

PG SOFT

对接文档

外部应用编程接口 (External API) v1.11



目录

更新记录	4
1. 概述	14
2. API 对接模式	15
2.1 API 格式	15
2.1.1 请求参数	15
2.1.2 返回结果	15
2.1.3 哈希校验 (任选项)	16
2.2 免费游戏 [最新版本: v1]	19
2.2.1 获取免费游戏	19
2.2.2 获取已转换为红利游戏/现金的免费游戏列表	24
2.2.3 获取已转换为红利游戏并再转为现金的免费游戏列表	27
2.2.4 获取免费游戏的玩家列表	30
2.2.5 获取免费游戏的未注册玩家列表	34
2.2.6 获取免费游戏的所有玩家详情	37
2.2.7 获取免费游戏的单个玩家详情	41
2.2.8 创建免费游戏	45
2.2.9 按投注金额创建免费游戏	48
2.2.10 将玩家转移到固定免费游戏中	51
2.2.11 将玩家转移到弹性免费游戏中	53
2.2.12 取消免费游戏	57
2.2.13 取消玩家的免费游戏	59
2.2.14 取消未注册玩家的免费游戏	61
2.2.15 取消玩家的所有免费游戏	63
2.2.16 获取免费游戏的玩家汇总	65
2.3 红利 [最新版本: v1]	68
2.3.1 获取红利游戏列表	68
2.3.2 获取已转换为现金的红利游戏列表	72
2.3.3 获取红利游戏的玩家列表	75
2.3.4 获取红利游戏的未注册玩家列表	79
2.3.5 获取红利游戏的所有玩家详情	82
2.3.6 获取红利游戏的单个玩家详情	86

2.3.7	创建红利游戏	90
2.3.8	将玩家转移到固定红利游戏中	93
2.3.9	将玩家转移到弹性红利游戏中	95
2.3.10	取消红利游戏	98
2.3.11	取消已注册玩家的红利游戏	100
2.3.12	取消未注册玩家的红利游戏	102
2.3.13	取消玩家的所有红利游戏	104
2.4	锦标赛 [最新版本: v2]	106
2.4.1	获取锦标赛列表	106
2.4.2	获取锦标赛玩家列表	112
2.4.3	获取锦标赛的最高排名	115
2.4.4	注册锦标赛玩家	118
2.4.5	获取锦标赛的现金奖励列表	121
2.4.6	创建锦标赛	125
2.4.7	更新锦标赛	129
2.4.8	更新附属锦标赛信息	131
2.4.9	获取锦标赛信息	133
2.4.10	创建锦标赛信息	136
2.4.11	更新锦标赛信息	138
2.4.12	删除锦标赛信息	140
2.4.13	删除锦标赛玩家	142
2.4.14	重置玩家	144
2.5	投注历史记录 [最新版本: v4]	146
2.5.1	获取历史记录	147
2.5.2	获取特定时间内的历史记录	152
2.5.3	获取单一玩家的历史记录	157
2.5.4	获取玩家的每日汇总	161
2.5.5	获取玩家在特定时间内的每日汇总	165
2.5.6	获取投注汇总	169
2.5.7	获取每小时投注汇总	174
2.5.8	获取玩家的投注总和	177
2.5.9	获取玩家未完成的投注	181
2.5.10	获取投注详情	184
2.5.11	获取失败投注	187

2.5.12	待处理投注.....	189
2.5.13	获取待处理投注.....	191
2.5.14	手动解决待处理投注.....	193
2.5.15	手动重置待处理游戏状态.....	196
2.6	游戏 [最新版本: v2].....	198
2.6.1	获取游戏列表.....	198
2.6.2	获取游戏的合法投注金额.....	202
2.6.3	更改游戏发布状态.....	205
2.6.4	获取游戏证书.....	207
2.7	现金 [最新版本: v2].....	209
2.7.1	获取钱包交易.....	209
2.7.2	获取手动调整详情.....	212
2.8	玩家 [最新版本: v3].....	215
2.8.1	获取在线玩家人数.....	215
2.8.2	获取玩家的线上状态.....	217
2.8.3	获取在线玩家列表.....	219
2.8.4	创建玩家.....	221
2.8.5	踢出玩家.....	223
2.8.6	冻结玩家.....	225
2.8.7	恢复玩家.....	227
2.8.8	检查玩家状态.....	229
3.	附录.....	231
3.1	平台.....	231

更新记录

版本	日期	描述
1.0	2019-04-03	初稿
1.1	2019-04-12	添加新内容 <ul style="list-style-type: none"> 2.1.5 获取免费游戏的未注册玩家 2.1.12 取消已注册玩家的免费游戏 2.2.4 取消未注册的红利玩家 2.2.11 取消已注册玩家的红利 2.4.6 取消投注汇总 修订概述 修订 API 请求参数 <ul style="list-style-type: none"> 2.1.6 创建免费游戏 2.1.11 取消为主场玩家的免费游戏 2.2.5 创建红利 2.2.10 取消未注册玩家的红利 修订 API 描述 <ul style="list-style-type: none"> 2 API 对接模式 2.1.2 获取已转换为红利的免费游戏列表 2.1.3 获取已转换为现金的免费游戏列表 2.1.10 取消已注册玩家的免费游戏 2.1.11 取消未注册玩家的免费游戏 2.2.2 获取已转换为现金的红利列表 2.2.9 取消已注册玩家的红利 2.2.10 取消未注册玩家的红利
1.2	2019-04-15	为所有投注历史的 API 添加了注意 (2.4)
1.3	2019-04-17	添加新内容 <ul style="list-style-type: none"> 2.1.6 获取免费游戏的所有玩家 2.1.7 为玩家获取免费游戏 2.2.5 获取所有红利玩家 2.2.6 获取玩家的红利 修订 API 描述 & 请求描述 <ul style="list-style-type: none"> 2.1.4 获取免费游戏玩家 2.1.5 获取免费游戏的未注册玩家

		<ul style="list-style-type: none"> 2.2.3 获取红利玩家 2.1.5 获取未注册的红利玩家 2.4.6 获取投注汇总 2.5.1 获取游戏列表 修订 API 请求 <ul style="list-style-type: none"> 2.1.5 获取免费游戏的未注册玩家 2.1.7 创建免费游戏 2.2.4 获取未注册的红利玩家 2.2.6 创建红利 修订 API 响应 <ul style="list-style-type: none"> 2.1.6 获取免费游戏的所有玩家 2.1.7 创建免费游戏 2.2.5 获取所有红利玩家 2.2.6 创建红利 2.4.6 获取投注汇总 2.5.2 获取游戏列表
1.4	2019-07-12	添加新内容 <ul style="list-style-type: none"> 2.1.9 按投注金额创建免费游戏 2.3.1 获取锦标赛 2.3.2 获取锦标赛玩家 2.5.2 获取游戏的合法投注金额 修订 API 描述 <ul style="list-style-type: none"> 2.5.1 获取游戏列表 (v2) 修订 API 请求 <ul style="list-style-type: none"> 2.1.8 创建免费游戏 修订 API 响应 <ul style="list-style-type: none"> 2.1.1 获取免费游戏
1.5	2019-08-22	修订 API 描述 <ul style="list-style-type: none"> 2.1.1 获取免费游戏 2.1.2 获取已转换为红利/现金的免费游戏列表 2.1.3 获取已转换为现金的免费游戏红利列表 2.1.4 获取免费游戏玩家 2.1.6 获取免费游戏的所有玩家 2.2.1 获取红利

		<ul style="list-style-type: none"> 2.2.2 获取已转换为现金的红利列表 2.2.3 获取红利玩家 2.2.5 获取所有红利玩家 2.3.1 获取锦标赛 2.3.2 获取锦标赛玩家 2.4.5 获取玩家在特定时间内的每日汇总 修订 API 请求 <ul style="list-style-type: none"> 2.1.7 为玩家获取免费游戏 2.2.9 将玩家转移到弹性红利中 2.4.1 获取历史记录 修订 API 响应 <ul style="list-style-type: none"> 2.4.1 获取历史记录 2.4.2 获取特定时间的历史记录 2.4.3 获取单一玩家的历史记录 2.4.6 获取投注汇总 添加新内容 <ul style="list-style-type: none"> 2.4.7 获取玩家的投注总和
1.6	2019-11-13	添加新内容 <ul style="list-style-type: none"> 2.4.7 获取每小时投注汇总 修订 API 版本 <ul style="list-style-type: none"> 2.4 投注历史记录 修订 API 描述 <ul style="list-style-type: none"> 2.4 投注历史记录 修订 API 描述 <ul style="list-style-type: none"> 2.4.1 投注历史记录 修订 API 响应 <ul style="list-style-type: none"> 2.4.1 获取历史记录 修订 API 响应 <ul style="list-style-type: none"> 2.4.2 获取特定时间的历史记录 修订 API 响应 <ul style="list-style-type: none"> 2.4.3 获取单个玩家的历史记录
1.7	2020-02-10	添加新内容 <ul style="list-style-type: none"> 2.5.3 更改游戏发布状态 2.7.1 获取在线玩家人数

		<ul style="list-style-type: none"> 2.7.2 获取玩家线上状态 修订 API 响应 <ul style="list-style-type: none"> 2.4.1 获取历史记录 2.4.2 获取特定时间的历史记录 2.4.3 获取单个玩家的历史记录 2.4.4 获取玩家的每日汇总 2.4.5 获取玩家在特定时间内的每日汇总 2.4.6 获取投注汇总 2.4.7 获取每小时投注汇总 2.4.8 获取玩家的投注总和 2.6.1 获取累积奖池列表 修订 API 响应参数描述 <ul style="list-style-type: none"> 2.1.6 获取免费游戏的所有玩家详情 2.1.7 获取免费游戏的单个玩家详情 2.2.5 获取红利游戏的所有玩家详情 2.2.6 获取红利游戏的单个玩家详情
1.71	2020-02-27	修订 API 请求 <ul style="list-style-type: none"> 2.4.5 获取玩家在特定时间内的每日汇总
1.8	2020-12-11	添加新的 API 版本 <ul style="list-style-type: none"> 2.7 玩家 添加新内容 <ul style="list-style-type: none"> 2.3.5 获取锦标赛的现金奖励列表 2.4.9 获取玩家未完成的投注 2.4.10 获取投注详情 2.4.11 获取失败投注 2.4.12 待处理投注 <ul style="list-style-type: none"> 2.4.12.1. 获取待处理投注 2.4.12.2. 手动解决待处理投注 2.4.13 手动重置待处理游戏状态 2.7.3 获取在线玩家列表 2.7.4 创建玩家 2.7.5 踢出玩家 2.7.6 冻结玩家 2.7.7 恢复玩家

- 2.7.8 检查玩家状态

修订 API 标题或描述

- 2.1.3 获取已转换为红利游戏并再转为现金的免费游戏列表
- 2.1.10 将玩家转移到固定免费游戏中
- 2.1.13 取消玩家的免费游戏
- 2.2.6 获取红利游戏的单个玩家详情
- 2.2.7 创建红利游戏
- 2.4 投注历史记录
- 2.4.1 获取历史记录
- 2.4.2 获取特定时间内的历史记录
- 2.4.3 获取单一玩家的历史记录
- 2.4.4 获取玩家的每日汇总
- 2.4.5 获取玩家在特定时间内的每日汇总
- 2.4.6 获取投注汇总
- 2.4.7 获取每小时投注汇总
- 2.4.8 获取玩家的投注总和
- 2.4.9 获取玩家未完成的投注
- 2.4.10 获取投注详情
- 2.4.11 获取失败投注
- 2.4.12 待处理投注
- 2.6.1 获取累积奖池列表
- 2.7.2 获取玩家的线上状态

修订 API 请求或响应内容

- 2.1.1 获取免费游戏
- 2.1.4 获取免费游戏玩家
- 2.1.5 获取免费游戏的未注册玩家
- 2.1.6 获取免费游戏的所有玩家详情
- 2.1.7 获取免费游戏的单个玩家详情
- 2.1.8 创建免费游戏
- 2.1.9 按投注金额创建免费游戏
- 2.1.10 将玩家转移到固定免费游戏中
- 2.1.11 将玩家转移到弹性免费游戏中
- 2.2.1 获取红利游戏
- 2.2.2 获取已转换为现金的红利游戏列表

		<ul style="list-style-type: none"> • 2.2.3 获取红利游戏玩家 • 2.2.4 获取未注册的红利游戏玩家 • 2.2.5 获取红利游戏的所有玩家详情 • 2.2.6 获取红利游戏的单个玩家详情 • 2.2.9 将玩家转移到弹性红利游戏中 • 2.2.13 取消玩家的所有红利游戏 • 2.3.1 获取锦标赛 • 2.3.2 获取锦标赛玩家 • 2.3.4 注册锦标赛玩家 • 2.3.5 获取锦标赛的现金奖励列表 • 2.4.1 获取历史记录 • 2.4.2 获取特定时间内的历史记录 • 2.4.3 获取单一玩家的历史记录 • 2.4.4 获取玩家的每日汇总 • 2.4.5 获取玩家在特定时间内的每日汇总 • 2.4.6 获取投注汇总 • 2.4.7 获取每小时投注汇总 • 2.4.8 获取玩家的投注总和 • 2.4.9 获取玩家未完成的投注 • 2.4.10 获取投注详情 • 2.4.11 获取失败投注 • 2.4.12 待处理投注 • 2.6.1 获取累积奖池列表 • 2.7.2 获取玩家的线上状态
1.9	2021-05-20	添加新的 API 响应参数 <ul style="list-style-type: none"> • 2. API 方法的所有 API 添加新的 API <ul style="list-style-type: none"> • 2.4.6 创建锦标赛 • 2.4.7 更新锦标赛 • 2.4.8 创建免费游戏 • 2.4.9 获取锦标赛信息 • 2.4.10 创建锦标赛信息 • 2.4.11 更新锦标赛信息 • 2.4.12 删除锦标赛信息

		<ul style="list-style-type: none"> 2.4.13 删除锦标赛玩家 2.4.14 重置玩家 <p>修订 API 响应</p> <ul style="list-style-type: none"> 2.5.7 获取每小时投注汇总 <p>修订 API 响应描述</p> <ul style="list-style-type: none"> 2.5.7 获取每小时投注汇总 2.5.8 获取玩家的投注总和 <p>修订 API 请求</p> <ul style="list-style-type: none"> 2.5.8 获取玩家的投注总和 2.5.9 获取玩家未完成的投注 <p>添加新内容</p> <ul style="list-style-type: none"> 2.1.3 哈希校验
1.10	2021-08-20	<p>修订 API 请求或响应内容</p> <ul style="list-style-type: none"> 2.1.3 哈希校验 2.2.4 获取免费游戏玩家 2.2.6 获取免费游戏的所有玩家详情 2.2.7 获取免费游戏的单个玩家详情 2.2.8 创建免费游戏 2.2.9 按投注金额创建免费游戏 2.2.10 将玩家转移到固定免费游戏中 2.2.11 将玩家转移到弹性免费游戏中 2.2.15 取消玩家的所有免费游戏 2.3.3 获取红利游戏玩家 2.3.5 获取红利游戏的所有玩家详情 2.3.6 获取红利游戏的单个玩家详情 2.3.7 创建红利游戏 2.3.8 将玩家转移到固定红利游戏中 2.3.9 将玩家转移到弹性红利游戏中 2.4.1 获取锦标赛列表 2.4.2 获取锦标赛玩家 2.4.3 获取锦标赛的最高排名 2.4.5 获取锦标赛的现金奖励列表 2.4.6 创建锦标赛 2.4.7 更新锦标赛

		<ul style="list-style-type: none"> • 2.4.8 更新附属锦标赛信息 • 2.4.9 获取锦标赛信息 • 2.4.10 创建锦标赛信息 • 2.4.11 更新锦标赛信息 • 2.5.1 获取历史记录 • 2.5.2 获取特定时间内的历史记录 • 2.5.3 获取单一玩家的历史记录 • 2.5.6 获取投注汇总 • 2.5.7 获取每小时投注汇总 • 2.5.8 获取玩家的投注总和 • 2.5.9 获取玩家未完成的投注 • 2.5.10 获取投注详情 • 2.5.13 获取待处理投注 • 2.6.1 获取游戏列表 • 2.9.3 获取在线玩家列表 <p>添加新的 API</p> <ul style="list-style-type: none"> • 2.2.16 获取免费玩家汇总 • 2.6.4 获取游戏证书 • 2.7.1 获取钱包交易 • 2.7.2 获取手动调整详情
1.11	2022-06-27	<p>删除 API</p> <ul style="list-style-type: none"> • 2.8 累积奖池 <p>修订 API 请求或响应内容</p> <ul style="list-style-type: none"> • 2.1.3 哈希校验 (任选项) • 2.2.1 获取免费游戏 • 2.2.2 获取已转换为红利游戏/现金的免费游戏列表 • 2.2.3 获取已转换为红利游戏并再转为现金的免费游戏列表 • 2.2.4 获取免费游戏的玩家列表 • 2.2.5 获取免费游戏的未注册玩家列表 • 2.2.6 获取免费游戏的所有玩家详情 • 2.2.7 获取免费游戏的单个玩家详情 • 2.2.8 创建免费游戏

- 2.2.9 按投注金额创建免费游戏
- 2.2.12 取消免费游戏
- 2.2.13 取消玩家的免费游戏
- 2.2.14 取消未注册玩家的免费游戏
- 2.2.15 取消玩家的所有免费游戏
- 2.2.16 获取免费玩家汇总
- 2.3.1 获取红利游戏列表
- 2.3.2 获取已转换为现金的红利游戏列表
- 2.3.3 获取红利游戏的玩家列表
- 2.3.4 获取红利游戏的未注册玩家列表
- 2.3.5 获取红利游戏的所有玩家详情
- 2.3.6 获取红利游戏的单个玩家详情
- 2.3.7 创建红利游戏
- 2.3.10 取消红利游戏
- 2.3.11 取消已注册玩家的红利游戏
- 2.3.12 取消未注册玩家的红利游戏
- 2.3.13 取消玩家的所有红利游戏
- 2.4.2 获取锦标赛玩家
- 2.4.3 获取锦标赛的最高排名
- 2.4.4 注册锦标赛玩家
- 2.4.6 创建锦标赛
- 2.5.1 获取历史记录
- 2.5.2 获取特定时间内的历史记录
- 2.5.3 获取单一玩家的历史记录
- 2.5.5 获取玩家在特定时间内的每日汇总
- 2.5.6 获取投注汇总
- 2.5.8 获取玩家的投注总和
- 2.5.10 获取投注详情
- 2.5.13 获取待处理投注
- 2.5.14 手动解决待处理投注

- 2.5.15 手动重置待处理游戏状态
- 2.6.1 获取游戏列表
- 2.7.1 获取钱包交易
- 2.7.2 获取手动调整详情
- 2.8.3 获取在线玩家列表

1. 概述

外部 API 让您得以透过编程方式使用 PGSoft 工具和服务。您可以通过这些 API 检索信息，并创建、修改或删除 PG 所提供的各种服务，例如免费游戏、红利游戏和锦标赛等。

PGSoft 的 API 皆按服务划分。每种服务在本文档中都有个别的章节。本文档所包含的所有 API 都是可选的，运营商可依据需求选择 API 对接。

2. API 对接模式

2.1 API 格式

2.1.1 请求参数

外部 API 使用了 HTTP 表单请求方式及 RESTful 端点结构。请使用以下内容类型定义请求格式：

`Content-Type: application/x-www-form-urlencoded`

2.1.2 返回结果

API 的返回结果为 JSON 格式。系统将返回 HTTP 状态码 200 以表示成功与失败的请求。h

所有 API 方式

`Content-Type: application/json`

参数：

参数名称	数据类型	描述
data ¹	JSON Object	API 对接模式的响应信息
error ²	JSON Object	发生错误或异常时的错误信息

例子：

成功的返回结果

```
{
  "data": {
    [API method response. JSON object format may vary depending on API methods]
  },
  "error": null,
}
```

失败/错误的返回结果

```
{
  "data": null,
  "error": {
    "code": "[Error code]",
    "message": "[Error message]"
  }
}
```

¹ 此栏目将在错误响应中显示空值。

² 此栏目将在成功应用编程接口响应中显示空值。请参阅有关[错误响应](#)的章节以了解更多关于错误响应的格式。

2.1.3 哈希校验（任选项）

我们强烈建议运营商使用哈希校验以保护每个请求。若运营商选择使用哈希校验，PG 系统将对每个请求头字段中的哈希信息进行验证。

2.1.3.1 标头字段

下表描述了下述示例中的各种请求头：

组成	说明
Host	API 域名 例子： apiexample.pgsoft.com
x-date	以 YYYYMMDD 为格式的当前日期（UTC） 例子： 20190902
x-content-sha256	请求主体内容字符串的 SHA256 哈希值 例子： 1700116101f424b9f6fc695b4dbaf2b7b0ee763ba1b3b53298e3069143ed46f1
Authorization	多个组成部分的组合，用于请求认证 例子： PWS-HMAC-SHA256 Credential=20190902/OPERATOR_TOKEN_EXAMPLE/pws/v1, SignedHeaders=host;x-content-sha256;x-date, Signature=d78220cf06ae85f9d1db11dad9c3fd926799619eab3d28574aadb8cf328cd7aa

例子

```
Host:
apiexample.pgsoft.com
x-date: 20190902
x-content-sha256:
1700116101f424b9f6fc695b4dbaf2b7b0ee763ba1b3b53298e3069143ed46f1
Authorization: PWS-HMAC-SHA256
Credential=20190902/OPERATORTOKENEXAMPLE/pws/v1,SignedHeaders=
host;x-content-sha256;x-date,Signature=
d78220cf06ae85f9d1db11dad9c3fd926799619eab3d28574aadb8cf328cd7aa
```

请求头: x-content-sha256

计算请求主体内容字符串的 SHA256 哈希值:

例子

```
POST 主体:
operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&count=5000&bet_
type=1&row_version=1346592723000

x-content-sha256:
1700116101f424b9f6fc695b4dbaf2b7b0ee763ba1b3b53298e3069143ed46f1
```

请求头: 验证

下表描述了下述示例中 Authorization 标头值的各个组成部分:

组成	说明
Credential	<p>您的访问密钥 ID 和范围信息, 包括用于计算签名的日期, 运营商令牌和服务。</p> <p>该字符串具有以下形式:</p> <pre>{x-date}/{operator_token}/pws/v1</pre> <p>例子:</p> <pre>20190902/OPERATOR_TOKEN_EXAMPLE/pws/v1</pre>
SignedHeaders	<p>以分号分隔的请求头列表, 用于计算签名。该列表仅包含标头名称, 并且标头名称必须为小写。</p> <p>固定值:</p> <pre>host;x-content-sha256;x-date</pre>
Signature	<p>以 64 个小写的十六进制字符表示的 256-bit 签名。</p> <p>例子:</p> <pre>d78220cf06ae85f9d1db11dad9c3fd926799619eab3d28574aadb8cf328cd7aa</pre>

要计算签名, 您首先需要包含 {host} {x-content-sha256} {x-date} 的签名字符串。接着, 您将可以使用签名密钥计算出签名字符串的 HMAC-SHA256 哈希值:

```
hmac-sha256({salt}, {host}{x-content-sha256}{x-date})
```

计算主体内容字符串的 HMAC-SHA256 哈希值:

例子

```
Salt           : SALT_EXAMPLE
Host           : api.example.pgsoft.com
x-date         : 20190902
x-content-sha256 :
1700116101f424b9f6fc695b4dbaf2b7b0ee763ba1b3b53298e3069143ed46f1
```

签名:

```
d78220cf06ae85f9d1db11dad9c3fd926799619eab3d28574aadb8cf328cd7aa
```

2.2 免费游戏 [最新版本: v1]

2.2.1 获取免费游戏

获取已创建的免费游戏列表:

