UCCX et SocialMiner : Analyse du journal de classement des discussions en bulles et post-discussion

Contenu

Introduction
Journaux requis
Flux de discussion bulle
Analyse des journaux
Étape 1. Journaux de la console client : Cliquez pour discuter.
Étape 2. SM ccppublicapps : DB dip vers mmca_webhook
Étape 3. UCCX MADM : Reçoit une demande de configuration de widget.
Étape 4. SM Publicapps:200 OK dans la configuration du widget.
Étape 5. Journaux de la console client : Réponse au navigateur client qui affiche les détails dans JSON.
Étape 6. Applications publiques SM : Demande de conversation entrante.
Étape 7. Exécution de SM : Le contact social est créé.
Étape 8. Applications publiques SM : Met à jour le SocialContact créé vers l'API publique.
Étape 9. Exécution de SM : Envoie la notification à l'API CCX Webservice sur MADM.
Étape 10. UCCX MADM : Webservices affiche la demande de conversation entrante.
Étape 11. UCCX MADM : Envoyez un message JMS au moteur CCX.
Étape 12. MIVR UCCX :
Étape 13. Applications publiques SM : La session Jointures et discussions de l'agent a démarré.
Étape 14. Journaux de la console client : Un agent rejoint la salle de discussion.
Étape 15. SM Publicapps : l'utilisateur quitte la salle de discussion.
Étape 16. Journaux de la console client : Le navigateur utilisateur affiche l'évaluation des conversations réussies.
Étape 17. Applications publiques SM : ccppublicapps reçoit l'évaluation post-discussion du navigateur.
Étape 18. MIVR UCCX : Mise à jour XMPP reçue du SM.
Étape 19. MIVR UCCX : Écrire le classement de conversation dans la base de données.

Introduction

Ce document décrit le flux de discussion en bulle ainsi que l'analyse de journal de bout en bout pour une discussion en bulle de travail, qui peut être utilisée comme référence de travail pour résoudre les problèmes.

La solution Unified Contact Center Express (UCCX) avec la version d'UCCX et SocialMiner (SM) 11.6(2) ont ajouté la nouvelle fonctionnalité de discussion en mode bulle.

La fonctionnalité de Bubble Chat (ou Chat Bubble) vous permet de joindre une entreprise en utilisant un formulaire de chat flottant et peu intrusif qui se déplace avec la page Web (avec

défilement), est totalement personnalisable et met également à jour instantanément toute personnalisation sans avoir besoin de redéployer le formulaire Web sur le site.

Contribution de Jayant Suneja, Arunabh Bhattacharjee, Cisco Engineering.

Journaux requis

Afin de suivre l'ensemble du flux, les journaux couvrent l'initiateur de chat (client) à l'UCCX.

