

发送图片消息

本 SDK 还可以支持发送图片消息，内部封装了图片上传、图片消息发送等逻辑，请按以下步骤进行接入。

一、配置图片上传实例

在 SDK 实例化完毕后，聊天室实例会有 `uploader` 属性（即为图片上传功能的实例）。接入方需提供一个 `file` 类型的 `input` 元素，并在该元素加载完毕后，调用 `uploader.initImgUploader()` 传入 `input` 元素去初始化该图片上传实例。例如：

```
const $imgUploadInput = document.querySelector('#img-upload-input');
// 配置图片上传实例
chat.uploader.initImgUploader($imgUploadInput);
```

图片上传实例配置完毕后，接入方只需要控制该 `input` 元素的展示样式。当观众点击触发该元素并进行图片选择后，SDK 内部会监听到这个动作，并启动图片上传，过程中触发相应事件。

二、监听本地发送图片消息相关事件

(1) `SELF_CHAT_IMG` 事件

即“本地发送图片”事件，当图片开始上传时，该事件会被触发，此时可以将对应的消息插入到本地进行展示。然后监听下列消息，展示上传进度或结果。

(2) `CHAT_IMG_UPLOAD_PROGRESS` 事件

“图片上传进度”事件。指示图片开始上传、上传完毕或上传失败等状态。

当图片上传完成后，SDK 内部会发送对应的图片消息到后端服务，后端将进一步处理。

当监听到 `chat.events.CHAT_IMG_UPLOAD_PROGRESS` 事件中，`eventData.content.status` 与 `$chat.uploader.UploadImgStatus.UPLOAD_FAIL` 匹配时，说明上传过程出错，可尝试通过 `chat.uploader.resend(imgMsgId: String)` 来重新上传发送。

(3) `SELF_CHAT_IMG_RESULT` 事件

“本地发送图片消息结果”事件，指示后端处理图片消息的结果。事件参数的 `result` 字段代表图片是否通过违规鉴定，当且仅当 `Boolean` 值为 `false` 表示不通过，此时可展示对应发送结果(如修改已插入本地的消息)。未通过鉴定的图片将不被广播给其他用户，也无法从历史消息 API 中获取到。

三、可通过编程的方式触发文件选择弹窗

```
chat.uploader.selectFile()
```