

UCCX und SocialMiner: Analyse des Bubble Chat- und Post Chat-Bewertungsprotokolls

Inhalt

[Einleitung](#)

[Erforderliche Protokolle](#)

[Fluss des Blasenchats](#)

[Protokollanalyse](#)

[Schritt 1. Client Console protokolliert: Zum Chat klicken.](#)

[Schritt 2: SM-Cublicapps: DB dip to mmca_webhook](#)

[Schritt 3: UCCX MADM: Empfängt Anforderung für die Widget-Konfiguration.](#)

[Schritt 4. SM Publicapps:200 OK auf der Widget-Konfiguration.](#)

[Schritt 5: Protokolle der Client-Konsole: Antwort auf den Clientbrowser, der Details in JSON anzeigt.](#)

[Schritt 6. SM Publicapps: Eingehende Chat-Anfrage](#)

[Schritt 7: SM-Laufzeit: Sozialer Kontakt wird hergestellt.](#)

[Schritt 8: SM Publicapps: Aktualisiert den erstellten SocialContact auf die öffentliche API.](#)

[Schritt 9: SM-Laufzeit: Sendet die Benachrichtigung an die CCX-Webservice-API auf MADM.](#)

[Schritt 10: UCCX MADM: Webservices zeigt eingehende Chat-Anfragen an.](#)

[Schritt 11: UCCX MADM: Senden Sie die JMS-Nachricht an CCX Engine.](#)

[Schritt 12: UCCX MIVR:](#)

[Schritt 13. SM Publicapps: Die Agent-Teilnahme und Chats-Sitzung wurde gestartet.](#)

[Schritt 14. Protokolle der Client-Konsole: Agent nimmt am Chatraum teil](#)

[Schritt 15. SM Publicapps: Benutzer verlässt den Chat-Raum.](#)

[Schritt 16. Protokolle der Client-Konsole: Der Benutzer-Browser zeigt die erfolgreiche Chat-Bewertung an.](#)

[Schritt 17. SM Publicapps: ccublicapps erhält die Bewertung "Nach dem Chat" vom Browser.](#)

[Schritt 18: UCCX MIVR: XMPP-Update von SM empfangen.](#)

[Schritt 19: UCCX MIVR: Chat-Bewertung in Datenbank schreiben.](#)

Einleitung

In diesem Dokument wird der Blasenchat-Fluss zusammen mit der End-to-End-Protokollanalyse für einen funktionierenden Blasenchat beschrieben. Dieser kann als funktionierende Referenz zur Fehlerbehebung verwendet werden.

Die Unified Contact Center Express (UCCX)-Lösung mit der Einführung von UCCX und SocialMiner (SM) 11.6(2) hat die neue Bubble Chat-Funktion hinzugefügt.

Die Bubble Chat (oder Chat Bubble) Funktion ermöglicht es Ihnen, ein Unternehmen mit einem minimal aufdringlichen, schwebenden Chat-Webformular zu erreichen, das mit der Webseite (mit Scrollen) bewegt wird, vollständig anpassbar ist und auch sofort jede Anpassung aktualisiert, ohne dass das Webformular auf der Website neu bereitgestellt werden muss.

Erforderliche Protokolle

Um den gesamten Fluss zu verfolgen, werden Protokolle über den Chat-Initiator (Kunden) an UCCX erstellt.

- Protokolle der Client-Konsole: Dies sind die Browser-Konsolenprotokolle, in denen der Endbenutzer den Chat initiiert.
- SM-Protokolle: **ccublicapps** Protokolle, **Laufzeitprotokolle**, **Tomcat** Protokolle.
- UCCX-Protokolle: MIVR-Protokolle (Engine-Protokolle mit **SS_CHAT** und **SS_ROUTEANDQUEUE**-Debugging), MADM-Protokolle (**CCX-Admin**-Protokolle mit **UCCX_WEBSERVICES**).

