



SYMPHONY

Connected System Plugin for **Appian**

Table of Contents

Overview	3
Features	3
Connected System Configuration	4
Steps for Obtaining Credentials	5
Create IM or MIM	5
Create Room	6-9
Deactivate or Reactivate Room	10-11
Add Member to Room	12-13
Remove Member from Room	13-14
List Streams of User	14-17
List Stream Members	17-19
Send Message	20-22
Get Message	22-24
Suppress Message	24-25
List Messages	25-28
Download Attachment	28-29

Overview

Securely communicate across your organization's preferred channels with encrypted, auditable messaging—all within Appian!

The use of non-compliant messaging platforms has led enterprises to face billions in regulatory fines. Symphony helps enable auditable, compliant messaging with clients via WhatsApp, WeChat and SMS. With seamless federation across prominent messaging and voice networks, Symphony delivers fast, compliance-enabled communication with clients within their preferred channels. As an industry leader, Symphony helps over a thousand financial institutions facilitate encrypted and compliant communication between clients while simultaneously eliminating internal communication silos.

With the Symphony Chat connected system, you can receive Symphony notifications, create Symphony chats and chat rooms, and quickly manage your conversations. Use this connected system alongside the Symphony component plug-in to embed an out-of-the-box, themable, and easy to use Symphony UI directly into your Appian workflow.

Features

- Get Message
- Send Message
- Suppress Message
- Create IM
- Create Room
- Deactivate Room
- Add Member
- Remove Member
- List User Streams

Connected System Configuration

Connected System Properties



Symphony

Connected System to use the Messages and Conversations API for sending messages and creating conversations in Symphony.

Version: 1

Name *

SCSDA CS Symphony

Description

Connected system

Symphony Configuration

User Name ? *

devcertbot2780

Symphony Pod domain ? *

https://develop2.symphony.com

RSA Private Key ? *

.....

Connection successful

TEST CONNECTION

CANCEL

SAVE

The Symphony Connected System requires the following credentials: Username, Symphony Pod Domain and RSA Private Key.

Steps for obtaining credentials

- Follow the Step 1 and Step 2 mentioned in the following URL:
<https://docs.developers.symphony.com/building-bots-on-symphony/authentication/rsa-a-uthentication>
- Obtain the Private Key file from Step 2 in the above process.
- Provide the Username, Symphony Pod domain and the entire contents of the private key file in the Connected System configuration respectively.
- Click on Test Connection to make sure the connection is successful. If not, please revisit the steps and make sure to provide valid credentials.

Conversation Integrations

- **Create IM or MIM**

Creates a new single or multi-party instant message conversation or returns an existing IM or MIM between the specified users and the calling user.

If an user ID with the existing conversation is provided, it returns the Stream ID of the existing conversation.

The screenshot shows the configuration and test results for the 'Create IM or MIM' operation. On the left, the configuration panel includes:

- Connected System:** SCSDA CS Symphony
- Operation:** Conversation Integrations (Modifies Data)
- Choose Integrations:** Create IM or MIM
- User IDs:** A list containing 'riuserIds'

At the bottom of the configuration panel is a 'TEST REQUEST' button. On the right, the test results are displayed:

Rule Input Name	Expression	Value
userIds (List of Text String)	1 "349026222359920"	List of Text String: 1 item "349026222359920"

Below the table, the test results show a 'Success!' message, a 'Time' of 220 ms, and a 'Value: Result' containing a dictionary with an 'id' field: 'wLw4ywEhr4IOvmXbDPK-Hn__nmlGDjzdA' (Text).

Input:

User IDs (List of Text) - Required

Description: Provide a single user ID to create an IM or list of user IDs to create a MIM along with the calling user.

Output: Dictionary

Unset

```
{  
  id: Stream ID  
}
```

• Create Room

Creates a new chat room with the given name, description and the room configuration.

Connected System *
SCSDA CS Symphony x

Operation *
Conversation Integrations (Modifies Data) v
Integrations to create IM, Rooms and get members.

