

# **Chat Download Product Documentation**



## Copyright Notice

©2013-2024 Tencent Cloud. All rights reserved.

Copyright in this document is exclusively owned by Tencent Cloud. You must not reproduce, modify, copy or distribute in any way, in whole or in part, the contents of this document without Tencent Cloud's the prior written consent.

## Trademark Notice



All trademarks associated with Tencent Cloud and its services are owned by Tencent Cloud Computing (Beijing) Company Limited and its affiliated companies. Trademarks of third parties referred to in this document are owned by their respective proprietors.

## Service Statement

This document is intended to provide users with general information about Tencent Cloud's products and services only and does not form part of Tencent Cloud's terms and conditions. Tencent Cloud's products or services are subject to change. Specific products and services and the standards applicable to them are exclusively provided for in Tencent Cloud's applicable terms and conditions.

# Contents

## Download

[SDK and Demo Source Code](#)

## Update Log

[Web](#)

[Native](#)

[Flutter](#)

[Electron](#)

[Unity](#)

[React Native](#)

# Download

## SDK and Demo Source Code

Last updated : 2024-04-11 09:39:54

# Update Log

## Web

Last updated : 2024-05-09 14:53:09

### 3.0.0 @2023.6.25

#### New features

A better, safer, more powerful, and more flexible SDK built with new architecture. See [@tencentcloud/chat](https://tencentcloud.com/chat).

### 2.25.0 @2022.12.8

#### New features

Added [clearHistoryMessage](#) API to clear messages from local storage and the cloud.

Supported message extension (Premium edition feature).

Supported ordinary group and community group attributes.

Supported compatibility with [wx.chooseMedia](#)

Supported one-to-one message read receipts, whose data structure is aligned with that of native Chat SDK, in [Message.readReceiptInfo](#).

Added the 2101 error code: Users who are not members of the audio-video group cannot send messages to the audio-video group.

#### Changes

The log reporting backup channel uses a dedicated cluster domain name `https://events.im.qqcloud.com` (a trusted domain configuration must be added to the platform).

#### Bug fixing

Fixed the runtime error (Failed to read the 'localStorage' property from 'Window': Access is denied for this document) caused by cookies blocked.

### 2.24.1 @2022.11.11

#### New features

Added the English version of the declaration TS file.

RESTful APIs support pushing custom profile field modifications to the SDK.

#### Bug fixing

Fixed the abnormal results returned in certain scenarios for [getMessageListHopping](#).

### 2.24.0 @2022.11.3

#### New features

Supported mini game environment integration.

Added the local audit feature to the local audit plugin [tim-profanity-filter-plugin](#).

[getFriendProfile](#): added support for pulling custom friend and profile fields by default for better user experience.

[getGroupApplicationList](#): added support for pulling the list of all group joining requests.

RESTful APIs support pushing custom field modifications to the SDK.

Supported sending topic messages that are excluded from the unread count.

Supported sending common community messages that are excluded from the unread count.

Message sending supports VoIP Push.

### Bug fixing

Fixed friend profile related issues.

## 2.23.1 @2022.9.29

### New features

[createTextMessage](#): Supported creating a targeted group message, which is sent to specified members in a group and cannot be received by other members in the group.

Supported sending videos in MOV format.

RESTful API [Updating Friends](#): Supported pushing to SDK.

[getFriendProfile](#): Supported pulling custom friend and profile fields.

[getConversationList](#): Added the `isSyncCompleted` field to the return data to indicate whether synchronizing the conversation list from the cloud is completed.

The receiving of a message from the community to which a topic belongs can be notified to the access side through the [MESSAGE\\_RECEIVED](#) event.

### Bug fixing

Fixed the issue where roaming messages could not be pulled for some group conversations when the number of groups in the group list exceeded the upper limit of 5,000.

Fixed the issue where, when `setConversationCustomData` was called to set custom fields for a conversation, `customData` of the conversation was "" after the user logged in again.

## 2.23.0 @2022.9.16

### New features

The SDK supports environments outside the Chinese mainland.

Added [getTotalUnreadMessageCount](#) API to get the total unread message count of a conversation.

Added the [TOTAL\\_UNREAD\\_MESSAGE\\_COUNT\\_UPDATED](#) event. By listening for this event, the access side can receive notifications of a change in the total unread count.

Added [markGroupMemberList](#) API to mark a member of an audio-video group (Premium edition required).

When a member is removed from a group, or a group is deleted, the SDK updates the conversation group to which the group conversation belongs at the same time.

Supported independent subpackaging.

Web: When an account logged in on multiple instances, the SDK proactively recovers the message history of the most recent contacts after network reconnection to ensure message reliability.

### Bug fixing

Fixed the out-of-sync issue of the conversation `lastMessage` recall state that may occur in multi-instance login web scenarios.

Fixed the issue of pinning conversations to the top when recent contacts were synchronized.

## 2.22.0 @2022.8.18

### New features

Supported packaging the uni-app into the native app for offline push. For details, see [registerPlugin](#).

Supported getting the list of online members of an audio-video group. For details, see [getGroupMemberList](#) (Premium edition required).

Supported blocking a member of an audio-video group. For details, see [deleteGroupMember](#) (Premium edition required).

Added [setConversationCustomData](#) to set custom conversation data.

Added [markConversation](#) to mark a conversation (Premium edition required).

Added [getConversationGroupList](#) to get the list of conversation groups (Premium edition required).

Added [createConversationGroup](#) to create a conversation group (Premium edition required).

Added [deleteConversationGroup](#) to delete a conversation group (Premium edition required).

Added [renameConversationGroup](#) to rename a conversation group (Premium edition required).

Added [addConversationsToGroup](#) to add a conversation to a conversation group (Premium edition required).

Added [deleteConversationsFromGroup](#) to delete a conversation from a conversation group (Premium edition required).

### Bug fixing

Fixed the issue where the unread count of a topic was not updated after a topic message recall notification was received.

## 2.21.2 @2022.8.8

### New features

Supported creating and sending audio messages on web clients.

Added the ID field to messages combined to the [createMergerMessage](#) API for creating a combined message.

## 2.21.1 @2022.8.3

### Bug fixing

Fixed the message duplication problem that [resendMessage](#) can cause.

## 2.21.0 @2022.7.28

### New features

Added [setSelfStatus](#) to set one's own custom status.

Added [getUserStatus](#) to query a user's status.

Added [subscribeUserStatus](#) to subscribe to a user's status.

Added [unsubscribeUserStatus](#) to unsubscribe from a user's status.

Added a feature of [setMessageRemindType](#): Synchronizing the settings of group and topic message muting across clients and instances.

Added a feature of [createFileMessage](#): Sending a file message.

Added a feature of [modifyMessage](#): Modifying `cloudCustomData` for messages of all types.

Added the `isBroadcastMessage` field to [Message](#) to support broadcast messages for audio-video groups.

Supported synchronizing group joining options across terminals and instances.

Supported @ all members for an ordinary community and its topics and `lastMessage` for topics.

### Changes

Webworker is enabled by default in the international website and private environment when the browser supports webworker.

### Bug fixing

Fixed the issue where `lastMessage.payload` was set to `undefined` when receiving a message without updating the conversation's `lastMessage`.

Fixed the compensation for group messages caused by online messages did not start.

Fixed the group roaming message pulling exception that occurred after a user frequently left a group and joined the group again.

Fixed the issue where the lag in pulling the group list by page caused the result of pulling group conversation roaming messages to be an empty array.

Fixed known topic issues.

## 2.20.1 @2022.6.27

### Changes

Aligned with the native SDK experience, where only group records are deleted and group conversations are not deleted after users leave or are kicked out of a non-audio-video group or the group is deleted.

Made [deleteMessage](#) unable to delete group system notifications; if a deletion attempt is made, an error message will be reported.

Supported HTTP for rich media messages of the on-premises deployment.

### Bug fixing

Fixed the issue where group conversations were occasionally lost during the switch between the foreground and background.

Fixed the issue where the `lastMessage` of a one-to-one conversation was abnormally updated.

## 2.20.0 @2022.6.9

### New features

Added [modifyMessage](#) to modify a message.

Added [getMessageListHopping](#) to pull the conversation message list by specified sequence or time range.



Supported read receipts for one or more one-to-one messages (Premium edition required).

Added the `isPeerRead` field for `lastMessage` of a one-to-one conversation to indicate whether a message was read by the receiver.

Excluded group tips from the unread count of a conversation.

Added `TIM.TYPES.KICKED_OUT_REST_API` to support the RESTful API [Invalidating Account Login States](#).

### Changes

Optimized [getMessageList](#) to pull the roaming messages.

### Bug fixing

Fixed the issue where the conversation list was not updated due to parameter issues after the [deleteMessage](#) API was called successfully.

Fixed the issue where `Cannot add property markTimeline, Object is not extensible` occurred during debugging on real devices of some models.

## 2.19.1 @2022.5.7

### New features

Supported topic creation in a [community](#) for stronger interactions.

Added [getJoinedCommunityList](#) to get the list of topic-enabled communities.

Added [createTopicInCommunity](#) to create a topic.

Added [deleteTopicFromCommunity](#) to delete a topic.

Added [updateTopicProfile](#) to set the topic profile.

Added [getTopicList](#) to get the topic list.

Added the `TIM.EVENT.TOPIC_CREATED` event, which will be triggered when a topic is created.

Added the `TIM.EVENT.TOPIC_DELETED` event, which will be triggered when a topic is deleted.

Added the `TIM.EVENT.TOPIC_UPDATED` event, which will be triggered when the topic profile is updated.

## 2.18.2 @2022.4.22

### Changes

Optimized the audio-video group user experience.

### Bug fixing

Fixed the issue where the statistics in certain use cases were inaccurate.

Fixed the issue where the result returned by the [getGroupMessageReadMemberList](#) API was inaccurate.

## 2.18.0 @2022.4.8

### New features

Added [sendMessageReadReceipt](#) for sending group message read receipts.

Added [getMessageReadReceiptList](#) for pulling the list of group message read receipts.

Added [getGroupMessageReadMemberList](#) for pulling the list of members who have (or have not) read a group message.

Added [findMessage](#) for querying local messages in a specified conversation by messageId.

Aligned with the native Chat experience of the conversation unread count change after a message is recalled.

### Changes

The rule for concatenating the [message ID](#) is `${senderTinyID}-${clientTime}-${random}`, which is the same as that for concatenating the message ID of native Chat.

When the SDK is in the not ready state, specific reasons are provided for the access side.

### Bug fixing

After a group member was removed from a group, the `Conversation.groupProfile.memberCount` value that other group members obtained from the [CONVERSATION\\_LIST\\_UPDATED](#) event callback was not updated.

## 2.17.0 @2022.3.2

### New features

Supports [Community groups](#).

Supports group notifications for recent contacts' `Conversation.lastMessage`.

`Message.payload.memberList` supports getting the nickname, profile photo, and other information of group members who joined or left a group.

Supports WEBP images for image message sending.

Supports video cover [snapshotUrl](#) for video message sending.

Improved message transmission efficiency and reduced events such as [CONVERSATION\\_LIST\\_UPDATED](#).

### Bug fixing

After a user sent a message with custom data (cloudCustomData), `cloudCustomData` was empty when the user logged in again.

When a user logged in again after a [login](#) failure, the SDK reported the error of repeated login.

After [getGroupProfile](#) was called, `Conversation.groupProfile` was inconsistent with the latest group profile.

## 2.16.3 @2022.2.11

### Bug fixing

Fixed login failures that occurred when a Windows client accessed after a packaged Android application (on some devices).

## 2.16.2 @2022.2.10

### New features

Supports sending file messages after uni-app packages native apps.

Supports the international website in India.

### Bug fixing

Fixed some emoji rendering issues.

## 2.16.1 @2022.1.14

## New features

Added support for Alipay to send .image images.

When `deleteConversation` is called to delete a conversation, historical messages in the conversation are deleted as well.

## Bug fixing

An error occurred when the downstream file message `fileName` was an empty string.

Fixed the issue caused by the group attribute API call sequence.

The `__wxConfig is not defined` issue occurred when uni-app packaged apps to Baidu and other platforms.

## 2.16.0 @2022.1.5

### New features

`setMessageRemindType` supports setting the **Mute Notifications** mode for C2C conversations.

`setAllMessageRead` supports quickly marking unread messages of all conversations as read.

`sendMessage` supports excluding sent messages from the conversation's unread message count and not updating the conversation's `lastMessage`.

Allows new members of an audio-video group to view historical messages before joining the group (you must activate the Premium edition package to use the feature).

### Changes

The SDK uses the `strict mode`.

The conversations with deleted accounts are filtered out for the conversation list.

Optimized the update timing of `nick` and `avatar` for roaming messages.

When receiving peer (friend) profile update information, the SDK updates `conversation.userProfile` accordingly.

### Bug fixing

WebSocket persistent connections were disconnected abnormally due to non-UTF-8 characters.

The runtime error `e.getOnlineOnlyFlag is not a function` occurred when a copied message was passed in to call `deleteMessage`.

After `deleteMessage` was called, the `lastMessage` of the corresponding conversation was not correctly updated.

An error occurred when calculating the unread message count of one-to-one conversations.

Rendering exceptions occurred if `nick` and `avatar` were not carried in real-time messages of one-to-one conversations.

`lastMessage.payload` was occasionally `null` for conversations.

Pre-signed image thumbnail upload URLs did not take effect.

When a user mentioned (@) a group member and logged in again to pull roaming messages, the corresponding `message.atUserList` was an empty array.

An error occurred when group notifications (group owner changing) are processed.

Some statistics errors.

## 2.15.0 @2021.10.29

### New features

Supports the international website.

[createLocationMessage](#) supports sending geographical location messages.

Supports uploading images, videos, documents for easy download and preview (compatible with uniapp).

Added the `nick` and `nameCard` parameters to the `lastMessage` data structure of [Conversation](#) to better display the sender's information of the `lastMessage` in a group chat.

### Changes

[getConversationList](#) supports getting multiple specified conversations at a time.

Increases the stability of persistent connections.

### Bug fixing

The [CONVERSATION\\_LIST\\_UPDATED](#) event was not sent after login when there was no conversation list cache or no pagination for recent contacts.

In some scenarios, the `isCompleted` parameter was always `false` in the response to the call of the [getMessageList](#) API.

The `index` parameter was missing on the recipient side when `index` was set to `0` in the call of the [createFaceMessage](#) API.

## 2.14.0 @2021.9.24

### New features

[pinConversation](#) supports pinning a conversation to the top.

[initGroupAttributes](#) and other group attribute related APIs support seat management for audio chat rooms.

### Changes

When a group message is sent, the SDK automatically adds the `nameCard` attribute to the message body to facilitate display on the access side.

Forced logout due to multi-client login or multi-instance login no longer triggers server-side logout callbacks.

### Bug fixing

When roaming messages were pulled in one-to-one conversations, messages occasionally got lost.

Group joining remarks (`applyMessage`) were missing.

## 2.13.1 @2021.8.27

### Changes

When a user consecutively calls the [login](#) API before login, error code `2025` is returned, indicating repeated login.

After WebSocket reconnection, the SDK logs in the user again and synchronizes unread messages to ensure message reliability.

### Bug fixing

When a user consecutively called the [login](#) API before login, the unread message count of the conversation was incorrect.

If `nameCard` passed in an empty string when the [setGroupMemberNameCard](#) API was called, the SDK reported an error.

When the [getGroupMemberList](#) API was called, the value of `muteUntil` in the packet returned was incorrect.

### 2.13.0 @2021.8.23

#### New features

Supports friend relationship chain. For more information, see [Usage Guide](#).

#### Bug fixing

An error was occasionally reported when WebSocket persistent connections were disconnected.

### 2.12.2 @2021.8.6

#### New features

Supported video upload progress callback.

#### Changes

The unread message count of a conversation no longer includes the group notifications about not saving the modifications on custom group fields to the roaming server.

#### Bug fixing

Users in an audio-video group occasionally failed to receive group notifications on the group joining by themselves.

When a user used a RESTful API to send C2C messages with `random` being set to `0`, the receiver triggered the [MESSAGE\\_RECEIVED](#) event twice.

### 2.12.1 @2021.7.20

#### New features

Supports counting unread messages in meeting groups.

The [TIM.EVENT.MESSAGE\\_MODIFIED](#) event is added. When a third-party calls back a modified message, the SDK uses this event to notify the message sender of the message modification.

#### Bug fixing

Fixed the issue where group roaming messages occasionally get lost when they are pulled.

Fixed the `xx.toFixed is not a function` issue that may occur during uni-app integration.

### 2.12.0 @2021.7.5

#### New features

[deleteMessage](#) supports deleting messages.

During conversation list synchronization, `lastMessage` can be set to a recalled message.

[getGroupMemberList](#) supports pulling the group joining time `joinTime`.

### Bug fixing

The `nick` value is incorrect in the notifications sent when a user is set or canceled as the admin.

## 2.11.2 @2021.6.16

### New features

Supports WebSocket. [WebSocket Upgrade Guide](#)

Allows uni-app to send image, video, and other file messages.

## 2.10.2 @2021.4.27

### New features

The custom field `cloudCustomData` can be set during message creation to meet diverse business needs.

When [createGroup](#) or [addGroupMember](#) is called, if a single user exceeds the maximum number of groups a single user can join, use `overLimitUserIDList` to notify the access side.

### Bug fixing

After an audio-video group (AVChatRoom) was created in the [console](#) and a group owner was specified, messages sent by the RESTful API for [Sending System Messages in a Group](#) would be repeated on the group owner side after the group owner joined the group.

Nickname was missing when [createForwardMessage](#) was called.

Occasional errors occurred when [downloadMergerMessage](#) was called.

## 2.10.1 @2021.3.19

### New features

The [createMergerMessage](#) API for creating combined messages.

The [createForwardMessage](#) API for creating forward messages.

When an account logs in on multiple instances or clients, once conversation read is reported on one instance or client, the unread count of the conversation will be synchronously cleared on the web client.

### Changes

The MTA statistics feature is deprecated.

### Bug fixing

Web: when an account logged in on multiple instances, the profile photo and nickname of the other party in a one-to-one conversation were incorrect.

When you called back and called the RESTful API to recall messages frequently after sending messages, some of them were not recalled correctly.

## 2.9.3 @2021.2.3

### Changes

If a user hasn't joined a group (not an audio-video group), calling [quitGroup](#) will return error code 2623, indicating that the user is not in the group.

### Bug fixing

`avatar` (profile photo) or `nick` (nickname) was inconsistent in the one-to-one conversation message list.

## 2.9.2 @2021.1.26

### New features

Supports sending and receiving one-to-one messages with `avatar` (profile photo) and `nick` (nickname) displayed.

Supports the Tencent Cloud Chat upload plugin [tim-upload-plugin](#). This plugin enables more secure file upload, supports web, Baidu, Toutiao, and Alipay platforms, and is merely 26 KB. For more information, see [registerPlugin](#).

### Bug fixing

When a user joined an audio-video group anonymously after logging out, the error code 70402 was returned in the response packet during a long polling.

The browser environment was misjudged during Taro 3.0+ integration.

When the image type and size verification failed, there were errors in the returned data structure.

## 2.9.1 @2020.12.23

### Bug fixing

A compilation error occurred when [tim-wx-sdk.js](#) was imported into the basic library 2.14.1 of Developer Tools.

## 2.9.0 @2020.12.15

### New features

The [createTextAtMessage](#) API allows users to specify @ a specific member or @ all members during a group chat. [Message](#) adds the `namecard` attribute to display group members' group name cards (i.e., their nicknames in a group).

## 2.8.5 @2020.11.23

### Changes

The [logout](#) API can be called when the SDK is not ready.

### Bug fixing

Errors occurred in SDK operations when read receipts and read notifications existed at the same time.

Attempts to anonymously re-join an audio-video group after logout failed.

The group list was cleared abnormally.

## 2.8.4 @2020.11.4

### New features

The Baidu, Toutiao, and Alipay platforms are supported (currently on the Baidu, Toutiao, and Alipay platforms, image, video, or file messages, or other messages that need to be uploaded to COS, cannot be sent).

The third-party frameworks of MPX and uni-app are supported.

## 2.8.1 @2020.10.29

### New features

Images in BMP format can be sent.

### Changes

`unreadCount` and `lastMessage` of the [conversation object](#) are not updated when the sender sends an online message and the recipient receives the online message.

### Bug fixing

The SDK could not enter the ready state due to problems synchronizing the list of recent contacts.

## 2.8.0 @2020.10.20

### New features

[getGroupOnlineMemberCount](#) supports querying the number of online users in an audio-video group.

Supports image compression. The access side can choose to display the original image or thumbnail based on business requirements. For more information, see [ImagePayload](#).

### Bug fixing

Compatibility issues when Taro 3.x integrates WebIM

### Changes

SDK size reduction. The size of [tim-js-sdk](#) is reduced by 8.5%, and that of [tim-wx-sdk](#) is reduced by 15%.

## 2.7.8 @2020.9.24

### New features

The [TIM.create](#) API adds the `oversea` parameter. When this parameter is set to `true`, the SDK uses a domain name outside the Chinese mainland to avoid interference.

### Bug fixing

The return value for calling relevant APIs was `undefined` when the SDK was in the `not ready` state.

Issues related to statistics

## 2.7.7 @2020.8.12

### New features

The [TIM.EVENT.SDK\\_RELOAD](#) event was added.

### Bug fixing

Audio-video groups occasionally failed to pull messages in cases where the network was reconnected after a long disconnection.

The type and value of `imageFormat` of an image message were inconsistent with those of the actual image.

The nicknames displayed in work groups and public groups were incorrect.

## 2.7.6 @2020.7.9



**Bug fixing**

Messages occasionally failed to be pulled if an audio-video group (AVChatRoom) was used for a long time.

**2.7.5 @2020.7.2****Bug fixing**

After the RESTful API for [creating a work group](#) was called to create a work group successfully and the group members were specified, messages from group members would fail to be sent.

**2.7.2 @2020.6.30****Bug fixing**

Occasionally, when [joinGroup](#) was called, the SDK prompted "Already in the group" but in fact the user was not in the group. Consequently, the user could not send or receive messages.

The count of messages sent in a temporary meeting group was incorrect.

**2.7.0 @2020.6.8****New features**

Supports one-to-one message read receipts (indicating whether the peer has read your messages). For more information, see the event [TIM.EVENT.MESSAGE\\_READ\\_BY\\_PEER](#). In a [message](#) that has already been read by the peer, the value of `isPeerRead` is `true`.

**Bug fixing**

After a user joined a chat room (ChatRoom), the newly created conversation did not display the last message.

After login, a user who had not joined an audio-video group (AVChatRoom) could still send a message to the audio-video group (AVChatRoom).

**2.6.6 @2020.5.27****Bug fixing**

In audio-video groups (AVChatRoom), messages were occasionally repeatedly displayed on the screen.

An error was reported when [getMessageList](#) received an empty message.

If [login](#) was called again after [logout](#), error `70001` occasionally occurred when [joinGroup](#) was called.

**2.6.4 @2020.5.8****New features**

The [sendMessage](#) API added the sending option to support the sending of online messages (no offline or roaming messages; cannot be used for AVChatRoom or BChatRoom) and the configuration of [offline push](#).

**2.6.3 @2020.4.26****Bug fixing**

Message content was lost because the input `payload.data` `payload.extension` type of `createCustomMessage` is incorrect.

Multiple messages contained in a response to a single request were disordered.

The unread count could not be cleared occasionally after the read count is reported because the number of unread one-to-one conversions overflows.

`TIM.EVENT.ERROR` `event.data.code` and `event.data.undefined` were undefined occasionally.

## 2.6.2 @2020.4.16

### New features

`updateGroupProfile` supports muting and unmuting all.

`getGroupMemberList` supports getting the group member muting deadline timestamp `muteUntil`.

### Bug fixing

The unread count could not be cleared when the latest group message was a group prompt.

## 2.6.1 @2020.4.8

### Bug fixing

Files could not be uploaded occasionally when the uploaded COS signature was invalid and not updated in a timely manner.

## 2.6.0 @2020.3.30

### New features

The web client supports creating and sending video messages of up to 100 MB by calling `createVideoMessage`.

The `nick` and `avatar` attributes are added to `Message` to display the nickname and profile photo address of the message sender in an audio-video group (AVChatRoom). You need to set the nickname and profile photo address in advance by calling `updateMyProfile`.

Web: when an account logs in on multiple instances, the one-to-one message recall notification can be synchronized across these instances.

After `updateGroupProfile` is called to successfully modify custom group fields, group members can receive group notifications and obtain related content `Message.payload.newGroupProfile.groupCustomField`.

### Changes

`TIM.EVENT.GROUP_SYSTEM_NOTICE_RECEIVED` is deprecated and replaced by `MESSAGE_RECEIVED`.

### Bug fixing

Errors occurred occasionally when the `getGroupList` API was called.

## 2.5.2 @2020.3.13

### Changes

When `searchGroupByID` fails, the log level is degraded to warning and the prompt text is modified.

### Bug fixing

Anonymous users or visitors failed to join [TIM.TYPES.GRP\\_AVCHATROOM](#) groups and had statistical problems.  
Other known issues

### 2.5.1 @2020.3.5

#### Changes

When [login](#) is successful, the key-value pair `repeatLogin: true` is added for the `imResponse.data` callback object to identify repeated login of a login account.

#### Bug fixing

The priority of messages received at the receiver side of an audio-video group is different from that set on the sender side.

### 2.5.0 @2020.2.28

#### New features

The network status change event [TIM.EVENT.NET\\_STATE\\_CHANGE](#) is added, and the access side can make related prompts and guidance based on this event.

**Changes**[Error codes](#) are reduced and optimized.

#### Bug fixing

After an audio-video group was created in the [console](#) and a group owner was specified, messages sent by other group members will be repeated on the group owner side after the group owner joins the group.

When groups were created and terminated in the [console](#) or using a RESTful API frequently, the SDK did not deliver the [TIM.EVENT.GROUP\\_SYSTEM\\_NOTICE\\_RECEIVED](#) event.

[getMessageList](#) failed to get the group message list occasionally.

### 2.4.2 @2020.2.7

**New features**[Message priorities](#), [enumerated values](#), and [use cases](#) can be set for group messages.

### 2.4.1 @2020.1.14

#### Changes

Anonymous users or visitors can only join [TIM.TYPES.GRP\\_AVCHATROOM](#) groups.

#### Bug fixing

Some online messages could not be pulled occasionally.

After a system notification from an audio-video group was received, the [TIM.EVENT.MESSAGE\\_RECEIVED](#) event was not delivered.

In some scenarios, the group message recall result was inaccurate.

Other known issues

### 2.4.0 @2020.1.3

#### New features

The [revokeMessage](#) API is added.

