# P2Chat协议设计(v0.0.1)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本 | 状态 | 日期 | 修改人 | 说明 |
| v0.0.1 | 新建 | 2017.7.9 | BrianYi | 编写一期设计方案 |

## 编写目的

## 消息协议

### 消息格式

MessageCP由消息头部(Header)和消息载荷(Payload)构成：

|  |  |
| --- | --- |
| Header  16 bytes | Payload  0 – 256K bytes |

### 消息头部

|  |  |  |
| --- | --- | --- |
| 字段 | 大小(Bytes) | 含义 |
| MSG\_CODE | 2 | 消息码 |
| FLAGS | 2 | 标记 |
| SIZE | 4 | 消息的总长度 |
| MSG\_ID | 4 | 消息标识符(与回应消息相匹配) |
| ATTR\_NUM | 4 | 属性的数量 |

标记字段，格式如下:

|  |  |  |
| --- | --- | --- |
| 标记 | 值 | 含义 |
| BINARY | 0x0001 | 若标记被设定，则ATTR\_NUM表示二进制字节数 |
| END\_OF\_FILE | 0x0002 | 文件传输最后一部分 |

### 消息载荷

|  |  |  |
| --- | --- | --- |
| 字段 | 大小(Bytes) | 含义 |
| ATTR\_CODE | 4 | 属性码 |
| FLAGS | 1 | 标记 |
| DATA\_TYPE | 1 | 数据类型 |
| DATA | 0 – 256 KB | 取决于数据类型 |

标记字段，格式如下:

|  |  |  |
| --- | --- | --- |
| 标记 | 值 | 含义 |
| SIGNED | 0x01 | 数据类型为有符号 |
| UNSIGNED | 0x02 | 数据类型为无符号 |

数据类型字段，格式如下:

|  |  |  |
| --- | --- | --- |
| 字段 | 值 | 含义 |
| INT16 | 0x00 | 16位整型 |
| INT32 | 0x01 | 32位整型 |
| INT64 | 0x02 | 64位整型 |
| FLOAT32 | 0x03 | 32位浮点型 |
| FLOAT64 | 0x04 | 64位浮点型 |
| BINARY | 0x05 | 二进制型 |
| STRING | 0x06 | 字符串型(宽字符) |

由于数据类型的不同，数据部分的格式如下:

**INT16**

|  |
| --- |
| INT16 值(16位)  2 Bytes |

**INT32**

|  |
| --- |
| INT32 值(32位)  4 Bytes |

**INT64**

|  |
| --- |
| INT64 值(64位)  8 Bytes |

**FLOAT32**

|  |
| --- |
| FLOAT32 值(32位)  4 Bytes |

**FLOAT64**

|  |
| --- |
| FLOAT64 值(64位)  8 Bytes |

**BINARY**

|  |  |
| --- | --- |
| LENGTH 值(32位)  4 Bytes | Content (二进制)  0 – 256K bytes |

**STRING**

|  |  |
| --- | --- |
| LENGTH 值(32位)  4 Bytes | Content  0 – 256K bytes |

## Client-Server 通讯

|  |  |  |
| --- | --- | --- |
| **Client** |  | **Server** |
| MSG\_SERVER\_GET\_INFO | 🡺 |  |
|  | 🡸 | MSG\_REQUEST\_SUCCEED |
| MSG\_LOGIN(连接建立) | 🡺 |  |
|  | 🡸 | MSG\_REQUEST\_SUCCEED(携带SessionID) |
| MSG\_USER\_GET\_INFO(携带SessionID) | 🡺 |  |
|  | 🡸 | MSG\_REQUEST\_SUCCEED(携带SessionID) |
| …. | …. | …. |

**(注：Server生成的SessionID为时间戳，通过**

**chrono::system\_clock::now().time\_since\_epoch().count()获取)**

## Client-Client 通讯

|  |  |  |
| --- | --- | --- |
| **Client A** |  | **Client B** |
| MSG\_USER\_ONLINE | 🡺 |  |
|  | 🡸 | MSG\_REQUEST\_SUCCEED(携带SessionID) |
| … | … | … |
| MSG\_USER\_OFFLINE(携带SessionID) | 🡺 |  |
|  | 🡸 | MSG\_REQUEST\_SUCCEED |

**(注：Client生成的SessionID为随机数，通过63次位运算生成，即：**

**int bitPos = 63;**

**\_\_int64 sessionId = 0;**

**while (bitPos--)**

**sessionId |= (((sessionId >> bitPos) | (rand() % 2)) << bitPos);**

**)**

## 消息请求

### MSG\_SERVER\_GET\_INFO

|  |  |  |
| --- | --- | --- |
| 属性Code | 数据类型 | 含义 |
| ATTR\_SERVER\_XXX | 期待返回的类型 | 获取服务端XXX信息 |

回应:

成功：MSG\_REQUEST\_SUCCEED，携带相应的数据

失败：MSG\_REQUEST\_FAILED，携带失败原因

### MSG\_SERVER\_SET\_INFO

|  |  |  |
| --- | --- | --- |
| 属性Code | 数据类型 | 含义 |
| ATTR\_SERVER\_XXX | 数据类型 | 修改服务端XXX信息 |