Choose Integrations *
Create Room v

Name
rllName
Provide a name for the room.

Description
rllDescription
Provide a description for the room.

Can Members Invite
rllisMemberInvite
If true, any chat room participant can add new participants. If false, only owners can add new participants. Default : false.

Discoverable
rllisDiscoverable
If true, non-participants can search for this room. If false, only participants can search for this room. Default : false.

Public
rllisPublic

Rule Input Name	Expression	Value
name (Text)	1 "Test"	"Test"
Description (Text)	1 "test"	"test"
isMemberInvite (Boolean)	1	<input type="radio"/> True <input type="radio"/> False
isDiscoverable (Boolean)	1	<input type="radio"/> True <input type="radio"/> False
isPublic (Boolean)	1	<input type="radio"/> True <input type="radio"/> False
isReadOnly (Boolean)	1	<input type="radio"/> True <input type="radio"/> False
isEnabledProtection (Boolean)	1	<input type="radio"/> True <input type="radio"/> False
isCrossPod (Boolean)	1	<input type="radio"/> True <input type="radio"/> False

Provide a description for the room.

Can Members Invite

[rllsMemberInvite](#)

If true, any chat room participant can add new participants. If false, only owners can add new participants. Default : false.

Discoverable

[rllsDiscoverable](#)

If true, non-participants can search for this room. If false, only participants can search for this room. Default : false.

Public

[rllsPublic](#)

If true, anyone can join this room. If false, members must be added to this room. Default : false.

Read Only

[rllsReadOnly](#)

If true, only room owners can send messages. Default : false.

Enable Copy Protection

[rllsEnableProtection](#)

If true, users cannot copy content from this room. Default : false.

Cross Pod

[rllsCrossPod](#)

If true, this room is a cross pod room. Default : false.

Enable History

[rllsEnableHistory](#)

If true, new members can view the room chat history of the room. Default: false.

isCrossPod (Boolean) 1 True False

isEnableHistory (Boolean) 1 True False

Set as default test values

Success!

Time
335 ms
Prepare: < 1 ms - Execute: 335 ms (Send/Wait/Receive: N/A) - Transform: < 1 ms

Value: Result

- Dictionary
 - roomSystemInfo Dictionary
 - createdByUserId 3.490262e+14 (Number (Decimal))
 - active true (Boolean)
 - id "FB7wHWV5G_Juzg9KXMyDH__njmCsmDA" (Text)
 - creationDate 1.677661e+12 (Number (Decimal))
 - roomAttributes Dictionary
 - viewHistory false (Boolean)
 - keywords List of Variant - 0 items
 - public false (Boolean)
 - discoverable false (Boolean)
 - name "Test" (Text)
 - description "test" (Text)
 - readOnly false (Boolean)
 - copyProtected false (Boolean)
 - crossPod false (Boolean)
 - multiLateralRoom false (Boolean)
 - membersCanInvite false (Boolean)

TEST REQUEST

Inputs

Name (Text) - Required

Description: Provide a name for the room.

Description (Text) - Required

Description: Provide a description for the room.

Can Members Invite (Boolean) - Optional

Description: If true, any chat room participant can add new participants. If false, only owners can add new participants. Default : false.

Discoverable (Boolean) - Optional

Description: If true, non-participants can search for this room. If false, only participants can search for this room. Default : false.

Public (Boolean) - Optional

Description: If true, anyone can join this room. If false, members must be added to this room. Default : false.

Read Only (Boolean) - Optional

Description: If true, only room owners can send messages. Default : false.

Enable Copy Protection (Boolean) - Optional

Description: If true, users cannot copy content from this room. Default : false.

Cross Pod (Boolean) - Optional

Description: If true, this room is a cross pod room. Default : false.

Enable Copy Protection (Boolean) - Optional

Description: If true, new members can view the room chat history of the room. Default: false.

Enable History (Boolean) - Optional

Description: If true, new members can view the room chat history of the room. Default: false.