The `isRevoked` attribute is added to [Message](#). The attribute value `true` identifies recalled messages.

The message recall event notification [TIM.EVENT.MESSAGE\\_REVOKED](#) is added.

[Forced logout due to multi-client login](#) and [forces logout due to UserSig expiration](#) are added to the forced-logout event notification [TIM.EVENT.KICKED\\_OUT](#).

### Changes

The maximum size of files uploaded through [createFileMessage](#) is increased from 20 MB to 100 MB.

`msgMemberInfo` and `shutupTime` of [group prompts](#) will be deprecated. Use `memberList` and `muteTime` instead.

The [Chat smart customer service entry](#) is added to the console.

### Bug fixing

Listening events could not be canceled by calling the [off](#) API.

The value and type of the `isRead` attribute in [Message](#) were incorrect.

The error code and error message were incorrect when the video file in a sent video message exceeded the maximum size.

The content of updated custom fields was incorrect occasionally.

The [JOIN\\_STATUS\\_ALREADY\\_IN\\_GROUP](#) event occurred occasionally when a user logged in and joined an audio-video group.

core-js caused potential performance issues.

## 2.3.2 @2019.12.18

**Changes** [getUserProfile](#) and [updateMyProfile](#) support [custom profile fields](#).

### Bug fixing

Messages were lost in combined messages obtained using [getMessageList](#).

## 2.3.1 @2019.12.13

### New features

[createImageMessage](#) and [createFileMessage](#) support passing in [File](#) objects.

The [createFaceMessage](#) API is added to create emoji messages.

The message notification efficiency for [TIM.TYPES.GRP\\_AVCHATROOM](#) groups is optimized to improve the user experience.

### Changes

When messages fail to be sent, the SDK returns the actual error codes and error messages.

When [logout](#) is called, only the message channel of the current instance logs out.

When a callback function passed in by the access side is encapsulated for security purposes and the logic of the callback function is incorrect, errors can be captured and located quickly.

The SDK provides Chinese error information when [Chat server-side error codes](#) are received.

### Bug fixing

[TIM.EVENT.CONVERSATION\\_LIST\\_UPDATED](#) was triggered several times when a message was sent.

The SDK reported errors when files, such as images were uploaded if [registerPlugin](#) was not called or incorrect parameters were entered.

Long polling did not stop after a [TIM.TYPES.GRP\\_AVCHATROOM](#) group was disbanded.

When "multi-instance" or "multi-client" login was enabled, other instances or clients failed to receive messages after a web instance was logged out.

The SDK reported errors occasionally due to the structure of session lists that were pulled.

### 2.2.1 @2019.11.28

#### Changes

The logic for getting group roaming messages is optimized.

#### Bug fixing

The SDK reported [error code 2901](#) after the group owner of an audio-video group modified the group profile.

After the group admin processed apps for joining a group, processed apps can be received after refresh.

### 2.2.0 @2019.11.21

#### New features

Support creating and sending video messages via the [createVideoMessage](#) API. Video messages can be synced across platforms. You need to update to the latest versions of the [TUIKit](#) and [SDK](#).

The [getGroupMemberProfile](#) API is added.

Compatible with audio and file messages sent by Native Chat SDK v3.x.

Location messages [GeoPayload](#) can be received.

#### Changes

Up to 100 groups can be written to local storage. The SDK does not write the full group list when there are more than 100 groups.

#### Bug fixing

Long polling of [TIM.TYPES.GRP\\_AVCHATROOM](#) groups continues after logout.

The group contact cards in message instances of [TIM.TYPES.GRP\\_AVCHATROOM](#) groups did not have values.

Errors were reported when Internet Explorer 10 was used.

Users could not join groups anonymously.

### 2.1.4 @2019.11.7

#### Changes

When the `Promise` status returned by an SDK API is `rejected`, the SDK no longer delivers a [TIM.EVENT.ERROR](#) event.

Updates to a user's profile are immediately written to the local cache.

#### Bug fixing

Code running failed after SDK integration when Angular zone.js modified prototype chains.

After a group owner created and joined a [TIM.TYPES.GRP\\_AVCHATROOM](#) group, the group owner could not receive messages.

Initialization failed when the group list was excessively large.

### 2.1.3 @2019.10.31

#### Changes

Combined messages (multiple message elements in one message) sent via RESTful API calls or the legacy Chat version are compatible. For more information, see [Compatibility Guide](#).

#### Bug fixing

The unread count was inaccurate.

Messages were disordered because read messages were not reported.

Empty image messages were sent successfully but could not be rendered. The SDK did not support sending empty image messages.

Empty file messages were sent with incorrect message status. The SDK did not support sending empty file messages.

SDK code errors were reported occasionally when [getGroupMemberList](#) was called.

### 2.1.2 @2019.10.25

**New features**[getGroupList](#) supports pulling group profile information, including the group owner ID and group member count.

#### Bug fixing

SDK code errors were reported when a RESTful API is used to send custom group notifications in an audio-video chat room.

The SDK did not send a request to pull historical messages when a user re-joined a left group and called the

`getMessageList` API.

SDK code errors were reported when upload failed.

### 2.1.1 @2019.10.18

#### New features

Support [sending audio messages](#). Audio messages can be synced across platforms. You need to update to the latest versions of the [TUIKit and SDK](#).

**Bug fixing**[getMessageList](#) could still pull historical messages in a quit group after rejoining.

### 2.1.0 @2019.10.16

#### New features

Web supported receiving [audio messages](#).

Web supported receiving [video messages](#).

#### Changes

The [getMessageList](#) API can pull up to 15 messages at a time.

[TIM.TYPES.MSG\\_SOUND](#) is deprecated and replaced by [TIM.TYPES.MSG\\_AUDIO](#).

#### Bug fixing

[getMessageList](#) could not pull messages in deleted group chats.

Group system notifications did not contain group names.

When a conversation was created after receiving a new message, the conversation did not have the profile of the message sender.

### 2.0.11 @2019.10.12

#### Bug fixing

Image messages failed to be sent under the React framework.

### 2.0.9 @2019.9.19

#### New features

The actual width and height of an image are detected before the image message is sent.

#### Changes

The HTTPS protocol is used by default.

[TIM.EVENT.GROUP\\_SYSTEM\\_NOTICE\\_RECEIVED](#) events are sent when new group system notifications are received.

#### Bug fixing

Flickering screen occurred after an image message is sent.

JPG or other images failed to be sent.

# Native

Last updated : 2024-08-20 14:17:31

## 8.1.6116 @2024.08.14 - Enhanced Version

### SDK

Signaling now supports disabling callbacks before and after message sending.

Fixed an issue where after deleting a pinned conversation, new messages could not retrieve the conversation.

Fixed an occasional issue where after deleting a conversation and receiving a conversation deletion callback, there would still be occasional conversation update callbacks.

Fixed occasional crash issues.

### TUIKit & Demo

Updated customer service plugin agreement.

Fixed a crash issue on the iOS side of the group note plugin.

## 8.1.6103 @2024.07.26 - Enhanced Version

### SDK

Long connection supports HTTP protocol, enhancing network penetration capabilities.

Topic message reception options now support setting whether to follow the community.

Topics add a new message reception option to only receive @ messages.

Read receipts for private chat messages now include the read time.

Support disabling cloud callbacks before and after message sending.

Fixed an issue where the conversation list did not update in real-time when group - avatars or names were changed.

Fixed an issue where pulling the grouped conversation list was abnormal when logging in without a network connection.

Optimized Android HTTPS security issues.

Optimized rich media file download authentication.

Enhanced database performance for Android.

### TUIKit & Demo

TPush standalone push product launched with a smaller package size.

Push adds a login-free push feature.

Push introduces a new intelligent detection strategy for available channels.



Push adds a push registration timeout protection mechanism.

TUIChat optimizes the UI display of pinned messages and security prompts.

TUIChat international version UI supports "User is typing" status.

Optimized the UI experience for message sending and receiving, reducing lag issues.

## 8.0.5895 @2024.06.07 - Enhanced Version

### SDK

Added support for offline push settings with a large icon on the right side.

Added support for returning operator information to pinned messages.

Added support for returning group pinned messages that were deleted by oneself.

Added support for preserving conversation grouping information when deleting conversations.

Added system notifications for when conversation grouping information is eliminated by the backend.

Local message search now supports searching all single or group chat messages.

Completed system notifications for friend group changes.

The entry tips message and callback in a regular group can distinguish between active joining and being invited into the group.

Improved database query throughput and speed.

Optimized read receipt performance by merging duplicate requests.

Optimized the time range for group signaling synchronization after login.

Fixed an issue with pulling messages after inserting local messages in a topic.

### TUIKit & Demo

TUIKit now supports message pinning.

TUIKit supports asynchronous striking of file messages.

TUIKit allows for entering the main interface to view local data without logging in after a network disconnection.

TUIKit no longer exposes the original message content when referencing and replying to a retracted message.

TUIKit replaces libopencore with AAC+M4A.

TUIChat Android supports saving images without extensions to the gallery.

## 7.9.5680 @2024.04.19 - Enhanced Version

### SDK

Fixed the issue of the pinned message list returning in the wrong order.

Fixed the issue of incorrect parsing of the Tips type in pinned messages.

Fixed the issue of log writing failure on some Android phones.

Fixed the issue of occasionally incomplete pulling of group roaming messages from old to new.

Fixed the issue of occasionally failing to pull local messages when pulling historical messages from topics.

Fixed the issue where sessions deleted in a conversation group are reactivated after logging in again.

## 7.9.5666 @2024.04.07 - Enhanced Version

### SDK

New visionOS SDK, compatible with Apple Vision Pro.

Group conversation supports message pinning.

Add the function of receiving group @ reminder offline notifications during Do Not Disturb mode.

Support setting friend remarks in the Accept Friend Request interface.

Add handling of invitations to join groups.

Upgrade vivo push package version in TIMPush.

Fix OV device crash issue in TIMPush.

Add OfflinePushExtInfo support for push through feature in TIMPush.

Fix the issue of not receiving the notification of being kicked out due to network disconnection.

Fix the issue of occasionally not receiving group messages when joining a live group immediately in the login callback.

Fix the issue of still receiving session change callbacks after receiving the delete session callback.

Fix the issue of occasional reset of local data in messages.

Fix the issue of frequent triggering of onRecvMessageModified callback when fetching historical messages.

Fix the issue of no return value and no support for optional values in some Swift interfaces.

Fix the multi-endpoint login exception caused by iCloud sync between different types of devices with the same AppleID.

Fix related issues of communities and topics.

Fix the issue of failing to fetch historical messages on HarmonyOS platform.

Upgrade libcurl in Windows platform to 8.4.0.

Fix the issue of duplicate summary in merged forwarded messages in C++ interface.

Fix the issue of unable to download large images in C++ interface.

Fix the issue of incorrect group type in C++ interface.

Fix the issue of unable to set message custom data in C++ interface.

Fix the forwarding message failure in C++ interface.

### TUICKit & Demo

iOS components provide PrivacyInfo.xcprivacy privacy list file.

TUIChatBot plugin supports markdown text display.

TUIChat chat page header supports displaying call status.

## 7.8.5505 @2024.03.01 - Enhanced Version

### SDK

Add PrivacyInfo.xcprivacy privacy file for iOS & Mac SDK.

Fix the issue that C++ SDK cannot set localCustomData.

Fix the issue of data type error in Swift SDK location messages.

Fix the occasional issue of triggering **message update** callback when pulling historical messages.

Fix the issue of topic `at` message exception.

Fix the issue of fetching exceptions in group and topic lists.

Fix other stability issues.

## 7.8.5483 @2024.02.01 - Enhanced Version

### SDK

Support HarmonyOS Native Platform

Support Loongson architecture

Release TIMPush-UniApp

FCM push supports pass-through messages

Optimize push registration and reporting logic

Add permission group function for community topics

Add stranger attention/fan function

Support configuring cloud message audit policy

Support deleting accounts

Topic information supports obtaining the readSequence field

Fix the problem that after deleting a local inserted group message, the new message received does not update the unread count

Fix the problem that the SDK does not callback occasionally after the user subscribes to multiple official accounts at the same time

Fix the inconsistency of msgID before and after sending official account messages

Fix the problem that the callback does not occur occasionally after subscribing to conversation group unread counts

Fix the problem of abnormal creation time of topics

Fix the problem that when pulling topic information before and after joining the community, the unread count does not change.

Fix the problem of incorrect notification message type for topic information updates

Fix the problem that withdrawn status messages can be searched under certain conditions

Fix the issue of onApplicationProcessed callback being called multiple times

## TUIKit & Demo

Add the TUIEmojiPlugin plugin to support emoticon response functions

TUIChat supports adding and displaying gif dynamic emoticons

Upgrade the built-in small emoji pack in TUIChat to the new version of yellow face emojis

Fix the problem of TUIChat on iOS black screen when clicking image messages

## 7.7.5294 @2023.12.27 - Enhanced Version

### SDK

Optimized Room control logic

Solved the problem that the SDK cannot receive notifications of conversation deletion from the group when the local conversation does not exist

Solved the problem that a conversation's last message can still be searched when it is in a withdrawn state

Solved the problem that message senders cannot receive message change callbacks after live group messages are modified by third-party callbacks

Fixed occasional instability problems in the log module

Optimized community topic unread count logic

## 7.7.5282 @2023.12.18 - Enhanced Version

### SDK

Add V2TIMCommunityManager and V2TIMCommunityListener to integrate community topic interfaces

SDK supports Quic plugin

Optimized conversation filtering logic

CreateTime field is added in the topic information

Live group supports setting administrators and pulling administrator list

The regular expressions of local content moderation ignore English case by default

Fixed occasional jni issue

Fixed occasional callback error issue of weak network message sending

Fixed occasional intValue change issue fo group tips message

Fixed search failure issue for English issue mark by Android SDK

## TUIKit & Demo

Added ChatBot plugin for intelligent chat, supporting FAQ question-and-answer format and streaming text message display

TUIKit adapts to Gradle 8.0

Simplified TUIKit component initialization steps and increase component dependency configuration

TUIChat supports half screen horizontal display in RoomKit

TUIChat supports listeners such as click events and long press events for adding messages

Optimized TIMPush plugin

Optimized customer service plugin experience

Improved voice message click effect

## 7.6.5011 @2023.11.03 - Enhanced Version

### SDK

Add official account feature.

Online status supports returning terminal type.

Save security strike status locally after sending text and image messages are hit by security.

C interface layer's session information supplements session avatar and group specific type.

Optimize message sending failure status and resend logic.

Optimize the message response fetching logic when there is no network connection.

Fix the occasional network connection failure when switching sdkappid across sites.

Fix the issue of being able to search in the cloud after being kicked offline.

### TUIKit & Demo

Added push plugin, console supports viewing statistical indicator data, supports troubleshooting tools.

Added open-source customer service plugin TUICustomerServicePlugin.

TUIChat adds 60-second countdown for sending voice messages.

TUIChat uses the new recall interface, supports displaying the recall operator.

Optimize the display of images/videos/audios hit by security.

## 7.5.4864 @2023.10.13 - Enhanced Version

### SDK

Fixed occasional issue of no callback when retrieving historical messages.

Fixed occasional issue of local message loss.

Fixed occasional incorrect message response status.

Optimized automatic login logic after SDK is kicked out.

## 7.5.4852 @2023.09.27 - Enhanced Version

### SDK

Added notification for offline kicked users

Optimized retrieval of roaming messages

Community supports marking group members

Optimized C API header files

Added interface to ban the entire group chat

Added flag field for message response to indicate whether it is a self-response

Fixed occasional exception in local storage of message reaction

Fixed issue where the revoke-information of lastMessage in the conversation is empty when a message is revoked

Fixed issue where the message sender's remarks are empty when receiving friend messages

Fixed issue where clearing only group chat unread messages also clears unread messages in one-on-one chats

Fixed issue where onRecvNewMessage callback is not triggered for group messages received during network disconnection

Fixed occasional crash issue in server search

Fixed issue where unread count of topic information is not updated after reconnection and retrieval

Fixed issue where conversation's LastMessage is not updated after custom data is set for the last message of the conversation

Fixed crash issue when clearing unread notifications on Windows platform using Swift interface with Int.max as input

### TUIKit & Demo

TUIKit supports RTL language

TUIChat's voice playback supports manual switching between earpiece and speaker

Added voice recognition plugin TUIVoiceToText

Optimized file download

Replaced the latest cleanConversationUnreadMessageCount interface with read report.

## 7.4.4661 @2023.09.08 - Enhanced Version

### SDK

Fix the exception caused by clearing unread messages with one click.

Fix the occasional exception that occurs when searching for cloud messages.

## 7.4.4655 @2023.09.01 - Enhanced Version

### SDK

Optimize server anti-isolation logic in the network module.

Optimize HTTP routing logic in the network module.

Optimize logic for fetching historical messages.

Improve system logs for conversation groups.

Fix the issue of occasional inaccuracy in the unread message count for group conversations.

Fix the issue of occasional inaccuracy in the unread message count for topics in the community.

Fix the occasional issue of not receiving notifications for conversation group creation.

## 7.4.4643 @2023.08.11 - Enhanced Version

### SDK

Support subscribing and unsubscribing users

Support emoji replies

Support voice-to-text capability

Support revoking messages in AVChatRoom

Support setting global message reception options

Single forwarded message supports automatic renewal of rich media resources

Voice and video messages support security strike notifications

Message revoke supports "revoke reason" + "revoker"

Fixed the issue of inaccurate read status for the last message in a one-on-one chat session

Fixed the issue of friend remark information not being cleared in time after deleting a friend

Optimized handling of SQLite file corruption

### TUIKit & Demo

Support storing conversation group information in the cloud

iOS optimized video sending process

Android fixed the issue of continuous voice messages failing to autoplay

Android fixed the issue of incorrect time display in the call record list

Android fixed the issue of overlapping items in the Chat Lite message list

## 7.3.4358 @2023.06.21 - Enhanced Edition

### SDK

Support server message search

Support filtering conversation list and conversation total unread count by whether it contains unread messages

Support filtering conversation list and conversation total unread count by whether it contains @ messages

Adding online identification to group members obtained through getMemberList API

Total number of online group members can be obtained for all type of groups

Optimize the reconnection speed when switching from the background to the foreground

Optimize the slow problem of pulling local messages in weak network

Fix the problem of missing the application list after marking the group application as read

Fix the problem that the conversation cannot be retrieved after sending the first message to a non-friend account

Fix the problem that the conversation personal information is the previous account information when switching accounts to log in

Fix the problem that the local field disappears occasionally when pulling historical messages

Fix the problem that signaling notifications are not throw out occasionally on iOS platform

Fix the problem that the conversation unread count callback does not take effect after clearing all unread messages

Fix the problem that detail information are not filled in conversation object when receiving conversation update notification after deleting or renaming the conversation group

Fix the problem that conversation mark and conversation group change notification are discarded for conversations which do not exist locally

Fix the problem of symbol conflict between SDK and user code

## TUIKit & Demo

Support multiple conversation groups and conversation marks

Add quick meeting function

Adding the feature of playing the next voice message automatically

Picture and video viewing supports zooming in and out and adaptive horizontal and vertical screens

Picture and video upload and download support displaying progress

Picture and video messages support multi-select sending

The group information page supports modifying the invitation option

iOS supports adding suffix names when sending pictures

## 7.2.4146 @2023.05.12 - Enhanced Edition

### SDK

Fixed the issue that the conversation list failed to be returned before the login is successful.

Fixed the issue that the unread count of group conversations could not be cleared occasionally.



Fixed the issue that the last message of the group conversation was not updated occasionally when synchronizing the conversation list.

Fixed the issue that one end could not synchronize to the other end when it included unread group messages.

Fixed the issue that sending unread group messages to conference groups will fail.

Fixed the issue of occasional failure to obtain the total number of unread conversations based on filter conditions.

Fixed the issue that the local field of the message is occasionally lost when pulling historical messages.

Fixed the issue that the package MAC rename SDK prompts that the signature fails.

## TUIKit & Demo

Fixed overseas version TUIKit related experience problems

## 7.2.4123 @2023.04.25 - Enhanced Edition

### SDK

Supported clearing messages for topics.

Added the notification for conversation deletion.

Improved the synchronization speed of conversation lists after login.

Supported blocking members kicked out of the group from joining again for non-audio-video groups and community groups.

Supported checking western European languages by words in local content moderation.

Supported configuring approval options for joining a community group via application or invitation.

Android offline push supported second-level message category for vivo phones

Supported configuring the number of long polling tasks on console for audio-video groups.

Prohibited reverting timestamp for one-to-one message read reporting.

The sequence of the read reported group message cannot exceed the sequence of the last group message.

Android offline push is now adapted to Android 12 for Huawei, Mi and FCM channels.

Fixed the issue that the group name card fields are missing for group messages sent by yourself.

Fixed the occasional error that message modification callback is not triggered after a message is modified successfully.

Fixed the issue that repeated callback for onMemberKicked after a member is kicked out of a group.

Fixed the parsing error for multi-element message in Swift SDK.

Fixed the issue that there is no callback for pulling group messages under certain conditions.

Fixed the occasional issue that message unread count is not updated in time after a message is recalled.

Fixed the occasional OpenSSL crash issue for Android and C/C++.

### TUIKit & demo

Added call records page.

Supported specifying target language for message translation.

Supported customizing time limit for message recall in TUIChat

Fixed the no response issue when users press Enter to send messages with some third-party input methods

## 7.1.3925 @2023.03.07 - Enhanced Edition

### SDK

Supported pulling historical group messages by sequence list

Supported setting whether to clear historical messages when you delete a conversation

Added an API for deleting conversations in batches

Supported modifying the approval method of group member invitations

Supported group counters for community groups

Added a parameter for setting a message object to bypass the content moderation

Supported reporting one-to-one message read by timestamp

Supported reporting group message read by sequence

Supported getting timestamp of the read message in one-to-one chats

Supported getting sequence of the read message in group chats

Fixed the issue that intercepted messages due to local moderation were not saved in the local message database.

Fixed the issue that unread message count displayed when offline group members logged in again even though the messages had been excluded from the unread count.

Fixed the occasional issue of inaccurate isRead status of messages when users sent and received messages again after a one-to-one chat is deleted.

Fixed the occasional inconsistency of unread one-to-one message count in multi-device login scenarios.

Fixed the occasional crashes when file uploading failed.

Changed enumerated values of V2TIMGroupApplicationGetType to those of V2TIMGroupApplicationType

Changed the attribute name getType of V2TIMGroupApplication to applicationType

### TUIKit & demo

Supported group polling and one-by-one notes

Improved the minimalist theme

Optimized the signaling display logic

## 7.0.3754 @2023.01.06 - Enhanced Edition

### SDK

Supported mentioning (@) group members in all types of messages.

Supported getting the total message unread count by conversation filter.

Supported the meta counter for common groups and audio-video groups.

Supported text message translation.

Supported custom attributes for community groups.

Supported setting the Huawei category and Mi channel ID for offline push.

Optimized the QUIC network reconnection logic.

Added the base IP logic for COS rich media file download.

Supported emoji characters in the file paths of rich media messages sent in Windows.

Supported emoji characters in the C++ SDK log and initialization paths in Windows.

Fixed the failure in setting custom conversation marks in the v7a architecture.

Fixed the errors in setting the height of thumbnails or large images.

### TUIKit & demo

Supported gain control and AI-based noise reduction for TUIChat voice message recording.

Added the message translation capability in TUIChat.

Supported custom ringtones for Android FCM push.

Optimized the TUIChat performance in loading historical messages in iOS.

## 6.9.3557 @2022.11.29 - Enhanced Edition

### SDK

Fixed occasional crashes when obtaining `V2TIMOfflinePushInfo` from the message on Android.

Fixed occasional crashes in the Enhanced Edition Pro for Android.

Completed the JSON data content returned by the C API `TIMConvGetConvList` .

### TUIKit and demo

Released a new and simplified version of topics, which is more in line with global app styles.

## 6.8.3374 @2022.11.14 - Enhanced Edition

### SDK

Supported local text moderation on clients.

Released the Swift SDK.

Supported the group attribute feature for non-audio-video groups.

Optimized the logic for updating the number of members in a non-audio-video groups when someone entered the group.

Optimized the COS upload process.

Optimized the issue of unread message count after operations such as message recall in a community group.

Fixed the failure to deliver a notification when a custom friend field is set independently.

Fixed the double callbacks for group listening.

Fixed the issue where the topic profile under a community group would not be updated timely when a user left the group and then joined the group again.

Fixed the issue that sender's profile photo is empty in the callback for inserting a local message successfully.

Fixed the occasional error when setting message extension after repeated login.

Fixed the failure to update the message read receipt status in the conversation update callback after all messages in the conversation are marked as read.

Fixed the occasional emptiness of `send_user_id` in a message when a user sends a message immediately after login callback.

C++ API: Added an API to check whether a message is read.

C++ API: Fixed the failure to update the cursor in the result of the paged pulling of the conversation list.

## TUIKit and demo

Fixed the issue where a search webpage window popped out when a user long pressed a text message on iOS 16.

TUIChat-iOS: Supported sending GIF images.

TUIChat-iOS: Fixed image sending status exceptions.

TUIChat-iOS: Supported the deletion of time when deleting a message.

Fixed system exceptions caused by iOS TUIOfflinePush.

Fixed the issue where Android TUIOfflinePush push parameter settings did not take effect.

## 6.7.3184 @2022.09.29 - Enhanced Edition

### SDK

Supported the messaging extension.

Supported the signaling editing.

Supported VoIP for iOS offline push.

Supported Android offline push in HONOR phones.

Added the backup domain name in the access layer.

Fixed the issue where login and logout callbacks cannot be executed in a special network environment.

Fixed the issue where keeping the group profile empty did not trigger the notification callback.

Fixed the issue where Mute Notifications for group conversations were not updated after the user leaved a group and joined it again.

Fixed the crash triggered by sending message read receipts.

Fixed the issue for C-based APIs where read receipts for one-to-one messages cannot be sent with the SDK.

For the issue for PC where `TIMGroupModifyGroupInfo` cannot modify combined group attributes.

## TUIKit and demo

Optimized the group chat creation process.

Supported setting the background image for the chat area.

Optimized the theme logic.

Supported inviting new group members during a group call.

Supported animoji for Android.

Fixed the occasional messaging missing in the message list for Android.

Fixed the occasional error of message sending state in the message list for Android.

Fixed the issue for Android where the offline push component tried to get the phone model for several times.

Removed the global style modification for `UIAlertController` from iOS TUICore.

Added redirecting to the gallery during the shooting for iOS.

Fixed the issue for iOS where clicking the button triggered a crash after the chat history was cleared.