请求参数

API URL : {PgSoftAPIDomain}/FreeGame/v1/GetFreeGames

HTTP 方法 : POST

URL 参数:

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注: • 请把参数值设为 GUID 格式

主体参数:

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
page_number	Integer	是	返回数据的批号
row_count	Integer	是	个别批次的记录数量 注: 值度 1-5000
from_time	Long	是	免费游戏的开始时间范围
to_time	Long	是	(以毫秒为单位的 Unix 时间戳)
from_created_time	Long	否	免费游戏的创建时间范围
to_created_time	Long	否	(以毫秒为单位的 Unix 时间戳)
free_game_id	Integer	否	免费游戏的唯一标识符
currency	String	否	免费游戏的货币
free_game_name	String	否	免费游戏的名称

free_game_transfer_type	Integer	否	免费游戏的转换类型 0: 固定 3: 弹性
conversion_type	Char	否	完成免费游戏后, 红利的转换类型 C: 现金 B: 红利
status	Integer	否	免费游戏的状态 0: 无效 1: 活跃 2: 已过期

例子:

URL:

https://api.pg-bo.me/external/FreeGame/v1/GetFreeGames?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&page_number=1&row_count=500&from_time=1346592723000&to_time=1546592723000&from_created_time=1346592723000&to_created_time=1546592723000&free_game_id=12345¤cy=CNY&free_game_name=freegame2019&free_game_transfer_type=1&conversion_type=C&status=1

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
totalCount	Integer	记录总数
totalPage	Integer	可用批次的总数
result	JSON Array	免费游戏列表

JSON 数组

参数名称	数据类型	描述
operatorToken	String	运营商的唯一标识符
transaction_id	String	交易的唯一标识符
freeGameId	Integer	免费游戏的唯一标识符
freeGameName	String	免费游戏的名称
isEvent	Boolean	免费游戏类型的指标 True: 弹性 False: 固定
gameIds	Integer Array	游戏的唯一标识符
totalGame	Integer	每位玩家将获得的免费游戏总数
currencyCode	String	免费游戏的货币
coinSize	Decimal	免费游戏的投注大小
multiplier	Integer	免费游戏的投注倍数
minimumConversionAmount	Decimal	免费游戏的最低现金转换金额
maximumConversionAmount	Decimal	免费游戏的最高现金转换金额
conversionType	Char	完成免费游戏后，红利的转换类型 C: 现金 B: 红利
bonusRatio	Decimal	红利游戏的流水要求 (指免费游戏里的所赢金额在进入红利钱包之后，需要完成的免费游戏投注额倍数)
bonusRatioAmount	Decimal	红利游戏的流水要求 (指免费游戏里的所赢金额在进入红利钱包之后，需要完成的总金额)

bonusMinimumConversionAmount	Decimal	红利游戏的最低现金转换金额
bonusMaximumConversionAmount	Decimal	红利游戏的最高现金转换金额
bonusType	Integer	红利游戏的转换类型: 0: 红利 1: 现金 2: 红利 & 现金
status	Integer	免费游戏的状态 0: 无效 (已取消) 1: 活跃 2: 已过期
freeGameCountGiven	Integer	给予的免费游戏总数
numberOfPlayers	Integer	免费游戏中的玩家总数
expiredDate	Long	免费游戏的截止日期 (以毫秒为单位的 Unix 时间戳)
createdDate	Long	免费游戏的创建日期 (以毫秒为单位的 Unix 时间戳)
createdBy	String	创建免费游戏的 API 或 BackOffice 用户
updatedBy	String	更新免费游戏的 API 或 BackOffice 用户
isSupressDiscard	Boolean	允许玩家取消优惠的权限 True:不允许玩家取消优惠 False:允许玩家取消优惠

例子:

```
{
  "data": {
    "totalCount": 2583,
    "totalPage": 259,
    "result": [
      {
        "operatorToken": "abcd",
        "freeGameId": 1,
        "freeGameName": "free game1",

```

```

    "transactionId": "12345Test",
    "isEvent": false,
    "gameIds": [1],
    "totalGame": 20,
    "currencyCode": "USD",
    "coinSize": 0.05,
    "multiplier": 2,
    "minimumConversionAmount": 1000.00,
    "maximumConversionAmount": 1000.00;
    "conversionType": "B",
    "bonusRatio": 2,
    "bonusRatioAmount": 0,
    "bonusMinimumConversionAmount": 1000.00,
    "bonusMaximumConversionAmount": 1500.00;
    "bonusType": 0,
    "status": 2,
    "freeGameCountGiven": 0,
    "numberOfPlayers": 0,
    "expiredDate": 1499764521000,
    "createdDate": 1498036782000,
    "createdBy": "player1",
    "updatedBy": "player1",
    "isSupressDiscard": true
  },
  {
    "operatorToken": "abcd",
    "freeGameId": 2,
    "freeGameName": "free game2",
    "transactionId": "12345Test2",
    "isEvent": false,
    "gameIds": [1],
    "totalGame": 20,
    "currencyCode": "USD",
    "coinSize": 0.05,
    "multiplier": 2,
    "conversionType": "C",
    "bonusRatio": 0,
    "bonusRatioAmount": 0,
    "status": 1,
    "freeGameCountGiven": 0,
    "numberOfPlayers": 0,
    "expiredDate": 1499732101000,
    "createdDate": 1498114200000,
    "createdBy": "player1",
    "updatedBy": "player1",
    "isSupressDiscard": true
  }
]
},
"error": null
}

```


2.2.2 获取已转换为红利游戏/现金的免费游戏列表

获取已转换为红利游戏或现金的免费游戏列表：

请求参数

API URL : {PgSoftAPIDomain}/FreeGame/v1/GetFreeGameConvertedWallet

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
page_number	Integer	是	返回数据的批号
row_count	Integer	是	个别批次的记录数量 注：值度 1-5000
free_game_id	Integer	是	免费游戏的唯一标识符
from_transaction_time	Long	是	免费游戏玩家交易的时间范围 (以毫秒为单位的 Unix 时间戳)
to_transaction_time	Long	是	
player_name	String	否	玩家的唯一标识符 注：实行模糊匹配

例子:

URL:

https://api.pg-bo.me/external/FreeGame/v1/GetFreeGameConvertedWallet?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&page_number=1&row_count=500&free_game_id=12345&player_name=player1&from_transaction_time=1346592723000&to_transaction_time=1346592723000

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
totalConvertedAmount	Decimal	已通过免费游戏转换成现金的总金额
totalCount	Integer	记录总数
totalPage	Integer	可用批次的总数
result	JSON Array	免费游戏列表

JSON 数组

参数名称	数据类型	描述
balanceId	Integer	免费游戏玩家钱包的唯一标识符
playerName	String	玩家的唯一标识符
currencyCode	String	玩家的货币
gameId	Integer	游戏的唯一标识符
transactionAmount	Decimal	已转换的玩家总数
transactionDateTime	Long	转换时间

例子:

```
{
  "data": {
    "totalConvertedAmount": 10,
    "totalCount": 1,
    "totalPage": 1,
    "result": [
      {
        "balanceId": 13787,
        "playerName": "qa18",
        "currencyCode": "CNY",
        "gameId": 8,
        "transactionAmount": 10,
        "transactionDateTime": 1547449915000
      },
      {
        "balanceId": 13788,
        "playerName": "qa18",
        "currencyCode": "CNY",
        "gameId": 8,
        "transactionAmount": 10,
        "transactionDateTime": 1547449915100
      }
    ],
    "error": null
  }
}
```

2.2.3 获取已转换为红利游戏并再转为现金的免费游戏列表

获取已从免费游戏转换为红利游戏，并再从红利游戏转换为现金的游戏列表：

请求参数

API URL : {PgSoftAPIDomain}/FreeGame/v1/GetFreeGameBonusConvertedWallet

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
page_number	Integer	是	返回数据的批号
row_count	Integer	是	个别批次的记录数量 注：值度 1-5000
free_game_id	Integer	是	免费游戏的唯一标识符
from_transaction_time	Long	是	免费游戏玩家交易的时间范围 (以毫秒为单位的 Unix 时间戳)
to_transaction_time	Long	是	
player_name	String	否	玩家的唯一标识符 注：实行模糊匹配
status	Integer	否	红利游戏状态 0: 无效 1: 活跃 2: 已过期

例子:

URL:

https://api.pg-bo.me/external/FreeGame/v1/GetFreeGameBonusConvertedWallet?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&page_number=1&row_count=500&free_game_id=12345&player_name=player1&from_transaction_time=1346592723000&to_transaction_time=1346592723000&status=1

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
totalConvertedAmount	Decimal	已通过免费游戏转换为现金的总金额
totalCount	Integer	记录总数
totalPage	Integer	可用批次的总数
result	JSON Array	免费游戏列表

JSON 数组

参数名称	数据类型	描述
balanceId	Integer	免费游戏玩家钱包的唯一标识符
playerName	String	玩家的唯一标识符
currencyCode	String	玩家的货币
gameId	Integer	游戏的唯一标识符
transactionAmount	Decimal	已转换的玩家总数
transactionDateTime	Long	转换时间 (以毫秒为单位的 Unix 时间戳)

例子:

```
{
  "data": {
    "totalConvertedAmount": 10,
    "totalCount": 1,
    "totalPage": 1,
    "result": [
      {
        "balanceId": 13787,
        "playerName": "qa18",
        "currencyCode": "CNY",
        "gameId": 8,
        "transactionAmount": 10,
        "transactionDateTime": 1547449915000
      },
      {
        "balanceId": 13788,
        "playerName": "qa18",
        "currencyCode": "CNY",
        "gameId": 8,
        "transactionAmount": 10,
        "transactionDateTime": 1547449915100
      }
    ],
    "error": null
  }
}
```

2.2.4 获取免费游戏的玩家列表

获取免费游戏的已注册玩家列表：

请求参数

API URL : {PgSoftAPIDomain}/FreeGame/v1/GetFreeGamePlayers

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
page_number	Integer	是	返回数据的批号
row_count	Integer	是	个别批次的记录数量 注：值度 1-5000
free_game_id	Integer	是	免费游戏的唯一标识符
from_time	Long	是	免费游戏的开始时间范围 (以毫秒为单位的 Unix 时间戳)
to_time	Long	是	
player_name	String	否	免费游戏玩家的唯一标识符 注：实行模糊匹配
from_created_time	Long	否	玩家被注册到免费游戏的时间范围 (以毫秒为单位的 Unix 时间戳)
to_created_time	Long	否	
free_game_name	String	否	免费游戏的名称
status	Integer	否	免费游戏钱包的状态 0: 无效 (已取消) 1: 活跃 2: 已过期

			3: 已转换
			4: 已完成
			5: 全新
			6: 已取消用户
			8: 待处理

例子:

URL:

https://api.pg-bo.me/external/FreeGame/v1/GetFreeGamePlayers?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&page_number=1&row_count=500&player_name=player1&from_time=1346592723000&to_time=1546592723000&from_created_time=1346592723000&to_created_time=1546592723000&free_game_id=12345&free_game_name=freegame2019&status=1

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
totalCount	Integer	记录总数
totalPage	Integer	可用批次的总数
result	JSON Array	免费游戏列表

JSON 数组

参数名称	数据类型	描述
balanceId	Integer	免费游戏玩家钱包的唯一标识符
playerId	Integer	免费游戏玩家的唯一标识符
playerName	String	免费游戏玩家的唯一标识符
walletType	String	免费游戏玩家的钱包类型 G: 免费游戏 B: 红利
balanceAmount	Decimal	玩家的免费游戏余额
bonusRatioAmount	Decimal	红利游戏的流水要求 (指免费游戏里的所赢金额在进入红利钱包之后, 需要完成的总金额)

bonusMaximumConversionAmount	Decimal	免费游戏红利的最高转换金额
freeGameMaximumConversionAmount	Decimal	免费游戏的最高转换金额
freeGameCount	Integer	玩家的免费游戏次数
status	Integer	免费游戏钱包的状态 0: 无效（已取消） 1: 活跃 2: 已过期 3: 已转换 4: 已完成 5: 全新 6: 已取消用户 8: 待处理
expiredDate	Long	免费游戏的截止日期（以毫秒为单位的 Unix 时间戳）
createdBy	String	创建免费游戏的 API 或 BackOffice 用户
updatedBy	String	更新免费游戏的 API 或 BackOffice 用户
updatedAt	Long	玩家被注册或重新注册到免费游戏的日期（以毫秒为单位的 Unix 时间戳）
createdAt	Long	玩家被注册到免费游戏的日期（以毫秒为单位的 Unix 时间戳）

例子:

```
{
  "data": {
    "totalCount": 2583,
    "totalPage": 259,
    "result": [
      {
        "balanceId": 111261,
        "playerId": 7443,
        "playerName": "player1",
        "walletType": "B",
        "balanceAmount": 874.4,
        "bonusRatioAmount": 874.4,
        "bonusMaximumConversionAmount": 0,
        "freeGameMaximumConversionAmount": 0,
        "expiredDate": 1546991999000,
        "status": 2,
        "createdBy": "operator",
        "updatedBy": "operator",
        "createdDate": 1546923615000,
        "updatedAt": 1546991999000
      },
      {
        "balanceId": 111252,
        "playerId": 7443,
        "playerName": "player2",
        "walletType": "G",
        "balanceAmount": 8.8,
        "bonusMaximumConversionAmount": 0,
        "freeGameMaximumConversionAmount": 0,
        "freeGameCount": 6,
        "expiredDate": 1546991999000,
        "status": 6,
        "createdBy": "operator",
        "updatedBy": "operator",
        "createdDate": 1546918715000,
        "updatedAt": 1547028195000
      }
    ]
  },
  "error": null
}
```

2.2.5 获取免费游戏的未注册玩家列表

获取免费游戏的未注册玩家列表：

请求参数

API URL : {PgSoftAPIDomain}/FreeGame/v1/GetFreeGamePendingPlayers

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
page_number	Integer	是	返回数据的批号
row_count	Integer	是	个别批次的记录数量 注：值度 1-5000
free_game_id	Integer	是	免费游戏的唯一标识符
player_name	String	否	免费游戏未注册玩家的唯一标识符 注：实行模糊匹配
created_by	String	否	注册玩家的 API 或 BackOffice 用户
status	Integer	否	待定钱包的状态 0: 无效 (已取消) 1: 待处理 2: 已转移

例子:

URL:

https://api.pg-bo.me/external/FreeGame/v1/GetFreeGamePendingPlayers?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&page_number=1&row_count=500&player_name=player1&free_game_id=12345&created_by=external&status=1

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
totalCount	Integer	记录总数
totalPage	Integer	可用批次的总数
result	JSON Array	免费游戏未注册玩家列表

JSON 数组

参数名称	数据类型	描述
pendingWalletId	Integer	免费游戏玩家待定钱包的唯一标识符
playerName	String	免费游戏未注册玩家的唯一标识符
status	Integer	待定钱包的状态 0: 无效 (已取消) 1: 待处理 2: 已转移
updatedAt	Long	玩家被注册或重新注册到免费游戏的日期 (以毫秒为单位的 Unix 时间戳)
createdAt	Long	玩家被注册到免费游戏的日期 (以毫秒为单位的 Unix 时间戳)
createdBy	String	注册玩家的 API 或 BackOffice 用户
updatedAt	String	注册或重新注册玩家的 API 或 BackOffice 用户

例子:

```
{
  "data": {
    "totalCount": 2,
    "totalPage": 1,
    "result": [
      {
        "pendingWalletId": 140392,
        "playerName": "eddywang9999",
        "status": 1,
        "createdDate": 1554351552000,
        "updatedAt": 1554351552000,
        "createdBy": "External API",
        "updatedBy": "External API"
      },
      {
        "pendingWalletId": 140393,
        "playerName": "eddywang99999",
        "status": 1,
        "createdDate": 1554351682000,
        "updatedAt": 1554351682000,
        "createdBy": "External API",
        "updatedBy": "External API"
      }
    ]
  },
  "error": null
}
```

2.2.6 获取免费游戏的所有玩家详情

获取免费游戏的所有玩家的详细信息：

请求参数

API URL : {PgSoftAPIDomain}/FreeGame/v1/GetFreeGameAllPlayers

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
page_number	Integer	是	返回数据的批号
row_count	Integer	是	个别批次的记录数量 注：值度 1-5000
free_game_id	Integer	是	免费游戏的唯一标识符
from_time	Long	是	免费游戏的开始时间范围 (以毫秒为单位的 Unix 时间戳)
to_time	Long	是	免费游戏的结束时间范围 (以毫秒为单位的 Unix 时间戳)
player_name	String	否	免费游戏未注册玩家的唯一标识符 注：实行模糊匹配
from_created_time	Long	否	免费游戏的创建时间范围 (以毫秒为单位的 Unix 时间戳)
to_created_time	Long	否	
free_game_name	String	否	免费游戏的名称
status	Integer	否	免费游戏钱包的状态 0: 无效 (已取消)

			1: 活跃 2: 已过期 3: 已转换 4: 已完成 5: 全新 6: 已取消用户 8: 待处理
--	--	--	--

例子:

URL:

https://api.pg-bo.me/external/FreeGame/v1/GetFreeGameAllPlayers?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&page_number=1&row_count=500&player_name=player1&from_time=1346592723000&to_time=1546592723000&from_created_time=1346592723000&to_created_time=1546592723000&free_game_id=12345&free_game_name=freegame2019&status=1

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
totalCount	Integer	记录总数
totalPage	Integer	可用批次的总数
result	JSON Array	免费游戏未注册玩家列表

JSON 数组

参数名称	数据类型	描述
balanceld	Integer	免费游戏玩家钱包的唯一标识符
playerId	Integer	免费游戏玩家的唯一标识符
playerName	String	免费游戏玩家的唯一标识符
walletType	String	玩家的钱包类型 G: 免费游戏 B: 红利

balanceAmount	Decimal	未转换免费游戏的玩家余额，或免费游戏的已转换金额
bonusRatioAmount	Decimal	红利游戏的流水要求 (指免费游戏里的所赢金额在进入红利钱包之后，需要完成的总金额)
bonusMinimumConversionAmount	Decimal	免费游戏红利的最低转换金额
bonusMaximumConversionAmount	Decimal	免费游戏红利的最高转换金额
freeGameMaximumConversionAmount	Decimal	免费游戏的最高转换金额
status	Integer	免费游戏的钱包状态 0: 无效 (已取消) 1: 活跃 2: 已过期 3: 已转换 4: 已完成 5: 全新 6: 已取消用户 8: 待处理
expiredDate	Long	免费游戏的截止日期 (以毫秒为单位的 Unix 时间戳)
updatedAt	Long	玩家被注册或重新注册到免费游戏的日期 (以毫秒为单位的 Unix 时间戳)
createdDate	Long	玩家被注册到免费游戏的日期 (以毫秒为单位的 Unix 时间戳)

例子:

```
{
  "data": {
    "totalCount": 2583,
    "totalPage": 259,
    "result": [
      {
        "balanceId": 111261,
        "playerId": 7443,
        "playerName": "player1",
        "walletType": "B",
        "balanceAmount": 874.4,
        "bonusRatioAmount": 874.4,
        "bonusMinimumConversionAmount": 0,
        "bonusMaximumConversionAmount": 0,
        "freeGameMaximumConversionAmount": 0,
        "expiredDate": 1546991999000,
        "status": 2,
        "createdDate": 1546923615000,
        "updatedAt": 1546991999000
      },
      {
        "balanceId": 111252,
        "playerId": 7443,
        "playerName": "player2",
        "walletType": "G",
        "balanceAmount": 8.8,
        "bonusMinimumConversionAmount": 0,
        "bonusMaximumConversionAmount": 0,
        "freeGameMaximumConversionAmount": 0,
        "freeGameCount": 6,
        "expiredDate": 1546991999000,
        "status": 6,
        "createdDate": 1546918715000,
        "updatedAt": 1547028195000,
      }
    ]
  },
  "error": null
}
```

2.2.7 获取免费游戏的单个玩家详情

获取免费游戏里单个玩家的详细信息：

请求参数

API URL : {PgSoftAPIDomain}/FreeGame/v1/GetPlayerFreeGames

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
page_number	Integer	是	返回数据的批号
row_count	Integer	是	个别批次的记录数量 注：值度 1-5000
player_name	String	是	免费游戏未注册玩家的唯一标识符
status	Integer Array	否	免费游戏钱包的状态 0: 无效 (已取消) 1: 活跃 2: 已过期 3: 已转换 4: 已完成 5: 全新 6: 已取消用户 8: 待处理

例子:

URL:

https://api.pg-bo.me/external/FreeGame/v1/GetPlayerFreeGames?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&page_number=1&row_count=500&player_name=player1&status=1&status=2

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
totalCount	Integer	记录总数
totalPage	Integer	可用批次的总数
result	JSON Array	免费游戏未注册玩家列表

JSON 数组

参数名称	数据类型	描述
freeGameId	Integer	免费游戏的唯一标识符
freeGameName	String	免费游戏的名称
transactionId	String	交易的唯一标识符
gameIds	Integer Array	游戏的唯一标识符
gameCount	Integer	给予的免费游戏总数
totalGame	Integer	每位玩家将获得的免费游戏总数
balanceAmount	Decimal	未转换免费游戏的玩家余额, 或已转换免费游戏的已转换金额
minimumConversionAmount	Decimal	免费游戏的最低转换金额
maximumConversionAmount	Decimal	免费游戏的最高转换金额
multiplier	Integer	免费游戏的投注倍数

coinSize	Decimal	免费游戏的投注大小
expiredDate	Long	免费游戏的截止日期 (以毫秒为单位的 Unix 时间戳)
createdDate	Long	免费游戏的创建时间 (以毫秒为单位的 Unix 时间戳)
createdBy	String	创建免费游戏的 API 或 BackOffice 用户
updatedBy	String	更新免费游戏的 API 或 BackOffice 用户
isSupressDiscard	Boolean	允许玩家取消优惠的权限 True: 不允许玩家取消优惠 False: 允许玩家取消优惠
status	Integer	免费游戏钱包的状态 0: 无效 (已取消) 1: 活跃 2: 已过期 3: 已转换 4: 已完成 5: 全新 6: 已取消用户 8: 待处理
conversionType	Char	完成免费游戏后, 红利的转换类型 C: 现金 B: 红利

例子:

```
{
  "data": {
    "totalCount": 2583,
    "totalPage": 259,
    "result": [
      {
        "freeGameId": 909,
        "freeGameName": "Free Game 1",
        "gameIds": [25],
        "gameCount": 10,
        "totalGame": 5,
        "balanceAmount": 0,
        "transactionId": "12345Test",
        "minimumConversionAmount": 1,
        "maximumConversionAmount": 500,
        "multiplier": 10,
        "coinSize": 10,
        "expiredDate": 1552089599000,
        "status": 2,
        "conversionType": "B",
        "createdDate": 1498036782000,
        "createdBy": "player1",
        "updatedBy": "player1",
        "isSupressDiscard": true
      },
      {
        "freeGameId": 864,
        "freeGameName": "Free Game 2",
        "gameIds": [6],
        "gameCount": 55,
        "totalGame": 50,
        "balanceAmount": 0,
        "transactionId": "12345Test",
        "minimumConversionAmount": 1,
        "maximumConversionAmount": 500,
        "multiplier": 1,
        "coinSize": 1,
        "expiredDate": 1556672461000,
        "status": 5,
        "conversionType": "C",
        "createdDate": 1498036782000,
        "createdBy": "player1",
        "updatedBy": "player1",
        "isSupressDiscard": true
      }
    ]
  },
  "error": null
}
```

