2K节目源转4K节目源的操作方法及说明。

## 使用限制

- 分辨率倍增目前仅支持华东1（杭州）、华东2（上海）地域。单击开通分辨率倍增专用管道申请分辨率倍增管道。申请管道时，请备注分辨率倍增，收费标准会在管道开通后告知。启用管道，请参见开启管道。
- 分辨率倍增转码速度较慢，建议使用3分钟以内的短视频测试。

## 使用方法

分辨率倍增转码可以通过控制台创建转码任务或工作流触发，使用前请先开启分辨率倍增转码管道。

> **说明** 如果您需要通过API触发分辨率倍增转码任务，具体操作请参见媒资转码。

## 分辨率倍增转码模板说明

- 全平台(MP4)-SDTo HD（标清转高清）：对于标清的经典老片，去除胶片颗粒和压缩噪音，加以超分辨率技术，生成720P甚至1080P的高清版本。
- 全平台(MP4)-2KTo4K（2K转4K）：对于1080P影片，利用基于海量视频训练的超分辨率技术，生成高品质4K节目源。

## 转码准备

在进行转码任务之前，需要进行一些设置确保功能能够正常使用，如果已设置请忽略。

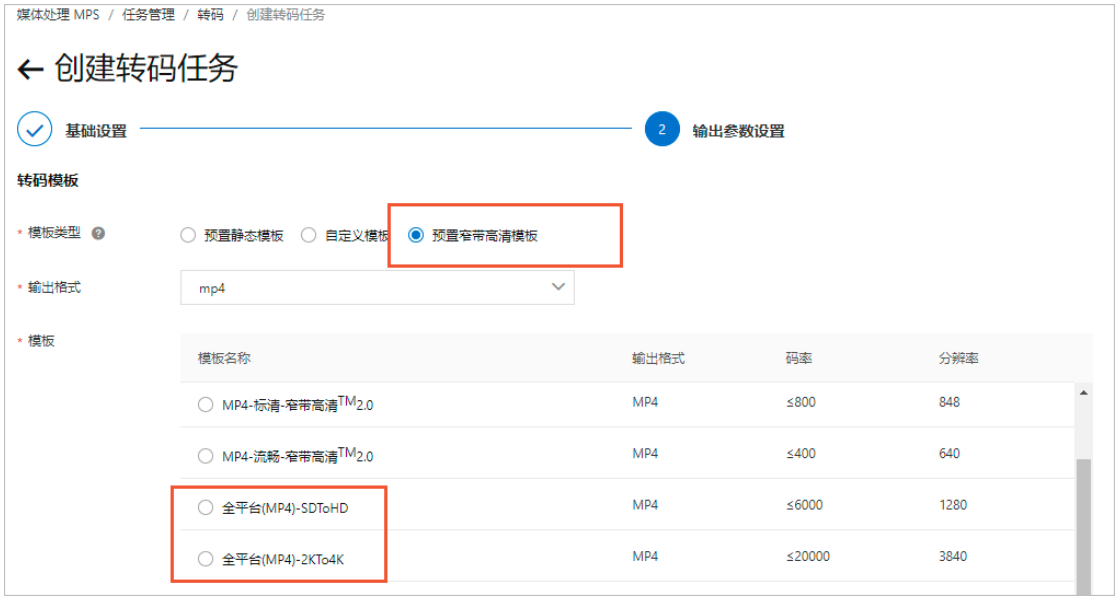
- 参考前置条件完成所需设置。
- 媒体处理操作的媒体资源均存放在OSS的Bucket中，需要添加媒体Bucket才可以在任务、工作流中选择对应的Bucket作为输入和输出路径。具体操作，请参见添加媒体Bucket。
- 开启管道。如果管道列表中没有您需要的管道类型，请先提交工单开通所需类型的管道。
- （可选）如预置转码模板不能满足您的实际转码需要，您可以自定义转码模板。具体操作，请参见创建自定义模板。
- （可选）如您有添加水印的需求，请先创建水印模板。具体操作，请参见添加水印模板。
- （可选）如您需要在工作完成后接收消息通知，请先配置消息通知服务。具体操作，请参见设置转码消息通知。

## 创建转码任务触发分辨率倍增转码

1. 登录媒体处理控制台。
2. 在顶部菜单栏左侧选择地域。

3. 创建转码任务，具体操作请参见创建转码任务。其中转码模板需按照如下步骤设置。

i. 在**输出参数设置**界面，**模板类型**选项中，选择**预置窄带高清模板**中模板名以**全平台（MP4）**开头的模板。




ii. 单击**提交任务**。

在控制台左侧导航栏，选择**任务管理**，在任务管理页面**创建转码任务**页签下方列表中查看转码结果及详细信息。

## 工作流触发分辨率倍增转码

1. 登录媒体处理控制台。
2. 在顶部菜单栏左侧选择地域。

3. 添加媒体Bucket。如媒体Bucket已经添加，则直接进入下一步。
4. 创建转码工作流。具体操作，请参见创建转码工作流。其中**转码**节点需按照如下步骤配置。

   i. 在**转码**节点基础配置中，单击**转码模板**右侧**选择**。

ii. 在**模板类型**选项中，选择**预置窄带高清模板**中模板名以**全平台（MP4）**开头的模板。单击**确定**完成分辨率倍增模板选择。

iii. 其余参数根据您的需要配置，配置完成后单击**确定**。

按照创建转码工作流配置完**发布**节点保存后，完成分辨率倍增工作流创建。

## 后续步骤

- 工作流创建成功后，将自动激活为启用状态，上传至**输入**节点绑定的**输入路径**的音视频文件会自动触发该工作流的执行。您可以通过媒体处理控制台进行上传视频文件，也可以通过OSS上传视频文件（推荐），文件上传后自动生成已经执行了工作流条件的媒体文件。上传视频文件，请参见上传视频。
- 查看生成媒体文件，具体操作，请参见媒体列表。查看工作流执行状态具体操作，请参见执行实例。

# 5.2. 视频AI

## 5.2.1. 视频AI概述

媒体处理提供视频生产、媒体审核、视频DNA、智能标签等视频AI服务。本文为您简要介绍这些功能。

### 应用场景

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：

- 分析文件内容生成智能封面、绿幕抠图、智能图标模糊等多种媒体处理功能。
- 进行内容审核，检测违规的视频内容，适用于短视频平台、直播平台、传媒审核等场景。
- 识别并搜索视频中重复或相似片段，适用于原创识别、视频查重、快速审核、广告分成、视频溯源等场景。

- 准确识别内容，自动输出多维度标签，适用于媒资检索、个性化推荐、智能广告投放等场景。

## 功能介绍

- 视频生产：基于阿里云媒体AI技术，提供多种形式的媒体内容处理及内容生成能力，提升媒体内容生产效率和质量。
  - 智能封面：通过对视频内容的理解，结合画面美学和海量生产数据，选出最优的关键帧或关键片段作为视频封面，提升视频点击转化及用户体验。
    - 封面图片：选取展现视频内容效果最优的图片，作为视频的图片封面。
    - 封面动图：提取视频多帧精彩瞬间，拼接生成GIF动图，作为视频的动图封面。
  - 视频摘要：提取视频精彩内容片段，融合生成最具代表性的5s短视频摘要。
  - 图标模糊：智能识别视频中的logo图标位置，并对图标进行模糊处理，恢复视频未添加logo的原始状态，辅助视频内容的后期制作及再生产。
  - 智能横转竖：智能识别视频画面主体，进行比例适应裁剪，将横屏拍摄的视频转换成适于移动端竖屏播放的视频，助力移动时代视频内容二次生产及多端传播投放。
  - 智能去字幕：采用文字检测及图像融合处理技术，智能检测并去除视频或图像中的文字字幕，还原视频纯净画面。
  - 智能抠图：包含人像抠图和绿幕抠图。
    - 人像抠图：人像抠图采用海量数据训练的神经网络模型，自动检测视频图像中的人像轮廓，进行高清细节抠图，精确提取人像图像。
    - 绿幕抠图：绿幕抠图可对视频图像的画面前景背景进行分析提取，检测纯色背景并替换为指定的背景图，适用于影视制作、短视频制作及后期编辑场景。
  - 字幕提取：对视频中的字幕进行识别，提取字幕文本内容及时间信息，输出为srt字幕文件，支持中英文字幕识别提取，可用于视频字幕加工场景。
  - 歌曲检测：对副歌和音乐节奏进行检测。
    - 副歌检测：对歌曲进行段落分析，识别音乐中的副歌“高潮片段”，提取副歌段落时间信息，可以用于音乐推荐试听，热门片段K歌等场景。
    - 音乐节奏检测：音乐节奏检测可分析识别音乐中的多层次的节奏点，返回节奏点时间信息，可用于音视频生产、音乐推荐等场景。
  - 智能混音：智能混音对多个输入音频进行音轨平衡处理及智能混音，输出混音合成音频，支持人声、音乐等多种音频处理。
  - PPT拆条：对包含PPT演示的视频进行智能分析，将PPT演示视频拆分成多个时间片段，并自动为每个片段生成标题，用于场景课程类视频自动编辑。
- 媒体审核：基于海量标注数据和深度学习算法，提供视频图像文件内容质量审核功能，从语音、文字、视觉等多维度精准识别媒体文件中包含的违禁内容。

- 媒体内容审核：对媒体文件的标题、简介、内容、封面等媒资全维度内容进行审核，识别语音、文字、画面中的违规内容。
  - 涉黄识别：通过神经网络算法结合实时更新的亿级样本库，智能识别语音、文字、画面中的色情和性感内容。
  - 暴恐涉政识别：采用深度学习算法结合实时更新的样本库，快速识别武器、血腥场面、特殊装束、爆炸烟光、特殊标识、聚众、游行等十余个维度的暴恐涉政内容。
  - 广告/二维码识别：智能检测媒体文件中的文字、水印、二维码，有效识别牛皮癣广告、二维码、小程序码等多种形式的图文广告。
  - Logo识别：采用目标检测技术，精准识别视频图片中的各种logo图标，如台标、商标、水印等，常用于版权保护等场景。
  - 不良场景识别：结合行为分析和时间序列对比技术，精准识别画中画、吸烟、车内直播、无意义画面等需要监管的不良场景。
  - 语音垃圾识别：采用先进的声学模型与语言模型，识别语音中存在的涉黄、暴恐涉政、辱骂等违规信息，支持中文、英文语音识别。
- 媒体质量审核：对媒体文件的内容质量进行分析，识别画面、音频中的多种类型质量问题。
  - 画面质量审核：识别媒体文件画面质量问题，包括抖动重影、模糊、低光照、过曝光、黑屏、白屏、噪点、马赛克、雪花、静帧、跳帧、二次录制等。
  - 音频质量审核：对媒体音频质量进行智能分析，识别音频卡顿、静音、无音轨等音频质量问题。
  - 美观度分值：综合评判画面质量与画面美感，给出视频美观度分值，衡量视频内容美观度质量。

- 视频DNA：基于阿里云视频识别技术，通过一个二进制串来唯一标记一个视频，提供提取视频中图像、音频等指纹特征对比功能，可以解决重复视频查找、视频片段查源等问题。
  - 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。
  - 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。
  - 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，在此基础上，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。

- 智能标签：基于对视频中的视觉、文字、语音、行为等信息进行分析，结合多模态信息融合及对齐技术，实现高准确率内容识别，同时将非结构化信息转化为结构化信息。
  - 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。
    - 视觉识别：分析视频的视觉图像信息，识别人物、物体、场景等图像内容。
    - 文字识别：识别视频图像中的文字，提取图像文本内容。
    - 语音识别：分析视频中的语音信息，提取语音文本内容。
    - 运动行为识别：分析视频动态信息，识别运动、动作、行为事件等内容。

- 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。
  - 多模态对齐：识别多模态间不同元素的联系和对齐关系，实现视频文字、语音、视觉等不同模态信息的对齐理解。
  - 多模态融合：整合不同模态间的模型和特征，提高理解能力和效率，实现语音文字文本纠错能力，提升识别准确率。
  - 多模态联合表征：将多模态信息进行整合及联合表征，实现视频内容的一体化识别。
- 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。
  - 智能标签提取：标签处理集成自定义过滤、标签去重、标签合并、权重排序等算法，提供高效、准确、实用的智能打标能力。
  - 多维度标签输出：结合融合理解结果及标签体系，输出多维度视频标签，包括视频分类标签、人物标签、物体场景标签、文本标签等。具体标签类别包括：人脸、地域、标识、组织机构、表情、人物角色、动作事件、场景、物体、视频分类等大类标签，及上万个细分标签。
  - 支持自定义标签：支持地标词库注册、人脸自注册、未知人脸聚类、标签黑名单、标签热词等标签自定义功能，满足不同场景下的使用需求。

## 功能使用

- 视频生产
- 媒体审核
- 视频DNA
- 智能标签

# 5.2.2. 视频生产

视频生产服务基于阿里云媒体AI技术，提供多种形式媒体内容处理及内容生成能力，支持智能封面、智能横转竖、绿幕抠图、人像抠图、智能图标模糊、智能去字幕、字幕提取、副歌检测、音乐节奏检测、PPT拆条等多种媒体处理生成功能，提升媒体内容生产效率和质量。

## 使用限制

视频生产各功能支持的地域如下表所示：

| 功能 | 华东1（杭州） | 华东2（上海） | 华北2（北京） | 亚太东南1（新加坡） |
| --- | --- | --- | --- | --- |
| VideoH2V（视频横转竖） | √ | √ | √ | × |
| VideoDelogo（视频去图标） | √ | √ | √ | × |
| Cover（智能封面） | √ | √ | √ | × |
| VideoClip（视频摘要） | × | √ | √ | √ |
| ImageMatting（图片人像分割） | × | √ | × | × |

| 功能 | 华东1（杭州） | 华东2（上海） | 华北2（北京） | 亚太东南1（新加坡） |
| --- | --- | --- | --- | --- |
| ImageH2V（横转竖） | × | √ | √ | × |
| ImageDelogo（图片去图标） | × | √ | √ | × |
| CaptionExtraction（字幕提取） | × | √ | × | × |
| VideoDetext（视频去字幕） | × | √ | √ | × |
| VideoGreenScreenMatting（绿幕抠图） | × | √ | × | × |
| MusicSegmentDetect（副歌检测） | √ | √ | × | × |
| AudioMixing（混音处理） | × | √ | × | × |
| VideoSplit（PPT拆条） | √ | √ | √ | × |
| AudioBeatDetection（节奏检测） | √ | √ | × | × |
| ASR（语音识别） | √ | √ | √ | × |
| AsyncTextToSpeech（异步语音合成） | √ | √ | √ | × |
| TextToSpeech（语音合成） | × | √ | × | × |
| ImageCartoonize（人脸风格化） | × | √ | × | × |

## 应用场景

- 传媒、娱乐视频智能生产

  对视频内容进行图标、字幕、抠图等处理，快速制作加工视频内容。

- 视频多端投放

  对一次拍摄产生的视频使用智能横转竖服务进行比例转换，生成适配不同终端播放的版本，向多种平台及终端进行投放和分发，提升视频内容传播效率。

## 使用方法

视频生产中的智能封面功能可以通过控制台创建智能封面任务或工作流触发。

> **说明** 如您需要通过API创建视频生产任务，请参考视频生产进行操作。

## 使用准备

在使用前需要进行一些设置确保功能能够正常使用，如已设置请忽略。

- 参考前置条件完成所需设置。
- 媒体处理操作的媒体资源均存放在OSS的Bucket中，需要添加媒体Bucket才可以在任务、工作流中选择对应的Bucket作为输入和输出路径。具体操作，请参见添加媒体Bucket。
- 开启管道。如果管道列表中没有智能封面管道，请先提交工单开通智能封面管道。
- （可选）如您需要在工作完成后接收消息通知，请先配置消息通知服务。具体操作，请参见设置转码消息通知。