## 6.6.3002 @2022.08.18 - Enhanced Edition

### SDK

Supported labeling a member of an audio-video group.

Supported removing a member from an audio-video group.

Fixed the occasional crash of the topic update callback for Android.

Fixed incorrect enumerated values in notifications of group joining option changes.

Fixed the issue where no callback for `onTopicInfoChanged` was received after custom topic fields were set.

Optimized the issue for Android where the network IP was requested repeatedly.

### TUIKit and demo

Supported marking conversations as unread/read, hiding conversations, and folding group conversations.

Replaced TUICalling with TUICallKit.

TUICallKit supported displaying the audio or video call in a floating window.

Supported enabling or disabling group grid profile photos as needed.

TUIChat supported customizing background images.

Optimized the TUIOfflinePush component for Android to support calling back the app when users click received messages in the notification bar and to support packaging into native plugins for uni-app.

The TUIKit for Android supported a three-tier community mode: community - group - topic.

Supported showing emoji in the input box for iOS.

Supported showing default profile photos by group type.

Supported showing security tips all the time in the Demo.

Fixed the compatibility and compliance issues arising from using the WebView in the TUICore theme component.  
Fixed the issue where repeated messages displayed occasionally when users clicked messages pushed offline to enter the chat interface.

## 6.5.2816 @2022.07.29 - Enhanced Edition

### SDK

Optimized the routing policies of the IM IDC in India.  
Optimized the callback for the rich media message upload/download progress  
Optimized the compliance required for obtaining the device process info at an Android client.  
Fixed crashes in creating topics successively.  
Fixed the occasional crash occurred in the Windows based packet sending.  
Fixed the crash that occurred in the Android v7a architecture when a friend in the blocked list is added again.

## 6.5.2803 @2022.07.15 - Enhanced Edition

### SDK

Added the support for [marking conversations](#).  
Added the support for [grouping conversations](#).  
Added the support for customizing chat fields.  
Added the support for [advanced conversation pull APIs](#)  
Supported receiving broadcast messages of audio-video groups.  
Supported sending notifications of changes to group joining options.  
Added the support for syncing the changes to group message receiving options across terminals.  
Optimized the routing logic for persistent connections and added the support for rotation policy.  
Scheduled to support for Tencent Cloud Japan.  
Upgraded the authorization ID for HTTPDNS routing requests.  
Added the support for HTTPDNS routing at Tencent Cloud International.  
Optimized the start sequence of long polling requests in audio-video groups.  
Fixed the issue where `lastMessage` is missing in topic profile under certain conditions.  
Optimized the logic of pinning a chat to the top.  
Optimized the statistics collection logic of end-to-end message time.  
Fixed the issue where duplicate messages occasionally appeared in historical messages due to the resending of failed messages.  
Fixed the issue where garbled characters were displayed when emojis were sent on earlier mobile phones.  
Fixed the issue where the `faceURL` was empty in the message returned by `onRecvMessageModified`.

Fixed the issue where the group invitation signaling sent when the invitee was offline couldn't be received after login.

## TUIKit and demo

Supported displaying the "Typing..." status in one-to-one chat.

Supported displaying the online status of friends in chats and contacts.

No longer displayed the "Recall" option 2 minutes after a message was sent.

Made custom messages intercompatible across terminals.

Fixed the issue where chats were not rearranged in certain scenarios on Android.

Removed the TPNS channel from the offline push component.

## 6.3.2619 @2022.06.29 - Enhanced Edition

### SDK

Fixed the occasional crashes when the topic list was obtained.

Fixed the exception in getting the conversation list after a topic was deleted.

## 6.3.2609 @2022.06.16 - Enhanced Edition

### SDK

[Added online status query, "go online/offline" notification, and custom status.](#)

Supported pulling the member list (up to 1,000 persons) of an audio-video group.

Supported @all messages for a topic.

Added the friend adding time in the profile of a friend.

Fixed the issue where cross-platform SQL execution errors occurred.

Added community topic APIs for the cross-platform SDK.

Fixed the issue where the unread count was occasionally incorrect when messages of a specified topic was pulled after login.

Fixed the issue where the call result of the API for getting the list of groups a user has joined was occasionally null when the network was unavailable.

Fixed the issue where the call result of the API for getting the group owner userID of a group a user has joined was occasionally null.

Fixed the issue where the role in the member profile obtained by the new group owner was not upgraded after a group was transferred.

Fixed the issue where search result error occurred when several senders were provided as parameters for the message search API.

Fixed the issue where the name and profile photo of a sender were inconsistent after message search.

Fixed the issue where modifying `cloudCustomData` as null did not apply.

Fixed the issue for iOS where the returned value of `elemType` was 0 after `cloudCustomData` of audio messages was modified.

Fixed the issue for iOS where the read receipts of one-to-one messages were occasionally not called back.

Optimized .so loading for Android

## TUIKit and demo

Supported stickers in chat messages.

Supported quoting a chat message.

Supported replying to a chat message.

Supported read receipts for one-to-one messages.

Unified the time display format of conversation lists.

Supported version upgrade check for the demo app.

Upgraded the brand logo for the demo app.

## 6.2.2363 @2022.04.29 - Enhanced Edition

### SDK

Optimized the network quality of the International Site.

Fixed the issue where a read message was displayed as unread after the app was uninstalled and reinstalled.

Fixed the issue where when the profile of a non-friend user was obtained, the values of custom fields cannot be updated after they were changed to null.

Fixed the issue where the `lastMsg` of a one-to-one conversation was inconsistent with the `lastMsg` in the message history when both sides sent a message simultaneously.

Fixed the issue where after the group owner of a public group approved a group joining request, the callback received by the applicant was incorrect.

Fixed the issue where the `nameCard` of a message sent by a user was null when the user checked this message.

Fixed the issue in some cases where the conversation list was not rearranged after messages were sent.

## TUIKit and demo

Added the offline push component and simplified the integration process.

Added support for read receipts for group messages.

Added support for the dark theme for iOS.

Fixed the issue for Android where the app crashed when an excessively large image was sent and previewed.

Fixed the issue for Android where after a video message was sent, the duration displayed in the message was inconsistent with the actual duration of the video.



Fixed the issue for Android where a user cannot continue to handle friend requests after the user handled a friend request.

## 6.1.2166 @2022.04.02 - Enhanced Edition

### SDK

Fixed the issue where no data was returned when two or more userIDs were entered for `senderUserIDList` to search for local messages.

Fixed the issue where the SDK for Android called back only one message when a user recalled multiple messages with the RESTful API.

Fixed occasional crashes in quickly clearing unread messages for Windows.

### TUikit and demo

Released the International Edition demo.

Switched offline push back to vendor channels.

Switched the login with mobile number to the aPaaS service.

Fixed the failure of audio/video call sync across multiple clients.

## 6.1.2155 @2022.03.18 - Enhanced Edition

### SDK

Added support for setting the sound for Android offline push messages.

Added the API for setting network proxy for mobile SDKs.

Supplemented offline push APIs for the C/C++ platform.

Added support for automatically synchronizing signaling messages in a group after login.

Fixed the issue where a user cannot get complete custom fields after receiving a notification on custom field changes.

Fixed the notification muting status return error that occasionally occurred when the conversation list was pulled under a weak network.

Optimized the log printing logic.

Optimized error descriptions.

### TUikit

Upgraded the personal information protection law for TUikit demo to meet compliance requirements.

Fixed the issue where a user cannot initiate an audio/video call by tapping the banner notification after receiving an offline push notification.

Fixed the issue where a user cannot initiate an audio/video call by directly opening the app after switching the app to background and receiving an offline push notification.

## 6.0.1992 @2022.02.09 - Enhanced Edition

### SDK

Fixed occasional crashes when sending two consecutive messages to a deleted or nonexistent group.

### TUIKit

Added the theme setting capability.

Added the language setting capability.

Added the group profile feature of group management.

Added the file message feature of animation upload/download.

Added the redirection entry "Received XX new messages" when browsing historical messages.

Added the redirection entry "Back to the latest position" when browsing historical messages.

Added the entry for one-click redirection to group @ messages.

Optimized the display style of the last message in the conversation list.

Added the selected state for text messages.

Optimized the A2 and D2 error descriptions.

iOS 15 system UI adaptation.

## 6.0.1975 @2022.01.14 - Enhanced Edition

### SDK

Released SDK version for all-platform C++ APIs.

Added the feature of integrating the TPNS channel for offline push.

Added change notification for custom fields of personal profile.

Fixed the issue where the returned content was occasionally empty when a user attempts to obtain friend remarks.

Optimized network type log printing.

Supplemented the message priority fields of the message object for iOS.

Fixed the issue where the message object returned for callback of inserting local messages was incomplete in the C interface version.

Switched the offline push for the open source demo of the official TUIKit to the TPNS channel.

## 5.9.1886 @2021.12.31 - Enhanced Edition

## SDK

Fixed the issue of incomplete unread messages in the callback after a user logged in and synchronized C2C unread messages.

Fixed the issue of incomplete returned messages after a user pulled local messages.

Fixed the issue where no friend information update callback was available on terminals with this account logged in after the friend remarks were modified.

Fixed HTTPS request errors on the Linux platform.

Fixed the issue where no result was returned for querying the custom fields of friends in the C interface version.

Optimized the error code description of the network layer.

## TUIKit and demo

Added support for displaying and viewing image and video messages by sliding to left or right.

Added support for re-editing recalled messages.

Added the feature of click to call back for signaling messages of audio and video calls.

Added the feature of displaying the sending status of the last message in the conversation list.

Added the "Confirm" pop-up window for deleting messages.

For the Android demo, fixed the issue where offline push was unavailable on Honor phones.

For Android, fixed the issue where the read status of messages in the current conversation was mistakenly updated after the messages of the conversation were forwarded to a third person and read.

For Android, fixed the crash that occurred when the number of audio and video call members exceeded 9.

Unified `minSdkVersion` of the Android TUI component to 16.

For iOS, improved the transmission speed of videos after recording.

For iOS, fixed the black frame of the video cover.

## 5.9.1872 @2021.12.20 - Enhanced Edition

### SDK

Added support for sending targeted group messages.

Added support for COS download authentication.

Added support for AES encryption channels of persistent connections.

Added support for access point anti-islanding for the connection logic.

Added support for backend configuration of COS file upload and download concurrence.

Added the option of giving priority to IPv6 in a dual-stack network.

Added the feature of displaying 20 historical messages from before a user joins an audio-video group (AVChatRoom).

This feature is available only for the Premium edition.

Added support for backend configuration of the maximum number of roaming messages when a user gets historical group messages.

Optimized the description of some error codes.

Fixed the issue where the backend returned `Failed` for quickly clearing unread messages if there were deleted group conversations in the local storage.

Fixed the issue where when a user sent through RESTful API messages excluded from the unread count, the SDK occasionally failed to clear the unread count when issuing a read report.

Fixed the issue where when a user logged in again after quickly entering or leaving a group through RESTful API during offline, an error occurred occasionally when the user got the joined group list.

Fixed the issue where when a user sent messages excluded from unread count when the recipient is offline, the unread count was occasionally inaccurate after the recipient was online again.

### TUIKit and demo

Added support for selecting a country/region on the login page.

Added the feature of clearing initialization operations performed before a user accepts the privacy agreement.

## 5.8.1696 @2021.12.10 - Enhanced Edition

### SDK

Fixed the failure to quickly clearing the unread message count of conversations including disbanded or left group conversations.

### TUIKit

Added the message reply feature.

Changed the default skin and optimized the UI logic.

iOS: fixed the occasional failure to load resource files.

## 5.8.1672 @2021.11.30 - Enhanced Edition

### SDK

Optimized the device information getting logic to meet compliance requirements.

Fixed the crashes in quickly clearing the unread message count under certain conditions.

## 5.8.1668 @2021.11.19 - Enhanced Edition

## SDK

Added the feature of quickly clearing the total unread message count of all conversations.

Supported communities of up to 100,000 members. This feature is available only for the Premium edition.

Added the feature of automatically excluding conversations whose message receiving option is "Receive but not notify" or "Not receive" when getting the total unread message count of all conversations.

Added support for Chinese SM algorithms for encrypted tunnels of persistent connections.

Fixed the issue where, when historical messages were pulled, the end tag was incorrectly determined occasionally.

Fixed the issue where, when the SDK was upgraded from the Basic Edition to Enhanced Edition in overriding mode, audio-video groups that users previously joined had unread message count.

Fixed the failure to setting auto read reporting for accounts in special formats.

Fixed the occasional error of connecting to incorrect servers during frequent network reconnections in private environments.

Fixed the issue where, in multi-client sync scenario, when a user received a group message sent by the user, the SDK automatically clears the unread message count of the group conversation.

Fixed the issue where there is occasionally no callback when users log in again after going offline and being kicked off.

Cross-platform SDK for C: added support for supplementing offline push fields when receiving a new message.

## TUIKit

Optimized the notification muting logic.

Optimized the logic for displaying a red dot for unread messages in the conversation list.

Added support for allowing users to trigger group @ messages by tapping and holding the group profile photo.

Added support for allowing users to tap a button to stop voice message playback.

Added the feature of quickly clearing the total unread message count of all conversations.

Added support for the community group feature.

## 5.7.1435 @2021.09.30 - Enhanced Edition

### SDK

Fixed the issue where local data was not updated in time after group profile custom fields were modified.

Fixed the issue of synchronizing a large number of conversations to be pinned to the top.

Fixed the issue where Android device timeout signaling did not contain the custom data entered during invitation.

Fixed the issue where empty profiles overwrote local profiles due to network request failures during non-friend profile pulling.

Fixed the issue where historical group messages could be pulled after a user left the group and then joined the group again.

Fixed the issue where the callback function `onFriendListDeleted` was called twice after a friend was deleted.

Fixed the issue where the friend remarks of the last message of a conversation were empty.

Fixed the issue where, after the IM SDK was initialized, there was no callback for a `getConversationList` API call by a user who has not logged in.

Fixed the issue where, if failed messages were sent in a group conversation after the network was disconnected, there was no unread message count displayed when the first message was received in the conversation after the network connection was restored.

Fixed the issue where the unread message count could not be obtained in the first conversation with a stranger.

Fixed the issue where the Mute Notifications option for group messages was not updated in certain conditions.

Fixed the issue where incomplete content was called back after group attribute update.

Added the listener addition and deletion APIs for SDKs, groups, relationships, and conversations.

iOS: fixed the issue where audio-video group (AVChatRoom) creation failed when the group joining mode was not set.

## 5.1.66 @2021.09.22 - Basic Edition

### Android

Removed the feature of getting Wi-Fi information.

## 5.6.1202 @2021.09.10 - Enhanced Edition

### SDK

Fixed the issue where, after a user left a group and then joined the same group again, the system included the messages that were not received during this period into the unread message count of the conversation.

Fixed the failure to delete group messages that failed to be sent by muted users.

Fixed the issue where, when historical messages were pulled, the nicknames and profile photos of message senders were occasionally restored to previous ones.

Added support for setting whether to support unread message count in meeting groups.

Added support for the international websites of Singapore, South Korea, and Germany, supporting acceleration domain names.

Fixed the issue where received image messages occasionally had incorrect image formats.

Fixed the issue where, when video messages were sent in Windows, thumbnail sending occasionally failed.

Optimized the report of the success rate of receiving common group messages.

Fixed the issue where, after group members are muted in an audio-video group, the muting period obtained through getting the group member profile is 0.

## 5.6.1200 @2021.08.31 - Enhanced Edition

## SDK

### Common changes

Optimized the login speed.

Added support for the international websites of Singapore, South Korea, and Germany.

Added support for commercial HTTP DNS.

Optimized the group attribute logic to solve the concurrency problem when group attributes are modified on multiple devices at the same time.

Improved the message database query speed.

Improved the network connection policy.

Optimized the search of image, video, and voice messages.

Reduced the time for getting the conversation list ( `getConversationList` ).

Optimized the third-party callback logic for server-side status change: when login on a device causes logout on another device, the server-side logout callback is no longer triggered.

Removed the feature of read reporting for audio-video groups.

Unified login error codes.

Changed the friend search callback parameter `V2TIMFriendInfo` to `V2TIMFriendInfoResult` so that the friend relationship can be determined based on `relationType` .

Added the API for getting offline push configuration for the message object.

Fixed the occasional database crash during the update of user profiles.

Fixed the database query and operation failures before SDK initialization is completed.

Fixed the issue where read receipts became invalid after an app is uninstalled and then reinstalled.

Fixed the issue where `onFriendListAdded` was occasionally called twice.

Fixed the failure to delete messages that are inserted locally.

Fixed the issue where group profile custom fields were not saved when group notifications are saved to the database.

iOS: removed the `Tag_Profile_Custom_` prefix when getting custom user fields and removed the `Tag_SNS_Custom_` prefix when getting custom friend fields.

Android: removed the carrier name and Wi-Fi information getting features and AndroidX dependency.

Android: fixed crashes caused by non-UTF8 encoding.

### TUIKit and demo

Added the pulling-by-page logic for group member list related pages.

Android: optimized the issue where the entire conversation list was loaded each time a message was deleted or recalled.

Android: optimized the issue where the group profile photo was loaded each time when the conversation list was loaded.

iOS: fixed the issue where the displayed number of records was incorrect when a user clears the original keyword and enters a new one to search for messages.

iOS: fixed the issue where searched custom messages were not displayed on the chat screen.

## 5.5.897 @2021.07.29 - Enhanced Edition

### SDK

Fixed occasional crashes when data was reported.

### Android

Removed the calling of `getSimOperatorName()` for getting the carrier name.

## 5.1.65 @2021.07.29 - Basic Edition

### Android

Removed the calling of `getSimOperatorName()` for getting the carrier name.

## 5.5.892 @2021.07.14 - Enhanced Edition

### SDK

Added support for message search by multiple keywords combined with AND or OR.

Added support for message search by a specified message sender account.

Added support for historical message pulling by a specified time range.

Added support for historical group message pulling by a specified sequence.

Added notifications for message modifications by a third-party callback.

Added the API for getting the maximum number of group members that can be added to a group.

Added the `orderBy` field for sorting conversation objects to facilitate sorting conversations without the last message at the app layer.

Optimized the audio-video group message receiving latency by making the backend complete account switching in advance.

Upgraded the network connection scheduling protocol to reduce the network connection time outside the Chinese mainland.

Optimized the conversation list pulling logic.

Optimized the group member pulling logic and enabled local cache.

Fixed the issue where log callback was not triggered when the log level was lower than Debug.

Fixed the issue where group member profiles obtained did not include friend remarks.



Fixed the issue where the obtained list of groups the user has joined contained groups to be approved by the group owner.

Fixed the stability issue reported online.

## 5.4.666 @2021.06.03 - Enhanced Edition

### SDK

Changed the name of Lite Edition SDK to Enhanced Edition SDK.

Added support for message, group, and friend search.

Added a parameter to specify whether to update the last message of the conversation during message sending.

Added support for clearing the roaming messages of a conversation while retaining the conversation.

Added support for concurrent multi-device login on the same platform.

Reduced the time for network connection and login.

Optimized the data reporting feature.

Optimized the offline push logic to support disabling offline push globally.

Optimized the offline push logic to allow setting the message classification field `classification` for vivo phone offline push.

Fixed the occasional incorrectness of the unread message count of one-to-one conversations.

Optimized the historical message pulling speed.

Added support for adding emojis and locations to multi-element messages.

Fixed the issue where, if an offline user changed the user's nickname in a group, the user's nickname in the corresponding conversation was not updated in a timely manner when the user logged in the next time.

Fixed the issue where the 20005 error code was occasionally reported when read messages of one-to-one conversations were reported.

## 5.3.435 @2021.05.20 - Lite Edition

### SDK

Added the API for deleting conversation roaming messages.

Fixed the issue where some Android phones could not receive network status change notifications over persistent connections.

Optimized the logic for pulling user profiles to avoid requesting the backend every time when strangers request for user profiles.

Fixed the issue where group profiles and historical messages could not be obtained when the groups were deleted but conversations were retained.

Fixed the issue where conversations were out of order when you got them via the API for getting a conversation list.

Fixed the issue where group conversations in Mute Notifications mode were filtered out when getting the total message unread count.

Fixed the occasional crashes caused by iOS HTTP requests.

## 5.1.62 @2021.05.20 - Basic Edition

### SDK

Fixed known issues.

## 5.3.425 @2021.04.19 - Lite Edition

### SDK

Added support for pinning a conversation to the top.

Added support for setting the Mute Notifications option for one-to-one messages.

Added support for sending messages that are excluded from the unread count.

Added support for getting local conversation and message data when there is no network connection or your login fails.

Added XCFramework (supporting Mac Catalyst) to the SDK for iOS.

Added the API for getting the conversation unread count.

Added the `birthday` field to personal profiles.

Fixed the issue where, when group @ messages were recalled, the conversations of the @ target users still contained the group @ notification.

Fixed the issue where, for some Android phones, the network would be disconnected and connected again after a successful initial network connection during persistent connections.

Fixed the issue where users could not set custom fields when creating a group in the SDK for iOS.

Fixed the issue where users with special accounts could not search for local messages via `findMessage` .

## 5.1.60 @2021.04.06 - Standard Edition

### iOS

Fixed the issue where the SDK may be rejected by the App Store for using IDFA related keywords.

## 5.2.212 @2021.04.06 - Lite Edition

## iOS

Fixed the issue where the SDK may be rejected by the App Store for using IDFA related keywords.

## 5.2.210 @2021.03.12 - Lite Edition

### SDK

#### Common changes

Added support for forwarding multiple messages as a combined single message.

Optimized the logic of persistent connections, improving the quality of connections outside the Chinese mainland.

Specified login error codes in a detailed way to distinguish whether the network is normal during login.

Optimized the logic of COS upload, providing better experience of sending rich media messages.

Added the advanced API for getting historical messages.

Added the API for getting conversations in batches.

Added the API for checking friend relationships in batches.

Fixed the issue where two messages were generated in the local database after a message that failed to be sent was sent again.

Fixed the issue where the muting time called back was incorrect when the group member profile was changed.

Fixed the issue where the width of the image called back was incorrect when an image message was received.

Fixed the issue where the console still printed logs after `LogLevel` was set to `None`.

Fixed the issue where the `add_source` field of adding friends was incorrect.

Fixed the issue where sometimes the sending progress called back was negative when a video file greater than 24 MB was sent.

## 5.1.56 @2021.03.03 - Standard Edition

### SDK

#### Common changes

Optimized the logic of persistent connection, improving the quality of connections outside the Chinese mainland.

Optimized data reporting and specified error codes related to network timeout in a detailed way.

Fixed known stability issues.

## iOS

Fixed occasional failures of extracting logs in the iOS SDK.

## Android

Replaced the log component of the Android SDK to improve stability.

## Windows

Fixed the issue where the client thread might block the SDK logic thread when a new message callback was triggered in the Windows SDK.

## 5.1.138 @2021.02.05 - Lite Edition

### SDK

#### Common changes

Optimized logging.

Optimized the policy of persistent connection, improving the quality of connections outside the Chinese mainland.

Fixed the issue where sometimes the last message was incorrect when multiple one-to-one messages were sent or received in the same second.

Fixed the issue where sometimes there was no callback for querying the conversation list.

Fixed the issue where sometimes the sequence number of a one-to-one message was incorrect.

#### Android

Fixed the issue where sometimes a negative upload progress was displayed when a video greater than 24 MB was sent.

Fixed occasional crashes when messages were sent.

## 5.1.50 @2021.02.05 - Standard Edition

### SDK

V2 APIs added the `random` field for message objects.

Added support for recalling the `lastMsg` message in a conversation.

Fixed occasional exceptions in the status of the last message obtained via the `getMessage` API.

Fixed the issue where messages were delayed when user profiles were frequently pulled after messages were received.

Fixed the issue where deleting the account might cause the failure to pull the group member list.

Fixed the issue where the message might not be found when `findMessage` was called after `insertLocalMessage`.

Fixed the issue where a conversation update callback was triggered when a conversation was deleted.

Fixed the issue of the Android SDK where the nicknames of historical group messages were not timely updated.

Improved the database stability of the iOS SDK.

### TUIKit and demo

Fixed the issue of the Android TUIKit where a black screen was displayed when you tried to view the original images that were not downloaded.

Fixed the internationalization issue for iOS.

Fixed the issue for iOS where images were overwritten when multiple images were sent at a time.

Fixed the issue of the iOS 14 operating system where there was no response when you clicked the "add" or "delete" button on the group details page.

Fixed the issue of the iOS 14 operating system where the tab bar disappeared after you left a group conversation and went back to the message list.

## 5.1.21 @2021.01.15 - Standard Edition

### SDK

#### Android

Fixed the issue where custom messages with the extended field `extension` failed to be sent on the Android platform.

### TUIKit and demo

#### iOS/Android

Improved internationalization support by eliminating the issue where there were Chinese characters in the English edition.

## 5.1.137 @2021.01.29 - Lite Edition

### SDK

#### Common changes

Fixed the issue where there was no callback for the login API occasionally when a user logged in to the same account repeatedly on multiple iOS devices or Android devices.

#### Android

Fixed crashes that occurred occasionally when a low-end Android device tried to obtain the log path.

## 5.1.136 @2021.01.27 - Lite Edition

### SDK

#### Common changes

V2 APIs added an API for log callbacks.

Fixed the issue where the UserID of the @ target user in the group @ message was empty.

Fixed the issue where audio-video group messages occasionally could not be received.

Fixed the occasional issue of incorrect login status in the case of frequent network reconnection.

Fixed the issue where users occasionally failed to log in again after going offline and being kicked off.

Fixed occasional crashes in DNS resolution.

## 5.1.132 @2021.01.22 - Lite Edition

### SDK

#### Common changes

Added support for overload protection in the network module.

Fixed the issue where some sessions occasionally were lost when the standard edition was upgraded to the Lite Edition.

Fixed the issue where the `onUserSigExpired` callback could not be received after the login information expired.

Fixed the issue where a member received the `onMemberKicked` callback after being kicked out of a group and joining the group again.

## 5.1.131 @2021.01.19 - Lite Edition

### SDK

#### Common changes

Added the API for forwarding a single message.

Optimized the logic of receiving audio-video group messages. When an audio-video group receives a message, the sender's nickname and profile photo are no longer queried.

Fixed the issue where there was no conversation update notification when the last message in a conversation was deleted.

Fixed the issue where the unread messages count in one-to-one conversations occasionally was cleared when the one-to-one messages were synchronized after login.

Fixed the issue where the last message in a conversation was not updated when the conversation list was synchronized after a user went offline and then online.

#### Android

Fixed the issue where the settings of the custom message field `description` and personal profile fields `level` and `role` did not take effect.

Fixed occasional crashes during deinitialization.

