

Alibaba Cloud ApsaraVideo for Live Best Practices

Issue: 20190813

Legal disclaimer

Alibaba Cloud reminds you to carefully read and fully understand the terms and conditions of this legal disclaimer before you read or use this document. If you have read or used this document, it shall be deemed as your total acceptance of this legal disclaimer.

1. You shall download and obtain this document from the Alibaba Cloud website or other Alibaba Cloud-authorized channels, and use this document for your own legal business activities only. The content of this document is considered confidential information of Alibaba Cloud. You shall strictly abide by the confidentiality obligations. No part of this document shall be disclosed or provided to any third party for use without the prior written consent of Alibaba Cloud.
2. No part of this document shall be excerpted, translated, reproduced, transmitted, or disseminated by any organization, company, or individual in any form or by any means without the prior written consent of Alibaba Cloud.
3. The content of this document may be changed due to product version upgrades, adjustments, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and the updated versions of this document will be occasionally released through Alibaba Cloud-authorized channels. You shall pay attention to the version changes of this document as they occur and download and obtain the most up-to-date version of this document from Alibaba Cloud-authorized channels.
4. This document serves only as a reference guide for your use of Alibaba Cloud products and services. Alibaba Cloud provides the document in the context that Alibaba Cloud products and services are provided on an "as is", "with all faults" and "as available" basis. Alibaba Cloud makes every effort to provide relevant operational guidance based on existing technologies. However, Alibaba Cloud hereby makes a clear statement that it in no way guarantees the accuracy, integrity, applicability, and reliability of the content of this document, either explicitly or implicitly. Alibaba Cloud shall not bear any liability for any errors or financial losses incurred by any organizations, companies, or individuals arising from their download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, bear responsibility for any indirect, consequential, exemplary, incidental, special, or punitive damages, including lost profits arising from the use

or trust in this document, even if Alibaba Cloud has been notified of the possibility of such a loss.

5. By law, all the content of the Alibaba Cloud website, including but not limited to works, products, images, archives, information, materials, website architecture, website graphic layout, and webpage design, are intellectual property of Alibaba Cloud and/or its affiliates. This intellectual property includes, but is not limited to, trademark rights, patent rights, copyrights, and trade secrets. No part of the Alibaba Cloud website, product programs, or content shall be used, modified, reproduced, publicly transmitted, changed, disseminated, distributed, or published without the prior written consent of Alibaba Cloud and/or its affiliates. The names owned by Alibaba Cloud shall not be used, published, or reproduced for marketing, advertising, promotion, or other purposes without the prior written consent of Alibaba Cloud. The names owned by Alibaba Cloud include, but are not limited to, "Alibaba Cloud", "Aliyun", "HiChina", and other brands of Alibaba Cloud and/or its affiliates, which appear separately or in combination, as well as the auxiliary signs and patterns of the preceding brands, or anything similar to the company names, trade names, trademarks, product or service names, domain names, patterns, logos, marks, signs, or special descriptions that third parties identify as Alibaba Cloud and/or its affiliates).
6. Please contact Alibaba Cloud directly if you discover any errors in this document.

Generic conventions

Table -1: Style conventions

Style	Description	Example
	This warning information indicates a situation that will cause major system changes, faults, physical injuries, and other adverse results.	 Danger: Resetting will result in the loss of user configuration data.
	This warning information indicates a situation that may cause major system changes, faults, physical injuries, and other adverse results.	 Warning: Restarting will cause business interruption. About 10 minutes are required to restore business.
	This indicates warning information, supplementary instructions, and other content that the user must understand.	 Notice: Take the necessary precautions to save exported data containing sensitive information.
	This indicates supplemental instructions, best practices, tips, and other content that is good to know for the user.	 Note: You can use Ctrl + A to select all files.
>	Multi-level menu cascade.	Settings > Network > Set network type
Bold	It is used for buttons, menus, page names, and other UI elements.	Click OK.
Courier font	It is used for commands.	Run the <code>cd / d C :/ windows</code> command to enter the Windows system folder.
<i>Italics</i>	It is used for parameters and variables.	<code>bae log list --instanceid Instance_ID</code>
[] or [a b]	It indicates that it is an optional value, and only one item can be selected.	<code>ipconfig [-all -t]</code>

Style	Description	Example
<code>{}</code> or <code>{a b}</code>	It indicates that it is a required value, and only one item can be selected.	<code>switch {stand slave}</code>

Contents

Legal disclaimer.....	I
Generic conventions.....	I
1 Live Q&A scheme.....	1
1.1 Introduction.....	1
1.2 Detailed implementation procedure.....	8
1.3 OBS-based live Q&A.....	20
2 How to record live videos.....	28
3 Pure live audio and live video.....	30
4 How to perform overseas live activities.....	31
5 Live video security.....	34
5.1 Authentication code example.....	34

1 Live Q&A scheme

1.1 Introduction

Live Q&A is an activity during which users enter a live stream channel at the scheduled time and complete an online quiz under the guidance of a caster. Users who give correct answers to all twelve questions can share the bonus set for each activity. At the beginning of 2018, Internet circles witnessed a boom in "online Q&A games with big bonuses." You can introduce this interactive game to your business to build a highly interactive user community. The following section uses the live Q&A scheme provided by ApsaraVideo as an example to describe how to implement the live Q&A feature.

- Before starting live Q&A, you must [register an Alibaba Cloud account](#) and [activate ApsaraVideo Live](#).
- In the live Q&A scheme, we call an API through the application server to insert Supplemental Enhancement Information (SEI) signals into live streams, and use the Q&A player of ApsaraVideo to parse the SEI signals.

Procedure

Step 1: Understand the live Q&A process.

1. When the caster raises a question, the on-site director prepares to deliver the question.
2. The on-site director calls an API through the application server on the access side to insert several SEI frames into the current location of the live video stream. The SEI content can be customized based on the business needs.
3. ApsaraVideo Player SDK receives the video stream, parses the SEI frames in the stream, and sends a callback message to the application. The application immediately requests the question content from the application server and displays the question content to users.



Note:

Because stream ingest has a fixed delay and the on-site director uses a cloud service to insert SEI signals, SEI signals may be inserted into an image earlier than expected. To resolve this issue, the on-site director needs to measure the stream

ingest delay in advance and adds a fixed delay when inserting SEI signals into the live stream.