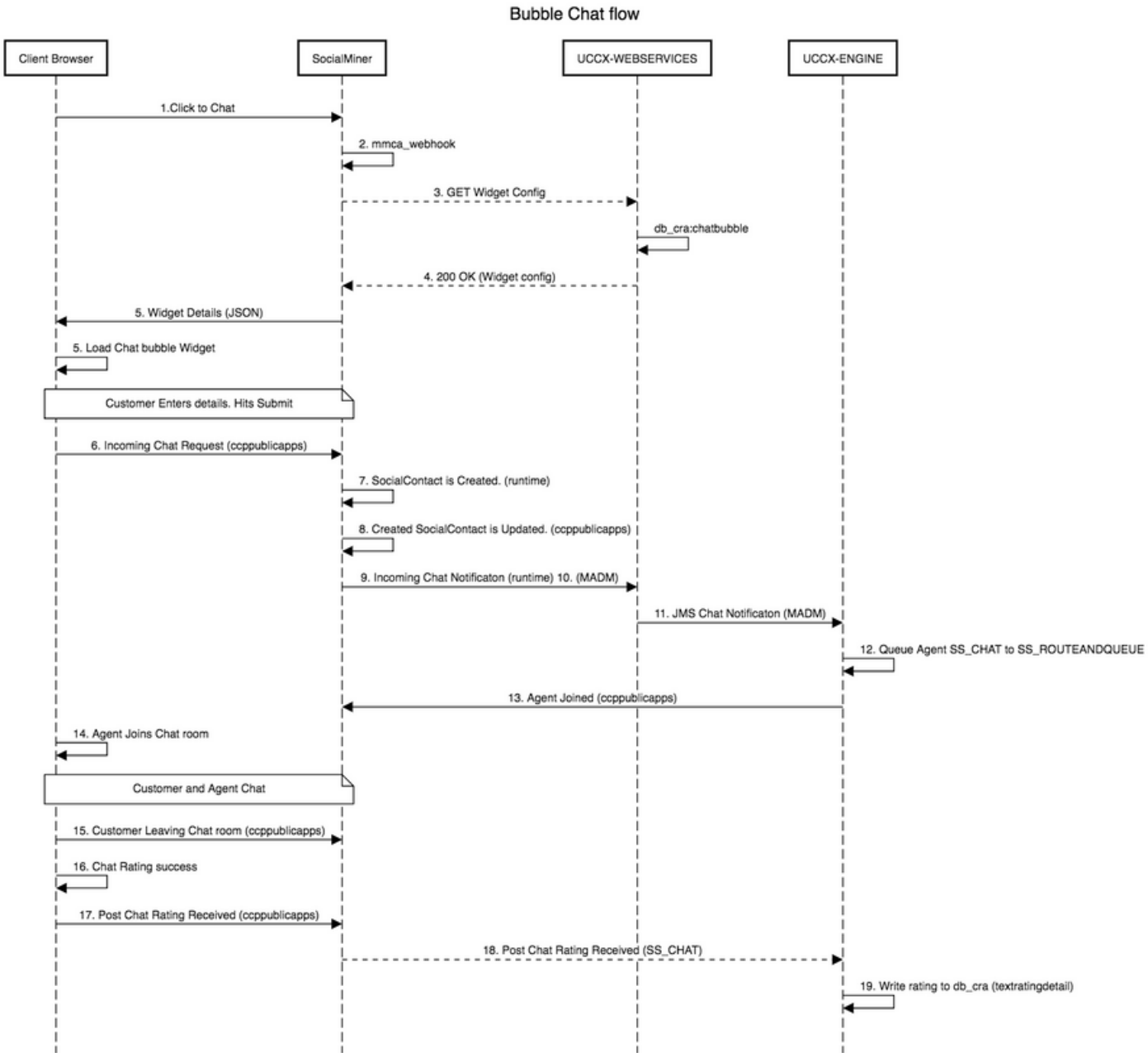
- Journaux de la console client : Il s'agit des journaux de console du navigateur dans lesquels l'utilisateur final lance la conversation.
- Journaux SM : **ccppublicapps** logs, **runtime** logs, **tomcat** logs.
- Journaux UCCX : Journaux MIVR (journaux du moteur avec débogage de **SS_CHAT** et **SS_ROUTEANDQUEUE**), journaux MADM (journaux d'**administration CCX** avec **UCCX_WEBSERVICES**).

Le reste de l'analyse (une fois la salle de discussion jointe) est identique au flux de discussion classique (c'est-à-dire un simple événement XMPP entre le gadget SocialMiner Agent sur Finesse et le client final - qui sont connectés via un tunnel XMPP)

Flux de discussion bulle

Le flux peut être résumé par les 19 étapes présentées dans cette image.

Chaque étape est décrite dans les journaux.



Analyse des journaux

Lorsqu'un client clique sur le bouton Cliquer pour discuter, le formulaire de discussion Bulle est présenté à l'aide du code JavaScript dans le widget de discussion.

Le formulaire de discussion en bulle effectue un rendu côté serveur, où il récupère la configuration de la discussion pour charger l'interface utilisateur de l'utilisateur final afin de démarrer une discussion.

