

# Director Mini API

## 前言

文档简介	1.1
HTTP 状态码	1.2
DEMO: 命令行	1.3
版本历史	1.4

## 背景音乐

first	2.1
getPlayList	2.2
last	2.3
next	2.4
pause	2.5
play	2.6
playAction	2.7
playByFile	2.8
playByIndex	2.9
previous	2.10
seekByProgress	2.11

## GFX

actionByIndex	3.1
actionByName	3.2
actionByUUID	3.3
clear	3.4
getGFXList	3.5

## 记分牌

adjustBallStrike	4.1
adjustOut	4.2
clearPitchCount	4.3
firstPeriod	4.4
gameTimeAction	4.5
lastPeriod	4.6
nextPeriod	4.7
pauseGameTime	4.8
playGameTime	4.9
previousPeriod	4.10
reset	4.11
resetGameTime	4.12
setCountdown	4.13
setFootballStoppageTime	4.14
setGameTime	4.15
setOnBaseRunners	4.16
setPeriod	4.17
setScoreOfGuest	4.18

setScoreOfHome	4.19
setTimeFormat	4.20
showBallStrike	4.21
showCountdown	4.22
showFootballStagePlayer	4.23
showFootballStoppageTime	4.24
showGameName	4.25
showGameTime	4.26
showInning	4.27
showOnBaseRunners	4.28
showOut	4.29
updateScoreOfGuest	4.30
updateScoreOfHome	4.31

## 计时器

pause	5.1
play	5.2
playAction	5.3
reset	5.4

## 秒表

pause	6.1
play	6.2
playAction	6.3
reset	6.4

## 网页

downAction	7.1
endAction	7.2
goBack	7.3
goForward	7.4
homeAction	7.5
leftAction	7.6
pageDownAction	7.7
pageUpAction	7.8
reload	7.9
reset	7.10
rightAction	7.11
scale	7.12
upAction	7.13

## 回放

addEvent	8.1
backward	8.2
enterReplayMode	8.3
exitReplayMode	8.4
forward	8.5

<a href="#">getEventList</a>	8.6
<a href="#">muteAudio</a>	8.7
<a href="#">muteMic</a>	8.8
<a href="#">nextFrame</a>	8.9
<a href="#">pause</a>	8.10
<a href="#">play</a>	8.11
<a href="#">playAction</a>	8.12
<a href="#">previousFrame</a>	8.13
<a href="#">replayEvent</a>	8.14
<a href="#">replayFromSecondsAgo</a>	8.15
<a href="#">rewindToStart</a>	8.16
<a href="#">seek</a>	8.17
<a href="#">seekByProgress</a>	8.18
<a href="#">selectLens</a>	8.19
<a href="#">setSpeed</a>	8.20

## PTZ

<a href="#">AIHumanTrackingAction</a>	9.1
<a href="#">autoFocus</a>	9.2
<a href="#">focusExact</a>	9.3
<a href="#">focusFar (已弃用)</a>	9.4
<a href="#">focusNear (已弃用)</a>	9.5
<a href="#">focusStop (已弃用)</a>	9.6
<a href="#">getDeviceList</a>	9.7
<a href="#">goToPreset</a>	9.8
<a href="#">moveDown</a>	9.9
<a href="#">moveDownLeft</a>	9.10
<a href="#">moveDownRight</a>	9.11
<a href="#">moveHome</a>	9.12
<a href="#">moveLeft</a>	9.13
<a href="#">moveRight</a>	9.14
<a href="#">moveStop</a>	9.15
<a href="#">moveUp</a>	9.16
<a href="#">moveUpLeft</a>	9.17
<a href="#">moveUpRight</a>	9.18
<a href="#">recordAction</a>	9.19
<a href="#">reset</a>	9.20
<a href="#">selectedControl</a>	9.21
<a href="#">setHumanTrackingMode</a>	9.22
<a href="#">setHumanTrackingSpeed</a>	9.23
<a href="#">setHumanTrackingType</a>	9.24
<a href="#">startAIHumanTracking</a>	9.25
<a href="#">startRecording</a>	9.26
<a href="#">stopAIHumanTracking</a>	9.27
<a href="#">stopRecording</a>	9.28
<a href="#">storePreset</a>	9.29
<a href="#">wakeUp</a>	9.30
<a href="#">zoomExact</a>	9.31
<a href="#">zoomIn</a>	9.32
<a href="#">zoomOut</a>	9.33

zoomStop	9.34
----------	------

## 录制

action	10.1
screenshot	10.2
start	10.3
stop	10.4

## 场景

first	11.1
freeze	11.2
freezeToggle	11.3
ftbToggle	11.4
getSceneList	11.5
last	11.6
next	11.7
pauseVideo	11.8
playVideo	11.9
previewToPGM	11.10
previous	11.11
seekVideo	11.12
seekVideoBackward	11.13
seekVideoByProgress	11.14
seekVideoForward	11.15
seekVideoToEnd	11.16
seekVideoToStart	11.17
select3DPattern	11.18
selectDvePattern	11.19
selectStingerFile	11.20
setFTBDuration	11.21
setQuickSwitch	11.22
setTransitionAnimation	11.23
setTransitionDuration	11.24
switchByIndex	11.25
switchByName	11.26
switchByUUID	11.27
toggleFreeze	11.28
toggleFTB	11.29
toggleOffFTB	11.30
toggleOnFTB	11.31
unfreeze	11.32
videoPlayAction	11.33

## 节目

getShowList	12.1
switchById	12.2
switchByIndex	12.3
switchByName	12.4

## 推流

<a href="#">actionById</a>	13.1
<a href="#">actionByIndex</a>	13.2
<a href="#">actionByName</a>	13.3
<a href="#">getStreamServersList</a>	13.4
<a href="#">stopAll</a>	13.5

## 快捷键

<a href="#">action</a>	14.1
<a href="#">getConfig</a>	14.2

## 系统设置

<a href="#">reboot</a>	15.1
<a href="#">screenshot</a>	15.2
<a href="#">setBrightness</a>	15.3
<a href="#">setUSBCMode</a>	15.4
<a href="#">shutdown</a>	15.5

## 音量

<a href="#">getConfig</a>	16.1
<a href="#">getMonitorDevices</a>	16.2
<a href="#">monitorMicInput</a>	16.3
<a href="#">outputMicToUsb</a>	16.4
<a href="#">setMonitorDevice</a>	16.5
<a href="#">setScope</a>	16.6
<a href="#">setState</a>	16.7
<a href="#">setStreamAudioState</a>	16.8
<a href="#">setStreamAudioVolume</a>	16.9
<a href="#">setVolume</a>	16.10
<a href="#">solo</a>	16.11

# 文档简介

针对 Director Mini，我们开放了丰富的 API，方便开发人员与设备交互，如切换画面、叠加 GFX、播放音视频、控制音量及启停直播录制等。这些 API 基于 HTTP 协议，是一种轻量级、无连接状态、无响应数据的接口。通过本文档，您可以更详细地了解每个 API 的功能和请求方式。

本文档中的 API 可适用于以下产品：

- Director Mini（固件版本：3.0）

## HTTP 状态码

Code	说明
200	成功
400	请求包含语法错误, 请求缺少必要参数。
403	服务器已理解请求, 但拒绝执行它。操作的内容不存在或操作行为禁止执行。
404	服务器找不到请求的网页。请求的接口书写错误。

## DEMO: 命令行

在不同操作系统中，可以安装 wget 和 curl 两个工具，安装后可以在命令行中通过 wget 或 curl 命令来调用 Director Mini API。

不同操作系统中，下边示例的 cookie 文件存放位置不同，请根据实际情况修改。

### wget

1.根据场景下标切换场景

```
wget "http://10.10.3.21:8080/V1.0/scene/switchByIndex?index=0" -q -O -
```

2.切换至下一场景

```
wget "http://10.10.3.21:8080/V1.0/scene/next" -q -O -
```

### curl

1.根据场景下标切换场景

```
curl "http://10.10.3.21:8080/V1.0/scene/switchByIndex?index=0"
```

2.切换至下一场景

```
curl "http://10.10.3.21:8080/V1.0/scene/next"
```



# 版本历史

## 版本V3.0

### 背景音乐

#### 新增接口

- [getPlayList](#)
- [playByFile](#)
- [playByIndex](#)
- [seekByProgress](#)

### GFX

#### 新增接口

- [actionByUUID](#)
- [getGFXList](#)

### 记分牌

#### 新增接口

- [resetGameTime](#)
- [setFootballStoppageTime](#)
- [showFootballStagePlayer](#)
- [showFootballStoppageTime](#)

#### 接口更新

- [adjustBallStrike](#)
  - 新增入参: uuid
- [adjustOut](#)
  - 新增入参: uuid
- [clearPitchCount](#)
  - 新增入参: uuid
- [firstPeriod](#)
  - 新增入参: uuid
- [gameTimeAction](#)
  - 新增入参: uuid
- [lastPeriod](#)
  - 新增入参: uuid
- [nextPeriod](#)
  - 新增入参: uuid
- [pauseGameTime](#)
  - 新增入参: uuid
- [playGameTime](#)
  - 新增入参: uuid
- [previousPeriod](#)
  - 新增入参: uuid
- [reset](#)
  - 新增入参: uuid
- [setCountdown](#)
  - 新增入参: uuid
- [setGameTime](#)
  - 新增入参: uuid
- [setOnBaseRunners](#)
  - 新增入参: uuid
- [setPeriod](#)
  - 新增入参: uuid
- [setScoreOfGuest](#)

- 新增入参: uuid
- [setScoreOfHome](#)
  - 新增入参: uuid
- [setTimeFormat](#)
  - 新增入参: uuid
- [showBallStrike](#)
  - 新增入参: uuid
- [showCountdown](#)
  - 新增入参: uuid
- [showGameName](#)
  - 新增入参: uuid
- [showGameTime](#)
  - 新增入参: uuid
- [showInning](#)
  - 新增入参: uuid
- [showOnBaseRunners](#)
  - 新增入参: uuid
- [showOut](#)
  - 新增入参: uuid
- [updateScoreOfGuest](#)
  - 新增入参: uuid
- [updateScoreOfHome](#)
  - 新增入参: uuid