2.2.8 创建免费游戏

创建免费游戏：

注意

- 建议每个请求不超过 200 名玩家。

请求参数

API URL : {PgSoftAPIDomain}/FreeGame/v1/CreateFreeGame

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
currency	String	是	免费游戏的货币
free_game_name	String	是	免费游戏的名称
expired_date	long	是	免费游戏的截止日期 (以毫秒为单位的 Unix 时间戳)
bonus_minimum_conversion_amount	Decimal	否*	红利的最低转换金额 空值：无转换金额限制
bonus_maximum_conversion_amount	Decimal	否*	红利的最高转换金额 空值：无转换金额限制
minimum_conversion_amount	Decimal	否*	免费游戏的最低转换金额 空值：无转换金额限制

maximum_conversion_amount	Decimal	否	免费游戏的最高转换金额 空值：无转换金额限制
bonus_type	Integer	否*	红利的下注条件 0: 红利 1: 现金 2: 红利和现金
bonus_ratio_amount ^{t**}	Decimal	否*	红利游戏的流水要求 (指免费游戏里的所赢金额在进入红利钱包之后, 需要完成的总金额)
bonus_ratio ^{**}	Decimal	否*	红利游戏的流水要求 (指免费游戏里的所赢金额在进入红利钱包之后, 需要完成的免费游戏投注额倍数) 投注额 = 红利比率 x 已转换的红利金额
conversion_type	Char	是	完成免费游戏后的转换类型 C: 现金 B: 红利
multiplier	Integer	是	免费游戏的投注倍数
coin_size	Decimal	是	免费游戏的投注大小
game_count	Integer	是	每位玩家将获得的免费游戏总数
game_ids	Integer Array	是	游戏的唯一标识符 (注: 只接受有着相同基本投注的游戏)
is_event	Boolean	是	免费游戏类型: True: 弹性 False: 固定
player_name	String	否	玩家的唯一标识符。系统将创建免费游戏并直接分配给该玩家。

			注：若已提供 <code>player_name</code>，免费游戏类型将被更改为弹性模式
<code>transaction_id</code>	String	是	交易的唯一标识符 注：对于重复的请求，系统将返回对应的免费游戏 ID
<code>non_discardable</code>	Boolean	否	不允许玩家放弃优惠： True：不允许玩家放弃优惠 False：允许玩家放弃优惠

*仅适用于红利转换类型。

** 若有提供 `bonus_ratio_amount`，`bonus_ratio` 的流水条件将被覆盖。

例子：

URL：

https://api.pg-bo.me/external/FreeGame/v1/CreateFreeGame?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体：

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl¤cy=CNY&free_game_name=FreeGame2019&expired_date=1745289243000&status=1&bonus_maximum_conversion_amount=10&maximum_conversion_amount=1&bonus_type=1&bonus_ratio_amount=2&conversion_type=B&multiplier=5&coin_size=20&game_count=20&game_ids=1&game_ids=2&is_event=true&player_name=player1&transaction_id=12345&non_discardable=False

返回结果

参数：

JSON 对象

参数名称	数据类型	描述
<code>freeGameId</code>	Integer	免费游戏的唯一标识符

例子：

```
{
  "data": {
    "freeGameId ": 10915
  },
  "error": null
}
```


2.2.9 按投注金额创建免费游戏

按照需要的投注金额创建免费游戏：

注意

- 建议每个请求不超过 200 名玩家。

请求参数

API URL : {PgSoftAPIDomain}/FreeGame/v1/CreateFreeGameByBetAmount

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符（GUID） 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
currency	String	是	免费游戏的货币
free_game_name	String	是	免费游戏的名称
expired_date	long	是	免费游戏的截止日期（以毫秒为单位的 Unix 时间戳）
bonus_minimum_conversion_amount	Decimal	否*	红利的最低转换金额 空值：无转换金额限制
bonus_maximum_conversion_amount	Decimal	否*	红利的最高转换额 空值：无转换金额限制
minimum_conversion_amount	Decimal	否	免费游戏的最低转换金额 空值：无转换金额限制

maximum_conversion_amount	Decimal	否	免费游戏的最高转换金额 空值：无转换金额限制
bonus_type	Integer	否*	红利的下注条件 0: 红利 1: 现金 2: 红利和现金
bonus_ratio_amount*	Decimal	否*	红利游戏的流水要求 (指免费游戏里的所赢金额在进入红利钱包之后, 需要完成的总金额)
bonus_ratio**	Decimal	否*	红利游戏的流水要求 (指免费游戏里的所赢金额在进入红利钱包之后, 需要完成的免费游戏投注额倍数) 投注额 = 红利比率 x 已转换的红利金额
conversion_type	Char	是	完成免费游戏后的红利转换类型 C: 现金 B: 红利
bet_amount	Decimal	是	每个游戏所需的投注金额。 注：系统会自动取整至最接近的倍数和投注大小
game_count	Integer	是	每位玩家将获得的免费游戏总数
game_ids	Integer Array	是	游戏的唯一标识符 (注：只接受有着相同基本投注的游戏)
is_event	Boolean	是	免费游戏类型： True: 弹性 False: 固定

player_name	String	否	玩家的唯一标识符。系统将创建免费游戏并直接分配给该玩家。 注：若已提供 player_name，免费游戏类型将被更改为弹性模式
transaction_id	String	是	交易的唯一标识符。 注：对于重复的请求，系统将返回对应的免费游戏 ID
non_discardable	Boolean	否	不允许玩家放弃优惠： True：不允许玩家放弃优惠 False：允许玩家放弃优惠

*仅适用于红利转换类型。

**若有提供 *bonus_ratio_amount*, *bonus_ratio* 的流水条件将被覆盖。

例子：

URL：

https://api.pg-bo.me/external/FreeGame/v1/CreateFreeGameByBetAmount?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体：

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl¤cy=CNY&free_game_name=FreeGame2019&expired_date=1745289243000&status=1&bonus_maximum_conversion_amount=10&maximum_conversion_amount=1&bonus_type=1&bonus_ratio_amount=2&conversion_type=B&bet_amount=0.6&game_count=20&game_ids=1&game_ids=2&is_event=true&player_name=player1&transaction_id=12345&non_discardable=False

返回结果

参数：

JSON 对象

参数名称	数据类型	描述
freeGameId	Integer	免费游戏的唯一标识符

例子：

```
{
  "data": {
    "freeGameId ": 10915
  },
  "error": null
}
```

2.2.10 将玩家转移到固定免费游戏中

添加玩家到固定免费游戏中：

注意

- 建议每个请求不超过 200 名玩家。

请求参数

API URL : {PgSoftAPIDomain}/FreeGame/v1/TransferInFreeGame

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
free_game_id	Integer	是	免费游戏的唯一标识符
player_names	String Array	是	玩家的唯一标识符
transfer_reference	String	是	转移的唯一标识符

例子：

URL:

https://api.pg-bo.me/external/FreeGame/v1/TransferInFreeGame?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&free_game_id=12&player_names=player1&player_names=player2&transfer_reference=freegame_20181128

返回结果

参数:

JSON 数组

参数名称	数据类型	描述
transactionId	String	交易的唯一标识符
operatorToken	String	运营商的唯一标识符
playerName	String	玩家的唯一标识符
currencyCode	String	玩家的货币
status	Integer	转移请求的状态 0: 失败 1: 成功 2: 待定（对于尚未创建的玩家，免费游戏将在创建玩家后自动转移给该玩家）

例子:

```
{
  "data": [
    {
      "transactionId": 10,
      "operatorToken": "abc123",
      "playerName": "player1",
      "currencyCode": "CNY",
      "status": 1
    },
    {
      "transactionId": 1,
      "operatorToken": "1234",
      "playerName": "player2",
      "currencyCode": "CNY",
      "status": 1
    }
  ],
  "error": null
}
```

2.2.11 将玩家转移到弹性免费游戏中

添加玩家到弹性免费游戏中。运营商可以覆盖特定玩家的免费游戏设置。

注意

- 建议每个请求不超过 200 名玩家。

请求参数

API URL : {PgSoftAPIDomain}/FreeGame/v1/TransferInFlexibleFreeGame

HTTP 方法 : POST

URL 参数:

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注: • 请把参数值设为 GUID 格式

主体参数:

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
player_free_games	JSON String Array	是	覆盖玩家的免费游戏默认信息
free_game_id	Integer	是	免费游戏的唯一标识符
transfer_reference	String	是	转移的唯一标识符
allow_multiple	Boolean	是	允许玩家重新注册 True: 玩家可以重新注册免费游戏的次数 False: 玩家只能在免费游戏中注册一次

player_free_games JSON 字符串:

参数名称	数据类型	必需项	描述
player_name	String	是	运营商的唯一标识符
free_game_count	Integer	是	免费游戏的计数将覆盖玩家的默认计数 值度: 值 ≥ 1
is_unlimited_bonus_maximum_conversion_amount	Bool	是	将红利的最高转换金额设置为无限制
is_unlimited_free_game_maximum_conversion_amount	Bool	是	将免费游戏的最高转换金额设置为无限制
bonus_minimum_conversion_amount	Decimal	否*	可为玩家覆盖的最低红利转换金额 值度: Value ≥ 1 默认值: 0 注: 如果值 = 0, 系统将使用默认转换金额
bonus_maximum_conversion_amount	Decimal	否*	可为玩家覆盖的最高红利转换金额 值度: 值 ≥ 1 默认值: 0 注: 如果值 = 1, 系统将使用默认转换金额
free_game_minimum_conversion_amount	Decimal	否*	可为玩家覆盖的最低免费游戏转换金额 值度: 值 ≥ 1 默认值: 0 注: 如果值 = 1, 系统将使用默认转换金额
free_game_maximum_conversion_amount	Decimal	否**	可为玩家覆盖的最高免费游戏转换金额 值度: 值 ≥ 1 默认值: 0 注: 如果值 = 1, 系统将使用默认转换金额
说明	String	否	转移参考说明

*仅适用于 *is_unlimited_bonus_maximum_conversion_amount* 参数值为错误的情况。

**仅适用于 *is_unlimited_free_game_maximum_conversion_amount* 参数值为错误的情况。

例子:

URL:

https://api.pg-bo.me/external/FreeGame/v1/TransferInFlexibleFreeGame?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

```
operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&player_free_games={"player_name":"player1","free_game_count":10,"is_unlimited_bonus_maximum_conversion_amount":false,"is_unlimited_free_game_maximum_conversion_amount":false,"bonus_maximum_conversion_amount":100.50,"free_game_maximum_conversion_amount":100.50,"description":"VIP"}&free_game_id=3&transfer_reference=freegame_20181128&allow_multiple=false
```

返回结果

参数:

JSON 数组

参数名称	数据类型	描述
transactionId	String	交易的唯一标识符
operatorToken	String	运营商的唯一标识符
playerName	String	玩家的唯一标识符
currencyCode	String	玩家的货币
freeGameCount	Integer	玩家的免费游戏数量
status	Integer	转移请求的状态 0: 失败 1: 成功 2: 待定 (对于尚未创建的玩家, 免费游戏将在创建玩家后自动转移给该玩家)

例子:

```
{
  "data": [
    {
      "transactionId": 10,
      "operatorToken": "abc123",
      "playerName": "player1",
      "currencyCode": "CNY",
      "freeGameCount": 10,
      "status": 1
    },
    {
      "transactionId": 1,
      "operatorToken": "1234",
      "playerName": "player2",
      "currencyCode": "CNY",
      "freeGameCount": 10,
      "status": 1
    }
  ],
  "error": null
}
```

2.2.12 取消免费游戏

取消免费游戏：

请求参数

API URL : {PgSoftAPIDomain}/FreeGame/v1/CancelFreeGame

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： <ul style="list-style-type: none"> 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
free_game_id	String	是	免费游戏的唯一标识符

例子：

URL：

https://api.pg-bo.me/external/FreeGame/v1/CancelFreeGame?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体：

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&free_game_id=10919

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
actionResult	Boolean	请求结果: False: 失败 True: 成功

例子:

```
{
  "data": {
    "actionResult": true
  },
  "error": null
}
```

2.2.13 取消玩家的免费游戏

取消特定玩家的免费游戏：

请求参数

API URL : {PgSoftAPIDomain}/FreeGame/v1/CancelPlayerFreeGame

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
player_name	String	是	玩家的唯一标识符
balance_id	Integer	是	玩家免费游戏钱包的唯一标识符

例子：

URL：

https://api.pg-bo.me/external/FreeGame/v1/CancelPlayerFreeGame?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体：

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&player_name=player1&balance_id=10919

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
actionResult	Boolean	请求结果: False: 失败 True: 成功

例子:

```
{
  "data": {
    "actionResult": true
  },
  "error": null
}
```

2.2.14 取消未注册玩家的免费游戏

取消特定未注册玩家的免费游戏：

请求参数

API URL : {PgSoftAPIDomain}/FreeGame/v1/CancelPendingPlayerFreeGame

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
pending_wallet_id	Integer	是	免费游戏玩家钱包的唯一标识符

例子：

URL:

https://api.pg-bo.me/external/FreeGame/v1/CancelPendingPlayerFreeGame?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&pending_wallet_id=10919

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
actionResult	Boolean	请求结果: False: 失败 True: 成功

例子:

```
{
  "data": {
    "actionResult": true
  },
  "error": null
}
```

2.2.15 取消玩家的所有免费游戏

取消已注册/未注册玩家的所有免费游戏：

请求参数

API URL : {PgSoftAPIDomain}/FreeGame/v1/CancelPlayerAllFreeGame

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
player_name	String	是	玩家的唯一标识符
free_game_id	Integer	是	免费游戏的唯一标识符

例子：

URL：

https://api.pg-bo.me/external/FreeGame/v1/CancelPlayerAllFreeGame?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体：

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&player_name=player1&free_game_id=10919

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
actionResult	Boolean	请求结果: True: 成功

例子:

```
{
  "data": {
    "actionResult": true
  },
  "error": null
}
```

2.2.16 获取免费游戏的玩家汇总

获取免费游戏的玩家汇总：

请求参数

API URL : {DataGrabAPIDomain}/FreeGame/v1/GetFreeGamePlayerSummary

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： <ul style="list-style-type: none"> 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
row_version	Long	否	数据更新时间 (以毫秒为单位的 Unix 时间戳)
player_name	String	否	玩家的唯一标识符
row_count	Integer	否	个别批次的记录数量 注： <ul style="list-style-type: none"> 默认值: 1 值度: 1-5000
currency	String	否	玩家的货币
free_game_id	Integer	否	免费游戏的唯一标识符

例子:

URL:

https://api.pg-bo.me/external-datagrabber/FreeGame/v1/GetFreeGamePlayerSummary?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&row_version=1346592723000&player_name=testplayer1&row_count=10¤cy=CNY&free_game_id=12345

返回结果

主体参数:

JSON 对象

参数名称	数据类型	描述
lastRowVersion	Long	数据更新时间 (以毫秒为单位的 Unix 时间戳)
result	JSON Array	结果列表

JSON 数组

参数名称	数据类型	描述
freeGameId	Ulong	免费游戏 ID 的唯一标识符
freeGameParentId	Ulong	免费游戏母 ID 的唯一标识符
playerName	String	玩家的唯一标识符
currencyCode	String	玩家的货币
totalFreeGame	Integer	给予的免费游戏总数
coinSize	Decimal	免费游戏的投注大小
multiplier	Integer	免费游戏的投注倍数
payout	Decimal	从免费游戏转换为现金的金额
remainCount	Integer	剩余的免费游戏总数
conversionType	Char	完成免费游戏后的转换类型: C: 现金 B: 红利
status	Integer	免费游戏钱包的状态 0: 无效 (已取消) 1: 活跃

		2: 已过期 3: 已转换 4: 已完成 5: 全新 6: 已取消用户 8: 待处理
createdTime	Long	免费游戏的创建时间
updatedAt	Long	免费游戏的更新时间

例子:

```
{
  "data": {
    "lastRowVersion": 1526371744000,
    "result": [
      {
        "freeGameId": 2,
        "freeGameParentId": 2,
        "playerName": "qa1",
        "currencyCode": "CNY",
        "totalFreeGame": 1,
        "coinSize": 0.01,
        "multiplier": 10,
        "payout": 0.0,
        "remainCount": 1,
        "conversionType": "Bonus",
        "status": 0,
        "createdTime": 1526371189000,
        "updatedAt": 1526371744000
      }
    ]
  },
  "error": null
}
```

2.3 红利 [最新版本: v1]

2.3.1 获取红利游戏列表

获取已创建的红利游戏列表:

请求参数

API URL : {PgSoftAPIDomain}/Bonus/v1/GetBonus

HTTP 方法 : POST

URL 参数:

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注: • 请把参数值设为 GUID 格式

主体参数:

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
page_number	Integer	是	返回数据的批号
row_count	Integer	是	个别批次的记录数量 注: 值度 1-5000
from_time	Long	是	红利游戏的开始时间范围 (以毫秒为单位的 Unix 时间戳)
to_time	Long	是	
from_created_time	Long	否	红利游戏的创建时间范围 (以毫秒为单位的 Unix 时间戳)
to_created_time	Long	否	
bonus_id	Integer	否	红利游戏的唯一标识符
currency	String	否	红利游戏的货币
bonus_name	String	否	红利游戏的名称
bonus_transfer_type	Integer	否	红利游戏的转移类型 0: 固定 3: 弹性

status	Integer	否	红利游戏状态 0: 无效 1: 活跃 2: 已过期
--------	---------	---	------------------------------------

例子:

URL:

https://api.pg-bo.me/external/Bonus/v1/GetBonus?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&page_number=1&row_count=500&from_time=1346592723000&to_time=1546592723000&from_created_time=1346592723000&to_created_time=1546592723000&bonus_id=12345¤cy=CN¥&bonus_name=BonusGame2019&bonus_transfer_type=1&status=1

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
totalCount	Integer	记录总数
totalPage	Integer	可用批次的总数
result	JSON Array	多个红利游戏的详情列表

JSON 数组

参数名称	数据类型	描述
operatorToken	String	运营商的唯一标识符
bonusId	Integer	红利游戏的唯一标识符
bonusName	String	红利游戏的名称
isEvent	Boolean	红利游戏类型: True: 弹性 False: 固定
gameIds	Integer Array	游戏的唯一标识符
currencyCode	String	红利游戏货币

bonusRatio	Decimal	红利游戏的流水要求 指需要完成的投注额倍数
bonusRatioAmount	Decimal	红利游戏的流水要求 指需要完成的总金额
bonusMinimumConversionAmount	Decimal	免费游戏转现金的最高转换金额
bonusMaximumConversionAmount	Decimal	免费游戏转现金的最低转换金额
bonusType	Integer	红利游戏流水条件 0: 红利 1: 现金 2: 红利和现金
status	Integer	红利游戏的状态 0: 无效（已取消） 1: 活跃 2: 已过期
bonusAmountGiven	Decimal	已送出的红利总额
numberOfPlayers	Integer	红利游戏中的玩家总数
expiredDate	Long	红利游戏的截止日期 (以毫秒为单位的 Unix 时间戳)
createdDate	Long	红利游戏的创建日期 (以毫秒为单位的 Unix 时间戳)
isSuppressDiscard	Boolean	允许玩家取消优惠的权限 True: 不允许玩家取消优惠 False: 允许玩家取消优惠

例子:

```
{
  "data": {
    "totalCount": 2583,
    "totalPage": 259,
    "result": [
      {
        "operatorToken": "abcd",
        "bonusId": 118,
        "bonusName": "bonus118",
        "bonusType": 0,
        "isEvent": false,
        "gameIds": [
          1
        ],
        "amount": 57000,
        "currencyCode": "MYR",
        "bonusRatio": 3,
        "bonusRatioAmount": 171000,
        "bonusMinimumConversionAmount": 1000.00,
        "bonusMaximumConversionAmount": 1500.00,
        "status": 1,
        "numberOfPlayers": 0,
        "bonusAmountGiven": 0,
        "createdDate": 1497001371000,
        "isSuppressDiscard": false
      },
      {
        "operatorToken": "abcd",
        "bonusId": 119,
        "bonusName": "bonus119",
        "bonusType": 0,
        "isEvent": false,
        "gameIds": [
          1
        ],
        "amount": 500,
        "currencyCode": "USD",
        "bonusRatio": 2,
        "bonusRatioAmount": 1000,
        "status": 1,
        "numberOfPlayers": 1,
        "bonusAmountGiven": 0,
        "createdDate": 1497239059000
      }
    ]
  },
  "error": null
}
```


2.3.2 获取已转换为现金的红利游戏列表

获取已转换为现金的红利游戏列表:

请求参数

API URL : {PgSoftAPIDomain}/Bonus/v1/GetBonusConvertedWallet

HTTP 方法 : POST

URL 参数:

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注: • 请把参数值设为 GUID 格式

主体参数:

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
page_number	Integer	是	返回数据的批号
row_count	Integer	是	个别批次的记录数量 注: 值度 1-5000
bonus_id	Integer	是	红利的唯一标识符
from_transaction_time	Long	是	红利玩家交易的时间范围 (以毫秒为单位的 Unix 时间戳)
to_transaction_time	Long	是	
player_name	String	否	玩家的唯一标识符 注: 实行模糊匹配

例子:

URL:

https://api.pg-bo.me/external/Bonus/v1/GetBonusConvertedWallet?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&page_number=1&row_count=500&bonus_id=12345&player_name=player1&from_transaction_time=1346592723000&to_transaction_time=1346592723000

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
totalConvertedAmount	Decimal	已通过红利游戏转换成现金的总金额
totalCount	Integer	记录总数
totalPage	Integer	可用批次的总数
result	JSON Array	红利游戏已转换为现金的玩家的详细信息列表

JSON 数组

参数名称	数据类型	描述
balanceId	Integer	玩家红利游戏钱包的唯一标识符
playerName	String	玩家的唯一标识符
currencyCode	String	玩家的货币
gameId	Integer	游戏的唯一标识符
transactionAmount	Decimal	玩家的转换金额
transactionDateTime	Long	转换时间 (以毫秒为单位的 Unix 时间戳)

例子:

```
{
  "data": {
    "totalConvertedAmount": 10,
    "totalCount": 1,
    "totalPage": 1,
    "result": [
      {
        "balanceId": 13787,
        "playerName": "qa18",
        "currencyCode": "CNY",
        "gameId": 8,
        "transactionAmount": 10,
        "transactionDateTime": 1547449915000
      },
      {
        "balanceId": 13788,
        "playerName": "qa18",
        "currencyCode": "CNY",
        "gameId": 8,
        "transactionAmount": 10,
        "transactionDateTime": 1547449915100
      }
    ],
    "error": null
  }
}
```

2.3.3 获取红利游戏的玩家列表

获取红利游戏的已注册玩家列表：

请求参数

API URL : {PgSoftAPIDomain}/Bonus/v1/GetBonusPlayers

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
page_number	Integer	是	返回数据的批号
row_count	Integer	是	个别批次的记录数量 注：值度 1-5000
bonus_id	Integer	是	红利的唯一标识符
from_time	Long	是	红利游戏的开始时间范围 (以毫秒为单位的 Unix 时间戳)
to_time	Long	是	
player_name	String	否	红利玩家的唯一标识符 注：实行模糊匹配
from_created_time	Long	否	玩家被注册到红利游戏的时间范围 (以毫秒为单位的 Unix 时间戳)
to_created_time	Long	否	
status	Integer	否	红利钱包的状态 0: 无效 (已取消) 1: 活跃 2: 已过期 3: 已转换

			4: 已完成
			5: 全新
			6: 已取消用户
			8: 待处理

例子:

URL:

https://api.pg-bo.me/external/Bonus/v1/GetBonusPlayers?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&page_number=1&row_count=500&player_name=player1&from_time=1346592723000&to_time=1546592723000&from_created_time=1346592723000&to_created_time=1546592723000&bonus_id=12345&status=1

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
totalCount	Integer	记录总数
totalPage	Integer	可用批次的总数
result	JSON Array	红利游戏的玩家列表

