

云视频播放器 Web SDK 参考文档

VERSION 2015-10

UCLLOUD

专业云计算服务厂商 <http://www.ucloud.cn>

目录

1 简介和云视频.....	3
1.1 概述.....	3
1.2 阅读对象.....	3
2 功能特点.....	4
2.1 系统功能.....	4
2.2 播放器功能.....	4
3 SDK 使用说明.....	5
3.1 引用 SDK.....	5
3.2 创建 SDK 显示容器.....	5
3.3 创建 SDK 实例.....	5
4 SDK 使用进阶.....	6
4.1 接收 SDK 消息事件.....	6
4.2 调用 SDK 接口方法.....	6
5 点播播放器 API 介绍.....	7
5.1 控制接口.....	7
5.1.1 ump.Player.....	7
5.1.2 Player::create.....	7
5.1.3 Player::load.....	7
5.1.4 Player::play.....	7
5.1.5 Player::pause.....	8
5.1.6 Player::stop.....	8
5.1.7 Player::seek.....	8
5.1.8 Player::getMute.....	8
5.1.9 Player::setMute.....	8
5.1.10 Player::getVolume.....	8
5.1.11 Player::setVolume.....	9
5.1.12 Player::getFullscreen.....	9
5.1.13 Player::setFullscreen.....	9
5.1.14 Player::getState.....	9
5.1.15 Player::getDuration.....	9
5.1.16 Player::getPosition.....	10
5.1.17 Player::setBulletData.....	10
5.2 事件响应接口.....	10
5.2.1 Player::onInited.....	10
5.2.2 Player::onError.....	10
6 直播播放器 API 介绍.....	11
6.1 控制接口.....	11
6.1.1 ump.Player1vN.....	11
6.1.2 Player1vN::create.....	11
6.1.3 Player1vN::joinRoom.....	11
6.1.4 Player1vN::stop.....	12
6.1.5 Player1vN::getMute.....	12

6.1.6	Player1vN::setMute.....	12
6.1.7	Player1vN::getVolume.....	12
6.1.8	Player1vN::setVolume.....	12
6.1.9	Player1vN::getState.....	12
6.2	事件响应接口.....	13
6.2.1	Player1vN::onInit.....	13
6.2.2	Player1vN::onError.....	13
6.2.3	Player1vN::onState.....	13
6.2.4	Player1vN::onUserAction.....	14
7	SDK 升级历史信息文档变更.....	15
8	反馈和建议.....	16

1 简介和云视频

1.1 概述

UCloud 云计算（上海优刻得信息科技有限公司）是国内技术与服务最为顶尖的基础云服务商，公司专注于基础云计算的产品研发与运营，是上海市云计算基地重点企业，国家工信部首批认证通过的“可信云计算企业”。

我们围绕用户的需求持续创新，通过整合优秀的合作伙伴资源，深耕互联网领域，打造全天候 90 秒内快速响应的服务体系，在国内云计算领域构筑了独树一帜的竞争优势。

UCloud 云计算平台已经为游戏、移动互联网、大数据、电子商务、Saas 等多个领域提供 IT 基础架构支撑，得益于业内领先的云计算技术和技术团队的专业经验，UCloud 已成功为国内外上万家企业用户提供服务，大幅降低了用户使用 IT 基础设施的成本及技术门槛。

UCloud 云视频 Web SDK（以下简称“云视频 SDK”）是上海优科得信息科技有限公司推出运行于浏览器中的软件开发工具（SDK），为网站开发者提供简单、快捷的接口，帮助开发者实现浏览器上的媒体相关应用开发。

云视频 SDK 的完整下载包中包含 docs、samples 两部分，解压后的目录结构如下所示：

- |--demos : 存放使用 SDK 制作的演示链接地址。
- |--docs : 存放播放器 Web SDK 使用手册和相关 API 文档。
- |--samples : 存放使用 Web SDK 的代码样例。