## 5.1.129 @2021.01.13 - Lite Edition

## SDK

### Common changes

Fixed the issue where a conversation update callback was triggered when a user tried to get the conversation list and there was no conversation update.

Fixed the issue where the last message in a conversation was not cleared when a user tried to delete all the messages in the conversation.

### iOS

Fixed the issue where the returned information was not `nil` when a non-signaling message was passed in using the `getSignallingInfo` method.

### Android

Fixed occasional crashes caused by JNI local reference table exceeding the limit.

## 5.1.20 @2021.01.08 - Standard Edition

## SDK

### Common changes

V2 custom messages added the `desc` and `ext` fields.

V2 user profile APIs added the `role` and `level` fields.

Optimized V2 APIs. Whether your login is successful or not, you can get the data of the local conversation list and local historical messages.

V2 added the `getHistoryMessageList` API to support getting cloud or local messages and getting messages sent before or after a specific time.

Optimized the issue in getting the profile photos of one-to-one messages.

Optimized the security and renewal of rich media message file upload.

Fixed the issue where the local paths of sent rich media messages were empty.

Fixed the issue where when a local message was inserted into a group, the previous message was displayed as the `lastMessage` of the conversation after you logged out and logged back in.

Fixed the Elem out-of-order issue.

Fixed the issue where the @ prompt still existed in the message list after the group @ message was recalled.

Fixed the issue where system messages were pulled when you pulled the offline historical group messages after going online.

Fixed the issue where two offline push notifications were received when only one signaling invitation for a voice call was sent.

Fixed the issue where the settings of local "custom message data" became invalid when there were too many messages.

Fixed the issue where the unread number did not decrease after an unread group message was recalled.

Fixed other stability issues.

### iOS and Mac

Fixed receiver crashes that occurred when `array json` was passed for custom messages.

Fixed crashes after calling `deleteConversation` and passing the wrong conversation ID.

Fixed the issue where the last draft in the draft box could not be deleted.

### TUIKit and demo

Fixed the issue where the information of the conversation pinned to the top was not deleted after you deleted the friend or left the group on the iOS platform.

Fixed the issue where after a user was set as the administrator, the console still showed that the user did not have the administrator permissions on the iOS platform.

Fixed crashes that occurred when the thumbnail was empty on the iOS platform.

Fixes the issue where there was an exception in the height of a recalled long-text message on the iOS platform.

Fixed the issue where group muting tips were not displayed on the iOS platform.

Optimized the time display of the conversation UI on the iOS platform.

Fixed crashes that occurred when a user clicked **Back** after the creation of a live room entered the countdown process on the Android platform.

Fixed the issue where the call interface did not disappear when a member refused to answer the call in a group chat on the Android platform.

Fixed the issue where the small window was not closed when a viewer in the live room was kicked offline in the small window mode on the Android platform.

Fixed occasional crashes that occurred when someone joined a group on Android devices.

## 5.1.125 @2021.01.08 - Lite Edition

### SDK

#### Common changes

V2 APIs added the `random` field for message objects.

V2 APIs added the `description` and `extension` fields for custom messages.

V2 APIs added the `role` and `level` fields for user profile objects.

Fixed the database compatibility issue in the upgrade from versions below 4.8.1 to the Lite Edition.

Fixed the issue where users occasionally received the callbacks of messages sent by themselves.

Fixed the issue where there was no callback when you tried to get the list of groups that you joined when you hadn't joined any group.

Fixed the issue where there was no conversation update callback when setting group message receiving options.

Fixed the issue where occasionally there was no end callback for conversation synchronization.

Fixed occasional crashes during conversation synchronization.



## 5.1.123 @2020.12.31 - Lite Edition

### SDK

#### Common changes

Fixed the issue where the Android edition could not receive custom group system messages sent via the RESTful API.

Optimized the method of generating the value of the `random` field for a message.

Optimized log printing to facilitate troubleshooting.

Fixed occasional crashes in the network module.

## 5.1.122 @2020.12.25 - Lite Edition

### SDK

#### Common changes

Fixed the issue where there might be no callback when setting conversation drafts.

Fixed the issue where the message sender information was not completed when searching for messages via

`findMessage` .

Fixed the issue where it might fail to search for messages via `findMessage` after inserting local messages.

Fixed the issue where conversation objects were not updated when setting group message receiving options.

Fixed the issue where conversation change notifications were not sent when personal or group nicknames or profile photos were changed.

Fixed the issue where the last message in a conversation was not updated when inserting local messages.

Enabled on-cloud control over personal profile update cycle.

#### iOS

Fixed occasional crashes caused by improper dictionary or array operations.

#### Android

Fixed occasional crashes when deleting messages.

## 5.1.121 @2020.12.18 - Lite Edition

### SDK

#### Common changes

Optimized the group profile pull logic. For audio-video groups, users' own group member information does not need to be pulled.

Improved log printing and added the device type field.

Fixed the issue where, when a message recall notification was received in a one-to-one conversation, the status of the last message in the conversation was not updated.

Fixed the issue of excessive message delay during long polling in an audio-video group.

Fixed the issue where, when a user logged in to the same account repeatedly and then joined the same audio-video group, the message long polling module did not update the message pull key.

### **iOS**

Fixed the issue where, when a JSON array was passed in for custom message fields on iOS, the signaling module on the receiving end crashed during parsing.

### **Android**

Fixed occasional crashes when setting conversation drafts.

## 5.1.118 @2020.12.11 - Lite Edition

### **SDK**

#### **Common changes**

Optimized the message deduplication logic and fixed the issue where repeated callbacks were triggered for the same message.

Added an API for local insertion of one-to-one messages.

Fixed the issue where the unread group message count did not decrease when unread group messages were deleted or recalled.

Fixed the issue where messages that failed to be sent could not be deleted.

Fixed the issue where the deletion failure callback was triggered when a user attempted to delete a conversation in a group that the user had left or a group that had been deleted.

Fixed the issue where the setting failure callback was triggered when a user attempted to enable group message read reports for a group that the user had left or a group that had been deleted.

### **iOS**

Fixed the issue where setting the signature in personal profiles failed.

### **Android**

Fixed the issue where adding a friend to a blocklist occasionally led to crashes.

Fixed the issue where no message ID was returned when a message was sent.

## 5.1.10 @2020.12.04 - Standard Edition

### **SDK**

#### **Common changes**

V2 APIs added support for custom group fields and multi-element messages.

V2 APIs added an API for the local insertion of one-to-one messages.

Mitigated the issue of message loss for ordinary groups and audio-video groups.

Fixed the issue where messages that failed to be sent could not be deleted.

Fixed the one-to-one conversation issue where, if the first message was sent online, the read receipt was not received.

Fixed the issue where, after a recalled message was returned through the API for pulling historical messages, the message status was incorrect.

Fixed the failure to return information of all friend lists when `null` was entered as the friend list name for the API for obtaining friend list information on iOS.

Fixed known stability issues.

## 5.1.115 @2020.12.04 - Lite Edition

### SDK

#### Common changes

Optimized synchronization between the signaling timeout threshold and server time.

Fixed occasional failures in establishing connections on a weak network.

#### iOS

Completed API header files.

#### Android

Fixed crashes by replacing Gson with JSON.

## 5.1.111 @2020.12.01 - Lite Edition

### SDK

#### Common changes

Improved log printing.

Fixed known stability issues.

## 5.1.2 @2020.11.11 - Standard Edition

### SDK

#### iOS and Mac

iOS allows iPhones and iPads to be online at the same time.

Mac supports the ARM64 architecture.

#### Android

Fixed a stability issue in the Android edition.

Substituted the standard TRTC dependency package.

## 5.1.110 @2020.11.26 - Lite Edition

### SDK

#### Common changes

Added all V2 APIs.

Added the conversation feature.

Added the relationship chain feature.

Added the group @ feature.

iOS allows iPhones and iPads to be online at the same time.

Added support for multi-element message sending.

Supplemented custom fields in group profiles.

Fixed known stability issues.

## 5.1.1 @2020.11.05 - Standard Edition

### SDK

#### iOS/Android

Added an API to obtain the number of online users in an audio-video group (AVChatRoom).

Added an API to query messages based on the unique ID.

Added an API to obtain the server calibration timestamp.

Optimized the login speed.

Optimized the group profile pull logic.

Fixed the issue where pulling local messages failed after users left a group.

Fixed the issue where, after a successfully sent message was modified by a third-party callback, the message on the sender end was not promptly updated.

Fixed the issue where, after configuration via the console, conversations of meeting groups still did not support unread counts.

Fixed the issue where users in an audio-video group (AVChatRoom) occasionally failed to receive messages.

Fixed some other occasional stability issues.

### TUIKit and demo

#### iOS/Android

Group members can input @All .

TUIKit components added international support.

Added support for selecting videos when sending image messages through the Android edition.

Optimized the timeout logic for voice and video call requests.

Updated Android offline push to be dependent on the TPNS package.

Group live streaming added an opening animation.

Group live streaming added support for a small livestreaming window.

## 5.0.108 @2020.11.02 - Lite Edition

### SDK

#### Common changes

Fixed a stability issue in the iOS edition.

Fixed the occasional message callback failures in the Android edition.

## 5.0.10 @2020.10.15 - Standard Edition

### SDK

#### iOS/Android

Optimized signaling APIs to support the setting of `onlineUserOnly` for online messages and `offlinePushInfo` for offline push messages.

Optimized the async callback for the API for obtaining a single conversation.

Added an API for obtaining group types for conversations to facilitate display filtering of the conversation list.

### TUIKit and demo

#### iOS/Android

Added [group livestreaming](#) features, such as co-anchoring, gifts, beauty filter, and voice changing.

Added [live rooms](#) that support co-anchoring, PK, likes, gifts, beauty filter, on-screen comments, following friends, and other features.

Optimized the recognition of audio and video signaling.

## 5.0.106 @2020.09.21 - Lite Edition

### SDK

#### Common changes

Fixed known stability issues.

## 5.0.6 @2020.09.18 - Standard Edition

### SDK

#### Common changes

Added the group @ feature.

Added the `deleteMessages` API for iOS and Android, which will simultaneously delete local and roaming messages.

When deleting a conversation, the `deleteConversation` API also deletes local and roaming messages.

API2.0 added APIs for setting and obtaining custom fields for user profiles, friend profiles, and group member profiles.

Optimized image upload compatibility issues.

Fixed the issue where after the group message receiving option was modified and then immediately obtained, the option remained unchanged.

Fixed the issue where after a local C2C conversation was deleted, C2C system notifications updated the conversation but the message `elem` was empty.

Fixed the issue where image upload failed when the userID contained Chinese characters.

Fixed the issue where after an account with special characters successfully set the user nickname and entered the group to send a message, the nickname was still blank in the new message callback received by other group members.

Fixed known crashes.

#### iOS

Fixed the crash issue that occurred when message listening was removed.

Fixed the issue where deleting a conversation peer account led to exceptions in obtaining the conversation.

Mitigated the issue of initialization lag.

#### Android

Optimized the processing for signaling sending timeout failure.

Fixed the issue of invalid custom data for the signaling cancellation API.

Fixed the issue where attempts to delete all attributes failed when `null` was passed in for the `keys` of the group attribute deletion API.

Fixed the issue where signaling group calls could still be accepted or rejected after being accepted or rejected.

Fixed the multi-element resolution issue for API 2.0.

#### Windows

Fixed the known issue of memory leak.

Optimized log upload.

Fixed the issue where a user who simultaneously logged in to the same account from multiple PCs of the same model was not forced offline.

Fixed the issue where received messages were out of order on a PC.

## TUIKit and demo

### iOS

Added the group @ feature.

Added new emoji packs.

Updated the SDWebImage dependent library.

Optimized UI display for applications to join a group.

Optimized the text display of voice and video calls.

### Android

Added the group @ feature.

Fixed the issue where the contacts displayed during group creation might be inconsistent with those actually selected.

Fixed the issue where the display of custom messages might be out of order.

Fixed occasional crashes of AVCallManager and TRTCAVCallImpl.

Added new emoji packs.

## 5.0.102 @2020.09.04 - Lite Edition

### SDK

#### Common changes

Released the Android & iOS Lite-Edition SDK.

Compared with the standard edition SDK, the Lite Edition SDK removed the friend and conversation capabilities and optimized some service logic to ensure higher execution efficiency and a smaller installation package size.

## 4.9.1 @2020.07.24 - Standard Edition

### SDK

#### Common changes

Optimized login outside the Chinese mainland.

Fixed file upload failures in some regions outside the Chinese mainland.

Fixed file upload failures for accounts containing the @ symbol.

Fixed occasional errors with unread count of one-to-one messages.

Fixed occasional exceptions in conversation `showName` display.

Added an API for obtaining the download URL of file messages.

### iOS

Fixed the issue where there was no callback when users attempted to obtain one-to-one messages while network connection was not available.

## Android

Fixed occasional crashes of signaling parsing APIs.

Fixed occasional crashes when obtaining offline push information.

Fixed the issue of no callback when API 2.0 `getFriendApplicationList` carried no data, and fixed the issue of no callback when non-members were specified for `getGroupMembersInfo` .

## Windows

Added detailed group information when users obtain the list of groups joined.

Fixed the failure to send small files.

Fixed error 6002 reported by logs.

## TUIKit and demo

### iOS

Added push of offline voice and video calls and enabled redirection to the call answering interface.

Fixed failure to delete or recall custom messages.

Optimized the interface.

Migrated the voice and video code from Swift to Objective-C to substantially reduce third-party dependent libraries.

Added support for TUIKit pod integration of two types of voice and video dependent libraries: LiteAV\_TRTC and LiteAV\_Professional.

### Android

Optimized the offline push of the demo and upgraded the push SDK version for each vendor.

Added push of offline voice and video calls and enabled redirection to the call answering interface.

## 4.8.50 @2020.06.22 - Standard Edition

### SDK

#### Common changes

Fixed the API 2.0 issue where the `onMemberEnter` callback was not triggered when someone entered an audio-video group (AVChatRoom).

Added the `groupID` parameter to the `onGroupInfoChanged` and `onMemberInfoChanged` callbacks of API 2.0.

Fixed the issue where there was no conversation update callback after a one-to-one message was sent successfully.

Fixed the issue where a user failed to receive messages after switching accounts and joining the same audio-video group (AVChatRoom).

Fixed the occasional issue of incorrect callback sequence during unread message synchronization after login.

Adding signaling APIs.

Added the custom group attribute API for audio-video groups (AVChatRoom).

Fixed known crashes.



## Android

Changed the default log storage location to `/sdcard/Android/data/package name/files/log/tencent/imsdk` to be compatible with Android Q.

## Windows

Fixed group member role issues during group creation.

## TUIKit and demo

### iOS

TUIKit replaced API 2.0.

Integrated TRTC to realize the voice and video call feature.

Added the deep-color mode.

### Android

TUIKit replaced API 2.0.

Integrated TRTC to realize the voice and video call feature.

Supports AndroidX.

## 4.8.10 @2020.05.15

## SDK

### Common changes

iOS and Android support IPv6.

Audio-video groups (AVChatRoom) support dynamic updates of the group member list.

Fixed xlog crashes.

### iOS and Mac

Fixed the failure of iOS to send big files.

Fixed the exceptions that occurred when `getFriendRemark` was triggered to fetch the sender's friend remark in a V2TIMMessage message.

### Android

IM SDK supports AndroidX.

Fixed the crashes of Android devices caused by network permission issues.

## 4.8.1 @2020.04.30

## SDK

### Common changes

Launched brand-new API 2.0 for iOS & Android.

Fixed conversation errors when users logged in to different accounts in certain scenarios.

## 4.7.10 @2020.04.23

### SDK

#### Common changes

Fixed login timeout in some network environments.

Fixed inaccurate unread counts in some scenarios.

## 4.7.2 @2020.04.03

### SDK

#### Common changes

Fixed a data error.

## 4.7.1 @2020.03.23

### SDK

#### Common changes

Optimized the local log size.

Optimized the login time.

Fixed the multi-terminal unread count synchronization issue.

Added the `getFriendList` API.

The iOS and Android SDKs enable you to set the message title and content to display on the offline push notifications bar of iOS and Android devices, respectively.

## 4.6.102 @2020.02.28

### SDK

#### Common changes

Fixed slow message pulling in some scenarios.

Fixed the compatibility issue with sending 3.x version audio messages to later versions.

Fixed the issue where the identifiers of some conversations in the obtained conversation list were null.

Fixed known crashes.

Fixed SOCKS5 proxy users' password verification issue.

Optimized the pending group processing logic.

Improved the file upload limit to 100 MB.

Optimized COS upload.

Fixed the issue where an exception was returned for obtaining the friend list if there was no friend.

## 4.6.56 @2020.01.08

### SDK

#### Common changes

Mitigated the issue where memory grew when user profiles were frequently pulled.

Improved compatibility with special characters in user profiles.

Fixed known crashes.

Fixed occasional login failures when accounts are switched frequently.

Fixed reconnection in the pressure test.

## 4.6.51 @2019.12.23

### SDK

#### Common changes

Improved network connection quality to quickly detect network quality changes.

Optimized audio-video group message handling.

#### iOS and Mac

Changed all IMSDK listeners from strong references to weak references of external objects.

Added the `getSenderNickname` API for messages.

#### Android

Fixed the issue where offline users are kicked off.

Fixed upload progress callback exceptions on devices running earlier Android versions.

Fixed memory leak during login.

Added the `getSenderNickname` API for messages.

#### Windows

Fixed the issue where messages failed to be sent to newly added friends.

Improved modification and query of custom fields for group information and group member information.

Improved callbacks for all APIs to ensure that callbacks will be called and that objects are transferred to JSON strings only when callbacks succeed and empty strings are returned when callbacks fail.

## TUIKit and demo

### Android

Profile photos displayed in conversation lists can be set with rounded corners.

Fixed the account switching exception when a conversation is pinned to the top.

## 4.6.1 @2019.11.13

### SDK

#### Common changes

Roaming messages can be recalled.

Fixed the unread count error when a user was invited to join a group in silent mode through a RESTful API.

Fixed occasional message sending exceptions due to poor network connection.

Fixed incorrect logic for role filter conditions when group members are obtained.

Fixed the issue where the SDK failed to get the group name the first time users sent a message in a group created by a RESTful API.

Fixed the issue where `getUsersProfile` failed to get user information after caching was disabled.

Fixed the issue where voice message files without a suffix could not be downloaded after they were received.

#### iOS and Mac

Added OPPOChannelID settings to fix the issue where OPPO mobile phones running Android 8.0 or later failed to receive iOS push messages.

Optimized annotations to `getGroupList` return objects.

### Android

Offline pushed channelID on OPPO mobile phones running Android 8.0 or later can be configured in the console.

The ext, sound, and desc fields of `TIMCustomElem` have been deprecated.

### Windows

Fixed the exceptional type field of group system messages.

Fixed inconsistent group type and header file in the returned group information.

Fixed the issue where specifying custom group fields failed during group creation.

Added sender profile and offline push configuration to messages.

## TUIKit and demo

### iOS

Added the video call feature.

Added 3x3 grid display of group profile photos.

Optimized the conversation list, contacts, and chat UIs.

### Android

Added a method to set whether to display read receipts.

Added 3x3 grid display of group profile photos.

Optimized the conversation list, contacts, and chat UIs.

Fixed compatibility issues with the input method, UI, and file selection for some mobile phones.

Fixed messy display of custom messages.

Fixed slow contact loading in the stress test.

Fixed the conflicts with other library resources.

Fixed ineffective cache directory settings.

## 4.5.111 @2019.10.16

### SDK

#### Common changes

Fixed the paging issue of the API used to get the list of group members of a specified type.

Added file format extension to the URL generated upon sending a file message.

Added the notification callback after custom group fields are modified.

Local user and group information can be obtained before login by calling the `initStorage` method.

Fixed the memory leak issue.

Fixed the issue with incorrect message status codes after sent messages are recalled.

Fixed the issue with incorrect `getMessage` callback error codes.

Fixed incorrect one-to-one chat unread count after an app is killed and restarted.

#### iOS and Mac

Fixed occasional login failures for sleeping Mac devices.

#### Android

Fixed stability issues in some scenarios.

Fixed the issue where OPPO mobile phones running Android 8.0 or later could not receive offline push notifications.

Optimized the return types of the `getElementCount` API.

#### Windows

Improved the network reconnection speed for cross platform libraries.

Fixed the Windows public group management setting failure.

Added JVM configuration to cross-platform libraries to facilitate passing jvm from an Android environment.

### TUIKit and demo

#### iOS

Added support for sending and receiving voice messages to and from web applications.

Fixed the issue where TUIKit resource files could not be found when swift loading.

Fixed the issue where a friend's alias could not be seen on the chat interface after it was modified.

Fixed the issue where the conversation list did not refresh promptly after a conversation was pinned to the top.

### Android

Added support for sending and receiving voice messages to and from web applications.

Added support for setting the input box style.

Displayed a red dot on unread voice messages.

Fixed the issue where video messages could not be played on x86 devices.

Fixed conflicts between FileProvider and the integration side.

Fixed the issue where audio permissions could not be identified on some mobile phone models.

Fixed the issue where the profile photo cannot be loaded in specific conditions.

Fixed occasional incomplete display of bubbles.

## 4.5.55 @2019.10.10

### SDK

#### Common changes

Fixed crashes when networks are switched multiple times.

Improved network connection quality.

Optimized annotations of some APIs.

### Android

Optimized HTTP request restrictions on Android 9.0 or later.

### iOS and Mac

Optimized pod integration.

## 4.5.45 @2019.09.18

### SDK

#### Common changes

Improved network connection quality.

Fixed the exceptional unread count when new messages are received after a group chat is deleted.

Fixed the issue where deleted conversations could still be obtained from the conversation update callback.

Optimized the logic for pulling custom group/group member fields.

### Android

Deprecated the `setOfflinePushListener` API and `TIMOfflinePushNotification` class in `TIMManager`.

### TUIKit and demo

## iOS

Fixed the NSString + Common.h class conflict issue.

Fixed the incomplete group tip display issue.

## Android

Added read receipts.

Compatible with typing display in earlier versions.

Fixed the issue where resent messages failed to immediately appear at the bottom of the chat window.

Fixed the issue where profile photos in a group chat failed to be displayed under specific conditions.

Fixed the issue where multi-element group messages could not be displayed.

Fixed crashes caused by specific messages.

Fixed the group admin permission error.

Fixed the issue where files sent by web applications could not be received.

# 4.5.15 @2019.08.30

## SDK

### Common changes

Improved the speed of sending file messages for users outside the Chinese mainland.

Fixed the issue where the message status fetched by `getLastMessage` was incorrect after a message was recalled. Fixed the issue where the callback is called multiple times after message listening was recalled.

Fixed the issue where the backend failed to obtain the muting time after a member is muted, left the group, and joined the group again.

Fixed the issue where the message time was ineffective during `savemsg` after the message time was proactively modified.

Fixed the issue where no callback occurred occasionally upon login.

Fixed the issue where `rand` and `timestamp` of a recalled group message were empty.

Fixed the issue where UserSig in a callback expired when the user was logged out. Fixed the issue where reconnection continued when the user was logged out.

## Android

Added support for FCM push notifications on Android devices in the backend.

Fixed the issue where an error was reported when null was passed for getting a specified friend list.

Fixed checkEquals crashes in specified scenarios.

## Windows

Added the `unique_id` field to `MessageLocator`.

Added support for 64-bit Windows.

Added user profile APIs and relationship chain APIs to the cross-platform library.

## TUIKit and demo

### iOS

Added support for sending custom messages.

Added read receipts for one-to-one messages.

Added a red dot to unplayed audio messages.

### Android

Fixed the demo memory leak issue in some scenarios.

Fixed crashes in some scenarios.

Fixed the incorrect custom message color issue.

Fixed the incorrect or incomplete bubble display issue.

Fixed the issue where conversation lists failed to display profile photos.

Fixed the issue where the title bar color could not be changed by ConversationLayout.

Added support for 64-bitijkplayer.

Added support for multi-element messages.

## 4.4.900 @2019.08.07

### SDK

#### Common changes

Fixed stability issues in some scenarios.

Optimized the unread message count.

Improved the latest conversation list loading speed after login.

Added the log cleaning feature.

Fixed message loss when synchronizing a large number of unread one-to-one messages.

After a user leaves an audio-video group, system messages about members leaving the group will not be pushed to the user's device.

Fixed the issue where group system messages occasionally failed to be delivered to users.

Added the frequency limit logic to `onRefresh/onRefreshConversations`.

Optimized exceptional `saveMessage` ordering.

#### iOS and Mac

Changed the `getGroupInfo` callback parameter to `TIMGroupInfoResult` to fetch the error codes corresponding to each group.

Optimized the display style of push notifications for 4.x versions to keep consistency with 2.x and 3.x versions.

Fixed the issue where login accounts that contain Chinese characters failed to send images, files, and videos.

#### Android

Fixed the issue where mobile phones running the 4.2.2 system version failed to load so.



Fixed the issue where `getGroupInfo` returns an incorrect amount of data.

Changed the `getGroupInfo` callback parameter to `TIMGroupDetailInfoResult` to fetch the error codes corresponding to each group.

Used the `com.tencent.imsdk.TIMGroupReceiveMessageOpt` class in a unified manner.

## Windows

Fixed the issue where the Windows configuration file path is garbled.

## TUIKit and demo

### iOS

Modified the iOS demo UI, including the default profile photo and four feature icons (camera, video, album, and file) on the input interface.

Added the profile card to "Me" and put personal information in the profile card.

Added the feature to view the large image by tapping the profile photo.

Modified the style of the small gray bar in group chats in the demo so that the member nickname becomes blue and tapping the nickname will redirect the member to the member's profile page.

Optimized the logic for displaying nicknames in groups in the demo.

Optimized the logic for displaying profile photos on the chat interface.

Added tap feedback to all interfaces, allowing users to set and customize feedback in TUIKit.

### Android

Added `MotionEvent.ACTION_CANCEL` event handling for audio messages in chats.

Added profile photo display in the conversation list, chat interface, detailed profile, and contacts.

Added profile photo change in user profiles.

Added Intent redirection to offline push functions.

Added random profile photos for one-to-one chats and group chats.

Added prompts for granting and revoking the group admin role for a group member.

Added prompts for muting and unmuting group members.

Fixed the issue where the text "You've recalled a message" was not displayed in tips after a message was recalled.

Fixed the issue where the content of a recalled message was always displayed as the last message in the conversation list.

Fixed the white screen issue on the chat interface after offline messages were received on Meizu mobile phones.

Fixed the issue where the chat conversation pinned to the top did not update to the last message when new messages came in.

Fixed Toast notifications when the username or password is empty.