## 计时器

### 接口更新

- [pause](#)
  - 新增入参: uuid
- [play](#)
  - 新增入参: uuid
- [playAction](#)
  - 新增入参: uuid
- [reset](#)
  - 新增入参: uuid

## 秒表

### 接口更新

- [pause](#)
  - 新增入参: uuid
- [play](#)
  - 新增入参: uuid
- [playAction](#)
  - 新增入参: uuid
- [reset](#)
  - 新增入参: uuid

## 网页

### 新增接口

- [downAction](#)
- [endAction](#)
- [goBack](#)
- [goForward](#)
- [homeAction](#)
- [leftAction](#)
- [pageDownAction](#)
- [pageUpAction](#)
- [reload](#)

- [reset](#)
- [rightAction](#)
- [scale](#)
- [upAction](#)

## 回放

### 新增接口

- [nextFrame](#)
- [previousFrame](#)
- [seekByProgress](#)

### 接口更新

- [backward](#)
  - 新增入参: `offset`
- [enterReplayMode](#)
  - 新增入参: `lensIndex`
- [forward](#)
  - 新增入参: `offset`
- [getEventList](#)
  - 新增入参: `number`, `customName`, `clipSuffixName`
- [replayEvent](#)
  - 新增入参: `lensIndex`
- [replayFromSecondsAgo](#)
  - 新增入参: `lensIndex`
- [seek](#)
  - 修改入参: `progress`

## PTZ

### 新增接口

- [focusExact](#)
- [zoomExact](#)

### 接口更新

- [getDeviceList](#)
  - 新增入参: `id`, `layerName`, `controlType`, `isSupportPtz`

### 弃用接口

- [focusFar](#)
- [focusNear](#)
- [focusStop](#)

## 场景

### 新增接口

- [getSceneList](#)
- [previewToPGM](#)
- [seekVideo](#)
- [seekVideoBackward](#)
- [seekVideoByProgress](#)
- [seekVideoForward](#)
- [seekVideoToEnd](#)
- [seekVideoToStart](#)
- [select3DPattern](#)
- [selectDvePattern](#)
- [selectStingerFile](#)
- [switchByUUID](#)

## 接口更新

- [setTransitionAnimation](#)
  - 修改入参: name
- [switchByIndex](#)
  - 修改入参: switchEffect
- [switchByName](#)
  - 修改入参: switchEffect

## 节目

### 新增接口

- [getShowList](#)
- [switchById](#)

## 推流

### 新增接口

- [actionById](#)
- [getStreamServersList](#)

## 快捷键

- [action](#)
- [getConfig](#)

## 系统

### 新增接口

- [screenshot](#)
- [setBrightness](#)

## 音量

### 新增接口

- [getConfig](#)
- [outputMicToUsb](#)
- [setScope](#)
- [solo](#)

### 接口更新

- [monitorMicInput](#)
  - 修改入参: on

## first 接口

跳转至第一首背景音乐。

### 请求方式

```
POST/GET http://ip/V1.0/bgm/first
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

# getPlayList 接口

获取背景音乐播放列表。

## 请求方式

POST/GET http://ip/V1.0/bgm/getPlayList

## 请求数据

无

## 返回数据

```
{
  "bgmList": [
    {
      "singer": "孙燕姿",
      "artworkPath": "/storage/emulated/0/Director/Files/audio/audio_19b3b7793e8b6837fafaef300955ba83.png",
      "duration": 240,
      "fileMD5": "19b3b7793e8b6837fafaef300955ba83",
      "filePath": "/storage/emulated/0/Director/Files/audio/audio_19b3b7793e8b6837fafaef300955ba83.mp3",
      "notExist": false,
      "title": "彩虹金刚"
    }
  ],
  "message": "SUCCESS",
  "status": 0
}
```

属性	说明
status	返回状态。0: 成功
message	业务状态描述
bgmList	背景音乐播放列表

属性	说明
title	标题
filePath	文件地址
fileMD5	文件 MD5 值
duration	持续时长, 毫秒
singer	演唱者, 艺术家
artworkPath	缩图文件地址
notExist	文件是否丢失, true: 已丢失 false: 未丢失

[HTTP Status Code](#)

## last 接口

跳转至最后一首背景音乐。

### 请求方式

```
POST/GET http://ip/V1.0/bgm/last
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## next 接口

跳转至下一首背景音乐。

### 请求方式

```
POST/GET http://ip/V1.0/bgm/next
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)



## pause 接口

暂停背景音乐。

### 请求方式

```
POST/GET http://ip/V1.0/bgm/pause
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## play 接口

播放背景音乐。

### 请求方式

```
POST/GET http://ip/V1.0/bgm/play
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## playAction 接口

暂停/播放背景音乐。

### 请求方式

```
POST/GET http://ip/V1.0/bgm/playAction
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## playByFile 接口

按文件播放背景音乐。

### 请求方式

```
POST/GET http://ip/V1.0/bgm/playByFile?filePath=/storage/emulated/0/Director/Files/audio/audio_19b3b7793e8b6837fafaef300955ba83.mp3
```

### 请求数据

参数	类型	说明
filePath	String	文件路径

### 返回数据

[HTTP Status Code](#)

## playByIndex 接口

按索引播放背景音乐。

### 请求方式

```
POST/GET http://ip/V1.0/bgm/playByIndex?index=1
```

### 请求数据

参数	类型	说明
index	Int	背景音乐索引, 1、2、3...

### 返回数据

[HTTP Status Code](#)

## previous 接口

跳转至上一首背景音乐。

### 请求方式

```
POST/GET http://ip/V1.0/bgm/previous
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## seekByProgress 接口

背景音乐播放进度跳转到指定位置。

### 请求方式

```
POST/GET http://ip/V1.0/bgm/seekByProgress?progress=0.5
```

### 请求数据

参数	类型	说明
progress	Double	背景音乐进度的百分比，取值范围0~1

### 返回数据

[HTTP Status Code](#)

## actionByIndex 接口

根据下标显示或不显示 GFX。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/actionByIndex?index=1&on=true
```

### 请求数据

参数	类型	说明
index	Int	GFX 在列表中的位置, 1、2...
on	Bool	是否应用到合成画面。True: 应用, False: 不应用。 非必要参数, 无此参数时由设备显示或不显示 GFX。

### 返回数据

[HTTP Status Code](#)



## actionByName 接口

根据名称显示或不显示 GFX。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/actionByName?name=clock&on=true
```

### 请求数据

参数	类型	说明
name	String	GFX名称
on	Bool	是否应用到合成画面。True: 应用, False: 不应用。 非必要参数, 无此参数时由设备自动显示或不显示 GFX。

### 返回数据

[HTTP Status Code](#)

## actionByUUID 接口

根据 GFX ID 显示或不显示 GFX。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/actionByUUID?uuid=1719559382373&on=true
```

### 请求数据

参数	类型	说明
uuid	String	GFX 唯一标识
on	Bool	是否应用到合成画面。True：应用，False：不应用。 非必要参数，无此参数时由设备显示或不显示 GFX。

### 返回数据

[HTTP Status Code](#)

## clear 接口

清除所有 GFX。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/clear
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## getGFXList 接口

获取 GFX 列表。

### 请求方式

POST/GET http://ip/V1.0/gfx/getGFXList

### 请求数据

无

### 返回数据

```
{
  "gfxList": [
    {
      "gfxSceneType": 1,
      "name": "Scoreboard",
      "uuid": "71465631-a682-49af-918e-e3713b6cbb18"
    },
    {
      "gfxSceneType": 2,
      "name": "Timer",
      "uuid": "6565d48c-d6c6-4fd7-8345-9cf62487f41d"
    },
    {
      "gfxSceneType": 9,
      "name": "Lower Third",
      "uuid": "e7d00fe7-188f-4fda-aa68-04f82a6fc844"
    }
  ],
  "totalCount": 5,
  "message": "SUCCESS",
  "status": 0
}
```

参数	类型	说明
status	Int	业务状态码 0: 成功
message	String	业务状态描述
totalCount	Int	GFX 总数
gfxList	Array of <a href="#">GFXInfo</a>	GFX 数据列表

#### GFXInfo

参数	类型	说明
uuid	String	GFX 唯一标识
name	String	GFX 名称
gfxSceneType	Int	GFX 类型, 0: 普通类型 1: 记分牌 2: 计时器 3: 秒表 4: 自定义 5: 棒球记分牌 6: 网页 8: 足球记分牌 9: PAG

[HTTP Status Code](#)

## adjustBallStrike 接口

修改棒球投球数。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/adjustBallStrike?ballChange=-3&strikeChange=-3&uuid=1718789860579
```

### 请求数据

参数	类型	说明
ballChange	Int	坏球变化, 正值为加, 负值为减
strikeChange	Int	好球变化, 正值为加, 负值为减
uuid	String	记分牌画面唯一标识, 非必要参数, 无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## adjustOut 接口

修改棒球出局数。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/adjustOut?change=-3&uuid=1718789860579
```

### 请求数据

参数	类型	说明
adjustOut	Int	出局数变化, 正值为加, 负值为减
uuid	String	记分牌画面唯一标识, 非必要参数, 无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## clearPitchCount 接口

清空棒球投球数。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/clearPitchCount?uuid=1718789860579
```

### 请求数据

参数	类型	说明
uuid	String	记分牌画面唯一标识，非必要参数，无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)



## firstPeriod 接口

转到第一场或棒球的第一局上半局。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/firstPeriod?uuid=1718789860579
```

### 请求数据

参数	类型	说明
uuid	String	记分牌画面唯一标识，非必要参数，无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## gameTimeAction 接口

暂停/开始/恢复比赛计时。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/gameTimeAction?uuid=1718789860579
```

### 请求数据

参数	类型	说明
uuid	String	记分牌画面唯一标识，非必要参数，无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## lastPeriod 接口

转到最后一场次（不支持棒球）。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/lastPeriod?uuid=1718789860579
```