1.2 阅读对象

本文档面向所有使用该 SDK 的开发人员、测试人员、合作伙伴以及对此感兴趣的其他用户，要求读者具有一定的 HTML 和 Javascript 编程经验。

2 功能特点

2.1 系统功能

支持功能	内容
FlashPlayer 版本	支持 FlashPlayer 10.1 版本或以上
流媒体格式	HTTP Streaming、RTMP
封装格式	FLV、MP4

2.2 播放器功能

功能模块	说明
云视频点播播放器	使用 flash 开发，支持 FLV/MP4 渐进式和流式下载播放，支持基于 RTMP 视频流播放，包含基础播放、多清晰度、弹幕等功能。
云视频直播播放器	专门用于直播的播放器，支持 RTMP 协议。

3 SDK 使用说明

使用 SDK 只需以下简单三个步骤，本章节以云视频播放器为例讲解，其他功能模块使用方法与云视频播放器类似，更多功能接口请查看对应的 API 文档。

3.1 引用 SDK

```
<script type="text/javascript"
src="http://player.ucloud.com.cn/release/js/ump.player_v1.min.js"></script>
```

3.2 创建 SDK 显示容器

在 html 页面中要显示的位置插入以下代码

```
<div id="mod_player"></div>
```

3.3 创建 SDK 实例

```
//创建播放器实例
var player = new ump.Player();

//设置传递给播放器的参数
var params = {
    autostart:true,           //是否自动播放
    file:"http://mediademo.ufile.ucloud.com.cn/ucloud_promo_140s.mp4", //文件 url
    markname:"视频文字水印"  //视频上的随机文字水印
};

player.create({
    width : 600,              //播放器宽度
    height : 400,             //播放器高度
    modId : "mod_player",    //填写播放器对应的 div id
    params : params           //传递给播放器的参数
});
```

4 SDK 使用进阶

4.1 接收 SDK 消息事件

```
//建议接收到 onInited 消息事件之后再执行播放器接口方法
player.onInited = function() {
    ump.log("播放器初始化完成!");
};
```

4.2 调用 SDK 接口方法

```
//加载指定 url 文件
player.load("http://mediademo.ufile.ucloud.com.cn/ucloud_promo_140s.mp4");
```

5 点播播放器 API 介绍

5.1 控制接口

5.1.1 ump.Player

参数列表：无

返回类型：Object

接口功能：创建播放器实例

代码示例：

```
var player = new ump.Player();
```

5.1.2 Player::create

参数列表：width 播放器宽度,height 播放器高度,
modId 填写播放器对应的 div id,params 播放文件的参数

返回类型：无

接口功能：创建播放器并显示在指定的 div 元素上

代码示例：

```
Player.create({
    width : 600,
    height : 400,
    modId : "mod_player",
    params : params
});
```

参数详解：

create 只有一个 Object 类型的参数 params，该参数是一个参数集合，该参数集合所包含的参数元素及使用

方法如下描述：

参数名称	参数描述	备注
autostart	设置是否在播放器载入后自动播放：true: 自动播放；false：不自动播放；默认为 false	可选
file	设置媒体流名称或文件名或 mp4 播放地址	必选
markname	设置视频水印文字	可选
bulleturl	设置弹幕文件地址	可选