JSON 数组

参数名称	数据类型	描述
balanceId	Integer	玩家红利钱包的唯一标识符
playerId	Integer	红利游戏玩家的唯一标识符
playerName	String	红利游戏玩家的唯一标识符
walletType	String	红利游戏玩家的钱包类型 G: 免费游戏 B: 红利
balanceAmount	Decimal	玩家的红利余额
bonusRatioAmount	Decimal	红利游戏的流水要求 指需要完成的总金额
MinimumConversionAmount	Decimal	红利的最低转换金额

MaximumConversionAmount	Decimal	红利的最高转换金额
status	Integer	红利钱包的状态 0: 无效 (已取消) 1: 活跃 2: 已过期 3: 已转换 4: 已完成 5: 全新 6: 已取消用户 8: 待处理
expiredDate	Long	红利游戏的截止日期 (以毫秒为单位的 Unix 时间戳)
updatedAt	Long	玩家被注册或重新注册到红利游戏的日期 (以毫秒为单位的 Unix 时间戳)
createdAt	Long	

例子:

```
{
  "data": {
    "totalCount": 2583,
    "totalPage": 259,
    "result": [
      {
        "balanceId": 111261,
        "playerId": 7443,
        "playerName": "player1",
        "walletType": "B",
        "balanceAmount": 874.4,
        "bonusRatioAmount": 874.4,
        "MinimumConversionAmount": 0,
        "MaximumConversionAmount": 0,
        "expiredDate": 1546991999000,
        "status": 2,
        "createdDate": 1546923615000,
        "updatedAt": 1546991999000
      },
      {
        "balanceId": 111252,
        "playerId": 7443,
        "playerName": "player2",
        "walletType": "G",
        "balanceAmount": 8.8,
        "MinimumConversionAmount": 0,
        "MaximumConversionAmount": 0,
        "expiredDate": 1546991999000,
        "status": 6,
        "createdDate": 1546918715000,
        "updatedAt": 1547028195000
      }
    ]
  },
  "error": null
}
```

2.3.4 获取红利游戏的未注册玩家列表

获取红利游戏的未注册玩家列表：

请求参数

API URL : {PgSoftAPIDomain}/Bonus/v1/GetBonusPendingPlayers

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
page_number	Integer	是	返回数据的批号
row_count	Integer	是	个别批次的记录数量 注：值度 1-5000
bonus_id	Integer	是	红利游戏的唯一标识符
player_name	String	否	未注册红利游戏玩家的唯一标识符 注：实行模糊匹配
created_by	String	否	注册玩家的 API 或 BackOffice 用户
status	Integer	否	未注册玩家的钱包状态： 0: 无效 (已取消) 1: 待处理 2: 已转移

例子:

URL:

https://api.pg-bo.me/external/Bonus/v1/GetBonusPendingPlayers?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&page_number=1&row_count=500&player_name=player1&bonus_id=12345&created_by=external&status=1

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
totalCount	Integer	记录总数
totalPage	Integer	可用批次的总数
result	JSON Array	未注册的红利游戏玩家的列表

JSON 数组

参数名称	数据类型	描述
pendingWalletId	Integer	未注册玩家的红利钱包的唯一标识符
playerName	String	未注册的红利游戏玩家的唯一标识符
status	Integer	未注册玩家的钱包状态 0: 无效 (已取消) 1: 待处理 2: 已转换
updateDate	Long	玩家被注册或重新注册到红利游戏的日期 (以毫秒为单位的 Unix 时间戳)
createDate	Long	
createBy	String	注册玩家的 API 或 BackOffice 用户
updateBy	String	

例子:

```
{
  "data": {
    "totalCount": 2,
    "totalPage": 1,
    "result": [
      {
        "pendingWalletId": 140392,
        "playerName": "eddywang9999",
        "status": 1,
        "createdDate": 1554351552000,
        "updatedAt": 1554351552000,
        "createdBy": "External API"
      },
      {
        "pendingWalletId": 140393,
        "playerName": "eddywang99999",
        "status": 1,
        "createdDate": 1554351682000,
        "updatedAt": 1554351682000,
        "createdBy": "External API"
      }
    ]
  },
  "error": null
}
```

2.3.5 获取红利游戏的所有玩家详情

获取红利游戏的所有玩家的详细信息：

请求参数

API URL : {PgSoftAPIDomain}/Bonus/v1/GetBonusAllPlayers

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符（GUID） 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
page_number	Integer	是	返回数据的批号
row_count	Integer	是	个别批次的记录数量 注：值度 1-5000
bonus_id	Integer	是	红利游戏的唯一标识符
from_time	Long	是	红利游戏的开始时间范围 (以毫秒为单位的 Unix 时间戳)
to_time	Long	是	
player_name	String	否	红利游戏玩家的唯一标识符 注：实行模糊匹配
from_created_time	Long	否	红利游戏的创建时间范围 (以毫秒为单位的 Unix 时间戳)
to_created_time	Long	否	
status	Integer	否	红利钱包的状态 0: 无效（已取消） 1: 活跃 2: 已过期 3: 已转换

			4: 已完成
			5: 全新
			6: 已取消用户
			8: 待处理

例子:

URL:

https://api.pg-bo.me/external/Bonus/v1/GetBonusAllPlayers?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&page_number=1&row_count=500&player_name=player1&from_time=1346592723000&to_time=1546592723000&from_created_time=1346592723000&to_created_time=1546592723000&bonus_id=12345&status=1

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
totalCount	Integer	记录总数
totalPage	Integer	可用批次的总数
result	JSON Array	所有红利游戏玩家的列表

JSON 数组

参数名称	数据类型	描述
balanceId	Integer	玩家红利钱包的唯一标识符
playerId	Integer	红利游戏玩家的唯一标识符
playerName	String	红利游戏玩家的唯一标识符
walletType	String	玩家的钱包类型 G: 免费游戏 B: 红利
balanceAmount	Decimal	未转换红利游戏的玩家余额, 或红利游戏的已转换金额
bonusRatioAmount	Decimal	红利游戏的流水要求 指需要完成的总金额

MinimumConversionAmount	Decimal	红利游戏的最低转换金额
MaximumConversionAmount	Decimal	红利游戏的最高转换金额
status	Integer	红利钱包的状态 0: 无效 (已取消) 1: 活跃 2: 已过期 3: 已转换 4: 已完成 5: 全新 6: 已取消用户 8: 待处理
expiredDate	Long	红利游戏的截止日期 (以毫秒为单位的 Unix 时间戳)
updatedAt	Long	玩家被注册或重新注册到红利游戏的日期 (以毫秒为单位的 Unix 时间戳)
createdAt	Long	

例子:

```
{
  "data": {
    "totalCount": 2583,
    "totalPage": 259,
    "result": [
      {
        "balanceId": 111261,
        "playerId": 7443,
        "playerName": "player1",
        "walletType": "B",
        "balanceAmount": 874.4,
        "bonusRatioAmount": 874.4,
        "MinimumConversionAmount": 0,
        "MaximumConversionAmount": 0,
        "expiredDate": 1546991999000,
        "status": 2,
        "createdDate": 1546923615000,
        "updatedAt": 1546991999000
      },
      {
        "balanceId": 111252,
        "playerId": 7443,
        "playerName": "player2",
        "walletType": "G",
        "balanceAmount": 8.8,
        "MinimumConversionAmount": 0,
        "MaximumConversionAmount": 0,
        "expiredDate": 1546991999000,
        "status": 6,
        "createdDate": 1546918715000,
        "updatedAt": 1547028195000
      }
    ]
  },
  "error": null
}
```

2.3.6 获取红利游戏的单个玩家详情

获取红利游戏里单个玩家的详细信息：

请求参数

API URL : {PgSoftAPIDomain}/Bonus/v1/GetPlayerBonus

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符（GUID） 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
page_number	Integer	是	返回数据的批号
row_count	Integer	是	个别批次的记录数量 注：值度 1-5000
player_name	String	是	红利玩家的唯一标识符
status	Integer Array	否	红利钱包的状态 0: 无效（已取消） 1: 活跃 2: 已过期 3: 已转换 4: 已完成 5: 全新 6: 已取消用户 8: 待处理

例子:

URL:

https://api.pg-bo.me/external/Bonus/v1/GetPlayerBonus?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&page_number=1&row_count=500&player_name=player1&status=1&status=1&status=2

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
totalCount	Integer	记录总数
totalPage	Integer	可用批次的总数
result	JSON Array	单个玩家的红利游戏列表

JSON 数组

参数名称	数据类型	描述
bonusId	Ulong	红利游戏的唯一标识符 注: 对于从免费游戏转换而来的红利, 系统将返还免费游戏 ID
bonusName	String	红利游戏的名称
gameIds	Integer Array	游戏的唯一标识符
balanceAmount	Decimal	未转换红利游戏的玩家余额, 或红利游戏的已转换金额
maximumConversionAmount	Decimal	红利的最高转换金额
bonusRatioAmount	Decimal	红利游戏的流水要求 指需要完成的总金额
bonusParentType	Integer	母红利类型: B: 正常创建的红利游戏 G: 从免费游戏转换而来的红利游戏

status	Integer	红利钱包的状态 0: 无效（已取消） 1: 活跃 2: 已过期 3: 已转换 4: 已完成 5: 全新 6: 已取消用户 8: 待处理
createdDate	Long	创建红利游戏的日期 (以毫秒为单位的 Unix 时间戳)
expiredDate	Long	红利游戏的截止日期 (以毫秒为单位的 Unix 时间戳)
createdBy	String	创建或更新红利游戏的 API 或 BackOffice 用户
updatedBy	String	

例子:

```
{
  "data": {
    "totalCount": 2583,
    "totalPage": 259,
    "result": [
      {
        "bonusId": 11385,
        "bonusName": "Bonus 1",
        "bonusParentType": "B",
        "gameIds": [
          9
        ],
        "balanceAmount": 1000000,
        "bonusRatioAmount": 1000,
        "maximumConversionAmount": 5000,
        "status": 2,
        "createdDate": 1531981770000,
        "expiredDate": 1532015999000,
        "createdBy": "Operator1",
        "updatedBy": "Operator1"
      },
      {
        "bonusId": 12271,
        "bonusName": "Bonus 2",
        "bonusParentType": "G",
        "gameIds": [
          27
        ],
        "balanceAmount": 1000000,
        "bonusRatioAmount": 10000,
        "maximumConversionAmount": 500,
        "status": 2,
        "createdDate": 1537413312000,
        "expiredDate": 1537487999000,
        "createdBy": "Operator1",
        "updatedBy": "Operator1"
      }
    ]
  },
  "error": null
}
```

2.3.7 创建红利游戏

创建新的红利游戏：

注意

- 建议每个请求不超过 200 名玩家。

请求参数

API URL : {PgSoftAPIDomain}/Bonus/v1/CreateBonus

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
currency	String	是	红利游戏货币
bonus_name	String	是	红利游戏的名称
bonus_amount	decimal	是	给予每个玩家的红利金额
expired_date	long	是	红利游戏的截止日期 (以毫秒为单位的 Unix 时间戳)
minimum_conversion_amount	Decimal	否	红利游戏的最低转换金额 空值：无转换金额限制
maximum_conversion_amount	Decimal	否	红利游戏的最高转换金额 空值：无转换金额限制
bonus_type	Integer	是	红利游戏的流水条件

			0: 红利 1: 现金 2: 红利和现金
bonus_ratio_amount* *	Decimal	否	红利游戏的流水要求 指需要完成的投注额倍数
bonus_ratio**	Decimal	是	红利游戏的流水要求 指需要完成的总金额 投注额 = 红利比率 x 已转换的红利金 额
game_ids	Integer Array	是	游戏的唯一标识符 注：只接受有着相同基本投注的游戏
is_event	Boolean	是	红利游戏类型： True: 弹性 False: 固定
player_name	String	否	玩家的唯一标识符。系统将创建并直接 分配红利游戏给该玩家。 注：若已提供 player_name，免费游 戏类型将被更改为弹性模式
transaction_id	String	是	交易的唯一标识符 注：对于重复的请求，系统将返回对应 的免费游戏 ID
non_discardable	Boolean	否	不允许玩家放弃优惠： True: 不允许玩家放弃优惠 False: 允许玩家放弃优惠

** 若有提供 *bonus_ratio_amount*, *bonus_ratio* 的流水条件将被覆盖。

例子：

URL:

https://api.pg-bo.me/external/Bonus/v1/CreateBonus?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

```
operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&currency=CNY&bonus_name=Bonus
2019&bonus_amount=100.00&expired_date=1745289243000&status=1&maximum_conversion_
ount=1&bonus_type=1&bonus_ratio_amount=2&game_ids=1&game_ids=2&is_event=true&playe
r_name=player1&transaction_id=12345&non_discardable=False
```

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
bonusId	Integer	红利游戏的唯一标识符

例子:

```
{
  "data": {
    "bonusId ": 10915
  },
  "error": null
}
```

2.3.8 将玩家转移到固定红利游戏中

将玩家转移到已创建的固定红利游戏中。运营商只能将玩家转移到预先设置的红利游戏中。

注意

- 建议每个请求不超过 200 名玩家。

请求参数

API URL : {PgSoftAPIDomain}/Bonus/v1/TransferInBonus

HTTP 方法 : POST

URL 参数:

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注: • 请把参数值设为 GUID 格式

主体参数:

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
bonus_id	Integer	是	红利游戏的唯一标识符
player_names	String Array	是	玩家的唯一标识符
transfer_reference	String	是	转移的唯一标识符

例子:

URL:

https://api.pg-bo.me/external/Bonus/v1/TransferInBonus?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&bonus_id=12&player_names=player1&player_names=player2&transfer_reference=bonusgame_20181128

返回结果

参数:

JSON 数组

参数名称	数据类型	描述
transactionId	String	交易的唯一标识符
operatorToken	String	运营商的唯一标识符
playerName	String	玩家的唯一标识符
currencyCode	String	玩家的货币
status	Integer	转移请求的状态 0: 失败 1: 成功 2: 待处理（对于尚未创建的玩家，红利游戏将在创建玩家后自动分配给该玩家）

例子:

```
{
  "data": [
    {
      "transactionId": 10,
      "operatorToken": "abc123",
      "playerName": "player1",
      "currencyCode": "CNY",
      "status": 1
    },
    {
      "transactionId": 1,
      "operatorToken": "1234",
      "playerName": "player2",
      "currencyCode": "CNY",
      "status": 1
    }
  ],
  "error": null
}
```

2.3.9 将玩家转移到弹性红利游戏中

将玩家转移到已创建的弹性红利游戏中。运营商可以覆盖特定玩家的红利游戏设置。

注意

- 建议每个请求不超过 200 名玩家。

请求参数

API URL : {PgSoftAPIDomain}/Bonus/v1/TransferInFlexibleBonus

HTTP 方法 : POST

URL 参数:

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注: • 请把参数值设为 GUID 格式

主体参数:

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
player_bonuses	JSON String Array	是	覆盖玩家的默认红利游戏信息
bonus_id	Integer	是	红利游戏的唯一标识符
transfer_reference	String	是	转移的唯一标识符
allow_multiple	Boolean	是	True: 玩家可以重新注册红利游戏的次数 False: 玩家只能在红利游戏中注册一次

player_bonuses 字符串:

参数名称	数据类型	必需项	描述
player_name	String	是	运营商的唯一标识符
bonus_amount	Integer	是	玩家的自定义红利金额 值度: 值≥1
is_unlimited_bonus_maximum_conversion_amount	Boolean	是	将红利游戏最高转换金额设置为无限制
bonus_minimum_conversion_amount	Decimal	否*	玩家自定义的红利游戏最低转换金额 值度: 值≥1 默认值: 0 注: 如果值 = 0, 系统将使用默认转换金额
bonus_maximum_conversion_amount	Decimal	否*	玩家自定义的红利游戏最高转换金额 值度: 值≥1 默认值: 0 注: 如果值 = 0, 系统将使用默认转换金额
说明	String	否	转移的参考说明

*仅适用于 *is_unlimited_bonus_maximum_conversion_amount* 参数值为错误的情况。

例子:

URL:

https://api.pg-bo.me/external/Bonus/v1/TransferInFlexibleBonus?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

```
operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&player_bonuses={"player_name": "player1", "bonus_amount": 10, "is_unlimited_bonus_maximum_conversion_amount": false, "bonus_maximum_conversion_amount": 100.50, "description": "VIP"}&bonus_id=3&transfer_reference=bonusgame_20181128&allow_multiple=false
```

返回结果

参数:

JSON 数组

参数名称	数据类型	描述
transactionId	String	交易的唯一标识符
operatorToken	String	运营商的唯一标识符
playerName	String	玩家的唯一标识符
currencyCode	String	玩家的货币
status	Integer	转移请求的状态 0: 失败 1: 成功 2: 待处理（对于尚未创建的玩家，红利游戏将在创建玩家后自动分配给该玩家）

例子:

```
{
  "data": [
    {
      "transactionId": 10,
      "operatorToken": "abc123",
      "playerName": "player1",
      "currencyCode": "CNY",
      "status": 1
    },
    {
      "transactionId": 1,
      "operatorToken": "1234",
      "playerName": "player2",
      "currencyCode": "CNY",
      "status": 1
    }
  ],
  "error": null
}
```

2.3.10 取消红利游戏

取消已创建的红利游戏：

请求参数

API URL : {PgSoftAPIDomain}/Bonus/v1/CancelBonus

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符（GUID） 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
bonus_id	String	是	红利游戏的唯一标识符

例子：

URL：

https://api.pg-bo.me/external/Bonus/v1/CancelBonus?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体：

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&bonus_id=10919

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
actionResult	Boolean	请求结果: False: 失败 True: 成功

例子:

```
{
  "data": {
    "actionResult": true
  },
  "error": null
}
```

2.3.11 取消已注册玩家的红利游戏

取消特定已注册玩家的红利游戏：

请求参数

API URL : {PgSoftAPIDomain}/Bonus/v1/CancelPlayerBonus

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符（GUID） 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
player_name	String	是	玩家的唯一标识符
balance_id	Integer	是	玩家红利钱包的唯一标识符

例子：

URL：

https://api.pg-bo.me/external/Bonus/v1/CancelPlayerBonus?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体：

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&player_name=player1&balance_id=10919

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
actionResult	Boolean	请求结果: False: 失败 True: 成功

例子:

```
{
  "data": {
    "actionResult": true
  },
  "error": null
}
```

2.3.12 取消未注册玩家的红利游戏

取消特地未注册玩家的红利游戏：

请求参数

API URL : {PgSoftAPIDomain}/Bonus/v1/CancelPendingPlayerBonus

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符（GUID） 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
pending_wallet_id	Integer	是	玩家红利钱包的唯一标识符

例子：

URL：

https://api.pg-bo.me/external/Bonus/v1/CancelPendingPlayerBonus?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体：

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&pending_wallet_id=10919

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
actionResult	Boolean	请求结果 False: 失败 True: 成功

例子:

```
{
  "data": {
    "actionResult": true
  },
  "error": null
}
```


2.3.13 取消玩家的所有红利游戏

取消玩家的所有红利游戏：

请求参数

API URL : {PgSoftAPIDomain}/Bonus/v1/CancelPlayerAllBonus

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： <ul style="list-style-type: none"> 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
player_name	String	是	玩家的唯一标识符
bonus_id	Integer	是	红利的唯一标识符

例子：

URL：

https://api.pg-bo.me/external/Bonus/v1/CancelPlayerAllBonus?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体：

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&player_name=player1&bonus_id=10919

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
actionResult	Boolean	请求结果: False: 失败 True: 成功

例子:

```
{
  "data": {
    "actionResult": true
  },
  "error": null
}
```

2.4 锦标赛 [最新版本: v2]

2.4.1 获取锦标赛列表

获取锦标赛列表:

请求参数

API URL : {PgSoftAPIDomain}/Tournament/v2/GetTournaments

HTTP 方法 : POST

URL 参数:

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注: • 请把参数值设为 GUID 格式

主体参数:

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
from_created_time	Long	是	锦标赛开始和结束的时间范围 (以毫秒为单位的 Unix 时间戳)
to_created_time	Long	是	
status	Integer Array	否	锦标赛状态: 0: 取消 1: 进行中 2: 全新 3: 结束
game_ids	Integer Array	否	游戏的唯一标识符
language	String	否	锦标赛信息的相应语言: en-US: 英语 (默认)

			zh-CN: 中文
start_time	Long	否	锦标赛开始和结束的时间范围 (以毫秒为单位的 Unix 时间戳)
end_time	Long	否	
currency	string	否	锦标赛货币

例子:

URL:

https://api.pg-bo.me/external/Tournament/v2/GetTournaments?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&status=1&status=2&game_ids=1&game_ids=2&language=zh-cn&start_time=1559145500000&end_time=1559145600000¤cy=CNY&from_created_time=1559145500000&to_created_time=1559145600000

返回结果

参数:

JSON 数组

参数名称	数据类型	描述
tournamentKey	String	锦标赛的唯一键
tournamentId	Long	锦标赛的唯一标识符
subTournamentId	Long	副锦标赛的唯一标识符
name	String	锦标赛的名称
description	String	锦标赛的说明
registrationFees	String	锦标赛的注册费
prize_type	Integer	锦标赛奖励类型 0: 物品奖励 1: 现金奖励
prize	String Array	锦标赛的奖励
registrationUrl	String	锦标赛的注册 URL
hasSubTournaments	Boolean	说明副锦标赛是否存在
subTournamentStatus	Decimal	副锦标赛的状态
subTournamentStartTime	Long	副锦标赛的开始时间

		(以毫秒为单位的 Unix 时间戳)
subTournamentEndTime	Long	副锦标赛的结束时间 (以毫秒为单位的 Unix 时间戳)
numberOfPlayers	Integer	锦标赛的参赛人数
initialPoint	Decimal	锦标赛的初始余额
qualifiedPoint	Decimal	锦标赛玩家的排名资格
gameMode	Integer	锦标赛模式： 1: 累积模式 2: 消耗模式 3: 混合模式（累积 + 消耗）
rankStatus	Integer	锦标赛的排名状态： 0: 禁用 1: 启用 2: 停止（锦标赛排名已完成）
status	Integer	锦标赛状态： 0: 取消 1: 进行中 2: 全新 3: 结束
startTime	Long	锦标赛的开始时间 (以毫秒为单位的 Unix 时间戳)
endTime	Long	锦标赛的结束时间 (以毫秒为单位的 Unix 时间戳)
createdTime	Long	锦标赛信息的创建或更新时间 (以毫秒为单位的 Unix 时间戳)
updatedAt	Long	
isCashMode	Boolean	锦标赛模式： 现金锦标赛 积分锦标赛
isRequiredRegister	Boolean	允许玩家手动或自动报名锦标赛

group	String Array	货币列表
gameIds	Uint Array	游戏 ID 列表
turnOverLimit	Decimal	玩家参加锦标赛所需的积分（玩家投注或花费的总金额）
createdBy	String	创建锦标赛的 API 或 BackOffice 用户
operatorInfos	JSON Array	运营商详情
isGlobal	Boolean	表明该锦标赛是否为全球锦标赛
numberOfPlayersLimit	Integer	可参加比赛的玩家数量
creatorOperatorToken	String	锦标赛的创建者

operatorInfos JSON 数组

参数名称	数据类型	描述
operatorName	String	运营商名称
operatorToken	String	运营商的唯一标识符
group	String Array	货币列表
createdTime	Long	锦标赛的创建时间
updatedAt	Long	锦标赛的最近更新时间
createdBy	String	锦标赛的创建者
updatedBy	String	锦标赛的最近更新源