Si vous ouvrez le widget de discussion, vous verrez l'URL configurée dans ce format :

https://<SOCIALMINER>**/ccp/ui/BubbleChat.html?host=**<SOCIALMINER>**&wid=**<WIDGET_ID>**&secure=true**

<SOCIALMINER> et <WIDGET_ID> sont respectivement le nom de domaine complet de SocialMiner et l'ID de widget.

La configuration reçue ici est au format JSON.

Étape 1. Journaux de la console client : Cliquez pour discuter.

1. Une fois que le client final appuie sur **Click to chat** sur la page Web, il effectue un GET vers SM via ces trois URL.

2. Notez que l'ID de widget est envoyé dans la troisième requête.

3. Tous ces devrait revenir avec un **200 OK** réussi afin de charger la fenêtre de discussion bulle.

```
1) GET https://sm-fqdn/ccp/ui/BubbleChat.html?host=sm-fqdn&wid=1&secure=true
[HTTP/1.1 200 OK 0ms]

2) GET https://sm-fqdn/ccp/ui/js/ccp/bubblechat/ccp-chat-components.js
[HTTP/1.1 200 OK 0ms]

3) GET https://sm-fqdn/ccp/bubblechat?wid=1
[HTTP/1.1 200 OK 289ms]
```

Étape 2. SM ccppublicapps : DB dip vers mmca_webhook

1. SM effectue un saut de base de données vers la table **mmca_webhook de** la base de données Informix SM et récupère l'URL Webhook pour cette demande de discussion entrante à l'aide de l'ID de widget. Dans notre cas, il s'agit de **widegetid=1**.

2. SM utilise ensuite cette URL de webhook pour récupérer la configuration du widget dans UCCX.

```
0000000323: 10.78.91.166: Aug 06 2018 09:24:09.538 +0530: %CCBU_________CCPAPI-6-REST_API_INFO:
Fetching bubble chat config from hook url:
https://uccx-fqdn/uccx-webservices/chat/config/1

0000000324: 10.78.91.166: Aug 06 2018 09:24:09.538 +0530: %CCBU_________CCPAPI-6-REST_API_INFO:
Fetching bubble chat config from hook url:
https://uccx-fqdn/uccx-webservices/chat/config/1
```

Étape 3. UCCX MADM : Reçoit une demande de configuration de widget.

1. Les services Web UCCX dans le cadre de MADM reçoivent cette demande pour obtenir la configuration du widget.

2. La configuration du widget contient les polices, les couleurs, le style du widget de chat, les informations de formulaire, etc.

3. UCCX effectue ensuite une analyse DB et récupère la configuration du widget dans la table **chatbubble** sur UCCX Informix **(db_cra)**.

4. Cette configuration est retournée au navigateur de l'utilisateur via l'API **ccppublicapps** comme réponse JSON.

```
7199526: Aug 06 09:31:21.235 IST %MADM-UCCX_WEBSERVICES-7-UNK:[http-bio-443-exec-14]
```

```
ServiceLogger:
Request Url: https://uccx-fqdn/uccx-webservices/chat/config/1
7199527: Aug 06 09:31:21.235 IST %MADM-UCCX_WEBSERVICES-7-UNK:[http-bio-443-exec-14]
ServiceLogger:
Match Found for https://uccx-fqdn/uccx-webservices/chat/config/1
7199528: Aug 06 09:31:21.235 IST %MADM-UCCX_WEBSERVICES-7-UNK:[http-bio-443-exec-14]
ServiceLogger:
ChatResource: Received a chat config GET request for chat widget id : 1
```

Étape 4. SM Publicapps:200 OK dans la configuration du widget.