回应:

成功：MSG\_REQUEST\_ SUCCEED

失败：MSG\_REQUEST\_FAILED，携带失败原因

### MSG\_USER\_GET\_INFO

|  |  |  |
| --- | --- | --- |
| 属性Code | 数据类型 | 含义 |
| ATTR\_USER\_ID | INT32 | 用户名ID |
| ATTR\_USER\_XXX | 期待返回的类型 | 获取服务端用户XXX信息 |

回应：

成功：MSG\_REQUEST\_SUCCEED，携带相应的数据

失败：MSG\_REQUEST\_FAILED，携带失败原因

### MSG\_USER\_SET\_INFO

|  |  |  |
| --- | --- | --- |
| 属性Code | 数据类型 | 含义 |
| ATTR\_USER\_ID | INT32 | 用户ID |
| ATTR\_USER\_XXX | 数据类型 | 修改服务端用户XXX信息 |

回应：

成功：MSG\_REQUEST\_SUCCEED

失败：MSG\_REQUEST\_FAILED，携带失败原因

### MSG\_GROUP\_GET\_INFO

|  |  |  |
| --- | --- | --- |
| 属性Code | 数据类型 | 含义 |
| ATTR\_GROUP\_ID | INT32 | 群组ID |
| ATTR\_GROUP\_XXX | 期待返回的类型 | 获取服务端群组XXX信息 |

回应：

成功：MSG\_REQUEST\_SUCCEED，携带相应的数据

失败：MSG\_REQUEST\_FAILED，携带失败原因

### MSG\_GROUP\_SET\_INFO

|  |  |  |
| --- | --- | --- |
| 属性Code | 数据类型 | 含义 |
| ATTR\_GROUP\_ID | INT32 | 群组ID |
| ATTR\_GROUP\_XXX | 数据类型 | 修改服务端群组XXX信息 |

回应：

成功：MSG\_REQUEST\_SUCCEED

失败：MSG\_REQUEST\_FAILED，携带失败原因

### MSG\_LOGIN(与Server连接建立)

|  |  |  |
| --- | --- | --- |
| 属性Code | 数据类型 | 含义 |
| ATTR\_USER\_USERNAME | STRING | 用户名 |
| ATTR\_USER\_PASSWORD | STRING | 密码 |

回应：

成功：MSG\_REQUEST\_SUCCEED，携带相应的数据(ATTR\_USER\_ID：用户id, ATTR\_SESSION\_ID：会话ID)

*(Server生成一个SessionID与该Client对应，Client端也保留该SessionID，后续所有消息通讯都需带有SessionID)*

失败：MSG\_REQUEST\_FAILED，携带失败原因

### MSG\_LOGOUT(与Server连接断开)

|  |  |  |
| --- | --- | --- |
| 属性Code | 数据类型 | 含义 |
| ATTR\_SESSION\_ID | INT64 | 会话ID |

回应：

成功：MSG\_REQUEST\_SUCCEED

*(Server从Dictionary中移除该SessionID)*

失败：MSG\_REQUEST\_FAILED，携带失败原因

### MSG\_USER\_MSG\_PACKET

|  |  |  |
| --- | --- | --- |
| 属性Code | 数据类型 | 含义 |
| ATTR\_SESSION\_ID | INT64 | 会话ID |
| ATTR\_MSG\_PACKET | STRING | 发送数据 |

回应：

成功：MSG\_REQUEST\_SUCCEED

失败：MSG\_REQUEST\_FAILED，携带失败原因

### MSG\_GROUP\_MSG\_PACKET

|  |  |  |
| --- | --- | --- |
| 属性Code | 数据类型 | 含义 |
| ATTR\_SESSION\_ID | INT64 | 会话ID |
| ATTR\_GROUP\_ID | INT32 | 发送方群组ID |
| ATTR\_MSG\_PACKET | STRING | 发送数据 |

回应：

成功：MSG\_REQUEST\_SUCCEED

失败：MSG\_REQUEST\_FAILED，携带失败原因

### MSG\_USER\_ONLINE(与Client连接建立)

|  |  |  |
| --- | --- | --- |
| 属性Code | 数据类型 | 含义 |
| ATTR\_USER\_ID | INT32 | 发送方用户ID |

回应：

成功：MSG\_REQUEST\_SUCCEED，携带相应的数据(ATTR\_SESSION\_ID：会话ID)

*(接收方Client生成一个SessionID与发送方Client对应，双方Client端保留该SessionID，后续所有消息通讯都需带有SessionID)*

失败：MSG\_REQUEST\_FAILED，携带失败原因

### MSG\_USER\_OFFLINE(与Client连接断开)

|  |  |  |
| --- | --- | --- |
| 属性Code | 数据类型 | 含义 |
| ATTR\_SESSION\_ID | INT64 | 会话ID |

回应：

成功：MSG\_REQUEST\_SUCCEED

失败：MSG\_REQUEST\_FAILED，携带失败原因