### 请求数据

参数	类型	说明
uuid	String	记分牌画面唯一标识，非必要参数，无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## nextPeriod 接口

转到下一场次或棒球的下一个半局。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/nextPeriod?uuid=1718789860579
```

### 请求数据

参数	类型	说明
uuid	String	记分牌画面唯一标识，非必要参数，无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## pauseGameTime 接口

暂停比赛计时。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/pauseGameTime?uuid=1718789860579
```

### 请求数据

参数	类型	说明
uuid	String	记分牌画面唯一标识，非必要参数，无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## playGameTime 接口

启动或继续比赛计时。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/playGameTime?uuid=1718789860579
```

### 请求数据

参数	类型	说明
uuid	String	记分牌画面唯一标识，非必要参数，无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## previousPeriod 接口

转到上一场次或棒球的上一个半局数。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/previousPeriod?uuid=1718789860579
```

### 请求数据

参数	类型	说明
uuid	String	记分牌画面唯一标识，非必要参数，无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## reset 接口

重置记分牌。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/reset?uuid=1718789860579
```

### 请求数据

参数	类型	说明
uuid	String	记分牌画面唯一标识，非必要参数，无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)



## resetGameTime 接口

重置记分牌比赛时间。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/resetGameTime?uuid=1718789860579
```

### 请求数据

参数	类型	说明
uuid	String	记分牌唯一标识, 非必要参数, 无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## setCountdown 接口

设置比赛倒计时时间。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/setCountdown?h=1&m=2&s=3&sss=100&time=10:00&uuid=1718789860579
```

### 请求数据

参数	类型	说明
h	Int	时, 取值范围: 0~∞
m	Int	分, 取值范围: 0~∞
s	Int	秒, 取值范围: 0~∞
sss	Int	毫秒, 取值范围: 0~999
time	String	比赛时间,如:01:05:06、05:06、6:06、9.8、9、0.5。存在此值时将忽略h、m、s及sss参数
uuid	String	记分牌画面唯一标识, 非必要参数, 无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## setFootballStoppageTime 接口

设置足球记分牌补时时时间。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/setFootballStoppageTime?seconds=3&uuid=1718789860579
```

### 请求数据

参数	类型	说明
seconds	Int	补时时间, 单位:秒
uuid	String	记分牌画面唯一标识, 非必要参数, 无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## setGameTime 接口

设置比赛时间。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/setGameTime?h=1&m=2&s=3&sss=100&time=10:00&uuid=1718789860579
```

### 请求数据

参数	类型	说明
h	Int	时, 取值范围: 0~∞
m	Int	分, 取值范围: 0~∞
s	Int	秒, 取值范围: 0~∞
sss	Int	毫秒, 取值范围: 0~999
time	String	比赛时间,如:01:05:06、05:06、6:06、9.8、9、0.5。存在此值时将忽略h、m、s及sss参数
modifyTimeFormat	Boolean	设置时间参数的同时是否修改时间格式, True: 是 False: 否。
uuid	String	记分牌画面唯一标识, 非必要参数, 无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## setOnBaseRunners 接口

跑垒设置。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/setOnBaseRunners?position=-3&touchBase=true&uuid=1718789860579
```

### 请求数据

参数	类型	说明
position	Int	位置, 1:一垒 2:二垒 3:三垒
touchBase	Boolean	是否上垒, true:是 false:否
uuid	String	记分牌画面唯一标识, 非必要参数, 无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## setPeriod 接口

转到指定场次或局数。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/setPeriod?period=1&uuid=1718789860579
```

### 请求数据

参数	类型	说明
period	Int	场次在列表中的位置, 1、2、3... 棒球记分牌局数, 1: 第一局上半场、2: 第一局下半场、3: 第二局上半场 ...
uuid	String	记分牌画面唯一标识, 非必要参数, 无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## setScoreOfGuest 接口

设置客队得分。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/setScoreOfGuest?score=3&uuid=1718789860579
```

### 请求数据

参数	类型	说明
score	Int	客队得分
uuid	String	记分牌画面唯一标识, 非必要参数, 无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## setScoreOfHome 接口

设置主队得分。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/setScoreOfHome?score=3&uuid=1718789860579
```

### 请求数据

参数	类型	说明
score	Int	主队得分
uuid	String	记分牌画面唯一标识, 非必要参数, 无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)



# setTimeFormat

设置比赛时间格式。

## 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/setTimeFormat?timeFormat=1&uuid=1718789860579
```

## 请求数据

参数	类型	说明
timeFormat	Int	时间格式。 0: h:mm:ss (时:分:秒) 1: mm:ss (分:秒) 2: mm:ss.d (分:秒.十分之一秒) 3: mm:ss, ss.d (分:秒, 不足一分钟时显示: 秒.十分之一秒)
uuid	String	记分牌画面唯一标识, 非必要参数, 无此入参时视为控制叠加到直播画面上的记分牌

## 返回数据

[HTTP Status Code](#)

## showBallStrike 接口

显示棒球投球数。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/showBallStrike?show=true&uuid=1718789860579
&uuid=1718789860579
```

### 请求数据

参数	类型	说明
show	Boolean	是否显示棒球投球数, true:显示 false:隐藏
uuid	String	记分牌画面唯一标识, 非必要参数, 无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## showCountdown 接口

显示比赛倒计时。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/showCountdown?show=true&uuid=1718789860579
```

### 请求数据

参数	类型	说明
show	Boolean	是否显示比赛倒计时, true:显示 false:隐藏
uuid	String	记分牌画面唯一标识, 非必要参数, 无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## showFootballStagePlayer 接口

显示足球记分牌换人。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/showFootballStagePlayer?show=true&uuid=1718789860579
```

### 请求数据

参数	类型	说明
show	Boolean	是否显示换人。非必要参数，无此入参时，根据记分牌的当前状态切换为显示或隐藏。 true: 显示; false: 隐藏
uuid	String	记分牌画面唯一标识，非必要参数，无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## showFootballStoppageTime 接口

显示足球记分牌补时时间。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/showFootballStoppageTime?show=true&uuid=1718789860579
```

### 请求数据

参数	类型	说明
show	Boolean	是否显示补时时间。非必要参数，无此入参时，根据记分牌的当前状态切换为显示或隐藏。 true: 显示; false: 隐藏
uuid	String	记分牌画面唯一标识，非必要参数，无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## showGameName 接口

显示比赛名称。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/showGameName?show=true&uuid=1718789860579
```

### 请求数据

参数	类型	说明
show	Boolean	是否显示比赛名称, true:显示 false:隐藏
uuid	String	记分牌画面唯一标识, 非必要参数, 无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## showGameTime 接口

显示比赛计时时间。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/showGameTime?show=true&uuid=1718789860579
```

### 请求数据

参数	类型	说明
show	Boolean	是否显示比赛计时时间, true:显示 false:隐藏
uuid	String	记分牌画面唯一标识, 非必要参数, 无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## showInning 接口

显示棒球局数。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/showInning?show=true&uuid=1718789860579
```

### 请求数据

参数	类型	说明
show	Boolean	是否显示棒球局数, true:显示 false:隐藏
uuid	String	记分牌画面唯一标识, 非必要参数, 无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)



## showOnBaseRunners 接口

显示棒球跑垒情况。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/showOnBaseRunners?show=true&uuid=1718789860579
```

### 请求数据

参数	类型	说明
show	Boolean	是否显示棒球跑垒情况, true:显示 false:隐藏
uuid	String	记分牌画面唯一标识, 非必要参数, 无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## showOut 接口

显示棒球出局数。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/scoreboard/showOut?show=true&uuid=1718789860579
```

### 请求数据

参数	类型	说明
show	Boolean	是否显示出局数, true:显示 false:隐藏
uuid	String	记分牌画面唯一标识, 非必要参数, 无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## updateScoreOfGuest 接口

修改客队比分。

### 请求方式

POST/GET <http://ip/V1.0/gfx/scoreboard/updateScoreOfGuest?changeScore=3&uuid=1718789860579>

### 请求数据

参数	类型	说明
changeScore	Int	比分变化, 正值为加分, 负值为减分
uuid	String	记分牌画面唯一标识, 非必要参数, 无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## updateScoreOfHome 接口

修改主队比分。

### 请求方式

POST/GET <http://ip/V1.0/gfx/scoreboard/updateScoreOfHome?changeScore=3&uuid=1718789860579>

### 请求数据

参数	类型	说明
changeScore	Int	比分变化, 正值为加分, 负值为减分
uuid	String	记分牌画面唯一标识, 非必要参数, 无此入参时视为控制叠加到直播画面上的记分牌

### 返回数据

[HTTP Status Code](#)

## pause 接口

暂停计时器。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/countdown/pause?uuid=1718789860579
```

### 请求数据

参数	类型	说明
uuid	String	计时器画面唯一标识，非必要参数，无此入参时视为控制叠加到直播画面上的计时器

### 返回数据

[HTTP Status Code](#)

## play 接口

启动或恢复计时器。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/countdown/play?uuid=1718789860579
```

### 请求数据

参数	类型	说明
uuid	String	计时器画面唯一标识，非必要参数，无此入参时视为控制叠加到直播画面上的计时器

### 返回数据

[HTTP Status Code](#)

## playAction 接口

启动、恢复或暂停计时器。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/countdown/playAction?uuid=1718789860579
```

### 请求数据

参数	类型	说明
uuid	String	计时器画面唯一标识，非必要参数，无此入参时视为控制叠加到直播画面上的计时器

### 返回数据

[HTTP Status Code](#)

## reset 接口

重置计时器。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/countdown/reset?uuid=1718789860579
```

### 请求数据

参数	类型	说明
uuid	String	计时器画面唯一标识，非必要参数，无此入参时视为控制叠加到直播画面上的计时器

### 返回数据

[HTTP Status Code](#)



## pause 接口

暂停秒表。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/stopwatch/pause?uuid=1718789860579
```