SM reçoit le 200 OK pour la requête GET qu'il a envoyée à CCX pour récupérer la configuration du widget de conversation.

```
10.78.91.166: Aug 06 2018 09:31:21.297 +0530: %CCBU_________CCPAPI-6-REST_API_INFO:
Received success response within 100 milliseconds

10.78.91.166: Aug 06 2018 09:32:38.684 +0530: %CCBU__CCPPUBLICAPPS-6-PUBLICAPPS_SESSION_CREATED:
%[session.id=3F8B8C08D7E8144C7B1AD7AF144A4C1E][session_timeout=300]: ccppublicapps: new
publicapps
session
```

Étape 5. Journaux de la console client : Réponse au navigateur client qui affiche les détails dans JSON.

1. Voici un exemple de réponse JSON reçue côté client dans le cadre de toutes les opérations ci-dessus.

2. Ce fichier JSON est utilisé pour charger la fenêtre de discussion en bulles à travers la police configurée, l'énoncé du problème, la couleur, etc. telle qu'elle est reçue.

   Note: Toutes les opérations ci-dessus sont effectuées afin de charger la fenêtre de discussion en bulle afin que le client puisse maintenant remplir ses coordonnées pour commencer à discuter avec le centre de contact. Le flux suivant une fois que le client a envoyé les détails du chat est identique à celui du flux de chat classique/hérité.


```
GET https://sm-fqdn/ccp/bubblechat?wid=1[HTTP/1.1 200 OK 289ms]
Response:
{"feedId":"100000","postChatRatingEnabled":true,"messages":{"agentJoinTimeoutMsg":"All our
customer care representatives are busy. You may wait or try again
later.","transcriptPopupPositiveMsg":"Yes","transcriptPopupMsg":"Chat has ended. Do you want to
download the chat transcript?","transcriptPopupNegativeMsg":"No","connectivityErrorMsg":"Chat
disconnected due to inactivity timeout or connection failure.","agentLeftMsg":" ${agent_alias}
has left the chat","ratingButtonText":"Submit","offHourMessage":"Sorry, we are not available at
the moment.","textForTypingMsg":"Type your message and press Enter","agentJoinedMsg":"
${agent_alias} has joined","closeChatPopupMsg":"Do you want to close the
chat?","closeChatPopupNegativeMsg":"No","ratingLabel":"Rate your chat
experience","chatErrorMsg":"Chat service is currently unavailable. Try
later.","closeChatPopupPositiveMsg":"Yes","welcomeMessage":"Thank you for contacting us. A
customer care representative would assist you
soon."},"contextServiceFieldSets":"","chatForm":{"bubbleStyle":{"titleText":"CCBU Care
Startup","titleTextColor":"#0AB7D7","buttonText":"Start
Chat","buttonTextColor":"#FFFFFF","buttonBackgroundColor":"#0AB7D7","problemStmtCaption":"Choose
a problem statement","afterResumeNewChatMsg":"New
Message","agentMessageTextColor":"#FFFFFF","agentMessageBackgroundColor":"#0AC391","fontFace":"H
elvetica"},"formFields":["Title","Name","Details","Email","PhoneNumber","AddressLine1","AddressL
ine2","Anything important to note","Teléfono","Cédula"],"problemStatements":{"caption":"Choose a
```

problem
statement","statements":[{"statement":"Insuarance","csqTag":"Chat_Csq1"},{"statement":"Debit","c
sqTag":"Chat_Csq1"},{"statement":"Credit","csqTag":"Chat_Csq1"},{"statement":"Account","csqTag":
"Chat_Csq1"}]}},"proactiveChat":false,"name":"Bubble1162","id":1}

Étape 6. Applications publiques SM : Demande de conversation entrante.

La demande de discussion entrante est reçue par SM et un contact social est créé :

```
0000001136: 10.78.91.166: Aug 06 2018 09:32:38.692 +0530: %CCBU__CCPPUBLICAPPS-6-
CHAT_ROOM_DETAILS: %[Nickname=Jayant
Suneja][RoomJid=socialminer_chat.16@conference.127.0.0.1][RoomSubject=Chat with Jayant
Suneja][social_contact_id=null]: Create chat room with the following details
```

Étape 7. Exécution de SM : Le contact social est créé.