## 创建智能封面单任务

1. 登录媒体处理控制台。
2. 在顶部菜单栏左侧选择地域。
3. 在左侧导航栏，选择**任务管理**。
4. 选择**智能封面**，单击**创建任务**。
5. 设置参数。

| 参数 | 说明 |
| --- | --- |
| **待处理文件地址** | 单击**选择**，在**Bucket**下拉列表中，选择Bucket名称。**路径**下方会显示对应Bucket已经创建好的文件夹，在文件夹下选择一个地址作为输入路径。<br>**说明** **待处理文件地址**为OSS Bucket输入存储路径，如果下拉列表中没有Bucket，请先添加Bucket。具体操作，请参见添加媒体Bucket。 |
| **智能封面管道** | 在下拉列表中选择管道名称。<br>**说明** 如果**智能封面管道**下拉列表中没有管道名称，请先开启智能封面专用管道。具体操作，请参见开启管道。 |
| **输出路径** | 单击**选择**，在**Bucket**下拉列表中，选择Bucket名称。**路径**下方会显示对应Bucket已经创建好的文件夹，在文件夹下选择一个地址作为输出路径。<br>**说明** **输出路径**为OSS Bucket输出存储路径，如果下拉列表中没有Bucket，请先添加Bucket。具体操作，请参见添加媒体Bucket。 |

6. 单击**提交任务**，完成创建智能封面任务。

   任务提交后您可以在当前页签下方列表中查看任务的执行状态以及配置信息和输入、输出地址等。

## 创建智能封面工作流

1. 登录媒体处理控制台。
2. 在顶部菜单栏左侧选择地域。
3. 在左侧导航栏，选择**工作流管理 > 工作流设置**。
4. 单击**创建工作流**。
5. 设置工作流基础信息。

| 参数 | 说明 |
| --- | --- |
| **工作流名称** | 在文本框中输入工作流名称。 |
| **配置方案** | 在下拉列表中选择**自定义**。<br>**说明** 为满足通用场景需求，媒体处理提供了多种预置工作流，您也可以在**配置方案**下拉列表中选择一个预置工作流的进行快速编辑。预置工作流说明如下：<br>○ **M3U8**：进行单张封面截图，并把视频转为M3U8格式。<br>○ **预置智能模板工作流**：进行单张封面截图，并根据视频信息进行智能分析后转码，对转码后的标清视频截图。<br>○ **多码率多格式工作流**：进行单张封面截图，并生成多种视频格式和码率视频。<br>○ **（FVL、M3U8、MP4）多码率工作流**：进行单张封面截图，并生成多码率视频。<br>更多详情，请参见预置模板详情。 |

6. 设置**输入**节点。
   i. 点击**输入**节点右侧的笔形图标，进入**输入设置**页面。

ii. 设置输入参数。

| 参数 | 说明 |
| --- | --- |
| **输入路径** | 单击**选择**，在**Bucket**下拉列表中，选择Bucket名称。**路径**下方会显示对应Bucket已经创建好的文件夹，在文件夹下选择一个地址作为输入路径。<br>**说明** 如果下拉列表中没有Bucket，请先添加Bucket。具体操作，请参见添加媒体Bucket。 |
| **转码管道** | 在下拉列表中选择管道。<br>**说明** 如果下拉列表中没有管道名称，请先开启管道。具体操作，请参见开启管道。 |
| （可选）**消息通知** | 单击**开关按钮**，选择消息队列或消息主题，在下拉列表中选择一个队列通知或主题通知。<br>**说明** 如果下拉列表中没有队列通知或主题通知，请先开启消息通知。具体操作，请参见设置工作流消息通知。 |

iii. 单击**确定**，完成**输入**节点设置。

7. 添加**智能封面**节点。

i. 单击**输入**节点右侧加号图标选择**智能封面**。

ii. 单击**智能封面**节点右侧笔形图标。

iii. 设置智能封面。

| 参数 | 说明 |
| --- | --- |
| **名称** | 在文本输入框中输入封面名称。 |
| **智能封面管道** | 在下拉列表中选择管道。<br>说明 如果**智能封面管道**下拉列表中没有管道名称，请先开启智能封面专用管道。具体操作，请参见开启管道。 |
| **输出Bucket** | 选择输出路径后自动显示，不用手动设置。 |
| **输出路径** | 单击**选择**，选择OSS输出地址。 |
| （可选）**设置封面** | 单击**开关按钮**，将图片设为封面。 |

iv. 单击**确定**，完成**智能封面**节点设置。

8. 设置**发布**节点。

i. 单击**发布**节点右侧笔形图标。

ii. 设置发布。

| 参数 | 说明 |
| --- | --- |
| **媒体发布类型** | 选择**媒体发布类型**为**自动**。<br>■ **手动**发布：工作流执行成功后，将媒体设置为未发布状态，之后您可以手动发布该媒体。手动发布媒体文件，请参见媒体列表。<br>■ **自动**发布：工作流执行成功后，将媒体设置为发布状态。<br>说明<br>■ 发布含义：设置媒体所有播放资源、截图文件的访问权限继承所在Bucket的访问权限。<br>■ 不发布含义：媒体所有播放资源、截图文件的访问权限为私有。 |

iii. 单击**确定**，完成发布节点设置。

9. 单击**保存**，完成工作流创建。
创建完成后，当有符合条件的新文件进入指定路径时，会自动触发工作流执行。

## 5.2.3. 媒体审核

媒体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媒体文件中包含的违禁内容，包括色情、暴恐、政治敏感、广告识别等多个功能模块，同时支持视频图像文件内容质量审核功能，应用于短视频平台、直播平台、传媒审核等场景。

## 使用限制

媒体审核中的视频审核功能，目前开通的地域为华北2（北京）、华东2（上海），其他地域暂不支持。

## 应用场景

- 精准高效的UGC内容审核：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，识别媒体质量问题，有效减少人工审核的工作量。
- 实时准确的直播内容审核：实时监控直播视频流的语音、文字、画面内容，快速响应，控制业务风险。
- 高效精准的点播内容审核：高效识别点播视频中的涉暴、涉政类镜头，紧跟时政最新动态，严控违规内容。

## 使用方法

媒体审核中的视频审核功能可以通过控制台创建视频审核任务或工作流触发。

> **说明** 如您需要通过API创建媒体审核任务，请参考媒体审核操作。

## 使用准备

在使用前需要进行一些设置确保功能能够正常使用，如已设置请忽略。

- 参考前置条件完成所需设置。
- 媒体处理操作的媒体资源均存放在OSS的Bucket中，需要添加媒体Bucket才可以在任务、工作流中选择对应的Bucket作为输入和输出路径。具体操作，请参见添加媒体Bucket。
- 开启管道。如果管道列表中没有您需要的管道类型，请先提交工单开通所需类型的管道。
- （可选）如您需要在工作完成后接收消息通知，请先配置消息通知服务。具体操作，请参见设置转码消息通知。

## 创建视频审核单任务

1. 登录媒体处理控制台。
2. 在顶部菜单栏左侧选择地域。
3. 在左侧导航栏，选择**任务管理**。
4. 选择**视频审核**，单击**创建任务**。
5. 设置**输入**和**输出**参数。

| 参数 | 说明 |
| --- | --- |

| 参数 | 说明 |
| --- | --- |
| **待处理文件地址** | 单击**选择**，在**Bucket**下拉列表中，选择Bucket名称。**路径**下方会显示对应Bucket已经创建好的文件夹，在文件夹下选择一个地址作为输出路径。<br>**说明** **待转码文件地址**为OSS Bucket输入存储路径，如果下拉列表中没有Bucket，请先添加Bucket。具体操作，请参见添加媒体Bucket。 |
| **视频审核管道** | 在**视频审核管道**下拉列表中选择管道名称。<br>**说明** 如果视频审核管道下拉列表中没有管道名称，请先开启视频审核管道。具体操作，请参见开启管道。 |
| **审核类型** | 默认勾选**色情**、**暴恐涉政**。 |
| **存储路径** | 单击**选择**，在**Bucket**下拉列表中，选择Bucket名称。**路径**下方会显示对应Bucket已经创建好的文件夹，在文件夹下选择一个地址作为输出路径。<br>**说明** **输出路径**为OSS Bucket输出存储路径，如果下拉列表中没有Bucket，请先添加Bucket。具体操作，请参见添加媒体Bucket。 |

6. 单击**提交任务**，完成创建提交视频审核任务。

   任务提交后您可以在当前页签下方列表中查看任务的执行状态以及配置信息和输入、输出地址等。

## 创建审核工作流

1. 登录媒体处理控制台。
2. 在顶部菜单栏左侧选择地域。
3. 在左侧导航栏，选择**工作流管理 > 工作流设置**。
4. 单击**创建工作流**。
5. 设置工作流基础信息。

| 参数 | 说明 |
| --- | --- |
| **工作流名称** | 在文本框中输入工作流名称。 |

| 参数 | 说明 |
| --- | --- |
| **配置方案** | 在下拉列表中选择**自定义**。<br>**说明** 为满足通用场景需求，媒体处理提供了多种预置工作流，您也可以在**配置方案**下拉列表中选择一个预置工作流的进行快速编辑。预置工作流说明如下：<br>○ **M3U8**：进行单张封面截图，并把视频转为M3U8格式。<br>○ **预置智能模板工作流**：进行单张封面截图，并根据视频信息进行智能分析后转码，对转码后的标清视频截图。<br>○ **多码率多格式工作流**：进行单张封面截图，并生成多种视频格式和码率视频。<br>○ **（FVL、M3U8、MP4）多码率工作流**：进行单张封面截图，并生成多码率视频。<br>更多详情，请参见预置模板详情。 |

6. 设置**输入**节点。
   i. 点击**输入**节点右侧的笔形图标，进入**输入设置**页面。

ii. 设置输入参数。

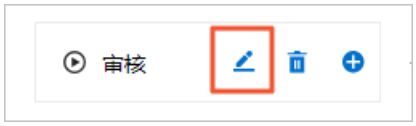
| 参数 | 说明 |
| --- | --- |
| **输入路径** | 单击**选择**，在**Bucket**下拉列表中，选择Bucket名称。**路径**下方会显示对应Bucket已经创建好的文件夹，在文件夹下选择一个地址作为输入路径。<br>**说明** 如果下拉列表中没有Bucket，请先添加Bucket。具体操作，请参见添加媒体Bucket。 |
| **转码管道** | 在下拉列表中选择管道。<br>**说明** 如果**转码管道**下拉列表中没有管道名称，请先开启管道。具体操作，请参见开启管道。 |
| （可选）**消息通知** | 单击**开关按钮**，选择消息队列或消息主题，在下拉列表中选择一个队列通知或主题通知。<br>**说明** 如果下拉列表中没有队列通知或主题通知，请先开启消息通知。具体操作，请参见设置工作流消息通知。 |

iii. 单击**确定**，完成**输入**节点设置。

7. 添加**审核**节点。

i. 单击**输入**节点右侧加号图标选择**审核**。

ii. 单击**审核**节点右侧笔形图标。



iii. 设置**审核**参数。

| 参数 | 说明 |
| --- | --- |
| **审核管道** | 在下拉列表中选择管道。<br>说明 如果**审核管道**下拉列表中没有管道名称，请先开启视频审核专用管道。具体操作，请参见开启管道。 |
| **输出路径** | 单击**选择**，选择OSS输出地址。 |
| **是否终止工作流** | 在选项中，根据实际需求进行选择。 |

iv. 单击**确定**，完成**审核**节点设置。

8. 设置**发布**节点。
   i. 单击**发布**节点右侧笔形图标。
   ii. 设置发布。

| 参数 | 说明 |
| --- | --- |
| **媒体发布类型** | 选择**媒体发布类型**为**自动**。<br>■ **手动**发布：工作流执行成功后，将媒体设置为未发布状态，之后您可以手动发布该媒体。手动发布媒体文件，请参见媒体列表。<br>■ **自动**发布：工作流执行成功后，将媒体设置为发布状态。<br>说明<br>■ 发布含义：设置媒体所有播放资源、截图文件的访问权限继承所在Bucket的访问权限。<br>■ 不发布含义：媒体所有播放资源、截图文件的访问权限为私有。 |

   iii. 单击**确定**，完成发布节点设置。

9. 单击**保存**，完成工作流创建。
   创建完成后，当有符合条件的新文件进入指定路径时，会自动触发工作流执行。

# 5.2.4. 视频DNA

视频DNA通常是一个二进制串，用来唯一标记一个视频。您可以使用视频DNA服务提取并比对视频中的图像、音频等指纹特征，解决重复视频查找、视频片段查源等问题，适用于原创识别、视频查重、快速审核、广告分成、视频溯源等业务场景。

## 使用限制

视频DNA功能，目前开通的地域为华北2（北京）、华东1（杭州）、华东2（上海）、亚太东南1（新加坡）、亚太东南2（雅加达）、亚太南部1（孟买），其他地域暂不支持。

## 使用方法

视频DNA可以通过控制台创建视频DNA任务或工作流触发。

> **说明** 如您需要通过API创建视频DNA任务，请参考视频DNA进行操作。

## 特性说明

- 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。

## 使用准备

在使用前需要进行一些设置确保功能能够正常使用，如已设置请忽略。

- 参考前置条件完成所需设置。
- 媒体处理操作的媒体资源均存放在OSS的Bucket中，需要添加媒体Bucket才可以在任务、工作流中选择对应的Bucket作为输入和输出路径。具体操作，请参见添加媒体Bucket。
- 开启管道。如果管道列表中没有视频DNA管道，请先提交工单开通视频DNA管道。
- 提交新建DNA库，建完库后提交工单并提供UID、使用区域、DNA库ID等信息进行绑定。
- （可选）如您需要在工作完成后接收消息通知，请先配置消息通知服务。具体操作，请参见设置转码消息通知。

## 创建视频DNA单任务

1. 登录媒体处理控制台。
2. 在顶部菜单栏左侧选择地域。

3. 在左侧导航栏，选择**任务管理**。
4. 选择**视频DNA**，单击**创建任务**。
5. 设置参数。

| 参数 | 说明 |
| --- | --- |

| 参数 | 说明 |
| --- | --- |
| **待处理文件地址** | 单击**选择**，在**Bucket**下拉列表中，选择Bucket名称。**路径**下方会显示对应Bucket已经创建好的文件夹，在文件夹下选择一个地址作为输入路径。<br>说明 **待转码文件地址**为OSS Bucket输入存储路径，如果下拉列表中没有Bucket，请先添加Bucket。具体操作，请参见添加媒体Bucket。 |
| **视频DNA管道** | 在下拉列表中选择管道名称。<br>说明 如果**视频DNA管道**下拉列表中没有管道名称，请先开启视频DNA管道。具体操作，请参见开启管道。 |
| **入库规则** | 在**入库规则**选项中选择规则，默认为**仅入库不重复内容**。 |