例子:

```
{
  "data": [ "data": [
    {
      "tournamentKey": "0_P-2819-3297_P",
      "tournamentId": 2819,
      "name": "Testing",
      "creatorOperatorToken": "abcd",
      "operatorInfos": [
        {
          "operatorToken": "abcd",
          "group": [
            "CNY"
          ],
          "createdTime": 1633941729000,
          "updatedAt": 1633941729000
        }
      ],
      "description": "",
      "group": [
        "CNY"
      ],
      "gameIds": [
        29
      ],
      "registrationFees": "",
      "prizeType": 0,
      "prize": {
        "winnerPrizeDetails": [
          {
            "positionTo": 1,
            "item": "3"
          }
        ]
      },
      "registrationUrl": "",
      "hasSubTournaments": false,
      "numberOfPlayersLimit": 33,
      "numberOfPlayers": 2,
      "initialPoint": 1.00000,
      "qualifiedPoint": 33.00000,
      "turnoverLimit": 11.0,
      "gameMode": 1,
      "rankStatus": 1,
      "isCashMode": false,
      "status": 1,
      "startTime": 1633910400000,
      "endTime": 1651256325000,
      "subTournamentId": 3297,
      "subTournamentStatus": 1,
      "subTournamentStartTime": 1633910400000,
      "subTournamentEndTime": 1651256325000,
      "createdTime": 1633936037000,
      "updatedAt": 1633936037000,
      "createdBy": "admin",
    }
  ]
}
```

```
        "isRequiredRegister": true,  
        "isGlobal": false  
    },  
],  
    "error": null  
}
```


2.4.2 获取锦标赛玩家列表

获取锦标赛的玩家列表：

请求参数

API URL : {PgSoftAPIDomain}/Tournament/v2/GetTournamentPlayersAsync

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
sub_tournament_id	Integer	是	附属锦标赛的唯一标识符
page_number	Integer	是	返回数据的批号
row_count	Integer	是	个别批次的记录数量 注：值度 1-5000
player_names	String	否	锦标赛玩家的唯一标识符
from_created_time	Long	否	玩家被添加到锦标赛的时间范围。 (以毫秒为单位的 Unix 时间戳)
to_created_time	Long	否	

例子：

URL:

https://api.pg-bo.me/external/Tournament/v2/GetTournamentPlayersAsync?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&page_number=1&row_count=1000&
sub_tournament_id=1460&player_names=qafmcny18&from_created_time=1559145500000&to_c
reated_time=1559145600000

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
totalCount	Integer	记录总数
totalPage	Integer	可用批次的总数
result	JSON Array	锦标赛的玩家列表

JSON 数组

参数名称	数据类型	描述
tournamentPlayerId	String	锦标赛玩家的唯一标识符
operatorName	String	运营商名称
operatorToken	String	运营商的唯一标识符
playerName	String	玩家的唯一标识符
balanceAmount	Decimal	锦标赛玩家的余额
point	Decimal	锦标赛玩家目前的获胜点
remainingRevive	Integer	剩余的锦标赛重玩次数
remainingRegister	Integer	剩余的锦标赛报名次数
createdTime	Long	锦标赛玩家信息的创建时间 (以毫秒为单位的 Unix 时间戳)
updatedAtTime	Long	锦标赛玩家信息的更新时间 (以毫秒为单位的 Unix 时间戳)
createdBy	String	将玩家注册到锦标赛中的 API 或 BackOffice 用户

例子:

```
{
  "data": {
    "totalCount": 1,
    "result": [
      {
        "tournamentPlayerId": 0-456,
        "operatorName": "Pocket Game",
        "operatorToken": "1234",
        "playerName": "qafmcny18",
        "balanceAmount": 96,
        "point": 544.05,
        "remainingRevive": 0,
        "remainingRegister": 9,
        "createdTime": 1561640502000,
        "updatedAt": 1561640502000,
        "createdBy": "mei"
      },
      {
        "tournamentPlayerId": 0-456,
        "operatorName": "Pocket Game",
        "operatorToken": "1234",
        "playerName": "qafmcny19",
        "balanceAmount": 96,
        "point": 544.05,
        "remainingRevive": 0,
        "remainingRegister": 9,
        "createdTime": 1561640502000,
        "updatedAt": 1561640502000,
        "createdBy": "mei"
      }
    ],
    "totalPage": 1
  },
  "error": null
}
```

2.4.3 获取锦标赛的最高排名

获取锦标赛的最高排名：

请求参数

API URL : {PgSoftAPIDomain}/Tournament/v2/GetTournamentTopRankingsAsync

HTTP 方法 : POST

URL 参数:

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注: • 请把参数值设为 GUID 格式

主体参数:

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
tournament_id	Long	是	锦标赛的唯一标识符
count	Integer	是	排名记录数 注: 值度 1-5000
player_name	String	否	玩家的唯一标识符

例子:

URL:

https://api.pg-bo.me/external/Tournament/v2/GetTournamentTopRankingsAsync?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&tournament_id=41&count=10&player_name=player1

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
top_ranking	JSON Array	锦标赛最高排名列表

JSON 数组

参数名称	数据类型	描述
tournamentPlayerId	String	锦标赛玩家的唯一标识符
operatorName	String	运营商名称
operatorToken	String	运营商的唯一标识符
playerName	String	玩家的唯一标识符
point	Decimal	玩家的当前锦标赛点数
totalSpendPoint	Decimal	玩家的总支出点数
previousRank	Integer	玩家先前的排名
rank	Integer	玩家当前的排名
lastUpdatedOn	Long	最后更新排名记录的时间 (以毫秒为单位的 Unix 时间戳)
IsRankingQualifiedPlayer	Boolean	识别玩家是否有资格参加锦标赛

例子:

```
{
  "data":{
    "top_ranking": [
      {
        "tournamentPlayerId": 0-456,
        "operatorName": "Pocket Game",
        "operatorToken": "1234",
        "playerName": "player1",
        "point": 30360.98,
        "totalSpendPoint": 69376.8,
        "previousRank": 1,
        "rank": 1,
        "lastUpdatedOn": 1540380199000,
        "IsRankingQualifiedPlayer": true
      },
      {
        "tournamentPlayerId": 0-457,
        "operatorName": "Pocket Game",
        "operatorToken": "1234",
        "playerName": "player2",
        "point": 10000.74,
        "totalSpendPoint": 33,
        "previousRank": 2,
        "rank": 2,
        "lastUpdatedOn": 1540380199000
        "IsRankingQualifiedPlayer": true
      }
    ]
  },
  "error": null
}
```

2.4.4 注册锦标赛玩家

将多个玩家添加到锦标赛游戏中：

请求参数

API URL : {PgSoftAPIDomain}/Tournament/v2/CreateTournamentPlayersAsync

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符（GUID） 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
tournament_id	Long	是	锦标赛的唯一标识符 注：请使用 PG BackOffice 来创建锦标赛
player_names	String Array	是	玩家的唯一标识符
allow_re-register	Boolean	是	允许玩家重新报名同一个锦标赛 True: 允许 False: 不允许

例子：

URL：

https://api.pg-bo.me/external/Tournament/v2/CreateTournamentPlayersAsync?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体：

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&tournament_id=41&player_names=player1 &player_names=player2&allow_re-register=true

返回结果

参数：

JSON 数组

参数名称	数据类型	描述
operatorToken	Integer	运营商的唯一标识符
tournamentPlayerId	String	锦标赛玩家的唯一标识符
playerName	String	玩家的唯一标识符
balanceAmount	Integer	玩家的初始点
registerAllowed	Integer	允许玩家重新报名锦标赛的次数
reviveAllowed	Integer	允许玩家重置锦标赛的次数
status	Integer	锦标赛玩家的状态： 0: 重复 1: 成功 2: 已达报名次数上限 3: 无效玩家
tournamentKey	String	锦标赛的唯一键

例子:

```
{
  "data": [
    {
      "tournamentKey": "1855-2266-833649_P",
      "tournamentPlayerId": 0-456,
      "operatorToken": "1234",
      "playerName": " player1",
      "balanceAmount": 10000,
      "registerAllowed": 98,
      "reviveAllowed": 3,
      "status": 1
    },
    {
      "tournamentKey": "1865-2266-833649_P",
      "tournamentPlayerId": 0-457,
      "operatorToken": "1234",
      "playerName": " player1",
      "balanceAmount": 10000,
      "registerAllowed": 98,
      "reviveAllowed": 3,
      "status": 1
    },
    {
      "tournamentKey": "1866-2266-833649_P",
      "tournamentPlayerId": 0-458,
      "operatorToken": "1234",
      "playerName": " player2",
      "balanceAmount": 10000,
      "registerAllowed": 98,
      "reviveAllowed": 3,
      "status": 1
    }
  ],
  "error": null
}
```

2.4.5 获取锦标赛的现金奖励列表

获取锦标赛的现金奖励列表：

请求参数

API URL : {PgSoftAPIDomain}/Tournament/v2/GetTournamentAutoRewardCashPrize

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
start_time	Long	是	记录开始和结束的时间范围 (以毫秒为单位的 Unix 时间戳) 注： 值为 1 - 14 天
end_time	Long	是	
status	Integer	是	奖励状态 1： 获取奖励成功 0： 获取奖励失败
transaction_id	Long	否	奖励交易的唯一标识符
tournament_ids	Long	是	锦标赛的唯一标识符
player_name	String	否	锦标赛玩家的唯一标识符
currency	String	否	锦标赛货币
page_number	Integer	否	返回数据的批号 默认值： 1

row_count	Integer	否	个别批次的记录数量 默认值：5000
-----------	---------	---	-----------------------

例子：

URL：

https://api.pg-bo.me/external/Tournament/v2/GetTournamentAutoRewardCashPrize?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体：

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&start_time=1559145500000&end_time=1559145600000&status=1

返回结果

参数：

JSON 数组

参数名称	数据类型	描述
totalCount	Integer	锦标赛记录总数
totalPage	Integer	锦标赛结果总页数
result	JSON Array	锦标赛的优胜者列表

result JSON 数组

参数名称	数据类型	描述
tournamentId	Integer	锦标赛的唯一标识符
currency	string	锦标赛货币
winners	JSON Array	锦标赛的优胜者列表

winners JSON 数组

参数名称	数据类型	描述
transactionId	Long	奖励交易的唯一标识符
transactionTime	Long	奖励的交易时间 (以毫秒为单位的 Unix 时间戳)
tournamentPlayerId	String	锦标赛玩家的唯一标识符
operatorName	String	运营商名称

operatorToken	String	运营商的唯一标识符
playerName	String	玩家的唯一标识符
point	Decimal	玩家在锦标赛中所赢得的总积分
rank	Integer	锦标赛的玩家排名
playerCurrency	String	玩家使用的货币
adjustmentAmount	Decimal	玩家在锦标赛里的奖励金额
message	String	如果有错误会弹出错误信息

例子:

```
{
  "data": {
    "totalCount": 3,
    "result": [
      {
        "tournamentId": 175,
        "currency": "CNY",
        "winners": [
          {
            "transactionId": 27021004,
            "transactionTime": 1599811942000,
            "tournamentPlayerId": 0-456,
            "operatorName": "Pocket Game",
            "operatorToken": "abcd",
            "playerName": "calven77407A5275",
            "point": "233.03332",
            "rank": 2,
            "playerCurrency": "CNY",
            "adjustmentAmount": 20000.00000,
          }
        ]
      },
      {
        "tournamentId": 184,
        "currency": "CNY",
        "winners": [
          {
            "transactionId": 27021007,
            "transactionTime": 1599811943000,
            "tournamentPlayerId": 0-457,
            "operatorName": "Pocket Game",
            "operatorToken": "abcd",
            "playerName": "JiaXiangHong",
            "point": "213.03332",
            "rank": 1,
            "playerCurrency": "CNY",
            "adjustmentAmount": 200.00000
          },
          {

```

```
        "transactionId": 27021008,  
        "transactionTime": 1599811943000,  
        "operatorName": "Pocket Game",  
        "operatorToken": "abcd",  
        "tournamentPlayerId": 0-458,  
        "point": "633.03332",  
        "playerName": "dong",  
        "rank": 2,  
        "playerCurrency": "CNY",  
        "adjustmentAmount": 100.00000  
    }  
  ]  
},  
"totalPage": 1  
},  
"error": null  
}
```

2.4.6 创建锦标赛

创建锦标赛：

请求参数

API URL : {PgSoftAPIDomain}/Tournament/v2/CreateTournament

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符（GUID） 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
tournament_info	JSON Array	是	锦标赛信息
has_sub_tournaments	Boolean	是	识别该锦标赛是否有附属锦标赛
game_mode	Uint	是	积分锦标赛： 1：累积模式 2：消耗模式 3：混合模式 现金锦标赛： 4：净利润率 5：赢奖倍数 6：净利润 7：赢奖金额 8：赢奖次数 9：旋转次数

			10: 总投注额
qualified_point	Decimal	是	排行榜的积分要求
initial_point	Decimal	是	锦标赛的初始积分（仅适用于积分锦标赛）
game_ids	Uint Array	是	锦标赛游戏
currency	String Array	是	锦标赛货币
start_time	Long	是	锦标赛开始时间
end_time	Long	是	锦标赛结束时间
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
maximum_revive_allowed	Uint	否	锦标赛的重玩次数上限
maximum_register_allowed	Uint	否	锦标赛的重新报名次数上限
sub_tournament_info	JSON Array	取决于 has_sub_tournament	附属锦标赛信息
number_of_player_limit	Uint	否	报名锦标赛的玩家人数上限
minimum_bet_amount	Decimal	否	报名锦标赛的最低投注额
minimum_bet_count	Uint	否	报名锦标赛的最低投注次数
leaderboard_player	Uint	否	排行榜所显示的玩家人数
player_multiplier	Uint	否	所有锦标赛玩家的奖金倍数
required_register	Bool	否	允许玩家手动报名当前的锦标赛
participant_tokens	String Array	是	为当前锦标赛注册的运营商

tournament_info JSON 数组

参数名称	数据类型	描述
language	String	锦标赛语言
info	JSON Array	锦标赛信息

Info JSON 数组

参数名称	数据类型	描述
name	String	锦标赛名称
description	String	附加说明
rules	String	锦标赛规则
winning_criteria	String	获胜要求
registration_fees	String	锦标赛注册费
registration_url	String	锦标赛注册 URL
prize_type	Uint	锦标赛奖励类型 0: 物品奖励 1: 现金奖励 注: 现金奖励将在锦标赛结束后自动转入玩家钱包。
prize	JSON Array	奖励信息

Prize JSON 数组

参数名称	数据类型	描述
position_to	Integer	获奖人数范围 例子: 3 = 当前奖励将颁发给第一至第三名优胜者。
item	String	奖品 注: 现金奖励将依据游戏内的货币基础汇率进行派奖。

sub_tournament_info JSON 数组

参数名称	数据类型	描述
game_ids	Uint Array	附属锦标赛游戏 ID
start_time	Long	附属锦标赛开始时间
end_time	Long	附属锦标赛结束时间

例子:

URL:

https://api.pg-bo.me/external/Tournament/v2/CreateTournament?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

```
tournament_info={"language":"fi-
FI","info":{"name":"TournamentName","description":"TournamentDesc","rules":"Tournamen
tRules","winning_criteria":"WinningCriteria","registration_fees":"RegistrationF
ees","registration_url":"RegistrationUrl","prize_type":0,"prize_currency":"CNY","p
rize":{"winner_prize_details":[{"position_to":1,"item":"1st
prize"}, {"position_to":3,"item":"2nd prize"}, {"position_to":10,"item":"3rd
prize"}]}}}&sub_tournament_info={"game_ids":[1,2,3],"start_time":1613623305000,"en
d_time":1613796105000}&has_sub_tournaments=true&start_time=1613623305000&end_time=
1613796105000&currency=JPY&qualified_point=200&initial_point=199&game_mode=1&game_
ids=1&game_ids=2&operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&maximum_revi
ve_allowed=10&maximum_register_allowed=10&number_of_player_limit=10&minimum_bet_am
ount=3.1&minimum_bet_count=5&leaderboard_player=100&player_multiplier=2&required_r
egister=true&participant_tokens=abc123&participant_tokens=abc1234
```

返回结果

主体参数:

参数名称	数据类型	描述
tournamentId	Uint	创建锦标赛 ID

例子:

```
{
  "data": {
    "tournamentId": 1859
  },
  "error": null
}
```

2.4.7 更新锦标赛

更新锦标赛：

请求参数

API URL : {PgSoftAPIDomain}/Tournament/v2/UpdateTournament

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符（GUID） 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
tournament_id	Integer	是	锦标赛的唯一标识符
start_time	Long	是	锦标赛开始时间
end_time	Long	是	锦标赛结束时间
qualified_point	Decimal	是	排行榜的积分要求
game_ids	Integer Array	是	锦标赛游戏
currency	String Array	是	锦标赛的货币列表
number_of_player_limit	Integer	否	报名锦标赛的玩家人数上限
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
participant_tokens	String Array	是	为当前锦标赛注册的运营商

例子:

URL:

https://api.pg-bo.me/external/Tournament/v2/UpdateTournament?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

tournament_id=1757&start_time=1604016000000&end_time=1604102400000&qualified_point=9639&game_ids=2&game_ids=1¤cy=JPY¤cy=CNY&number_of_player_limit=9639&operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&participant_tokens=abc1234

返回结果

主体参数:

参数名称	数据类型	描述
data	Boolean	更新状态

例子:

```
{
  "data": true,
  "error": null
}
```

2.4.8 更新附属锦标赛信息

更新附属锦标赛信息：

请求参数

API URL : {PgSoftAPIDomain}/Tournament/v2/UpdateSubTournament

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
tournament_id	Uint	是	锦标赛的唯一标识符
sub_tournament_info	JSON Array	是	附属锦标赛信息
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
participant_tokens	String Array	是	运营商的唯一标识符

sub_tournament_info JSON 数组

参数名称	数据类型	描述
sub_tournament_id	Uint	附属锦标赛的唯一标识符
Interval_start_time	Long	附属锦标赛的开始/结束时间（必须在主锦标赛的开始/结束时间之内）
Interval_end_time	Long	
game_ids	Uint Array	附属锦标赛游戏

例子:

URL:

`https://api.pg-bo.me/external/Tournament/v2/UpdateSubTournament?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311`

主体:

`tournament_id=1859&operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&sub_tournament_info:{"sub_tournament_id":1737,"interval_start_time":1604016000000,"interval_end_time":1604102400000,"game_ids":[1,2]}&participant_tokens=abc123&participant_tokens=abc124`

返回结果

主体参数:

参数名称	数据类型	描述
data	Boolean	更新状态

例子:

```
{
  "data": true,
  "error": null
}
```

2.4.9 获取锦标赛信息

获取锦标赛信息：

请求参数

API URL :

{PgSoftAPIDomain}/Tournament/v2/GetTournamentInfoLanguageByTournamentId

HTTP 方法 : POST

URL 参数:

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注: • 请把参数值设为 GUID 格式

主体参数:

参数名称	数据类型	必需项	描述
tournament_id	Uint	是	锦标赛的唯一标识符
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码

例子:

URL:

https://api.pg-bo.me/external/Tournament/v2/GetTournamentInfoLanguageByTournamentId?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

tournament_id=1845&operator_token=abc123&secret_key=a1b25cde5f3gh46ijk1

返回结果

主体参数:

参数名称	数据类型	描述
tournamentInfolId	UInt	锦标赛信息 ID 的唯一标识符
language	String	锦标赛所支持的语言
name	String	锦标赛名称
description	String	附加说明
rules	String	锦标赛规则
winningCriteria	String	获胜要求
registrationFees	String	锦标赛注册费
registrationUrl	String	锦标赛注册 URL
prizeType	UInt	锦标赛奖励类型 0: 物品奖励 1: 现金奖励
prize	JSON Array	锦标赛优胜者的奖励详情
prize_currency	String	锦标赛奖励的币种
createdTime	Long	锦标赛的创建时间
updatedAtTime	Long	锦标赛的更新时间
createdBy	String	将玩家注册到锦标赛中的 API 或 BackOffice 用户
updatedBy	String	更新锦标赛中的 API 或 BackOffice 用户

例子:

```
{
  "data": [
    {
      "tournamentInfoId": 1876,
      "language": "ja-JA",
      "name": "ILName",
      "description": "ILDesc",
      "rules": "ILRules",
      "winningCriteria": "ILWinningCt",
      "registrationFees": "ILRegistrationFees",
      "registrationUrl": "ILRegistrationUrl",
      "prizeType": 1,
      "prize": {
        "winnerPrizeDetails": {
          "1": {
            "positionFrom": 1,
            "positionTo": 1,
            "item": "myItem"
          },
          "2": {
            "positionFrom": 2,
            "positionTo": 2,
            "item": "myItem2"
          },
          "3": {
            "positionFrom": 3,
            "positionTo": 3,
            "item": "myItem3"
          }
        }
      },
      "prizeCurrency": "CNY",
      "createdTime": 1606969557000,
      "updatedAt": 1607315386000,
      "createdBy": "External API",
      "updatedBy": "External API"
    }
  ],
  "error": null
}
```


2.4.10 创建锦标赛信息

创建锦标赛信息:

请求参数

API URL : {PgSoftAPIDomain}/Tournament/v2/CreateTournamentInfoLanguage

HTTP 方法 : POST

URL 参数:

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注: • 请把参数值设为 GUID 格式

主体参数:

参数名称	数据类型	必需项	描述
tournament_id	UInt	是	锦标赛的唯一标识符
name	String	是	锦标赛名称
language	String	是	锦标赛所支持的语言
description	String	否	附加说明
rules	String	否	锦标赛规则
winning_criteria	String	否	获胜要求
registration_fees	String	否	锦标赛注册费
registration_url	String	否	锦标赛注册 URL
prize	JSON Array	否	锦标赛奖励
prize_type	UInt	否	锦标赛奖励类型 0: 物品奖励 1: 现金奖励
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码

prize_currency	String	是	锦标赛奖励的币种
----------------	--------	---	----------

prize JSON 数组

参数名称	数据类型	描述
winner_prize_details	JSON Array	锦标赛优胜者的奖励

prize JSON 数组

参数名称	数据类型	描述
position_to	Integer	获奖人数范围 例子：3 = 当前奖励将颁发给第一至第三名优胜者。
item	String	奖品

例子：

URL：

https://api.pg-bo.me/external/Tournament/v2/CreateTournamentInfoLanguage?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体：

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&tournament_id=1845name=InfoLanguageName&language=ja-JA&description=XXX&rules=XXX&winning_criteria=XXX®istration_fees=XXX®istration_url=XXX&prize:{\"winner_prize_details\": [{\"position_to\":1,\"item\":\"myItem\"},{\"position_to\":2,\"item\":\"myItem2\"},{\"position_to\":3,\"item\":\"myItem3\"}]}&prize_type=1&prize_currency=\"CNY\"

返回结果

主体参数：

参数名称	数据类型	描述
tournamentInfoId	Uint	锦标赛信息的唯一标识符

例子：

```
{
  "data": 1895,
  "error": null
}
```

2.4.11 更新锦标赛信息

更新现有的锦标赛信息：

请求参数

API URL : {PgSoftAPIDomain}/Tournament/v2/UpdateTournamentInfoLanguage

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
tournament_id	Uint	是	锦标赛的唯一标识符
tournament_info_id	String	是	锦标赛信息的唯一标识符
name	String	是	锦标赛名称
language	String	是	锦标赛所支持的语言
description	String	否	附加说明
rules	String	否	锦标赛规则
winning_criteria	String	否	获胜要求
registration_fees	String	否	锦标赛注册费
registration_url	String	否	锦标赛注册 URL
prize	JSON Array	否	锦标赛奖励
prize_type	Uint	否	锦标赛奖励类型 0: 物品奖励 1: 现金奖励
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码

prize JSON 数组

参数名称	数据类型	描述
winner_prize_details	JSON Array	锦标赛优胜者的奖励

prize JSON 数组

参数名称	数据类型	描述
position_to	Integer	获奖人数范围 例子：3 = 当前奖励将颁发给第一至第三名优胜者。
item	String	奖品

例子：

URL：

https://api.pg-bo.me/external/Tournament/v2/UpdateTournamentInfoLanguage?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体：