### 请求数据

参数	类型	说明
uuid	String	秒表画面唯一标识，非必要参数，无此入参时视为控制叠加到直播画面上的秒表

### 返回数据

[HTTP Status Code](#)

## play 接口

启动或恢复秒表。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/stopwatch/play?uuid=1718789860579
```

### 请求数据

参数	类型	说明
uuid	String	秒表画面唯一标识，非必要参数，无此入参时视为控制叠加到直播画面上的秒表

### 返回数据

[HTTP Status Code](#)

## playAction 接口

启动/恢复/暂停秒表。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/stopwatch/playAction?uuid=1718789860579
```

### 请求数据

参数	类型	说明
uuid	String	秒表画面唯一标识，非必要参数，无此入参时视为控制叠加到直播画面上的秒表

### 返回数据

[HTTP Status Code](#)

## reset 接口

重置秒表。

### 请求方式

```
POST/GET http://ip/V1.0/gfx/stopwatch/reset?uuid=1718789860579
```

### 请求数据

参数	类型	说明
uuid	String	秒表画面唯一标识，非必要参数，无此入参时视为控制叠加到直播画面上的秒表

### 返回数据

[HTTP Status Code](#)

## downAction 接口

当前直播画面中的网页内容响应下方向键（向下滚动）。

### 请求方式

```
POST/GET http://ip/V1.0/graphics/downAction
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## endAction 接口

当前直播画面中的网页内容响应 End 键（滚到页面底部）。

### 请求方式

```
POST/GET http://ip/V1.0/graphics/endAction
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## goBack 接口

当前直播画面中的网页返回上一页。

### 请求方式

```
POST/GET http://ip/V1.0/graphics/goBack
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## goForward 接口

当前直播画面中的网页前往下一页。

### 请求方式

```
POST/GET http://ip/V1.0/graphics/goForward
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)



## homeAction 接口

当前直播画面中的网页内容响应 Home 键（滚到页面顶部）。

### 请求方式

```
POST/GET http://ip/V1.0/graphics/homeAction
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## leftAction 接口

当前直播画面中的网页内容响应左方向键（向左滚动）。

### 请求方式

```
POST/GET http://ip/V1.0/graphics/leftAction
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## pageDownAction 接口

当前直播画面中的网页内容响应 PgDn 键（向下翻页）。

### 请求方式

```
POST/GET http://ip/V1.0/graphics/pageDownAction
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## pageUpAction 接口

当前直播画面中的网页内容响应 PgUp 键（向上翻页）。

### 请求方式

```
POST/GET http://ip/V1.0/graphics/pageUpAction
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## doReload 接口

当前直播画面中的网页内容重新加载。

### 请求方式

```
POST/GET http://ip/V1.0/graphics/reload
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## doReset 接口

当前直播画面中的网页内容恢复默认大小。

### 请求方式

```
POST/GET http://ip/V1.0/graphics/reset
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## rightAction 接口

当前直播画面中的网页内容响应右方向键（向右滚动）。

### 请求方式

```
POST/GET http://ip/V1.0/graphics/rightAction
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## doScale 接口

控制当前直播画面中的网页内容缩放。

### 请求方式

```
POST/GET http://ip/V1.0/graphics/doScale?scale=2.0
```

### 请求数据

参数	类型	说明
scale	Float	缩放比例, 0.25~5.0

### 返回数据

[HTTP Status Code](#)



## upAction 接口

当前直播画面中的网页内容响应上方向键（向上滚动）。

### 请求方式

```
POST/GET http://ip/V1.0/graphics/upAction
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## addEvent 接口

保存精彩瞬间。

### 请求方式

```
POST/GET http://ip/V1.0/replay/addEvent?duration=5
```

### 请求数据

参数	类型	说明
duration	Int	精彩瞬间持续时长, 秒, 3~60

### 返回数据

[HTTP Status Code](#)

## backward 接口

快退。

### 请求方式

```
POST/GET http://ip/V1.0/replay/backward?offset=5000
```

### 请求数据

参数	类型	说明
offset	Int	快退步长, 单位: 毫秒。非必要参数, 为0或无此参数时, 使用设备预置的步长。

### 返回数据

[HTTP Status Code](#)

## enterReplayMode 接口

进入快速回放模式，并从缓冲的开头开始播放。

### 请求方式

```
POST/GET http://ip/V1.0/replay/enterReplayMode?lensIndex=1
```

### 请求数据

参数	类型	说明
lensIndex	Int	先回放的镜头，非必要参数，仅在双镜头回放时生效，默认值-1 0: 镜头 1; 1: 镜头 2; -1: 按预置规则回放

### 返回数据

[HTTP Status Code](#)

## exitReplayMode 接口

退出回放模式。

### 请求方式

```
POST/GET http://ip/V1.0/replay/exitReplayMode
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## forward 接口

快进。

### 请求方式

```
POST/GET http://ip/V1.0/replay/forward?offset=5000
```

### 请求数据

参数	类型	说明
offset	Int	快进步长，单位：毫秒。非必要参数，为0或无此参数时，使用设备预置的步长。

### 返回数据

[HTTP Status Code](#)

# getEventList 接口

获取精彩事件列表

## 请求方式

POST/GET http://ip/V1.0/replay/getEventList

## 请求数据

无

## 返回数据

```
{
  "status": 0,
  "message": "SUCCESS",
  "eventList": [
    {
      "replayRecordId": "88",
      "totalSize": 370502,
      "duration": 5000,
      "number": 4
    },
    {
      "replayRecordId": "87",
      "totalSize": 2361703,
      "duration": 5000,
      "number": 3,
      "customName": "Event xxxx"
    },
    {
      "replayRecordId": "87_clip",
      "totalSize": 1535456,
      "duration": 5000,
      "number": 3,
      "customName": "Event xxxx",
      "clipSuffixName": "_clip"
    }
  ]
}
```

属性	说明
replayRecordId	精彩事件唯一标识
totalSize	精彩事件文件大小, 字节
duration	时长, 毫秒
number	精彩事件序号
customName	自定义名称
clipSuffixName	剪辑后的后缀名称, 保留字段

[HTTP Status Code](#)

## muteAudio 接口

将回放内容静音。

### 请求方式

```
POST/GET http://ip/V1.0/replay/muteAudio?mute=true
```

### 请求数据

参数	类型	说明
mute	Boolean	是否静音, false:否 true:是。 非必要参数, 无此参数时由设备静音或取消静音。

### 返回数据

[HTTP Status Code](#)



## muteMic 接口

回放时，静音麦克风。

### 请求方式

```
POST/GET http://ip/V1.0/replay/muteMic?mute=true
```

### 请求数据

参数	类型	说明
mute	Boolean	是否静音，false:否 true:是。 非必要参数，无此参数时由设备静音或取消静音。

### 返回数据

[HTTP Status Code](#)

## nextFrame 接口

前进一帧(仅在暂停回放时生效, 2.5.0及其以上版本固件支持)。

### 请求方式

```
POST/GET http://ip/V1.0/replay/nextFrame
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## pause 接口

暂停回放。

### 请求方式

```
POST/GET http://ip/V1.0/replay/pause
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## play 接口

开始或继续回放。

### 请求方式

```
POST/GET http://ip/V1.0/replay/play
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## playAction 接口

开始、继续或暂停回放。

### 请求方式

```
POST/GET http://ip/V1.0/replay/playAction
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## previousFrame 接口

后退一帧(仅在暂停回放时生效, 2.5.0及其以上版本固件支持)。

### 请求方式

```
POST/GET http://ip/V1.0/replay/previousFrame
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## replayEvent 接口

回放精彩事件。

### 请求方式

```
POST/GET http://ip/V1.0/replay/replayEvent?replayRecordId=23&lensIndex=1
```

### 请求数据

参数	类型	说明
replayRecordId	Int	精彩事件唯一标识, 非必要参数, 无此参数时, 视为回放最后一个精彩事件
lensIndex	Int	先回放的镜头, 非必要参数, 仅在双镜头回放时生效, 默认值-1 0: 镜头 1; 1: 镜头 2; -1: 按预置规则回放

### 返回数据

[HTTP Status Code](#)

## replayFromSecondsAgo 接口

自缓冲区结束点前 N 秒进入回放。

### 请求方式

```
POST/GET http://ip/V1.0/replay/replayFromSecondsAgo?seconds=5&lensIndex=1
```

### 请求数据

参数	类型	说明
seconds	Int	秒, 3~60
lensIndex	Int	先回放的镜头, 非必要参数, 仅在双镜头回放时生效, 默认值-1 0: 镜头 1; 1: 镜头 2; -1: 按预置规则回放

### 返回数据

[HTTP Status Code](#)



## rewindToStart 接口

回退到起始位置。

### 请求方式

```
POST/GET http://ip/V1.0/replay/rewindToStart
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## seek 接口

设置回放进度。

### 请求方式

```
POST/GET http://ip/V1.0/replay/seek?progress=5000
```

### 请求数据

参数	类型	说明
progress	Int	回放进度, 单位: 毫秒

### 返回数据

[HTTP Status Code](#)

## seekByProgress 接口

设置回放进度。

### 请求方式

```
POST/GET http://ip/V1.0/replay/seekByProgress?progress=0.5
```

### 请求数据

参数	类型	说明
progress	Double	回放进度的百分比，取值范围0~1

### 返回数据

[HTTP Status Code](#)

## selectLens 接口

选择回放画面。

### 请求方式

POST/GET http://ip/V1.0/replay/selectLens?lensIndex=0

### 请求数据

参数	类型	说明
lensIndex	Int	回放画面的组合模式 0: 镜头 1 1: 镜头 2 2: 组合画面

### 返回数据

[HTTP Status Code](#)

## setSpeed 接口

设置播放速度。

### 请求方式