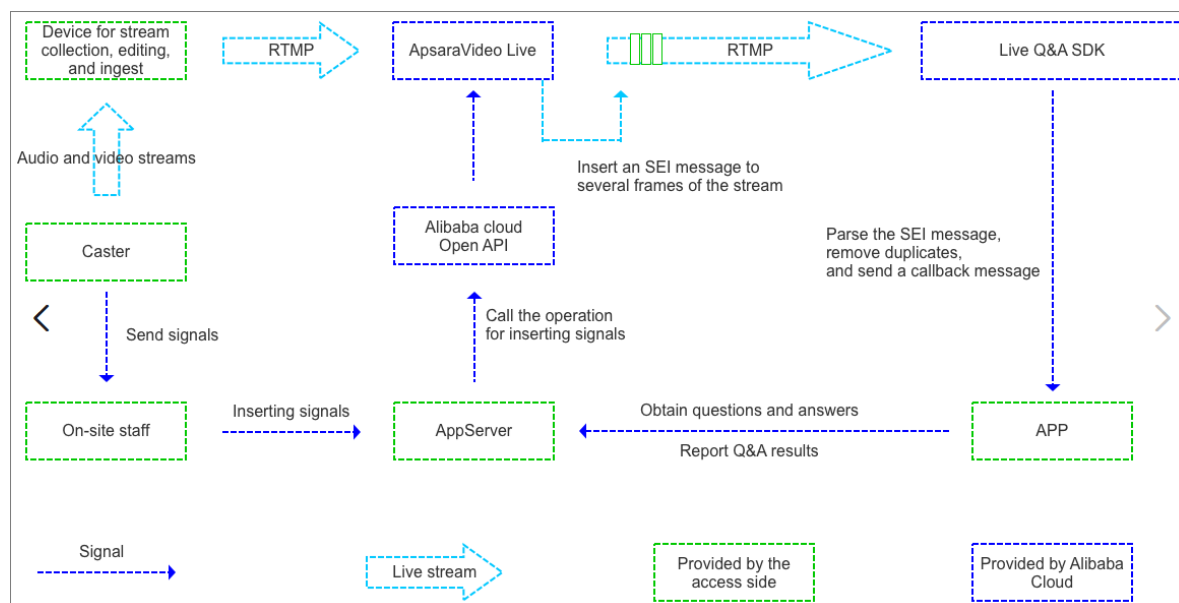
Assume that the on-site caster raises a question at 12:00:00 and that there is a 1-second delay for ingesting streams from the site to Alibaba Cloud. In this scenario, the on-site director needs to call the API to insert SEI signals at 12:00:01. Alternatively, the on-site director can call the API at 12:00:00 and set `delay` to `1000` milliseconds in the API. This ensures that the SEI signals are inserted into the image just at the right time when the caster raises the question.



Note:

- After receiving the SEI signals, the client needs to obtain the question from the application server, which consumes a certain amount of communication time. In the actual Q&A system, you can extend the answer submission period to an appropriate extent, for example, by 1 second based on the original one.
- After users answer the question, the client needs to report the answers, which consumes a certain amount of communication time. In the actual Q&A system, the application server allows a grace period for users to submit answers. For example, answers received within a 1-second delay of the original answer submission period are all considered valid.

The following figure shows the architecture of the live Q&A scheme.



Step 2: Activate ApsaraVideo Live and prepare for starting live Q&A.

- Activate ApsaraVideo Live first.

- Submit a ticket containing the domain name and application name to enable the feature of inserting SEI signals on ApsaraVideo.
- When ingesting streams, add the `wm = 1` parameter to the ingest URL.

Step 3: Call the API through the application server.

As described in the previous process, the application server on the access side needs to call the following API to insert SEI signals into a live stream.



Note:

Alternatively, you can use the [custom OBS tool](#) provided by ApsaraVideo to deliver questions.

1. API for inserting SEI signals

- Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to AddTrancodeSEI.
DomainName	String	Yes	The CDN domain name.
AppName	String	Yes	The name of the live streaming application.
StreamName	String	Yes	The name of the live stream. SEI will be inserted into this stream and all its encoded streams.
Text	String	Yes	The SEI text. Length constraint : 4,000 bytes.

Name	Type	Required	Description
Pattern	String	Yes	Indicates whether to insert SEI into each keyframe or frame. Valid values: keyframe and frame.
Repeat	Integer	Yes	The number of times that SEI is inserted. A value of -1 indicates that SEI is inserted for unlimited times until the end of live streaming.
Delay	Integer	Yes	The delay for inserting SEI upon the request. Unit: milliseconds.

- Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.

- Error codes

Error code	Error message	HTTP status code	Description
InvalidDomain. NotFound	The domain provided does not exist in our records.	404	The error message returned when the domain name does not exist or does not belong to the current user.

Error code	Error message	HTTP status code	Description
IllegalOperation	Illegal domain operate is not permitted.	403	The error message returned when the current operation is not supported. For example, the domain name is not used for live streaming.
InvalidParams	Invalid parameters	400	The error message returned when a parameter is invalid.
InternalError	The request processing has failed due to some unknown error.	500	The error message returned when an unknown error occurred in the background.

2. Examples

```
https://live.aliyuncs.com?Action=AddTranCodeSEI&Domain=test101.cdnpe.com&App=xxx&Stream=xxx&Text=hahaha&Pattern=frame&Repeat=300&Delay=0<common request parameters>
```



Note:

In the live Q&A scenario, we recommend that you set `Pattern` to `frame` and `Repeat` to three times the keyframe interval. (In the example, the value of `Repeat` is 300.) These settings ensure that the player can obtain the SEI once it receives one frame in three keyframe intervals no matter whether frames are lost in such intervals. Because the SEI in frames may be duplicate with each other, the player needs to deduplicate the SEI.

Step 4: Integrate and use ApsaraVideo Player SDK.

Integrate ApsaraVideo Player SDK first. For more information, see [Player SDK > Product introduction](#).

For live Q&A, ApsaraVideo Player SDK provides the feature of calling back SEI. The following section describes the SEI callback APIs for Android and iOS devices.

1. SEI callback API for Android devices

Set the SEI callback API in the player to obtain SEI for business processing. The methods are as follows:

- **setSEIDataListener**

```
public void setSEIDataListener ( MediaPlayer rSEIDataListener listener )
```

Description: Set the SEI callback API in the player to obtain SEI in streams.

Parameter: listener, which indicates the callback listener.

- **MediaPlayer.MediaPlayerSEIDataListener**

```
public void onSEIData ( String data )
```

Description: Listen on SEI callback messages.

Parameter: data, which indicates the SEI to be called back.



Note:

In a playback process, the same SEI is called back only once.

Examples

```
aliVcMedia Player = new AliVcMedia Player ( DemoLiveAnswerActivity.this , videoSurface );
aliVcMedia Player . setSEIDataListener ( new MediaPlayer . MediaPlayerSEIDataListener () {
    @Override
    public void onSEIData ( String data ) {
        praseData ( data ); // Parse SEI data .
    }
});
aliVcMedia Player . setMaxBufferDuration ( 3 * 1000 );
aliVcMedia Player . prepareAnd Play ( liveUrl );

private void praseData ( String data ) {
    // Parse the SEI into the format that meets the business needs and then perform business processing .
    JSONObject jsonObject = null ;
    try {
        jsonObject = new JSONObject ( data );
    } catch ( JSONException e ) {
        e . printStackTrace ();
    }

    SeiInfo seiInfo = SeiInfo . getFromJson ( jsonObject );
    if ( seiInfo == null ) {
        Toast . makeText ( getApplicationContext () , " SEI data error " , Toast . LENGTH_SHORT ) . show ();
        return ;
    }
}
```

```

        if ( seiInfo . type . equals ( " startAnsw e r " )) {
            // Request question content from the applicatio n
            server .
                requestQue stion ( seiInfo . questionId );
        } else if ( seiInfo . type . equals ( " showResult " )) {
            // Request the Q & A summary from the applicatio n
            server .
                requestQue stionRepor t ( seiInfo . questionId ,
                seiInfo . showTime );
        }
    }
}

```

For more information, see the demo.

2. SEI callback API for iOS devices

Listen on the SEI data notification in the player to obtain SEI data. The notification is as follows:

AliVcMediaPlayerSeiDataNotification

Examples

Add a listening notification API.

```

// The message received during live Q & A .
[[ NSNotifica tionCenter defaultCen ter ] addObserver :
self
                                selector :@ selector (
onSeiData :)
                                name : AliVcMedia
PlayerSeiD ataNotific ation object : self . mediaPlaye r ];

```

Process the SEI in the listening notification.

```

// Answer questions .
- ( void ) onSeiData :( NSNotifica tion *) notificati on {
    /*
    {
        " questionId ": " 001 ",
        " type ": " startAnsw e r "
    } Start to answer questions .

    {
        " questionId ": " 001 ",
        " type ": " showResult ",
        " showTime ": " 5 "
    } Show the Q & A result .

    */
    NSDictionary * dict = [ notificati on userInfo ];
    if ( dict ) {
        NSString * seiData = [ dict objectForKey :@" seiData
"];
        if ( seiData ) {
            NSLog (@" sei data is %@", seiData );

```

```
}  
  
}
```

For more information, see the demo.

