

获取暂存视频大纲

接口描述

- 根据频道ID与暂存文件ID查询AI大纲与答题内容（若已生成）
- 接口支持 HTTPS 协议

接口URL

```
http://api.polyv.net/live/v4/channel/record-file/subtitle/outline/get-by-fileId
```

[在线API调用](#)

请求方式

```
GET
```

接口约束

- 接口同时支持 HTTP、HTTPS，建议使用 HTTPS 确保接口安全
- 接口调用有频率限制，[详细请查看](#)

请求参数描述

参数名	必选	类型	说明
appId	true	String	账号appId【详见 获取密钥 】
timestamp	true	Long	当前13位毫秒级时间戳，3分钟内有效

参数名	必选	类型	说明
sign	true	String	签名，为32位大写的MD5值。 生成签名的appSecret 密钥作为通信数据安全的关键信息，严禁保存在客户端直接使用，所有API都必须通过客户自己服务器中转调用POLYV服务器获取响应数据【详见 签名生成规则 】
channelId	true	Integer	频道ID
fileId	true	String	暂存文件ID

示例

```
http://api.polyv.net/live/v4/channel/record-file/subtitle/outline/get-by-fileId?
appId=frrlr1zazn3&sign=79767F7E02132E1904C304E4F225CC2A&timestamp=1634783281943&channelId=100001&fileId=recordFileId123
```

响应参数描述

参数名	类型	说明
code	Integer	状态码，与 http 状态码相同，用于确定基本的响应状态
status	String	响应结果，由业务决定，成功返回success，失败返回error
success	Boolean	是否成功响应
requestId	String	请求ID，每次请求生成的唯一UUID，仅可用于排查、调试，不应该和业务挂上钩
error	Object	状态码非200时的错误信息【详见 Error字段描述 】
data	Object	【详见 Data字段说明 】

Error参数描述

参数名	类型	说明
code	Integer	错误代码，用于确定具体的错误原因
desc	String	错误描述，与 error.code 对应

Data参数描述

参数名	类型	说明
status	String	大纲生成状态：init-初始化，srt_processing-字幕生成中，outline_processing-大纲生成中，audit-待审核，success-完成，fail-生成失败
outline	Object	大纲JSON，未生成或生成失败时为 空
question	Object	答题JSON，未生成或生成失败时为 空
failReason	String	失败原因（status=fail时返回）

Java请求示例

快速接入基础代码请 [下载相关依赖源码](#)：[点击下载源码](#)，下载后加入到自己的源码工程中即可。测试用例中的 **HttpUtil.java** 和 **LiveSignUtil.java** 都包含在下载文件中。

强烈建议您使用[直播Java SDK](#)完成API的功能对接，直播Java SDK 对API调用逻辑、异常处理、数据签名、HTTP请求线程池进行了统一封装和优化。

```

private static final Logger log = LoggerFactory.getLogger(getClass());
/**
 * 获取暂存视频大纲
 * @throws IOException
 * @throws NoSuchAlgorithmException
 */
@Test
public void getRecordFileOutlineTest() throws IOException, NoSuchAlgorithmException {
    //公共参数，填写自己的实际参数
    String appId = super.appId;
    String appSecret = super.appSecret;
    String timestamp = String.valueOf(System.currentTimeMillis());

    //业务参数
    String url = "http://api.polyv.net/live/v4/channel/record-file/subtitle/outline/get-by-fileId";
    String channelId = "100001";
    String fileId = "recordFileId123";

    //http 调用逻辑
    Map<String, String> requestMap = new HashMap<>();
    requestMap.put("appId", appId);
    requestMap.put("timestamp", timestamp);
    requestMap.put("channelId", channelId);
    requestMap.put("fileId", fileId);
    requestMap.put("sign", LiveSignUtil.getSign(requestMap, appSecret));

    String response = HttpUtil.get(url, requestMap);
    log.info("测试获取暂存视频大纲成功: {}", response);
    //do somethings
}

```

响应示例

系统全局错误说明详见[全局错误说明](#)

成功示例

```
{
  "code": 200,
  "status": "success",
  "requestId": "b38bdd97dc404213a2436e7a086b9db9.82.16347832877195279",
  "data": {
    "status": "success",
    "outline": {
      "title": "直播回放大纲",
      "items": [
        {
          "title": "开场介绍",
          "startTime": 0,
          "endTime": 120
        }
      ]
    },
    "question": [
      {
        "question": "本场主题是什么？ ",
        "answer": "产品发布",
        "startTime": 60
      }
    ]
  },
  "success": true
}
```

异常示例

```
{
  "code": 400,
  "status": "error",
  "requestId": "d310b70bc329403f87f77f9203d50f89.128.16360832149273615",
  "error": {
    "code": 20001,
    "desc": "application not found."
  },
  "success": false
}
```