5.1.3 Player::load

参数列表：视频 url 地址

返回类型：无

接口功能：加载视频信息

代码示例：

```
player.load("http://xxxx.ucloud.com.cn/sy_sd.mp4");
```

5.1.4 Player::play

参数列表：无

返回类型：无

接口功能：开始播放

代码示例：

```
player.play();
```

5.1.5 Player::pause

参数列表：无

返回类型：无

接口功能：暂停播放

代码示例：

```
player.pause();
```

5.1.6 Player::stop

参数列表：无

返回类型：无

接口功能：停止播放

代码示例：

```
player.stop();
```

5.1.7 Player::seek

参数列表：number- 目标播放时间

返回类型：无

接口功能：定位当前媒体内容开始播放的位置

代码示例：

```
player.seek(num);
```

5.1.8 Player::getMute

参数列表：无

返回类型：Boolean

接口功能：获取当前是否为静音状态

代码示例：

```
player.getMute();
```

5.1.9 Player::setMute

参数列表：Boolean - true：静音；false：非静音

返回类型：无

接口功能：设置播放器是否静音

代码示例：

```
player.setMute(isMute);
```

5.1.10 Player::getVolume

参数列表：无

返回类型：Number

接口功能：获取当前播放视频的音量

代码示例：

```
player.getVolume();
```

5.1.11 Player::setVolume

参数列表：Number- volume 音量,0 至 100

返回类型：无

接口功能：设置播放器音量

代码示例：

```
player.setVolume(volume);
```

5.1.12 Player::getFullscreen

参数列表：无

返回类型：Boolean

接口功能：获取当前播放器是否为全屏状态

代码示例：

```
player.getFullscreen();
```

5.1.13 Player::setFullscreen

参数列表：Boolean - true：全屏，false：非全屏

返回类型：无

接口功能：设置播放器是否全屏播放

代码示例：

```
player.setFullscreen(flag);
```

5.1.14 Player::getState

参数列表：无

返回类型：String

接口功能：获取当前播放器状态

代码示例：

```
player.getState();
```

返回值详解：

"BUFFERING":缓冲中

"IDLE":空闲状态

"PLAYING":播放中

"PAUSED":暂停

5.1.15 Player::getDuration

参数列表：无

返回类型：Number

接口功能：获取视频的总时长

代码示例：

```
player.getDuration();
```

5.1.16 Player::getPosition

参数列表：无

返回类型：Number

接口功能：获取当前播放视频的位置

代码示例：

```
player.getPosition();
```

5.1.17 Player::setBulletData

参数列表：Array,

例：[{content:"弹幕评论 1",color:"#ffffff"},{content:"弹幕评论 2",color:"#ffffff"}]

返回类型：无

接口功能：设置当前弹幕内容

代码示例：

```
player.setBulletData(data);
```

5.2 事件响应接口

5.2.1 Player::onInited

参数列表：无

返回类型：无

接口功能：播放器初始化完成后派发的消息事件

代码示例：

```
player.onInited = function(){  
    ump.log("播放器初始化完成！");  
}
```

5.2.2 Player::onError

参数列表：无

返回类型：Object

接口功能：播放器错误消息

代码示例：

```
player.onError= function(info){  
    if(!info){  
        return;  
    }  
    if(info.desc){  
        ump.log("播放器错误提示： "+info.desc);  
    }  
}
```

6 直播播放器 API 介绍

6.1 控制接口

6.1.1 ump.Player1vN

参数列表：无

返回类型：Object

接口功能：创建播放器实例

代码示例：

```
var player = new ump.Player1vN();
```

6.1.2 Player1vN::create

参数列表：width 播放器宽度,height 播放器高度,

modId 填写播放器对应的 div id, params 播放文件的参数

返回类型：无

接口功能：创建播放器并显示在指定的 div 元素上

代码示例：

```
player.create({
    width : 600,
    height : 400,
    modId : "mod_player1vN",
    params : params
});
```

参数详解：

create 只有一个 Object 类型的参数 params ，该参数是一个参数集合，该参数集合所包含的参数元素及使用方法如下描述：

当使用实时直播时，一下参数有效：

参数名称	参数描述	备注
roomId	房间 id，该 id 由客户自己管理	必选
autostart	设置是否在播放器载入后自动播放：true:自动播放；false:不自动播放	可选

当使用 rtmp 直播时，以下参数有效：

参数名称	参数描述	备注
playurl	rtmp 推流地址	必选
autostart	设置是否在播放器载入后自动播放：true:自动播放；false:不自动播放	可选
controlbar	"bottom":控制栏在底部显示,"none":隐藏控制栏。默认为"bottom"	可选