6. 单击**提交任务**，完成创建提交视频DNA任务。

   任务提交后您可以在当前页签下方列表中查看任务的执行状态以及配置信息和输入、输出地址等。

## 创建视频DNA工作流

1. 登录媒体处理控制台。
2. 在顶部菜单栏左侧选择地域。

3. 在左侧导航栏，选择**工作流管理 > 工作流设置**。
4. 单击**创建工作流**。
5. 设置工作流基础信息。

| 参数 | 说明 |
| --- | --- |
| **工作流名称** | 在文本框中输入工作流名称。 |

| 参数 | 说明 |
| --- | --- |
| **配置方案** | 在下拉列表中选择**自定义**。<br>**说明** 为满足通用场景需求，媒体处理提供了多种预置工作流，您也可以在**配置方案**下拉列表中选择一个预置工作流的进行快速编辑。预置工作流说明如下：<br>○ **M3U8**：进行单张封面截图，并把视频转为M3U8格式。<br>○ **预置智能模板工作流**：进行单张封面截图，并根据视频信息进行智能分析后转码，对转码后的标清视频截图。<br>○ **多码率多格式工作流**：进行单张封面截图，并生成多种视频格式和码率视频。<br>○ **（FVL、M3U8、MP4）多码率工作流**：进行单张封面截图，并生成多码率视频。<br>更多详情，请参见预置模板详情。 |

6. 设置**输入**节点。
   i. 点击**输入**节点右侧的笔形图标，进入**输入设置**页面。

ii. 设置输入参数。

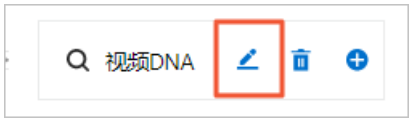
| 参数 | 说明 |
| --- | --- |
| **输入路径** | 单击**选择**，在**Bucket**下拉列表中，选择Bucket名称。**路径**下方会显示对应Bucket已经创建好的文件夹，在文件夹下选择一个地址作为输入路径。<br>**说明** 如果下拉列表中没有Bucket，请先添加Bucket。具体操作，请参见添加媒体Bucket。 |
| **转码管道** | 在下拉列表中选择管道。<br>**说明** 如果下拉列表中没有管道名称，请先开启管道。具体操作，请参见开启管道。 |
| （可选）**消息通知** | 单击**开关按钮**，选择消息队列或消息主题，在下拉列表中选择一个队列通知或主题通知。<br>**说明** 如果下拉列表中没有队列通知或主题通知，请先开启消息通知。具体操作，请参见设置工作流消息通知。 |

iii. 单击**确定**，完成**输入**节点设置。

7. 添加**视频DNA**节点。

i. 单击**输入**节点右侧加号图标选择**视频DNA**。

ii. 单击**视频DNA**节点右侧笔形图标。



iii. 设置**视频DNA**参数。

| 参数 | 说明 |
| --- | --- |
| **视频DNA管道** | 在下拉列表中选择管道。<br>说明 如果**视频DNA管道**下拉列表中没有管道名称，请先开启视频DNA专用管道。具体操作，请参见开启管道。 |
| **入库规则** | ■ **仅入库不重复内容**：表示DNA库中只保存不重复的视频的DNA，重复视频将不会对其DNA进行入库操作。<br>■ **所有视频均不入库**：表示对视频只做比对并不需要保留视频DNA。 |
| **是否终止工作流** | 在选项中，根据实际需求进行选择。 |

iv. 单击**确定**，完成**视频DNA**节点设置。

8. 设置**发布**节点。

i. 单击**发布**节点右侧笔形图标。

ii. 设置发布。

| 参数 | 说明 |
| --- | --- |
| **媒体发布类型** | 选择**媒体发布类型**为**自动**。<br>■ **手动**发布：工作流执行成功后，将媒体设置为未发布状态，之后您可以手动发布该媒体。手动发布媒体文件，请参见媒体列表。<br>■ **自动**发布：工作流执行成功后，将媒体设置为发布状态。<br>说明<br>■ 发布含义：设置媒体所有播放资源、截图文件的访问权限继承所在Bucket的访问权限。<br>■ 不发布含义：媒体所有播放资源、截图文件的访问权限为私有。 |

iii. 单击**确定**，完成发布节点设置。

9. 单击**保存**，完成工作流创建。
创建完成后，当有符合条件的新文件进入指定路径时，会自动触发工作流执行。

## 5.2.5. 智能标签

智能标签，是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，适用于媒资检索、个性化推荐、智能广告投放等场景。

## 使用限制

智能标签功能，目前开通的地域为华北2（北京）、华东2（上海）、华东1（杭州），其他地域暂不支持。

## 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。

## 使用方法

智能标签功能可以通过控制台创建智能标签任务触发。

> **说明** 如您需要通过API创建智能标签任务，请参考智能标签操作。

## 文件类型说明

智能标签可以处理的文件类型和格式如下：

| 视频 | 音频 | 图片 |
| --- | --- | --- |
| 视频格式：avi、flv、mkv、mpg、mp4、ts、mov、mxf | 音频格式：mp3、wav | 图片格式：jpg、jpeg、png |
| 编码格式：MPEG2,、MPEG-4、H.264、H.265/HEVC | 不涉及 | 不涉及 |
| 视频时长：≤4小时 | 音频时长：≤4小时 | 不涉及 |
| 视频大小：≤4 GB | 音频大小：≤400 MB | 图片大小：≤4 MB |
| 视频分辨率：240P~2160P，为保证分析效果，建议分辨率≥720P。 | 不涉及 | 图片分辨率：≤2160P |

## 使用前准备

在使用前需要进行一些设置确保功能能够正常使用，如已设置请忽略。

- 参考前置条件完成所需设置。
- 媒体处理操作的媒体资源均存放在OSS的Bucket中，需要添加媒体Bucket才可以在任务、工作流中选择对应的Bucket作为输入和输出路径。具体操作，请参见添加媒体Bucket。
- 开启管道。如果管道列表中没有智能标签管道，请先提交工单开通智能标签管道。
- 创建智能标签模板。
- （可选）如您需要在工作完成后接收消息通知，请先配置消息通知服务。具体操作，请参见设置转码消息

通知。

## 创建智能标签单任务

1. 登录媒体处理控制台。
2. 在顶部菜单栏左侧选择地域。
3. 在左侧导航栏，选择**任务管理**。
4. 选择**智能标签**，单击**创建任务**。
5. 设置参数。

| 参数 | 说明 |
| --- | --- |
| **待处理文件地址** | 单击**选择**，在**Bucket**下拉列表中，选择Bucket名称。**路径**下方会显示对应Bucket已经创建好的文件夹，在文件夹下选择一个地址作为输入路径。<br>**说明** **待转码文件地址**为OSS Bucket输入存储路径，如果下拉列表中没有Bucket，请先添加Bucket。具体操作，请参见添加媒体Bucket。 |
| **智能标签管道** | 在下拉列表中选择管道名称。<br>**说明** 如果智能标签管道下拉列表中没有管道名称，请先开启智能标签专用管道。详细操作，请参见开启管道 。 |
| **标题** | 单击输入框，输入文件标题，标题内容会参与多模态识别进行视频分类分析。 |
| **模板** | 在下拉列表中选择模板名称。<br>**说明** 如果智能标签模板下拉列表中没有模板名称，请先创建智能标签模板。具体操作，请参见智能标签模板。 |
| **返回OCR文字识别原始结果** | 单击**开关按钮**，切换是否返回文字识别结果。开启后，标签任务可返回OCR文字识别原始结果。 |
| **返回ASR语音识别原始结果** | 单击**开关按钮**，切换是否返回语音识别结果。开启后，标签任务可返回ASR语音识别原始结果。 |

6. 单击提交任务，完成创建智能标签任务。

   任务提交后您可以在当前页签下方列表中查看任务的执行状态以及配置信息和输入、输出地址等。

# 5.3. 视频加密

# 5.3.1. 视频加密概述

视频加密是对视频内容保护的一种手段，对视频中的内容进行加密，可有效防止视频泄露和盗链问题，广泛用于在线教育及财经等领域。

## 使用场景

适用于在线教育、PGC等需要版权保护的场景，防止非法的下载及传播。

## 方案对比

媒体处理提供两种加密方案：阿里云私有加密、HLS标准加密。

| 加密方案 | 阿里云私有加密 | HLS标准加密 |
| --- | --- | --- |
| 描述 | MPS会对视频内容进行私有的AES128加密。 | MPS对视频内容进行HLS协议中规定的加密。 |
| 安全程度 | 高。 | 低。需要业务方进行密钥的保护，如登录cookie、refer限制等。 |
| 准备 | • 开通KMS，创建及管理加密密钥。<br>• 开通CDN，播放分发视频。 | 需业务方自己准备加密密钥。 |
| 触发方式 | 自动触发。需使用媒体工作流进行转码，创建媒体，以媒体为单位进行编码、加密、播放。 | 手动触发。 |
| 使用方式 | 1. 创建媒体工作流，开启加密。<br>2. 配置CDN。<br>3. 上传视频。<br>4. 使用MPS提供的SDK进行播放。<br>完整的操作步骤，请参见阿里云私有加密。 | 1. 提交转码任务并配置标准加密相关参数。详细步骤请参见HLS数据加密。<br>2. 通过URL进行播放。 |
| 支持输出格式 | M3U8 | M3U8 |
| 播放限制 | • 必须使用阿里云媒体处理（MPS）提供的播放（iOS、Android、Flash）SDK进行播放，H5不支持加密视频播放。<br>• 必须使用阿里云CDN进行视频分发。 | 支持HLS规定的加密播放即可。 |

# 5.3.2. 阿里云私有加密

视频加密是对视频内容保护的一种手段。对视频中的内容进行加密，可有效防止视频泄露和盗链问题，广泛用于在线教育及财经等领域。阿里云目前支持两种加密方式：阿里云私有加密和HLS标准加密。本文为您介绍阿里云私有加密的操作步骤。

## 使用说明

媒体处理私有加密功能已迁移至视频点播进行使用。如果您之前未接入阿里云私有加密，接入步骤请参见阿里云视频加密。

## 使用限制

- 阿里云私有加密目前只支持M3U8的输出格式。
- 阿里云私有加密目前只支持中国内地地域使用，中国香港、海外地域暂不支持此功能。

## 使用方法

阿里云私有加密可以通过工作流或API触发。由用户托管，可以自动触发。

## 使用准备

- 开通KMS（密钥管理服务）。
- 开通CDN（内容分发服务）。

## 设置视频加密

1. 创建用于视频加密的转码工作流。

   创建转码工作流操作请参见创建转码工作流。创建过程中必须按照以下步骤设置视频加密相关信息：

   i. 在**转码**节点的**基础配置**页面，单击**高级设置**。

   ii. 单击**加密**开关。

   iii. 选择**阿里云私有加密**。

| 加密方式 | 说明 |
| --- | --- |
| **阿里云私有加密** | 对视频内容进行私有的AES128加密，只能通过阿里云（iOS、Android、Flash）播放器解码播放。安全程度高。 |

   iv. 其他参数按需设置，设置完成后单击**确定**。

   v. 单击**发布**节点笔形图标。

   vi. **媒体发布类型**选择**自动**。转码完成后，可立即播放。

2. 配置输出Bucket的CDN域名。具体操作，请参见配置加速域名。

## 上传视频

配置好视频加密功能后，即可上传视频。上传完成后，会自动进行转码并对视频内容进行加密处理。

1. 通过媒体处理控制台或OSS上传视频。具体操作，请参见上传视频。
2. 上传完成后视频为**初始**状态。

> **说明** 工作流媒体发布类型设置为自动才有初始状态，设置为手动则上传完成后状态为未发布，加密工作流状态设置为自动。

3. 转码完成后自动发布，发布之后**媒体状态**变为**已发布**。此时您可以进行播放。

## 视频播放

阿里云私有加密视频必须使用阿里云播放器播放，加密视频播放的具体操作请参见播放器SDK。

## 相关文档

- HLS标准加密

# 5.3.3. HLS标准加密

视频加密是对视频内容保护的一种手段。对视频中的内容进行加密，可有效防止视频泄露和盗链问题，广泛用于在线教育及财经等领域。阿里云目前支持两种加密方式：阿里云私有加密和HLS标准加密。本文为您介绍通过控制台进行HLS标准加密的操作步骤。

## 使用限制

HLS标准加密目前只支持M3U8的输出格式。

## 使用方法

用户在控制台创建了HLS标准加密工作流，创建成功后上传视频至加密工作流对应的输入路径，自动触发HLS加密转码。

## 接入准备

接入媒体处理的HLS加密功能需要完成以下准备：

1. 开通相关阿里云服务。

   如果您尚未开通MPS，OSS，KMS， RAM，CDN服务，请先开通服务。

   i. 开通MPS服务。具体操作，请参见开通MPS服务。

   ii. 开通OSS服务。具体操作，请参见开通OSS服务。

   iii. 开通KMS服务。具体操作，请参见开通KMS服务。

   iv. 开通RAM服务并授权。具体操作，请参见开通RAM服务并授权。

   v. 开通CDN服务。具体操作，请参见开通CDN服务。

2. 授权媒体处理访问KMS。
   i. 登录RAM访问控制台。
   ii. 单击**授权**进入**新增授权**页面。
   iii. 在**授权主体**搜索框搜索*AliyunMtsDefaultRole*，选择系统创建的可供媒体处理使用的角色。
   iv. 在**选择权限**下方的搜索框搜索*KMS*并选择*AliyunKMSFullAccess*，然后单击**确定**。

   完成授权后，媒体处理可以访问您的KMS服务。收到加密视频请求时，媒体处理调取KMS接口获取数据密钥加密视频。
3. 配置OSS输出Bucket域名及回源Host。具体操作，请参见配置加速域名。如已配置请忽略。

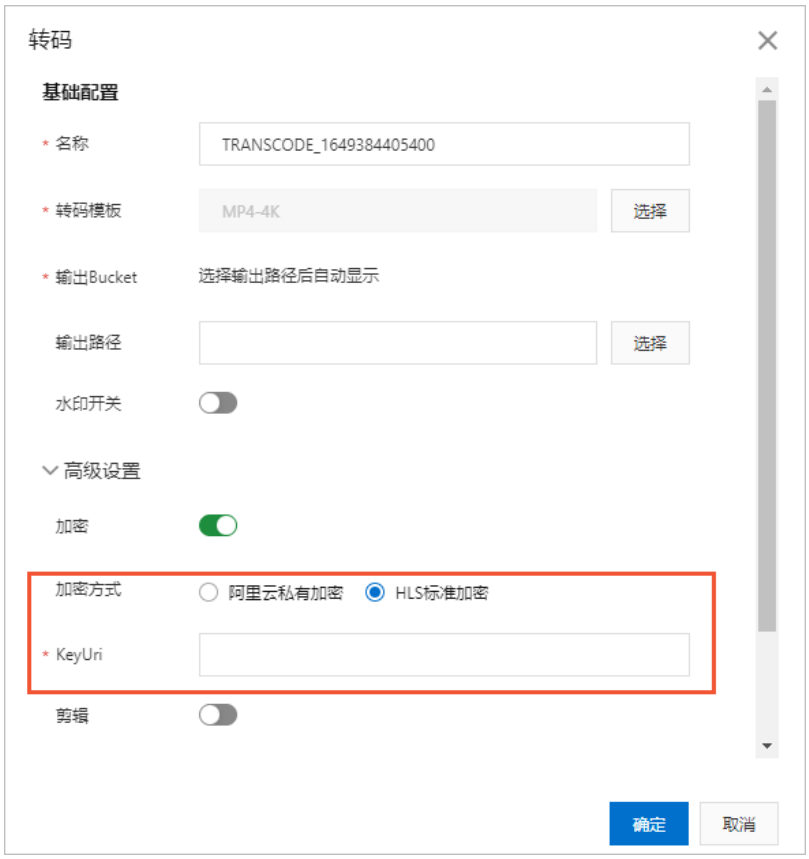
   > **说明** OSS域名可以手动输入阿里云OSS Bucket的外网域名（OSS外网域名可前往OSS控制台查看），如：`exampleBucket****.oss-cn-hangzhou.aliyuncs.com`，也可以直接选择同账号下需要加速的OSS Bucket。不支持OSS内网域名。