Le contact social est créé par le moteur SocialMiner (service Runtime).

```
0000399687: 10.78.91.166: Aug 06 2018 09:32:38.764 +0530: %CCBU__________FEEDS-6-
SOCIAL_CONTACT_CREATED:
%[SOCIAL_CONTACT_ID=0D66B2241000016500235A740A4E5BA6][SOCIAL_CONTACT_LINK=https://sm-
fqdn/ccp/socialcontact/0D66B2241000016500235A740A4E5BA6]: A new social contact was created
```

Étape 8. SM Publicapps : Met à jour le SocialContact créé vers l'API publique.

Le contact social est créé la mise à jour est visible dans les journaux **ccppublicapps**.

```
0000000337: 10.78.91.166: Aug 06 2018 09:32:38.767 +0530: %CCBU_________CCPAPI-6-
CREATE_SOCIAL_CONTACT: SocialContact successfully created with requestId: y3xQxE
0000001142: 10.78.91.166: Aug 06 2018 09:32:38.768 +0530: %CCBU__CCPPUBLICAPPS-6-CHAT_CREATED:
%[Contact=https://sm-fqdn/ccp-
webapp/ccp/socialcontact/0D66B2241000016500235A740A4E5BA6][Room=socialminer_chat.16@conference.1
27.0.0.1][Session=3F8B8C08D7E8144C7B1AD7AF144A4C1E][social_contact_id=null]: Chat created
```

Étape 9. Exécution de SM : Envoie la notification à l'API CCX Webservice sur MADM.

Une notification est envoyée du SM aux services Web CCX informant CCX de l'arrivée d'une nouvelle discussion.

```
0008541000: 10.78.91.166: Aug 06 2018 09:32:38.940 +0530: %CCBU___NOTIFICATION-6-
NOTIFICATION_SUCCEEDED:
%[Notification=com.cisco.ccbu.ccp.notification.Notification@1beea47[enqueueTime=1533528158834,ru
le=com.cisco.ccbu.oamp.omgr.transfer.HttpNotificationRule@15a2eb1[httpUrl=https://uccx-
fqdn/uccx-webservices/contact,httpUsername= Chat Notification for HA -
Backup,description=Created by CCX application as part of CCX chat
configuration.,changeStamp=1,lastUpdated=1527579740301,ouId=4],screenPopUrl=http://sm-
fqdn/results.jsp?scID=0D66B2241000016500235A740A4E5BA6&campaignID=CCX Chat Campaign,body=
version="1.0" encoding="UTF-8" standalone="yes"?>
<SocialContact>
<author>Jayant Suneja</author>
<description></description>
```

<extensionFields>
<extensionField>
<name>AddressLine1</name>
<value>1900</value>
</extensionField>
<extensionField>
<name>AddressLine2</name>
<value>9th street</value>

Étape 10. UCCX MADM : Webservices affiche la demande de conversation entrante.

CCX reçoit un contact de discussion entrant sur MADM avec UCCX_WEBSERVICES activé pour le débogage.

```
7199571: Aug 06 09:32:39.079 IST %MADM-UCCX_WEBSERVICES-7-UNK:[http-bio-443-exec-7]
ServiceLogger: New incoming contact : <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<contactXML>
<author>Jayant Suneja</author>
<id>0D66B2241000016500235A740A4E5BA6</id>
<integrationAuthTokenGUID>d64fe6ac-9687-4ecd-9f00-0e6ec02c6945</integrationAuthTokenGUID>
<notificationTag>ccx_chat_req</notificationTag>
<publishedDate>1533528158684</publishedDate>
<refURL>http://sm-fqdn/ccp-webapp/ccp/socialcontact/0D66B2241000016500235A740A4E5BA6>
<replyTemplateURL>http://sm-fqdn/multisession/ui/templates/reply/cisco_agent_ms_chat.jsp>
<sourceType>chat</sourceType>
<status>unread</status>
<statusReason>unknown</statusReason>
<statusTimestamp>1533528158756</statusTimestamp>
<tags>
```

Étape 11. UCCX MADM : Envoyez un message JMS au moteur CCX.

