GENESYS™

Chat Server Administration Guide

Contact Identification for Regular and Anonymous Chat

4/4/2025

Contact Identification for Regular and Anonymous Chat

Whenever a new chat session is being created, Chat Server communicates with UCS to create a chat interaction record where the chat transcript of a completed chat session is stored as a part of the contact history. Each interaction record can be associated with a contact record that identifies and describes a customer profile. Such an association is done automatically by UCS during the interaction record creation, based on the presence of "contact attributes" in the attached data.

Contact attributes can be comprised of a default set of attributes, which are defined for each interaction type (chat, email, voice). Additionally, or alternatively, custom contact attributes can be configured. You can find more information on this on the Contact Identification page, in the eServices Administrator's Guide.

Anonymous chat recommendations

If a large volume of anonymous chats is expected, a solution must be configured to avoid an association of all anonymous interactions with a single contact record (which often leads to performance implications). There are two alternative approaches to achieve this:

- Recommended: no default or custom contact attributes are provided when starting the chat session. See below on how to achieve this with the Genesys Chat Widget or through the GMS API.

- Alternative: If for some reason, there is a need to associate every chat record with a contact record, then a special custom attribute must be defined (for example: `ExternalCustomerId`), and it must be configured either as the only contact attribute for the media type, or it must be set with a highest priority if several contact attributes are defined (see Customize Contact Identification per Media Type for directions on how to configure this). The value for this attribute can be assigned either arbitrarily (for example, a new UUID each time) or with values coming from a customer CRM solution.

> ### Important
>
> Genesys strongly recommends not using First and Last name with arbitrary values for this purpose as it can lead to performance implications due to the specifics of the contact identification algorithm. `EmailAddress` should not be used either with random values as the email address can potentially be used later by other components (in other words, when sending the transcript email).

The list of contact attributes for a new chat session can come through the GMS API in Request Chat from:

- Genesys Chat Widget. In this case, the Widget must be launched without asking for contact attributes, but instead by only sending a nickname (which is not being used for contact identification, but is required by GMS API if first/last name is absent). The nickname can either be hard-coded into the Widget or requested on the initial Widget screen. For example, from "launcher.html" this can be done by adding the following to "webchat" under the "Additional Configuration Options" section:

```
{
  "form":{
    "wrapper":"<table></table>",
      "inputs":[
        {
          "id":"cx_webchat_form_nickame",
          "name":"nickname",
          "maxlength":"100",
          "placeholder":"mandatory",
          "label":"Nick Name"
        },
        {
          "id":"cx_webchat_form_subject",
          "name":"subject",
          "maxlength":"100",
          "placeholder":"@i18n:webchat.ChatFormPlaceholderSubject",
          "label":"@i18n:webchat.ChatFormSubject"
        }
      ]
  }
}
```

This defines the nickname instead of First and Last name. Alternatively, the subject can also be removed completely from the initial form. For more information about Widget configuration, see Customizable Chat Registration Form.

- A custom web chat application. In this case, only the "nickname" must be provided in "Request Chat", while `firstName`, `lastName`, and `emailAddress` must not be sent.

How contact identification works for chat

If the user data includes an attribute called `EmailAddress`, UCS looks for a contact in its database whose `EmailAddress` attribute has the same value as the user data attribute. The name of the user data attribute must be exactly `EmailAddress` —if it is `email_address` or anything else, UCS will not try to match its value with the stored value of `EmailAddress`. Or, if UCS finds no matching contact, it creates a new one using the user data.

For either a matching contact or a new one, UCS sends the following, as data about the contact for this interaction, to Chat Server:

- The matched attribute (if not email address, then phone number, and so on).

- The attribute `ContactID`.

- All other attributes of this contact that UCS has stored in its database, except:

- If any user data has an attribute name that matches an attribute name in the UCS `Contacts` table, UCS returns the value of the attribute from the user data, not the value from the `Contacts` table. It does not modify the value in the `Contacts` table.

The last point can cause a problem, as in the following example:

1. Home user Steve Jones wants to open a chat session. In the web interface, he types in his correct email address sjones@here, then erroneously types his first name as Speve.

2. UCS finds a contact record for sjones@here.

3. UCS returns to Chat Server data about an existing contact whose email address is sjones@here and whose first name is Speve. UCS still has the correct first name Steve in its database, but the user data, with the erroneous Speve, preempts the correct data for the purposes of this chat interaction.

4. The system uses the user data to generate the message prompt that marks the home user in the chat display. As a result, the chat session displays something like the following:
    ```
    14:52:20 SpeveJ has joined the session
    14:52:30 SpeveJ > Hi.
    ```

5. The Agent Desktop displays the incorrect first name (in the user data on the lower left pane) and the correct first name (on the Customer Records pane on the right). The agent sees the incorrect first name and opens the chat session by typing, "Hello Speve, how can I help you?"

6. The interaction passes through a strategy that generates an automatic response, which opens, "It was good chatting with you, Speve."

To avoid this type of problem, be sure that the system (including strategies and desktop) as well as its users refer to the UCS database, rather than user data, for contact attributes. In the example just cited, the agent must be sure to look at the Customer Records (right-hand) pane of the Desktop for the name of the contact.

It is also advisable to closely monitor the inventory of contact attributes that can become user data.