**MsgRecord 消息载体**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **名称** | **示例** | **说明** |
| Id | 消息ID |  |  |
| ConvrType | 消息分组类型 | 如：单聊、群聊、系统通知 | 枚举 |
| ConvrTitle | 消息分组名称 | 如：橘子、开发组、道诚科技、智能办公项目 | 单聊时为发言人名；企业系统信息时为企业名称；项目组时为项目名称；群聊时为创建群组时的名称 |
| MsgType | 消息类型 | 如：Chat、Task、Approval、Attend、Notification、Report、Maibox | 枚举 |
| MsgTypeName | 消息类型名称 | 如：IM聊天、任务、审批、考勤、公告、报表、邮箱 | 对应枚举类型的描述 |
| MsgTitle | 消息标题 | 如：橘子的请假申请需要您审批 |  |
| MsgContent | 消息内容 | 如：  [  {"title": "请假类型", "value": "事假"},  {"title": "开始时间", "value": "2016-04-28 16:52"},… ] | 消息为系统消息时，放业务字段JSON。给定结构：  T{"title": "请假类型", "value": "事假"} 数组 |
| SpeakerType | 发送人类型 | 如：User、System | 枚举 |
| SpeakerName | 发送人名称 | 如：橘子、系统消息 |  |
| SpeakerId | 发送人ID | 如：UserId |  |
| TargetType | 听众类型 | 如：All、User、Group、Company、Project | 枚举 |
| TargetId | 消息接收方Id | 如：UserId；GroupId、OwnCompId、ProjectId | 用于Jpush、环信发送 |
| AudienceIds | 听众人员Id集合 | 如：UserId,UserId,UserId |  |
| OccurTime | 消息发生时间 |  |  |
| BusinessId | 关联业务Id |  |  |
| OperateResult | 操作结果 | 如：已通过 |  |

后台业务调用消息推送：  
1、实例化一个新的 MsgRecord；

2、调用全局JPush 连接类发送消息。Pharos.Business. GlobalJPush

MsgRecord 构造方法

public MsgRecord(string createrId, Enum.MsgType msgType, TargetType targetType, string targetId, string msgTitle, string msgContent)

（示例）

MsgRecord msg = new MsgRecord("1BD2980F86DA46D4B79E8E49A39BCA2D", Entity.Enum.MsgType.Approval, Entity.Enum.TargetType.Company, "2", "小明的请假需要您审批", [{\"title\": \"请假类型\", \"value\": \"事假\"},{\"title\": \"开始时间\", \"value\": \"2016-04-28 16:52\"} ]，” 2F34ACDF86DA46D4B79E8E49ADEEFE34”);

new GlobalJPush().SendPush(msg);