Le moteur est maintenant informé du nouveau contact via le bus de messages JMS, c'est-à-dire que UCCX MADM (Administration Webservice API) informe le MIVR (Engine) de mettre ce contact en file d'attente.

```
7199573: Aug 06 09:32:39.080 IST %MADM-UCCX_WEBSERVICES-7-UNK:[http-bio-443-exec-7]
ServiceLogger: Sending JMS message to Contact Topic: ContactEvent [operation=ADD,
contactXML=ContactXML [author=Jayant Suneja, id=0D66B2241000016500235A740A4E5BA6,
replyType=chat, notificationTag=ccx_chat_req, refURL=http://sm-fqdn/ccp-
webapp/ccp/socialcontact/0D66B2241000016500235A740A4E5BA6, replyTemplateURL=http://sm-
fqdn/multisession/ui/templates/reply/cisco_agent_ms_chat.jsp, integrationAuthTokenGUID=d64fe6ac-
9687-4ecd-9f00-0e6ec02c6945, userExtensionFields=[UserExtensionField [name=AddressLine1,
value=1900], UserExtensionField [name=AddressLine2, value=9th street], UserExtensionField
[name=Anything important to note, value=Bill Issue], UserExtensionField [name=Cédula, value=Bill
Issue], UserExtensionField [name=Details, value=Test Bubble Chat], UserExtensionField
[name=Email, value=xxxx@gmail.com], UserExtensionField [name=Name, value=Jayant Suneja],
UserExtensionField [name=PhoneNumber, value=97xxxxxxx], UserExtensionField [name=Teléfono,
value=Bill Issue], UserExtensionField [name=Title, value=Mr.], UserExtensionField
[name=ccxqueuetag, value=Chat_Csq1], UserExtensionField [name=h_chatOrigin, value=CISCO_BUBBLE],
UserExtensionField [name=h_chatRoom, value=socialminer_chat.16@conference.127.0.0.1],
UserExtensionField [name=h_widgetName, value=Bubble1162]], tags=[ccx_chat_req], sourceType=chat,
```

```
status=unread, statusReason=unknown, publishedDate=1533528158684, title=CCX_Chat,
campaignPublicId=null, statusTimestamp=1533528158756]]
```

Étape 12. MIVR UCCX :

Le sous-système de conversation (SS_CHAT) met en file d'attente le contact de conversation et le sous-système de routage et de file d'attente (SS_ROUTEANDQUEUE) attribue la conversation à l'agent. Ceci est visible dans les journaux du moteur UCCX (MIVR) avec SS_CHAT et SS_ROUTEANDQUEUE activés pour le débogage.

```
3722751: Aug 06 09:32:39.144 IST %MIVR-SS_CHAT-7-UNK:[pool-8-thread-15] ChatSubsystemLogger:
com.cisco.wf.subsystems.chat.observer.ContactNotificationObserver : Contact Notification
received - ContactNotification [requestId=, contactId=0D66B2241000016500235A740A4E5BA6,
contactState=QUEUED, handlerQueue=-1, startTime=1533528159105, endTime=-1, queueWaitDuration=0,
handlingDuration=-1,
terminalStateContact=ChatContact(Contact[id=0D66B2241000016500235A740A4E5BA6,state=QUEUED,dispRe
ason=UNKNOWN]), dispositionReason=UNKNOWN, mediaType=CHAT, sequenceNumber=0,
statusTimeStamp=1533528158756]. Notification Type UPDATE

3722752: Aug 06 09:32:39.144 IST %MIVR-SS_ROUTEANDQUEUE-7-UNK:[pool-8-thread-15]
RouteAndQueueSubsystemLogger: ALLOCATED Contact [0D66B2241000016500235A740A4E5BA6] to Agent
[Jabber@ge.nt1]
```

L'agent reçoit une notification pour accepter la discussion et une salle est créée entre l'utilisateur et l'agent.
Le flux de tous ces mêmes que la conversation héritée donc ne couvrant pas beaucoup de journaux pour ce flux.