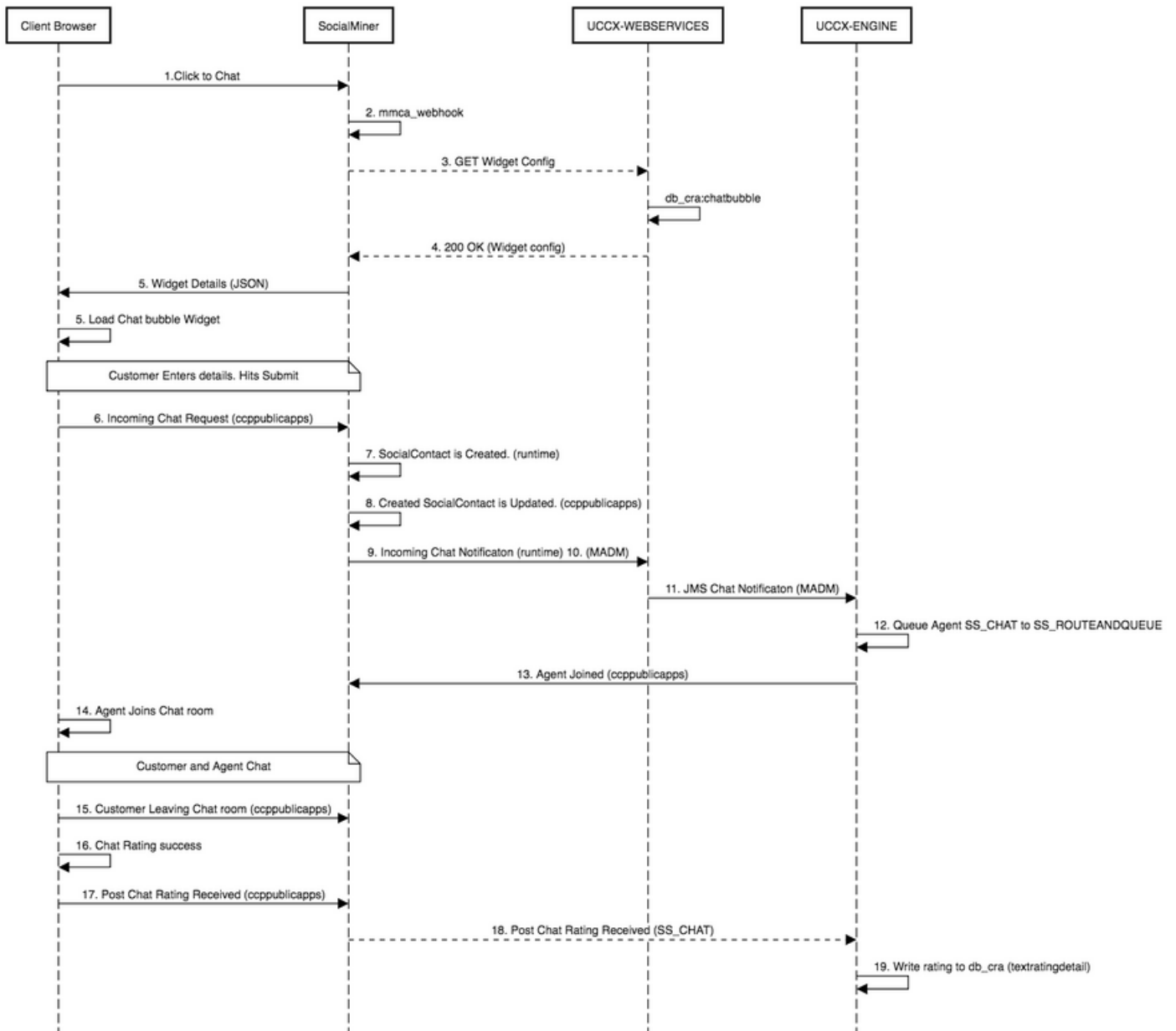
Der Rest der Analyse (nach dem Beitritt des Chatraums) ist identisch mit dem klassischen Chat-Fluss (d. h. einfache XMPP-Ereignisse zwischen dem SocialMiner Agent-Gadget auf Finesse und dem Endkunden, die über einen XMPP-Tunnel verbunden sind).

