

设置频道虚拟人数及虚拟人

接口描述

- 1、设置频道虚拟人数及自定义虚拟人列表

2、接口支持https协议

接口URL

http://api.polyv.net/live/v4/channel/robot/setting-robot-list/update

[在线API调用](#)

请求方式

POST

接口约束

- 1、接口同时支持HTTP 、HTTPS ， 建议使用HTTPS 确保接口安全， 接口调用有频率限制， [详细请查看](#)
- 2、请设置请求头 `contentType:application/json`

请求参数描述

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

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

请求体参数描述

参数名	必选	类型	说明
channelId	true	Integer	频道号
robotNumber	true	Integer	聊天室显示虚拟人数，最大值为600000
virtualBookingNumber	false	Integer	虚拟预约人数
addRobotModel	true	String	人数显示模式 timely：立即生效 fixed_time：分时生效
changeTime	false	Integer	生效时间， addRobotModel为 fixed_time分时生效时必填，单位秒，最小值为20（20秒），最大值为18000（300分钟）
robotList	false	Array	自定义机器人列表，最多支持添加600000个机器人。1：不传值时使用内容库机器人。2：数量小于robotNumber时使用系统生成的机器人补充

robotList字段说明

参数名	必选	类型	说明
name	true	String	机器人昵称
avatar	true	String	机器人头像地址

示例

http://api.polyv.net/live/v4/channel/robot/setting-robot-list/update?
appId=xxx&sign=xxx×tamp=xxx&channelId=3335806

```
{
  "channelId": 3335806,
  "robotNumber": 500,
  "addRobotModel": "timely",
  "virtualBookingNumber": 100,
  "robotRandomMemberEnabled": "N",
  "robotList": [
    {
      "name": "观众1",
      "avatar": "https://liveimages.videocc.net/uploadimage/avatar/1.png"
    },
    {
      "name": "观众2",
      "avatar": "https://liveimages.videocc.net/uploadimage/avatar/2.png"
    }
  ]
}
```

响应参数描述

参数名	类型	说明
code	Integer	状态码，与 http 状态码相同，用于确定基本的响应状态
status	String	响应结果，由业务决定，成功返回 success，失败返回error
success	Boolean	是否成功响应

参数名	类型	说明
requestId	String	请求ID，每次请求生成的唯一的UUID，仅可用于排查、调试，不应该和业务挂上钩
error	Object	状态码非200时的错误信息【详见 Error字段描述 】
data	Object	成功响应时返回null

Error参数描述

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

Java请求示例

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

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

```

private static final Logger log = LoggerFactory.getLogger(getClass());
/**
 * 设置频道虚拟人数及虚拟人
 * @throws IOException
 * @throws NoSuchAlgorithmException
 */
@Test
public void updateRobotListSettingTest() throws IOException, NoSuchAlgorithmException {
    //公共参数,填写自己的实际参数
    String appId = super.appId;
    String appSecret = super.appSecret;
    String timestamp = String.valueOf(System.currentTimeMillis());

    //业务参数
    String url = "http://api.polyv.net/live/v4/channel/robot/setting-robot-list/update";
    String channelId = "3335806";

    //签名参数
    Map<String, String> requestMap = new HashMap<>();
    requestMap.put("appId", appId);
    requestMap.put("timestamp", timestamp);
    requestMap.put("channelId", channelId);
    requestMap.put("sign", LiveSignUtil.getSign(requestMap, appSecret));

    //请求体参数
    Map<String, Object> jsonMap = new HashMap<>();
    jsonMap.put("channelId", Integer.valueOf(channelId));
    jsonMap.put("robotNumber", 500);
    jsonMap.put("addRobotModel", "timely");
    List<Map<String, String>> robotList = new ArrayList<>();
    Map<String, String> robot1 = new HashMap<>();
    robot1.put("name", "观众1");
    robot1.put("avatar", "https://liveimages.videocc.net/uploadimage/avatar/1.png");
    robotList.add(robot1);
    jsonMap.put("robotList", robotList);

    url = HttpUtil.appendUrl(url, requestMap);
    String response = HttpUtil.postJsonBody(url, JSON.toJSONString(jsonMap), null);

    log.info("测试设置频道虚拟人数及虚拟人成功: {}", response);
    //do somethings
}

```

响应示例

系统全局错误说明详见[全局错误说明](#)

成功示例

```
{
  "code": 200,
  "status": "success",
  "requestId": "61b9319c9e3b42979f8a152dfa6b3f40.73.16626900842179815",
  "data": null,
  "success": true
}
```

异常示例

```
{
  "code": 400,
  "status": "error",
  "requestId": "4081dbac03e6441e8bdd301d8feee5a2.117.16360830583026105",
  "error": {
    "code": 30004,
    "desc": "找不到频道"
  },
  "success": false
}
```