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&tournament_id=1845&tournament_info_id=1876&name=ILName&language=ja-JA&description=ILDesc&rules=ILRules&winning_criteria=ILWinningCt®istration_fees=ILRegistrationFees®istration_url=ILRegistrationUrl&prize:{"winner_prize_details":[{"position_to":1,"item":"myItem"}, {"position_to":2,"item":"myItem2"}, {"position_to":3,"item":"myItem3"}]}&prize_type=1&prize_currency="CNY"

返回结果

主体参数：

参数名称	数据类型	描述
data	Boolean	更新状态

例子：

```
{
  "data": true,
  "error": null
}
```

2.4.12 删除锦标赛信息

从锦标赛中删除某特定信息：

请求参数

API URL : {PgSoftAPIDomain}/Tournament/v2/DeleteTournamentInfoLanguage

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
tournament_id	UInt	是	锦标赛信息的唯一标识符
tournament_info_id	String	是	锦标赛信息 ID 的唯一标识符
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码

例子：

URL：

https://api.pg-bo.me/external/Tournament/v2/DeleteTournamentInfoLanguage?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体：

tournament_id=1845&tournament_info_id=1875&operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl

返回结果

主体参数:

参数名称	数据类型	描述
data	Boolean	更新状态

例子:

```
{
  "data": true,
  "error": null
}
```

2.4.13 删除锦标赛玩家

删除锦标赛玩家及其进度：

请求参数

API URL : {PgSoftAPIDomain}/Tournament/v2/CancelTournamentPlayer

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
tournament_player_id	String	是	锦标赛玩家 ID 的唯一标识符
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码

例子：

URL：

https://api.pg-bo.me/external/Tournament/v2/CancelTournamentPlayer?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体：

tournament_player_id=843662&operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl

返回结果

主体参数:

参数名称	数据类型	描述
data	Boolean	移除玩家状态

例子:

```
{
  "data": true,
  "error": null
}
```


2.4.14 重置玩家

重置特定玩家的锦标赛进度：

请求参数

API URL : {PgSoftAPIDomain}/Tournament/v2/RevivePlayers

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
player_name	String	是	玩家名称的唯一标识符
sub_tournament_id	Uint	是	附属锦标赛的唯一标识符
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码

例子：

URL：

https://api.pg-bo.me/external/Tournament/v2/RevivePlayers?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体：

tournament_player_id=843662&operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl

返回结果

主体参数:

参数名称	数据类型	描述
tournamentKey	String	锦标赛的唯一键
tournamentPlayerId	String	锦标赛玩家 ID 的唯一标识符
operatorToken	String	运营商的唯一标识符
playerName	String	玩家名称的唯一标识符
balanceAmount	Decimal	重置时给予玩家的金额
registerAllowed	Integer	允许玩家重新报名锦标赛的次数
reviveAllowed	Integer	允许玩家重置锦标赛的次数

例子:

```
{
  "data": {
    "tournamentKey": "1870-2283-843667_P",
    "tournamentPlayerId": 0-456,
    "operatorToken": "a*d",
    "playerName": "hltestdev",
    "balanceAmount": 200.000000,
    "registerAllowed": 9,
    "reviveAllowed": 9
  },
  "error": null
}
```

2.5 投注历史记录 [最新版本: v4]

为获取各种类型的投注历史数据，PG Soft 提供了各种不同的外部应用编程接口(API)。

请注意，API 数据仅在每次旋转之后的 3 – 5 分钟之内可用。

此历史记录仅提供状态为“已完成”的投注记录，运营商可通过 PG 后台查看待处理和失败投注。

注意：

外部投注历史的 API 为专用 API，并使用独立的 URL: **{DataGrabAPIDomain}**

2.5.1 获取历史记录

获取投注历史记录：

注意

- 运营商可获得最近 60 天的投注历史记录。
- 此 API 仅适用于报告目的，我们不建议您使用此 API 于任何其他业务操作。若您需要使用此 API 于业务运营，请联系 PG。
- PG 或将通过更新数据行版本来重新推送数据。运营商需通过 *betId* 来识别和更新受重复数据。
- 数据行版本并不是唯一值。运营商必须为每个请求提取至少 1500 条记录。

步骤

- **步骤一：**在第一个 GetHistory API 调用中，设置 *row_version* = 1
- **步骤二：**保存每个获取数据的请求调用中的 *rowVersion* 最大值
- **步骤三：**在接下来的调用中，将 *row_version* 值设置为步骤二里所保存的 *rowVersion*
- 重复第二及第三步骤，直到返回的记录少于所需数量（例如：每个请求 1500 条记录）
- 如果返回的记录数少于所需数量，请停止并等待一个时间间隔（建议 5 分钟）再进行下一个 API 调用
- 可通过检查每个调用中重复的 *betId* 来识别重复的记录

请求参数

API URL : {DataGrabAPIDomain}/Bet/v4/GetHistory

HTTP 方法 : POST

URL 参数:

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注: • 请把参数值设为 GUID 格式

主体参数:

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
count	Integer	是	每批的记录数 注: 值度 1500 - 5000
bet_type	Integer	是	投注类型的投注记录: 1: 真实游戏
row_version	Long	是	数据更新时间 (以毫秒为单位的 Unix 时间戳) 注: • 首次呼叫时将值设置为 1 • 运营商需保存每个呼叫最大的 row_version 以用作下一个呼叫请求的 row_version 值
hands_status	Long	否	投注状态: 0:全部 (默认) 1: 非最后一手投注 2: 最后一手投注 3: 已调整

例子:

URL:

https://api.pg-bo.me/external-datagraber/Bet/v4/GetHistory?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&count=5000&bet_type=1&row_version=1346592723000&hands_status=0

返回结果

参数:

JSON 数组

参数名称	数据类型	描述
parentBetId	Ulong	母注单的唯一标识符
betId	Ulong	子投注的唯一标识符 (唯一键值)
playerName	String	玩家的唯一标识符
gameId	Integer	游戏的唯一标识符
betType	Integer	投注记录类别: 1: 真实游戏
transactionType	Integer	交易类别: 1: 现金 2: 红利 3: 免费游戏
platform	Integer	投注记录平台 (请参考 平台 以获取更多信息)
currency	String	记录货币
betAmount	Decimal*	玩家的投注额
winAmount	Decimal*	玩家的所赢金额
jackpotRtpContributionAmount	Decimal*	玩家的奖池返还率贡献额
jackpotContributionAmount	Decimal*	玩家的奖池贡献额
jackpotWinAmount	Decimal*	玩家的奖池金额

balanceBefore	Decimal*	玩家交易前的余额
balanceAfter	Decimal*	玩家交易后的余额
handsStatus	Integer	投注状态： 1: 非最后一手投注 2: 最后一手投注 3: 已调整
rowVersion	Long	数据更新时间 (以毫秒为单位的 Unix 时间戳)
betTime	Integer	当前投注的开始时间 (以毫秒为单位的 Unix 时间戳)
betEndTime	Integer	当前投注的结束时间 (以毫秒为单位的 Unix 时间戳)
isFeatureBuy	Boolean	表示旋转类型： True: 特色旋转 False: 普通旋转

*根据游戏和余额，系统将可能返回多达 6 个小数位。

例子:

```
{
  "data": [
    {
      "betId": 1443839146530971234,
      "parentBetId": 1443839146530971234,
      "playerName": "player1",
      "currency": "CNY",
      "gameId": 92,
      "platform": 1,
      "betType": 1,
      "transactionType": 1,
      "betAmount": 500.0,
      "winAmount": 0.0,
      "jackpotRtpContributionAmount": 0.0,
      "jackpotContributionAmount": 0.0,
      "jackpotWinAmount": 0.0,
      "balanceBefore": 78962.39,
      "balanceAfter": 78462.39,
      "handsStatus": 2,
      "rowVersion": 1633073283123,
      "betTime": 1633073052123,
      "betEndTime": 1633073052123,
      "isFeatureBuy": false
    },
    {
      "betId": 1443839117799981234,
      "parentBetId": 1443839028721351234,
      "playerName": "player1",
      "currency": "CNY",
      "gameId": 92,
      "platform": 1,
      "betType": 1,
      "transactionType": 1,
      "betAmount": 0.0,
      "winAmount": 0.0,
      "jackpotRtpContributionAmount": 0.0,
      "jackpotContributionAmount": 0.0,
      "jackpotWinAmount": 0.0,
      "balanceBefore": 79462.39,
      "balanceAfter": 79462.39,
      "handsStatus": 2,
      "rowVersion": 1633073284123,
      "betTime": 1633073045123,
      "betEndTime": 1633073045123,
      "isFeatureBuy": false
    }
  ],
  "error": null
}
```


2.5.2 获取特定时间内的历史记录

获取特定时间段内的投注历史记录：

注意

- 运营商可获得最近 60 天的投注历史记录。
- 此 API 仅适用于报告目的，我们不建议您使用此 API 于任何其他业务操作。若您需要使用此 API 于业务运营，请联系 PG。
- 由于数据可能会在一个小时内更新，我们建议运营商将数据抓取延迟到一个小时后，并通过自动化技术抓取数据。
- 除此之外，我们强烈建议运营商使用“获取历史记录” API，而非“获取特定时间内的历史记录” API，以避免数据更新延迟。

步骤

- **步骤一：**在第一个 GetHistory API 调用中，将 from_time 设置为您要的 unix 时间戳，并将 to_time 设置为当前时间
- **步骤二：**保存每个获取数据的请求调用中的 betEndTime 最大值
- **步骤三：**在接下来的调用中，将 from_time 参数值设置为步骤二里所保存的 betEndTime，to_time 参数值可保持不变（当前时间）
- 重复第二及第三步骤，直到返回的记录少于所需数量（例如：每个请求 1500 条记录）
- 如果返回的记录数少于所需数量，请停止并等待一个时间间隔（建议 5 分钟）再进行下一个 API 调用
- 可通过检查每个调用中重复的 betId 来识别重复的记录

请求参数

API URL : {DataGrabAPIDomain}/Bet/v4/GetHistoryForSpecificTimeRange

HTTP 方法 : POST

URL 参数:

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注: • 请把参数值设为 GUID 格式

主体参数:

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
count	Integer	是	每批的记录数 注: 值度 1500 - 5000
bet_type	Integer	是	投注类型的投注记录: 1: 真实游戏
from_time	Long	是	投注记录开始和结束的时间范围 (以毫秒为单位的 Unix 时间戳)
to_time	Long	是	
			注: 值度为 1 - 40 天

例子:

URL:

https://api.pg-bo.me/external-datagraber/Bet/v4/GetHistoryForSpecificTimeRange?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&count=5000&bet_type=1&from_time=1346592723000&to_time=1546592723000

返回结果

参数：

JSON 数组

参数名称	数据类型	描述
parentBetId	Ulong	母注单的唯一标识符
betId	Ulong	子投注的唯一标识符（唯一键值）
playerName	String	玩家的唯一标识符
gameId	Integer	游戏的唯一标识符
betType	Integer	投注记录的类别： 1: 真实游戏
transactionType	Integer	交易类别： 1: 现金 2: 红利 3: 免费游戏
platform	Integer	投注记录平台 (请参考 平台 以获取更多信息)
currency	String	记录货币
betAmount	Decimal*	玩家的投注额
winAmount	Decimal*	玩家的所赢金额
jackpotRtpContributionAmount	Decimal*	玩家的奖池返还率贡献额
jackpotContributionAmount	Decimal*	玩家的奖池贡献额
jackpotWinAmount	Decimal*	玩家的奖池金额
balanceBefore	Decimal*	玩家交易前的余额
balanceAfter	Decimal*	玩家交易后的余额
handsStatus	Integer	投注状态： 1: 非最后一手投注 2: 最后一手投注 3: 已调整
rowVersion	Long	数据更新时间 (以毫秒为单位的 Unix 时间戳)
betTime	Integer	当前投注的开始时间 (以毫秒为单位的 Unix 时间戳)

betEndTime	Integer	当前投注的结束时间 (以毫秒为单位的 Unix 时间戳)
isFeatureBuy	Boolean	表示旋转类型： True：特色旋转 False：普通旋转

*根据游戏和余额，系统将可能返回多达 6 个小数位。

例子:

```
{
  "data": [
    {
      "betId": 1443924025750851234,
      "parentBetId": 1443924025750851234,
      "playerName": "player1",
      "currency": "CNY",
      "gameId": 89,
      "platform": 1,
      "betType": 1,
      "transactionType": 1,
      "betAmount": 6.0,
      "winAmount": 0.0,
      "jackpotRtpContributionAmount": 0.0,
      "jackpotContributionAmount": 0.0,
      "jackpotWinAmount": 0.0,
      "balanceBefore": 17762.94,
      "balanceAfter": 17756.94,
      "handsStatus": 2,
      "rowVersion": 1633093531234,
      "betTime": 1633093281234,
      "betEndTime": 1633093281234,
      "isFeatureBuy": false
    },
    {
      "betId": 1443924030033241234,
      "parentBetId": 1443924030033241234,
      "playerName": "player1",
      "currency": "CNY",
      "gameId": 89,
      "platform": 1,
      "betType": 1,
      "transactionType": 1,
      "betAmount": 6.0,
      "winAmount": 36.0,
      "jackpotRtpContributionAmount": 0.0,
      "jackpotContributionAmount": 0.0,
      "jackpotWinAmount": 0.0,
      "balanceBefore": 17756.94,
      "balanceAfter": 17786.94,
      "handsStatus": 1,
      "rowVersion": 1633093521234,
      "betTime": 1633093291234,
      "betEndTime": 1633093291234,
      "isFeatureBuy": false
    }
  ],
  "error": null
}
```

2.5.3 获取单一玩家的历史记录

获取单一玩家的投注历史记录：

注意

- 运营商可获得最近 60 天的投注历史记录。
- 此 API 仅适用于报告目的，我们不建议您使用此 API 于任何其他业务操作。若您需要使用此 API 于业务运营，请联系 PG。

请求参数

API URL : {DataGrabAPIDomain}/Bet/v4/GetPlayerHistory

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： <ul style="list-style-type: none"> • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
player_name	String	是	玩家的唯一标识符
bet_type	Integer	是	投注类型的投注记录： 1: 真实游戏
start_time	Long	是	投注记录开始和结束的时间范围 (以毫秒为单位的 Unix 时间戳) 注： 值度 1 - 7 天
end_time	Long	是	

例子:

URL:

https://api.pg-bo.me/external-datagraber/Bet/v4/GetPlayerHistory?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&player_name=player1&start_time=1346592723000&end_time=1546592723000

返回结果

参数:

JSON 数组

参数名称	数据类型	描述
parentBetId	Ulong	母注单的唯一标识符
betId	Ulong	子投注的唯一标识符 (唯一键值)
playerName	String	玩家的唯一标识符
gameId	Integer	游戏的唯一标识符
betType	Integer	投注记录的类别: 1: 真实游戏
transactionType	Integer	交易类别: 1: 现金 2: 红利 3: 免费游戏
platform	Integer	投注记录平台 (请参考 平台 以获取更多信息)
currency	String	记录货币
betAmount	Decimal*	玩家的投注额
winAmount	Decimal*	玩家的所赢金额
jackpotRtpContributionAmount	Decimal*	玩家的奖池返还率贡献额
jackpotContributionAmount	Decimal*	玩家的奖池贡献额
jackpotWinAmount	Decimal*	玩家的奖池金额
balanceBefore	Decimal*	玩家交易前的余额
balanceAfter	Decimal*	玩家交易后的余额
rowVersion	Long	数据更新时间

		(以毫秒为单位的 Unix 时间戳)
betTime	Integer	当前记录日期 (以毫秒为单位的 Unix 时间戳)
handsStatus	Integer	投注状态： 1: 非最后一手投注 2: 最后一手投注 3: 已调整
betEndTime	Integer	当前投注的结束时间 (以毫秒为单位的 Unix 时间戳)
isFeatureBuy	Boolean	表示旋转类型： True: 特色旋转 False: 普通旋转

*根据游戏和余额，系统将可能返回多达 6 个小数位。

例子:

```
{
  "data": [
    {
      "betId": 1443458363362385920,
      "parentBetId": 1443458363362385920,
      "playerName": "fongying",
      "currency": "CNY",
      "gameId": 94,
      "platform": 1,
      "betType": 1,
      "transactionType": 1,
      "betAmount": 5.0,
      "winAmount": 0.0,
      "jackpotRtpContributionAmount": 0.0,
      "jackpotContributionAmount": 0.0,
      "jackpotWinAmount": 0.0,
      "balanceBefore": 10001704.78,
      "balanceAfter": 10001699.78,
      "handsStatus": 2,
      "rowVersion": 1632982509049,
      "betTime": 1632982266626,
      "betEndTime": 1632982266654,
      "isFeatureBuy": false
    },
    {
      "betId": 1443458366315175936,
      "parentBetId": 1443458366315175936,
      "playerName": "fongying",
      "currency": "CNY",
      "gameId": 94,
      "platform": 1,
      "betType": 1,
      "transactionType": 1,
      "betAmount": 5.0,
      "winAmount": 0.0,
      "jackpotRtpContributionAmount": 0.0,
      "jackpotContributionAmount": 0.0,
      "jackpotWinAmount": 0.0,
      "balanceBefore": 10001699.78,
      "balanceAfter": 10001694.78,
      "handsStatus": 2,
      "rowVersion": 1632982515184,
      "betTime": 1632982267325,
      "betEndTime": 1632982267354,
      "isFeatureBuy": false
    }
  ],
  "error": null
}
```

2.5.4 获取玩家的每日汇总

获取玩家的每日投注汇总：

注意

- 运营商可获得最近 60 天的投注历史记录。
- 此 API 仅适用于报告目的，我们不建议您使用此 API 于任何其他业务操作。若您需要使用此 API 于业务运营，请联系 PG。
- *PG 将通过更新数据行版本来重新推送数据。运营商需通过 betId 来识别和更新受重复数据。*
- 数据行版本并不是唯一值。运营商必须为每个请求提取至少 1500 条记录。

请求参数

API URL : {DataGrabAPIDomain}/Bet/v4/GetPlayerDailySummary

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符（GUID） 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
count	Integer	是	每批的记录数 注： 值度 1500 - 5000
bet_type	Integer	是	投注记录的类别： 1: 真实游戏
row_version	Long	是	数据更新时间 (以毫秒为单位的 Unix 时间戳)

time_zone	Integer	否	时区： -5: 世界标准时间 (UTC)-5 -4: 世界标准时间 (UTC)-4 0: 世界标准时间 (UTC) (默认) 1: 世界标准时间 (UTC)+1 4: 世界标准时间 (UTC)+4 7: 世界标准时间 (UTC)+7 8: 世界标准时间 (UTC)+8
-----------	---------	---	---

例子:

URL:

https://api.pg-bo.me/external-datagraber/Bet/v4/GetPlayerDailySummary?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&count=5000&bet_type=1&row_version=1346592723000&time_zone=8

返回结果

参数:

JSON 数组

参数名称	数据类型	描述
playerName	String	玩家的唯一标识符
playerType	Integer	玩家类别: 1: 真实玩家 2: 测试玩家
gameId	Integer	游戏的唯一标识符
betType	Integer	投注记录的类别: 1: 真实游戏
transactionType	Integer	交易类别: 1: 现金 2: 红利 3: 免费游戏
platform	Integer	投注记录平台 (请参考 平台 以获取更多信息)
currency	String	记录货币
hands	Integer	主投注计数
games	Integer	子投注计数
betAmount	Decimal*	玩家的投注额
winLossAmount	Decimal*	玩家输赢的总金额
jackpotContributionAmount	Decimal*	玩家的奖池贡献额
jackpotWinAmount	Decimal*	玩家的奖池金额
rowVersion	Long	数据更新时间 (以毫秒为单位的 Unix 时间戳)
summaryDate	Integer	当前记录日期 (以毫秒为单位的 Unix 时间戳)

*根据游戏和余额，系统将可能返回多达 6 个小数位。

例子:

```
{
  "data": [
    {
      "playerName": " player1",
      "playerType": 1,
      "gameId": 39,
      "betType": 1,
      "transactionType": 1,
      "platform": 1,
      "currency": "EUR",
      "hands": 2,
      "games": 2,
      "betAmount": 60.05,
      "winLossAmount": -60.05,
      "jackpotContributionAmount": 0,
      "jackpotWinAmount": 0,
      "rowVersion": 1545971022000,
      "summaryDate": 1545955200000
    },
    {
      "playerName": "player2",
      "playerType": 1,
      "gameId": 24,
      "betType": 1,
      "transactionType": 1,
      "platform": 1,
      "currency": "EUR",
      "hands": 9,
      "games": 9,
      "betAmount": 0.09,
      "winLossAmount": 1.91,
      "jackpotContributionAmount": 0,
      "jackpotWinAmount": 0,
      "rowVersion": 1546238743000,
      "summaryDate": 1546214400000
    }
  ],
  "error": null
}
```

2.5.5 获取玩家在特定时间内的每日汇总

获取玩家在特定时间段内的每日投注汇总：

注意

- 运营商可获得最近 60 天的投注历史记录。
- 此 API 仅适用于报告目的，我们不建议您使用此 API 于任何其他业务操作。若您需要使用此 API 于业务运营，请联系 PG。
- PG Soft 可通过更新数据行版本来重新推送数据。运营商需通过 *playerName*, *gameId* and *summaryDate* 来识别和更新重复数据。

请求参数

API URL : {DataGrabAPIDomain}/Bet/v4/GetPlayerDailySummaryForSpecificTimeRange
HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符（GUID） 注： <ul style="list-style-type: none"> • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
count	Integer	是	每批的记录数 注： 值度 1500 - 5000
bet_type	Integer	是	投注记录的类别： 1: 真实游戏
row_version	Long	是	数据的更新时间 (以毫秒为单位的 Unix 时间戳)
from_time	Long	是	投注记录开始和结束的时间范围

to_time	Long	是	(以毫秒为单位的 Unix 时间戳) 注: 值为 1 - 40 天
time_zone	Integer	否	时区: -5: 世界标准时间 (UTC)-5 -4: 世界标准时间 (UTC)-4 0: 世界标准时间 (UTC) (默认) 1: 世界标准时间 (UTC)+1 4: 世界标准时间 (UTC)+4 8: 世界标准时间 (UTC)+8

例子:

URL:

https://api.pg-bo.me/external-datagrabber/Bet/v4/GetPlayerDailySummaryForSpecificTimeRange?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&count=5000&bet_type=1&from_time=1346592723000&to_time=1546592723000&time_zone=8

返回结果

参数:

JSON 数组

参数名称	数据类型	描述
playerName	String	玩家的唯一标识符
playerType	Integer	玩家类别: 1: 真实玩家 2: 测试玩家
gameId	Integer	游戏的唯一标识符
betType	Integer	投注记录的类别: 1: 真实游戏
transactionType	Integer	交易类别: 1: 现金 2: 红利 3: 免费游戏
platform	Integer	投注记录平台 (请参考 平台 以获取更多信息)
currency	String	记录货币
hands	Integer	主投注计数
games	Integer	子投注计数
betAmount	Decimal*	玩家的投注额
winLossAmount	Decimal*	玩家输赢的总金额
jackpotContributionAmount	Decimal*	玩家的奖池贡献额
jackpotWinAmount	Decimal*	玩家的奖池金额
rowVersion	Long	数据更新时间 (以毫秒为单位的 Unix 时间戳)
summaryDate	Integer	当前记录日期 (以毫秒为单位的 Unix 时间戳)

