Genesys Mobile Services API Reference

Chat API Version 1 Push Notification and Background State

5/10/2025

Contents

Chat API Version 1 Push Notification and Background State

Mobile Application Background State Issues

If an iOS or Android mobile application is moved to the background state, the operating system does not close active TCP connections. So, if your app does not handle the foreground and background events, the server-side of your CometD connection isn't notified.

More specifically, if your app handles chat with GMS, the agent may send a new message which would result in sending the long poll response. Here is a list of the various scenarios which can happen according to your implementation.

- The Cometd server attempts to send a long poll response but fails and closes the socket of the current long-poll request.
- The Cometd server sets a 10-second timer; if the app does not reconnect within this period, the app is considered to be gone and its context is destroyed.

Later, if the app returns to the foreground, it does not receive chat notifications from the Cometd server and the chat session appears to be broken.

To avoid these types of issues, your app should listen for events that both iOS and Android submit when your app enters the background (or foreground) state.
[+] Read more about background state
See the Official documentation.

- Apple provides background instructions for iOS, here
- Android provides best background practices here

Then, your app should implement the Cometd Meta Disconnect message detailed below to handle background State issues.

Cometd Meta Disconnect Messages

To handle background state, your app must send Cometd `/meta/disconnect` messages to GMS and later, if your app returns to foreground, it should establish a new Cometd session as detailed below.

1. When your app enters the background state, it sends a Comet`/meta/disconnect` message to GMS which includes the highest transcript position received by your app, as for example:

```
{
    "channel": "/meta/disconnect",
```

```
   "clientId":  "71k6evjfbffqq9eir3jhn6udxz",
   "ext:
   {
        "transcriptPosition": "4"
   },
   "id": "4"
}
```

2. If GMS receives a `/meta/disconnect` message, it flags your app as disconnected and sends a chat `Notice.Custom` event to the agent. The default value is `customer is not online`. You can configure this text. **[+] Tell me how**
   1. Edit your GMS application (with Configuration Manager for example).
   2. Navigate to **Options > Service.<service_name>**.
   3. Edit the `_agent_timeout_notification_message` value.
3. If the agent sends a new message while your app is disconnected, GMS starts a configurable timer. You can set this timer value to zero **[+] Tell me how**
   1. Edit your GMS application (with Configuration Manager for example).
   2. Navigate to **Options > Service.<service_name>**.
   3. Edit the `_client_timeout_notification` value.

   . When the timer expires, GMS sends a push notification (Google FCM, or HTTP). Typically, the notification results in a dialog that allows the user to optionally return the app to the foreground.
4. If the app returns to the foreground, it must start a new Cometd session with GMS. GMS flags the app as connected and sends a Cometd notification that includes all chat transcript events which occurred after the app entered the background state, as for example:

```
[
   {
      "data":{
         "id":"fdf93730c1e411e4a5e8f1be76b28efd",
         "message":
            { "chatSessionId":"0001BaAFC4J8000N", "transcriptPosition":
"6", "chatServiceMessage":"Chat service is available", "startedAt":
"2015-03-03T20:36:12Z", "chatIxnState":"TRANSCRIPT", "transcriptToShow":[
["Notice.TypingStarted","agentName","is typing","28","AGENT"],
["Message.Text","agentName","Hello","29","AGENT"] ] },
         "tag":
"service.chat.refresh.180-655e104c-a6cf-447d-b94b-f132d49a35fe"
      },
      "channel":"/_genesys"
   },
   { "successful":true, "channel":"/meta/connect" }
]
```

Important

- If your app returns to foreground before the timer expires, no push notification is sent, but your app must still start a new Cometd session and it will receive a Cometd notification (which includes all new transcript data) from GMS.

- If your app does not return to foreground after the first push notification is sent, GMS sends additional push notification messages for each new agent event. Note that these events are sent immediately.

Content of Push Notification Messages

GMS triggers push notification messages if the agent submits new chat events to the app flagged as disconnected. The following events can result in a notification:

- Notice.TypingStarted
- Notice.TypingStopped
- Notice.Joined
- Notice.Left
- Notice.PushUrl
- Notice.Custom
- Message.Text

Your application's configuration includes a filter to exclude events from triggering a push notification. The `filtering_chat_events` option in the push section sets the default value for this filter to `Notice.TypingStarted,Notice.TypingStopped`. You can still set a different value for your service. **[+] Tell me how**

1. Edit your GMS application (with Configuration Manager for example).
2. Navigate to **Options > Service.<service_name>**.
3. Edit the `_filtering_chat_events` option to add new event types to exclude.

For the Android and HTTP push notification types, notification messages include the content of the filtered agent events. Currently, this is not supported for iOS push notification messages due to the 256-byte payload size limit for this notification type.

The following example of the push notification message includes the content of the filtered agent events.

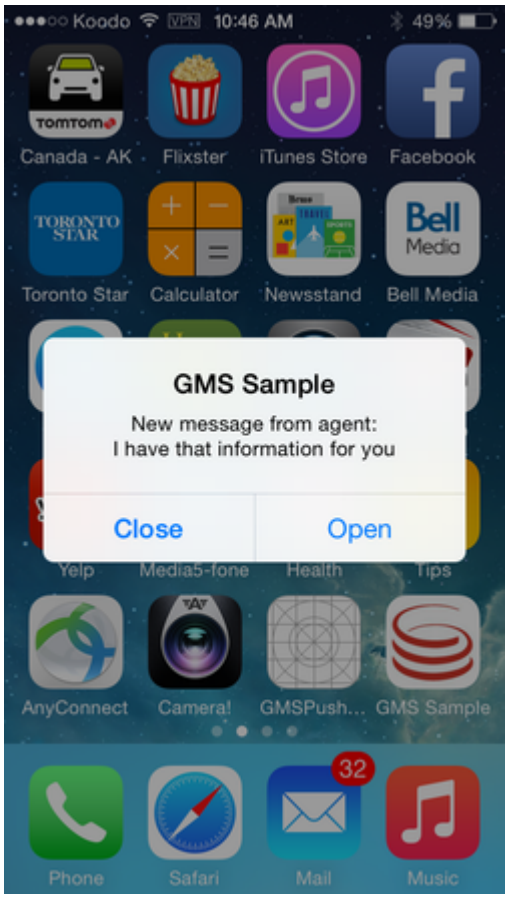
```
"message": {
  "message": "New message from Agent",
  "serviceId": "180-e941460d-1eb0-4f0b-9022-b99ca9ee41f7",
  "lastTranscript": [
     {"Message.Text": "A line of text."}
     {"Message.Text": "Another line of text."}
  ]
}
```

By default, the message attribute (or notification message) is set to `New message from Agent`, but you can change this text in your service options. **[+] Tell me how**

1. Edit your GMS application (with Configuration Manager for example).
2. Navigate to **Options > Service.<service_name>**.
3. Edit the value of the `_client_timeout_notification_message` option.

The notification is tagged as `chat.newagentmessage`. In addition to the notification message, the notification content provides the `lastTranscript` text that can be displayed to the user.

- You can display this the `lastTranscript` text in a popup a or banner notification to the user.



Specific Configuration for Chat Push Notification

To customize your GMS configuration for push notification messages (for both iOS and Android), you can create a `push.provider.event.chat.newagentmessage` section in the **Options** tab (with Configuration Manager for instance). You can add there any of your Android or iOS options.

- This additional configuration is not mandatory.

> **Tip**
>
> - Read more about options of the GMS push section here.
> - Read more about the OS specific capabilities associated with the notification message here.