

# 恩布互联云通讯平台

## SDK API JQUERY 版开发手册

entboost, version 2014.177 0.3.0

2014-06-13

恩布互联、恩布云通信平台、entboost 等商标归  
深圳市恩布网络科技有限公司，版本所有  
2013-2014

## 目 录

1. 概述.....	3
1.1. 功能描述.....	3
1.2. 命名空间.....	3
2. 登录验证.....	3
2.1. 概述.....	3
2.2. 登录上线接口.....	3
2.2.1. 游客登录.....	3
2.2.2. 普通用户登录.....	4
2.3. 登出下线接口.....	5
2.3.1. 登出下线.....	5
3. 用户管理.....	6
3.1. 呼叫用户接口.....	6
3.1.1. 呼叫对方用户.....	6
4. 聊天会话.....	7
4.1. 聊天会话接口.....	7
4.1.1. 发送富文本消息.....	7
4.1.2. 发送文件.....	8
5. 异步事件.....	9
5.1. 资源加载事件.....	9
5.1.1. 加载表情完毕.....	9
5.1.2. 加载企业架构信息完毕.....	9
5.1.3. 加载个人群组信息完毕.....	10
5.1.4. 加载个人联系人信息完毕.....	10
5.2. 会话管理事件.....	10
5.2.1. 会话结束.....	10
5.2.2. 会话编号已变更.....	10
5.2.3. 接收到富文本信息.....	11
5.3. 其它事件.....	12
5.3.1. 其它未详细定义的错误信息.....	12
6. 附录.....	12
6.1. 错误信息对照表.....	12

# 1. 概述

## 1.1. 功能描述

功能	描述
富文本格式	支持文本、表情混合排版传输
表情	支持企业自定义动画表情
文件	支持文件秒传功能
离线功能	支持发离线消息功能
游客功能	支持免注册登录，游客使用功能

## 1.2. 命名空间

- A. 命名空间：jQuery.ebMsg 或 \$.ebMsg
- B. \$.ebMsg.eventHandle 是事件回调对象，系统产生的事件都由此对象中进行传递，典型使用如下：
 

```
$.ebMsg.eventHandle.onXXX = function (p1, p2) { xxx };
```
- C. \$.ebMsg.errCodeMap 是错误信息对照表（具体定义见附录【错误信息对照表】）

# 2. 登录验证

## 2.1. 概述

验证用户合法性并设置上、下线状态。

## 2.2. 登录上线接口

### 2.2.1. 游客登录

格式：\$.ebMsg.logonVisitor()

成功后回调 successCallback()函数，失败后调用 failureCallback()函数。

输入参数	默认值（非空）	描述
------	---------	----

<b>successCallback</b>	可选	<p>发送成功后回调函数，格式定义如下： sucessCallback(clientInfo)</p> <p>clientInfo 参数表示当前用户信息，结构如下： user_type: 用户类型“visitor”或“normal” um_appname: UM 服务应用名称 my_account: 当前用户账号名 my_uid: 当前用户编号 username: 当前用户昵称 description: 当前用户描述</p>
<b>failureCallback</b>	可选	<p>发送失败后回调函数，格式定义如下： failureCallback(error)</p> <p>错误类型： errCodeMap.LOGON_FAILURE 验证出错 errCodeMap.ONLINE_FAILURE 设置在线出错</p>

返回协议：

参数	默认值（非空）	描述
无		

## 2.2.2. 普通用户登录

格式：\$.ebMsg.logonAccount()

成功后回调 successCallback()函数，失败后调用 failureCallback()函数。

输入参数	默认值（非空）	描述
<b>account</b>	必填	用户账号
<b>password</b>	必填	密码
<b>successCallback</b>	可选	<p>发送成功后回调函数，格式定义如下： sucessCallback(clientInfo)</p> <p>clientInfo 参数表示当前用户信息，结构如下： user_type: 用户类型“visitor”或“normal” um_appname: UM 服务应用名称 my_account: 当前用户账号名 my_uid: 当前用户编号 username: 当前用户昵称</p>

		description: 当前用户描述
<b>failureCallback</b>	可选	发送失败后回调函数，格式定义如下： <b>failureCallback(error)</b>  错误类型： <b>errCodeMap.LOGON_FAILURE</b> 验证出错 <b>errCodeMap.ONLINE_FAILURE</b> 设置在线出错

返回协议：

参数	默认值（非空）	描述
无		

## 2.3. 登出下线接口

### 2.3.1. 登出下线

格式：\$.ebMsg.offline()

成功后回调 **successCallback()**函数，失败后调用 **failureCallback()**函数。

输入参数	默认值（非空）	描述
<b>successCallback</b>	可选	发送成功后回调函数，格式定义如下： <b>sucessCallback()</b>
<b>failureCallback</b>	可选	发送失败后回调函数，格式定义如下： <b>failureCallback(error)</b>  错误类型： <b>errCodeMap.OFFLINE_FAILURE</b> 下线出错