*根据游戏和余额，系统将可能返回多达 6 个小数位数。

例子:

```
{
  "data": [
    {
      "playerName": " player1",
      "playerType": 1,
      "gameId": 39,
      "betType": 1,
      "transactionType": 1,
      "platform": 1,
      "currency": "EUR",
      "hands": 2,
      "games": 2,
      "betAmount": 60.05,
      "winLossAmount": -60.05,
      "jackpotContributionAmount": 0,
      "jackpotWinAmount": 0,
      "rowVersion": 1545971022000,
      "summaryDate": 1545955200000
    },
    {
      "playerName": "player2",
      "playerType": 1,
      "gameId": 24,
      "betType": 1,
      "transactionType": 1,
      "platform": 1,
      "currency": "EUR",
      "hands": 9,
      "games": 9,
      "betAmount": 0.09,
      "winLossAmount": 1.91,
      "jackpotContributionAmount": 0,
      "jackpotWinAmount": 0,
      "rowVersion": 1546238743000,
      "summaryDate": 1546214400000
    }
  ],
  "error": null
}
```

2.5.6 获取投注汇总

获取每次投注的游戏汇总报表：

注意

- 运营商可获得最近 60 天的投注历史记录。
- 此 API 仅适用于报告目的，我们不建议您使用此 API 于任何其他业务操作。若您需要使用此 API 于业务运营，请联系 PG。
- PG 或将通过更新数据行版本来重新推送数据。运营商需通过 *betId* 来识别和更新受重复数据。
- 数据行版本并不是唯一值。运营商必须为每个请求提取至少 1500 条记录。

请求参数

API URL : {DataGrabAPIDomain}/Bet/v4/GetHandsSummary

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符（GUID） 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
count	Integer	是	每批的记录数 注： 值度 1500 - 5000
bet_type	Integer	是	投注类型的投注记录： 1: 真实游戏
hands_status	Long	是	投注状态：

			0: 全部 1: 未完成 2: 已完成
row_version	Long	是	数据更新时间 (以毫秒为单位的 Unix 时间戳)

例子:

URL:

https://api.pg-bo.me/external-datagrabber/Bet/v4/GetHandsSummary?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&count=5000&bet_type=1&hands_status=2&row_version=1346592723000

返回结果

参数：

JSON 数组

参数名称	数据类型	描述
parentBetId	Ulong	母注单的唯一标识符
playerName	String	玩家的唯一标识符
gameId	Integer	游戏的唯一标识符
betType	Integer	投注记录的类别： 1: 真实游戏
transactionType	Integer	交易类别： 1: 现金 2: 红利 3: 免费游戏
platform	Integer	投注记录平台 (请参考 平台 以获取更多信息)
currency	String	记录货币
betAmount	Decimal*	玩家的投注额
winAmount	Decimal*	玩家的所赢金额
jackpotContributionAmount	Decimal*	玩家的奖池贡献额
jackpotWinAmount	Decimal*	玩家的奖池金额
balanceBefore	Decimal*	玩家交易前的余额
balanceAfter	Decimal*	玩家交易后的余额
handStatus	Integer	投注状态： 1: 非最后一手投注 2: 最后一手投注 3: 已调整
rowVersion	Long	更新数据的时间 (以毫秒为单位的 Unix 时间戳)
handStartTime	Integer	当前投注的开始时间 (以毫秒为单位的 Unix 时间戳)
handEndTime	Integer	当前投注的结束时间 (以毫秒为单位的 Unix 时间戳)

collapseSpinCount	Integer	消除的普通旋转次数
collapseFreeSpinCount	Integer	消除的免费旋转次数
isFeatureBuy	Boolean	表示旋转类型： True：特色旋转 False：普通旋转

*根据游戏和余额，系统将可能返回多达 6 个小数位。

例子:

```
{
  "data": [
    {
      "parentBetId": 1443100640863068160,
      "playerName": "fongying",
      "gameId": 103,
      "betType": 1,
      "transactionType": 1,
      "platform": 1,
      "currency": "CNY",
      "betAmount": 6.0,
      "winAmount": 0.0,
      "jackpotContributionAmount": 0.0,
      "jackpotWinAmount": 0.0,
      "balanceBefore": 10001632.88,
      "balanceAfter": 10001626.88,
      "rowVersion": 1632897431960,
      "handStartTime": 1632896978967,
      "handEndTime": 1632896978967,
      "handStatus": 2,
      "collapseSpinCount": 0,
      "collapseFreeSpinCount": 0,
      "isFeatureBuy": false
    },
    {
      "parentBetId": 1443100653622140928,
      "playerName": "fongying",
      "gameId": 103,
      "betType": 1,
      "transactionType": 1,
      "platform": 1,
      "currency": "CNY",
      "betAmount": 6.0,
      "winAmount": 0.0,
      "jackpotContributionAmount": 0.0,
      "jackpotWinAmount": 0.0,
      "balanceBefore": 10001626.88,
      "balanceAfter": 10001620.88,
      "rowVersion": 1632897431960,
      "handStartTime": 1632896982003,
      "handEndTime": 1632896982003,
      "handStatus": 2,
      "collapseSpinCount": 0,
      "collapseFreeSpinCount": 0,
      "isFeatureBuy": false
    }
  ],
  "error": null
}
```

2.5.7 获取每小时投注汇总

获取运营商每小时的投注记录报表：

注意

- 运营商可以获取最近 60 天的投注记录。
- 此 API 仅适用于报告目的，我们不建议您使用此 API 于任何其他业务操作。若您需要使用此 API 于业务运营，请联系 PG。

请求参数

API URL : {DataGrabAPIDomain}/Bet/v4/GetHandsSummaryHourly

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符（GUID） 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
from_time	Long	是	投注记录开始和结束的时间范围 (以毫秒为单位的 Unix 时间戳) 注： 值度为 1 - 40 天
to_time	Long	是	
currency	String	否	记录货币
transaction_types	Integer Array	否	交易类别： 1: 现金 2: 红利 3: 免费游戏

例子:

URL:

https://api.pg-bo.me/external-datagrabber/Bet/v4/GetHandsSummaryHourly?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&from_time=1567296000000&to_time=1569801600000¤cy=cny&transaction_types=1&transaction_type_s=2

返回结果

参数:

JSON 数组

参数名称	数据类型	描述
dateTime	Integer	每小时记录的日期和时间
totalHands	Integer	游戏投注总计数
currency	String	记录中玩家使用的货币
totalBetAmount	Decimal*	总投注额
totalWinAmount	Decimal*	派彩总金额
totalPlayerWinLossAmount	Decimal*	玩家的总输赢
totalCompanyWinLossAmount	Decimal*	公司的总输赢
transactionType	Integer	交易类别: 1: 现金 2: 红利 3: 免费游戏
totalCollapseSpinCount	Integer	消除的普通旋转总数
totalCollapseFreeSpinCount	Integer	消除的免费旋转总数

*根据游戏和余额, 系统将可能返回多达 6 个小数位数。

例子:

```
{
  "data": [
    {
      "dateTime": 1633093200123,
      "totalHands": 6,
      "currency": "CNY",
      "totalBetAmount": 23.8,
      "totalWinAmount": 114.0,
      "totalPlayerWinLossAmount": 90.2,
      "totalCompanyWinLossAmount": -90.2,
      "transactionType": 1,
      "totalCollapseSpinCount": 8,
      "totalCollapseFreeSpinCount": 0
    },
    {
      "dateTime": 1633093200123,
      "totalHands": 13,
      "currency": "EUR",
      "totalBetAmount": 15.0,
      "totalWinAmount": 3.4,
      "totalPlayerWinLossAmount": -11.6,
      "totalCompanyWinLossAmount": 11.6,
      "transactionType": 1,
      "totalCollapseSpinCount": 2,
      "totalCollapseFreeSpinCount": 0
    },
    {
      "dateTime": 1633093200213,
      "totalHands": 2,
      "currency": "THB",
      "totalBetAmount": 80.0,
      "totalWinAmount": 0.0,
      "totalPlayerWinLossAmount": -80.0,
      "totalCompanyWinLossAmount": 80.0,
      "transactionType": 1,
      "totalCollapseSpinCount": 0,
      "totalCollapseFreeSpinCount": 0
    }
  ],
  "error": null
}
```

2.5.8 获取玩家的投注总和

获取玩家在特定时间范围内的投注总和：

注意

- 运营商可获得最近 **40** 天的投注历史记录。
- 此 API 仅适用于报告目的，我们不建议您使用此 API 于任何其他业务操作。若您需要使用此 API 于业务运营，请联系 PG。

请求参数

API URL : {DataGrabAPIDomain}/Bet/v4/GetPlayerBetsSummationTotal

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符（GUID） 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
bet_type	Integer	是	投注类型的投注记录： 1: 真实游戏
from_time	Long	是	投注记录开始和结束的时间范围 (以毫秒为单位的 Unix 时间戳) 注： 值度为 1 - 40 天 时间将按小时四舍五入。
to_time	Long	是	例子： 如果运营商发送的请求为 1616250600000 (20/3/2021 2:30:00 PM) ~ 1616254200000

			(20/3/2021 3:30:00 PM), 则系统将返回 1616252400000 (20/3/2021 3:00:00 PM)至 1616256000000 (20/3/2021 4:00:00 PM)的数据
currency	String	否	记录货币
game_ids	Integer Array	否	游戏的唯一标识符
player_modes	Integer Array	否	游戏模式: 0: 固定 1: 现金红利
player_type	Integer	否	玩家类别: 0: 所有 1: 真实玩家 2: 测试玩家
transaction_type	Integer	否	交易类别: 0: 所有 1: 现金 2: 红利 3: 免费游戏
time_zone	Integer	否	时区: -5: 世界标准时间 (UTC)-5 -4: 世界标准时间 (UTC)-4 0: 世界标准时间 (UTC) (默认) 1: 世界标准时间 (UTC)+1 4: 世界标准时间 (UTC)+4 8: 世界标准时间 (UTC)+8

例子:

URL:

https://api.pg-bo.me/external-datagrabber/Bet/v4/GetPlayerBetsSummationTotal?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&bet_type=1&from_time=1346592723000&to_time=1346592723000

返回结果

参数:

JSON 数组

参数名称	数据类型	描述
totalCollapseSpinCount	Integer	消除的普通旋转总数
totalCollapseFreeSpinCount	Integer	消除的免费旋转总数
currency	String	记录货币
totalRowCount	Integer	记录总数
totalGames	Integer	游戏回合总计数
totalHands	Integer	游戏投注总计数
totalBetAmountConverted	Decimal*	已转换的玩家总投注金额 (货币为欧元)
totalWinLossAmountConverted	Decimal*	已转换的玩家总输赢金额 (货币为欧元)
totalCompanyWinLossAmountConverted	Decimal*	已转换的公司总输赢金额 (货币为欧元)
totalJackpotRtpContributionAmountConverted	Decimal*	已转换的总奖池返回率贡献额记录 (货币为欧元)
totalJackpotContributionAmountConverted	Decimal*	已转换的总奖池贡献额记录 (货币为欧元)
totalJackpotWinAmountConverted	Decimal*	已转换的总奖池赢额记录 (货币为欧元)
totalBetAmount	Decimal*	总投注金额记录
totalWinLossAmount	Decimal*	总输赢金额记录
totalCompanyWinLossAmount	Decimal*	公司的总输赢金额记录
totalJackpotRtpContributionAmount	Decimal*	总奖池返回率贡献额记录
totalJackpotContributionAmount	Decimal*	总奖池贡献额记录
totalJackpotWinAmount	Decimal*	总奖池赢额记录

*根据游戏和余额，系统将可能返回多达 6 个小数位数。

例子:

```
{
  "data": [
    {
      "totalCollapseSpinCount": 8,
      "totalCollapseFreeSpinCount": 0,
      "currency": "CNY",
      "totalRowCount": 1,
      "totalGames": 14,
      "totalHands": 6,
      "totalBetAmountConverted": 3.05092,
      "totalWinLossAmountConverted": 11.56274,
      "totalCompanyWinLossAmountConverted": -11.56274,
      "totalJackpotRtpContributionAmountConverted": 0.0,
      "totalJackpotContributionAmountConverted": 0.0,
      "totalJackpotWinAmountConverted": 0.0,
      "totalBetAmount": 23.8,
      "totalWinLossAmount": 90.2,
      "totalCompanyWinLossAmount": -90.2,
      "totalJackpotRtpContributionAmount": 0.0,
      "totalJackpotContributionAmount": 0.0,
      "totalJackpotWinAmount": 0.0
    },
    {
      "totalCollapseSpinCount": 2,
      "totalCollapseFreeSpinCount": 0,
      "currency": "EUR",
      "totalRowCount": 1,
      "totalGames": 16,
      "totalHands": 13,
      "totalBetAmountConverted": 15.0,
      "totalWinLossAmountConverted": -11.6,
      "totalCompanyWinLossAmountConverted": 11.6,
      "totalJackpotRtpContributionAmountConverted": 0.0,
      "totalJackpotContributionAmountConverted": 0.0,
      "totalJackpotWinAmountConverted": 0.0,
      "totalBetAmount": 15.0,
      "totalWinLossAmount": -11.6,
      "totalCompanyWinLossAmount": 11.6,
      "totalJackpotRtpContributionAmount": 0.0,
      "totalJackpotContributionAmount": 0.0,
      "totalJackpotWinAmount": 0.0
    }
  ],
  "error": null
}
```

2.5.9 获取玩家未完成的投注

获取玩家未完成的投注：

注意

- 运营商可获得最近 60 天的投注历史记录。
- 此 API 仅适用于报告目的，我们不建议您使用此 API 于任何其他业务操作。若您需要使用此 API 于业务运营，请联系 PG。

请求参数

API URL : {DataGrabAPIDomain}/Bet/v4/GetPlayerUnfinishedGamesBet

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符（GUID） 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
player_name	String	是	玩家的唯一标识符
from_time	Long	是	锦标赛开始和结束的时间范围 (以毫秒为单位的 Unix 时间戳) 注： 值度为 1 - 7 天
to_time	Long	是	
game_ids	Integer Array	否	游戏的唯一标识符
time_zone	Integer	否	时区： -5: 世界标准时间 (UTC)-5

			-4: 世界标准时间 (UTC)-4 0: 世界标准时间 (UTC) (默认) 1: 世界标准时间 (UTC)+1 4: 世界标准时间 (UTC)+4 8: 世界标准时间 (UTC)+8
--	--	--	---

例子:

URL:

https://api.pg-bo.me/external-datagraber/Bet/v4/GetPlayerUnfinishedGamesBet?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abcd&secret_key=abc123&game_ids=1&player_name=player123&date_from=1588723200000&date_to=1588982400000

返回结果

参数:

JSON 数组

参数名称	数据类型	描述
parentBetId	String	母注单的唯一标识符
gameId	Integer	游戏的唯一标识符
transactionType	Integer	交易类别: 1: 现金 2: 红利 3: 免费游戏
platform	Integer	投注记录平台
currency	String	记录货币
betAmount	Decimal	玩家投注
winAmount	Decimal	玩家派彩
jackpotRtpContributionAmount	Decimal*	玩家的奖池返还率贡献额
jackpotContributionAmount	Decimal	玩家的奖池贡献额
balanceBefore	Decimal	交易前的玩家余额
balanceAfter	Decimal	交易后的玩家余额
handStartTime	Long	投注的开始时间 (以毫秒为单位的 Unix 时间戳)

handEndTime	Long	投注的结束时间 (以毫秒为单位的 Unix 时间戳)
-------------	------	-------------------------------

例子:

```
{
  "data": [
    {
      "parentBetId": 1444964157178265600,
      "gameId": 108,
      "transactionType": 1,
      "platform": 1,
      "currency": "CNY",
      "betAmount": 375.0,
      "winAmount": 0.0,
      "jackpotRtpContributionAmount": 0.0,
      "jackpotContributionAmount": 0.0,
      "balanceBefore": 10001446.89,
      "balanceAfter": 10001071.89,
      "handStartTime": 1633341275871,
      "handEndTime": 1633341278912
    }
  ],
  "error": null
}
```


2.5.10 获取投注详情

检索多个投注的详细信息：

注意

- 运营商可以检索最近 60 天的投注历史记录。

请求参数

API URL : {DataGrabAPIDomain}/Bet/v4/GetBetDetails

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符（GUID） 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
bet_ids	Ulong	是	投注的唯一标识符

例子：

URL：

https://api.pg-bo.me/external-datagrabber/Bet/v4/GetBetDetails?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体：

operator_token=abcd&secret_key=abc123&bet_ids=1315486539764989952&bet_ids=1315486539764989960

返回结果

参数：

JSON 数组

参数名称	数据类型	描述
parentBetId	Ulong	母注单的唯一标识符
betId	Ulong	子投注的唯一标识符（唯一键值）
playerName	String	玩家的唯一标识符
gameId	Integer	游戏的唯一标识符
betType	Integer	交易记录类别： 1: 真实游戏
transactionType	Integer	交易类别： 1: 现金 2: 红利 3: 免费游戏
platform	Integer	投注记录平台 (请参阅平台以获取更多信息)
currency	String	记录中玩家所使用的货币
betAmount	Decimal*	玩家投注
winAmount	Decimal*	玩家派彩
jackpotRtpContribution Amount	Decimal*	玩家的奖池返还率贡献额
jackpotContributionAm ount	Decimal*	玩家的奖池贡献额
jackpotWinAmount	Decimal*	玩家的奖池所赢金额
balanceBefore	Decimal*	交易前的玩家余额
balanceAfter	Decimal*	交易后的玩家余额
handsStatus	Integer	投注状态： 1: 非最后一手投注 2: 最后一手投注 3: 已调整
rowVersion	Long	数据的更新时间 (以毫秒为单位的 Unix 时间戳)
betTime	Integer	当前投注的开始时间 (以毫秒为单位的 Unix 时间戳)

betEndTime	Integer	当前投注的结束时间 (以毫秒为单位的 Unix 时间戳)
isFeatureBuy	Boolean	表示旋转类型: True: 特色旋转 False: 普通旋转

*根据游戏和余额，系统将可能返回多达 6 个小数位数。

例子:

```
{
  "data": [
    {
      "betId": 1443924025750851584,
      "parentBetId": 1443924025750851584,
      "playerName": "jj02",
      "currency": "CNY",
      "gameId": 89,
      "platform": 1,
      "betType": 1,
      "transactionType": 1,
      "betAmount": 6.0,
      "winAmount": 0.0,
      "jackpotRtpContributionAmount": 0.0,
      "jackpotContributionAmount": 0.0,
      "jackpotWinAmount": 0.0,
      "balanceBefore": 17762.94,
      "balanceAfter": 17756.94,
      "handsStatus": 2,
      "rowVersion": 1633093533869,
      "betTime": 1633093289187,
      "betEndTime": 1633093289215,
      "isFeatureBuy": false
    }
  ],
  "error": null
}
```

2.5.11 获取失败投注

检索失败的投注：

注意

- 运营商可获得最近 60 天的投注历史记录。

请求参数

API URL : {DataGrabAPIDomain}/Bet/v4/GetFailedBets

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符（GUID） 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
from_time	Long	是	查询开始和结束的时间范围 (以毫秒为单位的 Unix 时间戳)
to_time	Long	是	注： 值为 1 - 40 天
count	Integer	是	个别批次的记录数量 注： 值为 1500 -5000

例子：

URL：

https://api.pg-bo.me/external-datagrabber/Bet/v4/GetFailedBets?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体：

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&count=5000&from_time=1346592723000&to_time=1546592723000

返回结果

参数:

JSON 数组

参数名称	数据类型	描述
betId	Integer	子投注的唯一标识符 (唯一键值)
playerName	String	玩家的唯一标识符
message	String	失败原因
betStartTime	Integer	当前投注的开始时间 (以毫秒为单位的 Unix 时间戳)

例子:

```
{
  "data": [
    {
      "betId": 10832988,
      "playerName": "testplayer1",
      "message": "Bet not found.",
      "betStartTime": 1600165980624
    }, {
      "betId": 10832989,
      "playerName": " testplayer2",
      "message": "Bet not found.",
      "betStartTime": 1600165980624
    }
  ],
  "error": null
}
```

2.5.12 待处理投注

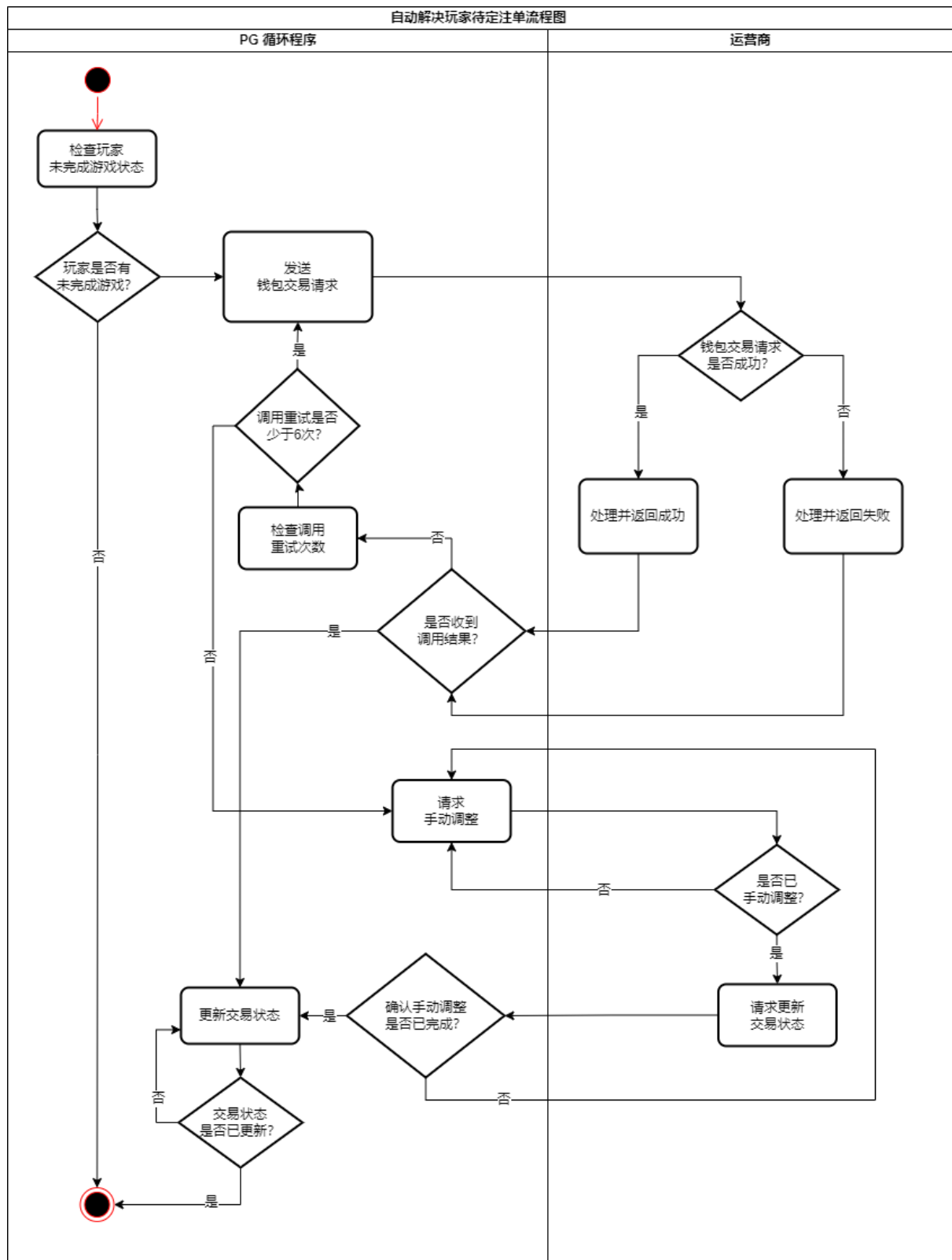
待处理投注将可能由于玩家游戏时所发生的不可预见的情况（例如：网络延迟，系统故障，系统超时等）而产生。

待处理投注通常会在玩家重新启动游戏时自动解决，但在某些罕见的情况下，该投注将保持在待处理状态：

- 遇到错误后，玩家未重新启动游戏。
- 解决待处理的投注时，PG Soft 系统未收到来自运营商的任何响应。
- 解决待处理的投注时，PG Soft 系统收到来自运营商的错误响应。