Considerations

- How to cope with the player delay

You can set the longest player delay to ensure the same delay on each terminal. Generally, a longer delay indicates a stronger jitter resistance capability, and a shorter delay indicates a higher video stalling rate. Considering the overall delay of the entire link for live Q&A, the delay setting affects the validity time of each question. If the player delay is 3 seconds and the answer submission period is 10 seconds, the validity time of each question is 13 (which equals 10 plus 3) seconds.

In `@property(nonatomic, readonly) int dropBufferDuration;`, the `dropBufferDuration` parameter specifies the player delay for iOS devices, in milliseconds.

In `public void setMaxBufferDuration(int duration);`, the `setMaxBufferDuration` API controls the player delay for Android devices. The `duration` parameter specifies the player delay, in milliseconds.

- How to cope with the question delivery delay

After receiving the SEI signals, the client needs to obtain the question from the application server, which consumes a certain amount of communication time.

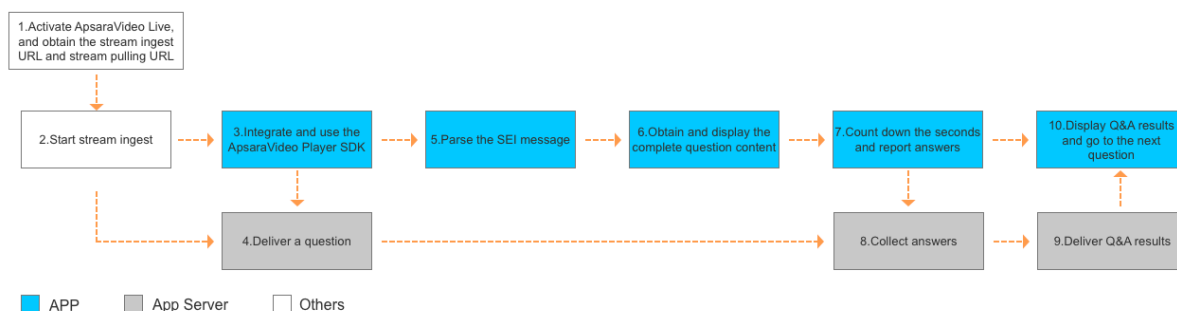
When delivering a question, increase the value of `expiredSeconds` as needed, for example, increase it by 1 second.

1.2 Detailed implementation procedure

This topic guides beginners through the procedure for creating a live Q&A service.

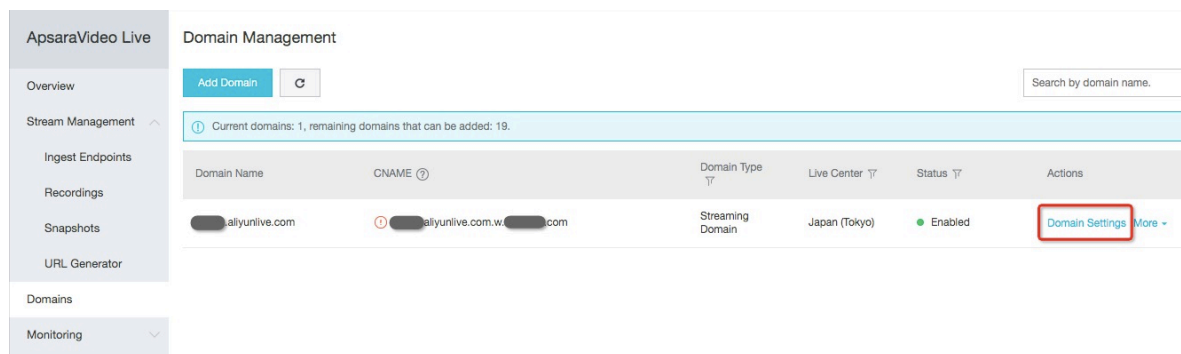
You can learn how to use ApsaraVideo Live to ingest streams and deliver questions

and how the application and application server interact with each other to complete live Q&A.



Step 1: Prepare for live Q&A.

1. Activate [ApsaraVideo Live](#) of Alibaba Cloud.
2. Log on to the ApsaraVideo Live console and choose Domain Management > Management to obtain the ingest URL and stream pulling URL. For more information, see [Stream ingest](#).



3. Submit a ticket to apply for enabling the feature of inserting SEI signals into ingest streams. Enter your live domain names and application name in the ticket.



Note:

The authentication feature is enabled by default. To prevent the risk of hotlinking, you must use a signed URL to ingest streams. For more information, see [Stream authentication](#).

Step 2: Start stream ingest.



Note:

- If you want to insert SEI signals into live streams on ApsaraVideo Live, you must add `wm = 1` to the ingest URL.

- After the stream ingest starts, an ingest URL that contains the `_sei-copy` string appears on the stream management page in addition to the target ingest URL. You can directly ignore the ingest URL. You only need to obtain the stream pulling URL based on your specified ingest URL.

Step 3: Integrate and use ApsaraVideo Player SDK.

ApsaraVideo Player SDK can parse SEI signals. You can directly integrate and use ApsaraVideo Player SDK. For more information about the download link, see SDK download.

- Android

- Supported operating system versions

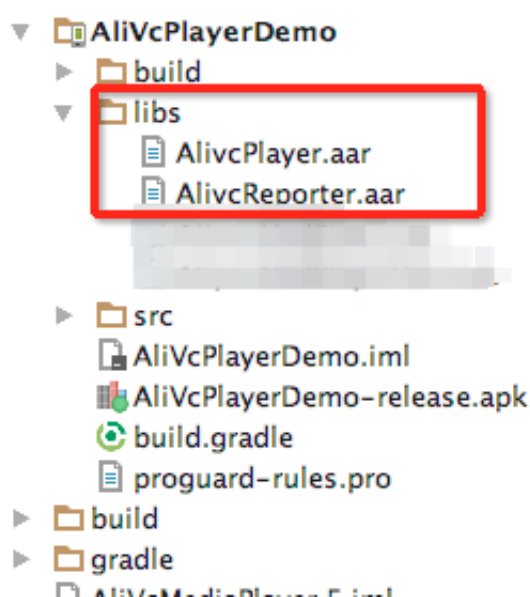
ApsaraVideo Player SDK supports Android 4.0 or later. The mobile phone chip must use the ARMv7 or ARM64 architecture.

- Runtime environment

We recommend that you use Android Studio as the development tool. This topic is written based on the Android Studio development environment.