返回协议：

参数	默认值（非空）	描述
无		

## 3. 用户管理

### 3.1. 呼叫用户接口

#### 3.1.1. 呼叫对方用户

格式：\$.ebMsg.callAccount

成功后回调 successCallback()函数，失败后调用 failureCallback()函数。

输入参数	默认值（非空）	描述
toAccount	必填	对方用户账号
existCallId	可选	已存在会话编号
successCallback		<p>回调 successCallback(callInfo, accountInfo)</p> <p><b>参数 callInfo(会话信息)，结构如下：</b></p> <p>call_id: 会话编号 chat_id: 聊天编号 chat_key: 聊天会话密钥 group_code: 群组或部门编号 cm_url: 会话服务地址 cm_appname: 会话服务应用名称 hangup: 会话是否已结束 accounts: 会话中各对话方信息(不包含自己), 数组结构</p> <pre>{   uid: 用户编号   offline: 是否离线状态   incall: 是否准备好对话环境 }, ...]</pre> <p><b>参数 accountInfo(对方账号信息)，结构如下：</b></p> <p>uid: 对方用户编号 from_account: 对方账号 fInfo: 电子名片 结构如下</p> <pre>{   type: 类型   empCode: 群成员编号   name: 名称   mobile: 手机</pre>

		telephone: 固定电话 email: 电子邮件 title: 职务 departmentName: 群或部门名称 entpriseName: 公司名称 }
<b>failureCallback</b>	可选	失败后回调函数，格式定义如下： <b>failureCallback(error)</b>  错误类型： errCodeMap.CALLACCOUNT_NOT_EXIST 对方不存在 errCodeMap.CALLACCOUNT_NO_RESPONSE 对方没有应答 errCodeMap.CALLACCOUNT_REJECT 对方拒绝通话 errCodeMap.PARAMETER_ERROR 参数错误 errCodeMap.TIMEOUT 超时

返回协议；

参数	默认值（非空）	描述
无		

## 4. 聊天会话

### 4.1. 聊天会话接口

#### 4.1.1. 发送富文本消息

格式：\$.ebMsg.sendMessage()

成功后回调 successCallback()函数，失败后调用 failureCallback()函数。

输入参数	默认值（非空）	描述
<b>callId</b>	必填	会话编号
<b>content</b>	必填	富文本内容
<b>successCallback</b>	可选	发送成功后回调函数，格式定义如下： sucessCallback()
<b>failureCallback</b>	可选	发送失败后回调函数，格式定义如下： failureCallback(error) {

```

if(error == errCodeMap.NOT_LOGON) {
    //没有登录或登录已经失效
} else {
    //其它错误
}
}

```

返回协议：

参数	默认值（非空）	描述
无		

#### 4.1.2. 发送文件

格式：\$.ebMsg.sendFile()

当服务器保存离线文件时，回调 sendingCallback 函数；

当对方接收文件时，回调 sentCallback 函数。

输入参数	默认值（非空）	描述
<b>fileElementId</b>		浏览文件的 file 控件 id，例如： <input type="file" id="abc" />，其中 “abc”就是 fileElementId
<b>chatId</b>		聊天编号
<b>sendingCallback</b>		<p>服务器保存离线文件时回调，格式如下：</p> <p>sendingCallback(state, callId, fileName, mimeType, url)</p> <p><b>参数解析</b></p> <p><b>state:</b> 状态类型</p> <p>errCodeMap.OK //服务器离线保存成功</p> <p>errCodeMap.SENDFILE_REQUEST_FAILURE //申请上传文件失败</p> <p>errCodeMap.UPLOAD_FILE_EMPTY //文件没有内容</p> <p>errCodeMap.CONTENT_TOO_LONG //文件超长</p> <p><b>callId:</b> 会话编号</p> <p><b>filename:</b> 文件名</p> <p><b>mimeType:</b> 文件 MIME type</p> <p><b>url:</b> 离线文件访问地址</p>
<b>sentCallback</b>		<p>对方接收文件时回调，格式如下</p> <p>sentCallback(state, callId, filename)</p>



**参数解析****state:** 状态类型

errCodeMap.OK //服务器离线保存成功

errCodeMap.UPLOAD\_FILE\_REJECT //对方拒绝或取消接收文件

**callId:** 会话编号**filename:** 文件名

返回协议;

参数	默认值（非空）	描述
无		

## 5. 异步事件

### 5.1. 资源加载事件

#### 5.1.1. 加载表情完毕

格式: \$.ebMsg.eventHandle.onLoadEmotions(emotions)

参数	默认值（非空）	描述
emotions		表情数组，格式如下： emotions[0].url: <a href="http://114.215.181.249/servlet.ebwebcm.res?resid=5433726985370040">http://114.215.181.249/servlet.ebwebcm.res?resid=5433726985370040</a> emotions[1].url: <a href="http://114.215.181.249/servlet.ebwebcm.res?resid=5433727080940093">http://114.215.181.249/servlet.ebwebcm.res?resid=5433727080940093</a>