## 创建HLS标准加密工作流

1. 创建HLS标准加密工作流。

   创建转码工作流操作请参见创建转码工作流。创建过程中必须按照以下步骤设置视频加密相关信息：
   i. 在**转码**节点的**基础配置**页面，单击**高级设置**。
   ii. 单击**加密**开关。

iii. 选择**HLS标准加密**并填写您提前准备好的**KeyUri**。




| 加密方式 | 说明 |
| --- | --- |
| **HLS标准加密** | 对视频内容进行HLS协议规定中的加密，需要业务方进行密钥的保护，如登录cookie、refer限制等。安全程度低。如选择此项需要您输入准备好的KeyUri。<br>注意 KeyUri是业务方的服务地址。 |

iv. 其他参数按需设置，设置完成后单击**确定**。

v. 单击**发布**节点笔形图标。

vi. **媒体发布类型**选择**自动**。转码完成后，可立即播放。

2. 上传视频至工作流指定的输入路径触发加密转码 。具体操作请参见上传视频。

   工作流设置并保存完成后，指定路径新增视频文件会自动触发加密转码。如需对存量视频进行加密处理，则需要您在提交转码任务时指定工作流，才会触发工作流执行。具体操作请参见HLS标准加密。

## 上传视频

配置好视频加密功能后，即可上传视频。上传完成后，会自动进行转码并对视频内容进行加密处理。

1. 通过媒体处理控制台或OSS上传视频。具体操作，请参见上传视频。
2. 上传完成后视频为**初始**状态。

> **说明** 工作流媒体发布类型设置为自动才有初始状态，设置为手动则上传完成后状态为未发布，加密工作流状态设置为自动。

3. 转码完成后自动发布，发布之后**媒体状态**变为**已发布**。此时您可以进行播放。

## 视频播放

HLS标准加密可以使用阿里云播放器或其他播放器播放，播放加密视频的具体操作请参见HLS视频加密与播放。

## 相关文档

- 阿里云私有加密

# 6.API用户路径

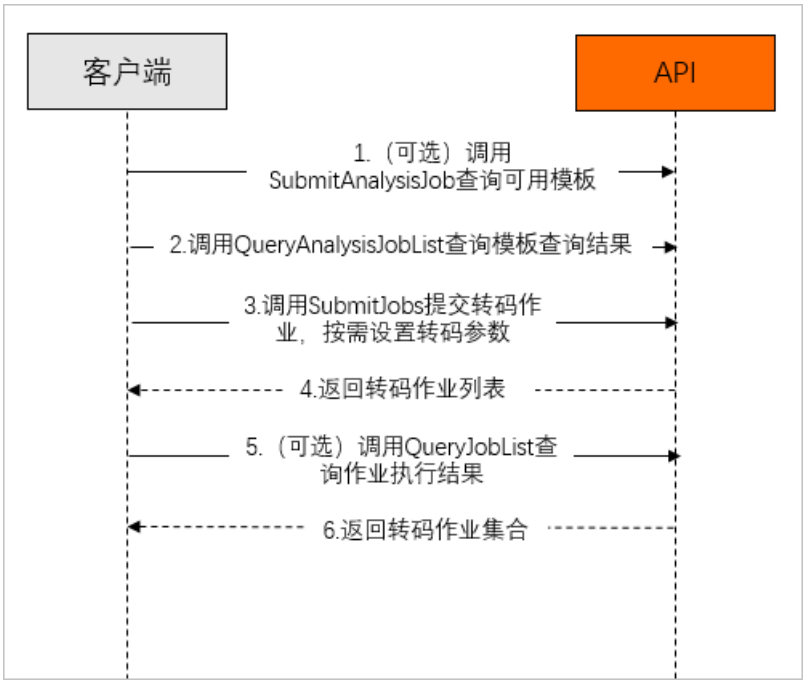
## 6.1. API用户路径概述

本文为您介绍通过API使用媒体处理服务的常用操作流程。

### 功能概览

- 媒资转码

  工作流无法满足用户场景时，需用户自己判断业务逻辑，使用API提交转码任务。例如：并不是所有的视频都需要转码，不同视频需要设置不同的转码配置。




- 视频拼接剪辑

  > **说明** 视频拼接剪辑功能目前只支持通过API使用，控制台暂不支持。

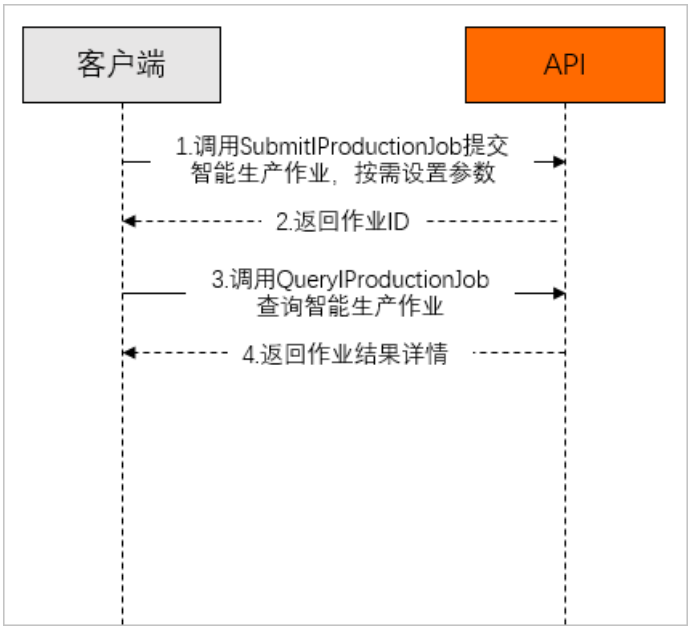
  拼接是把多个不同格式 、不同编码、分辨率的视频拼接在一起，输出成一个格式、编码、分辨率相同的新视频。常用于添加固定的片头和片尾、直播录制视频拼接。剪辑是指裁剪视频的某一段，输出成一个新视频。常用于截取视频中精彩或关键的内容。

- 视频AI

  视频AI包含视频生产、媒体审核、视频DNA、智能标签等，可以对音视频内容进行识别、分析和理解。

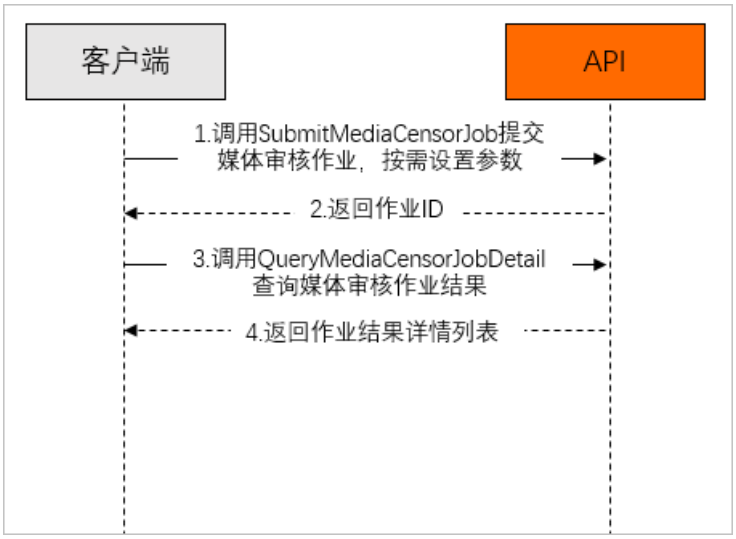
- 视频生产

  视频生产服务基于媒体AI技术，提供多种形式媒体内容处理及内容生成能力，支持智能封面、智能横转竖、绿幕抠图、人像抠图、智能图标模糊、智能去字幕、字幕提取、副歌检测、音乐节奏检测、PPT拆条等多种媒体处理生成功能，提升媒体内容生产效率和质量。




- 媒体审核

  媒体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媒体文件中包含的违禁内容，包括色情、暴恐、政治敏感、广告识别等多个功能模块，同时支持视频图像文件内容质量审核功能，应用于短视频平台、直播平台、传媒审核等场景。

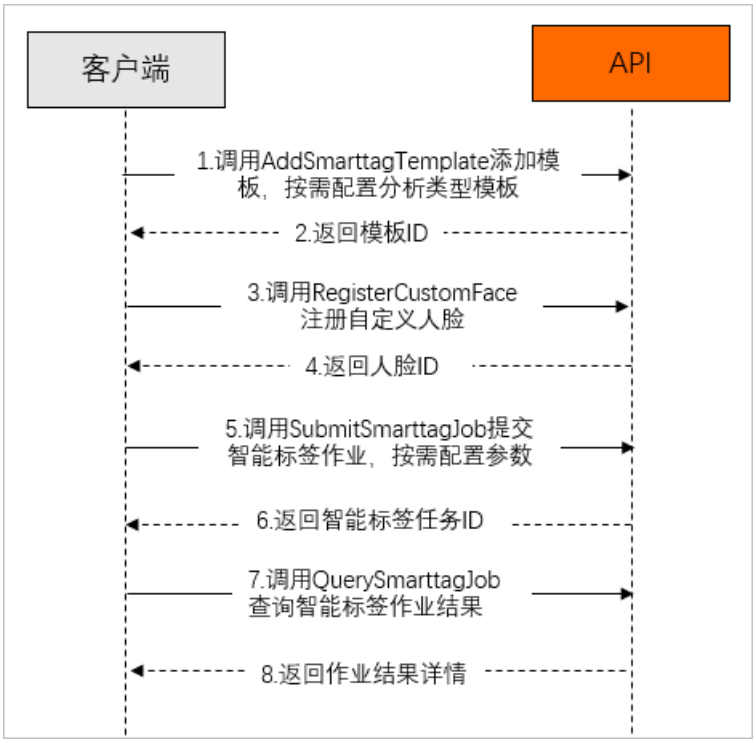


- 视频DNA

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

- 智能标签

  智能标签，是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，适用于媒资检索、个性化推荐、智能广告投放等场景。



- 视频加密

> **说明** 阿里云私有加密只支持控制台开启，自动触发。HLS标准加密支持控制台或API开启。控制台开启后需要手动指定工作流后触发，不建议通过控制台开启。

视频加密是对视频内容保护的一种手段，对视频中的内容进行加密，可有效防止视频泄露和盗链问题，广泛用于在线教育及财经等领域。

## 使用前准备

1. 开通服务。
   - 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。
   - 您已经开通媒体处理服务。具体操作，请参见开通服务。
2. 账号准备。

   > **说明** 阿里云账号AeecssKey是您访问阿里云API的密钥，具有该账户完全的权限，请您务必妥善保管！不要通过任何方式（如Github）将AccessKey公开到外部渠道，以避免被他人利用而造成安全威胁。强烈建议您遵循阿里云安全最佳实践，使用RAM用户（而不是阿里云账号）的AccessKey进行API调用。

   您已经准备好访问媒体处理服务使用的AccessKey。获取方法，请参见获取AccessKey。

   > **说明** 推荐您使用RAM用户AccessKey，为RAM用户授权请参见申请RAM用户权限。

3. （可选）如需接收任务状态回调通知，则需要在管道中配置消息服务MNS。具体操作，请参见设置转码消息通知。
4. 上传需要处理的音视频文件，具体操作，请参见上传视频。
5. 在使用API之前建议您参考请求结构了解API调用的规则等信息，帮助您更好的使用媒体处理API。

## 功能使用

- 媒资转码
- 视频拼接剪辑
- 视频AI
- HLS标准加密

# 6.2. 媒资转码

当控制台创建的转码任务和工作流均无法满足您的使用场景时，可以按需调用SubmitJobs接口进行转码。根据您的业务逻辑配置所需的转码参数即可。

## 使用限制

- API转码支持HLS-AES128标准加密，暂不支持阿里云私有加密。
- API转码支持URL播放，不支持媒体ID播放。
- 如您需要适应不同终端和网络环境播放，需自己关联多个格式的多个清晰度输出，实现多清晰度自动切换、多格式支持等逻辑。

## 转码准备

在进行转码任务之前，需要进行一些设置确保功能能够正常使用，如已设置请忽略。

- 参考前置条件完成所需设置。
- 媒体处理操作的媒体资源均存放在OSS的Bucket中，需要添加媒体Bucket才可以在任务、工作流中选择对应的Bucket作为输入和输出路径。详细操作请参见添加媒体Bucket。
- 开启管道。如果管道列表中没有您需要的管道类型，请先提交工单开通所需类型的管道。
- 将需要处理的视频上传至OSS。详细操作请参见上传视频。
- （可选）如预置转码模板不能满足您的实际转码需要，您可以自定义转码模板。详细操作请参见新增自定义转码模板。
- （可选）如您有添加水印的需求，请先创建水印模板。详细操作请参见创建水印模板。
- （可选）如您需要在作业完成后接收消息通知，请先配置消息通知服务。详细操作请参见设置转码消息通知。

## 操作步骤

1. 将需要转码的媒资文件上传至OSS。您可以选择控制台上传、使用OSS相关上传工具上传等方法进行上传文件。
2. 调用SubmitJobs接口提交转码作业。

   详细操作请参见提交转码作业，创建过程中必须按照以下步骤设置转码相关信息：

   > **说明**
   > - 一个转码作业生成一个输出文件，允许批量提交作业。
   > - 媒体处理中转码需要重新编码，不保证转码完成后码率与源文件一致，如果需要文件流码率不变，可以使用转封装。转封装模板详细信息，请参见预置模板详情。

   i. 选择所需管道。

   > **说明** 倍速转码依赖管道，如需进行倍速转码请先开启专用管道。开启管道请参见开启管道。

| 场景 | 参数设置 |
| --- | --- |
| 倍速转码 | 将PipelineId参数设置为倍速转码管道ID。您可以通过媒体处理控制台或调用搜索管道接口获取倍速转码管道ID。 |

ii. 根据您的实际转码需要选择转码模板，如系统预置模板不能满足您的转码需求，您可以自定义转码模板。自定义转码模板请参见新增自定义转码模板。模板参数设置请参见下表：

| 场景 | 参数设置 |
| --- | --- |
| 窄带高清1.0转码 | 将Output参数中的TemplateId根据您的实际转码需要设置为自定义窄带高清TM1.0模板ID或预置窄带高清TM模板ID。自定义窄带高清TM1.0模板ID在**转码模板**页面获取，预置窄带高清模板ID信息请参见窄带高清TM预置模板。 |
| 窄带高清2.0转码 | 将Output参数中的TemplateId根据您的实际转码需要设置为自定义窄带高清TM2.0模板ID或预置窄带高清TM2.0模板ID。自定义窄带高清TM2.0模板ID在转码模板页面获取，预置窄带高清TM2.0模板ID信息请参见窄带高清TM预置模板。<br>**说明** 触发窄带高清TM2.0转码任务之前需要先开启窄带高清TM2.0管道，详细步骤请参见开启管道。 |
| 分辨率倍增转码 | 将Output参数中的TemplateId设置为：<br>■ S00000003-400040（标清转高清）。<br>■ S00000003-400070（2K转4K）。 |

iii. （可选）选择所需水印模板。

iv. （可选）如有加密需求，可按需配置Output中的Encryption参数。

> **说明** 数据加密完成后只支持m3u8格式的输出。API加密支持base64和KMS加密方式。详情参见参数详情。

3. 接收到消息通知，调用查询转码作业接口查询作业执行情况，获取输出文件URL。
4. （可选）如您需要查看管道中的所有转码作业，可以调用列出转码作业接口，通过作业状态，创建时间区间，列出转码作业。