待处理投注解决时间的默认间隔为 60 分钟。超过该时间间隔的待处理投注将由 PG Soft 系统自动解决。此时间间隔是可自定义的，每个运营商都可自定义各自解决投注的时间间隔。

若在几次尝试之后，待处理投注仍解决失败，则需要由运营商手动解决该待处理投注。PG Soft 会在获得运营商的确认之后，将这些投注状态标记为完成。



2.5.13 获取待处理投注

检索待处理投注：

注意

- 运营商可获得最近 60 天的投注历史记录。

请求参数

API URL : {DataGrabAPIDomain}/Bet/v4/GetPendingBets

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符（GUID） 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
player_names	string	否	玩家的唯一标识符
from_time	Long	是	查询开始和结束的时间范围 (以毫秒为单位的 Unix 时间戳) 注： 值为 1 - 40 天
to_time	Long	是	
count	Integer	是	个别批次的记录数量 注： 值为 1500 - 5000

例子:

URL:

https://api.pg-bo.me/external-datagrabber/Bet/v4/GetPendingBets?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&player_names=testplayer1&player_names=testplayer2&count=5000&from_time=1346592723000&to_time=1546592723000

返回结果

参数:

JSON 数组

参数名称	数据类型	描述
playerName	String	玩家的唯一标识符
gameId	String	游戏的唯一标识符
betStatus	Integer	投注状态 1: 待处理
betId	Ulong	子投注的唯一标识符 (唯一键值)
betAmount	Decimal	游戏的投注金额
betTime	Integer	当前投注的开始时间 (以毫秒为单位的 Unix 时间戳)

例子:

```
{
  "data": [
    {
      "playerName": "player1",
      "gameId": 38,
      "betStatus": 1,
      "betId": 1444876729369241234,
      "betAmount": 500.0,
      "betTime": 1633320431123
    }
  ],
  "error": null
}
```

2.5.14 手动解决待处理投注

手动解决待处理的投注。

请求参数

API URL : {DataGrabAPIDomain}/Bet/v4/ManualResolveBet

HTTP 方法 : POST

URL 参数:

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注: • 请把参数值设为 GUID 格式

主体参数:

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
player_name	string	是	玩家的唯一标识符
bet_id	Integer	是	子投注的唯一标识符 (唯一键值)
is_success_bet	Boolean	是	投注状态: True: 投注成功 False: 投注失败
game_id	Integer	是	游戏的唯一标识符
balance_before_adjustment	Decimal	否	调整前的玩家余额 (仅适用于成功的投注)

例子:

URL:

https://api.pg-bo.me/external-datagrabber/Bet/v4/ManualResolveBet?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&player_name=testplayer1&bet_id=1325714346441543680&is_success_bet=true&game_id=1

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
resolveStatus	Boolean	投注的解决结果: True: 成功 False: 失败
betStatus	Integer	投注状态: 0: 非待处理 1: 待处理 2: 失败 3: 成功 4: 已调整

请参考以下用例以了解 API 的返回结果:

实际的投注 状态	第一个请求里的 Is_Success_Bet 值	第二个请求里的 Is_Success_Bet 值	API 返回结果
成功投注	True	-	resolveStatus: False, betStatus: 0
成功投注	True	True	resolveStatus: False, betStatus: 0
成功投注	False	-	resolveStatus: False, betStatus: 0
成功投注	False	False	resolveStatus: False, betStatus: 0
失败投注	True	-	resolveStatus: False, betStatus: 0
失败投注	False	-	resolveStatus: False, betStatus: 0
待处理投注	True	-	resolveStatus: True, betStatus: 3

待处理投注	True	True	resolveStatus: True, betStatus: 3
待处理投注	False	-	resolveStatus: True, betStatus: 2
待处理投注	False	False	resolveStatus: True, betStatus: 2
待处理投注	True	False	resolveStatus: False, betStatus: 3
待处理投注	False	True	resolveStatus: False, betStatus: 2

例子:

```
{
  "data": {
    "resolveStatus": true,
    "betStatus": 3
  },
  "error": null
}
```

错误的返回结果

代码	信息
1307	无效的玩家游戏会话

2.5.15 手动重置待处理游戏状态

手动重置待处理游戏的状态：

请求参数

API URL : {DataGrabAPIDomain}/Bet/v4/ManualResetGameState

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符（GUID） 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
player_name	string	是	玩家的唯一标识符
game_id	Integer	是	游戏的唯一标识符
operator_player_session	string	是	玩家的运营商会话

例子：

URL：

https://api.pg-bo.me/external-datagrabber/Bet/v4/ManualResetGameState?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体：

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&player_name=testplayer1&game_id=1

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
resetStatus	Boolean	投注的解决结果: True: 成功 False: 失败

例子:

```
{
  "data": {
    "resetStatus": true
  },
  "error": null
}
```

2.6 游戏 [最新版本: v2]

2.6.1 获取游戏列表

获取游戏列表:

请求参数

API URL : {PgSoftAPIDomain}/Game/v2/Get

HTTP 方法 : POST

URL 参数:

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注: • 请把参数值设为 GUID 格式

主体参数:

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
currency	String	是	游戏合法投注金额的货币
language	String	否	数据内容的语言 en-us: 英文 zh-cn: 中文
status	Integer	否	游戏状态: 0: 无效的游戏 1: 活跃的游戏 空值: 无效和活跃的游戏都将被选择

例子:

URL:

https://api.pg-bo.me/external/Game/v2/Get?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl¤cy=CNY&language=en-us&status=1

返回结果

参数:

JSON 数组

参数名称	数据类型	描述
gameId	Integer	游戏的唯一标识符
gameName	String	游戏名称
gameCode	String	游戏的唯一识别码
gameLegalBetAmounts	JSON Array	游戏的合法投注金额
status	Integer	全球游戏状态: 0: 无效 1: 活跃 2: 已暂停 注: 唯有在游戏状态和发布状态都显示活跃时才能进入游戏
releaseStatus	Integer	运营商的游戏状态: 0: 无效 1: 活跃 2: 已暂停 注: 唯有在游戏状态和发布状态都显示活跃时才能进入游戏
isSupportFreeGame	Boolean	表示是否支持免费游戏: True: 支持 False: 不支持
category	Integer	游戏类型:

		1: 视频老虎机游戏 2: 卡牌游戏
--	--	-----------------------

gameLegalBetAmounts JSON 数组

参数名称	数据类型	描述
gameId	Integer	游戏的唯一标识符
gameTypeId	Integer	游戏类型（仅限卡牌游戏）： 0: 无 1: 百家乐 – 免佣 2: 百家乐 – 传统
legalBetAmounts	JSON Array	游戏里可用的合法投注金额

legalBetAmounts JSON 数组

参数名称	数据类型	描述
coinSize	Decimal	游戏的投注大小
betMultiplier	Integer	游戏的投注级别
baseBet	Integer	游戏的基本投注
betAmount	Decimal	游戏的投注金额

例子:

```
{
  "data": [
    {
      "gameId": 1,
      "gameName": "Honey Trap of Diao Chan",
      "gameCode": "diaochan",
      "status": 1,
      "releaseStatus": 1,
      "isSupportFreeGame": true,
      "category": 1,
      "gameLegalBetAmounts": [
        {
          "gameId": 1,
          "gameTypeId": 0,
          "legalBetAmounts": [
            {
              "coinSize": 0.03,
              "betMultiplier": 1,
              "baseBet": 30,
              "betAmount": 0.9
            },
            {
              "coinSize": 0.03,
              "betMultiplier": 2,
              "baseBet": 30,
              "betAmount": 1.8
            }
          ]
        }
      ]
    }
  ]
}
```

2.6.2 获取游戏的合法投注金额

获取游戏的合法投注金额：

请求参数

API URL : {PgSoftAPIDomain}/Game/v2/GetGameLegalBetAmounts

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符（GUID） 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
currency	String	是	游戏合法投注金额的货币
game_ids	Integer Array	是	游戏的唯一标识符

例子：

URL：

https://api.pg-bo.me/external/Game/v2/GetGameLegalBetAmounts?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体：

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl¤cy=CNY&game_ids=1&game_ids=2

返回结果

参数:

JSON 数组

参数名称	数据类型	描述
gameId	Integer	游戏的唯一标识符
gameTypeId	Integer	游戏类型 (仅限卡牌游戏) : 0: 无 1: 百家乐 – 免佣 2: 百家乐 – 传统
legalBetAmounts	JSON Array	游戏里可用的合法投注金额

legalBetAmounts JSON 数组

参数名称	数据类型	描述
coinSize	Decimal	游戏的投注大小
betMultiplier	Integer	游戏的投注级别
baseBet	Integer	游戏的基本投注
betAmount	Decimal	游戏的投注金额

例子:

```
{
  "data": [
    {
      "gameId": 1,
      "gameTypeId": 0,
      "legalBetAmounts": [
        {
          "coinSize": 0.03,
          "betMultiplier": 1,
          "baseBet": 30,
          "betAmount": 0.9
        },
        {
          "coinSize": 0.03,
          "betMultiplier": 2,
          "baseBet": 30,
          "betAmount": 1.8
        }
      ]
    }
  ],
  "error": null
}
```

2.6.3 更改游戏发布状态

更改游戏发布状态：

请求参数

API URL : {PgSoftAPIDomain}/Game/v2/UpdateGameStatus

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
release_status	Integer	是	游戏的发布状态： 0: 无效 1: 活跃
game_ids	Integer Array	是	游戏的唯一标识符

例子：

URL：

https://api.pg-bo.me/external/Game/v2/UpdateGameStatus?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体：

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&release_status=1&game_ids=1&game_ids=2

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
actionResult	Boolean	操作请求的返回结果: true: 成功 false: 失败

例子:

```
{
  "data":{
    "actionResult": true
  },
  "error": null
}
```

2.6.4 获取游戏证书

获取游戏证书：

请求参数

API URL : {PgSoftAPIDomain}/Game/v2/GetGamesCert

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
game_ids	Integer Array	否	每个游戏的唯一标识符 空值：将选择所有游戏
game_codes	String Array	否	每个游戏的唯一识别码 空值：将选择所有游戏

例子：

URL：

https://api.pg-bo.me/external/Game/v2/GetGamesCert?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体：

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&game_ids=1&game_code=diaochan

返回结果

主体参数:

JSON 对象

参数名称	数据类型	描述
game_id	Integer	每个游戏的唯一标识符
game_name	String	每个游戏的唯一名称
game_code	String	每个游戏的唯一识别码
cert	JSON Array	证书列表

JSON 数组

参数名称	数据类型	描述
hash	String	游戏文件的哈希值
file	String	游戏文件名称

例子:

```
{
  "data": [
    {
      "game_id": 1,
      "game_name": "Honey Trap of Diao Chan",
      "game_code": "diaochan",
      "cert": [
        {
          "hash": "e2cac0a05c7407c32fd50c4e6fb6779161dfecc",
          "file": "PG.Common.GameCore.RNG.dll"
        },
        {
          "hash": "c4280d2afa00fd17e6b037bd7f4a1fa53b620775",
          "file": "PG.Common.GameCore.Slot.Engine.dll"
        },
        {
          "hash": "144c068a7011bb22e57e967e3561200ae2b8f41e",
          "file": "PG.Game.DiaoChan.Engine.dll"
        },
        {
          "hash": "bb95b970f2b57c62c96c111e8ddb6ac3961b0881",
          "file": "PG.SlotGames.Source.DiaoChan.dll"
        }
      ]
    }
  ],
  "error": null
}
```

2.7 现金 [最新版本: v2]

2.7.1 获取钱包交易

获取钱包交易:

注意:

- 运营商仅可获得最近 40 天的记录。

请求参数

API URL : {DataGrabAPIDomain}/cash/v2/GetWalletTransactions

HTTP 方法 : POST

URL 参数:

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注: • 请把参数值设为 GUID 格式

主体参数:

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
row_version	Long	是	数据更新时间 (以毫秒为单位的 Unix 时间戳)
player_name	String	否	玩家的唯一标识符
row_count	Integer	否	个别批次的记录数量 注: • 默认值: 1 • 值度: 1-5000
currency	String	否	玩家的货币

例子:

URL:

https://api.pg-bo.me/external-datagrabber/cash/v1/GetWalletTransactions?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&row_version=1346592723000&player_name=testplayer1&row_count=10¤cy=CNY

返回结果

主体参数:

JSON 对象

参数名称	数据类型	描述
lastRowVersion	Long	数据更新时间 (以毫秒为单位的 Unix 时间戳)
result	JSON Array	结果列表

JSON 数组

参数名称	数据类型	描述
transactionId	String	交易的唯一标识符
playerName	String	玩家的唯一标识符
currencyCode	String	玩家的货币
transactionType	Integer	交易类型 101: 派彩 201: 投注
transactionAmount	Decimal	交易金额
transactionFrom	Decimal	交易前的余额
transactionTo	Decimal	交易后的余额
transactionDateTime	Long	交易日期和时间

例子:

```
{
  "data": {
    "lastRowVersion": 1598617560000,
    "result": [
      {
        "transactionId": 13777918,
        "playerName": "Player1",
        "currencyCode": "CNY",
        "transactionType": 100,
        "transactionAmount": 2.00,
        "transactionFrom": 20001873.53,
        "transactionTo": 20001875.53,
        "transactionDateTime": 1598617560000
      }
    ]
  },
  "error": null
}
```

2.7.2 获取手动调整详情

获取手动调整详情：

注意：

- 运营商仅可获得最近 60 天的记录。

请求参数

API URL : {DataGrabAPIDomain}/Cash/v2/GetManualAdjustmentDetails

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： <ul style="list-style-type: none"> • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
currency	String	否	玩家的货币
row_version	Long	是	数据更新时间 (以毫秒为单位的 Unix 时间戳)
row_count	Integer	否	个别批次的记录数量 注： <ul style="list-style-type: none"> • 默认值: 1 • 值度: 1-5000
transaction_type	String	否	交易类型： 900：外部调整 901：锦标赛调整

transaction_reference_id	String	否	调整的参考 ID
status	Integer	否	手动调整状态 1: 成功 0: 待处理

例子:

URL:

https://api.pg-bo.me/external-datagrabber/Cash/v2/GetManualAdjustmentDetails?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&row_count=10¤cy=CNY&row_version=1346592723000&row_count=1&transaction_type=901&transaction_reference_id=adjustment-001&status=1

返回结果

主体参数:

JSON 对象

参数名称	数据类型	描述
lastRowVersion	Long	数据更新时间 (以毫秒为单位的 Unix 时间戳)
result	JSON Array	结果列表

JSON 数组

参数名称	数据类型	描述
playerName	String	玩家的唯一标识符
operatorToken	String	运营商的唯一标识符
currencyCode	String	玩家的货币
transactionReferenceId	String	调整的参考 ID
transactionType	Integer	交易类型: 900: 外部调整 901: 锦标赛调整

transactionAmount	Decimal	调整金额
transactionDateTime	Long	调整日期和时间
status	Integer	手动调整状态 1: 成功 0: 待处理
transactionRemark	String	调整备注

例子:

```
{
  "data": {
    "lastRowVersion": 1624444862653,
    "result": [
      {
        "playerName": "abc",
        "operatorToken": "aabbcc",
        "currencyCode": "CNY",
        "transactionReferenceId": "CASH-1234-1234567",
        "transactionType": 901,
        "transactionAmount": 1000.0,
        "transactionDateTime": 1624444862653,
        "status": 1,
        "transactionRemark": ""
      }
    ]
  },
  "error": null
}
```

2.8 玩家 [最新版本: v3]

2.8.1 获取在线玩家人数

获取在线的玩家人数:

请求参数

API URL : {PgSoftAPIDomain}/Player/v3/GetOnlinePlayerCount

HTTP 方法 : POST

URL 参数:

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注: • 请把参数值设为 GUID 格式

主体参数:

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
game_id	Integer	否	游戏的唯一标识符 注: 若未提供参数值, 系统将返回所有游戏的在线玩家人数。

例子:

URL:

https://api.pg-bo.me/external/Player/v3/GetOnlinePlayerCount?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&game_id=1

返回结果

参数:

JSON 对象

参数名	数据类型	描述
totalCount	Integer	记录总数
totalPlayerCount	Integer	在线玩家总数
onlinePlayerSummary	JSON Array	游戏的在线玩家列表

JSON 数组

参数名	数据类型	描述
rowVersion	Long	数据的更新时间 (以毫秒为单位的 Unix 时间戳)
gameId	Integer	游戏的唯一标识符
playerCount	Integer	在线玩家总数
status	Integer	玩家状态 1: 在线

例子:

```
{
  "data": {
    "onlinePlayerSummary": [
      {
        "rowVersion": 1581073260000,
        "gameId": 0,
        "playerCount": 1,
        "status": 1
      },
      {
        "rowVersion": 1581073260000,
        "gameId": 52,
        "playerCount": 1,
        "status": 1
      },
      {
        "rowVersion": 1581073260000,
        "gameId": 68,
        "playerCount": 1,
        "status": 1
      }
    ],
    "totalCount": 3,
    "totalPlayerCount": 3
  },
  "error": null
}
```

2.8.2 获取玩家的线上状态

获取最后 10 名玩家的线上状态：

请求参数

API URL : {PgSoftAPIDomain}/Player/v3/GetPlayersOnlineStatus

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
player_names	String Array	是	玩家的唯一标识符

例子：

URL：

https://api.pg-bo.me/external/Player/v3/GetPlayersOnlineStatus?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体：

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&player_names=player1&player_names=player2

返回结果

参数

JSON 数组

参数名称	数据类型	描述
playerName	String	玩家的唯一标识符
gameId	String	游戏的唯一标识符
status	Integer	玩家的线上状态 0: 离线 1: 在线

例子:

```
{
  "data": [
    {
      "playerName": "player1",
      "status": 1,
      "gameId": 12
    },
    {
      "playerName": "player2",
      "status": 0
    }
  ],
  "error": null
}
```

2.8.3 获取在线玩家列表

获取在线玩家列表。

请求参数

API URL : {PgSoftAPIDomain}/Player/v3/GetOnlinePlayers

HTTP 方法 : POST

URL 参数:

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注: <ul style="list-style-type: none"> 请把参数值设为 GUID 格式

主体参数:

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
row_count	Integer	否	个别批次的记录数量 注: <ul style="list-style-type: none"> 默认值: 1000 值度: 1-1000
row_version	Long	否	数据更新时间 (以毫秒为单位的 Unix 时间戳) 默认值: 1

例子:

URL:

https://api.pg-bo.me/external/Player/v3/GetOnlinePlayers?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&row_count=10

返回结果

参数：

JSON 数组

参数名称	数据类型	描述
playerName	String	玩家的唯一标识符
gameId	Integer	游戏的唯一标识符
rowVersion	Long	数据的更新时间 (以毫秒为单位的 Unix 时间戳) 默认值：1

例子：

```
{
  "data": [
    {
      "playerName": "tesplayer01",
      "gameId": 60,
      "rowVersion": 1607503788000
    },
    {
      "playerName": "tesplayer02",
      "gameId": 65,
      "rowVersion": 1607503809000
    }
  ],
  "error": null
}
```

2.8.4 创建玩家

在 PG 游戏系统中创建新玩家帐户：

注意

由于玩家帐户将被自动创建，以供初次访问 PG 游戏的玩家使用，所以此 API 的执行是非强制性的。

请求参数

API URL : {PgSoftAPIDomain}/v3/Player/Create

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
player_name	String	是	玩家的唯一标识符 • 玩家账号中的字母不区分大小写 • 只允许使用字母、数字和 "@"、 "-","_" 符号 注：最多 50 个字符
nickname	String	否	玩家的用户名 注：最多 50 个字符
currency	String	是	玩家货币

例子:

URL:

https://api.pg-bo.me/external/v3/Player/Create?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&player_name=player123¤cy=EUR&nickname=player123

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
action_result	Integer	请求状态 1: 成功 0: 失败

例子:

```
{
  "data": {
    "action_result": 1
  },
  "error": null
}
```

错误的返回结果

代码	信息
1034	请求无效
1035	操作失败
1200	内部服务器错误
1204	无效的运营商
1305	无效的玩家 (玩家已存在)
1315	玩家正在进行另一项操作

2.8.5 踢出玩家

将玩家从 PG 游戏中踢出，玩家仍然可以重新登录游戏：

请求参数

API URL : {PgSoftAPIDomain}/v3/Player/Kick

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
player_name	String	是	玩家的唯一标识符

例子：

URL:

https://api.pg-bo.me/external/v3/Player/Kick?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&player_name=player123

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
action_result	Integer	请求状态 1: 成功 0: 失败

例子:

```
{
  "data": {
    "action_result": 1
  },
  "error": null
}
```

错误的返回结果

代码	信息
1034	请求无效
1035	操作失败
1200	内部服务器错误
1204	无效的运营商
1305	无效的玩家 (玩家已存在)
1315	玩家正在进行另一项操作

2.8.6 冻结玩家

冻结玩家帐户，被冻结的玩家将无法再访问 PG 游戏：

请求参数

API URL : {PgSoftAPIDomain}/v3/Player/Suspend

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
player_name	String	是	玩家的唯一标识符

例子：

URL:

https://api.pg-bo.me/external/v3/Player/Suspend?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&player_name=player123&reason=suspendedplayer

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
action_result	Integer	请求状态 1: 成功 0: 失败

例子:

```
{
  "data": {
    "action_result": 1
  },
  "error": null
}
```

错误的返回结果

代码	信息
1034	请求无效
1035	操作失败
1200	内部服务器错误
1204	无效的运营商
1305	无效的玩家 (玩家已存在)
1315	玩家正在进行另一项操作

2.8.7 恢复玩家

恢复被冻结的玩家帐户：

请求参数

API URL : {PgSoftAPIDomain}/v3/Player/Reinstate

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符（GUID） 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
player_name	String	是	玩家的唯一标识符
reason	String	是	恢复玩家的原因

例子：

URL：

https://api.pg-bo.me/external/v3/Player/Reinstate?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体：

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&player_name=player123&reason=resumesuspendedplayer

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
action_result	Integer	请求状态 1: 成功 0: 失败

例子:

```
{
  "data": {
    "action_result": 1
  },
  "error": null
}
```

错误的返回结果

代码	信息
1034	请求无效
1035	操作失败
1200	内部服务器错误
1204	无效的运营商
1305	无效的玩家 (玩家已存在)
1315	玩家正在进行另一项操作

2.8.8 检查玩家状态

查看特定玩家的帐户状态：

注意

此 API 不用于检查在线玩家的状态，而仅用于检查玩家账户在 PG 系统中的状态。请使用后台以查看线上活跃的玩家。

请求参数

API URL : {PgSoftAPIDomain}/v3/Player/Check

HTTP 方法 : POST

URL 参数：

参数名称	数据类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注： • 请把参数值设为 GUID 格式

主体参数：

参数名称	数据类型	必需项	描述
operator_token	String	是	运营商的唯一标识符
secret_key	String	是	PGSoft 和运营商之间的共享密码
player_name	String	是	玩家的唯一标识符

例子：

URL：

https://api.pg-bo.me/external/v3/Player/Check?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体：

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&player_name=player123

返回结果

参数:

JSON 对象

参数名称	数据类型	描述
player_name	String	玩家的唯一标识符
status	Integer	请求状态 0: 已关闭 1: 活跃 2: 不活跃/已停用 3: 已冻结 4: 玩家正在进行另一项操作

例子:

```
{
  "data": {
    "player_name": "player123",
    "status": 1
  },
  "error": null
}
```

3. 附录

3.1 平台

代码	类别	平台
1	Web	Windows
2		macOS
3		Android
4		iOS
5		其它
6	Cordova	Android
7		iOS
8	Electron	Windows
9		macOS
10	Native	Windows
11		macOS
12		Android
13		iOS
98	others	System
99		挂机系统