

client.messageAction

Description

This event is triggered when a user presses the message action button.

If your app enables this button, it will be provided next to every message. It will also be provided if a user selects multiple messages.

```
client.messageAction Event
{
  "name": "client.messageAction",
  "userId": "u:cfc76545-3400-4864-892a-513a9f4ae409",
  "userName": "Alyssa P. Hacker",
  "chat": "u:1e6429de-16b7-48b3-acc0-c2d7ec5ad195",
  "chatName": "Ben Bitdiddle",
  "messageUids": [ "fd4877b719b1", "59e2da001af2" ]
  "messages": [ { "from": "u:1e6429de-16b7-48b3-acc0-c2d7ec5ad195",
                  "to": "u:cfc76545-3400-4864-892a-513a9f4ae409",
                  "uid": "fd4877b719b1",
                  "text": "Hello",
                  "timestamp": 1481808450532 },
                { "from": "u:cfc76545-3400-4864-892a-513a9f4ae409",
                  "to": "u:1e6429de-16b7-48b3-acc0-c2d7ec5ad195",
                  "uid": "59e2da001af2",
                  "text": "Hello again",
                  "timestamp": 1481808450533 } ]
}
```

In addition to the chat in which the message action button was pressed, Flock also provides a list of messages uids for which the button was pressed. If your app needs more information about the messages, it should call the [chat.fetchMessages](#) method.

Optionally, Flock might provide the entire [Message](#) object alongwith the event itself, so you can skip calling [chat.fetchMessages](#) if the `messages` attribute is present. Note that this attribute is strictly optional however (especially when many messages have been selected). Your app should not depend on the presence of this attribute.

Event Attributes

Attribute Name	Attribute Type	Attribute Description
userId	String	User identifier
userName	String	Name of the user
chat	String	Id of the user or group whose chat tab is open
chatName	String	Name of the user or group whose chat tab is open
messageUids	Array	An array of message uids
messages	Array	An array of Message objects. This is an optional attribute that may not always be present. If present, this allows you to skip calling chat.fetchMessages .

Response

Your app can respond with a `text` value – this should be a short (upto 100 chars) string that is displayed to the user who pressed the button.

```
{  
  "text": "<toaster message>"  
}
```