Étape 13. Applications publiques SM : La session Jointures et discussions de l'agent a démarré.

La session de discussion a été établie une fois le tunnel XMPP créé. Le widget de chat de l'utilisateur et le gadget de chat SocialMiner (hébergé sur Finesse) sont maintenant connectés via XMPP et vous pouvez voir les événements de présence envoyés et reçus.

```
0000001149: 10.78.91.166: Aug 06 2018 09:32:46.842 +0530: %CCBU__CCPPUBLICAPPS-6-
PRESENCE_EVENT_RECEIVED: %[CustomerNickname=Jayant
Suneja][Nickname=Jabber@ge.nt1][SocialContactID=0D66B2241000016500235A740A4E5BA6][Status=joined]
[Username=socialminer_chat.16@conference.127.0.0.1/Jabber@ge.nt1]: Received presence event
0000001152: 10.78.91.166: Aug 06 2018 09:32:51.168 +0530: %CCBU__CCPPUBLICAPPS-6-
TYPING_STATUS_RECEIVED:
%[From=socialminer_chat.16@conference.127.0.0.1/Jabber@ge.nt1][Nickname=Jabber@ge.nt1][SocialCon
tactID=0D66B2241000016500235A740A4E5BA6][Text=composing]: Typing Status received
```

Étape 14. Journaux de la console client : Un agent rejoint la salle de discussion.

Les journaux côté client indiquent que l'agent rejoint la salle de discussion. La même chose est affichée dans le widget de discussion.

```
Mon Aug 06 2018 09:32:48 GMT+0530 (India Standard Time):CISCO_BUBBLE_CHAT:Received chat room
status event of type chatroom_agent_joined
```

> **Note**: Le client met fin à la discussion. Ici, le flux est un peu différent par rapport à l'ancien chat car il y a une nouvelle fonctionnalité de **post chat évaluation** ajoutée avec bubble chat.

Étape 15. SM Publicapps : l'utilisateur quitte la salle de discussion.

Un utilisateur quitte la discussion et l'agent est désormais seul dans la salle de discussion. Cela indique également que l'utilisateur quitte la salle.

```
0000001185: 10.78.91.166: Aug 06 2018 09:33:26.738 +0530: %CCBU__CCPPUBLICAPPS-6-
LEAVE_CHAT_ROOM:
%[ChatRoomName=socialminer_chat.16@conference.127.0.0.1][Nickname=Jayant Suneja]
[User=ccp_chatclient][social_contact_id=0D66B2241000016500235A740A4E5BA6]: Leave chat room
```

> **Note**: Si un classement de post-discussion est activé, le SM reçoit ce classement une fois que l'utilisateur a envoyé le classement de post-discussion.

Étape 16. Journaux de la console client : Le navigateur utilisateur affiche l'évaluation des conversations réussies.

Envoi réussi de l'évaluation de conversation à l'aide d'un 200 OK reçu sur le navigateur de l'utilisateur.

```
Mon Aug 06 2018 09:33:34 GMT+0530 (India Standard Time):CISCO_BUBBLE_CHAT:Feedback submitted
successfully. Response code: 200
Mon Aug 06 2018 09:33:34 GMT+0530 (India Standard Time):CISCO_BUBBLE_CHAT:Successfully ended
the chat session. Response code: 200 ccp-chat-components.js:1:14634
Mon Aug 06 2018 09:33:34 GMT+0530 (India Standard Time):CISCO_BUBBLE_CHAT:Successfully cleaned
up
```

Étape 17. Applications publiques SM : ccppublicapps reçoit l'évaluation post-discussion du navigateur.