Fluss des Blasenchats

Der Fluss kann durch die in diesem Bild dargestellten 19 Schritte zusammengefasst werden.

Jeder Schritt wird in den Protokollen aufgeführt.

Bubble Chat flow



Protokollanalyse

Wenn ein Kunde auf die Schaltfläche Click to chat klickt, wird das Bubble-Chat-Formular mit dem JavaScript-Code im Chat-Widget angezeigt.

Das Bubble-Chat-Formular führt ein serverseitiges Rendering durch, bei dem die Chat-Konfiguration abgerufen wird, um die Benutzeroberfläche für den Endbenutzer zum Starten eines Chats zu laden.

Wenn Sie das Chat-Widget öffnen, wird die URL in folgendem Format konfiguriert:

`https://<SOCIALMINER>/ccp/ui/BubbleChat.html?host=<SOCIALMINER>&wid=<WIDGET_ID>&secure=true`
 <SOCIALMINER> und <WIDGET_ID> sind die FQDN von SocialMiner bzw. die Widget-ID.

Die hier erhaltene Konfiguration liegt im JSON-Formular vor.

Schritt 1. Client Console protokolliert: Zum Chat klicken.

1. Sobald der Endkunde auf der Webseite **Click to chat** drückt, erhält er SM über diese drei URLs.
2. Beachten Sie, dass die Widget-ID in der dritten Anfrage gesendet wird.
3. All diese sollten mit einem erfolgreichen **200 OK** zurückkehren, um Bubble-Chat-Fenster zu laden.

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]

Schritt 2: SM-Cublicapps: DB dip to mmca_webhook

1. SM führt ein db-Dip zur **mmca_webhook**-Tabelle in der SM Informix-Datenbank durch und ruft die Webhook-URL für diese eingehende Chat-Anfrage mithilfe der Widget-ID ab. In unserem Fall ist es **widegetid=1**.
2. SM verwendet dann diese Webhook-URL, um die Widget-Konfiguration von UCCX abzurufen.

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>

Schritt 3: UCCX MADM: Empfängt Anforderung für die Widget-Konfiguration.

1. Die UCCX-Webservices als Teil von MADM erhalten diese Anforderung für die Widget-Konfiguration.
2. Die Widget-Konfiguration enthält Schriftarten, Farben, das Format des Chat-Widgets, Formulareinfos usw.
3. UCCX führt dann ein DB-Dip aus und ruft die Widget-Konfiguration aus der **Chat-Bubble**-Tabelle in UCCX Informix (**db_cra**) ab.
4. Diese Konfiguration wird über die API ccublicapps als JSON-Antwort an den Browser des Benutzers zurückgegeben.

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

Schritt 4. SM Publicapps:200 OK auf der Widget-Konfiguration.

SM erhält die 200 OK für die GET-Anforderung, die es an CCX gesendet hat, um die Chat-Widget-Konfiguration abzurufen.

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

Schritt 5: Protokolle der Client-Konsole: Antwort auf den Clientbrowser, der Details in JSON anzeigt.

1. Hier sehen Sie ein Beispiel für eine JSON-Antwort, die auf der Client-Seite im Rahmen der oben genannten Vorgänge empfangen wurde.
2. Diese JSON wird verwendet, um das Bubble-Chat-Fenster durch die konfigurierte Schriftart, Problemanweisung, Farbe usw. zu laden, wie es empfangen wurde.