Fixed the issue where GroupTips messages transferred from the group owner were displayed abnormally in TUIKit.

Fixed the Didn't find class "android.support.v4.content.FileProvider" error reported on some mobile phones.

Optimized the logic for pinning a chat to the top to arrange chats in chronological order starting from the most recent.

Fixed the issue where the soft keyboard and other layouts appeared in chats at the same time.

Fixed the issue where the Group Chats, Blocklist, and New Contacts items were not displayed on the Contacts interface when a user is newly registered with no contacts.

Fixed the issue where the video sound continued to play after a user taps the Back button on a mobile phone.

Fixed the issue where the playing voice message did not stop and its sound was also recorded during voice message recording.

Fixed the issue where videos sent by iOS devices failed to playback on some mobile phones.

## 4.4.716 @2019.07.16

### iOS and Mac

Organized and merged APIs.

Added APIs to get the download URLs of file, video, and voice messages.

Added the `disableStorage` API to disable all local storage.

Fixed the issue where the conversation on the sender's device could still get lastMsg after an online message was sent.

Removed the return value of `getSenderProfile` , and used callback instead.

Changed the group function `modifyRecvMessageOpt` to `modifyReceiveMessageOpt` .

Fixed the issue where video screenshots sent from a device running iOS 2.X or 3.X to a device running iOS 4.X could not be obtained.

Fixed occasional crashes when data was reported upon exit.

Optimized the login module (repeated login/frequent login/frequent account switching/automatic connection/offline user being kicked off).

Fixed the issue where the unread count could not be cleared after a member left a group or a group was deleted.

Fixed the issue where group deletion notifications could not be received occasionally.

Fixed the issue where longer time was required to deliver messages when the app went to the foreground after staying in the background for a long time.

Optimized the one-to-one chat unread count.

Changed the input parameter `TIMLoginParam` of `autoLogin` to `userID` .

Changed the input parameter `TIMLoginParam` of `initStorage` to `userID` .

Removed multi-account login APIs: `newManager` , `getManager` , and `deleteManager` .

Fixed occasional respondsToLocator crashes.

Fixed occasional crashes caused by `TIMGroupInfo > lastMsg` calling related functions.

### TUIKit

Optimized the recent contact list update algorithm to reduce the refresh frequency.

Fixed blocklist memory leak.

Added message bubble and profile photo click event callbacks.

Fixed the issue where the latest profile photo was not displayed in recent contacts or the chat window.

Optimized document annotations.

## Android

Organized and merged APIs.

Added all APIs in `TIMManagerExt` to `TIMManager`.

Added all APIs in `TIMConversationExt` to `TIMConversation`.

Added all APIs in `TIMGroupManagerExt` to `TIMGroupManager`.

Added all APIs in `TIMMessageExt` to `TIMMessage`.

Added all APIs in `TIMUserConfigMsgExt` to `TIMUserConfig`.

Retained APIs in `TIMManagerExt`, `TIMMessageExt`, `TIMConversationExt`, `TIMGroupManagerExt`, and `TIMUserConfigMsgExt` classes provisionally for compatibility purposes, which will be deprecated in the future.

Added options to add friends in one-way or two-way manner.

Added the `disableStorage` API to disable all local storage.

Added APIs to get the download URLs of file, video, and voice messages.

Fixed the issue where `queryUserProfile` was null on some Android mobile phones.

Fixed the issue where the conversation on the sender's device could still get `lastMsg` after an online message was sent.

Removed the return value of `getSenderProfile`, and used callback instead.

Fixed occasional crashes when data was reported upon exit.

Optimized the login module (repeated login/frequent login/frequent account switching/automatic connection/offline user being kicked off).

Fixed the issue where the unread count could not be cleared after a member left a group or a group was deleted.

Fixed the issue where group deletion notifications could not be received occasionally.

Fixed the issue where longer time was required to deliver messages when the app went to the foreground after staying in the background for a long time.

Optimized the one-to-one chat unread count.

## TUIKit

Short video messages in chats can be played in landscape or portrait orientation.

Added support for Javadoc documentation.

Fixed the issue where downloading a video that was being sent failed.

Fixed the issue where the `onSuccess` callback of the `GroupChatManagerKit.getInstance().sendMessage` method could be triggered twice.

Fixed the issue with short audio messages on the chat interface. Audio messages should be at least 1 second long.

For messages shorter than 1 second, "Message too short" is displayed.

Fixed the issue where a user could be invited to join a private group repeatedly.

Fixed the issue where remarks could not be empty.

Fixed the issue where the time displayed on the chat interface was incorrect when the system time of the device was incorrect.

Fixed the issue where voice messages sent locally could not be downloaded on another mobile phone from roaming messages.

Fixed the issue where the group owner failed to set the group name to null but a message stating that the setting was successful was displayed.

### Windows

Fixed the issues where various platforms sent Chinese characters when image, file, audio, and video messages contained Chinese paths.

Fixed the issue where `TIMMsgReportReaded` was invalid.

Fixed the issue where the received message and recalled message have different rand and seq.

Fixed occasional crashes when data was reported upon exit.

Optimized the login module (repeated login/frequent login/frequent account switching/automatic connection/offline user being kicked off).

Fixed the issue where the unread count could not be cleared after a member left a group or a group was deleted.

Fixed the issue where group deletion notifications could not be received occasionally.

Fixed the issue where longer time was required to deliver messages when the app went to the foreground after staying in the background for a long time.

## Patch 4.4.631 @2019.07.03

### Android

Fixed offline push issues and crashes.

## 4.4.627 @2019.06.27

### iOS and Mac

Fixed the message sending timeout issue when no network connection was available.

Fixed the issue where the message ID value was changed after the message was sent.

Fixed the disordered message issue.

Fixed the issue where messages were lost when chat room historical messages were pulled.

Fixed the issue with incorrect system message types.

Fixed the issue where the obtained original image size of an image message was 0.

Fixed the issue where mobile phones failed to send messages after the system time was changed.

Fixed the issue where reporting conversation read and getting the unread count failed in some cases.

Fixed the issue where online messages that had been sent could be obtained through `getLastMessage` of the conversation.

Fixed the issue where getting `lastMsg` status through the conversation was exceptional after the last message was recalled.

Fixed the issue where recalled message content still existed in the conversation list of the peer.

Fixed the issue where the sending status of image/voice/file messages was exceptional after network reconnection.

Fixed the issue where login accounts that contained special characters could not send audio and images.

Fixed the issue where the V4 version could not get the width and height of thumbnails sent by the V2 version.

Fixed the issue where recent conversations failed to be pulled after `saveMessage` was created for a conversation.

Fixed the issue where `getMessage` failed to get the `MemberChangeList` content of group tips.

Fixed the issue when `getLoginStatus` failed to get the login status.

Fixed the issue where applicants became group members after their requests to join the group were rejected.

Fixed the issue where a log file existed under the root directory of the drive letter after a log path was set.

Mac: fixed the issue where the callback failed to be received in case of force offline.

#### TUIKit

Optimized the group management page logic.

Fixed the iOS 13 compatibility issue.

Fixed known issues.

#### Android

Fixed the message sending timeout issue when no network connection was available.

Fixed the issue where the message ID value was changed after the message was sent.

Fixed the disordered message issue.

Fixed the issue where messages were lost when chat room historical messages were pulled.

Fixed the issue with incorrect system message types.

Fixed the issue with exceptional progress value when files were downloaded.

Fixed the issue where mobile phones failed to send messages after the system time was changed.

Fixed the issue where the sending status of image/voice/file messages was exceptional after network reconnection.

Fixed exceptional message sorting after a group was deleted or a user was muted.

Fixed the issue where reporting conversation read and getting the unread count failed in some cases.

Fixed the issue where recalled message content still existed in the conversation list of the peer.

Fixed the issue where the status fetched by `getLastMessage` of the conversation was exceptional after the last message was recalled.

Fixed the issue where sent online messages could be obtained through `getLastMessage` of the conversation.

Fixed the issue where the obtained original image size of an image message was 0.

Fixed the issue where the V4 version could not get the width and height of thumbnails sent by the V2 version.

Fixed the issue where `getLoginUser()` could still get login users after they were forced offline.

Fixed the issue where `getSenderProfile` returned blank information.

Fixed the issue where `getOpUser` of `TIMGroupSystemElem` was empty.

Fixed the issue where `getMessage` failed to get the `MemberChangeList` content of group tips.

Fixed the issue where recent conversations failed to be pulled after `saveMessage` was created for a conversation.

Fixed the issue where a log file existed under the root directory of the drive letter after a log path was set.

Fixed known TUIKit issues.

### Windows

Fixed the message sending timeout issue when no network connection was available.

Fixed the issue where the message ID value was changed after the message was sent.

Fixed the disordered message issue.

Fixed the issue where messages were lost when chat room historical messages were pulled.

Fixed the issue with incorrect system message types.

Fixed the issue where the iOS IM SDK module of the cross-platform library did not include the ARMv7-A architecture.

Fixed the issue where empty messages were not supported by the `TIMMsgReportReaded` API of the cross-platform library.

Fixed the issue where multiple IM instances could run on one cross-platform library device with the same account and would be kicked off.

Added the JSON key for getting the unique ID of messages to cross-platform library messages.

Fixed the issue where a log file existed under the root directory of the drive letter after a log path was set.

Fixed the issue where `getMessage` failed to get the `MemberChangeList` content of group tips.

Fixed the issue where getting lastMsg status through the conversation was exceptional after the last message was recalled.

Fixed the issue where reporting conversation read and getting the unread count failed in some cases.

## 4.4.479 @2019.06.12

### iOS

Fixed the issue with message loss when offline messages were pulled.

Fixed the login failure caused by changing SDKAppID.

Fixed the issue where voice messages failed to play.

Fixed crashes caused by recalling group messages.

Fixed the 6002 error when getting friend lists and creating groups.

Improved the message sending efficiency.

Optimized the cache to mitigate UI lag.

TUIKit

New UI design

New architecture design

Improved features such as contacts, group management, and relationship chain.

Fixed bugs.

### **Android**

Fixed the issue with message loss when offline messages were pulled.

Fixed the login failure caused by changing SDKAppID.

Fixed the issue where voice messages failed to play.

Fixed crashes caused by recalling group messages.

Fixed the 6002 error when getting friend lists and creating groups.

Fixed Android device crashes caused by creating groups with too many members.

Improved the message sending efficiency.

Optimized the cache to mitigate UI lag.

TUIKit

New UI design

New architecture design

Improved features such as contacts, group management, and relationship chain.

Fixed bugs.

### **Windows**

Fixed the issue with message loss when offline messages were pulled.

Fixed the login failure caused by changing SDKAppID.

Fixed the issue where voice messages failed to play.

Fixed crashes caused by recalling group messages.

Fixed the 6002 error when getting friend lists and creating groups.

Optimized the cache to mitigate UI lag.

Improved the message sending efficiency.

## 4.3.145 @2019.05.31

### **iOS**

Fixed the issue where the same message was received after switching to another account.

Fixed crashes caused by getting one-to-one roaming messages after the ticket expired.

Fixed the issue where new chat room members could not see the chat history.

Fixed FindMsg crashes.

Optimized group message synchronization.

Fixed occasional getRecvMessageOpt errors.

### **Android**

Fixed the issue where the same message was received after switching to another account.

Fixed crashes caused by getting one-to-one roaming messages after the ticket expired.

Fixed the issue where new chat room members could not see the chat history.

Fixed the issue where the same message listener was added repeatedly.

Fixed FindMsg crashes.

Optimized group message synchronization.

### Windows

Fixed the issue where the same message was received after switching to another account.

Fixed crashes caused by getting one-to-one roaming messages after the ticket expired.

Fixed the issue where new chat room members could not see the chat history.

Optimized group message synchronization.

## 4.3.135 @2019.05.24

### iOS

Added the `checkFriends` API to verify friends.

Added the `queryGroupInfo` API to get local data.

Deprecated `getGroupPublicInfo` and replaced it with `getGroupInfo`.

Fixed the issue where deleted messages could be seen in the message list.

Fixed the issue where local messages could not be obtained before login.

Fixed the pulling quantity and sorting issues of recent contacts.

Fixed group message synchronization after network reconnection.

Fixed the issue where identifying duplicates failed when a large number of messages were received in a short time.

Fixed the issue where the same message might be received again after the app restarted.

Fixed occasional errors in initialization and message synchronization.

Fixed occasional errors caused when `lastMsg` of a conversation was deleted.

Fixed the issue where `onRefreshConversation` was called back twice with identical data.

Fixed the issue where users could not obtain the chat history of a chat room before the time they joined the chat room.

Fixed the issue where `copyFrom` of `TIMMessage` failed to work.

Fixed the issue where `TIMGroupEventListener` failed to receive callbacks.

Fixed crashes reported online.

Optimized connection requests during reconnection.

Optimized the quality of first connections to different networks and access points outside the Chinese mainland.

Improved the network reconnection speed when iOS devices switch to Wi-Fi networks.

### Android

Added the `checkFriends` API to verify friends.

Added the `queryGroupInfo` API to get local data.

Deprecated the `getGroupDetailInfo` and `getGroupPublicInfo` APIs and replaced them with the `getGroupInfo` API.

Fixed the issue where deleted messages could be seen in the message list.



Fixed `modifyGroupOwner` and `getGroupMembersByFilter` callback issues.

Fixed the issue where local messages could not be obtained before login.

Fixed the pulling quantity and sorting issues of recent contacts.

Fixed group message synchronization after network reconnection.

Fixed the issue where identifying duplicates failed when a large number of messages were received in a short time.

Fixed the issue where the same message might be received again after the app restarted.

Fixed occasional errors in initialization and message synchronization.

Fixed occasional errors caused when `lastMsg` of a conversation was deleted.

Fixed the issue where `onRefreshConversation` was called back twice with identical data.

Fixed the issue where users could not obtain the chat history of a chat room before the time they joined the chat room.

Fixed crashes reported online.

Optimized connection requests during reconnection.

Optimized the quality of first connections to different networks and access points outside the Chinese mainland.

## Windows

Added support for custom field data reporting.

Added online messages.

Added use cases for recalling messages.

Fixed occasional failures in setting upload files.

Fixed the issue where deleted messages could be seen in the message list.

Fixed the pulling quantity and sorting issues of recent contacts.

Fixed group message synchronization after network reconnection.

Fixed the issue where identifying duplicates failed when a large number of messages were received in a short time.

Fixed the issue where the same message might be received again after the app restarted.

Fixed occasional errors caused when `lastMsg` of a conversation was deleted.

Fixed occasional errors in initialization and message synchronization.

The JSON string of a delivered message is returned in the callback indicating successful delivery.

Replaced `TIMSetRecvNewMsgCallback` with `TIMAddRecvNewMsgCallback` and `TIMRemoveRecvNewMsgCallback`.

Added SOCKS5 proxy configuration.

Optimized connection requests during reconnection.

Optimized the quality of first connections to different networks and access points outside the Chinese mainland.

## 4.3.118 @2019.05.10

### iOS

Added `querySelfProfile` and `queryUserProfile` to the `TIMFriendshipManager` class (reading local data).

Fixed the issue where `getLoginUser` returned a login user exception.

Fixed the issue where online reported user profiles failed to be obtained.

Fixed the issue where some local fields became invalid after the app restarted.

Fixed occasional errors when calling read reports after messages were deleted.

Fixed the online reported IM group issue.

Fixed the issue with conversation unread counts.

Fixed the issue with online messages.

Fixed the issue where messages failed to be re-sent occasionally.

Fixed the issue where local ticket expiration caused repeated reconnection.

Fixed crashes reported online.

Optimized the server connection strategy.

Optimized the network reconnection strategy.

Optimized the server overload strategy.

Optimized heartbeat to reduce unnecessary outbound packets.

Added support for importing through CocoaPods for TUIKit.

Added the Contacts interface for TUIKit.

Added the Adding Friends interface for TUIKit.

Added the Blocklist interface for TUIKit.

Added the Search Friend interface for TUIKit.

Added the New Friends interface for TUIKit.

Added the Remarks, Blocklist, and Delete Friend features to the friend's profile page for TUIKit.

Added support for modification of nicknames, personal signature, date of birth, gender, and location on the user profile page for TUIKit.

Improve the group pinning feature for TUIKit.

## Android

Added `querySelfProfile` and `queryUserProfile` to the `TIMFriendshipManager` class (reading local data).

Added the `addTime` field when getting a friend's profile.

Added support for the x86 and x86\_64 architecture.

Fixed the issue where `getLoginUser` returned a login user exception.

Fixed the issue where online reported user profiles failed to be obtained.

Fixed the issue where some local fields became invalid after the app restarted.

Fixed occasional errors when calling read reports after messages were deleted.

Fixed the online reported IM group issue.

Fixed the issue with conversation unread counts.

Fixed the issue with online messages.

Fixed the issue where messages failed to be re-sent occasionally.

Fixed the issue where local ticket expiration caused repeated reconnection.

Fixed crashes reported online.

Optimized the server connection strategy.

Optimized the network reconnection strategy.

Optimized the server overload strategy.

Optimized heartbeat to reduce unnecessary outbound packets.

Added the "pin chat to top" feature to TUIKit.

TUIKit: nickname and personal signature can be changed, and the nickname is displayed on the profile page.

TUIKit: fixed the issue where emojis sent by iOS devices failed to be displayed on Android devices.

TUIKit: fixed the unread message red dot issue.

TUIKit: fixed the issue where a message appeared stating that UIs were abnormal after the plus sign was tapped on Meitu M8 mobile phones.

TUIKit: fixed the issue where profile photos were scaled down after being set and did not fill the entire UI.

TUIKit: fixed the login and auto login logic.

TUIKit: fixed the ANR issue when the input content exceeds the maximum limit.

TUIKit: fixed the issue where no response was received when images were selected from the photo album and the **OK** button on the preview screen was tapped.

TUIKit: fixed the issue where the message deleting and recalling buttons were not displayed after image messages were tapped and held on the chat interface.

TUIKit: optimized and fixed crashes reported online.

## Windows

Fixed the issue where `getLoginUser` returned a login user exception.

Fixed the issue where online reported user profiles failed to be obtained.

Fixed the issue where some local fields became invalid after the app restarted.

Fixed occasional errors when calling read reports after messages were deleted.

Fixed the online reported IM group issue.

Fixed the issue with conversation unread counts.

Fixed the issue with online messages.

Fixed the issue where messages failed to be re-sent occasionally.

Fixed the issue where local ticket expiration caused repeated reconnection.

Fixed crashes reported online.

Optimized the server connection strategy.

Optimized the network reconnection strategy.

Optimized the server overload strategy.

Optimized heartbeat to reduce unnecessary outbound packets.

## 4.3.81 @2019.04.24

## iOS

Fixed crashes caused by adding message elements to drafts.

Fixed the issue where some accounts failed to pull conversation lists after an app was removed and reinstalled.

Fixed the issue where login failed when usersig expired in login state and the app was not restarted.

Fixed the issue where messages could not be sent and the usersig expiration callback was not received when usersig expired in login state.

Fixed the issue with getting group member counts.

Fixed the request timeout issue (error code 6012).

## Android

Added:

Supplemented relationship chain features such as blocklist, friend list, and friend request handling of earlier version SDKs.

Fixed:

Fixed the issue where an error was reported when the main process of the app was killed.

Fixed the issue with getting group member counts.

Fixed issues with setting and getting custom group fields and custom group member fields.

Fixed the issue where no `onError` callback was sent after getting group profile timed out.

Fixed the issue where some accounts failed to pull conversation lists after an app was removed and reinstalled.

Fixed the issue where login failed when usersig expired in login state and the app was not restarted.

Fixed the issue where messages could not be sent and the usersig expiration callback was not received when usersig expired in login state.

Fixed disordered messages.

Fixed the request timeout issue (error code 6012).

Updated relationship chain error codes.

TUIKit: fixed a critical bug with the `DateUtils` class (GitHub issue #75).

TUIKit: fixed a crash (GitHub issue #86).

TUIKit: fixed issues with using SDK without permissions.

TUIKit: fixed crashes after deleting conversation, deleting message, and long-pressing.

TUIKit: fixed the issue where `popupwindow` would not disappear.

TUIKit: fixed the issue with repeated messages.

TUIKit: fixed the issue with intercepting empty messages containing whitespace.

TUIKit: fixed the issue where unread counts did not update after conversations were deleted.

TUIKit: fixed the issue with the maximum number of characters in a message.

TUIKit: improved experience and fixed several Array Index Out of Bounds exceptions.

## Windows

Fixed some crashes.

Fixed the request timeout issue (error code 6012).

Fixed the issue where some accounts failed to pull conversation lists after an app was removed and reinstalled.

Fixed the issue where login failed when usersig expired in login state and the app was not restarted.

Fixed the issue where messages could not be sent and the usersig expiration callback was not received when usersig expired in login state.

## 4.2.52 @2019.04.17

### iOS

Added:

Supplemented relationship chain features such as blocklist, friend list, and friend request handling of earlier version SDKs.

Fixed:

Optimized API annotations.

Fixed the issue with ineffective group custom fields and group member custom fields.

Fixed the issue where `TIMMessage` failed to get user profiles through `senderProfile`.

Fixed the issue with read receipt callback and status.

Fixed the issue where the last message did not call back when unread messages were synchronized.

Fixed the issue where group messages occasionally could not be received.

Fixed the issue where login response packets could not be decrypted.

Added support for IP connection and login information reporting.

Fixed the message seq error.

### Android

Added:

Supplemented relationship chain features such as blocklist, friend list, and friend request handling of earlier version SDKs.

Fixed:

Fixed jni leak on Android.

Fixed incorrect group member roles.

Fixed recalling group message crashes after a member left the group and joined the group again.

Fixed the issue where emojis were not displayed in the TUIKit demo.

Fixed the issue where the second page would often contain repeated messages when group chat messages were received.

Fixed some crashes in the TUIKit demo.

Fixed the issue where `TIMMessage` failed to get user profiles through `senderProfile`.

Fixed the issue with read receipt callback and status.

Fixed the issue where the last message did not call back when unread messages were synchronized.

Fixed the issue where group messages occasionally could not be received.

Fixed the issue where login response packets could not be decrypted.

Added support for IP connection and login information reporting.

Fixed the message seq error.

### Windows

Added:

Supplemented relationship chain features such as blocklist, friend list, and friend request handling of earlier version SDKs.

Fixed:

Fixed the issue where `TIMMessage` failed to get user profiles through `senderProfile` .

Fixed the issue with read receipt callback and status.

Fixed the issue where the last message did not call back when unread messages were synchronized.

Fixed the issue where group messages occasionally could not be received.

Fixed the issue where login response packets could not be decrypted.

Added support for IP connection and login information reporting.

Fixed the message seq error.

## 4.2.28 @2019.04.08

### iOS

Optimized issues related to unread counts.

Optimized message read status.

Fixed disordered one-to-one messages sent by RESTful APIs.

Fixed occasional repeated roaming messages fetched.

Optimized the `uniqueId` empty implementation issue.

### Android

Added:

Added the logic for adding, deleting, and querying friends.

Fixed:

Optimized issues related to unread counts.

Optimized message read status.

Fixed disordered one-to-one messages sent by RESTful APIs.

Fixed occasional repeated roaming messages fetched.

Optimized the `uniqueId` empty implementation issue.

### Windows

Optimized issues related to unread counts.

Optimized message read status.

Mitigated disordered one-to-one messages sent by RESTful APIs.

Fixed occasional repeated roaming messages fetched.

## 4.2.10 @2019.03.29

### iOS

New features

Added the logic for adding, deleting, and querying friends.

Fixed:

Mitigated the timeout issue.

Optimized the auto login logic.

Fixed crashes.

Fixed occasional network connection exceptions.

### Android

Mitigated the timeout issue.

Optimized the auto login logic.

Mitigated the JNI leak issue.

Fixed crashes.

Fixed occasional network connection exceptions.

### Windows

Mitigated the timeout issue.

Fixed crashes.

Fixed occasional network connection exceptions.

## 4.2.9 @2019.03.27

### iOS and Mac

Fixed crashes in the IPv6 environment.

Fixed the issue where setting profiles to int failed.

### Android

Fixed the issue where setting profiles to int failed.

## 4.2.1 @2019.03.15

### iOS

Fixed the issue where clients did not receive relevant instructions after a group was deleted in the backend.

Fixed the issue where calling `deleteConversationAndMessage()` failed.

Fixed the issue where no messages were received after network reconnection (On the conversation interface, messages can be proactively pulled after network reconnection.)

**Android**

Fixed incorrect group pending and processed requests returned.

Fixed client crashes when the client went to the backend.

Fixed the issue where no messages were received after network reconnection.

Fixed occasional message sorting errors.

Fixed the issue where messages occasionally failed to be sent.

**Web**

Web IM can play .amr recordings.

**Windows**

Added the /source-charset:65001 compilation option.

Fixed crashes when the file system directly ran IMAPP.exe.

Fixed various compilation errors and crashes.

Removed X64 compilation (not supported at present).

## 4.0.13 @2019.03.13

**Android**

Fixed crashes caused by login after 3.x is upgraded to 4.x.

**iOS**

pod can directly integrate the TUIKit.framework.

Fixed crashes caused by login after 3.x is upgraded to 4.x.

**Windows**

Added the IM demo with the duilib library as a UI component.

Added usage instructions and integration guide.

## IM SDK 4.0.12 2019-3-11

**iOS**

TUIKit.framework supports bitcode 2.

Fixed ineffective group muting.

Fixed the feature for modifying a user's role in a group.

**Android**

Fixed ineffective group muting.

Fixed the feature for modifying a user's role in a group.

Fixed the issue with modifying group message receiving options.

Fixed the issue with ineffective offline push toggle.



## IM SDK 4.0.10 2019-3-7

Fixed the message receiving error when an audio-video group had more than 100 members.

## IM SDK 4.0.8 2019-3-6

Optimized the audio playback logic for TUIKit.

## IM SDK 4.0.7 2019-3-1

Fixed the compatibility issue with audio, file, and video messages between earlier and later versions.

Fixed "-5 tls exchange failed" where login was successful after uninstalling and then reinstalling the app.

## IM SDK 4.0.4 2019-2-28

Fixed the issue where an incorrect error code was returned when a user logged in after userSig expired. The correct error code is 6206.

Optimized the force offline logic.

## IM SDK 4.0.3 2019-2-25

Fixed the issue with third-party offline push.

## IM SDK 4.0.2 2019-2-20

Fixed the issue where bitcode packaging activation failed.

## IM SDK 4.0.1 2019-2-20

Fixed the issue where -5 is returned after login.

## iOS--IM SDK 4.0.0.1 2019-1-21

Added TUIKit.