- How to import ApsaraVideo Player SDK

1. Add the .aar files shown in the following figure to the libs folder.



2. Add a reference to the .aar files in the Gradle of the application.



- How to use the player

Add the following methods to the activity that needs to use ApsaraVideo Player SDK:

```

// Initialize the player . ( You only need to call
// this method once . We recommend that you initialize
// the player in the application .)
AliVcMedia Player . init ( getApplicationContext () );

// Create a player instance .
mPlayer = new AliVcMedia Player ( this , mSurfaceView );
mPlayer . setSEIData Listener ( new MediaPlayer .
    MediaPlayerSEIDataListener () {
    @Override
    public void onSEI_user UnregisteredData ( String data )
    {
    // Parse the received user data in the live
    stream . For more information , see Step 5 : The
    player parses the SEI .
        parserData ( data );
    }
    });
mPlayer . setMaxBufferDuration ( 3 * 1000 );

```

```
mPlayer . prepareAnd Play ( liveUrl );// Prepare and play  
the live stream .
```

For more information about how to use the Android player, see [ApsaraVideo Player Basic](#). For more information about the features and APIs of the Android player, see [API Reference](#).

.

- iOS

- Supported operating system versions

ApsaraVideo Player SDK supports iOS 8.0 or later.

- Runtime environment

We recommend that you use Xcode 8.0 or later for code compiling.

- How to import ApsaraVideo Player SDK

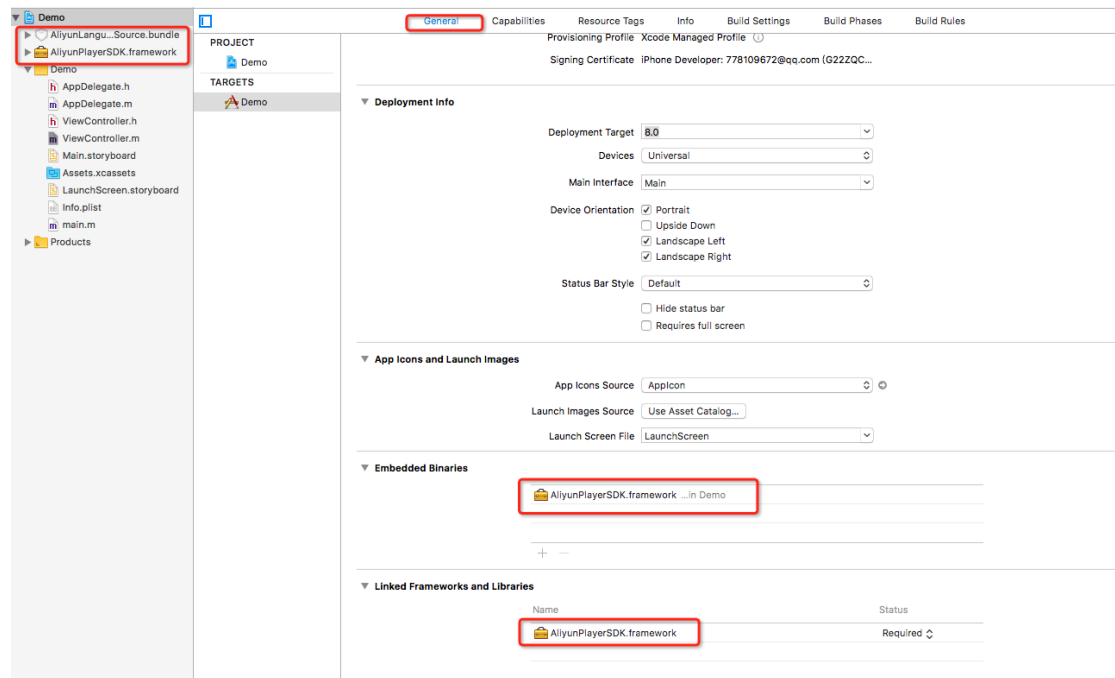
ApsaraVideo Player SDK supports both normal import and CocoaPods-based import methods. You can choose one method as needed.

1. Normal import

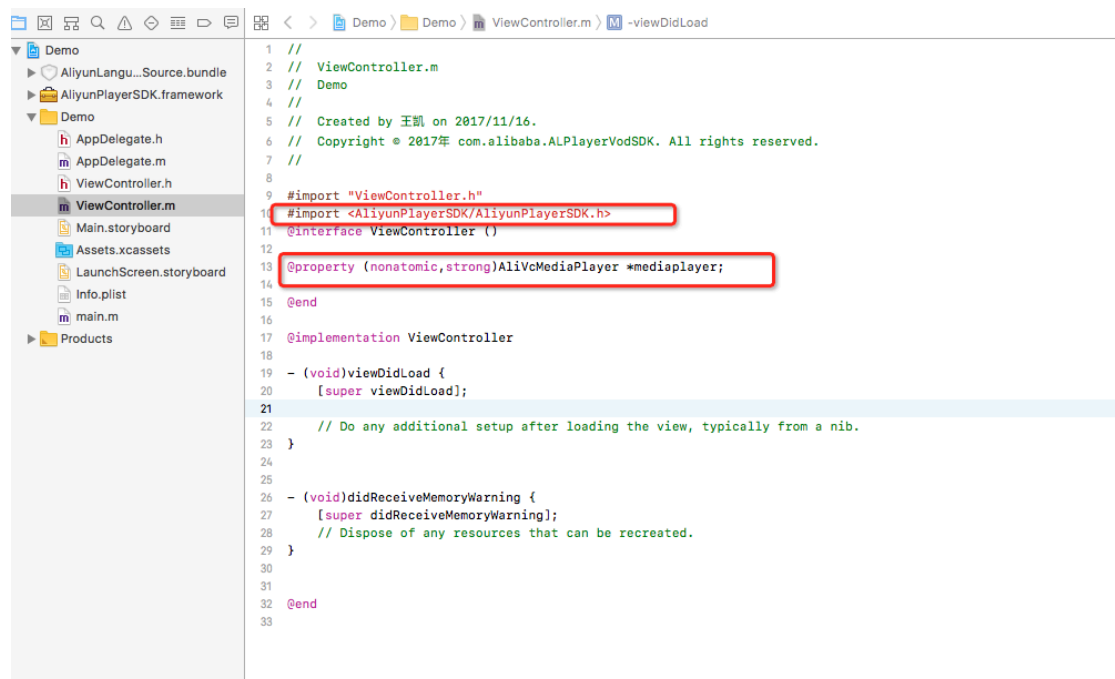
```
Import AliyunPlayerSDK . framework and AliyunLang  
uageSource . bundle to the project .
```

Open the project, select the target, and choose General > Embedded Binaries. Click the plus sign (+), and then select Add Other.... For ApsaraVideo Player

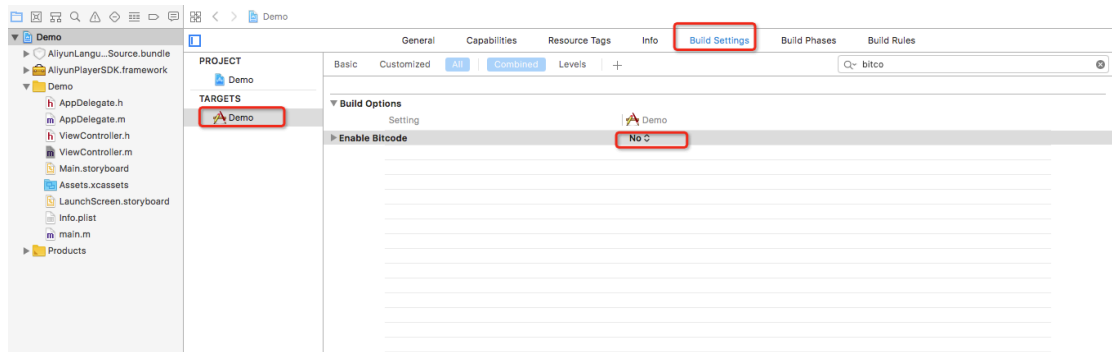
Basic, you only need to import the downloaded AliyunPlayerSDK.framework to the project and select the copy option.