Anmerkung: Alle oben genannten Vorgänge werden ausgeführt, um das Fenster für den Bubble-Chat zu laden. Der Kunde kann nun seine Kontaktdetails eingeben, um mit dem Contact Center zu chatten. Der weitere Ablauf ist identisch mit dem des klassischen/älteren Chat-Ablaufs, sobald der Kunde die Chat-Details übermittelt.

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", "problemStmntCaption": "Choose a problem statement", "afterResumeNewChatMsg": "New Message", "agentMessageTextColor": "#FFFFFF", "agentMessageBackgroundColor": "#0AC391", "fontFace": "Helvetica" }, "formFields": [ "Title", "Name", "Details", "Email", "PhoneNumber", "AddressLine1", "AddressLine2", "Anything important to note", "Teléfono", "Cédula" ], "problemStatements": { "caption": "Choose a problem statement", "statements": [ { "statement": "Insuarance", "csqTag": "Chat_Csql" }, { "statement": "Debit", "csqTag": "Chat_Csql" }, { "statement": "Credit", "csqTag": "Chat_Csql" }, { "statement": "Account", "csqTag": "Chat_Csql" }
```

```
"Chat_Csqr1" ] ] } } , " proactiveChat " : false , " name " : " Bubble1162 " , " id " : 1 }
```

Schritt 6. SM Publicapps: Eingehende Chat-Anfrage

Die eingehende Chat-Anfrage wird von SM empfangen, und der soziale Kontakt wird erstellt:

```
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
```

Schritt 7: SM-Laufzeit: Sozialer Kontakt wird hergestellt.

Social-Media-Kontakte werden von der SocialMiner-Engine (Runtime-Dienst) erstellt.

```
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
```

Schritt 8: SM Publicapps: Aktualisiert den erstellten SocialContact auf die öffentliche API.

Social Contact erstellt Update ist in ccublicapps Protokolle zu sehen.

```
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.127.0.0.1][Session=3F8B8C08D7E8144C7B1AD7AF144A4C1E][social_contact_id=null]: Chat created
```

Schritt 9: SM-Laufzeit: Sendet die Benachrichtigung an die CCX-Webservice-API auf MADM.

Von SM wird eine Benachrichtigung an CCX-Webdienste gesendet, die CCX über einen neuen eingehenden Chat informiert.

```
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@1bbee47[enqueueTime=1533528158834,rule=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>
```

Schritt 10: UCCX MADM: Webservices zeigt eingehende Chat-Anfragen an.

Der eingehende Chat-Kontakt wird von CCX auf MADM empfangen, wobei UCCX_WEBSERVICES für das Debugging aktiviert ist.

```
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>
```

Schritt 11: UCCX MADM: Senden Sie die JMS-Nachricht an CCX Engine.

Die Engine wird nun über den JMS-Message-Bus über den neuen Kontakt informiert, d. h. UCCX MADM (Administration Webservice API) informiert den MIVR (Engine), dass dieser Kontakt in die Warteschlange gestellt werden soll.

```
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]]
```

Schritt 12: UCCX MIVR:

Das Chat-Subsystem (SS_CHAT) stellt den Chat-Kontakt in die Warteschlange, und das Route and Queue-Subsystem (SS_ROUTEANDQUEUE) weist den Chat dem Agenten zu. Dies wird in den UCCX Engine-Protokollen (MIVR) mit aktivierten SS_CHAT- und SS_ROUTEANDQUEUE-Funktionen zum Debuggen angezeigt.

```
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]
```

Der Agent erhält eine Benachrichtigung, dass er den Chat akzeptiert, und es wird ein Raum zwischen Benutzer und Agent erstellt.

Der Fluss von all dem gleichen wie Legacy-Chat, sodass nicht viel von Protokollen für diesen Fluss abzudecken.

Schritt 13. SM Publicapps: Die Agent-Teilnahme und Chats-Sitzung wurde gestartet.

Die Chat-Sitzung wurde eingerichtet, sobald der XMPP-Tunnel erstellt wurde. Das Chat-Widget des Benutzers und das SocialMiner-Chat-Gadget (auf Finesse gehostet) sind jetzt über XMPP verbunden, und Sie können sehen, welche Presence-Ereignisse gesendet und empfangen werden.