## IM SDK 3.3.2 2018-7-5

Automatic read reporting is disabled by default.

Custom information types of profile relationship chains support integer.

Fixed the issue where the group member count obtained from local storage was incorrect.

Fixed the issue where the nickname carried in one-to-one chat messages was not updated in real time.

## IM SDK 2.7.2 2018-7-5

Automatic read reporting is disabled by default.

Custom information types of profile relationship chains support integer.

Added the message recalling feature.

Fixed the issue where the nickname carried in one-to-one chat messages was not updated in real time.

## Windows--IM SDK 2.5.8 2018-7-5

Fixed login failures in some cases.

Custom information types of profile relationship chains support integer.

## IM SDK 3.3.0 2018-4-4

### iOS

Added the level and role fields to `TIMUserProfile` .

### Android

Added support for offline push on Meizu mobile phones.

Added standard level and role attributes to user profiles.

Fixed the issue where UGC short video failed to be sent when a user logged in after logging out.

## IM SDK 2.7.0 2018-4-4

### iOS

Added custom data parameters to the API for inviting users to join a group.

## Android

Added support for offline push on Meizu mobile phones.

Added support for custom data for the API for inviting users to join a group.

## Windows--IM SDK 2.5.7 2018-3-13

Modified the login module to improve communication security.

Improved the message delivery capability with poor network connection.

Fixed occasional crashes when logs were printed.

## iOS--IM SDK 2.6.0 2018-3-13

Provided an API for deleting roaming messages.

Provided an API for serializing and deserializing message objects.

Fixed some known issues.

## iOS--IM SDK 3.2.0 2018-3-13

Fixed the issue where an error was reported when `getUserProfile` contained custom friend fields.

Optimized the group unread count update strategy.

Optimized the logic and strategy for local message storage.

Fixed some crashes.

## Android--IM SDK 3.2.0 2018-3-13

Fixed the issue where UGC short videos failed to be sent.

Fixed the issue with no callbacks for sent messages when the network connection is interrupted.

Fixed the issue where muting all did not take effect.

Optimized the logic and strategy for local message storage.

Fixed some crashes.

## Android--IM SDK 2.6.0 2018-3-13

Provided an API for deleting roaming messages.

Provided an API for serializing and deserializing message objects.  
Fixed some known issues.

## IM SDK 3.1.2 2017-12-12

Mitigated the network timeout issue on Android devices.  
Fixed the audio download error on Android devices.  
Fixed various crashes on Android devices.

## IM SDK 2.5.7 2017-11-08

Fixed SDK crashes when app processes were killed.  
Fixed the issue where offline messages were repeatedly pushed.  
Fixed the issue where internal accounts may be empty when `initStorage` and `login` are called at the same time.  
Optimized the network detection strategy.  
Fixed the error in getting friend lists.  
Fixed some crashes.

## IM SDK 3.1.1 2017-8-16

Optimized the regular log clearing mechanism.  
Fixed the issue where iOS QALSDK crashed upon initialization.  
Added the feature for muting all group members.  
iOS: fixed the multi-user login failure.  
Android: fixed crashes caused by getting group lists before login.

## IM SDK 2.5.6 2017-7-14

Fixed crashes during login and logout.  
Fixed crashes during push and recording.

## IM SDK 3.1.0 2017-7-3

Added IMUGCExt.framework and TXRTMPSDK.framework to provide short video recording and upload.  
Added the Recall Message feature.

## IM SDK 2.5.5 2017-6-6

Optimized the logic for internal response packets to reduce time consumption.  
Improved the log time granularity to millisecond.  
Fixed some crashes and message synchronization issues.

## IM SDKV3 3.0.2 2017-5-22

Fixed the issue where users cannot receive group messages in an audio-video group.  
Adjusted APIs.

- i. Deprecated `TIMFileElem` and the `setData` API in `TIMSoundElem` .
- ii. Corrected spelling of the `getConversionList` API in `TIMManagerExt` to `getConversationList` .

## IM SDKV3 3.0.1 2017-5-15

Fixed the issue where some .so libraries were incompatible with devices running systems earlier than Android 5.0.

## IM SDKV3 3.0 2017-5-8

Regrouped IM SDK and IMCore into IM SDK, IMMessageExt, IMGroupExt, and IMFriendExt.

Optimized the IM SDK initialization method to `initSdk`: and `setUserConfig`.

Names of IM SDK APIs and protocol callback methods start with lowercase letters.

IM SDK features: basic login, receiving and sending messages, profile, and group features

IMMessageExt features: full message features, including message pulling, local storage, and unread count

IMGroupExt features: full group features, including group type management and group member management

IMFriendExt features: full relationship chain features, including friend list and blocklist

## IM SDK 2.5.4 2017-4-28

Fixed the timer mechanism bug in the IM SDK.

## IM SDK 2.5.3 2017-4-17

### iOS

`sendOnlineMessage` supports group messages, which will not be saved to local storage, stored offline, or included in the unread count.

Added the `findMessages` method to get local messages by message ID.

`TIMIOSOfflinePushConfig` provides the option for setting APNs push muting.

Fixed the issue of excessive memory consumption when messages were received at high frequency.

### Android

Added the API for searching for messages. (For more information, see `findMessages` under `TIMConversation`.)

`sendOnlineMessage` supports group messages, which will not be saved to local storage, stored offline, or included in the unread count.

Added the configuration item that allows a device to receive APNs push notifications without playing a sound or vibration. (For more information, see `TIMMessageOfflinePushSettings.IOSSettings.NO_SOUND_NO_VIBRATION`.)

Optimized networking to improve SDK robustness in poor network connection.

### Windows

Fixed issues that may cause crashes.

### API changes:

Changed how `TIMMessageOfflinePushSettings.AndroidSettings` and `TIMMessageOfflinePushSettings.IOSSettings` are constructed.

For more information, see [Offline Push](#)

## IM Android SDK 2.5.2 2017-3-1

Fixed the issue where the return of outgoing packets occasionally timed out (return code 6205).

## IM SDK 2.5.1 2017-2-16

Limited the maximum size of log files to 50 MB.

Fixed the bug where the online state was returned after a user logged out and the app went to the backend.

iOS: updated the audio and file downloading strategy and supported HTTP and HTTPS download.

Fixed the status mismatch bug after messages failed to be sent when the user was not logged in.

## IM Web SDK 1.7 2016-12-20

Added support for multi-instance force offline.  
Added support for simultaneous online of multiple instances.  
Added support for synchronization of read group messages.  
Added support for synchronization of read one-to-one messages.  
Optimized the demo directory structure and code.  
Added the recent contacts list.

## IM SDK 2.5 2016-12-16

Optimized the `TIMOfflinePushInfo` object structure.  
Fixed audio and file download failures in iOS 9.1.  
Optimized network operations.  
Fixed some bugs.

## IM SDK 2.4.1 2016-11-24

Fixed the bug where `TIMGroupAssistant` exceptionally pulls the group profile after entering an audio-video group.  
Fixed the bug where disabling console print failed.  
Fixed the issue where various listeners became invalid when logout is called before login after initialization.

## IM SDK 2.4 2016-11-09

Full compatible with the ATS mode.  
Message forwarding feature: the `copyFrom` API forwards image and file messages by copying images and files without downloading them.  
The number of members in an audio-video group is dynamically updated. `TIMGroupEventListener` returns the current number of group members.  
Message filtering can be customized for audio-video groups.  
`TIMOfflinePushInfo` attributes support push notification settings of Mi and Huawei mobile phones.  
Optimized the process of pulling group roaming messages.  
Optimized the processes of uploading and downloading audio, files, and short videos.  
Throwing `onNewMessage` when pulling the recent contacts list can be disallowed.

## IM SDK 2.3 2016-9-13

Added support for push notifications to multiple apps with one appid.

Added `setOfflinePushToken` with callback to the Android version.

Optimized the message deletion logic to automatically filter messages in the DELETED state when messages were pulled.

iOS: moved database files from subdirectory Library/Caches/ to subdirectory Document/ to prevent them being cleared by the system.

Multiple TIMMessageListeners can be added and deleted in iOS versions.

Resident threads in iOS versions are named in a unified manner.

The API for getting conversation lists automatically filters conversations with the message count set to 0.

## IM Web SDK 1.6 2016-8-15

Web broadcast message requirements

Added friend system notifications.

Added profile system notifications.

## IM SDK 2.2 2016-8-10

Added support for conversation drafts.

Conversations can be marked whether to store messages to ensure more flexible message handling.

Roaming messages can be traversed from old to new, which applies to scenarios where message recording is needed.

Added ext and sounds of push notifications to messages, allowing setting push information for some messages.

Added `stopQALService` to the Android SDK, which turns off QALService when exiting the app.

Added support for network status monitoring and added error codes for network errors.

## IM SDK 2.1 2016-7-15

Added support for notification push to Mi and Huawei mobile phones.

Added support for the read receipts feature, which is optional depending on product needs.

Added support for typing reminder, which is optional depending on product needs.

Added standard fields such as the gender, date of birth, address, and language to profile relationship chains.

Notifications for joining and quitting a group contain the group member count.



Fixed some SDK and demo bugs.

## IM Web SDK 1.5 2016-7-13

Merged broadcasting chat room SDK capabilities.

Fixed issues with uploading images in Internet Explorer 8 and 9.

Added a group member count field to tips for joining and leaving groups.

Fixed some SDK and demo bugs.

## IM SDK 2.0 2016-6-16

The unread count can be synchronized between multiple online devices.

Historical messages can be imported when an app is migrated to ensure smooth migration.

Added the message notification status to group message attributes.

Added support for flexible settings for message priorities.

Push notifications can be filtered by attribute and tag.

## IM Web SDK 1.4 2016-6-7

Friends' message history can be pulled.

Red packets and like messages can be sent.

The API for creating groups supports custom group IDs and broadcasting chat rooms.

Optimized SDK APIs and merged the login and initialization APIs.

Optimized the demo directory structure and code.

## IM SDK 1.9.3 2016-5-31

Fixed resource destruction deadlocks when the winsdk process exited.

## IM SDK 1.9.2 2016-5-27

Added the ticket expiration callback.

Added support for IPv6 (iOS).

## IM SDK 1.9 2016-5-4

Added support for groups with more than 10,000 members (no limit on the number of members, which is suitable for broadcasting scenarios).

Reconstructed the IM demo for better experience and ease-of-use.

Messages can be sent based on their priorities.

Added storage and cache for group profiles and relationship chains.

Added APIs to synchronize group profiles and relationship chains and change callbacks.

Added support for getting friend profiles, including remarks and lists.

Added support for setting default group profile and relationship chain fields to be pulled.

Added support for disabling pulling recent contacts.

Added synchronizing the last message to the conversation list.

You can specify the group members whose group information, such as group name cards, is to be pulled.

Added support for passing in file paths for voice and file messages (messages can be resent).

Adapted to Android 6.0 dynamic permission management.

## IM SDK 1.8.1 2016-4-13

Android: optimized the auto-start process. (To modify configuration, see ReadMe.txt.)

Added the API for sending online messages in one-to-one chats. (The messages will be received only when the receiver is online and will not be stored when the receiver is offline.)

Added the API for batch sending messages.

Optimized Android performance.

## IM SDK 1.8 2016-3-23

Android offline push

Added the API to verify friend relationships.

Added the relationship chain custom field API.

Messages can be customized for local storage (for example, audio can be identified as read or unread).

Added the API to compress images, meeting the need for image compression in detached communication scenarios.

Customized messages' sound fields to specify APNs sounds.

Optimized callback APIs for online status change.

## IM SDK 1.7 2016-1-25

Added support for limiting the message sending frequency in groups.

Added support for group ownership transfer.

Group message notification intensity can be customized.

CS channels are established to remove the need for a persistent connection between the app and backend to reduce battery consumption.

Added configuration items, including message and recent contacts roaming switch, storage duration, and multi-device online switch to improve operational efficiency.

Downstream messages carry group member nicknames and contact cards to improve user experience and ease-of-use.

Simplified the SDK to reduce the installation package size.

## IM SDK 1.6 2015-12-25

Short video messages are supported to meet growing needs for video messages and social communication.

Added support for rule-based sorting of group members.

Added support for relationship chain friend lists.

Added support for group member contact cards to help users identify group members.

Added support for the message notifications switch, allowing users to turn on or off message notifications for one-to-one chats and group chats.

## IM SDK 1.5 2015-11-16

Added support for asynchronous download of message records.

Group messages can be deleted at the server side.

Users can be searched by nickname.

Groups can be searched by group name.

Event callbacks can be configured in the console.

User credentials of admin accounts can be downloaded.

Optimized some demo and technical logic.

## IM SDK 1.4 2015-10-16

Multi-device login is supported.

Messages from blocklisted users cannot be received.

Deleted friend recommendations.

APNs pushes nicknames.

Demo supports the guest mode and third-party account login.

## IM SDK 1.3 2015-09-10

Users can log in as guests without usernames and passwords.

Message roaming is supported. (Messages are stored for seven days by default.)

Recent contacts roaming and deletion are supported.

Real-time message synchronization through callbacks is supported.

Friend recommendation is supported after the recommendation logic has been defined.

Sending original images or thumbnails is supported for better user experience.

Added support for push notifications (available only to online Android users).

Added support for smooth migration.

Added support for deleting local messages to protect users' privacy.

## IM SDK 1.2 2015-08-18

One-to-one chats on the web platform are supported.

The maximum number of group members is increased to 10,000.

Added an API that provides message IDs to precisely locate messages.

Added remarks to user profiles.

Added support for viewing local messages when offline.

## IM SDK 1.1 2015-07-13

Windows C++ platform is supported.

Public groups and chat rooms are supported.

Added support for adding group introductions and announcements and added muting, message block, and group role setting.

Added APIs for user profile and relationship chain operations, such as setting nicknames, adding friends, and setting a blocklist.

Added support for file messages.

Optimized image messages: image quality includes the original image, thumbnail, and large image. Changed upload and download APIs. Image URLs can be passed.

Added log levels to the log callback API.

Added the logic to execute forced logout on one device in the event of repeated logins.

Added automatic crash reporting.

Added support for self-owned account and third-party account integration in hosting mode.

Added SMS authentication for user registration and login.

Added support for ticket verification using public keys and private keys generated by Tencent.

Added user and group management.

## IM SDK 1.0 2015-05-11

Added support for Android/iOS platforms.

Added support for integrating Tencent account and third-party account logins.

Added support for one-to-one chats and group chats (discussion groups).

Added support for text, emoji, image, audio, location, and custom messages.

APNs push notifications (token reporting, foreground and background switching event reporting)

Added the feature of storing messages locally.

# Flutter

Last updated : 2024-06-19 11:07:07

## Platform Support

We are committed to building a set of Chat SDK and TUIKit for all Flutter platforms, allowing you to run one set of code across all platforms.

Platform	Low-level SDK ( <a href="#">tencent_cloud_chat_sdk</a> )	UIKit ( <a href="#">tencent_cloud_chat_uikit</a> )	UIKit V2 ( <a href="#">tencent_cloud_chat</a> )
iOS	Supported by all versions	Supported by all versions	Supported by all versions
Android	Supported by all versions	Supported by all versions	Supported by all versions
<a href="#">Web</a>	Supported from v4.1.1+2	Supported from v0.1.5	<i>Will be supported by the following version</i>
<a href="#">macOS</a>	Supported from v4.1.8	Supported from v2.0.0	Supported by all versions
<a href="#">Windows</a>	Supported from v4.1.8	Supported from v2.0.0	Supported by all versions
<a href="#">Hybrid development</a> (Adding Flutter SDK to Existing Native Applications)	Supported from v5.0.0	Supported from v1.0.0	Supported by all versions

**Note :**

For web, macOS, and Windows platforms, you need to perform a few extra steps for integration. For details, see [Support for the Flutter for Web](#) and [Support for the Flutter for Desktop](#).

## Description

**Chat Flutter SDK (Low-level SDK)** refers to the [tencent\\_cloud\\_chat\\_sdk](#) package, a low-level Chat SDK, encompassing all client APIs and listener callbacks.

**Chat Flutter UIKit** refers to the [tencent\\_cloud\\_chat\\_uikit](#) package. Built upon the Low-level SDK, it also includes a comprehensive UI component library and business logic. **This is the earlier version of UIKit** that we launched at the beginning of 2022. By early 2024, all planned capabilities were fully developed, entering a stable version mode. Future updating will focus on detail optimization and compatibility with new versions of Flutter and development kits.

Th**Chat Flutter UIKit V2** refers to the **brand-new Chat UIKit** that is based on the [tencent\\_cloud\\_chat](#) package and includes various modular UI components such as `[tencent_cloud_chat_conversation]` ([https://pub.dev/packages/tencent\\_cloud\\_chat\\_conversation](https://pub.dev/packages/tencent_cloud_chat_conversation) !01186af0050afe1d3c2d7c04a863199e) and [tencent\\_cloud\\_chat\\_message](#). These components can be custom-assembled and used in combination. Compared to the old UIKit, the new one offers a more developer-friendly experience, a better integration method, more refined interface details, richer animations and interactions, support for dark mode, and stronger internationalization capabilities. It was officially released in early 2024, and future iterations will focus on this version.

## Update Logs

### Chat Flutter UIKit V2 - 1.4.0 @2024.06.14

#### General

**[Breakthrough]:** Added comprehensive support for **Web**, including both Mobile and Desktop browsers.

Added support to return the login status for the `initUIKit` method.

Added `addGlobalCallback` and `removeGlobalCallback` to `TencentCloudChatCoreController`, enabling the integration and management of custom `TencentCloudChatCallbacks` throughout your codebase. Improved device screen recognition logic.

#### Message (TencentCloudChatMessage)

**[Breakthrough]:** Added integration support of the new **Sticker Plugin**, allowing users to send and view a variety of stickers and emojis.

Added support for **Community and Topic** chats. Users can now participate in these group types. Added `topicID` to `TencentCloudChatMessageOptions` to specify the chat topic along with its corresponding `groupID` for community identification.

Added a unified format for all builders in `TencentCloudChatMessage`, modifying their parameters. The previous builders have been removed. Each builder in `TencentCloudChatMessageBuilders` now includes four standardized parameters: `Key? key`, `widgets`, `data`, and `methods`, enhancing usability and comprehension. Builders from other components will be updated to this unified format in future versions.

Added support to select message text on desktop, facilitating easier copying of entire or partial text messages.

Added markdown parsing support for text messages, which is disabled by default. Additionally, URLs in text messages can now be launched directly.

Added new customization options to `TencentCloudChatMessageConfig` :

`mentionGroupAdminAndOwnerOnly` , `showMessageSenderName` , `enableParseMarkdown` ,  
`enableAutoReportReadStatusForComingMessages` , `enableReplyWithMention` ,  
`attachmentConfig` , `additionalAttachmentOptionsForMobile` ,  
`additionalInputControlBarOptionsForDesktop` ,  
`defaultMessageSelectionOperationsConfig` , and `additionalMessageMenuOptions` .

Added `beforeMessageSending` and `beforeRenderMessageList` hooks to

`TencentCloudChatMessageLifecycleEventHandlers` , and added `onTapLink` ,  
`onPrimaryTapAvatar` , and `onSecondaryTapAvatar` to  
`TencentCloudChatMessageUIEventHandlers` for enhanced business logic customization.

Added new methods to `TencentCloudChatMessageController` : `updateMessages` ,  
`mentionGroupMembers` , `setMessageTextWithMentions` , and `scrollToBottom` , providing greater  
control over the component.

Added support to re-edit recalled messages.

Improved the UI display for text messages, particularly the time and status indicators, message bubble width, and more.

Improved the icons in both the message context menu and input attachment actions, replacing Material Icons with custom-designed icons.

Improved the display of quoted/replied messages in the message list, and changed the interaction for navigation.

Improved the logic for message replies and sending, including group member mentions, auto-focus when replying, scrolling to the bottom after sending a message, and media message sending.

Improved the performance of the message list, message status updates, and group member lists in large groups.

Fixed an issue where calls could not be initiated via message header actions.

Fixed video preview and playback issues.

Fixed various UI display errors, including boundary issues.

Fixed an issue where the message list could not be scrolled using a laptop touchpad.

Fixed an issue where recording on mobile phone may lose control in some cases.

## Chat Flutter UIKit V2 - 1.3.0 @2024.04.17

### General

Enhanced the `initUIKit` function with several improvements, streamlining the configuration process and increasing overall usability. Key updates include:

A new `components` parameter that consolidates component-related configurations, including the required `usedComponentsRegister` for manually declaring used components. It also allows for optional global configurations, builders, controllers, and event handlers for each component, affecting all instances of each component.



A new `onTencentCloudChatSDKEvent` callback within the `callbacks` parameter handles SDK-related events, replacing the previous `sdkListener` from `options`.

The `config` parameter now focuses on global configurations for the UIKit, removing `usedComponentsRegister` and `preloadDataConfig`. The `usedComponentsRegister` has been moved to the `components` parameter.

Removed the requirement for passing in `context`.

Introduced a new manager for each component, named by appending `Manager` to the component's name (e.g., `TencentCloudChatMessageManager`), providing the following functions for better and easier integration:

`register` : [Manually declaring the usage of each component](#) during the `initUIKit` call.

`controller` : [Taking control of each component](#) on a global scale.

`eventHandlers` : [Handling component-level events](#) on a global scale.

`builder` : [Dynamically updating UI builders](#) for all instances.

`config` : [Configuring components](#) for all instances dynamically.

Migrated the `register` from the `Instance` of each component to the `Manager`, as described in the previous point.

Refined the core data storage structure and performance, paving the way for future feature enhancements.

### Message (TencentCloudChatMessage)

Changed the default configurations for `enabledGroupTypesForMessageReadReceipt` to be empty. This requires specifying the group types for which the message read receipt feature should be enabled, after enabling them in the console.

Fixed an issue where the message read status could not be updated dynamically for one-to-one chats.

Reduced the number of rebuilds for message list items to improve performance.

## Chat Flutter UIKit V2 - 1.2.1 @2024.04.02

### Conversation (TencentCloudChatConversation)

Fixed an issue where the main component was released on desktop after switching login accounts.

Changed the button `Mark as Unread` to `Mark as Read` and implemented its functionality.

### Message (TencentCloudChatMessage)

Fixed an issue where some configurations in `TencentCloudChatMessageConfig` were invalid.

Fixed an issue where a permission request failed on both iOS and Android devices when installing the app for the first time. Also fixed an issue where permissions could not be manually enabled in settings.

Improved the message locating for the original message of a replied message.

Fixed an issue where messages in the message list could not be dynamically received and displayed in some cases, especially after login account switching.

Fixed an issue related to voice message playback.

Displayed a default avatar for users without a profile picture.

## Chat Flutter UIKit V2 - 1.2.0 @2024.03.28

### General

Added support for **tablet devices**, including adaptive UI for iPad and various Android tablets. Now you can deploy to all platforms (mobile, tablet, desktop, web) at once using a single codebase.

Introduced **callback** functionality, allowing handling of SDK API errors and specific UIKit events that require user attention with `eventCode` and `text` by default, on a global scale. Developers can use UIKit with

```
TencentCloudChatCoreController.initUIKit()
```

 and set the callbacks accordingly.

Enhanced global dialog styles for Apple devices with a more native Cupertino style.

Optimized global data storage structures and improved underlying performance.

Ensured that all data from the previous account is removed from memory after logging out, and no data remains when logging in with a new account.

Replaced the original `logout` method with the `resetUIKit({bool shouldLogout = false})` method in `TencentCloudChatCoreController` to ensure no data remains in UIKit after logout and avoid logging out twice after being kicked off. For specific usage, refer to the comment.

Added SVG support for avatars.

### Conversation (TencentCloudChatConversation)

Optimized time display in conversation items to improve readability.

Fixed an issue where the conversation unread counts could not be updated dynamically.

### Message (TencentCloudChatMessage)

Added support for long press on mobile devices and navigate to the original message by clicking quoted message on desktop devices.

Improved message location and navigation capabilities, including jumping to specific messages. Optimized performance and user experience. This capability is exposed by the `scrollToSpecificMessage` method in `TencentCloudChatMessageController`, which allows control to navigate to specific messages and optionally highlight the target message.

Removed the ability to download large images and view original images in image preview mode.

Optimized the calculation of message long-press menu height to improve accuracy, avoiding situations where menu items are not fully displayed. Also improved animation performance.

Added `showMessageTimeIndicator`, `showMessageStatusIndicator`, `defaultMessageSelectionOperationsConfig`, and `defaultMessageMenuConfig` to

`TencentCloudChatMessageConfig` for better customization of message bubbles, message selection menus, and message menus. For specific usage, refer to the comments for each parameter.

Removed `useGroupMessageReadReceipt` from `TencentCloudChatMessageConfig`. Please use `enabledGroupTypesForMessageReadReceipt` instead.

Improved the display position of text message status and time indicators, no longer occupying a separate column.

Enhanced the default time separator in the message list to support localized and internationalized date and time representations.

Resolved issues related to media preview and voice messaging functionality.

Fixed several bugs, reduced redundant page builds, improved performance, and minimized CPU and memory resource usage.

## Contact (`TencentCloudChatContact`)

Resolved the issue of contact names being too long and overflowing the boundaries.

## Chat Flutter UIKit V2 - 1.1.2 @2024.03.13

### General

Further enhanced the integration process.

Optimized screen type recognition logic for better adaptation to different screen types.

### Conversation (`TencentCloudChatConversation`)

Added a new `onTap` event, `onTapConversationItem`, to `TencentCloudChatConversationUIEventHandlers` of `TencentCloudChatConversationEventHandlers` on the `eventHandlers` in the `TencentCloudChatConversation` component. This allows for custom event handling when a conversation item is clicked. If it returns `false`, the default logic will be executed, navigating to the corresponding `TencentCloudChatMessage` widget.

Introduced a new builder, `conversationHeaderBuilder`, for customizing the header bar.

### Message (`TencentCloudChatMessage`)

Enhanced message list with localized date and time indicators, adapting to user's language settings for a localization experience.

## Chat Flutter UIKit V2 - 1.1.1 @2024.03.11

Reduced the number of steps, increasing the success rate of one-time integration, lowering the barrier to entry, and enhancing the integration process.

## Chat Flutter UIKit - 2.5.0 @2023.2.28

## Breaking Changes

Migrated to Flutter 3.19. Support for Flutter 3.16 and earlier versions has been discontinued.

## Notes

Starting from Flutter 3.19, it is recommended to apply Flutter's Gradle plugins using Gradle's declarative plugins `{}` block (also known as the Plugin DSL) ( [see details](#)).

In line with this, our sample app on the GitHub repo has also been migrated to this new approach. If you'd like to migrate to this new approach, see our [sample app repo](#).