Output: Dictionary

Unset

```
{
  roomSystemInfo (Dictionary)
  {
    createdByUserId : 3.490262e+14(Number (Decimal))
    active : true(Boolean)
    id : "fB7wHWv5G__juzg9KXMyDH___nljmCsmdA"(Text)
    creationDate : 1.677661e+12(Number (Decimal))
  }

  roomAttributes (Dictionary) {
    viewHistory : false (Boolean)
    keywords : List of Variant - 0 items
    public : false(Boolean)
    discoverable : false(Boolean)
    name : "Test"(Text)
    description : "test"(Text)
    readOnly : false(Boolean)
    copyProtected : false(Boolean)
    crossPod : false(Boolean)
    multiLateralRoom : false(Boolean)
    membersCanInvite : false(Boolean)
  }
}
```

• Deactivate or Reactivate Room

To Deactivate or reactivate an existing chatroom. At creation, a new chat room is active by default.

The screenshot shows a rule configuration interface. On the left, there are settings for 'Connected System' (SCSDA CS Symphony), 'Operation' (Conversation Integrations), and 'Choose Integrations' (Deactivate or Reactivate Room). Below these are fields for 'Stream ID' (riStreamId) and 'Active' (riIsActive). A 'TEST REQUEST' button is at the bottom.

Rule Input Name	Expression	Value
streamId (Text)	1 "JaGtLh-uD45CTeEcDQ4QdX__nmCOoVdA"	"JaGtLh-uD45CTeEcDQ4QdX__nmCOoVdA"
isActive (Boolean)	1	<input type="radio"/> True <input checked="" type="radio"/> False

Set as default test values

Success!

Time
295 ms
Prepare: 1 ms - Execute: 294 ms (Send/Wait/Receive: N/A) - Transform: < 1 ms

Value: Result

- Dictionary
 - roomSystemInfo Dictionary
 - createdByUserId 3.490262e+14 (Number (Decimal))
 - active false (Boolean)
 - id "JaGtLh-uD45CTeEcDQ4QdX__nmCOoVdA" (Text)
 - creationDate 1.677147e+12 (Number (Decimal))
 - immutableRoomAttributes Dictionary
 - public false (Boolean)
 - readOnly false (Boolean)
 - copyProtected false (Boolean)
 - roomAttributes Dictionary
 - discoverable false (Boolean)
 - name "Sample Room" (Text)
 - description "Sample Room description" (Text)
 - membersCanInvite false (Boolean)

Inputs

Stream ID (Text) - Required

Description: Provide the Room Stream ID to deactivate or reactivate.

Note: The stream ID must be a type of Room.

Active (Boolean) - Required

Description: If set to false, deactivates an active room. If set to true reactivates a deactivated room.

Output - Dictionary

Unset

```
{
  roomSystemInfo (Dictionary) {
    createdByUserId : 3.490262e+14(Number (Decimal))
    active : false(Boolean)
    id : "WEUbahiyIrbZYadKVxFhJn___nmIBBSbdA"(Text)
    creationDate : 1.67705e+12(Number (Decimal))
  }
  immutableRoomAttributes (Dictionary) {
    public : false(Boolean)
    readOnly : false(Boolean)
    copyProtected : false(Boolean)
  }
  roomAttributes (Dictionary) {
    discoverable : false(Boolean)
    name : "Sample Test Room"(Text)
    description : "Sample room with Naveen and DevCertBot4780"(Text)
    membersCanInvite : false(Boolean)
  }
}
```

• Add Member to Room

Adds a new member to an existing room.

The screenshot shows the configuration for the 'Add Member to Room' integration. On the left, the 'Connected System' is 'SCSDA CS Symphony'. The 'Operation' is 'Conversation Integrations (Modifies Data)'. The 'Choose Integrations' dropdown is set to 'Add Member to Room'. The 'Stream ID' is 'ri:streamid' and the 'User ID' is 'ri:userId'. A 'TEST REQUEST' button is at the bottom.