Évaluation des commentaires reçus au SM.

```
10.78.91.166: Aug 06 2018 09:33:34.277 +0530: %CCBU__CCPPUBLICAPPS-6-CHAT_FEEDBACK_RECEIVED :
%[ChatFeedback=com.cisco.ccbu.ccp.publicapps.api.chat.ChatFeedback@d82623[rating=4]][Session=
3F8B8C08D7E8144C7B1AD7AF144A4C1E][social_contact_id=0D66B2241000016500235A740A4E5BA6]:
Received chat feedback
```

> **Note**: Lorsque des commentaires de discussion sont reçus au niveau du service de messagerie avec des informations de notation, ces informations sont d'abord enregistrées dans le contact social sur le data store du service de messagerie avant d'avertir CCX. Si le data store SM est en panne, l'envoi de l'évaluation de conversation échoue avec un extrait de « CCPPUBLICAPPS-3-UPDATE_CHAT_SOCIALCONTACT_EXTENSION_FIELD_FAILED »

Étape 18. MIVR UCCX : Mise à jour XMPP reçue du SM.

SM envoie une mise à jour XMPP à CCX en l'informant de l'évaluation reçue de l'utilisateur final.

```
3723269: Aug 06 09:33:34.290 IST %MIVR-SS_CHAT-7-UNK:[Smack Packet Reader (1)]
ChatSubsystemLogger: createXMPPSession:
PacketExtensionProvider.parseExtension:<SocialContact><campaignPublicId>CCX_Chat_Campaign</campa
ignPublicId><extensionFields><extensionField><name>h_chatRoom</name><value>socialminer_chat.16@c
onference.127.0.0.1</value></extensionField><extensionField><name>h_chatOrigin</name><value>CISC
O_BUBBLE</value></extensionField><extensionField><name>C&#233;dula</name><value>Bill
Issue</value></extensionField><extensionField><name>Details</name><value>Test Bubble
Chat</value></extensionField><extensionField><name>PhoneNumber</name><value>97xxxxxxx</value></
extensionField><extensionField><name>h_agentName</name><value>Jabber@ge.nt1</value></extensionFi
eld><extensionField><name>h_rating</name><value>4</value>
```

Étape 19. MIVR UCCX : Écrire le classement de conversation dans la base de données.

L'évaluation de chat soumise est écrite dans la base de données CCX et enregistrée dans la table **textratingdetail**, qui est une table nouvellement ajoutée sur UCCX 11.6(2).

L'objectif principal de ce tableau est de stocker les évaluations de conversation à des fins de reporting.

```
3723276: Aug 06 09:33:34.299 IST %MIVR-SS_ROUTEANDQUEUE-7-UNK:[Smack Listener Processor (1)]
RouteAndQueueSubsystemLogger: com.cisco.wf.subsystems.routeandqueue.aggregator.historical.
HistoricalManager : Writing Historical Record: TRDR: ContactID=0D66B2241000016500235A740A4E5BA6,
rating=4, ratingTime=java.util.GregorianCalendar[time=1533528214299,areFieldsSet=true,
areAllFieldsSet=true,lenient=true,zone=sun.util.calendar.ZoneInfo[id="GMT",offset=0,dstSavings=0
,
useDaylight=false,transitions=0,lastRule=null],firstDayOfWeek=1,minimalDaysInFirstWeek=1,ERA=1,
YEAR=2018,MONTH=7,WEEK_OF_YEAR=32,WEEK_OF_MONTH=2,DAY_OF_MONTH=6,DAY_OF_YEAR=218,DAY_OF_WEEK=2,
DAY_OF_WEEK_IN_MONTH=1,AM_PM=0,HOUR=4,HOUR_OF_DAY=4,MINUTE=3,SECOND=34,MILLISECOND=299,
ZONE_OFFSET=0,DST_OFFSET=0]
```

À propos de cette traduction
Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.

À propos de cette traduction
Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.