## Chat Flutter UIKit V2 - 1.1.0 @2024.1.23

The entirely redeveloped and revamped version of UIKit has been officially released. Compared to the previous version of UIKit, there are several significant enhancements and optimizations, which include:

1. **Theme Customization:** Switch between **light and dark mode** on the fly, or customize your own theme with our rich set of options.
2. **Internationalization:** We've added support for more languages including Arabic, and introduced a **Middle Eastern UI**. Our powerful and user-friendly [localization tools](#) make it easier than ever to [customize the localization configuration and the translation](#), and provide a localized user experience.
3. **Performance Enhancements:** We've made significant improvements to the **performance of the message list**, and introduced efficient and precise message positioning capabilities.
4. **Multimedia Support:** Experience improved multimedia and file message handling, with continuous playback for voice messages and **swipeable multimedia message previews**.
5. **Detail Optimizations:** We've added numerous detail optimizations including rich animations, haptic feedback, and refined interfaces to enhance the user experience.
6. **New Features:** Enjoy new features like a grid-style avatar, redesigned forwarding panel, group member selector, and a new message long-press menu.
7. **Modular Packages:** Components are broken down into **modular packages**, allowing for on-demand imports and reducing unnecessary bloat.

Each modular package supports built-in navigation. For instance, you can automatically navigate from a Conversation to a Message to start a chat,

without the need to manually instantiate multiple pages and handle the transitions yourself. This greatly simplifies the complexity of development and integration.

8. **Developer-friendly Design:** We've introduced a more unified, standardized component parameter design, clearer code naming, and more detailed comments to make development easier and more efficient.

## Chat Flutter UIKit - 2.4.0 @2023.11.28

### Breaking Changes

Migrated to support Flutter 3.16.0.

Upgraded the minimum supported Android Gradle Plugin to the 7.3 version to meet Flutter requirements.

## Chat Flutter SDK (Low-level SDK) - 6.0.2-6.0.8 @2023.11.03

Fixed several bugs.

## Chat Flutter UIKit - 2.3.3 @2023.10.30

### New Features

Added a new lifecycle hook of `MessageListShouldmount` .

### Bug Fixes

Fixed an issue on time tag creator.

## Chat Flutter SDK (Low-level SDK) - 6.0.1 @2023.10.24

Non-friend user profile update listener

Non-friend user profile update callback

Group-wide mute callback

Group member tag & group member tag callback

Added message response interface and the corresponding callback

Added message recall with recall information callback

Set global message receiving options

Message cloud search

Conversation delete callback

Callback for unread count statistics by specified category in conversation

Delete conversations in batches

Get unread count of conversations by category

Listen for unread count changes by specified type

Clear conversation unread count (markxxxAsRead interface deprecated)

Improved offline push fields

Voice-to-text

## Chat Flutter UIKit - 2.3.2 @2023.09.27

### Improvements

Enhanced message list performance.

### Bug Fixes

Fixed an issue that prevented the group member addition/removal modal box from closing.

Addressed several other bugs.

## Chat Flutter UIKit - 2.3.1 @2023.09.13

### Bug Fixes

Resolved an issue that history messages can't be removed after a conversation was deleted.

Fixed an issue that the Android users were prevented from opening files sent by themselves.

## Chat Flutter UIKit - 2.3.0 @2023.08.30

### Breaking Changes

Upgraded and migrated to support Flutter 3.13.0. Support for Flutter 3.10 and earlier versions has been dropped.

### Recommendations

Customers who do not wish to upgrade to Flutter 3.13.0 are advised to continue using version 2.2.1 of our Chat UIKit.

However, we strongly recommend upgrading to Flutter 3.13.0 as it includes numerous performance improvements and introduces cutting-edge features.

## Chat Flutter UIKit - 2.2.1 @2023.08.29

### New Features

Introduced a new `groupMemberList` configuration in `TUIKitChat`; when specified, TUIKit will not load it automatically, optimizing network traffic usage.

Added support for image copying on desktop platforms.

### Bug Fixes

Fixed an issue preventing the removal of image loading status.

Resolved a problem that prevented images from being saved to the device gallery.

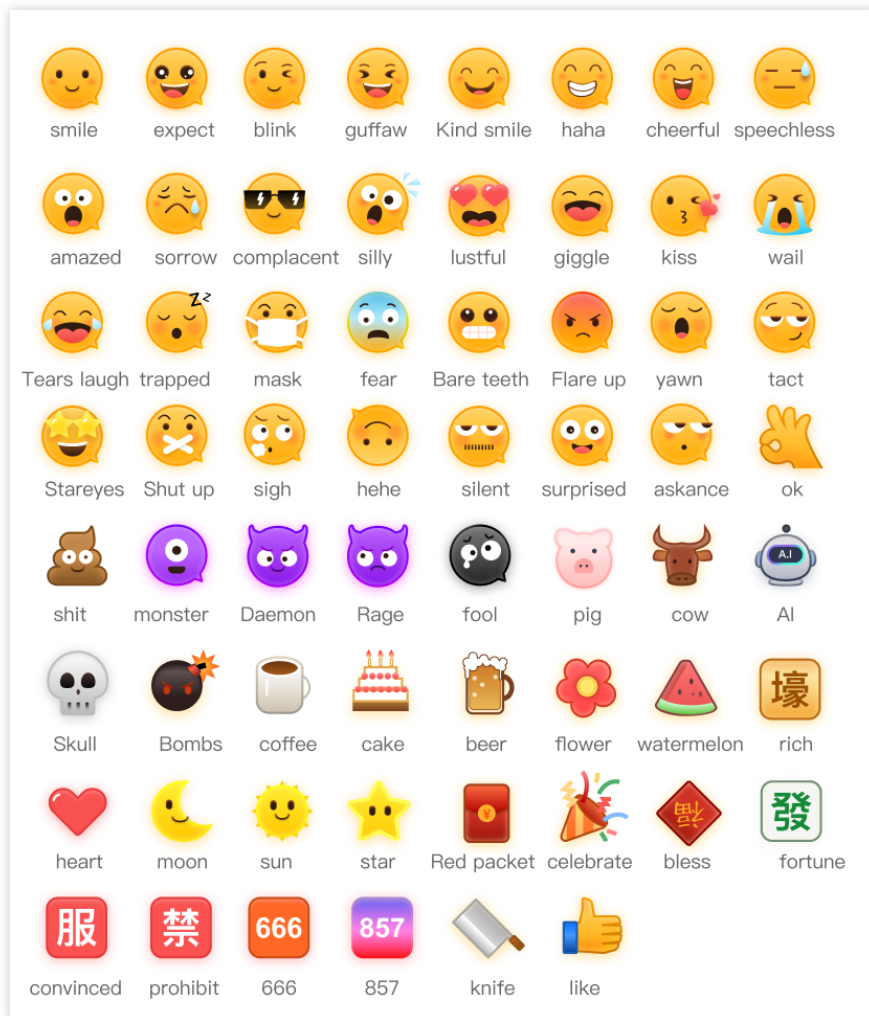
Addressed a potential issue causing the `mentionOtherMemberInGroup` function in `TIMUIKitChatController` to fail.

Corrected an issue that could lead to improper image rendering.

## Chat Flutter UIKit - 2.2.0 @2023.08.18

### New Features

Introduced a newly-designed set of Emoji image stickers, available for seamless integration within textual content, providing an enhanced user experience.



Streamlined the implementation of stickers, removing the need for additional complex coding. Full functionality is enabled by default, with customization options available through the `stickerPanelConfig` configuration in `TIMUIKitChatConfig`.

Extended support for rendering embedded image stickers within text messages when the `Markdown` parsing mode is activated, combining a rich, user-friendly experience with the ability to display formatted Markdown text.

## Improvements

Enhanced group chat functionality on the Desktop, enabling mentions ( `@` tag) to be inserted at any position within a composed message, rather than only at the end. Additionally, deleting `@` tags has been optimized.

Maintained message sending permissions for the group owner and administrators during "mute all" scenarios.

Enabled the use of a return `null` value for the `customHoverBar` to utilize the default.

Refined the revoke button functionality for group administrators.

Removed full-screen support for video previews on the Web and introduced an alternative "Open in New Window" button for an enlarged view.

Implemented UIKit log recording to facilitate issue identification and troubleshooting.

Introduced a delete button for the small PNG sticker selection panel on mobile devices, which previously was only available in the Unicode emoji selection panel.

## Bug Fixes

Resolved an issue preventing photo capturing on devices running Android 12 or lower.

Rectified display inaccuracies related to picture aspect ratios.

Addressed several issues concerning voice and video calls.

## Chat Flutter UIKit - 2.1.3+1 @2023.07.19

### New Features

Introduced [a new custom internationalization language scheme](#) that supports adding language packs, adding or modifying entries, and makes customizing i18n more accessible. This feature helps your app achieve a more convenient globalization process and easier customer acquisition worldwide.

Provided a seamless experience for previewing large images and playing videos within desktop environments (applications and web) by avoiding frequent page transitions. Enhanced the user experience for image previews and video playback. Please note that video playback is currently supported only on the web and not in desktop applications.

Supported to integrate with the new online customer service plugin (`tencent_cloud_chat_customer_service_plugin`).

Added two new life cycle hooks, `messageDidSend` and `messageShouldMount` to `ChatLifecycle`.

### Improvements

Optimized the usage, interface, and interaction of the sticker panel.

Enhanced mobile video playback interaction and UI.

Refined the error prompt when sending a 0 KB file fails.

Enabled users to close modals on desktop by clicking the bottom gray overlay area.

Improved the UI and interaction of image and video messages in the message list.

Added the ability to open self-sent file messages without downloading.

Optimized the download status animation of file messages on the web.

## Bug Fixes

Fixed an issue preventing mobile image previews from being dragged after zooming.

Resolved an issue that might cause the message selection status not to be removed after canceling a message forward action.

Addressed an issue that might cause the microphone usage not to end after sending a voice message, which means the microphone was not released.

## Chat Flutter UIKit - 2.1.2 @2023.06.20

### New Features

Introduced a new message recall mode, which enables group administrators to recall any message from any group member. To enable this feature, set `isGroupAdminRecallEnabled` in `TIMUIKitChatConfig` to `true`.



Added support for draft text functionality on the Web. Activate this feature by setting `isUseDraftOnWeb` in `TIMUIKitChatConfig` to `true`. Since the Chat SDK doesn't support this functionality, draft data will be stored in TUIKit's memory. Be aware that draft text will be lost upon refreshing the website.

Enabled using the default message abstract text when `abstractMessageBuilder` returns `null`.

### Improvements

The duration for video messages sent from the Web will no longer be displayed, as this type of video message does not contain an accurate video duration.

Removed the hover color on the message input area on Desktop.

Added auto-focus support for the message input area on Desktop.

Enhanced the rendering of text messages in markdown mode, particularly for clickable link extraction and HTML tag handling.

Limited the number of lines displayed for replied messages to a maximum of 2 lines to avoid occupying excessive space.

Optimized the message replying process, ensuring that a message referencing another message can still display the replied message, even when it is too old.

### Bug Fixes

Fixed an issue that could cause the profile page to display no data.

Fixed an issue that could prevent the message sending button from being displayed after selecting an emoji on mobile Web.

Fixed an issue that could prevent the message long-press menu from showing on mobile Web.

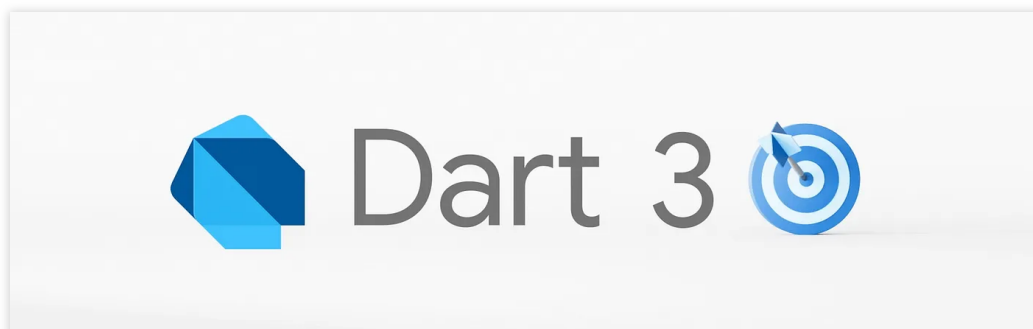
Fixed an issue where editing a message would carry over to another conversation when switching between conversations.

Fixed an issue that could prevent displaying the `Modal` on Desktop.

Fixed an issue that caused the `iconImageAsset` from the `MessageToolTipItem` class to not work properly.

## Chat Flutter UIKit - 2.1.0 @2023.05.30

### Breaking Changes



Migrated to **Flutter 3.10.0 and Dart 3.0.0**, no longer supporting projects with Flutter < 3.10.0 and Dart < 3.0.0.

Updated the minimum requirement for **Android AGP to 7.0**, projects with AGP < 7.0 are no longer supported. We highly recommend updating to these new versions for a better experience.

## New Features

Added several methods to `TIMUIKitChatController`, including `hideAllBottomPanelOnMobile`, `mentionOtherMemberInGroup`, `setInputTextField`, and `getGroupMemberList`. See the corresponding annotations for usage.

Added more parameter fields to the `TIMUIKitChatController`'s `sendMessage` method. For details, see the corresponding annotations.

Added `onSecondaryTapAvatar` to `TIMUIKitChat`, serving as callback trigger for secondary avatar clicks in the message list.

Introduced `isUseMessageHoverBarOnDesktop` and `desktopMessageInputFieldLines` to `TIMUIKitChatConfig`. For usage details, see the corresponding annotations.

## Improvements

Enhanced performance and user experience when switching conversations on Desktop, including features like text field auto-focus and draft text.

Enabled displaying correct new lines in markdown mode.

Changed the order of members in the mentioned member selection panel: Group Owner => Group Administrator => Member, sorted based on the code units' first differing position in the member show names.

Implemented auto-focus after clicking a member in the mentioned member selection panel.

Added text field auto-focus when replying to a message.

Updated other members' display names in at-tag messages to use `namecard`, followed by `nickname` and `userId`.

Widened Desktop message input area's control bar.

Replaced the default icon in Desktop's message input area from `png` to `svg` for better performance and clarity. `DesktopControlBarConfig` now supports defining `svgPath` for each item as well.

Improved Web platform detection.

Mentioning "all" or "at all" can now only be used by group owners and administrators.

Supported returning null for each message item builder in `MessageItemBuilder` to use the default message widget.

Enhanced group members filtering in the group member mentioned selection panel with case-insensitive fuzzy matching, leading to increased filtering accuracy.

For security purposes, downloading files by `fetch` and `blob` in the Web now replaces previewing files in a new browser tab, whereas previewing images and videos is displayed in a new tab on the Web.

Changed the default order in the message tooltip menu.

Previewing images and videos is set to open in a new tab on the Web.

Improved the ratio for sending video messages.

## Bug Fixes

Fixed issues when enabling the section function in markdown mode with `inEnableTextSelection` set to `true`.

Addressed an issue where the replied message was removed when selecting all text in the message and clicking backspace.

Fixed an issue where Chinese characters could not be entered while replying to a message.

Resolved some console errors during debugging.

Fixed an issue with links not opening in markdown mode.

Fixed an issue that caused two `Scrollbar`s to appear in the message input field on Desktop.

Solved an issue that might cause incorrect layout when the app is launched.

Addressed an issue where messages were directly sent when the Enter key was pressed while entering Chinese text.

Fixed related issues with the mentioned member selection panel on Desktop.

Resolved an issue where images couldn't be pasted directly into the message input area for sending on the Web.

Fixed an issue where files couldn't be sent on the Web.

Remedied an issue where media and files couldn't be opened when local downloaded resources were deleted; now, resources will automatically re-download.

Fixed an issue that caused the `iconImageAsset` of the `MessageToolTipItem` config to head internally to this chat UIKit.

Improved the downloading process of media and files by avoiding frequent calls to `setState`, thus preventing the entire project from re-rendering.

## Chat Flutter UIKit - 2.0.0 @2023.05.06

***The version 2.0.0 has undergone significant changes and has gone through 7 preview versions. If you want to know the specific updates for each preview version, you can go to [pub.dev](#) to check.***

***Here are some key points:***

### New Features:

***[Core capability]: TUIKit have been extended to support [all platforms](#), including iOS, Android, Web, Windows, and MacOS, with significant changes to the codebase. The user interface has been enhanced to adapt to devices with different screen types, and additional special capabilities are available for each type of device. For example, desktop devices now support file drag-and-drop for sending files.***

Information copying: Added the ability to copy information from the screen, such as groupID.

Callkit integration: The Callkit button no longer needs to be added to `MorePanelConfig`. If

`tencent_call_uikit` is installed, the video call and voice call buttons will be displayed automatically.

New chat configuration: `TIMUIKitChatConfig` now includes `offlinePushInfo`, which can customize the entire `offlinePushInfo` for each message. The priority of this field is higher than the previous separate configuration fields for this object.

New color configuration: Added `appbarTextColor` and `appbarBgColor` to configure the color of the AppBar. Also added `selectPanelBgColor` and `selectPanelTextIconColor` to configure the color of the message multi-selection panel.

New chat configuration: `isAllowLongPressAvatarToAt`. This option controls whether users are allowed to mention another user in the group by long-pressing on their avatar.

Added `additionalMessageToolTips` to `ToolTipsConfig`. This new property allows developers to add additional message operation tool tips beyond the default prompt items. The previous `additionalItemBuilder` has been replaced by this new property. With `additionalMessageToolTips`, developers only need to specify the data for the tool tip items, rather than providing the entire widget. This makes it easier to use, as you no longer need to worry about UI display.

Added `isPreloadMessagesAfterInit` to `TIMUIKitConfig`, which can determine whether TUIKit should preload some messages after initialization to speed up message display.

Introduced `isAutoReportRead` to `TIMUIKitChatConfig` to control read status reporting.

### Improvements:

Improved group management logic, non-administrators can no longer access the management interface.

Improved cursor positioning when sending messages.

Improved and optimized scrollbar functionality.

Enhanced clickable URL support in messages, URLs now support both with and without "https://" prefixes.

Improved compatibility: TUIKit is now compatible with Flutter versions 3.0.0 to 3.7.7.

Improved group management: Members in `workgroup` can no longer be muted.

Improved avatar: Ensures that the avatar can be as small as possible while covering the entire target box.

Eliminated the dependency on `fluttertoast`. All necessary customer reminders are now triggered through the `onTUIKitCallbackListener` info callback in your project. For more information, [please see this document](#).

Removed six other unnecessary dependency packages to reduce size and improve performance.

Improved the clarity of the `sendMessage` function in `TIMUIKitChatController` by replacing the use of `convID` to represent both `userID` and `groupID` with separate parameters.

The time separator on the message list: The default 12-hour display has been changed to 24-hour display.

Message translation now targets the language of TUIKit, rather than directly depending on the system language. The language of TUIKit can be automatically set to the system language or defined by the user. For more information, [please see this document](#).

Optimized the animation of the message text input area.

Message operation menu display: If there are no operation items and the message sticker reaction module is not used, the long-press message will not display.

Upgraded several dependencies to the latest version, including `ffi` upgraded to 2.0.1 and `file_picker` upgraded to 5.2.9.

Added support for the new permission authorization schema on Android 13 and `targetSdkVersion` greater than 33.

Corrected `extHight` to `extHeight` in `TIMUIKitChatConfig` and changed the default value to 1.3.

When `isAtWhenReply` is set to `true`, the reply or quote button is marked as Reply, otherwise it is marked as Quote.

The @member tag can now be deleted at once.

### Chat Flutter UIKit - 1.7.0 @2023.02.23

Addition: Support for quickly navigating to the first unread message in a group chat with more than 20 new unread messages, using the dynamic tongue located in the top right corner of the screen. This feature allows for swift movement through the messages, regardless of their quantity.

Addition: Customize the border radius for all avatars is now supported. You can set the default avatar border radius using `defaultAvatarBorderRadius` in `TIMUIKitConfig`.

Optimization: The delete button on the sticker sending panel has been improved for better usability.

Optimization: Some English labels on the screen have been updated to better reflect local expressions.

Fix: An issue causing errors when sending a large number of stickers has been resolved.

Fix: Some errors that were occurring in the sticker panel have been addressed.

Fix: An issue that caused errors on mentioning all members.

### Chat Flutter UIKit - 1.6.0 @2023.02.08

Added `scrollToConversation` to `TIMUIKitConversationController`. You can now easily navigate to a specific conversation in the conversation list and move to the next unread conversation by double-clicking the tab bar. For more information, see our [demo source code](#).

Optimized the performance of the historical message list while scrolling over a large distance.

### Chat Flutter UIKit - 1.5.0 @2023.02.02

Added `defaultAvatarAssetPath` to the global configuration `TIMUIKitConfig` to define the default profile photo.

Added support for Flutter 3.7.0.

Fixed the `chatBgColor` configuration.

### Chat Flutter UIKit - 1.4.0 @2023.01.13

Added the feature of translating the text in text messages and replied and quoted messages. To use the feature, you only need to long press the text and choose **Translate**. This feature can be enabled or disabled by the `showTranslation` parameter in `ToolTipsConfig`.

Optimized the position of the window that pops up when text is long pressed.

Optimized the keyboard pop-up event.

## Chat Flutter SDK (Low-level SDK) - 5.0.8 @2023.01.13

Added the group counter capability: Ordinary groups and audio-video groups support meta counters. For details, see groupCounter related APIs.

## Chat Flutter UIKit - 1.3.0 @2023.01.11

Fixed the failure to display the nickname of the new group owner in the group tip message after the group ownership is transferred.

Removed the confirmation window that pops up before a file is opened.

## Chat Flutter UIKit - 1.2.0 @2023.01.06

Fixed the failure to display the input box when the chat component is switched from recording to keyboard.

Fixed the issue where, when a combined message is sent to multiple recipients, only the first recipient can receive the message.

Optimized `MessageItemBuilder` so that it can be used to display the combined message page.

## Chat Flutter UIKit - 1.1.0 @2022.12.27

Embedded the emoji plug-in in TUIKit by default. Now we support three types of emojis: Unicode emoji, small image emoji, and big image emoji. The emoji use methods have been optimized. For more information, see [here](#).

Optimized the topic feature to support more custom capabilities.

Optimized the animation for the input area, keyboard, sticker panel, and "More" panel.

Optimized: Emojis, including Unicode and small images, can be inserted to any position in text messages.

Optimized: The profile photo in a profile can be previewed with a large image.

Optimized: The user ID in a user profile can be copied.

Optimized: Several UI details are added, including `TIMUIKitAddFriend` , `TIMUIKitAddGroup` , `TIMUIKitGroupProfile` , and `TIMUIKitProfile` .

Optimized: `TIMUIKitGroupProfile` and `TIMUIKitProfile` support modifying content by modifying IDs.

Optimized `TIMUIKitGroupChat` to display the loading animation when a user clicks the image/video download button.

Fixed some errors.

## Chat Flutter SDK (Low-level SDK) - 5.0.6 @2022.11.29

Fixed the iOS bundle version loss issue.

Improvement: The underlying native SDK has been upgraded to 6.9.3557.

## Chat Flutter UIKit - 1.0.1 @2022.11.28

Removed `groupTRTCTipsItemBuilder` from `MessageItemBuilder` . Please use `customMessageItemBuilder` instead.

## Chat Flutter UIKit - 1.0.0 @2022.11.23

Supported adding the Flutter module to your existing applications, that is, hybrid development. For more information, see [here](#).

Supported customizing stickers and emojis. **The use methods have changed greatly.**

Supported adding the Flutter module to your existing applications, that is, hybrid development. For more information, see [here](#).

Supported customizing stickers and emojis. **The use methods have changed greatly.**

Optimized the loading time of the historical message list, especially when there are a large number of media and file messages.

Supported scrolling in more panel areas.

Optimized the feature of loading the latest messages when scrolling back to the bottom for smoother loading.

Fixed the Android album image quantity issue.

Fixed the border crossing issue of long text in the group profile card.

Fixed some errors.

### Note

If you upgrade your TUIKit to this version, pay special attention to the changes of the emoji part (the second update item) and audio/video call part (the last but one update item). Otherwise, related capabilities will not work properly.

If you have any questions in the process of modification, feel free to contact us.

## Chat Flutter SDK (Low-level SDK) - 5.0.4 @2022.11.23

Multimedia message online URLs will no longer be returned by default. They need to be obtained by calling the `getMessageOnlineUrl` API.

Multimedia message local URLs ( `localurl` ) will no longer be returned by default. They will be returned only after you download messages successfully by calling the `downloadMessage` API.

The `onMessageDownloadProgressCallback` callback is added for `advanceMessageListener` and will be triggered when the multimedia message download progress is updated.

Added the `disableBadgeNumber` method to iOS clients. After the method is called, the application badge is not set by default when the application is switched to the background.

Supported adding the Flutter module to your existing applications, that is, hybrid development. For more information, see [here](#).

Optimized the underlying dynamic library download logic for PC clients.

Upgraded the underlying SDK version to 6.8.

Reconstructed the web client underlying SDK. You need to import the corresponding JS via npm as instructed [here](#).

Reconstructed the macOS client underlying SDK. You need to import the SDK as instructed [here](#).

### Note

Major changes have been made to multimedia messages and file messages. Please modify your existing logic for obtaining and rendering such messages according to the first four updates. Otherwise, the messages will not be

displayed.

If you have any questions in the process of modification, feel free to contact us.

### Chat Flutter UIKit - 0.1.8 @2022.10.21

Optimized the file batch download queue to allow a user to select multiple file messages at a time.

Optimized the group list widget to support automatic update.

Optimized camera shooting to support low-performance devices and automatically adjust the resolution for the devices.

Optimized the support for customizing the color and text styles of the app bar, especially on the `TIMUIKitChat` component.

Fixed the issue where friend notes or nicknames could not be displayed in group tips.

Fixed video playback errors.

Fixed several issues.

### Chat Flutter SDK (Low-level SDK) - 4.1.8 @2022.10.18

Added support for PC platforms, including macOS and Windows.

Added message extensions.

Added signaling editing.

Upgraded the underlying SDK.

Fixed the later-version JDK conversion issue.

Fixed several issues.

### Chat Flutter UIKit - 0.1.7 @2022.10.18

Added support for large and RAW images, especially those captured from the latest versions of iOS and the iPhone 14 Pro series, compressed and formatted before automatic sending.

Optimized performance and stability, especially the historical message list and startup.

Optimized the initialization of `TIMUIKitChat` as an idempotent operation.

Supported loading the latest messages when scrolling back to the bottom.

Optimized support for Flutter 2.x and 3.x series.

Permission support for iOS album: allowing only certain images.

