



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Composer Help

Chat Messages in Queue

Contents

- [1 Chat Messages in Queue](#)
 - [1.1 Resolution](#)

Chat Messages in Queue

When a chat server terminates unexpectedly or the server goes offline, the chats that were hosted on that chat server are terminated at the client end. However the chats remain in the system and continue to route to Workspace Desktop Edition (previously Interaction Workspace (IWS)).

IWS presents the chat, but displays a message that it cannot connect to the chat server.

When the agent can eventually mark the chat as done, IWS closes the chat in the agent IWS session, but it routes again, landing on any available agent IWS. This can occur for up to an hour.

The following is an example log entry:

```
14-03-28 12:17:10.138 [ChannelDefault] WARN ESDK - Channel tcp://ctizgesmipr002:4125/ closed
14-03-28 12:17:10.139 [ChannelDefault] WARN ESDK - Channel closing [Name] ChatSessionChannel.
Id001_90f30c5c-63c4-438b-a1b7-72c2dd7a8fc9 [Uri] tcp://ctizgesmipr002:
4125/ has 1 requests pending. They are lost
14-03-28 12:17:10.139 [ChannelDefault] DEBUG ESDK - Chat strategy 'ResyncStrategy'
Processing msg [Name] undefined <check channel even> [EndPoint] tcp://ctizgesmipr002:4125/
14-03-28 12:17:10.141 [ChannelDefault] DEBUG ESDK - Chat strategy 'ResyncStrategy'
Found 1 related channel for Event undefined <check channel event>
14-03-28 12:17:10.147 [ChannelDefault] WARN edia.InteractionChat - Protocol Closed message:
'Genesyslab.Platform.Commons.Protocols.ProtocolException: Exception occurred
during channel opening --> System.Net.Sockets.SocketException: No connection could be made
because the target machine actively refused it 172.203.217.149:4125
at System.Net.Sockets.Socket.EndConnect(IAsyncResult asyncResult)
at Genesyslab.Platform.Commons.Connection.CommonConnection.AsyncConnect(IAsyncResult res)
- End of inner exception stack trace ---' Previous Channel State:'Opening,
[InteractionChat: Id001/0006Ka9HFHRR0T2B]'
```

Resolution

Use the workflow diagram to stop chat interactions from being routed to an agent. The workflow must determine if a chat session is still accessible before sending the interaction to an agent. One of the possible solutions could be to use "dummy" chat ESP messages. If such a message could not be delivered to Chat Server (see below details about errors), this will indicate not to route the interaction to the agent (and so the interaction could be stopped in workflow). The workflow could make several attempts (in the case where chat High Availability mode will be enabled) before finally stopping the attempt to route.

In order to hide these dummy messages from chat parties, the following could be done:

1. Use some special keywords in the message and modify the web chat application not to show those to the customer. Agent still will be seeing them.
2. Use the "Visibility" parameter of the chat ESP message request. It could have values ALL (default), INT (only agents) or VIP (only supervisors). Setting Visibility=VIP will hide the message from customer and agent (only supervisor will see those and it will be saved in final transcript in Universal Contact Server).

Errors (in the Interaction Server log) when trying to deliver chat ESP message are as follows:

If ChatServer is not running (was not restarted)

```
00:04:08.485 Trc 24112 Cannot find appropriate 3rd-party server: name: [any], type: ChatServer
00:04:08.485 Trc 24102 Sending to Universal Routing Server: URServer: 'EventError' (52)
message:
...
    AttributeErrorCode [int] = 1
    AttributeErrorMessage [str] = "Not found by type"
...
```

After ChatServer was restarted and running

```
00:18:32.532 Trc 26015 Received message 'ExternalServiceFault' ('502') from client
'ChatServer' -
Third party server:0:920, message attributes:
  attr_envelope [list, size (unpacked)=488] =
    'Parameters' [list] = (size=80)
      'FaultCode' [str] = "100"
      'FaultString' [str] = "interaction with specified id was not found"
    'Service' [str] = "Chat"
    'Method' [str] = "Message"
```