The right side shows the 'Expression' tab with a table of rule inputs:

Rule Input Name	Expression	Value
userId (Text)	1 "349026222359920"	"349026222359920"
streamId (Text)	1 "IfIdIHsslJyDMqKC-kh3Mn__nitVeCIdA"	"IfIdIHsslJyDMqKC-Kh3Mn__nitVeCIdA"

Below the table is a 'Set as default test values' link. The 'Result' tab shows a 'Success!' message. The 'Time' section shows '294 ms' and 'Prepare: < 1 ms - Execute: 293 ms (Send/Wait/Receive: N/A) - Transform: 1 ms'. The 'Value: Result' section shows a dictionary with 'format "TEXT" (Text)' and 'message "Member added" (Text)'.

Inputs

Stream ID (Text) - Required

Description: Provide the Room Stream ID to add a member.

Note: The stream ID must be a type of Room.

User ID (Text) - Required

Description: Provide the ID of the user to be added to the room.

Output - Dictionary

Unset

```
{
```

```
  format : "TEXT"(Text)
```

```
  message : "Member added"(Text)
```

```
}
```

● Remove Member from Room

Removes an existing member from an existing room.

The screenshot displays a REST client interface with the following configuration and execution details:

- Connected System:** SCSDA CS Symphony
- Operation:** Conversation Integrations (Modifies Data)
- Choose Integrations:** Remove member from Room
- Stream ID:** r1streamid
- User ID:** r1userid

Rule Input Name	Expression	Value
userId (Text)	1 "349026222359920"	"349026222359920"
streamId (Text)	1 "1fId1Hss1JyDMqKC-Kh3Mn__n1tVeC1dA"	"1fId1Hss1JyDMqKC-Kh3Mn__n1tVeC1dA"

[Set as default test values](#)

Result | Request | Response

Success!

Time: 264 ms
Prepare: < 1 ms - Execute: 264 ms (Send/Wait/Receive: N/A) - Transform: < 1 ms

Value: Result

- Dictionary
 - format "TEXT" (Text)
 - message "Member removed" (Text)

TEST REQUEST

Inputs

Stream ID (Text) - Required

Description: Provide the Room Stream ID to remove a member.

Note: The stream ID must be a type of Room.

User ID (Text) - Required

Description: Provide the ID of the user to be removed from the room.

Output - Dictionary

Unset

{

format : "TEXT"(Text)

message : "Member removed"(Text)

}

• List Streams of User

Returns a list of all the streams of which the requesting user is a member.

The screenshot shows a configuration interface for a rule named "List Streams of User". On the left, there is a sidebar with the following sections:

- Connected System ***: SCSDA CS Symphony x
- Operation ***: Conversation Integrations (Reads Data)
- Choose Integrations ***: List Streams of User
- Stream Types**: A list editor containing "r1liststreams".

The main area displays a configuration table with the following columns: Rule Input Name, Expression, and Value.

Rule Input Name	Expression	Value
streams (List of Text String)	1 { 2 "IM", "MIM", "ROOM", "POST" 3 }	List of Text String: 4 Items "IM" "MIM" "ROOM" "POST"
includeInactiveStreams (Boolean)	1	<input type="radio"/> True <input type="radio"/> False
skip (Number (Integer))	1	<input type="text"/>
limit (Number (Integer))	1	<input type="text"/>

Below the table is a link: [Set as default test values](#)

At the bottom of the sidebar, there is a note: "A list of stream types that will be returned. Options are IM (1-1 instant messages), MIM (multipart instant messages), ROOM (rooms), POST (the user's wall). For more information, see the streamTypes section below. If not specified, all types of streams are returned."

streams (List of Text String)	1 { 2 "IM", "MIM", "ROOM", "POST" 3 }	"IM" "MIM" "ROOM" "POST"
includeInactiveStreams (Boolean)	1	<input type="radio"/> True <input type="radio"/> False
skip (Number (Integer))	1	

Place cursor on function, rule, or constant to display help

A list of stream types that will be returned. Options are IM (1-1 instant messages), MIM (multiparty instant messages), ROOM (rooms), POST (the user's wall). If not specified, all types of streams are returned.