Import the header file.



Choose Build Settings > Build Options and set Enable Bitcode to No.



2. CocoaPods-based import

Add the following statement to your Podfile:

```
pod 'AliyunPlayer_iOS / AliyunPlayerSDK'
```



Run the pod install or pod update command to integrate the SDK into the project.

Run the pod install command each time you add, delete, or update your Podfile.

Run the `project name .xcworkspace` file to open the project.

- How to use the player

```
-( AliVcmedia player *) mediaplayer {
    if (! _mediaplayer ) {
        // 1. Create a player instance .
        _mediaplayer = [[ AliVcMedia Player alloc ] init ];
    }
    return _mediaplayer ;
}

-( void ) player {

    self . contentView . frame = CGRectMake ( 0 , 0 , self .
view . width , self . view . height - 64 );
[ self . view addSubview : self . contentView ];
[ self . mediaPlayer create : self . contentView ];
self . mediaPlayer . scalingMode = scalingModeAspectFit ;
self . mediaPlayer . circlePlay = YES ;
self . mediaPlayer . dropBuffer Duration = 3000 ;
self . mediaPlayer . printLog = YES ;
AliVcMovie ErrorCode err = [ self . mediaPlayer
prepareToPlay : [ NSURL URLWithString : self . playUrl ] ];
```

```
// 5 . Check for an error to determine whether the
    stream can be played back .
    if ( err != ALIVC_SUCCESS ) {
        NSLog (@" chinese - playback failed ");
    } else {
        [ self . mediaPlayer play ];
    }
}
```

For more information about how to use the iOS player, see [ApsaraVideo Player Basic](#). For more information about the features and APIs of the iOS player, see [API Reference](#).

Step 4: Deliver a question through the application server.

When the caster is ready to broadcast a question, the on-site director needs to call an API of the application server to deliver the question. The API used for delivering the question contains the following parameters:

- `liveId` : the unique ID of the live Q&A activity.
- `questionId` : the ID of the question to be delivered.
- `expiredSeconds` : the validity period for submitting the answer to the question, in seconds. Only the answers submitted within the specified period are valid.

The application server completes the following actions after receiving the request:

- Finds the live stream information based on `liveId` , and calls an ApsaraVideo API to insert custom SEI signals (containing `questionId`) into the live stream. For more information about the API, see the API definition in the live Q&A scheme.
- Initializes the Q&A session. A Q&A session includes the content of the question, the absolute timestamp when answer submission expires, and the number of users that select each option.

Examples

```
https://live.aliyuncs.com?Action=AddTrancodeSEI&Domain=
test101.cdnpe.com&App=xxx&Stream=xxx&Text=hahaha&
Pattern=frame&Repeat=300&Delay=0<commonrequest
parameters>
```



Note:

In the live Q&A scenario, we recommend that you set `Pattern` to `frame` and `Repeat` to three times the keyframe interval. (In the example, the value of `Repeat` is 300.) These settings ensure that the player can obtain the SEI once it receives

one frame in three keyframe intervals no matter whether frames are lost in such intervals. Because the SEI in frames may be duplicate with each other, the player needs to deduplicate the SEI.

Considerations

- Because stream ingest has a fixed delay and the on-site director uses a cloud service to insert SEI signals, SEI signals may be inserted into an image earlier than expected. To resolve this issue, the on-site director needs to measure the stream ingest delay in advance and adds a fixed delay when inserting SEI signals into the live stream.
- To measure the stream ingest delay, the on-site director can place a stopwatch on the screen. For example, if an SEI signal is inserted at T_0 and the player detects the SEI signal when it plays the image with the clock time T_s , the stream ingest delay is $T_0 - T_s$ ($T_s < T_0$).

Assume that the on-site caster raises a question at 12:00:00 and that there is a 1-second delay for ingesting streams from the site to Alibaba Cloud. In this scenario, the on-site director needs to call the API to insert SEI signals at 12:00:01. Alternatively, the on-site director can call the API at 12:00:00 and set `delay` to `1000` milliseconds in the API. This ensures that the SEI signals are inserted into the image just at the right time when the caster raises the question.

- After receiving the SEI signals, the client needs to obtain the question from the application server, which consumes a certain amount of communication time. When delivering a question, increase the value of `expiredSeconds` as needed, for example, increase it by 1 second.

Step 5: The player parses the SEI.

The player listens on the callback of SEI signals. You can use the SEI callback API of the player to obtain the SEI delivered from the application server.

- Parse the SEI on an Android device.

1. Set the callback API.

setSEIDataListener

```
public void setSEIDataListener ( MediaPlayerSEIDataListener)
```

```
listener )
```

2. Listen on the callback of SEI signals.

```
MediaPlayer.MediaPlayerSEIDataListener public void onSEIData (
    String data )
```

- Parse the SEI on an iOS device.

1. Add a listening notification API.

```
// The notification received during live Q & A .
[[ NotificationCenter defaultCenter ] addObserver :
    self
                                selector :@ selector (
    onSeiData :)
                                name : AliVcMedia
    PlayerSeiDataNotification object : self . mediaPlayer ];
```

2. Process the SEI in the listening notification.

```
- ( void ) onSeiData :( NSDictionary *) notification {
    NSDictionary * dict = [ notification userInfo ];
    if ( dict ) {
        NSString * seiData = [ dict objectForKey :@" seiData "];
        if ( seiData ) {
            NSLog (@" sei data is %@", seiData );
        }
    }
}
```

Step 6: The application obtains and displays the complete question content.

After obtaining the `questionId` contained in the SEI, the application sends a request (containing `liveId` and `questionId`) to the application server to obtain the question content.

After receiving the request, the application server sends the application a response containing the question content and `remainSeconds`. The `remainSeconds` parameter specifies the number of seconds left for submitting the answer to the question. The value of `remainSeconds` is calculated by the following formula:

```
Value of remainSeconds = Absolute timestamp when answer
    submission expires - Current absolute timestamp
```

Step 7: The application counts down the seconds and reports the answers submitted by users to the application server.

- After receiving the response, the application displays the question content and starts to count down based on the value of `remainSeconds`. Only the users who correctly answered the previous question can select an option to answer the

current question. The users who incorrectly answered the previous question have failed the Q&A quiz and therefore cannot answer the current question.

- After a user selects an option, the application sends a request (containing `liveId`, `questionId`, `answerId`, and `userId`) to report the user's answer to the application server. The user cannot change the option.
- The users who do not select any options this time are considered to fail the Q&A quiz and therefore cannot answer the next question. No answer is reported for such users.
- After reporting the answers to a question, the application records them as the Q&A results for determining whether a user can answer the next question.

Step 8: The application server collects the answers.

After receiving the answers reported by the application, the application server needs to perform the following actions:

- Checks for the Q&A session based on `liveId` and `questionId` in the answer reporting request. If the specified Q&A session does not exist, the application server rejects the request.
- Checks whether the current absolute timestamp is later than the absolute timestamp when answer submission expires. If so, the answer submission times out and the answer becomes invalid.



Note:

Considering that there is a certain network delay for the application to upload answers, you can extend the timeout interval by a certain period of time, for example, by 2 seconds.

- Collects statistics on the number of users who correctly answer the question. The number increases by 1 for each user who correctly answers the question.

Step 9: Deliver the Q&A results.

When the caster announces the expiration of answer submission and is ready to announce the Q&A results, the on-site director needs to call an API of the application server to deliver the Q&A results. The call request contains `liveId` and `questionId`.