# 6.3. 视频拼接剪辑

媒体处理支持对视频进行剪辑、拼接等二次创作，通过调用SubmitJobs接口提交转码作业时设置相关参数实现。本文为您介绍视频拼接和视频剪辑涉及到的具体参数及配置方法。

## 视频拼接

拼接是把多个不同格式 、编码、分辨率的视频拼接在一起，输出成一个格式、编码、分辨率相同的新视频。常用于添加固定的片头和片尾、直播录制视频拼接。

在进行视频拼接时，您需要关注以下参数：

> **说明** 视频拼接、视频剪辑均在提交转码作业时设置，更多接口信息请参见提交转码作业。

- Input（输入文件）

  设置片头视频的OSS输入文件。

  > **说明** OSS的Location必须和MPS的地域对应。例如，OSS的 `oss-cn-hangzhou` 对应媒体处理的 `mts.cn-hangzhou` 。

- Output（输出文件）

  在输出参数中，您需要关注以下参数：

| 参数 | 描述 |
| --- | --- |
| Video | 设置输出最终视频的宽、高、码率等。如果多个拼接视频（包括片头、片尾）的宽、高比和最终输出的不一致，会自动填充黑边。建议您根据不同业务的分辨率实际情况，准备几个不同宽、高比的片头、片尾视频，以达到最好的效果。 |
| MergeList | 拼接设置。列表的顺序代表了拼接的顺序，所以列表的最后一个元素是片尾，最多支持5个（包含片头、片尾）视频拼接在一起。如果您需要拼接更多视频，请使用MergeConfigUrl参数。<br>**说明** MergeList和MergeConfigUrl不支持同时设置，您只能选择其中一个进行设置。<br>每个拼接视频都包含3个参数：<br>○ MergeURL<br>设置拼接视频的OSS URL地址。<br>**说明** 拼接视频的OSS地域必须和片头一致，不支持跨地域视频的拼接。<br>○ Start<br>拼接视频时，如果您期望值截取部分内容输出到最终视频，可以设置截取的开始时间点。默认值：0。<br>○ Duration<br>拼接视频时，如果您期望值截取部分内容输出到最终视频，可以设置相对于开始时间点（Start）的截取时长。默认从开始时间点（Start）到结尾的全部内容。 |

| 参数 | 描述 |
| --- | --- |
| MergeConfigUrl | 设置拼接视频的配置文件的OSS URL地址。文件的内容就是一个JSON对象，和MergeList参数的值完全一样。<br>说明<br>○ MergeList和MergeConfigUrl不支持同时设置，您只能选择其中一个进行设置。<br>○ 列表的顺序代表了拼接顺序，所以列表的最后一个元素是片尾，最多支持100个（包含片头、片尾）视频拼接在一起。 |

## 视频剪辑

在视频剪辑中，您需要关注以下参数：

- Input（输入文件）

  设置待剪辑视频的OSS输入文件。

  说明 OSS的Location必须和媒体处理服务的地域对应。例如，OSS的 `oss-cn-hangzhou` 对应媒体处理的 `mts.cn-hangzhou` 。

- Output（输出参数）

  在输出参数中，您需要关注以下参数：

| 参数 | 描述 |
| --- | --- |
| TimeSpan | 剪辑的时间区间。您可以根据实际需要设置不同的时间节点和剪辑时长。<br>每个时间区间包含3个参数：<br>○ Seek：剪辑开始的时间点。<br>○ Duration：剪辑持续时长。<br>○ End：截尾时长，表示切掉尾部的若干时长。<br>说明 设置End时，参数Duration失效。 |
| ConfigToClipFirstPart | 是否剪辑第一片。可设置的值为false（拼接完后剪辑）、true（先剪辑第一片后拼接）。默认值为false。 |

## 示例代码

例如一个720P（1280×720）的正片视频，拼接上片头片尾是480P（640×480）的MP4视频，输出分辨率时1280×720.所以在播放输出视频时，片头和片尾会出现左右黑边，正片视频显示正常。

具体的代码示例如下：

| 语言 | 相关文档 |
| --- | --- |
| Java | 视频拼接和简单剪辑Java SDK |
| Python | 视频拼接和简单剪辑Python SDK |
| PHP | 视频拼接和简单剪辑PHP SDK |
| Node.js | 视频拼接和简单剪辑Node.js SDK |

## 常见问题

拼接后的视频在部分iOS设备上播放会出现没有声音的问题，是由于iOS解码能力较差，需要适当调低质量才能正常播放。可以在提交转码任务时将编码级别（Profile）设置为main后重新编码即可。

# 6.4. 视频AI

媒体处理提供智能生产、媒体审核、视频DNA、智能标签等视频AI服务。本文为您介绍通过API实现上述功能的操作步骤。

## 转码准备

在进行任务之前，需要进行一些设置确保功能能够正常使用，如已设置请忽略。

- 参考前置条件完成所需设置。
- 媒体处理操作的媒体资源均存放在OSS的Bucket中，需要添加媒体Bucket才可以在任务、工作流中选择对应的Bucket作为输入和输出路径。详细操作请参见添加媒体Bucket。
- 开启管道。如果管道列表中没有您需要的管道类型，请先提交工单开通所需类型的管道。
- 将需要处理的视频上传至OSS。详细操作请参见上传视频。
- （可选）如您需要在作业完成后接收消息通知，请先配置消息通知服务。详细操作请参见设置转码消息通知。

## 视频生产

视频生产服务基于媒体AI技术，提供多种形式媒体内容处理及内容生成能力，支持智能封面、智能横转竖、绿幕抠图、人像抠图、智能图标模糊、智能去字幕、字幕提取、副歌检测、音乐节奏检测、PPT拆条等多种媒体处理生成功能，提升媒体内容生产效率和质量。

1. 提交智能生产作业。

   已成功上传至OSS的视频、图片、语音文件等，可以通过SubmitIProductionJob接口提交智能生产作业，通过指定FunctionName进行各项服务能力调用。详细参数信息请参见提交智能生产作业。

2. 接收回调信息。

   完成智能生产作业后，如果配置了管道MNS消息队列事件通知，会向消息服务指定的队列或主题发送消息。详细信息请参见接收消息通知。

3. 查询作业结果。

   您可以通过QueryIProductionJob接口查询智能生产作业的状态和结果，根据智能生产任务ID进行查询。详细参数信息请参见查询智能生产作业。

## 媒体审核

媒体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媒体文件中包含的违禁内容，包括色情、暴恐、政治敏感、广告识别等多个功能模块，同时支持视频图像文件内容质量审核功能，应用于短视频平台、直播平台、传媒审核等场景。

1. 提交媒体审核作业。

   已成功上传到OSS的视频、图片、语音文件，可以通过SubmitMediaCensorJob接口提交智能媒体审核作业。详细参数信息请参见提交媒体审核作业。

   > **说明** 媒体审核的标准可配置支持不同审核策略，例如色情审核可配置高准确率、高召回率、严格色情忽略性感审核策略，涉政暴恐可配置具体管控场景等，如需调整审核标准可提交工单并提供UID、地域及详细的需求描述进行配置。

2. 接收回调消息。

   完成智能媒体审核作业后，如果配置了管道MNS消息队列事件通知，会向消息服务指定的队列或主题发送消息。详细信息请参见接收消息通知。

3. 查询作业结果。

   您可以通过QueryMediaCensorJobDetail接口查询媒体审核作业的状态结果，根据媒体审核任务ID进行查询。详细参数信息请参见查询媒体审核作业详情。

## 视频DNA

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

1. 建立视频DNA库。

   提交新建视频DNA库任务，返回新建DNA库信息，待DNA库新建完成后DNA库状态会更新为active。提交任务的时候需要绑定DNA库。详细参数信息请参见提交新建DNA库。

   > **说明** 视频DNA库是用于记录和存放视频DNA的库。视频DNA服务可以支持：
   > - 按照用户（UID）维度进行视频DNA库的建立。
   > - 多个用户（UID）共用一个视频DNA库。
   > - 同一用户（UID）根据不同业务创建多个视频DNA库。

2. 提交增量视频DNA任务。
   - 如果您的视频存量少于100万分钟，可以直接通过提交视频DNA作业接口提交任务。详情参见提交DNA作业。
   - 如果您在实时情况下新增视频，也可以通过提交DNA作业接口进行处理。增量和存量可以一起通过该接口提交任务，完成视频DNA的处理。也可以待存量视频导入完成后，再通过提交DNA作业接口提交任务。

3. 视频处理结果查询。

   对于通过增量处理接口提交的视频，视频DNA的处理结果，您可以通过查询DNA作业或回调接口DNA作业结果反馈。

   处理结果中包含以下主要信息：
   - 作业是否有匹配到重复视频。
   - 匹配到重复视频，输出重复视频信息。
   - 视频之间重复的时码区间。

- 视频的重复程度（重复度为介于0~1之间的数据）。

## 智能标签

智能标签，是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，适用于媒资检索、个性化推荐、智能广告投放等场景。

1. 通过模板配置分析类型。

   智能标签作业可以通过模板设置分析类型，在提交作业时指定模板ID。模板ID可以通过添加模板获取。

2. 自定义人脸库设置。

   智能标签作业支持配置分析使用的公共人脸库或自定义人脸库范围，在提交作业时指定人脸库参数。自定义人脸库的具体操作，请参见注册自定义人脸。

3. 提交智能标签作业。

   对于已成功上传到媒体库的视频，可以通过API/SDK方式提交智能标签作业，详细参数信息请参见提交智能标签作业、智能标签SDK。

4. 接收回调信息。

   智能标签作业完成后，如果配置了管道MNS消息队列事件通知，会向消息服务指定的队列或主题发送消息。详细信息请参见接收消息通知。

5. 查询作业结果。

   通过API/SDK方式查询智能标签作业的状态结果，按照智能标签任务ID进行查询。详细参数信息请参见查询智能标签作业。

# 6.5. 数字水印

阿里云数字水印产品基于数字水印技术，支持对视频、图片进行数字信息的嵌入、提取。嵌入的水印信息隐藏于载体文件中，水印嵌入后不影响视频画面、图片的视觉质量及完整性，可用于版权保护、泄露溯源、广播监视、信息传输、文件验真等多种场景。

## 背景信息

数字水印是将信息隐藏到载体中的技术，人们难以感知水印嵌入后载体的变化，也无法直接获取水印信息。数字水印嵌入后，载体可以正常使用，相比于二维码等可见、可感知的信息传递方案，数字水印的隐蔽性更好，更美观，而且更安全。

## 产品功能

- 视频版权水印

  支持对视频版权水印嵌入及提取，可用于版权确认、机构版权溯源。

- 视频溯源水印

  支持视频溯源水印嵌入及提取。通过AB流方式，在用户请求视频播放时，实时生成与用户身份唯一对应的溯源水印切片文件，可用于视频点播用户溯源。

## 计费说明

数字水印功能目前公测中（公测至2022年5月9日晚24点），公测期间不收费，公测结束后按照数字水印定价收费。详细计费说明，请参见数字水印计费。

## 应用场景

使用数字水印技术，可有效对抗干扰手段，确认盗版侵权行为，实现对视频图片的版权内容保护，促进内容运营收费，保护从内容商到运营商的利益。

- 版权保护：添加版权方水印，版权归属确认

  版权视频通过嵌入版权方唯一识别ID，对版权视频打上不可见标签，通过提取和识别工具来获取ID及确认版权归属。

- 机构溯源水印：版权分发环节溯源，机构盗版泄漏权责追溯

  不同机构拿到嵌有独属标识水印信息的版权拷贝，当发生影片泄露时，只要检测其中的水印信息就可判断泄露影片的机构。

  > **说明** 机构溯源是一种溯源范围较小的版权溯源应用场景。例如：电影原片分发环节的溯源需求。电影发行方向各机构派发影片，机构的数量通常为几家到十几家不等。当影片发生泄露并被非法传播时，版权方需要知道影片是从哪家机构泄露的，进而封堵泄露源并追究相关人员的责任。

- 用户溯源水印： 播放环节溯源，用户盗版权责追溯

  采用AB流水印技术，通过对AB流切片的排列组合生成与用户关联的具有唯一性的视频流。当发生视频盗版时，只要检测其中的水印信息就可以追踪到泄露视频的用户，盗版权责追溯。

  > **说明** 视频泄露常见的一种方式是终端客户通过一些技术手段下载或录制，数字水印技术可进行用户溯源，查找盗版泄露是来自哪一个用户。

## 使用方法

媒体处理的数字水印相关功能目前仅支持通过API/SDK使用。

## 产品优势

- 效果美观：数字水印不影响载体的正常使用，并且难以感知水印信息的存在。
- 内容安全：水印信息无法被直接感知，难以对其做针对性攻击；嵌入的水印信息经过特殊编码，难以被篡改，更难被“移花接木”。
- 精准提取：经过特殊编码的水印信息能够保证提取时的准确性，水印技术提取的信息具备唯一性和准确性。
- 鲁棒性强：水印鲁棒性强，可抵抗各种常见类型盗版攻击。例如：视频压缩、旋转、分辨率转换、帧率转换、录屏、录制剪切和缩放等。
- 水印加密：经过特殊编码加密的水印信息难以被篡改，能够保证提取的准确性。

## 使用准备

1. 开通服务。
   - 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。
   - 您已经开通媒体处理服务。具体操作，请参见开通服务。
2. 账号准备。

> **说明** 阿里云账号AeecssKey是您访问阿里云API的密钥，具有该账户完全的权限，请您务必妥善保管！不要通过任何方式（如Github）将AccessKey公开到外部渠道，以避免被他人利用而造成安全威胁。强烈建议您遵循阿里云安全最佳实践，使用RAM用户（而不是阿里云账号）的AccessKey进行API调用。

您已经准备好访问媒体处理服务使用的AccessKey。获取方法，请参见获取AccessKey。

> **说明** 推荐您使用RAM用户AccessKey，为RAM用户授权请参见申请RAM用户权限。

3. （可选）如需接收任务状态回调通知，则需要在管道中配置消息服务MNS。具体操作，请参见设置转码消息通知。
4. 调用数字水印服务的音视频文件，需要先上传视频文件。具体操作，请参见上传视频。媒体库相关说明请参见媒体库简介。

## 功能使用

- 视频版权水印

  对于已成功上传到媒体库的视频，可以通过API/SDK方式提交智能标签作业，更多信息，请参见提交视频版权水印、提交版权水印提取作业。

- 视频溯源水印

  对于已成功上传到媒体库的视频，可以通过API/SDK方式提交用户溯源水印作业，更多信息，请参见提交视频溯源水印ab流、提交视频溯源水印m3u8、提交溯源水印提取作业。

# 6.6. HLS标准加密

视频加密是指对视频中的内容进行加密，可有效防止视频泄露和盗链问题，广泛用于在线教育及财经等领域。阿里云目前支持两种加密方式：阿里云私有加密（推荐）和HLS标准加密。本文介绍媒体处理HLS加密的原理和接入流程。

## 工作原理

媒体处理采用信封数据加密的方式加密视频。业务方调用阿里云密钥管理服务（KMS）生成数据密钥（DK）和信封数据密钥（EDK），然后利用数据密钥（DK）加密视频，并将加密后的文件和信封数据密钥（EDK）存储。播放器终端通过解密服务获取数据密钥（DK）请求解密播放视频。

> **说明**
> - 数据密钥（DK）也称明文密钥，生成后用于视频加密。
> - 信封数据密钥（EDK）也称密文密钥，是通过信封加密技术保密后的密文数据密钥。主要用于解密数据密钥，得到明文数据密钥。
> - HLS加密要求用户自己保护数据密钥（DK）。