Include Inactive Streams
[riIncludeInactiveStreams](#)
 Whether to include inactive conversations. A chatroom is inactive if it has been deactivated by an owner or admin. An IM or MIM is inactive if one of the participating users has been deactivated by an admin. If not specified, only active streams are returned. Default: false.

Skip
[riSkip](#)
 Number of streams to skip.

Limit
[riLimit](#)
 Maximum number of streams to return. If not specified, the default limit will be 50.

TEST REQUEST

Time
 228 ms
 Prepare: < 1 ms - Execute: 228 ms (Send/Wait/Receive: N/A) - Transform: < 1 ms

Value

- Dictionary
 - success **true** (Boolean)
 - result Dictionary
 - streams List of Dictionary - 7 items
 - Dictionary
 - streamAttributes Dictionary
 - members List of Number (Floating Point) - 1 item
3.490262e+14 (Number (Decimal))
 - streamType Dictionary
 - type "POST" (Text)
 - active **true** (Boolean)
 - id "IV-j3bQ5725Z2jm3dxSRAH__nmlHmT1dA" (Text)
 - crossPod **false** (Boolean)
 - Dictionary
 - streamAttributes Dictionary
 - members List of Number (Floating Point) - 2 items
3.490262e+14 (Number (Decimal))
3.490262e+14 (Number (Decimal))
 - streamType Dictionary
 - type "IM" (Text)
 - active **true** (Boolean)
 - id "wLw4ywEhr4IOVmXbDPK-Hn__nmlGDjzdA" (Text)

Inputs

Stream Types (List of Text) - Optional

Description: A list of stream types that will be returned. Options are :

- IM (1-1 instant messages),
- MIM (multiparty instant messages),
- ROOM (rooms),
- POST (the user's wall).

If not specified, all types of streams are returned.

Include Inactive Streams (Boolean) - Optional

Description : Whether to include inactive conversations. A chatroom is inactive if it has been deactivated by an owner or admin. An IM or MIM is inactive if one of the participating users has been deactivated by an admin. If not specified, only active streams are returned. Default: false.

Skip (Number (Integer)) - Optional

Description: Number of streams to skip.

Limit (Number (Integer)) - Optional

Description: Maximum number of streams to return. If not specified, the default limit will be 50.

Output - Dictionary

Unset

```
{
```

```
  success : true(Boolean)
  result: { (Dictionary)
    streams :{ (List of Dictionary )- 3 items
      {
        streamAttributes :{ (Dictionary)
          members : { (List of Number (Floating Point)) - 1
            item
              3.490262e+14(Number (Decimal))
            }
          }
        streamType : { (Dictionary)
          type : "POST"(Text)
        active : true(Boolean)
        id : "IV-J3bQS72SZ2Jm3dxSRAH___nmIHmT1dA"(Text)
        crossPod : false(Boolean)
      }
    }
    {
      streamAttributes : { (Dictionary)
        members : { (List of Number (Floating Point) ) - 2
          items
            3.490262e+14(Number (Decimal))
            3.490262e+14(Number (Decimal))
          }
        }
      streamType : { (Dictionary)
        type : "IM"(Text)
      }
    }
  }
}
```



```
        active : true(Boolean)
        id : "wLw4ywEhr4IOVmXbDPK-Hn___nmIGDjzdA"(Text)
        crossPod : false(Boolean)
    }
}
{
    streamType : { (Dictionary)
        type : "ROOM"(Text)
    }
    active : true(Boolean)
    id : "JaGtLh-uD45CTEecDQ4QdX___nmCO0oVdA"(Text)
    crossPod : false(Boolean)
}
roomAttributes : { (Dictionary)
    name : "Sample Room"(Text)
}
}
}
```

• List Stream Members

Returns a list of all the current members of a stream (IM, MIM, or chatroom).

The screenshot shows the configuration for the 'List Stream Members' operation. On the left, the 'Connected System' is 'SCSDA CS Symphony'. The 'Operation' is 'Conversation Integrations (Reads Data)'. The 'Choose Integrations' dropdown is set to 'List Stream Members'. The 'Stream ID' is 'r1streamid'. A 'TEST REQUEST' button is at the bottom.