6.1.3 Player1vN::joinRoom

参数列表：Object

返回类型：无

接口功能：加入房间并播放

代码示例:

```
player.joinRoom({  
    roomid:"roomid"  
});
```

6.1.4 Player1vN::stop

参数列表: 无

返回类型: 无

接口功能: 停止播放

代码示例:

```
player.stop();
```

6.1.5 Player1vN::getMute

参数列表: 无

返回类型: Boolean

接口功能: 获取当前是否为静音状态

代码示例:

```
player.getMute();
```

6.1.6 Player1vN::setMute

参数列表: Boolean - true: 静音; false: 非静音

返回类型: 无

接口功能: 设置播放器是否静音

代码示例:

```
player.setMute(isMute);
```

6.1.7 Player1vN::getVolume

参数列表: 无

返回类型: Number

接口功能: 获取当前播放视频的音量

代码示例:

```
player.getVolume();
```

6.1.8 Player1vN::setVolume

参数列表: Number- volume 音量,0 至 100

返回类型: 无

接口功能: 设置播放器音量

代码示例:

```
player.setVolume(volume);
```

6.1.9 Player1vN::getState

参数列表: 无

返回类型: String

接口功能: 获取当前播放器状态

代码示例：

```
player.getState();
```

返回值详解：

"loading":加载中

"playing":播放中

"stop" :停止

(备注:为达到直播最小延迟效果，故没有设置缓冲区大小，因此不会有缓冲中的状态)

6.2 事件响应接口

6.2.1 Player1vN::onInited

参数列表：无

返回类型：无

接口功能：播放器初始化完成后派发的消息事件

代码示例：

```
player.onInited = function(){
    ump.log("播放器初始化完成！");
}
```

6.2.2 Player1vN::onError

参数列表：Object

返回类型：无

接口功能：播放器错误消息

代码示例：

```
player.onError= function(info){
    if(!info){
        return;
    }
    if(info.type){
        ump.log("播放器错误类型："+info.type);
    }
    if(info.desc){
        ump.log("播放器错误描述："+info.desc);
    }
}
```

入口参数详解：

参数名称	参数描述	备注
type	错误类型描述	
desc	错误描述	

6.2.3 Player1vN::onState

参数列表：Object

返回类型：无

接口功能：播放器状态消息

代码示例：

```
player.onState= function(info){
    if(!info || (info && typeof info.type== undefined)){
        return;
    }
    ump.log("播放器状态: "+info.type);
}
```

入口参数详解：

参数名称	参数描述	备注
type	播放状态如下： "loading":加载中 "playing":播放中 "stop":停止 (备注:为达到直播最小延迟效果，故没有设置缓冲区大小，因此不会有缓冲中的状态)	

6.2.4 Player1vN::onUserAction

参数列表：Object

返回类型：无

接口功能：用户主动触发的行为消息，通过代码调用的接口不响应此事件

代码示例：

```
player.onUserAction = function(info){
    if(!info || (info && typeof info.type== undefined)){
        return;
    }
    ump.log("用户动作行为:"+info.type+" 对应的值:"+info.value);
}
```

入口参数详解：

参数名称	参数描述	备注
type	用户行为类型： "playBtnClick":点击播放按钮 "pauseBtnClick":点击暂停按钮 "fullscreen":点击全屏按钮 "normalscreen":点击退出全屏按钮 "muteBtnClick":点击静音按钮 "unmuteBtnClick":点击取消静音按钮 "volumeChange":音量发生变化	
value	相应动作行为对应的值： "volumeChange":对应音量值	

7 SDK 升级历史信息文档变更

版本号	发布日期	说明
1.0	2015.08.12	基础播放功能
1.1	2015.09.15	增加弹幕功能，多清晰度功能
1.2	2015.10.09	升级点播播放器 UI

8 反馈和建议

对于使用中发现的问题和建议进行反馈我们的技术人员会同您联系。