媒体处理的视频加密流程如下：

1. 业务方开通阿里云媒体处理服务（MPS）、存储服务（OSS）、访问控制（RAM）、密钥管理服务（KMS）以及内容分发网络（CDN）（如未开通）。

   **说明**

   - OSS是阿里云提供的数据存储服务，媒体处理操作的媒体资源均存放在OSS的Bucket中。
   - RAM是阿里云提供的访问控制服务，业务方通过RAM授权MPS访问业务方KMS服务。
   - KMS是阿里云提供的安全管理服务，主要负责数据密钥的生产、加密、解密等工作。
   - CDN是阿里云提供的内容分发网络，在HLS加密流程中，CDN会动态修改M3U8文件中的解密URI并返回给播放端。

2. 业务方授权媒体处理访问刚开通的阿里云密钥管理服务（KMS）。

   **说明** 授权访问是为了媒体处理在加密视频阶段调用KMS的GenerateDataKey接口生成数据密钥（DK）和信封数据密钥（EDK）。

3. 业务方配置OSS输出Bucket域名为CDN加速域名，配置OSS域名CNAME及回源Host。
4. 业务方创建视频加密工作流，传入OSS输出Bucket、Key URI等关键信息。

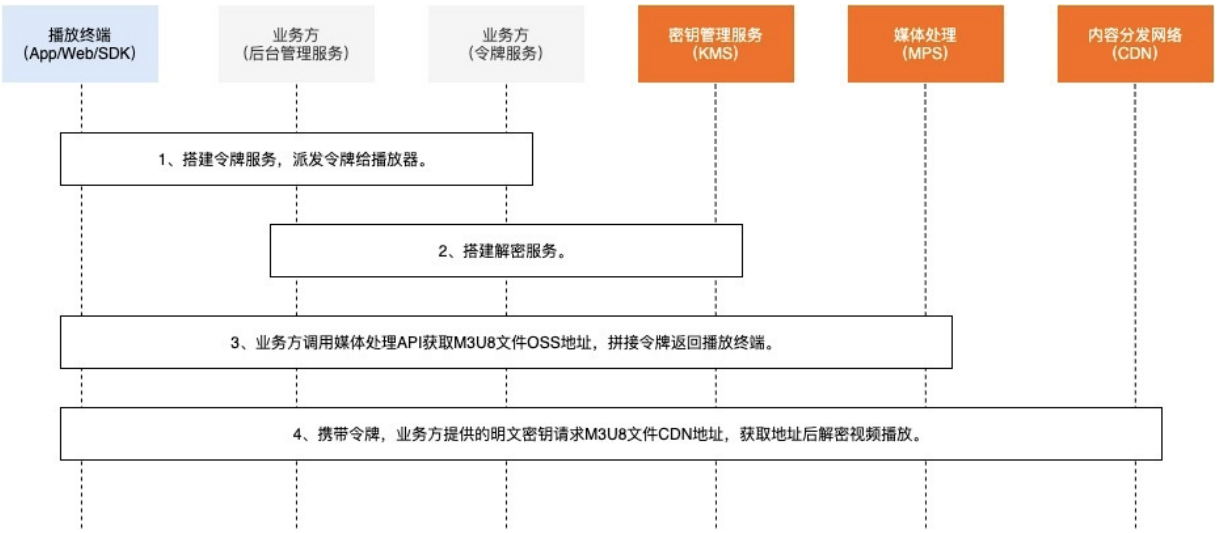
   Key URI是业务方的服务地址。媒体处理完成视频加密后的M3U8文件中会包含Key URI信息。

5. 业务方上传需要加密的视频，上传过程中指定创建好的视频加密工作流。
6. 视频上传完成后媒体处理自动触发加密转码。

   媒体处理调用GenerateDataKey接口生成数据密钥（DK）和信封数据密钥（EDK），并使用数据密钥（DK）加密视频。加密完成后，将业务方提供的Key URI与信封数据密钥（EDK）写入M3U8文件。

7. 媒体处理将M3U8文件及ts文件存入OSS的输出Bucket中。

播放端解密播放HLS加密视频的流程如下：




1. 业务方搭建令牌服务，用于颁发令牌MtsHlsUriToken。

   > **注意** 令牌服务指用于派发MtsHlsUriToken的服务。

2. 业务方调用KMS解密接口搭建解密服务，用于解密视频，同时提供数据密钥（DK）给播放终端。

   > **注意** KMS返回Base64加密后的数据密钥给业务方。业务方需要将调用KMS接口获得的数据密钥Base64 Decode之后返回给播放终端。

3. 业务方调用MPS的QueryMediaList接口获取视频M3U8文件的OSS地址，并将地址拼接MtsHlsUriToken后返回给播放终端。
4. 播放终端携带MtsHlsUriToken、数据密钥向阿里云CDN请求播放地址，阿里云CDN改写M3U8文件，将业务方的Key URI与信封加密密钥返回播放终端。播放终端解密播放视频。

## 业务方实现概览

在完整的HLS加密视频和解密播放流程中，需要业务方自行实现的代码逻辑包括：

- 创建加密工作流。

  > **说明** 尽管控制台也可以创建加密工作流，但为了更完整高效地使用HLS标准加密服务，建议您集成服务端SDK后创建。

- 搭建颁发及验证MtsHlsUriToken令牌服务，并校验解密令牌。推荐一个令牌只允许使用一次。
- 调用KMS服务的解密接口搭建解密服务，并将调用KMS接口获得的明文密钥Base64 Decode之后返回播放终端。

## 接入准备

接入媒体处理的HLS加密功能需要完成以下准备：

1. 开通相关阿里云服务。

   如果您尚未开通MPS，OSS，KMS， RAM，CDN服务，请先开通服务。

   i. 开通MPS服务。具体操作，请参见开通MPS服务。

   ii. 开通OSS服务。具体操作，请参见开通OSS服务。

iii. 开通KMS服务。具体操作，请参见开通KMS服务。

iv. 开通RAM服务并授权。具体操作，请参见开通RAM服务并授权。

v. 开通CDN服务。具体操作，请参见开通CDN服务。

2. 授权媒体处理访问KMS。

   i. 登录RAM访问控制台。

   ii. 单击**授权**进入**新增授权**页面。

   iii. 在**授权主体**搜索框搜索*AliyunMtsDefaultRole*，选择系统创建的可供媒体处理使用的角色。

   iv. 在**选择权限**下方的搜索框搜索*KMS*并选择*AliyunKMSFullAccess*，然后单击**确定**。

   完成授权后，媒体处理可以访问您的KMS服务。收到加密视频请求时，媒体处理调取KMS接口获取数据密钥加密视频。

3. 配置OSS输出Bucket域名及回源Host。具体操作，请参见配置加速域名。如已配置请忽略。

   > **说明** OSS域名可以手动输入阿里云OSS Bucket的外网域名（OSS外网域名可前往OSS控制台查看），如：`exampleBucket****.oss-cn-hangzhou.aliyuncs.com`，也可以直接选择同账号下需要加速的OSS Bucket。不支持OSS内网域名。

## 加密视频

请按以下指引完成视频加密：

1. 创建加密工作流。

   创建加密工作流需要集成阿里云SDK并添加媒体处理相关依赖。请根据您使用的开发语言选择查看对应代码示例。

   > **注意** 创建加密工作流时必须传入业务方的Key URI地址，媒体处理加密视频时将该地址写入M3U8文件存储到OSS。Key URI示例：`example.aliyundoc.com`。

| 语言 | 集成SDK | 加密工作流代码示例 |
| --- | --- | --- |
| Java | Java SDK安装 | 创建HLS标准加密工作流 |
| Python | Python SDK安装 | 创建HLS标准加密工作流 |
| PHP | PHP SDK安装 | 创建HLS标准加密工作流 |
| Node.js | Node.js SDK安装 | 创建HLS标准加密工作流 |

2. 上传视频触发加密转码。您可以通过媒体处理控制台或OSS控制台上传视频。具体操作请参见上传视频。

   > **说明** 由于加密工作流已经创建，上传视频时指定创建好的加密工作流即可自动触发媒体处理加密转码。

   加密完成后，可登录OSS控制台，在输出Bucket文件路径下查看M3U8文件。文件示例如下：

```
#EXTM3U
     #EXT-X-VERSION:3
     #EXT-X-TARGETDURATION:5
     #EXT-X-MEDIA-SEQUENCE:0
     #EXT-X-KEY:METHOD=AES-128,URI="https://example.aliyundoc.com?Ciphertext=aabbccdd
eeff&MediaId=fbbf98691ea44b7c82dd75c5bc8b****"
     #EXTINF:4.127544,
     15029611683170-00001.ts
     #EXT-X-ENDLIST
```

示例中，业务方配置的Key URI与媒体处理从KMS获取的信封数据密钥（EDK）包含在URI中。

## 播放HLS加密视频

请按以下指引完成视频解密播放：

1. 搭建令牌服务。

   说明　令牌服务需要根据您的加密逻辑自行搭建，以达到更高的视频安全等级。

2. 搭建解密服务。

   搭建一个本地HTTP服务，用于解密视频和获取解密密钥。媒体处理提供Java及Python代码示例。

   - Java代码示例

     Java SDK需要的依赖如下：

     - Java SDK Core
     - Java SDK KMS

     Java示例代码如下：

```
package com.aliyun.smallcode;
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 AuthorizationServer {
private static DefaultAcsClient client;
static {
```

```
String region = "";
String accessKeyId = "";
String accessKeySecret = "";
client = new DefaultAcsClient(DefaultProfile.getProfile(region, accessKeyId, accessKe
ySecret));
}
public class AuthorizationHandler implements HttpHandler {
public void handle(HttpExchange httpExchange) throws IOException {
String requestMethod = httpExchange.getRequestMethod();
if(requestMethod.equalsIgnoreCase("GET")){
//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);
}
private byte[] decrypt(String ciphertext) {
DecryptRequest request = new DecryptRequest();
request.setCiphertextBlob(ciphertext);
request.setProtocol(ProtocolType.HTTPS);
try {
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;
}
}
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");
return null;
}
```

```
}
}
private void startService() throws IOException {
HttpServerProvider provider = HttpServerProvider.provider();
//监听端口8888,能同时接受10个请求
HttpServer httpserver = provider.createHttpServer(new InetSocketAddress(8888), 10);
httpserver.createContext("/", new AuthorizationHandler());
httpserver.start();
System.out.println("server started");
}
public static void main(String[] args) throws IOException {
AuthorizationServer server = new AuthorizationServer();
server.startService();
}
}
```

- Python代码示例

  Python SDK需要的依赖如下：

  - pip install aliyun-python-sdk-core
  - pip install aliyun-python-sdk-kms
  - pip install aliyun-python-sdk-mts

  Python代码示例如下：

```
# -*- coding: UTF-8 -*-
from BaseHTTPServer import BaseHTTPRequestHandler
from aliyunsdkcore.client import AcsClient
from aliyunsdkkms.request.v20160120 import DecryptRequest
import cgi
import json
import base64
import urlparse
client = AcsClient("","","");
class AuthorizationHandler(BaseHTTPRequestHandler):
def do_GET(self):
self.check()
self.set_header()
cipertext = self.get_cihpertext()
plaintext = self.decrypt_cihpertext(cipertext)
print plaintext
key = base64.b64decode(plaintext)
print key
self.wfile.write(key)
def do_POST(self):
pass
def check(self):
#check MtsHlsUriToken, etc.
pass
def set_header(self):
self.send_response(200)
#cors
self.send_header('Access-Control-Allow-Origin', '*')
self.end_headers()
def get_cihpertext(self):
path = urlparse.urlparse(self.path)
query = urlparse.parse_qs(path.query)
return query.get('Ciphertext')[0]
def decrypt_cihpertext(self, cipertext):
request = DecryptRequest.DecryptRequest()
request.set_CiphertextBlob(cipertext)
response = client.do_action_with_exception(request)
jsonResp = json.loads(response)
return jsonResp["Plaintext"]
if __name__ == '__main__':
# Start a simple server, and loop forever
from BaseHTTPServer import HTTPServer
print "Starting server, use  to stop"
server = HTTPServer(('127.0.0.1', 8888), AuthorizationHandler)
server.serve_forever()
```

3. 调用媒体处理的查询媒体-使用媒体ID接口获取播放地址。

   您可以通过开发者门户直接调用API或将API集成到服务中调用。

4. 播放加密视频。

   您可以使用自己的播放器或阿里云播放器播放加密视频。

   - 如果使用非阿里云播放器，请自行实现播放逻辑。

- 如果使用阿里云播放器，请按照阿里云播放器的要求获取令牌和鉴权信息后播放。详细介绍请参见视频播放。

  您也可以借助一个在线播放器，测试HLS加密视频的播放。

  以阿里云播放器用户诊断工具为例，请将获取的播放地址，填入对话框中，单击**视频播放**。

  > **说明** 通过浏览器DEBUG，可以看到播放器自动请求了鉴权服务器，获取解密密钥，并进行解密播放。

  使用阿里云播放器测试播放的内部流程解析如下：

  - 获取播放地址后，播放器把OSS域名替换为CDN域名，再拼接上参数MtsHlsUriToken作为请求解密密钥的令牌向CDN获请求播放地址。请求示例：`https://example.aliyundoc.com/test_01.m3u8?MediaId=fbbf98691ea44b7c82dd75c5bc8b****&MtsHlsUriToken=<业务方颁发的令牌>`。

    > **注意**
    >
    > 如果您使用的是阿里云播放器，则无需自行拼接MtsHlsUriToken。如果您使用的是其它播放器，则需要自行拼接。

  - 收到请求后，阿里云CDN动态修改M3U8文件中的解密URI并返回修改后的地址给播放器，如原为`https://example.aliyundoc.com?Ciphertext=aabbccddeeff&MediaId=fbbf98691ea44b7c82dd75c5bc8b****`，修改后为`https://example.aliyundoc.com?Ciphertext=aabbccddeeff&MediaId=fbbf98691ea44b7c82dd75c5bc8b****&MtsHlsUriToken=<业务方颁发的令牌>`。
  - 播放器访问M3U8文件中EXT-X-KEY标签中的URI以获取解密密钥，此URI为业务方搭建的解密密钥接口。业务方调用KMS的Decrypt接口，并将获取的明文密钥Base64decode之后返回给播放器。播放器用数据密钥（DK）去解密加密过的ts文件进行播放。

# 6.7. 上传视频文件

## 6.7.1. 设置RAM用户和授权

本文为您介绍在进行解密播放之前如何设置RAM用户和授权。

1. 新建RAM用户。
   i. 登录RAM控制台。
   ii. 在左侧导航栏，选择**身份管理 > 用户**，单击**创建用户**。
   iii. 在**创建用户**页面的**用户账号信息**区域，输入**登录名称**和**显示名称**。
   iv. **访问方式**勾选**控制台访问**（**控制台密码**、**需要重置密码**、**MFA多因素认证**等按需设置）、**OpenAPI调用访问**，创建一个拥有和云账号一样完全访问MPS权限的RAM用户。
   v. 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，自动生成改RAM用户的AccessKey。

vi. 生成账号的AccessKey后，复制AccessKey ID和AccessKey Secret并妥善保存，用于后续访问。

> **注意** MPS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，请务必妥善保存！
>
> - AccessKey ID：用于标识用户。
> - AccessKey Secret：用户用于加密签名的字符串，以及MPS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。