Then, the application server calls an ApsaraVideo API to insert the SEI signal (containing `questionId`) into the live stream to instruct the application to request the Q&A results.

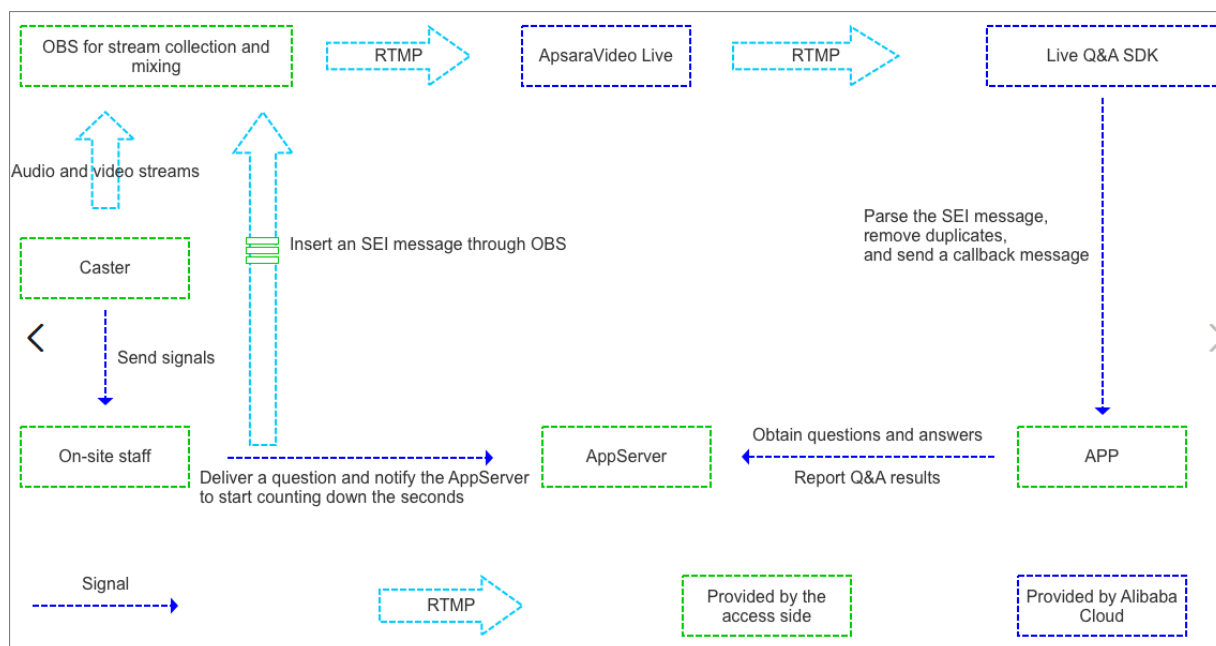
Step 10: The application displays the Q&A results and the Q&A quiz goes to the next question.

- When the application player obtains the SEI signal about the Q&A summary, the application checks whether the SEI signal is about the Q&A or Q&A summary. If the SEI signal is about the Q&A summary, the application sends a request containing `liveId` and `questionId` to the application server to obtain the Q&A summary.
- After receiving the Q&A summary, the application displays a summary dialog box , showing whether the answer is correct and the number of users who select each option.
- The summary dialog box is hidden after being displayed for a certain period of time. Then the next round of Q&A starts.

1.3 OBS-based live Q&A

I. Live Q&A process

To start live Q&A, you can insert SEI (containing question information) into live streams through the ingest SDK, custom OBS, or ApsaraVideo API. When the player parses the SEI, it sends a callback message to a user's application. Then the application requests the question content and displays it to the user. The following figure shows the live Q&A process.



**Note:**

For more information about how to start live Q&A, see [Introduction](#). Submit a ticket to apply for live Q&A according to the instructions. For more information about how to use the player, see [ApsaraVideo Player Basic](#).

II. OBS installation instructions

Install OBS on Mac devices

1. Download OBS and run it directly. For more information about the download link, see [SDK download](#).
2. A trust relationship needs to be established for installing OBS. To achieve this, choose System Preferences > Security & Privacy and click Open Anyway.



Install OBS on Windows devices

Download OBS, double-click the installation file to install it as prompted, and then run it. For more information about the download link, see [SDK download](#).

III. Parameter configuration

1. Start OBS and click Settings in the lower-right corner. In the Settings dialog box that appears, select Output on the left and confirm that the encoder in use is x264. Live Q&A activities are mainly performed on mobile phones. You need to adjust the output resolution to 9:16. To adapt to large-screen mobile phones, you can change the resolution to 540 × 960 pixels and adjust the bitrate accordingly.

**Note:**

To save traffic and ensure smooth playback, select the appropriate bitrate.



Recommended parameter values

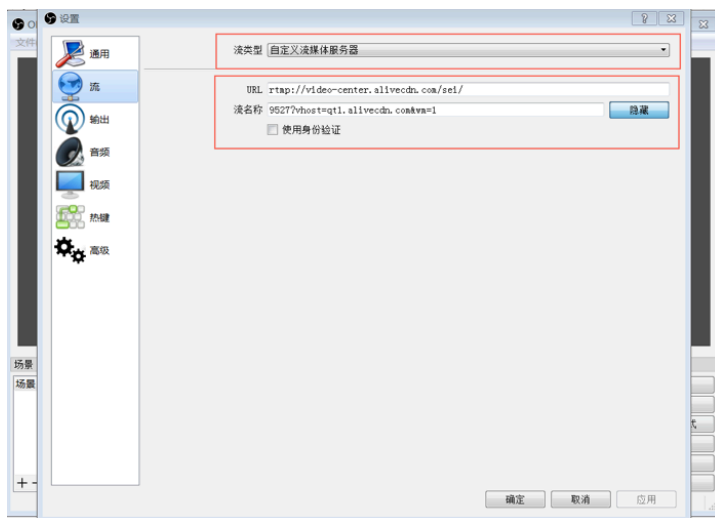
- Bitrate: 600
- Preset CPU usage: faster
- Keyframe interval: 1
- Profile: high
- Frames may be lost when video streams are transmitted to terminals over the network. We recommend that you insert SEI into live streams multiple times to minimize the impact of frame loss. SEI is inserted five times by default. You can change the setting as needed.

Recommended resolutions

Resolution	Bitrate	Frame rate
360 × 640 pixels	700 Kbit/s	20
540 × 960 pixels	1,000 Kbit/s	25

2. Select Stream on the left and set the RTMP ingest URL. Take the live center ingest URL `rtmp://video-center.alivecdn.com/sei/stream01?vhost=qt1.alivecdn.com` as an example. In the dialog box, set the URL to `rtmp://video-center.alivecdn.com/sei/`, which is the part from the start to the application name of the ingest URL provided by ApsaraVideo.

Set the stream name to `{ streamName }? vhost = qt1 . alivecdn . com` , which consists of the stream name and access token of the ingest URL provided by ApsaraVideo. Set `{ streamName }` based on the business needs.