```
POST/GET http://ip/V1.0/replay/setSpeed?speed=0.5
```

### 请求数据

参数	类型	说明
speed	double	播放速度, (0~1)

### 返回数据

[HTTP Status Code](#)

# AIHumanTrackingAction 接口

启动/停止人工智能跟踪 (OBSBOT 相机)。

## 请求方式

POST/GET http://ip/V1.0/ptz/AIHumanTrackingAction

## 请求数据

无

## 返回数据

[HTTP Status Code](#)

## autoFocus 接口

自动聚焦。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/autoFocus
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## focusExact 接口

设置聚焦位置。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/focusExact?focusValue=0.5
```

### 请求数据

参数	类型	说明
focusValue	double	聚焦位置, 取值范围 0 ~ 1 (数值越大距离越远)

### 返回数据

[HTTP Status Code](#)



## focusFar 接口

V3.0及以上版本弃用，请使用[focusExact](#)。

拉远焦距。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/focusFar?speed=0.5
```

### 请求数据

参数	类型	说明
speed	double	拉远速度, (0~1)

### 返回数据

[HTTP Status Code](#)

## focusNear 接口

V3.0及以上版本弃用，请使用[focusExact](#)。

推进焦距。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/focusNear?speed=0.5
```

### 请求数据

参数	类型	说明
speed	double	推进速度, (0~1)

### 返回数据

[HTTP Status Code](#)

## focusStop 接口

V3.0及以上版本弃用，请使用[focusExact](#)。

停止推进/拉远焦距。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/focusStop
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

# getDeviceList 接口

获取当前节目中所有可控制的 PTZ 设备。

## 请求方式

POST/GET http://ip/V1.0/ptz/getDeviceList

## 请求数据

无

## 返回数据

```
{
  "status": 0,
  "message": "SUCCESS",
  "current": "",
  "deviceList": [
    {
      "id": "fcd51e30551e4795879c8bdc8f5397cd",
      "isSupportPtz": true,
      "controlType": 102,
      "layerName": "PHONE CAMERA - Camera #1"
    },
    {
      "id": "2ccca046b59543bd931ad343f07445c5",
      "isSupportPtz": true,
      "controlType": 102,
      "layerName": "PHONE CAMERA - Camera #2"
    }
  ]
}
```

属性	说明
status	返回状态。0: 成功
current	当前控制的 PTZ 设备 ID
deviceList	可用的 PTZ 设备列表

属性	说明
id	PTZ 设备 ID
layerName	PTZ 设备显示名称
controlType	PTZ 设备类型 100: Visca UDP 102: 手机摄像头 103: UVC
isSupportPtz	是否支持 PTZ 控制 true: 支持; false: 不支持

[HTTP Status Code](#)

## goToPreset 接口

转到预设机位。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/goToPreset?position=1
```

### 请求数据

参数	类型	说明
position	Int	预设机位位置, 1~9

### 返回数据

[HTTP Status Code](#)

## moveDown 接口

云台移动, 方向:下。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/moveDown?speed=0.5
```

### 请求数据

参数	类型	说明
speed	double	移动速度, (0~1)

### 返回数据

[HTTP Status Code](#)

## moveDownLeft 接口

云台移动, 方向:左下。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/moveDownLeft?speed=0.5
```

### 请求数据

参数	类型	说明
speed	double	移动速度, (0~1)

### 返回数据

[HTTP Status Code](#)

## moveDownRight 接口

云台移动, 方向:右下。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/moveDownRight?speed=0.5
```

### 请求数据

参数	类型	说明
speed	double	移动速度, (0~1)

### 返回数据

[HTTP Status Code](#)



## moveHome 接口

云台回到中心位置。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/moveHome
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## moveLeft 接口

云台移动, 方向:左。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/moveLeft?speed=0.5
```

### 请求数据

参数	类型	说明
speed	double	移动速度, (0~1)

### 返回数据

[HTTP Status Code](#)

## moveRight 接口

云台移动, 方向:右。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/moveRight?speed=0.5
```

### 请求数据

参数	类型	说明
speed	double	移动速度, (0~1)

### 返回数据

[HTTP Status Code](#)

## moveStop 接口

云台停止移动。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/moveStop
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## moveUp 接口

云台移动, 方向:上。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/moveUp?speed=0.5
```

### 请求数据

参数	类型	说明
speed	double	移动速度, (0~1)

### 返回数据

[HTTP Status Code](#)

## moveUpLeft 接口

云台移动, 方向:左上。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/moveUpLeft?speed=0.5
```

### 请求数据

参数	类型	说明
speed	double	移动速度, (0~1)

### 返回数据

[HTTP Status Code](#)

## moveUpRight 接口

云台移动, 方向:右上。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/moveUpRight?speed=0.5
```

### 请求数据

参数	类型	说明
speed	double	移动速度, (0~1)

### 返回数据

[HTTP Status Code](#)

## recordAction 接口

开始/停止录制 (OBSBOT Tail Air) 。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/recordAction
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)



## reset 接口

重置 (OBSBOT 相机) 。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/reset
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## selectedControl 接口

选择控制的 PTZ 设备。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/selectedControl?id=35486f14-4799-4444-903b-22db131f0abb
```

### 请求数据

参数	类型	说明
id	String	PTZ 设备的唯一标识

### 返回数据

[HTTP Status Code](#)

## setHumanTrackingMode 接口

设置跟踪方式 (OBSBOT 相机)。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/setHumanTrackingMode?mode=normalTrack
```

### 请求数据

参数	类型	说明
mode	String	normalTrack: 常规跟踪 upperBody: 上半身跟踪 closeUp: 特写跟踪

### 返回数据

[HTTP Status Code](#)

## setHumanTrackingSpeed 接口

设置跟踪速度 (OBSBOT Tail Air) 。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/setHumanTrackingSpeed?speed=slow
```

### 请求数据

参数	类型	说明
speed	String	slow: 慢, fast: 快, standard: 标准

### 返回数据

[HTTP Status Code](#)

## setHumanTrackingType 接口

设置跟踪模式 (OBSBOT Tiny 2) 。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/setHumanTrackingType?type=1
```

### 请求数据

参数	类型	说明
type	int	0: 标准, 1: 运动

### 返回数据

[HTTP Status Code](#)

## startAIHumanTracking 接口

启动人工智能跟踪（OBSBOT 相机）。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/startAIHumanTracking
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## startRecording 接口

开始录制 (OBSBOT Tail Air) 。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/startRecording
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## stopAIHumanTracking 接口

关闭人工智能跟踪（OBSBOT 相机）。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/stopAIHumanTracking
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)



## stopRecording 接口

停止录制 (OBSBot 相机 Tail Air)。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/stopRecording
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## storePreset 接口

存储当前状态为预置点。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/storePreset?position=1
```

### 请求数据

参数	类型	说明
position	Int	预置点序号, 1~9

### 返回数据

[HTTP Status Code](#)

## wakeUp 接口

唤醒设备（OBSBOT Tiny 2）。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/wakeUp
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## zoomExact 接口

推进/拉远镜头。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/zoomExact?zoomValue=0.5
```

### 请求数据

参数	类型	说明
zoomValue	double	缩放范围, 取值范围 0 ~ 1 (数值越大镜头越近)

### 返回数据

[HTTP Status Code](#)

## zoomIn 接口

镜头推进，放大画面。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/zoomIn?speed=0.5
```

### 请求数据

参数	类型	说明
speed	double	推进速度, (0~1)

### 返回数据

[HTTP Status Code](#)

## zoomOut 接口

拉远推进, 缩小画面。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/zoomOut?speed=0.5
```

### 请求数据

参数	类型	说明
speed	double	拉远速度, (0~1)

### 返回数据

[HTTP Status Code](#)

## zoomStop 接口

停止推进/拉远镜头。

### 请求方式

```
POST/GET http://ip/V1.0/ptz/zoomStop
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## action 接口

开始/结束录制。

### 请求方式

```
POST/GET http://ip/V1.0/record/action
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)



## screenshot 接口

对正在直播的内容进行截屏。

### 请求方式

```
POST/GET http://ip/V1.0/record/screenshot
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## start 接口

开始录制。

### 请求方式

```
POST/GET http://ip/V1.0/record/start
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## stop 接口

停止录制。

### 请求方式

```
POST/GET http://ip/V1.0/record/stop
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## first 接口

切换至第一个场景。

### 请求方式

```
POST/GET http://ip/V1.0/scene/first
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## freeze 接口

冻结当前场景。

### 请求方式

```
POST/GET http://ip/V1.0/scene/freeze
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## freezeToggle 接口

切换当前场景画面冻结状态。

### 请求方式

```
POST/GET http://ip/V1.0/scene/freezeToggle
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## ftbToggle 接口

切换 FTB 启用状态。

### 请求方式

```
POST/GET http://ip/V1.0/scene/ftbToggle
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

# getSceneList 接口

获取场景列表。

## 请求方式

POST/GET http://ip/V1.0/scene/getSceneList

## 请求数据

NULL

## 返回数据

```
{
  "sceneList": [
    {
      "name": "PICTURE",
      "uuid": "17170511950740"
    },
    {
      "name": "HDMI 1",
      "uuid": "17170511950741"
    },
    {
      "name": "HDMI 2",
      "uuid": "17170511950742"
    },
    {
      "name": "Camera #1",
      "uuid": "17170511950754"
    },
    {
      "name": "HDMI 1",
      "uuid": "1717059172941"
    },
    {
      "name": "Camera #1",
      "uuid": "1717051945689"
    },
    {
      "name": "HDMI 1",
      "uuid": "1717059182814"
    },
    {
      "name": "Hamedal C10",
      "uuid": "1718156142532"
    }
  ],
  "totalCount": 8,
  "message": "SUCCESS",
  "status": 0
}
```

[HTTP Status Code](#)



参数	类型	说明
status	Int	业务状态码 0: 成功
message	String	业务状态描述
totalCount	Int	场景总数
sceneList	Array of <a href="#">SceneInfo</a>	场景数据列表

#### SceneInfo

参数	类型	说明
uuid	String	场景唯一标识
name	String	场景名称

## last 接口

切换至最后一个场景。

### 请求方式

```
POST/GET http://ip/V1.0/scene/last
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## next 接口

切换至下一个场景。

### 请求方式