2. 新建角色。
   i. 在左侧导航栏，单击**身份管理 > 角色**。
   ii. 在**角色**页签下，单击**创建角色**。
   iii. 在**选择类型**页签下，**可信实体类型**选择**阿里云账号**。单击**下一步**。
   iv. 在**角色名称**文本框中填写角色名称，**信任的云账号**选择**当前云账号**，单击**完成**。
   v. 单击左下角**关闭**按钮关闭当前页面。
   vi. 在RAM角色列表中找到创建的角色，单击角色名进入**角色基本信息**，复制**ARN**参数并妥善保存，用于后续访问。
3. 设置角色的权限。
   i. 在**角色**页签下，找到您创建的角色，单击**操作**列**添加权限**。
   ii. **授权范围**选择**整个云账号**。

| 授权范围 | 说明 |
| --- | --- |
| **整个云账号** | 权限在当前阿里云账号内生效。 |
| **指定资源组** | 权限在指定资源组内生效。 |

   iii. **授权主体**系统自动填入当前的RAM用户，无需更改。
   iv. **选择权限**选择**系统策略**，在**权限策略名称**列中单击需要添加的权限策略名称，添加后单击**确定**。添加完权限策略后单击**完成**。

> **说明** 如果您想要调整RAM用户的STS权限（例如，修改、增加、删除等），只需回到本步骤操作即可。

4. 关联RAM用户和角色。
   i. 在RAM控制台左侧导航栏中，选择**权限管理 > 权限策略**。在权限策略页签下，单击**创建权限策略**。
   ii. 选择**脚本编辑**。

iii. 将Resource字段，修改成您所获取的ARN参数。Action参数根据您的实际需要设置。设置完成后，单击**下一步**。

iv. 在**基本信息**界面**名称**文本框中，输入**策略名称**，单击**确定**，并完成验证。

v. 在左侧导航栏，选择**身份管理 > 用户**。

vi. 找到您所设置的RAM用户，单击**操作**列的**添加权限**。

vii. **选择权限**选择**自定义权限**，单击需要添加的权限。完成后单击**确定**，完成添加权限。

# 6.7.2. 请求安全令牌

通过媒体ID播放视频时，需要使用访问控制RAM服务的安全令牌。本文为您介绍如何获得安全令牌。

## 前提条件

在请求安全令牌之前，您需要安装访问控制（RAM）服务提供的STS SDK。本文以Java为例，详细操作及代码示例请参见Java示例。如需其他语言代码示例，请参见STS SDK概览。

## 背景信息

使用媒体工作流时，每个多媒体输入文件由媒体ID统一标识，一一对应。媒体ID可以关联多个格式、多个清晰度的输出，当您需要实现多清晰度自动切换、多格式支持播放时，可以使用媒体ID播放视频。在使用媒体ID播放时，需要使用访问控制服务的安全令牌。

> **说明** 为保证加密视频的安全性，加密视频的播放必须使用媒体ID的播放方式。

## 使用前准备

您在请求之前需要先准备好STS需要的角色的参数 `roleArn` 。

1. 登录 RAM 控制台，单击 **身份管理 > 角色**。
2. 在角色列表中找到具体 **角色名称** 后，单击角色名称进入基本信息页面。
3. 复制ARN参数并妥善保存，供后续使用。

1. 在pom.xml文件中引用STS SDK。

```
<repositories>
     <repository>
         <id>sonatype-nexus-staging</id>
         <name>Sonatype Nexus Staging</name>
         <url>https://oss.sonatype.org/service/local/staging/deploy/maven2/</url>
         <releases>
             <enabled>true</enabled>
         </releases>
         <snapshots>
             <enabled>true</enabled>
         </snapshots>
     </repository>
 </repositories>
<dependencies>
 <dependency>
   <groupId>com.aliyun</groupId>
   <artifactId>aliyun-java-sdk-sts</artifactId>
   <version>2.1.6</version>
 </dependency>
 <dependency>
   <groupId>com.aliyun</groupId>
   <artifactId>aliyun-java-sdk-core</artifactId>
   <version>2.2.0</version>
 </dependency>
</dependencies>
```

**说明** 请访问Maven仓库获取 `aliyun-java-sdk-core` 的最新版本。

2. 生成令牌。

```
//main函数
public static void main(String[] args) throws Exception {
   IClientProfile profile = DefaultProfile.getProfile(
                                          "cn-hangzhou",
                                          <accessKeyId>,
                                          <accessKeySecret>);
   DefaultAcsClient client = new DefaultAcsClient(profile);
   AssumeRoleResponse response = assumeRole(client, <roleArn>);
   AssumeRoleResponse.Credentials credentials = response.getCredentials();
   System.out.println(credentials.getAccessKeyId() + "\n" +
                      credentials.getAccessKeySecret() + "\n" +
                      credentials.getSecurityToken() + "\n" +
                      credentials.getExpiration());
}
//生成临时AK和Token的函数
private static AssumeRoleResponse assumeRole(
                                   DefaultAcsClient client,
                                   String roleArn)
                                   throws ClientException {
   final AssumeRoleRequest request = new AssumeRoleRequest();
   request.setVersion("2015-04-01");
   request.setMethod(MethodType.POST);
   request.setProtocol(ProtocolType.HTTPS);
   request.setDurationSeconds(900L);
   request.setRoleArn(roleArn);
   request.setRoleSessionName("test-token");
   return client.getAcsResponse(request);
}
```

3. 调整Token有效期。

   示例代码中生成的Token有效时间为900秒，可以根据实际需求调整（最小900秒，最大3,600秒）。

   在有效期内，不需要反复生成新的Token，可以复用已经生成的Token。您可以通过以下方式判断何时需要重新生成Token：

```
private static boolean isTimeExpire(String expiration) {
     Date nowDate = new Date();
     Date expireDate = javax.xml.bind.DatatypeConverter.parseDateTime(expiration).getTi
me();
     if (expireDate.getTime() <= nowDate.getTime()) {
         return true;
     } else {
         return false;
     }
}
```

# 6.7.3. 视频播放

本文介绍了视频播放的方式以及如何进行视频播放。

## 播放方式

- 通过播放地址播放：

- 场景：针对提交转码作业的输出视频，有确定的输出地址。直接把需要播放的URL地址（OSS URL或CDN URL）传给阿里云播放器SDK。
- 优点：服务端负责生成最终的播放URL地址，客户端实现简单。
- 缺点：服务端需要实现OSS、CDN的鉴权以及URL域名的拼接等，且客户端无法实现多个清晰度手动、自动切换。

- 通过媒体ID播放
  - 优点：使用媒体工作流时，每个多媒体输入文件由媒体ID（MediaId）唯一标识，一一对应。媒体ID可以关联多个格式、多个清晰度的输出，所以使用媒体ID播放时，可以实现多清晰度自动切换。多格式支持。
  - 播放方式：使用访问控制RAM服务的安全令牌。安全令牌不使用主账号的权限，而是使用子账号的权限，并且提供了有时效的访问权限，能很好的避免权限泄露等问题。

> **说明**
> - 为了保证加密视频的安全性，加密视频的播放必须使用媒体ID的播放方式。Web端播放加密视频请使用Flash播放器（HTML5不支持加密视频播放）。
> - 直播客户使用工作流方式转点播，可使用RecordUrl查询出媒体ID进行播放，接口为查询媒体-使用OSS文件地址。更多信息请参见媒资转码SDK。

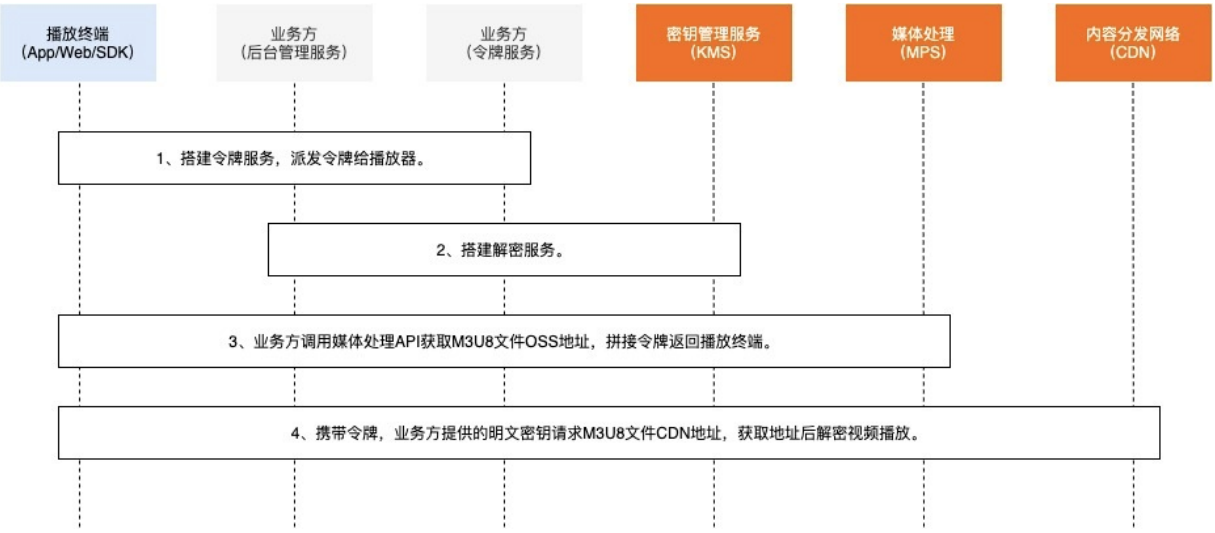
## 播放地址播放

服务端实现OSS、CDN的鉴权以及URL域名的拼接后，直接把需要播放的URL地址（OSS URL或CDN URL）传给播放器SDK。鉴权详细信息请参见播放鉴权。

## 媒体ID播放

通过媒体ID播放加密视频需要使用RAM服务的安全令牌。安全令牌详细信息请参见安全令牌。

通过媒体ID播放加密视频的流程如下图：




请按以下指引完成视频加密：

1. 业务方搭建令牌服务，用于颁发令牌MtsHlsUriToken。

> **注意** 令牌服务指用于派发MtsHlsUriToken的服务。

2. 业务方调用KMS解密接口搭建解密服务，用于解密视频，同时提供数据密钥（DK）给播放终端。

> **注意** KMS返回Base64加密后的数据密钥给业务方。业务方需要将调用KMS接口获得的数据密钥Base64 Decode之后返回给播放终端。

3. 业务方调用MPS的QueryMediaList接口获取视频M3U8文件的OSS地址，并将地址拼接MtsHlsUriToken后返回给播放终端。
4. 播放终端携带MtsHlsUriToken、数据密钥向阿里云CDN请求播放地址，阿里云CDN改写M3U8文件，将业务方的Key URI与信封加密密钥返回播放终端。播放终端解密播放视频。

## 安全令牌

1. **安全令牌**需在RAM用户授权时，增加角色授权策略，具体操作请参见设置RAM用户和授权。
2. 设置好后，获得安全令牌，参见请求安全令牌。更多语言示例请参Java示例、.NET示例、Python示例、Node.js示例、Go示例。

## 播放鉴权

播放器SDK主要参数如下表所示：

| 参数 | 描述 | 类型 | 是否必须 |
| --- | --- | --- | --- |
| vid | 媒体ID | String | 是 |
| source | 播放地址 | String | 是 |
| accId | STS AccessKeyId | String | 是 |
| accSecret | STS AccessKeySecret | String | 是 |
| stsToken | STS SecurityToken | String | 是 |
| domainRegion | 媒体工作流所在的区域（cn-shanghai，cn-hangzhou等） | String | 是 |
| authInfo | 鉴权信息，JSON字符串，组成见下面详细说明 | String | 是 |

authInfo是一个JSON对象，包含3个属性：

- ExpireTime：鉴权过期的UTC时间（其他时区的时间要转换成UTC时间），只能在设定的过期时间之前获取播放视频的URL，否则会导致权限过期无法播放。格式示例：2021-03-25T23:59:59Z。
- MediaId：媒体处理服务的媒体ID。
- Signature：鉴权签名，避免伪造的请求。

  Signature的计算过程如下：

  - 参数

```
ExpireTime="2017-03-25T23:59:59Z"
  MediaId="5aa0276ff6204ace950f75acf9e6187b"
```

- 签名字符串

  把参数按照字母顺序排序，key=value的格式，并使用（&）符号进行拼接。其中value遵循RFC-2396（UTF-8编码，并进行URLEncode）。更多信息，请参见URL Encoding说明。

```
ExpireTime=2017-03-25T23%3A59%3A59Z&MediaId=5aa0276ff6204ace950f75acf9e6187b
```

- 签名算法

  用HMAC_SHA1计算出二进制的结果后，做Base64编码。

  在类Unix系统中，可以使用openssl+base64命令来生成，在类Unix系统中，可以使用openssl+base64命令来生成，假设鉴权Key=”secret”，执行如下命令：

```
$echo -n 'ExpireTime=2017-03-25T23%3A59%3A59Z&MediaId=5aa0276ff6204ace950f75acf9e6****'
| openssl sha1 -binary -hmac 'secret' | base64
  z7mmSRuTXo4mydiWhRtbu8JKDpM=
```

- 最终结果

```
{
    "ExpireTime":"2017-03-25T23:59:59Z",
    "MediaId":"5aa0276ff6204ace950f75acf9e6****",
    "Signature":"z7mmSRuTXo4mydiWhRtbu8JKDpM="
  }
```

### 播放器的使用

MPS播放器已和点播播放器合并。具体使用请参见SDK简介。

> **说明** MPS用户可以使用新版提供的高级播放器进行升级。

# 6.8. 媒体库管理

## 6.8.1. 媒体库简介

本文为您介绍媒体库的功能及业务场景。

### 功能

- 媒体工作流管理：增、删、改、查以及激活和停止。
- 媒体工作流执行实例：遍历和查询。
- 媒体管理：增、删、改、查、搜索、维护媒体属性（标题、标签、封面、描述），以及设置媒体发布状态。
- 媒体类目管理：增、删、改、查。

### 业务场景

- 搜索媒体

  在媒体库中搜索满足条件的媒体集合。

  搜索方式：

  - 关键字搜索，逻辑“或”的关系：只要标题、标签、描述、类目任意属性能匹配即可。

- 组合条件搜索，逻辑“与”的关系：指定的组合属性中（标题、标签、描述、类目中的两个及以上），必须每个属性都匹配。

在搜索条件中，可以指定创建时间的区间来限定搜索范围。返回的结果也可以指定规则排序，按照创建时间升序或降序进行排序。

返回的结果比较多时，可以选择分页获取。

- 维护媒体属性

  每个媒体都包含四个基本属性：标题、标签、描述、类目，都可以通过API来设置。示例代码请参见媒体基本信息。

- 查询媒体详细信息

  一个媒体包含一个输入和若干个输出（视频、截图等）。可以通过查询返回媒体的详细输入、输出信息。示例代码请参见媒体详细信息。

  - 输入信息包含：视频基本属性（宽、高、时长、大小、码率、帧率）以及视频详情（容器封装、视频、音频、字幕流，以及封装和流的详细信息）。
  - 输出信息包含：视频包含基本属性（宽、高、时长、大小、码率、帧率）以及OSS的URL地址。截图包含类型（单帧、批量）以及OSS的URL地址。

- 管理媒体标签

  媒体库不提供全局标签管理，每个媒体的标签都是独立设置的，可以通过搜索媒体的API来查找所有设置了相同标签的媒体。示例代码请参见标签管理。

## 功能使用

您可以通过媒体库SDK使用媒体库，媒体库SDK支持Java、Python、PHP。也可以直接通过HTTP、HTTPS访问API，详情请参见API使用手册。

# 6.8.2. 媒体基本信息

本文为您介绍如何查询和更新媒体基本信息。

## 功能使用

媒体处理支持通过SDK或API查询和更新媒体基本信息。SDK安装，请参见Java SDK、Python SDK、PHP SDK，示例代码参见下文。如需通过API使用功能，具体接口信息请参见直接调用API处理媒体基本信息。