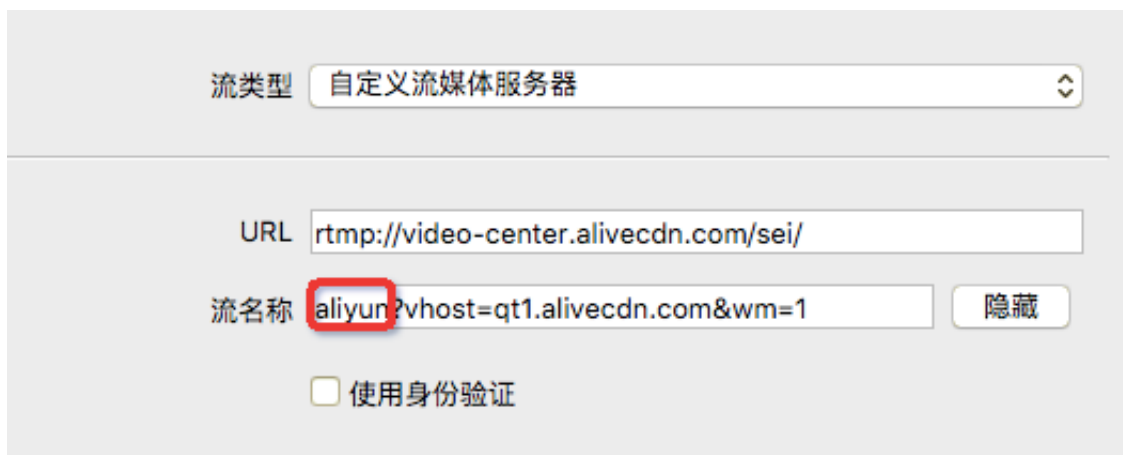


Note:

If you want to insert SEI signals into live streams on ApsaraVideo Live, you must add `wm=1` to the ingest URL. If you want to insert SEI signals into video streams through OBS, you do not need to add `wm=1` to the ingest URL.

IV. Q&A page

In the lower-right corner of the OBS page, click the live Q&A button. In the dialog box that appears, enter the Q&A web page address to load the Q&A page. The stream name must be the same as that you set in OBS. We recommend that you enter different stream names in different tests.



You can use internal OBS APIs to insert SEI signals into videos on the Q&A page. Then click the relevant buttons to deliver a question and its answer based on the signals

sent by the on-site caster. For more information about the page code, see [Deliver questions](#).



Note:

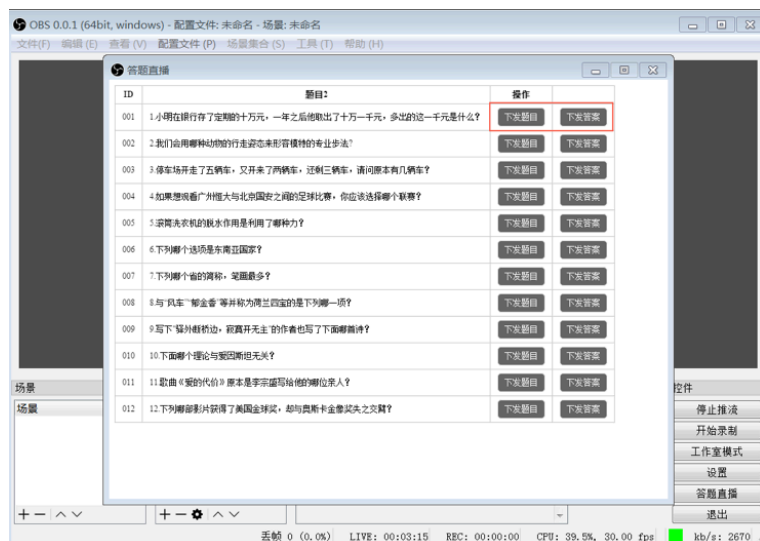
The network request parameters on the Q&A page include the stream ID. In the demo, the stream ID on the Q&A page is passed in through the streamName parameter in the ingest URL. This stream ID must be the same as the value of {streamName} in the ingest URL. In the demo, the stream name is aliyun. Replace it with the actual name of your ingest stream.



V. Scene configuration and live streaming

1. In the preview window, verify that the camera is collecting data properly and that the live scene is ready. Then click Start Ingest in the lower-right corner of the OBS page to start live streaming. For more information about the OBS configuration, see [OBS](#).
2. After the live streaming starts, the director inserts SEI into the Q&A web page based on the signals sent by the caster. When the caster starts broadcasting a question, the director clicks the Deliver question button next to the question to insert SEI into the video stream. Similarly, when the caster starts broadcasting

the answer to a question, the director clicks the Deliver answer button next to the question to display the Q&A result on the playback device.



VI. Considerations

- How do I insert SEI into a video stream in OBS on the Q&A page?

In the JavaScript section on the Q&A page, you must introduce the [qwebchannel.js](#) file of Qt. This file is used for calling back a local OBS object in the JavaScript code to insert SEI into the video stream. For example, when you click the question delivery button, the QWebChannel object is created to obtain the brige object built in locally. Then the sendSEI method of the brige object is called to insert the SEI. Therefore, when modifying the Q&A page, you must ensure that the qwebchannel.js file is introduced. In addition, ensure that the callback object name and method name are consistent. If any of the conditions is not met, the callback will fail.

- How do I organize the SEI on the Q&A page?

The SEI on the Q&A page includes question IDs, question content, and operations for delivering questions and answers. When you click the button to deliver a question or an answer, the SEI callback method is triggered to organize the SEI into a JSON string. Then the JSON string is sent to OBS for insertion into the video stream. In the Q&A page template, the JSON format used for delivering a question is as follows:

```
{
  "questionId": " 001 ", // The ID of the question to
  be delivered .
  "type": " startAnsw e r ",
```

```

    showTime : 15 // The time for displaying the answer
to the question .
}

```

After inserting the SEI, OBS needs to notify the application server of the due time for answering the question. Any answer submitted after the due time is invalid. The notification request is in the following format:

```

{
    liveId : seiLiveId ,
    questionId : qid ,
    expiredSeconds : 15 ,
    seiDelay : 2000 ,
    noSEI : true
}

```

The JSON format used for delivering an answer is as follows:

```

{
    "questionId": "001",
    "type": "showResult",
    "showTime": "5"
}

```

You can modify the Q&A page as needed based on the current JSON organization format. After receiving the SEI, the playback device requests the detailed question and answer information from the application server based on the question ID. Therefore, when editing the SEI, ensure that the question ID matches the question and answer information. Currently, the question ID corresponding to each question or answer delivery button on the Q&A page is set in the link attribute data-role.

- How is the application server notified of the delivered question and answer information?

When a question and its answer are delivered on the Q&A page, a POST request is sent to notify the application server through JavaScript code. When modifying the Q&A page, ensure that the endpoint corresponding to the request is updated to the correct address. In addition, the request contains the corresponding ingest stream ID, which must be the same as the custom value of liveId in the ingest stream name in step 2. In the demo, the value of liveId in the request body is in `qt1` . `alivecdn . com / sei /{ streamName }` format. The value of liveId is passed in through the streamName parameter in the ingest URL. For more information, see the stream configuration in the parameter configuration section.

- How do I watch live Q&A?

After confirming that the preceding operations are completed, ingest the stream in OBS. Currently, ApsaraVideo provides the latest ApsaraVideo Player SDK version to support live Q&A. You can download the demo to experience live Q&A. For more information, see [Download ApsaraVideo Player SDK for live Q&A](#).



2 How to record live videos

The recording service of ApsaraVideo Live converts live videos to on-demand videos, stores and manages the on-demand videos. You can store the recorded videos to OSS.

Recording by scenarios

You can perform recording configuration based on your needs and different scenarios

.

- Recording at domain name, AppName and StreamName levels

You can record exactly a specific stream by entering a specific domain name, AppName, and StreamName. For more information, see [Store live recordings to OSS](#).

- Recording at specified time

You can use API to control the start time and end time of recording, so as to record at the specified time. For more information, see [AddLiveAppRecordConfig](#).