```
POST/GET http://ip/V1.0/scene/next
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## pauseVideo 接口

暂停当前场景中的视频。

### 请求方式

```
POST/GET http://ip/V1.0/scene/pauseVideo
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## playVideo 接口

播放当前场景中的视频。

### 请求方式

```
POST/GET http://ip/V1.0/scene/playVideo
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## previewToPGM 接口

将当前预览画面应用为直播画面。

### 请求方式

```
POST/GET http://ip/V1.0/scene/previewToPGM?switchEffect=1
```

### 请求数据

参数	类型	说明
switchEffect	Int	转场方式, 0:硬切 1:渐变 非必要参数, 无此参数是将使用切换设置中预置的转场方式。

### 返回数据

[HTTP Status Code](#)

## previous 接口

切换至上一个视频。

### 请求方式

```
POST/GET http://ip/V1.0/scene/previous
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## seekVideo 接口

当前场景视频跳转到指定位置。

### 请求方式

```
POST/GET http://ip/V1.0/scene/seekVideo?progress=10000
```

### 请求数据

参数	类型	说明
progress	Int	视频进度目标位置, 单位: 毫秒

### 返回数据

[HTTP Status Code](#)



## seekVideoBackward 接口

当前场景视频快退。

### 请求方式

```
POST/GET http://ip/V1.0/scene/seekVideoBackward?offset=5000
```

### 请求数据

参数	类型	说明
offset	Int	视频快退时间, 单位: 毫秒

### 返回数据

[HTTP Status Code](#)

## seekVideoByProgress 接口

当前场景视频跳转到指定位置。

### 请求方式

```
POST/GET http://ip/V1.0/scene/seekVideoByProgress?progress=0.5
```

### 请求数据

参数	类型	说明
progress	Double	视频进度的百分比，取值范围0~1

### 返回数据

[HTTP Status Code](#)

## seekVideoForward 接口

当前场景视频快进。

### 请求方式

```
POST/GET http://ip/V1.0/scene/seekVideoForward?offset=5000
```

### 请求数据

参数	类型	说明
offset	Int	视频快进时间, 单位: 毫秒

### 返回数据

[HTTP Status Code](#)

## seekVideoToEnd 接口

当前场景中视频快进到结束位置。

### 请求方式

```
POST/GET http://ip/V1.0/scene/seekVideoToEnd
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## seekVideoToStart 接口

当前场景中视频回到起始位置。

### 请求方式

```
POST/GET http://ip/V1.0/scene/seekVideoToStart
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## select3DPattern 接口

选择 3D 样式。

### 请求方式

POST/GET http://ip/V1.0/scene/select3DPattern?pattern=1

### 请求数据

参数	类型	说明
pattern	Int	3D 样式 0: 翻转 1: 立方体 2: 框 3: 百叶窗 4: 开门 5: 剥离 6: 交换 7: 马赛克

### 返回数据

[HTTP Status Code](#)

## selectDvePattern 接口

选择 DVE 样式。

### 请求方式

POST/GET http://ip/V1.0/scene/selectDvePattern?pattern=1

### 请求数据

参数	类型	说明
pattern	Int	DVE 样式 0: 向上推 1: 向下推 2: 向左推 3: 向右推 4: 向左上推 5: 向右上推 6: 向左下推 7: 向右下推 8: 向上挤压 9: 向下挤压 10: 向左挤压 11: 向右挤压 12: 向左上挤压 13: 向右上挤压 14: 向左下挤压 15: 向右下挤压 16: 水平挤压 17: 垂直挤压 18: 中心挤压

### 返回数据

[HTTP Status Code](#)

## selectStingerFile 接口

选择特效视频 (Stinger) 使用的媒体文件。

### 请求方式

```
POST/GET http://ip/V1.0/scene/selectStingerFile?index=1
```

### 请求数据

参数	类型	说明
index	Int	Stinger 文件的索引, 1、2、3或4

### 返回数据

[HTTP Status Code](#)



## setFTBDuration 接口

设置 FTB 转场特效持续时间。

### 请求方式

```
POST/GET http://ip/V1.0/scene/setFTBDuration?duration=1000
```

### 请求数据

参数	类型	说明
duration	Int	持续时间, ms

### 返回数据

[HTTP Status Code](#)

## setQuickSwitch 接口

打开/关闭快速切换场景开关。

### 请求方式

```
POST/GET http://ip/V1.0/scene/setQuickSwitch?state=off
```

### 请求数据

参数	类型	说明
state	String	on:打开 off:关闭。 非必要参数，无此参数时由设备自动打开或关闭快速切换。

### 返回数据

[HTTP Status Code](#)

## setTransitionAnimation 接口

按名称设置场景切换时转场方式。

### 请求方式

POST/GET <http://ip/V1.0/scene/setTransitionAnimation?name=cut>

### 请求数据

参数	类型	说明
name	String	转场方式 cut: 硬切 fade: 渐变 dip: DIP wipe: 划像 dve: DVE stinger: 特效视频 3d: 3D

### 返回数据

[HTTP Status Code](#)

## setTransitionDuration 接口

设置场景转场特效持续时间。

### 请求方式

```
POST/GET http://ip/V1.0/scene/setTransitionDuration?duration=1000
```

### 请求数据

参数	类型	说明
duration	Int	持续时间, ms

### 返回数据

[HTTP Status Code](#)

## switchByIndex 接口

根据索引切换至指定场景。

### 请求方式

```
POST/GET http://ip/V1.0/scene/switchByIndex?index=1&switchEffect=1
```

### 请求数据

参数	类型	说明
index	Int	场景在列表中的索引, 1、2...
switchEffect	Int	转场方式, 0:硬切 1:渐变 非必要参数, 无此参数时将使用切换设置中预置的转场方式。

### 返回数据

[HTTP Status Code](#)

## switchByName 接口

根据名称切换至指定场景。

### 请求方式

```
POST/GET http://ip/V1.0/scene/switchByName?name=scene1&switchEffect=1
```

### 请求数据

参数	类型	说明
name	String	场景名称
switchEffect	Int	转场方式, 0:硬切 1:渐变 非必要参数, 无此参数时将使用切换设置中预置的转场方式。

### 返回数据

[HTTP Status Code](#)

## switchByUUID 接口

根据场景 ID 切换至指定场景。

### 请求方式

```
POST/GET http://ip/V1.0/scene/switchByUUID?uuid=17170511950740&switchEffect=1
```

### 请求数据

参数	类型	说明
uuid	String	场景唯一标识
switchEffect	Int	转场方式, 0:硬切 1:渐变 非必要参数, 无此参数时将使用切换设置中预置的转场方式。

### 返回数据

[HTTP Status Code](#)

## toggleFreeze 接口

冻结或取消冻结当前场景。

### 请求方式

```
POST/GET http://ip/V1.0/scene/toggleFreeze
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)



## toggleFTB 接口

启用/取消 FTB。

### 请求方式

```
POST/GET http://ip/V1.0/scene/toggleFTB
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## toggleOffFTB 接口

关闭 FTB。

### 请求方式

```
POST/GET http://ip/V1.0/scene/toggleOffFTB
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## toggleOnFTB 接口

启用 FTB。

### 请求方式

```
POST/GET http://ip/V1.0/scene/toggleOnFTB
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## unfreeze 接口

取消冻结当前场景。

### 请求方式

```
POST/GET http://ip/V1.0/scene/unfreeze
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## videoPlayAction 接口

播放/暂停当前场景中的视频。

### 请求方式

```
POST/GET http://ip/V1.0/scene/videoPlayAction
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## getShowList 接口

获取节目列表。

### 请求方式

POST/GET http://ip/V1.0/show/getShowList

### 请求数据

NULL

### 返回数据

```
{
  "showList": [
    {
      "frameHeight": 1080,
      "frameRate": 60000,
      "frameWidth": 1920,
      "inCurrent": true,
      "name": "Default Show12",
      "orientation": 0,
      "showId": 1717051195073
    },
    {
      "frameHeight": 1920,
      "frameRate": 60000,
      "frameWidth": 1080,
      "inCurrent": false,
      "name": "Default Show (9:16)",
      "orientation": 1,
      "showId": 1717051195074
    },
    {
      "frameHeight": 1080,
      "frameRate": 60000,
      "frameWidth": 1920,
      "inCurrent": false,
      "name": "1002",
      "orientation": 0,
      "showId": 1717739749370
    },
    {
      "frameHeight": 1080,
      "frameRate": 60000,
      "frameWidth": 1920,
      "inCurrent": false,
      "name": "Default Show",
      "orientation": 0,
      "showId": 1718270747433
    }
  ],
  "totalCount": 4,
  "message": "SUCCESS",
  "status": 0
}
```

[HTTP Status Code](#)

参数	类型	说明
status	Int	业务状态码 0: 成功
message	String	业务状态描述
totalCount	Int	节目总数
showList	Array	节目数据列表

StreamConfigInfo |参数|类型|说明| |----|:----|:----| |showId |Int |节目唯一标识 | |name |String |节目名称 | |orientation |Int |节目朝向  
0: 横屏; 1: 竖屏 | |inCurrent |Boolean |是否为当前演示节目  
false: 否; true: 是 | |frameRate |Int |画面帧率\*1000 | |frameWidth |Int |画面分辨率宽度 | |frameHeight |Int |画面分辨率宽度高度 |

## switchById 接口

根据节目 ID 切换到指定节目。

### 请求方式

```
POST/GET http://ip/V1.0/show/switchById?id=1717051195074
```

### 请求数据

参数	类型	说明
id	Int	节目唯一标识

### 返回数据

[HTTP Status Code](#)



## switchByIndex 接口

根据索引切换到指定节目。

### 请求方式

```
POST/GET http://ip/V1.0/show/switchByIndex?index=0
```

### 请求数据

参数	类型	说明
index	Int	节目在列表中的索引, 0、1、2...

### 返回数据

[HTTP Status Code](#)

## switchByName 接口

根据名称切换到指定节目。

### 请求方式