#### 5.1.2. 加载企业架构信息完毕

格式: \$.ebMsg.eventHandle.onLoadEntArchitecture(entArchi)

参数	默认值（非空）	描述
entArchi		企业架构

### 5.1.3. 加载个人群组信息完毕

格式: `$.ebMsg.eventHandle.onLoadGroup(groups)`

参数	默认值（非空）	描述
<code>entArchi</code>		个人群组信息数组

### 5.1.4. 加载个人联系人信息完毕

格式: `$.ebMsg.eventHandle.onLoadContact(contacts)`

参数	默认值（非空）	描述
<code>contacts</code>		个人联系人数组

## 5.2. 会话管理事件

### 5.2.1. 会话结束

格式: `$.ebMsg.eventHandle.onTalkOver(callId)`

参数	默认值（非空）	描述
<code>callId</code>		会话编号

### 5.2.2. 会话编号已变更

格式: `$.ebMsg.eventHandle.onChangeCallId(existCallId, newCallId)`

参数	默认值（非空）	描述
<code>existCallId</code>		原会话编号
<code>newCallId</code>		新会话编号

### 5.2.3. 接收到富文本信息

格式: \$.ebMsg.eventHandle.onReceiveMessage(callInfo, accountInfo, richInfo)

参数	默认值（非空）	描述
<b>callInfo</b>		会话信息，结构如下： call_id: 会话编号 chat_id: 聊天编号 chat_key: 聊天会话密钥 group_code: 群组或部门编号 cm_url: 会话服务地址 cm_appname: 会话服务应用名称 hangup: 会话是否已结束 accounts: 会话中各对话方信息(不包含自己)，数组结构 [[ uid: 用户编号 offline: 是否离线状态 incall: 是否准备好对话环境 ], ...]
<b>accountInfo</b>		对方账号信息，结构如下： uid: 对方用户编号 from_account: 对方账号 fInfo: 电子名片 结构如下 { type: 类型 empCode: 群成员编号 name: 名称 mobile: 手机 telephone: 固定电话 email: 电子邮件 title: 职务 departmentName: 群或部门名称 entpriseName: 公司名称 }
<b>richInfo</b>		富文本信息，结构如下： [[ type: "text" 文本 content: "内容内容内容内容..." ], { type: "screenshot" 截图

```
content:
  "http://abc.com/servlet.ebwebcm.res?file=53951.jpg"
},
{
  type: "emotion" 表情
  content:
    "http://abc.com/servlet.ebwebcm.res?resid=540940"
}}
```

## 5.3. 其它事件

### 5.3.1. 其它未详细定义的错误信息

格式: \$.ebMsg.eventHandle.onError(error)

参数	默认值（非空）	描述
<b>error</b>		错误信息对象，结构如下： {code: 错误代码, msg: 错误描述}

## 6. 附录

### 6.1. 错误信息对照表

```
errCodeMap =
{
  NETWORK_ERROR: {code:-2, msg:"网络错误"},
  UNKNOWN: {code:-1, msg:"未知错误"},
  OK: {code:0, msg:"成功"},

  LOGON_FAILURE: {code:2, msg:"登录失败"},
  ONLINE_FAILURE: {code:3, msg:"设置在线失败"},
  OFFLINE_FAILURE: {code:4, msg:"下线失败"},
  TIMEOUT: {code:5, msg:"超时"},
  PARAMETER_ERROR: {code: 6, msg:"参数错误"},

  CALLACCOUNT_FAILURE: {code:11, msg:"呼叫对方失败"},
  CALLACCOUNT_NOT_EXIST: {code:12, msg:"被叫方不存在"},
  CALLACCOUNT_NO_RESPONSE: {code:13, msg:"对方没有应答"},
}
```

```
CALLACCOUNT_REJECT: {code:14, msg:"对方拒绝通话"},

CHAT_INVALID: {code:21, msg:"聊天会话失效，等待重新载入"},
SESSION_INVALID: {code:22, msg:"当前用户会话失效，等待重登"},
CALLID_INVALID: {code:23, msg:"填入的CallId不正确"},
CALLID_NOT_EXIST: {code:24, msg:"CallId不存在"},
NOBODY_IN_CALL: {code:25, msg:"会话中没有对话方"},
CALL_NOT_READY: {code:26, msg:"会话没有准备好"},
NOT_LOGON: {code:27, msg:"当前用户未登录"},

CONTENT_TOO_LONG: {code: 30, msg:"内容过长"},
SENDFILE_REQUEST_FAILURE: {code:31, msg:"请求发送文件失败"},
UPLOAD_FILE_EMPTY: {code:32, msg:"没有指定上传文件"},
UPLOAD_FILE_REJECT: {code:33, msg:"对方取消或拒绝接收文件"}

//code=10000以上，自定义错误信息
}
```