- Recording based on your needs

You can configure a callback URL to exactly control the recording method of each stream. For more information, see [On-demand recording](#).

- Manual recording

The manual recording is disabled by default. You can use the API to record videos. For more information, see [Manual recording](#).

Recording based on storage position

You can store live recordings to OSS.

Store live recordings to OSS

The live recordings stored to OSS can serve the purpose of storage and playback. For more information, see [Store live recordings to OSS](#).

- If you want to preview and manage your recorded videos, you can preview and manage in the recording management list in the ApsaraVideo live console or in OSS console. For more information, see [View recorded files](#).
- If you want to deliver recorded videos, you can log on to the OSS console to configure a CDN domain name. CDN delivers your videos stored in OSS to nodes

across China. And users can access the nearest nodes to read files, which not only improves the access speed and experience of edge users, but also effectively saves the overall network cost. For more information, see [Configure OSS](#).

**Note:**

- You can record videos in flv format, mp4 format, and hls format.
- The recorded videos are original videos.
- You can index and edit the recorded hls files to quickly splice m3u8 video clips.
- The information of recorded files are stored for 6 months by default. Within 6 months, you can use API or the console to view recorded files. If the storage time exceeds 6 months, you must go to OSS to get data.

3 Pure live audio and live video

Alibaba Cloud supports pure live audio and live video. You can push pure live audio streams or pure live video streams to Alibaba Cloud video center. You can also directly play pure live audio streams or pure live video streams.

- Streaming

Audio and video streaming based on RTMP protocol are supported.

- Stream play

- You can add parameters at the end of the address to separately play pure audio or pure video streams.

- Play audio only: `onlyaudio=1`

- Play video only: `onlyvideo=1`

- Examples

- RTMP format : `rtmp:// domain name / APPName / Streamname ? onlyaudio = 1`

- FLV format : `http:// domain name / APPName / Streamname . flv ? onlyaudio = 1`



Note:

Only stream play addresses in FLV format and RTMP format are supported.

- Scenarios

- Audio and video streaming

- `rtmp:// streaming domain name / APPName / Streamname ?`

- Enter pure audio or video addresses, and you can play pure audio or video stream.

- Play audio only: `RTMP format : rtmp:// stream play domain name / APPName / Streamname ? onlyaudio = 1`

- Play video only: `FLV format : http:// stream play domain name / APPName / Streamname . flv ? onlyvideo = 1`

4 How to perform overseas live activities

Overseas live activities includes two types:

- Streamer is abroad or in Hong Kong, Macao or Taiwan, and audience is at home and abroad
- Streamer is at home, and audience is at home and abroad or in Hong Kong, Macao or Taiwan

We provide solutions in different scenarios.

Streamer is abroad or in Hong Kong, Macao or Taiwan, and audience is at home

Problems

The cross-border transmission is unstable, and the live video lagging rate is high.

Solution

ApsaraVideo Live provides premium streaming function, which uses stable leased line to transmits video streams to mainland China. You can use premium streaming function to solve the problems of unstable transmission and high video lagging rate. Perform the following steps to complete the procedure.

1. Add the domain name in China (Shanghai) region.



Note:

Premium streaming service is provided only in China (Shanghai) region. Add your domain name in China (Shanghai) region.

2. In Premium Streaming Acceleration Settings , configure the premium streaming function. For more information, see [Configure premium streaming](#) .

Attentions

For videos played in mainland China, the domain name must perform ICP registration. For more information, see [ICP Registration Support](#).

Streamer is abroad or in Hong Kong, Macao or Taiwan, and audience is abroad or in Hong Kong, Macao or Taiwan

Problems

- The audience are from a wide range, which the videos can not cover.

- The streamer is not in a specific location, and the network transmission is unstable , which leads to bad live video quality.

Solution

Currently, Alibaba Cloud has overseas centers in Germany, Singapore and Japan.

We recommend that you select a live center near the streamer, and stream the video streams to the live center nearby.

1. Edge ingestion.

Alibaba Cloud CDN nodes are distributed across main stream countries across the world. Edge ingestion service can push video streams to edge nodes closest to the streamer, and uses the network optimized by Alibaba Cloud to transmit the video streams to the live center. For more information, see [Configure edge ingestion](#).

2. Real-time network monitoring.

- To view the streaming status and the network condition of the streamer and ensure the stability of streams, you can log on to the [ApsaraVideo Live console](#) , and go to Stream Management > Ingest Endpoints > Stream Monitoring to ensure stability of streams. For more information, see [View frame rate and bit rate](#).
- You can also use API to get frame rate and bit rate. For more information, see [DescribeLiveStreamsFrameRateAndBitRateData](#).

Attentions

Alibaba Cloud has rich CDN nodes at home and abroad, covering more than 70 countries and regions in 6 continents. All the main stream carriers support overseas watching. However, when adding a live domain name, you must select Globe or Outside Mainland China.

< Add Domain

* Domain
Name Wildcard domains, such as *.test.com, are supported. [Learn more](#)

* Live Center

* Domain Type

* Edge Group
Different charges apply for mainland China and regions outside mainland China. Please select a region that fits your needs.

Streamer is at home, and audience is abroad or in Hong Kong, Macao or Taiwan

Problems

- How can I perform overseas acceleration for domestic videos?
- How can I ensure the important videos are transmitted to overseas quickly and stably?

Solution

1. When adding a domain name, select Globe or Outside Mainland China. For more information about adding a domain name, see [Add a domain name](#).
2. To transmit video streams to the region where the video watching quality must be ensured, you can log on to the [ApsaraVideo Live console](#).
3. Go to Domains > Stream Settings > Premium Streaming to enable the premium streaming function, and select Live Streaming in Applies to. For more information, see [Configure premium streaming](#).

5 Live video security

5.1 Authentication code example

Overview

For more information about URL authentication rules, see [Configure authentication](#).

Through this demo you can authenticate your URLs according to your needs. The following Python demo contains A authentication.

Python

```
import re
import time
import hashlib
import datetime
def md5sum ( src ):
    m = hashlib . md5 ()
    m . update ( src )
    return m . hexdigest ()
def a_auth ( uri , key , exp ):
    p = re . compile ("^( rtmp ://)?([^\?/]+)(/[^\?]*)?(\\?.*)?$")
    if not p :
        return None
    m = p . match ( uri )
    scheme , host , path , args = m . groups ()
    if not scheme : scheme = " rtmp ://"
    if not path : path = "/"
    if not args : args = ""
    rand = " 0 "      # " 0 " by default , other value is
ok
uid = " 0 "          # " 0 " by default , other value is
ok
sstring = "% s -% s -% s -% s -% s " %( path , exp , rand ,
uid , key )
hashvalue = md5sum ( sstring )
auth_key = "% s -% s -% s -% s " %( exp , rand , uid ,
hashvalue )
if args :
    return "% s % s % s % s & auth_key =% s " %( scheme , host
, path , args , auth_key )
else :
    return "% s % s % s % s ? auth_key =% s " %( scheme , host
, path , args , auth_key )
def main ():
    uri = " rtmp :// video - center . alivcdn . com / test / test
? vhost = xc . cdnpe . com "      # original uri
    key = "< input private key >"      #
private key of authorization
    exp = int ( time . time () ) + 1 * 3600      #
expiration time : 1 hour after current itme
    authuri = a_auth ( uri , key , exp )      #
auth type :
    print ( " URL : % s \ nAUTH : % s " %( uri , authuri ))
if __name__ == " __main__ ":
```

```
main ()
```