Composer Help

Twitter Block

5/9/2025

Contents

Twitter Block

Use the Twitter block to handle incoming Twitter interactions. Twitter handles can be monitored through pre-defined events and messages filtered based on keywords to determine the next action.

The Twitter block has the following properties:

Name Property

Find this property's details under Common Properties.

Block Notes Property

Find this property's details under Common Properties.

Exceptions Property

Find this property's details under Common Properties.

Condition Property

Find this property's details under Common Properties.

Logging Details Property

Find this property's details under Common Properties.

Log Level Property

Find this property's details under Common Properties.

Result Property

Click the down arrow under **Value** and select the variable to hold the raw data returned by the Social Messaging Server.

Application Property

Specify the Application Name of the Social Messaging Server.

1. Click under **Value** to display the **...** button.
2. Click the **...** button to open the Application Selection dialog box.
3. Click the down arrow opposite **Type** and select **Literal**, **Variable**, or **Configuration Server**.
4. If you selected *Configuration Server*, select the Social Messaging Server application from the drop-down of the **Value** field and click **OK**.
5. If you selected **Literal** or **Variable**, enter the literal or select a variable, and click **OK**.

Method Name Property

Select the method to be invoked. This is a mandatory field and the value can be a variable, literal, or method name enum (`SendMessage, DestroyMessage, GetUserInfo, RetweetTweet, FollowUser, UnfollowUser, GetFriends, GetFollowState, CreateFavorite, DestroyFavorite, GetMediaContent`).

[+] The input and output parameters for the different methods are listed here (mandatory parameters are in bold font).

Method Name	Description	Input Paramaters	Output Paramaters
MergeItxData	This ESP request is used to combine the attached data of two Twitter interactions related to the same post, update attached data of specified interaction, and return the final attached data subset that was added to the specified Twitter interaction. The necessity in such request appears when the routing business process detects that there is more than one interaction related to the same post (for example, a new comment(s) was added while the interaction was sitting in a Genesys interaction queue). In this	**_umsChannel**, **_twitterMergeItxId**	_twitterMergedUserData

Method Name	Description	Input Paramaters	Output Paramaters
	case, instead of delivering two interactions to an agent, the routing business process will try to merge them and deliver updated data to an agent.		
SendMessage	Request to send a Twitter message.	**_umsChannel**, **_twitterMsgType**, **_twitterMsgPlainText**, twitterInReplyToStatusId, _twitterToAddr, _twitterPictureName, _twitterPictureBody	_twitterMsgId
DestroyMessage	Request to destroy own status or direct message.	**_umsChannel**, **_twitterMsgType**, **_twitterMsgId**	_twitterMsgId, _twitterMsgPlainText
GetUserInfo	Request to get information about a Twitter user.	**_umsChannel**, **_twitterUserId**	'_twitterUserName, _twitterUserScreenName, _twitterUserDescr, _twitterUserCreatedAt, '_twitterUserImageUrl, _twitterUserLocation, _twitterUserTimeZone, _twitterUserFollowersCount, _twitterUserFriendsCount, _twitterUserStatusesCount, _twitterUserIsFollower, _twitterUserIsFollowee, _twitterPlaceFullName, _twitterIsFavorited, FirstName, LastName
RetweetTweet	Request to retweet a Twitter message.	**_umsChannel**, **_twitterMsgId**	_twitterMsgId, _twitterMsgPlainText
FollowUser	Request to follow another Twitter user.	**_umsChannel**, **_twitterUserId**	_twitterUserName, _twitterUserScreenName, _twitterUserDescr, _twitterUserCreatedAt, _twitterUserImageUrl, _twitterUserLocation, _twitterUserTimeZone, _twitterUserFollowersCount, _twitterUserFriendsCount, _twitterUserStatusesCount, _twitterUserIsFollower, _twitterUserIsFollowee, _twitterPlaceFullName, _twitterIsFavorited, FirstName, LastName
UnfollowUser	Request to unfollow another Twitter user.	**_umsChannel**, **_twitterUserId**	_twitterUserName, _twitterUserScreenName, _twitterUserDescr, _twitterUserCreatedAt,

Method Name	Description	Input Paramaters	Output Paramaters
			_twitterUserImageUrl, _twitterUserLocation, _twitterUserTimeZone, _twitterUserFollowersCount, _twitterUserFriendsCount, _twitterUserStatusesCount, _twitterUserIsFollower, _twitterUserIsFollowee, _twitterPlaceFullName, _twitterIsFavorited, FirstName, LastName
GetFriends	Request to get a list of friends of a Twitter user.	**_umsChannel**, **_twitterUserId**	_twitterUserIdList
GetFollowState	Request to get "a friend" and "a follower" states of a user.	**_umsChannel**, **_twitterUserId**	_twitterUserIsFriend, _twitterUserIsFollower
CreateFavorite	Request to set a Twitter (status) message as favorite.	**_umsChannel**, **_twitterMsgId**	_twitterMsgId, _twitterMsgPlainText
DestroyFavorite	Request to destroy a Twitter (status) message as favorite.	**_umsChannel**, **_twitterMsgId**	_twitterMsgId, _twitterMsgPlainText
GetMediaContent	Request to get content of a media attached to a direct Twitter message.	**_twitterMediaContentUrl**	_twitterMediaContentBin

Method Parameters Property

List of input parameters populated automatically based on the selected method in the **Method Name** property. The automatically populated values can be edited, if required. You can also add the new parameters by clicking the **ADD** button. The Method Parameters (key and value) can be specified either as a literal or variable.

Composer supports special keys such as, `TWITTER_CHANNEL (equivalent to _genesys.ixn.interactions[system.InteractionID].udata['_umsChannel']` in the code sample), which can be used to map values for the parameters.

> ### Tip
> Refer to the table in the **Method Name Property** section above for more information on custom parameters.

Service Timeout Property

Specify the timeout in seconds to invoke the ESP service on a timeout error. The default value is 10 seconds. You can specify the timeout value either as a literal or variable, but it must be a positive integer.

Enable Status Property

Find this property's details under Common Properties.

[+] Twitter Block Code Sample

```
<state id="RetweetMessage">
                <onentry>
                <log expr="_sessionid + ': Inside Twitter Block: RetweetMessage'" />
                   <script><![CDATA[
                               App_RetweetMessage.content = {};
                               App_RetweetMessage.content.params = {
                               _umsChannel:
_genesys.ixn.interactions[system.InteractionID].udata['_umsChannel'],
                               _twitterMsgId:
_genesys.ixn.interactions[system.InteractionID].udata['_twitterMsgId']

};
                ]]></script>
                <session:fetch requestid="App_RetweetMessage['requestid']"
                 srcexpr="'SocialMessagingServer' +'\\CFGSocialMS\\Twitter\\' +
'RetweetTweet'" attach_ixn_data="true" timeout="10" method="'esp'">
                <content _expr="App_RetweetMessage.content" />
                </session:fetch>
                </onentry>
                <transition event="session.fetch.done"
cond="_event.data.requestid==App_RetweetMessage['requestid']" target="$$_MY_PREFIX_$$.Exit1">
                <log expr="'Composer Application:default Block: RetweetMessage'" />
                <log expr="'Session FETCH DONE'" />
                <log expr="'Data Fetched:' + _event.data.content" />
                <script>storeEvent("RetweetMessage", _event);</script>
                <script>App_RetweetMessage['data'] = eval('(' + _event.data.content +
')');</script>
                 </transition>
</state>
```