Rule Input Name	Expression	Value
streamid (Text)	1 "wLw4ywEhr4IOVmxXbDPK-Hn__nmIGDjzdA"	"wLw4ywEhr4IOVmxXbDPK-Hn__nmIGDjzdA"

Set as default test values

Result: Request Response

Success!

Time: 213 ms
Prepare: < 1 ms - Execute: 213 ms (Send/Wait/Receive N/A) - Transform: < 1 ms

Value

- Dictionary
 - success: true (Boolean)
- result: Dictionary
 - members: List of Dictionary - 2 items
 - Dictionary
 - owner: false (Boolean)
 - joinDate: 1.677049e+12 (Number (Decimal))
 - id: 3.490262e+14 (Number (Decimal))
 - Dictionary
 - owner: false (Boolean)
 - joinDate: 1.677049e+12 (Number (Decimal))
 - id: 3.490262e+14 (Number (Decimal))
 - error: null (Null)
 - authType: Diagnostic

Inputs

Stream ID (Text) - Required

Description : Provide the Stream ID of the conversation to get its members.

Output: (Dictionary)

Unset

```
{
```

```
  success : true(Boolean)
```

```
result : { (Dictionary)
```

```
  members : { (List of Dictionary) - 2 items
```

```
    {
```

```
      owner : false(Boolean)
```

```
      joinDate : 1.677049e+12(Number (Decimal))
```

```
      id : 3.490262e+14(Number (Decimal))
```

```
    }
```

```
    {
      owner : false(Boolean)
      joinDate : 1.677049e+12(Number (Decimal))
      id : 3.490262e+14(Number (Decimal))
    }
  }
}
```

Message Integrations

• Send Message

Enables you to post a new message to a given list of streams (each stream can either be a chatroom, an IM or a multiparty IM).

The screenshot shows the configuration and execution of the 'Send Message' integration. On the left, the configuration panel includes:

- Connected System:** SCSDA CS Symphony
- Operation:** Message Integrations (Modifies Data)
- Choose Integrations:** Send Message
- Stream IDs:** A list containing 'r:1SID'.
- Message:** 'rimessage'
- Attachment:** 'rlattachment'

The main configuration table shows the following inputs and expressions:

Rule Input Name	Expression	Value
SID (List of Text String)	1 "wLw4ywEhr4IOVmxBDPK-Hn_nmlGDjzdA"	List of Text String: 1 Item "wLw4ywEhr4IOVmxBDPK-Hn_nmlGDjzdA"
message (Text)	1 "Hello World!"	"Hello World!"
attachment (Document)	1	spiderman

The execution results show a successful outcome with the following details:

- Time:** 2,731 ms
- Value: Result:**
 - success: true
 - messages: List of Dictionary - 1 Item
 - attachments: List of Dictionary - 1 Item
 - images: List of Dictionary - 1 Item
 - stream: Dictionary
 - streamType: "IM"
 - streamId: "wLw4ywEhr4IOVmxBDPK-Hn_nmlGDjzdA"
 - messageId: "pCzB2tss0Y3NCoyu9t2lS3_nlen75lBQ"
 - userAgent: "Agent-22.12.2-658-Linux-4.14.275-207.503.amzn2.x86_64"
 - originalFormat: "com.symphony.messageml.v2"

Inputs

Stream IDs (List of Text) - Required

Description : Provide the list of Stream IDs (conversation IDs of the IM, MIM or room).

Message (Text) - Required

Description : Provide the message content to be sent.

Attachment (Document) - Optional

Description: Provide the document to send as attachment. Supported file types :
"bmp", "doc", "png", "mpeg", "cdi", "jpg", "mpg", "xls", "pdf", "avi", "csv", "pptx", "tiff", "x-tiff", "wbmp", "docx", "tif", "ppt", "gif", "mov", "jpeg", "xlsx", "m4v", "mp4".