> **说明** 本文以PHP语言为例。

## SDK调用API处理媒体基本信息

- 查询媒体基本信息

  媒体处理提供2种查询方式查询媒体基本信息：通过媒体ID查询、通过OSS文件地址查询。

- 使用媒体ID查询媒体基本信息

> **说明** 如需了解参数详细信息，请参考查询媒体-使用媒体ID。

```
include_once 'aliyun-php-sdk-core/Config.php';
use Mts\Request\V20140618 as Mts;
$accessKeyID = 'test****'; // 替换成真实的ID
$accessKeySecret = 'test****'; // 替换成真实的secret
$profile = DefaultProfile::getProfile('<your Region>',//示例：cn-hangzhou
                                      $accessKeyID,
                                      $accessKeySecret);
$client = new DefaultAcsClient($profile);function queryMediaById($client, $mediaID)
{
     $request = new Mts\QueryMediaListRequest();
     $request->setAcceptFormat('JSON');
     $request->setMediaIds($mediaID);
     $response = $client->getAcsResponse($request);
     return $response;
}
function printMedia($media)
{
     if (array_key_exists('Title', $media)) {
       print_r('Title: '.$media->{'Title'}."\n");
     }
     if (array_key_exists('Description', $media)) {
       print_r('Description: '.$media->{'Description'}."\n");
     }
     if (array_key_exists('Tags', $media)) {
       print_r('Tags: '.$media->{'Tags'}->{'Tag'}[0]."\n");
     }
     if (array_key_exists('CoverURL', $media)) {
       print_r('CoverURL: '.$media->{'CoverURL'}."\n");
     }
     print_r('Format: '.$media->{'Format'}."\n");
     print_r('Resolution: '.$media->{'Width'}.'x'.$media->{'Height'}."\n");
     print_r('FileSize: '.$media->{'Size'}."\n");
     print_r('Bitrate: '.$media->{'Bitrate'}."\n");
     print_r('FPS: '.$media->{'Fps'}."\n");
}
$mediaID = 'test****'; // 替换成真实的mediaID
$medias = queryMediaById($client, $mediaID)->{'MediaList'}->{'Media'};
for ($i=0; $i < count($medias); $i++) {
     printMedia($medias[$i]);
}
```

- 使用OSS文件地址查询媒体基本信息

> **说明** OSS文件地址指媒体文件路径。例如：`http://example-bucket-****.oss-cn-hangzhou.aliyuncs.com/test/1****.mp4`。如需了解参数详细信息，请参考查询媒体-使用OSS文件地址。

```
function queryMediaByURL($client, $mediaURL)
{
     $request = new Mts\QueryMediaListByURLRequest();
     $request->setAcceptFormat('JSON');
     $request->setFileURLs($mediaURL);
     $response = $client->getAcsResponse($request);
     return $response;
}
$ossEndpoint = 'http://example-bucket-****.oss-cn-hangzhou.aliyuncs.com/';
// OSS的Object不需要"/"开始，替换成真实的OSS Object
$ossObject = 'test/example****.mp4';
$medias = queryMediaByURL($client,$ossEndpoint.urlencode($ossObject))->{'MediaList'}->{
'Media'};
for ($i=0; $i < count($medias); $i++) {
     printMedia($medias[$i]);
}
```

- 更新媒体属性

  媒体处理提供2种更新媒体属性的方式：全量属性更新、单个属性更新。

  - 全量属性更新

> **说明** 更新时必须指定所有字段，不设置的字段会被清空。如需了解参数详情，请参考更新媒体-基本信息。

```
function updateMediaAllField($client, $mediaID, $title, $description, $tags, $coverURL)
{
     $request = new Mts\UpdateMediaRequest();
     $request->setAcceptFormat('JSON');
     $request->setMediaId($mediaID);
     $request->setTitle($title);
     $request->setCateId(2663987);
     $request->setDescription($description);
     $request->setTags($tags);
     $request->setCoverURL($coverURL);
     $response = $client->getAcsResponse($request);
     return $response;
}
$mediaID = 'test'; // 替换成真实的mediaID
$media = updateMediaAllField($client, $mediaID,
                        'title', 'description', 'tags', 'coverURL')->{'Media'};
```

- 单个属性更新

> **说明** 不同的字段可以单独更新，使用的是不同的API，可以在不修改其他字段的情况下，方便的更新单个字段。本文通过“发布状态”举例，如需了解参数详情，请参考更新媒体-发布状态。

```
function updateMediaPublishState($client, $mediaID, $state)
{
     $request = new Mts\UpdateMediaPublishStateRequest();
     $request->setAcceptFormat('JSON');
     $request->setMediaId($mediaID);
     $request->setPublish($state);
     $response = $client->getAcsResponse($request);
     return $response;
}
$mediaID = 'test'; // 替换成真实的mediaID
// 更新"发布状态"的API没有返回值，通过捕获异常来判断是否执行成功
try {
     updateMediaPublishState($client, $mediaID, "true");
} catch (ClientException $e) {
     print_r('ClientException:'."\n");
     print_r($e);
} catch (ServerException $e) {
     print_r('ServerException:'."\n");
     print_r($e);
}
```

## 直接调用API处理媒体基本信息

> **说明** 直接调用API是指直接在OpenAPI门户调用该接口请求查询或修改媒体基本信息。

下表为查询和更新媒体信息的对应接口。

| 功能 | 接口 |
| --- | --- |
| 查询媒体基本信息-使用媒体ID | 查询媒体-使用媒体ID |
| 查询媒体基本信息-使用OSS文件地址 | 查询媒体-使用OSS文件地址<br>**说明** OSS文件地址指媒体文件路径。例如：`http://example-bucket-****.oss-cn-hangzhou.aliyuncs.com/test/1****.mp4` 。 |
| （全量）更新媒体属性 | 更新媒体-基本信息 |
| （单个）更新媒体属性 | • 更新媒体-类目<br>• 更新媒体-封面<br>• 更新媒体-发布状态 |

# 6.8.3. 媒体详细信息

## 简介

SDK的安装和使用，请参考媒体库SDK-PHP。

> **说明** 示例语言使用PHP，如您使用其他语言进行开发，请参考媒体库SDK-Java、媒体库SDK-Python。

一个媒体包含一个输入文件和若干输出文件。输入除了基本信息之外，还有详细的媒体信息。输出可以查询视频和截图的详细信息。

## 输入媒体信息

```
include_once 'aliyun-php-sdk-core/Config.php';
 use Mts\Request\V20140618 as Mts;
 $accessKeyID = 'test'; // 替换成真实的id
 $accessKeySecret = 'test'; // 替换成真实的secret
 $profile = DefaultProfile::getProfile('cn-hangzhou',
                                       $accessKeyID,
                                       $accessKeySecret);
 $client = new DefaultAcsClient($profile);
```

```
function queryMedia($client, $mediaID)
{
    $request = new Mts\QueryMediaListRequest();
    $request->setAcceptFormat('JSON');
    $request->setMediaIds($mediaID);
    $request->setIncludeMediaInfo("true");
    $response = $client->getAcsResponse($request);
    return $response;
}
function printMediaInfo($mediaInfo)
{
    print_r('Number of Streams: '.$mediaInfo->{'Format'}->{'NumStreams'}."\n");
    if (array_key_exists('Streams', $mediaInfo) &&
        array_key_exists('AudioStreamList', $mediaInfo->{'Streams'}) &&
        array_key_exists('AudioStream', $mediaInfo->{'Streams'}->{'AudioStreamList'})) {
        $audioStreams = $mediaInfo->{'Streams'}->{'AudioStreamList'}->{'AudioStream'};
        print_r('Audio Streams:'."\n");
        for ($i = 0; $i < count($audioStreams); $i++) {
            print_r("\t[".$i."]"."\n");
            print_r("\t\tCodecName: ".$audioStreams[$i]->{'CodecName'}."\n");
            print_r("\t\tChannels: ".$audioStreams[$i]->{'Channels'}."\n");
            print_r("\t\tSamplerate: ".$audioStreams[$i]->{'Samplerate'}."\n");
            print_r("\t\tDuration: ".$audioStreams[$i]->{'Duration'}."\n");
            print_r("\t\tBitrate: ".$audioStreams[$i]->{'Bitrate'}."\n");
        }
    }
    if (array_key_exists('Streams', $mediaInfo) &&
        array_key_exists('VideoStreamList', $mediaInfo->{'Streams'}) &&
        array_key_exists('VideoStream', $mediaInfo->{'Streams'}->{'VideoStreamList'})) {
        $videoStreams = $mediaInfo->{'Streams'}->{'VideoStreamList'}->{'VideoStream'};
        print_r('Video Streams:'."\n");
        for ($i = 0; $i < count($videoStreams); $i++) {
            print_r("\t[".$i."]"."\n");
            print_r("\t\tCodecName: ".$videoStreams[$i]->{'CodecName'}."\n");
            print_r("\t\tProfile: ".$videoStreams[$i]->{'Profile'}."\n");
            print_r("\t\tDuration: ".$videoStreams[$i]->{'Duration'}."\n");
            print_r("\t\tPixFmt: ".$videoStreams[$i]->{'PixFmt'}."\n");
            print_r("\t\tFps: ".$videoStreams[$i]->{'Fps'}."\n");
            print_r("\t\tBitrate: ".$videoStreams[$i]->{'Bitrate'}."\n");
            print_r("\t\tResolution: ".$videoStreams[$i]->{'Width'}.'x'.$videoStreams[$i]->
{'Height'}."\n");
        }
    }
}
$mediaID = 'test'; // 替换成真实的mediaID
$medias = queryMedia($client, $mediaID)->{'MediaList'}->{'Media'};
for ($i = 0; $i < count($medias); $i++) {
    printMediaInfo($medias[$i]->{'MediaInfo'});
}
```

## 输出

- 视频

```
function queryMedia($client, $mediaID)
{
    $request = new Mts\QueryMediaListRequest();
    $request->setAcceptFormat('JSON');
    $request->setMediaIds($mediaID);
    $request->setIncludePlayList("true");
    $response = $client->getAcsResponse($request);
    return $response;
}
function printOutputVideos($videos)
{
    print_r('Number of Output Video: '.count($videos)."\n");
    for ($i = 0; $i < count($videos); $i++) {
        print_r("\t[".$i."]"."\n");
        print_r("\t\tMediaWorkflowName: ".$videos[$i]->{'MediaWorkflowName'}."\n");
        print_r("\t\tActivityName: ".$videos[$i]->{'ActivityName'}."\n");
        print_r("\t\tFormat: ".$videos[$i]->{'Format'}."\n");
        print_r("\t\tDuration: ".$videos[$i]->{'Duration'}."\n");
        print_r("\t\tFps: ".$videos[$i]->{'Fps'}."\n");
        print_r("\t\tBitrate: ".$videos[$i]->{'Bitrate'}."\n");
        print_r("\t\tSize: ".$videos[$i]->{'Size'}."\n");
        print_r("\t\tResolution: ".$videos[$i]->{'Width'}.'x'.$videos[$i]->{'Height'}."\n
");
        print_r("\t\tURL: ".$videos[$i]->{'File'}->{'URL'}."\n");
    }
}
$mediaID = 'test'; // 替换成真实的mediaID
$medias = queryMedia($client, $mediaID)->{'MediaList'}->{'Media'};
for ($i = 0; $i < count($medias); $i++) {
    printOutputVideos($medias[$i]->{'PlayList'}->{'Play'});
}
```

- 截图

```
function queryMedia($client, $mediaID)
{
    $request = new Mts\QueryMediaListRequest();
    $request->setAcceptFormat('JSON');
    $request->setMediaIds($mediaID);
    $request->setIncludeSnapshotList("true");
    $response = $client->getAcsResponse($request);
    return $response;
}
function printOutputSnapshots($snapshots)
{
    print_r('Number of Output Snapshot: '.count($snapshots)."\n");
    for ($i = 0; $i < count($snapshots); $i++) {
        print_r("\t[".$i."]"."\n");
        print_r("\t\tMediaWorkflowName: ".$snapshots[$i]->{'MediaWorkflowName'}."\n");
        print_r("\t\tActivityName: ".$snapshots[$i]->{'ActivityName'}."\n");
        print_r("\t\tType: ".$snapshots[$i]->{'Type'}."\n");
        print_r("\t\tCount: ".$snapshots[$i]->{'Count'}."\n");
        print_r("\t\tURL: ".$snapshots[$i]->{'File'}->{'URL'}."\n");
    }
}
$mediaID = 'test'; // 替换成真实的mediaID
$medias = queryMedia($client, $mediaID)->{'MediaList'}->{'Media'};
for ($i = 0; $i < count($medias); $i++) {
    printOutputSnapshots($medias[$i]->{'SnapshotList'}->{'Snapshot'});
}
```

# 6.8.4. 标签管理

媒体库不提供全局的标签管理和设置，每个媒体的标签都是独立的。本文为您介绍如何为媒体文件添加和删除标签。

## 功能使用

媒体处理支持通过SDK或API管理标签。SDK安装请参见Java SDK、Python SDK、PHP SDK，示例代码见下文。如需通过API使用功能，请参见直接调用API管理标签。

> **说明** 本文以PHP语言为例。

## SDK调用API管理标签

- 添加标签

  标签的API支持单个标签的添加和删除，如果要一次设置多个标签，可以通过更新媒体-基本信息实现。添加标签的详细参数说明请参见更新媒体-添加标签，示例代码如下：

```
include_once 'aliyun-php-sdk-core/Config.php';
  use Mts\Request\V20140618 as Mts;
  $accessKeyID = 'exampleID'; // 替换成真实的id
  $accessKeySecret = 'exampleSecert'; // 替换成真实的secret
  $profile = DefaultProfile::getProfile('cn-hangzhou',
                                        $accessKeyID,
                                        $accessKeySecret);
  $client = new DefaultAcsClient($profile);
```

```
function addMediaTag($client, $mediaID, $tag)
  {
      $request = new Mts\AddMediaTagRequest();
      $request->setAcceptFormat('JSON');
      $request->setMediaId($mediaID);
      $request->setTag($tag);
      $response = $client->getAcsResponse($request);
      return $response;
  }
  $mediaID = 'exampleID'; // 替换成真实的mediaID
  // API没有返回值，通过捕获异常来判断是否执行成功
  try {
      addMediaTag($client, $mediaID, "testtag");
  } catch (ClientException $e) {
      print_r('ClientException:'."\n");
      print_r($e);
  } catch (ServerException $e) {
      print_r('ServerException:'."\n");
      print_r($e);
  }
```

- 删除标签

  删除标签的详细参数信息请参见更新媒体-删除标签，示例代码如下：

```
function deleteMediaTag($client, $mediaID, $tag)
  {
      $request = new Mts\DeleteMediaTagRequest();
      $request->setAcceptFormat('JSON');
      $request->setMediaId($mediaID);
      $request->setTag($tag);
      $response = $client->getAcsResponse($request);
      return $response;
  }
  $mediaID = 'exampleID'; // 替换成真实的mediaID
  // API没有返回值，通过捕获异常来判断是否执行成功
  try {
      deleteMediaTag($client, $mediaID, "testtag");
  } catch (ClientException $e) {
      print_r('ClientException:'."\n");
      print_r($e);
  } catch (ServerException $e) {
      print_r('ServerException:'."\n");
      print_r($e);
  }
```

## 直接调用API管理标签

说明 直接调用API是指直接在OpenAPI门户调用该接口对标签进行管理。

下表为标签管理对应的接口。

| 功能 | 接口 |
| --- | --- |
| 添加标签 | 更新媒体-添加标签 |
| 删除标签 | 更新媒体-删除标签 |