Fixed several issues.

### Chat Flutter UIKit - 0.1.5 @2022.09.22

Added web support. Now you can implement TUIKit on iOS/Android/Web platforms.

Added the feature of checking disk storage after login, which can be configured in `config` in `init`.

Added the following attributes to `TIMUIKitChatConfig`: `timeDividerConfig`, `notificationAndroidSound` (Huawei and Google push sound configuration), `isSupportMarkdown` (whether text messages support Markdown parsing), and `onTapLink`.



Removed the default emoji list due to copyright issues. You can provide TUIKit with your own emoji list via [tim\\_ui\\_kit\\_sticker\\_plugin](#).

Optimized: Now you can disable the display of @ messages in the conversation list

Optimized: Now you can set the return values of `notificationExt` and `notificationBody` in `TIMUIKitChatConfig` and `MessageItemBuilder` as `null`, and, in specific cases, you can use the default values as needed. This means you can control whether to use custom settings without redefining the same logic as TUIKit in code.

Optimized: Supported multi-line text messages.

Optimized the experience of `TIMUIKitChat`. In addition, to use `TIMUIKitChatController`, you need to pass in `controller`, as described [here](#).

### Chat Flutter SDK (Low-level SDK) - 4.1.3 @2022.09.21

Fixed some web issues.

### Chat Flutter SDK (Low-level SDK) - 4.1.1+2 @2022.08.25

Upgraded the underlying library version to 6.6.x.

Full support for Chat Flutter SDK for web.

### Chat Flutter SDK (Low-level SDK) - 4.1.0 @2022.08.09

Upgraded the underlying library version.

### Chat Flutter UIKit - 0.1.3 @2022.08.03

Added user input status.

Added the capability to respond to message emojis.

Added the display of user online status.

### Chat Flutter SDK (Low-level SDK) - 4.0.8 @2022.07.25

Added an advanced API for getting the conversation list, which supports pulling the conversation list by conversation type/tag.

Added an API for customizing conversation marks.

Added the conversation grouping capability.

Decreased the dependent Dart version to 2.0.0.

Supported multi-engine Flutter.

Supported offline push sound effect configuration on Android.

Supported custom user online status.

Upgraded the underlying library version to 6.5.x.

### Chat Flutter UIKit - 0.1.2 @2022.07.08

Fixed the issue where the original referenced third-party underlying recording library

`flutter_record_plugin_plus` could not be used.

### Chat Flutter UIKit - 0.1.1 @2022.07.07

Optimized the image preview logic.

Added lifecycle hooks for each component.

Added the mute status to group chat page.

Supported redirection upon clicking URLs in text messages and added the website information preview card.

Added TUIKit layer global event callbacks, including callbacks for messages that need to be prompted, Flutter layer errors, and Chat API layer errors. TUIKit no longer provides message pop-ups. You can customize pop-up windows based on callbacks and prompts.

Reconstructed the group profile component `TUIKitGroupProfile` and user profile component

`TUIKitProfile`, simplifying usage and enabling super-fast access.

### Chat Flutter SDK (Low-level SDK) - 4.0.7 @2022.07.07

Supported custom badge numbers in iOS.

Optimized the group joining request logic.

### Chat Flutter SDK (Low-level SDK) - 4.0.6 @2022.07.04

Upgraded the underlying library version to 6.2.x.

Fixed offline push information fields.

### Chat Flutter SDK (Low-level SDK) - 4.0.5 @2022.07.01

Added user online status query.

Supported requesting historical messages by message type.

Supported sending rich text messages.

### Chat Flutter UIKit - 0.1.0 @2022.06.10

Added the atomic development capability of the `TIMUIKitChat` component, and you can assemble the chat page by yourself through various sub-components.

Supported the capability to edit messages and update UIs.

Added group joining request approval page components.

Added Traditional Chinese as a language option.

Opened up more custom component parameters.

### Chat Flutter UIKit - 0.0.9 @2022.05.30

Supported offline push, with the newly released [tim\\_ui\\_kit\\_push\\_plugin](#) push plugin.

Supported Flutter 3.0.

Optimized the local preview of media messages.

### Chat Flutter SDK (Low-level SDK) - 4.0.2 @2022.05.27

Fixed the local video path issue.

### Chat Flutter SDK (Low-level SDK) - 4.0.1 @2022.05.23

Added the topic capability.

Added the message editing capability.

### Chat Flutter SDK (Low-level SDK) - 4.0.0 @2022.04.26

Upgraded the underlying library version to 6.2.x.

Fixed offline push information fields.

### Chat Flutter UIKit - 0.0.8 @2022.04.24

Added the group message read receipt capability.

Added a small toolbar in the lower right corner of the chat area to support returning to the bottom/displaying the number of new messages/@message reminder.

### Chat Flutter SDK (Without UI Library) 3.9.3 @2022.04.20

Fixed the issue where the `boolValue` of a group muting tip was lost.

Added the `key(string)-boolValue(bool)` format in addition to the existing `key(string)-value(string)` in the callback for group information modification.

Fixed the issue where the `nameCard` field of a conversation was not parsed by the instance.

Added APIs for group message read receipts.

[sendMessageReadReceipts](#): Sends group message read receipts.

[getMessageReadReceipts](#): Gets read receipts for messages sent by yourself.

[getgroupMessageReadMemberList](#): Gets the list of members who have (or have not) read a message sent by yourself.

Improved the Flutter for web.

### Chat Flutter UIKit - 0.0.7 @2022.04.13

Optimized experience.

### Chat Flutter UIKit - 0.0.6 @2022.04.08

Opened up the API for automatically displaying sent messages on the screen and provided more custom capability parameters.

Optimized user login authentication.

Optimized privacy policies to better align with personal information protection laws.

### Chat Flutter UIKit - 0.0.5 @2022.03.24

Opened up more custom capabilities of the chat area component `TIMUIKitChat` .

### Chat Flutter SDK (Low-level SDK) - 3.9.1 @2022.03.24

Upgraded the underlying library to v6.1.2155.

### Chat Flutter SDK (Low-level SDK) - 3.9.0 @2022.03.22

Modified GroupListener.

### Chat Flutter SDK (Low-level SDK) - 3.8.9 @2022.03.18

Fixed the registration result listening issue.

### Chat Flutter UIKit - 0.0.4 @2022.03.17

Added support for sending images and videos.

Optimized topic styles.

Optimized the search component.

### Chat Flutter UIKit - 0.0.3 @2022.03.14

Optimized component details.

Improved the automatic internationalization capability.

Added the global search component `TIMUIKitSearch` .

Added the in-conversation search component `TIMUIKitSearchMsgDetail` .

Added the friend adding component `TIMUIKitAddFriend` .

Added the group joining component `TIMUIKitAddGroup` .

Added topic styles.

### Chat Flutter SDK (Low-level SDK) - 3.8.4 @2022.03.14

Updated APIs.

### Chat Flutter UIKit - 0.0.2 @2022.03.02

Optimized the `TIMUIKitChat` component.

Supported automatic and manual switching between Simplified Chinese and English.

### Chat Flutter UIKit - 0.0.1 @2022.03.01

Launched Tencent Cloud Chat for Flutter (including the UI library and business logic).

Released the first seven main components, including the chat area, conversation list, contact and group profiles, contacts list, blocklist, and friend request list.

**Chat Flutter SDK (Low-level SDK) - 3.8.3 @2022.03.01**

Switched the token encoding format based on the environment.

**Chat Flutter SDK (Low-level SDK) - 3.8.2 @2022.02.21**

Updated group member parameter constraints.

**Chat Flutter SDK (Low-level SDK) - 3.8.0 @2022.02.17**

Upgraded the underlying API dependencies.

**Chat Flutter SDK (Low-level SDK) - 3.7.8 @2022.02.15**

Fixed the exception caused by force unwrapping.

**Chat Flutter SDK (Low-level SDK) - 3.7.7 @2022.02.10**

Fixed the Swift code warning.

Rewrote Swift's force unwrapping code.

Added the `id` field to the `message` instance returned by the `sendMessage` API.

**Chat Flutter SDK (Low-level SDK) - 3.7.5 @2022.01.23**

Upgraded the underlying library to v6.0.1975.

Supported the TPNS token for offline push configuration.

**Chat Flutter SDK (Low-level SDK) - 3.7.1 @2022.01.12**

Added the feature of returning the message creation ID for a message sending progress event.

Optimized the callback by reminding the business side that the callback error is caught in SDK and needs to be modified.

**Chat Flutter SDK (Low-level SDK) - 3.7.0 @2022.01.10**

Optimized the unpacking of `cloudCustomData`.

**Chat Flutter SDK (Low-level SDK) - 3.6.9 @2022.01.06**

Optimized the message reply parameters.

**Chat Flutter SDK (Low-level SDK) - 3.6.8 @2022.01.06**

Optimized the message reply API.

**Chat Flutter SDK (Low-level SDK) - 3.6.7 @2022.01.05**

Upgraded the compiling environment for iOS from 8.0 to 9.0.

## Chat Flutter SDK (Low-level SDK) - 3.6.6 @2021.12.30

Added the message reply API.

Fixed the issue for web where the release mode triggered an error.

## Chat Flutter SDK (Low-level SDK) - 3.6.5 @2021.12.17

Fixed syntax errors in Java.

## Chat Flutter SDK (Low-level SDK) - 3.6.4 @2021.12.17

Fixed the issue where there was no return for Android async registration events.

Fixed the issue where removing a general listening event triggered an error.

Added the UUID of a message being sent in its progress event.

## Chat Flutter SDK (Low-level SDK) - 3.6.3 @2021.12.9

Optimized the `addFriend` API: Changed `addType` from int to `FriendTypeEnum`.

Optimized the `acceptFriendApplication` API: Changed `acceptType` from int to `FriendResponseTypeEnum`.

Optimized the `checkFriend` API: Changed `checkType` from int to `FriendTypeEnum`.

Optimized the `createGroup` API: Changed `addOpt` from int to `GroupAddOptTypeEnum`.

Optimized the `deleteFromFriendList` API: Changed `deleteType` from int to `FriendTypeEnum`.

Optimized the `getGroupMemberList` API: Changed `filter` from int to `GroupMemberFilterTypeEnum`.

Optimized the `getHistoryMessageList` API: Changed `type` from int to `HistoryMsgGetTypeEnum`.

Optimized the `getHistoryMessageListWithoutFormat` API: Changed `type` from int to `HistoryMsgGetTypeEnum`.

Optimized the `getGroupMemberList` API: Changed `type` from int to `GroupMemberFilterTypeEnum`.

Optimized the `getGroupMemberList` API: Changed `filter` from int to `GroupMemberFilterTypeEnum`.

Optimized the `initSDK` API: Changed `loglevel` from int to `LogLevelEnum`.

Optimized the `refuseFriendApplication` API: Changed `acceptType` from int to `FriendApplicationTypeEnum`.

Optimized the `sendCustomMessage` API: Changed `priority` from int to `MessagePriorityEnum`.

Optimized the `sendFaceMessage` API: Changed `priority` from int to `MessagePriorityEnum`.

Optimized the `sendFileMessage` API: Changed `priority` from int to `MessagePriorityEnum`.

Optimized the `sendForwardMessage` API: Changed `priority` from int to `MessagePriorityEnum`.

Optimized the `sendImageMessage` API: Changed `priority` from int to `MessagePriorityEnum`.

Optimized the `sendLocationMessage` API: Changed `priority` from int to `MessagePriorityEnum`.

Optimized the `sendMergerMessage` API: Changed `priority` from int to `MessagePriorityEnum`.

Optimized the `sendSoundMessage` API: Changed `priority` from int to `MessagePriorityEnum`.

Optimized the `sendTextAtMessage` API: Changed `priority` from int to `MessagePriorityEnum`.

Optimized the `sendMessage` API: Changed `priority` from `int` to `MessagePriorityEnum`.

Optimized the `setGroupMemberRole` API: Changed `role` from `int` to `GroupMemberRoleTypeEnum`.

Changed the event callback mode to asynchronous.

### Chat Flutter SDK (Low-level SDK) - 3.6.2 @2021.12.9

Fixed the issue where no uuid was passed in for removing an advanced message.

### Chat Flutter SDK (Low-level SDK) - 3.6.1 @2021.12.8

Fixed the loss of the file progress event.

### Chat Flutter SDK (Low-level SDK) - 3.6.0 @2021.12.1

Added the support for multiple listener registrations and callbacks in modules.

Added the `markAllMessageAsRead` API for marking all messages as read.

Added the feature of parsing combined messages.

Upgraded the Native SDK to v5.8.1668.

### Chat Flutter SDK (Low-level SDK) - 3.5.6 @2021.11.25

Fixed the `checkFriend` failure.

Fixed the issue where `getC2CHistoryMessageList` API failed to get subsequent messages.

### Chat Flutter SDK (Low-level SDK) - 3.5.5 @2021.11.23

Adjusted the architecture.

### Chat Flutter SDK (Low-level SDK) - 3.5.4 @2021.11.22

Added the `downloadMergeMessage` API.

### Chat Flutter SDK (Low-level SDK) - 3.5.3 @2021.11.15

Added the `onTotalUnreadMessageCountChanged` event.

Added the `orderkey` field in the `V2TimConversation` API for conversation sorting.

### Chat Flutter SDK (Low-level SDK) - 3.5.2 @2021.11.12

Added support for web.

### Chat Flutter SDK (Low-level SDK) - 3.5.1 @2021.11.10

Added the logic to be compatible with array index out of bounds.

### Chat Flutter SDK (Low-level SDK) - 3.5.0 @2021.10.1

Fixed several known issues.

Added the following APIs:

callExperimentalAPI  
clearC2CHistoryMessage  
clearGroupHistoryMessage  
searchLocalMessages  
findMessages  
searchGroups  
searchGroupMembers  
getSignalingInfo  
addInvitedSignaling  
searchFriends

### Chat Flutter SDK (Low-level SDK) - 1.0.34 @2021.03.22

Fixed the issue for iOS where getting the message history triggered an error.

### Chat Flutter SDK (Low-level SDK) - 1.0.33 @2021.03.22

Changed the `minSdkVersion` value of the SDK to 16.

### Chat Flutter SDK (Low-level SDK) - 1.0.32 @2021.03.22

Fixed the crash that occurred when `lastMessage` in the conversation information was empty.

### Chat Flutter SDK (Low-level SDK) - 1.0.30-1.0.31 @2021.03.18

Fixed the crash that occurred when the `data` field of a custom message was null.

### Chat Flutter SDK (Low-level SDK) - 1.0.29 @2021.03.16

[Important] Fixed the issue where passing in parameters for getting the group member list triggered an error.

### Chat Flutter SDK (Low-level SDK) - 1.0.28 @2021.03.16

[Important] Changed the input parameters of the `checkFriends` API.

### Chat Flutter SDK (Low-level SDK) - 1.0.15-1.0.27 @2021.03.15

Added the group member custom field.

Improved iOS signaling.

Fixed the iOS signaling bug.

Added the feature of parsing a custom field into `string` before returning it.

Optimized the settings of custom fields of the profile.

Updated the `getHistoryMessageList` API for Android.

Fixed the issue for Android where passing in parameters for the `checkFriend` API triggered an error.



### Chat Flutter SDK (Low-level SDK) - 1.0.5-1.0.14 @2021.02.26

Fixed the issue where passing in parameters for the `deleteFriendApplication` API triggered an error.

Updated the Native SDK to v5.1.132.

Updated the Native SDK to v5.1.137.

Fixed the bug that occurred when passing in parameters for the signaling invitation API.

Fixed the issue where the signaling API did not return an ID.

Modified the SDK compression configuration.

Fixed signaling callback bugs.

Modified the return data of custom messages.

[Important] Modified the format of content returned for a signaling message. Please upgrade to this version or later to use signaling.

### Chat Flutter SDK (Low-level SDK) - 1.0.4 @2021.01.14

Upgraded the SDK for Android to v5.1.129.

Upgraded the SDK for iOS to v5.1.129.

### Chat Flutter SDK (Low-level SDK) - 1.0.3 @2021.01.13

Added support for Android and iOS platforms.

Added support for one-to-one chat and group chat (discussion and audio-video groups).

Added support for text, emoji, image, audio, and custom messages.

Added support for offline push of APNs (reporting of token and foreground/background switch).

Added the feature of storing messages locally.

### Chat Flutter SDK (Low-level SDK) - 0.0.1-1.0.2 @2020.12.01

Launched the Flutter SDK.

Invited users to join the beta test.

# Electron

Last updated : 2024-06-17 16:52:38

## IM Electron SDK 8.0.5895 @2024.6.12

Update the underlying dependency library to 8.0.5995.

## IM Electron SDK 7.9.5580 @2024.4.20

Update the underlying dependency library to 7.9.5580.

Modify the version number to be consistent with the underlying dependency library.

## IM Electron SDK 3.0.21 @2024.4.10

Fix the search group chat interface searchGroups.

Fix the invite group member interface inviteInGroup.

Fix the bug that the removed message listener cannot be removed.

## IM Electron SDK 3.0.20 @2024.3.28

Update the underlying version to 7.8.

Fixed the group messaging bug (batchSend).

Fixed searching group member bug (searchGroupMembers).

## IM Electron SDK 3.0.15 @ 2024.1.30

Fixed conversation clearing message bug.

## IM Electron SDK 3.0.14 @ 2024.1.19

Added cloud search capability.

## IM Electron SDK 3.0.13 @ 2024.1.18

Optimize privatized networking.

## IM Electron SDK 3.0.0 @ January 8, 2024

Enhance memory and CPU usage of the Electron SDK process.

Add text message translation interface.

Add voice message to text conversion interface.

## IM Electron SDK 2.0.21 @ November 10, 2023

Adjust input and output parameters of the signaling interface.

Add interfaces and callbacks for retrieving and modifying signals.

Fix an issue where local message search did not return messages.

## IM Electron SDK 2.0.20 @ November 1, 2023

Upgrade the underlying Linux IM SDK C version to 7.5.

## IM Electron SDK 2.0.19 @2023.10.16

Upgrade the underlying IM SDK version C to 7.5.

Modify the spelling of some fields, including (before modification --> after modification):

msg\_getmsglist\_param\_is\_rembles --> msg\_getmsglist\_param\_is\_ramble.

group\_search\_member\_result\_member\_info\_list --> group\_search\_member\_result\_member\_info\_list.

msg\_delete\_param\_is\_rembles --> msg\_delete\_param\_is\_ramble.

group\_attribute\_key --> group\_attribute\_key.

group\_attribute\_value --> group\_attribute\_value.

friend\_response\_identifier --> friend\_response\_identifier.

friend\_response\_action --> friend\_response\_action.

friend\_response\_remark --> friend\_response\_remark.

friend\_response\_group\_name --> friend\_response\_group\_name.

## IM Electron SDK 2.0.18 @2023.9.11

Modify some functions to pass message body, including: `MsgRevoke` , `MsgDelete` , `MsgListDelete` , `MsgReportedReaded`.

## IM Electron SDK 2.0.17 @2023.8.20

Upgrade the underlying IM SDK C version to 7.4.

Support Community Group Topic.

Support group member `online status` .

## IM Electron SDK 2.0.16 @2023.6.20

Support conversation group related callback.

## IM Electron SDK 2.0.15 @2023.5.12

Added an interface for obtaining user online status.

## IM Electron SDK 2.0.12 @2023.4.29

Modify the download interface.

Add user profile update callback and user status update callback.

## IM Electron SDK 2.0.9 @2023.3.31

Upgrade the underlying IM SDK C version to 7.1.

Support multiple registrations for `recvNewMessageCallback`.

Groups support pulling historical messages by sequence list.

Delete session support setting whether to clear history messages.

Provides an interface to delete sessions in batches.

Support modifying the approval method for inviting people into groups.

Whether the message object is added is a parameter for content auditing.

Support to get the total number of unread according to the filter of the session.

Ordinary group, live broadcast group and community support group counter meta counter.

The single chat read report supports reporting according to the message timestamp.

The group chat read report supports reporting according to the message sequence.

One-on-one chat sessions expose timestamps of read messages.

Group chat sessions expose the sequence of read messages.

## IM Electron SDK 2.0.7 @2023.02.07

Modify linux arm adaptation.

## IM Electron SDK 2.0.6 @2023.01.12

Add interface supported by underlying IM SDK C version 6.5 and above.

Add MacOS offline push.

## IM Electron SDK 2.0.4 @2022.12.12

Add MessageExtension parameter.

Modify the return value of GroupReadMemberList.

Add Message field.

## IM Electron SDK 2.0.2 @2022.11.1

Updated the underlying IM SDK C version to 6.7.

Supported message extension.

Optimized input and output parameters.

## IM Electron SDK 1.3.4 @2021.10.19

Officially available to users.

# Unity

Last updated : 2024-05-09 14:57:13

## 1.9.2 @2023.11.03

Added TIMSetMsgReactionsChangedCallback callback.  
Added TIMSetMsgAllMessageReceiveOptionCallback callback.  
Added TIMMsgSetAllReceiveMessageOpt interface.  
Added TIMMsgSetAllReceiveMessageOpt2 interface.  
Added TIMMsgGetAllReceiveMessageOpt interface.  
Added TIMMsgSearchCloudMessages interface.  
Added TIMMsgAddMessageReaction interface.  
Added TIMMsgRemoveMessageReaction interface.  
Added TIMMsgGetMessageReactions interface.  
Added TIMMsgGetAllUserListOfMessageReaction interface.  
Added TIMMsgConvertVoiceToText interface.

## 1.9.1 @2023.10.29

Fixed the bug where the Windows editor cannot be used.

## 1.9.0 @2023.10.25

Updated the underlying layer to v7.5.4864.  
Added an interface for subscribing to strangers' profiles.  
Added an interface for deleting a conversation list.  
Added an unread message count interface for cleaning a conversation.  
Modified some interface parameter fields (before modification -> after modification):  
msg\_getmsglist\_param\_is\_rembles -> msg\_getmsglist\_param\_is\_ramble  
msg\_delete\_param\_is\_rembles -> msg\_delete\_param\_is\_ramble  
group\_attribute\_key -> group\_attribute\_key  
group\_attribute\_value --> group\_attribute\_value  
friend\_response\_identifier --> friend\_response\_identifier  
friend\_response\_action --> friend\_response\_action  
friend\_response\_remark --> friend\_response\_remark  
friend\_response\_group\_name --> friend\_response\_group\_name  
group\_base\_info\_latest\_seq -> group\_base\_info\_latest\_seq  
friend\_add\_pendency\_info\_idenfifer -> friend\_add\_pendency\_info\_identifier  
user\_config\_is\_ingore\_grouptips\_unread -> user\_config\_is\_ignore\_grouptips\_unread  
message\_offlie\_push\_config -> message\_offline\_push\_config

msg\_search\_param\_send\_indentifier\_array —> msg\_search\_param\_send\_identifier\_array

group\_detial\_... -> group\_detail\_...

Added some interface parameters:

UserProfile

ConvInfo

ConversationListFilter

Message

MsgGetMsgListParam

MessageSearchParam

AndroidOfflinePushConfig

## 1.8.3 @2023.01.13

Added support for adding multiple callback functions to an event callback.

Added support for deleting specified callback functions.

Added conversation group related APIs and event callbacks.

Added support for setting custom conversation data event callbacks.

Added conversation marking APIs.

Added the advanced API for getting the conversation list.

Added group counter related APIs and event callbacks.

Added the text message translation API.

Added the API and event callback for getting/subscribing to the total unread message count by filter.

## 1.8.2 @2022.12.06

Supported Mac M1 chips for build.

Supported WebGL for build.

Fixed the parameter types of callbacks such as `GroupGetTopicInfoList` and `ConvGetConvList` .

Added parameters and callback data logs.

## 1.8.0 @2022.10.11

Fixed the conversion performance issue involved in first parameter serialization.

## 1.7.9 @2022.09.22

Fixed iOS build issues.

## 1.7.7 @2022.09.02

Added English API annotations.

Added topic, community, user status, and other APIs.

Upgraded the native SDK.

Fixed known issues.

## 1.7.6 @2022.06.24

Supported `string callback data` and `object callback data` .

## 1.7.5 @2022.05.23

Added APIs for group message read receipts.

Fixed the issue where the field with a value of `null` was ignored by Newtonsoft serialization.

Fixed the issue where the `uint64` field in `GroupPendencyResult` was changed to `ulong` .

## 1.6.4 @2022.01.13

Added SDK support for package manager import.

Added the feature of adding dependencies after iOS compilation.

## 1.6.0 @2021.12.21

Switched the underlying cross-platform C APIs.

Added support for the Windows, macOS, Android, and iOS platforms with unified APIs.

Note that v1.6.0 is incompatible with earlier versions.

## 1.5.1 @2021.11.24

Fixed the issue where `-1` is returned for `sequenceid` unexpectedly.



Added support for macOS.

Removed the simple message module and enabled the advanced message module for both message receiving and sending.

## 1.5.0 @2021.11.16

Added support for Windows.

## 1.4.0 @2021.08.03

Simplified the configuration process on iOS.

Fixed the IL2CPP packaging error on Android.

## 1.3.1 @2021.05.21

Fixed known issues.

## 1.3.0 @2021.05.10

Added the C# model to instantiate data returned by APIs.

Added the usage of the C# model to `ExampleEntry.cs`.

## 1.2.0 @2021.04.28

Added `sequenceID` to some message sending APIs to associate message requests and responses.

## 1.1.1 @2021.04.25

Separated the method of dynamically fetching userSig.

## 1.1.0 @2021.04.15

Added advanced message APIs.

Added signaling message APIs.

## 1.0.1 @2021.04.01

Initialized the project and implemented most APIs.

# React Native

Last updated : 2023-09-19 15:14:31

## Chat React Native SDK 0.1.9 @2022.12.05

New: Multimedia message online URLs ( `url` ) will no longer be returned by default. They need to be obtained by calling the `getMessageOnlineUrl` API.

New: Multimedia message local URLs ( `localurl` ) will no longer be returned by default. They will be returned only after you download messages successfully by calling the `downloadMessage` API.

New: The `onMessageDownloadProgressCallback` callback is added for `advanceMessageListener` and will be triggered when the multimedia message download progress is updated.

New: The message extension feature is added. For details, see Message > Message Extension.

Improvement: The underlying native SDK has been upgraded to 6.9.3557.

### Note

Major changes have been made to multimedia messages and file messages. Please modify your existing logic for obtaining and rendering such messages according to the first three updates. Otherwise, the messages will not be displayed. If you have any questions in the process of modification, please feel free to [contact us](#).

## Chat React Native SDK 0.1.0 @2022.07.14

Officially open to users

### Note

This is the first release of the Chat SDK for React Native, and it will be improved in future versions.