```
POST/GET http://ip/V1.0/show/switchByName?name=Default 1080p60 fps
```

### 请求数据

参数	类型	说明
name	String	节目名称

### 返回数据

[HTTP Status Code](#)

## actionById 接口

根据推流服务器唯一标识开始/停止推流。

### 请求方式

```
POST/GET http://ip/V1.0/stream/actionById?id=1717051195500&start=true
```

### 请求数据

参数	类型	说明
id	Int	推流服务器唯一标识
start	Bool	直播推流操作, true:开始 false:停止。 非必要参数, 无此参数时由设备自动开始或停止推流。

### 返回数据

[HTTP Status Code](#)

## actionByIndex 接口

根据索引开始/停止推流。

### 请求方式

```
POST/GET http://ip/V1.0/stream/actionByIndex?index=1&start=true
```

### 请求数据

参数	类型	说明
index	Int	推流服务器在列表中的索引, 1、2...
start	Bool	直播推流操作, true:开始 false:停止。 非必要参数, 无此参数时由设备自动开始或停止推流。

### 返回数据

[HTTP Status Code](#)

## actionByName 接口

根据名称开始/停止推流。

### 请求方式

```
POST/GET http://ip/V1.0/stream/actionByName?name=clock&start=true
```

### 请求数据

参数	类型	说明
name	String	GFX名称
start	Bool	直播推流操作, true:启动 false:停止。 非必要参数, 无此参数时由设备自动自动开始或停止推流。

### 返回数据

[HTTP Status Code](#)

# getStreamServersList 接口

获取推流服务器列表。

## 请求方式

POST/GET http://ip/V1.0/stream/getStreamServersList

## 请求数据

NULL

## 返回数据

```
{
  "serverList": [
    {
      "id": 1728357437257,
      "isConfigured": false,
      "name": "抖音",
      "type": 7
    },
    {
      "id": 1728357437258,
      "isConfigured": false,
      "name": "RTMP Server",
      "type": 4
    },
    {
      "id": 1728357437259,
      "isConfigured": false,
      "name": "SRT Caller",
      "type": 5
    }
  ],
  "totalCount": 3,
  "message": "SUCCESS",
  "status": 0
}
```

### HTTP Status Code

参数	类型	说明
status	Int	业务状态码 0: 成功
message	String	业务状态描述
totalCount	Int	推流服务器总数
serverList	Array of <a href="#">StreamConfigInfo</a>	推流服务器配置数据列表

### StreamConfigInfo

参数	类型	说明
id	Int	推流服务器唯一标识
name	String	推流服务器名称
type	Int	推流服务器类型 4: RTMP; 5: SRT; 7: 抖音
isConfigured	Boolean	是否为完整有效的推流服务器配置 false: 否; true: 是

## stopAll 接口

停止所有推流。

### 请求方式

```
POST/GET http://ip/V1.0/stream/stopAll
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)



## action 接口

触发快捷键。

### 请求方式

```
POST/GET http://ip/V1.0/shortcuts/action?id=1725525445613
```

### 请求数据

参数	类型	说明
id	Int	快捷键唯一标识

### 返回数据

[HTTP Status Code](#)

## getConfig 接口

获取快捷键配置列表。

### 请求方式

```
POST/GET http://ip/V1.0/shortcuts/getConfig
```

### 请求数据

NULL

## 返回数据

```
{
  "list": [
    {
      "functions": [
        {
          "id": 311,
          "name": "PTZ move left"
        },
        {
          "id": 29,
          "name": "Wait for 500 ms"
        },
        {
          "id": 317,
          "name": "PTZ stop moving"
        }
      ],
      "id": 1725522037805,
      "name": "Move Left"
    },
    {
      "functions": [
        {
          "id": 315,
          "name": "PTZ move right"
        },
        {
          "id": 29,
          "name": "Wait for 500 ms"
        },
        {
          "id": 317,
          "name": "PTZ stop moving"
        }
      ],
      "id": 1725524226309,
      "name": "Move Right"
    },
    {
      "functions": [
        {
          "id": 12,
          "name": "Switch to the next scene"
        }
      ],
      "id": 1725525445613,
      "name": "Next Scene"
    }
  ],
  "totalCount": 3,
  "message": "SUCCESS",
  "status": 0
}
```

[HTTP Status Code](#)

参数	类型	说明
status	Int	业务状态码 0: 成功
message	String	业务状态描述
totalCount	Int	快捷键总数
list	Array of <a href="#">ShortcutsInfo</a>	快捷键列表

#### ShortcutsInfo

参数	类型	说明
id	Int	快捷键唯一标识
name	String	快捷键名称
functions	Array of <a href="#">FunctionInfo</a>	快捷键功能列表

#### FunctionInfo

参数	类型	说明
id	Int	功能唯一标识
name	String	功能名称

## reboot 接口

重启设备。

### 请求方式

```
POST/GET http://ip/V1.0/system/reboot
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## screenshot 接口

系统截屏，可截取屏幕上的所有元素，等同于使用设备电源按钮进行截屏。

### 请求方式

```
POST/GET http://ip/V1.0/system/screenshot
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## setBrightness 接口

设置屏幕亮度。

### 请求方式

```
POST/GET http://ip/V1.0/system/setBrightness?brightness=1
```

### 请求数据

参数	类型	说明
brightness	Float	屏幕亮度, 取值范围 0 或 0.1~1.0

### 返回数据

[HTTP Status Code](#)

## setUSBCMode 接口

设置外接屏幕的显示内容。

### 请求方式

```
POST/GET http://ip/V1.0/system/setUSBCMode?mode=1
```

### 请求数据

参数	类型	说明
mode	Int	外接屏幕显示内容。 0: 复制主屏幕 1: 直播画面 2: 环出 HDMI 1 3: 环出 HDMI 2 4: 多视图 5: 预览画面

### 返回数据

[HTTP Status Code](#)



## shutdown 接口

关闭设备。

### 请求方式

```
POST/GET http://ip/V1.0/system/shutdown
```

### 请求数据

无

### 返回数据

[HTTP Status Code](#)

## getConfig 接口

获取音频配置。

### 请求方式

```
POST/GET http://ip/V1.0/volume/getConfig
```

### 请求数据

无

## 返回数据

```
{
  "audioList": [
    {
      "audioSourceId": 3,
      "meter": [
        -100,
        -100
      ],
      "audioInfo": {
        "beInMultiScenes": 0,
        "boost": false,
        "direction": 0,
        "isGFX": false,
        "label": "",
        "muted": 0,
        "name": "PROGRAM",
        "preview": false,
        "scope": 0,
        "solo": 0,
        "sourceType": 0,
        "supportAFV": false,
        "type": 3,
        "volume": 0,
        "volumeRecovery": 0
      }
    },
    {
      "audioSourceId": 1,
      "meter": [
        -100,
        -100
      ],
      "audioInfo": {
        "beInMultiScenes": 0,
        "boost": false,
        "direction": 0,
        "isGFX": false,
        "label": "",
        "muted": 0,
        "name": "MONITOR",
        "preview": false,
        "scope": 0,
        "solo": 0,
        "sourceType": 0,
        "supportAFV": false,
        "type": 1,
        "volume": -15,
        "volumeRecovery": 0
      }
    },
    {
      "audioSourceId": 513,
      "meter": [
        -100,
        -100
      ],
      "audioInfo": {
        "beInMultiScenes": 1,
        "boost": false,
```

```

        "direction": 0,
        "isGFX": false,
        "label": "",
        "muted": 0,
        "name": "MIC",
        "preview": false,
        "scope": 1,
        "solo": 0,
        "sourceType": 0,
        "supportAFV": false,
        "type": 512,
        "uuid": "mic",
        "volume": 0,
        "volumeRecovery": 0
    }
},
{
    "audioSourceId": 1792,
    "meter": [
        -100,
        -100
    ],
    "audioInfo": {
        "beInMultiScenes": 0,
        "boost": false,
        "direction": 0,
        "isGFX": false,
        "label": "",
        "muted": 1,
        "name": "HDMI 1",
        "preview": false,
        "scope": 0,
        "solo": 0,
        "sourceType": 6,
        "supportAFV": true,
        "type": 1792,
        "uuid": "eco0",
        "volume": 0,
        "volumeRecovery": 0
    }
},
{
    "audioSourceId": 2048,
    "meter": [
        -100,
        -100
    ],
    "audioInfo": {
        "beInMultiScenes": 0,
        "boost": false,
        "direction": 0,
        "isGFX": false,
        "label": "",
        "muted": 2,
        "name": "HDMI 2",
        "preview": false,
        "scope": 0,
        "solo": 0,
        "sourceType": 7,
        "supportAFV": true,
        "type": 2048,
        "uuid": "eco1",

```

```

        "volume": -13,
        "volumeRecovery": 0
    }
},
{
    "audioSourceId": 1539,
    "meter": [
        -100,
        -100
    ],
    "audioInfo": {
        "beInMultiScenes": 0,
        "boost": false,
        "direction": 0,
        "isGFX": false,
        "label": "BGM",
        "muted": 0,
        "name": "BGM",
        "preview": false,
        "scope": 0,
        "solo": 0,
        "sourceType": 0,
        "supportAFV": false,
        "type": 1539,
        "uuid": "global_bgm",
        "volume": 0,
        "volumeRecovery": 0
    }
},
{
    "audioSourceId": 266,
    "meter": [
        -100,
        -100
    ],
    "audioInfo": {
        "beInMultiScenes": 0,
        "boost": false,
        "direction": 0,
        "isGFX": false,
        "label": "Camera #1",
        "muted": 0,
        "name": "Camera #1",
        "preview": false,
        "scope": 0,
        "solo": 0,
        "sourceType": 30,
        "supportAFV": true,
        "type": 266,
        "uuid": "fcd51e30551e4795879c8bdc8f5397cd",
        "volume": 10,
        "volumeRecovery": 0
    }
},
{
    "audioSourceId": 268,
    "meter": [
        -100,
        -100
    ],
    "audioInfo": {
        "beInMultiScenes": 0,