```
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
```

Schritt 14. Protokolle der Client-Konsole: Agent nimmt am Chatraum teil

Die clientseitigen Protokolle zeigen an, dass der Agent dem Chatraum beitrifft. Dasselbe wird im Chat-Widget angezeigt.

```
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
```

Anmerkung: Der Kunde beendet jetzt den Chat. Hier ist der Fluss ein bisschen anders als im Vergleich zu Legacy-Chat, da es eine neue Funktion der **Post-Chat-Bewertung** mit Bubble-Chat hinzugefügt.

Schritt 15. SM Publicapps: Benutzer verlässt den Chat-Raum.

Ein Benutzer verlässt den Chat und nun befindet sich der Mitarbeiter alleine im Chat-Raum. Dies zeigt auch, dass der Benutzer den Raum verlässt.

```
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
```

Anmerkung: Wenn eine Bewertung nach dem Chat aktiviert ist, erhält das SM diese Bewertung, sobald der Benutzer die Bewertung nach dem Chat einreicht.

Schritt 16. Protokolle der Client-Konsole: Der Benutzer-Browser zeigt die erfolgreiche Chat-Bewertung an.

Erfolgreiche Übermittlung der Chat-Bewertung mit einem 200 OK, das im Benutzerbrowser empfangen wurde.

```
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
```

Schritt 17. SM Publicapps: ccpublicapps erhält die Bewertung "Nach dem Chat" vom Browser.

Feedback zur Bewertung wurde bei SM empfangen.

```
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
```

Anmerkung: Wenn bei SM ein Chat-Feedback mit Bewertungsinformationen eingeht, werden diese Informationen zuerst in einem sozialen Kontakt im SM-Datenspeicher gespeichert, bevor CCX benachrichtigt wird. Falls der SM-Datenspeicher ausgefallen ist, schlägt die Übermittlung der Chat-Bewertung mit dem Ausschnitt "CCPPUBLICAPPS-3-UPDATE_CHAT_SOCIALCONTACT_EXTENSION_FIELD_FAILED" fehl.

Schritt 18: UCCX MIVR: XMPP-Update von SM empfangen.

SM sendet ein XMPP-Update an CCX, das die vom Endbenutzer erhaltene Bewertung bestätigt.

```
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</campaignPublicId><extensionFields><extensionField><name>h_chatRoom</name><value>socialminer_chat.16@conference.127.0.0.1</value></extensionField><extensionField><name>h_chatOrigin</name><value>CISCO_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></extensionField><extensionField><name>h_rating</name><value>4</value>
```

Schritt 19: UCCX MIVR: Chat-Bewertung in Datenbank schreiben.

Die eingesendete Chat-Bewertung wird in die CCX-Datenbank geschrieben und in der Tabelle mit den Textinformationen gespeichert, die neu in UCCX 11.6(2) hinzugefügt wurde.

Der Hauptzweck dieser Tabelle besteht darin, Chat-Bewertungen für Berichtszwecke zu speichern.

```
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]
```

Informationen zu dieser Übersetzung

Cisco hat dieses Dokument maschinell übersetzen und von einem menschlichen Übersetzer editieren und korrigieren lassen, um unseren Benutzern auf der ganzen Welt Support-Inhalte in ihrer eigenen Sprache zu bieten. Bitte beachten Sie, dass selbst die beste maschinelle Übersetzung nicht so genau ist wie eine von einem professionellen Übersetzer angefertigte. Cisco Systems, Inc. übernimmt keine Haftung für die Richtigkeit dieser Übersetzungen und empfiehlt, immer das englische Originaldokument (siehe bereitgestellter Link) heranzuziehen.