```

```

        "boost": false,
        "direction": 0,
        "isGFX": false,
        "label": "机位 #1",
        "muted": 0,
        "name": "机位 #1",
        "preview": false,
        "scope": 0,
        "solo": 0,
        "sourceType": 32,
        "supportAFV": true,
        "type": 268,
        "uuid": "0fb2b8a73754415cbea02a08c9446f96",
        "volume": 10,
        "volumeRecovery": 0
    }
},
{
    "audioSourceId": 256,
    "meter": [
        -100,
        -100
    ],
    "audioInfo": {
        "beInMultiScenes": 0,
        "boost": false,
        "direction": 0,
        "isGFX": false,
        "label": "22",
        "muted": 2,
        "name": "22",
        "preview": false,
        "scope": 0,
        "solo": 0,
        "sourceType": 10,
        "supportAFV": true,
        "type": 256,
        "uuid": "ebdaeb581a9b41b89d26a67c7c018cae",
        "volume": 0,
        "volumeRecovery": 0
    }
},
{
    "audioSourceId": 276,
    "meter": [
        -100,
        -100
    ],
    "audioInfo": {
        "beInMultiScenes": 0,
        "boost": false,
        "direction": 0,
        "isGFX": false,
        "label": "Magewell",
        "muted": 2,
        "name": "Magewell",
        "preview": false,
        "scope": 0,
        "solo": 0,
        "sourceType": 54,
        "supportAFV": true,
        "type": 276,

```

```

        "uuid": "1721371716617",
        "volume": 0,
        "volumeRecovery": 0
    }
}
],
"micAudioDevice": {
    "audioSourceId": 513,
    "cardNo": 0,
    "devNo": 0,
    "deviceType": 0,
    "direction": 1,
    "id": 513,
    "isMic": true,
    "sourceType": 0
},
"monitorAudioDevice": {
    "audioSourceId": 0,
    "cardNo": 0,
    "devNo": 0,
    "deviceType": 0,
    "direction": 0,
    "id": 0,
    "isMic": true,
    "sourceType": 0
},
"monitorMicSound": true,
"message": "SUCCESS",
"status": 0
}

```

### 1. 返回状态

```
"status": 0
```

属性	说明
status	返回状态。0: 数据获取成功
message	返回状态描述
monitorMicSound	是否监听麦克风音频, true: 监听 false: 不监听
monitorAudioDevice	监听设备
micAudioDevice	麦克风设备
audioList	音频配置列表

### 2. 音频设备 AudioDevice

属性	说明
audioSourceId	音频资源ID
deviceType	设备类型, 0: 内置音频设备; 1: 外接音频设备
id	设备ID
devNo	设备序号
cardNo	声卡序号

### 3.音频配置 AudioInfo

属性	说明
type	音频类型
belnMultiScenes	音频是否存在于多个画面。为 1 时，支持 <a href="#">setScope</a> 1: 是; 0: 否
scope	音频是否设置为全局，即是否与其他场景共享配置。 0: 共享; 1: 不共享
supportAFV	是否支持设置为AFV（音频跟随视频） true: 支持; false: 不支持
solo	是否仅监听此路音频 1: 是; 0: 否
muted	音频状态 0: 一直打开; 1: 一直关闭; 2: AFV（音频跟随视频）
volume	音量, dB
label	标签
name	名称

其他未列举属性为预留参数。

[HTTP Status Code](#)



# getMonitorDevices 接口

获取监听设备列表。

## 请求方式

POST/GET http://ip/V1.0/volume/getMonitorDevices

## 请求数据

无

## 返回数据

```
{ "monitorAudioDeivceList": [ { "audioSourceId": 0, "cardNo": 0, "devNo": 0, "deviceType": 0, "direction": 0, "id": 0, "isMic": true, "sourceType": 0 }, { "audioSourceId": 2, "cardName": "USB Capture HDMI+", "cardNo": 3, "devNo": 0, "deviceType": 1, "direction": 0, "id": 2, "isMic": true, "name": "HDMI [USB Capture HDMI+]", "sourceType": 0 }, { "audioSourceId": 2, "cardName": "C204221201227", "cardNo": 4, "devNo": 0, "deviceType": 1, "direction": 0, "id": 2, "isMic": true, "name": "C204221201227 [C204221201227] #0", "sourceType": 0 } ], "message": "SUCCESS", "status": 0 }
```

### 1. 返回状态

"status": 0

属性	说明
status	返回状态。0: 数据获取成功
message	返回状态描述
monitorAudioDeivceList	监听设备列表

### 2. 监听设备列表 monitorAudioDeivceList [{...}]

```
"monitorAudioDeivceList": [
  {
    "audioSourceId": 0,
    "cardNo": 0,
    "devNo": 0,
    "deviceType": 0,
    "direction": 0,
    "id": 0
  },
  {
    "audioSourceId": 1,
    "cardNo": 0,
    "devNo": 0,
    "deviceType": 0,
    "direction": 0,
    "id": 1
  }
]
```

属性	说明
audioSourceId	音频资源ID
deviceType	设备类型, 0:内置音频设备 1:外接音频设备
id	设备ID
devNo	设备序号
cardNo	声卡序号

[HTTP Status Code](#)

## monitorMicInput 接口

输出麦克风声音到监听设备。

### 请求方式

```
POST/GET http://ip/V1.0/volume/monitorMicInput?on=true
```

### 请求数据

参数	类型	说明
on	Bool	输出麦克风声音到监听设备, true:是 false:否

### 返回数据

[HTTP Status Code](#)

## outputMicToUsb 接口

输出麦克风声音到 USB-C。

### 请求方式

```
POST/GET http://ip/V1.0/volume/outputMicToUsb?on=true
```

### 请求数据

参数	类型	说明
on	Bool	是否输出麦克风声音到 USB-C, true:是 false:否

### 返回数据

[HTTP Status Code](#)

## setMonitorDevice 接口

设置监听设备。

### 请求方式

```
POST/GET http://ip/V1.0/volume/setMonitorDevice?deviceId=1&deviceType=0&cardNo=0&devNo=0
```

### 请求数据

参数	类型	说明
deviceId	Int	音频设备ID, 数据来自 <a href="#">/volume/getMonitorDevices</a>
deviceType	Int	音频设备类型, 数据来自 <a href="#">/volume/getMonitorDevices</a>
cardNo	Int	声卡序号, 数据来自 <a href="#">/volume/getMonitorDevices</a>
devNo	Int	设备序号, 数据来自 <a href="#">/volume/getMonitorDevices</a>

### 音频设备类型

音频设备类型	说明
0	内部设备, 如:耳机插孔
1	外部设备, 如:蓝牙设备、USB 设备

### 返回数据

[HTTP Status Code](#)

## setScope 接口

是否将音频设置为全局，即是否与其他场景共享配置。

### 请求方式

```
POST/GET http://ip/V1.0/volume/setScope?type=512&scope=1
```

### 请求数据

参数	类型	说明
type	Int	音频类型
scope	音频是否设置为全局，即是否与其他场景共享配置。 0: 共享; 1: 不共享	

### 音频类型

音频类型	说明
0x0001	监听
0x0003	直播
0x0700	HDMI 1
0x0800	HDMI 2
0x0200	麦克风
0x0002	视频
0x0300	蓝牙输入音频
0x0600	背景音乐

### 返回数据

[HTTP Status Code](#)

## setState 接口

设置音频状态。

### 请求方式

```
POST/GET http://ip/V1.0/volume/setState?type=3&state=1
```

### 请求数据

参数	类型	说明
type	Int	音频类型
state	Int	音频状态, 0:始终混合该音频 (不静音) 1: 始终不混合该音频 (静音) 2: 表示当前场景有此音频跟随的画面时, 才混合该音频.(默认值) (AFV)

### 音频类型

音频类型	说明
0x0001	监听
0x0003	直播
0x0700	HDMI 1
0x0800	HDMI 2
0x0200	麦克风
0x0002	视频
0x0300	蓝牙输入音频
0x0600	背景音乐

### 返回数据

[HTTP Status Code](#)

## setStreamAudioState 接口

根据名称设置输入音频状态。

### 请求方式

```
POST/GET http://ip/V1.0/volume/setStreamAudioState?name=srt&state=1
```

### 请求数据

参数	类型	说明
name	String	拉流源或 USB 输入源名称
state	Int	音频状态, 0:始终混合该音频 (不静音) 1: 始终不混合该音频 (静音) 2: 表示当前场景有此音频跟随的画面时, 才混合该音频.(默认值) (AFV)

### 返回数据

[HTTP Status Code](#)



## setStreamAudioVolume 接口

根据名称设置输入音频源的音量。

### 请求方式

```
POST/GET http://ip/V1.0/volume/setStreamAudioVolume?name=rtmp&volume=0
```

### 请求数据

参数	类型	说明
name	String	拉流源或 USB 输入源名称
volume	Int	音量 dB 值, -40 ~ 10

### 返回数据

[HTTP Status Code](#)

# setVolume 接口

设置音频的音量。

## 请求方式

POST/GET http://ip/V1.0/volume/setVolume?type=0&volume=0

## 请求数据

参数	类型	说明
type	Int	音频类型
volume	Int	音量 dB 值, -40 ~ 10

## 音频类型

音频类型	说明
0x0001	监听
0x0003	直播
0x0700	HDMI 1
0x0800	HDMI 2
0x0200	麦克风
0x0002	视频
0x0300	蓝牙输入音频
0x0600	背景音乐

## 返回数据

[HTTP Status Code](#)

## solo 接口

单独监听某路音频。

### 请求方式

```
POST/GET http://ip/V1.0/volume/solo?type=1792
```

### 请求数据

参数	类型	说明
type	Int	音频类型，仅可监听一路。若当前已处于单独监听状态，使用该接口取消单独监听。

### 音频类型

音频类型	说明
0x0001	监听
0x0003	直播
0x0700	HDMI 1
0x0800	HDMI 2
0x0200	麦克风
0x0002	视频
0x0300	蓝牙输入音频
0x0600	背景音乐

### 返回数据

[HTTP Status Code](#)