Note : File size limit is 30 MB.

Output (Dictionary)

Unset

```
{
  success : true(Boolean)
  messages : { (List of Dictionary) - 1 item
    {
      attachments : { (List of Dictionary )- 1 item
        {
          images : (List of Dictionary )- 1 item
            {
              id :
                "external_349026222359922%2Fd7clKAes4
                hi4Ca5QA%2BJduA%3D%3D"(Text)
              dimension : "600"(Text)
            }
          size : 49077(Number (Integer))
          name : "spiderman.png"(Text)
          id :
            "external_349026222359922%2Fd65KWUwayE%2BEE%2B
            Ni6Y95Tg%3D%3D"(Text)
          }
        }
      }
    }
  stream : { (Dictionary)
    streamType : "IM"(Text)
    streamId : "wLw4ywEhr4IOVmXbDPK-Hn___nmIGDjzdA"(Text)
  }
  messageId : "mi699vKczWsuagoZEryjln___nljGF7UbQ"(Text)
  userAgent :
    "Agent-22.12.2-658-Linux-4.14.275-207.503.amzn2.x86_64"(Text)
  originalFormat : "com.symphony.messageml.v2"(Text)
  message : "<div data-format='PresentationML' data-version='2.0'>Hello
  World!</div>"(Text)
  user : { (Dictionary)
    displayName : "Dev Cert Bot 2780"(Text)
    userId : 3.490262e+14(Number (Decimal))
  }
}
```

```

    email : "devcertbot2780@bots.symphony.com"(Text)
    username : "devcertbot2780"(Text)
  }
  timestamp : 1.67767e+12(Number (Decimal))
  sid : "19b59c91-0c94-4275-a483-f5ab5b369e7d"(Text)
}
}
}

```

• Get Message

Returns details about the specified message with the given message ID.

Connected System *

SCSDA CS Symphony ✕

Operation *

Message Integrations (Reads Data) ▼

Integrations to send, get and list messages of conversations.

Choose Integrations *

Get Message ▼

Message ID

rlmessageid

Provide the ID of the message to get.

TEST REQUEST

Rule Input Name	Expression	Value
messageid (Text)	1 "JazoGpRmMGN9D3dosEksCX__n1oPr0obQ"	"JazoGpRmMGN9D3dosEksCX__n1oPr0obQ"

[Set as default test values](#)

534 ms

Prepare: < 1 ms - Execute: 534 ms (Send/Wait/Receive: N/A) - Transform: < 1 ms

Value

- ▼ Dictionary
 - success **true** (Boolean)
 - ▼ result Dictionary
 - ▼ attachments List of Dictionary - 1 item
 - ▼ Dictionary
 - ▼ images List of Dictionary - 1 item
 - ▼ Dictionary
 - id "external_349026222359922%2FD7QPe8XWG3oBBVCOSRhV3w%3D%3D" (Text)
 - dimension "600" (Text)
 - size 49077 (Number (Integer))
 - name "spiderman.png" (Text)
 - id "external_349026222359922%2FQtFY%2F55dMQHq9m%2F1hxDfG%3D%3D" (Text)
- ▼ stream Dictionary
 - streamType "IM" (Text)
 - streamId "wLw4ywEhr4IOVmXbDPK-Hn__nmIGDjzdA" (Text)
 - messageid "JazoGpRmMGN9D3dosEksCX__n1oPr0obQ" (Text)
 - userAgent "Agent-22.12.2-658-Linux-4.14.275-207.503.amzn2.x86_64" (Text)
 - originalFormat "com.symphony.messageml.v2" (Text)
 - message "<div data-format='PresentationML' data-version='2.0'>Hello world</div>" (Text)
- ▼ user Dictionary
 - displayName "Dev Cert Bot 2780" (Text)
 - userId 3.490262e+14 (Number (Decimal))
 - email "devcertbot2780@bots.symphony.com" (Text)
 - username "devcertbot2780" (Text)
 - timestamp 1.677583e+12 (Number (Decimal))

Input

Message ID (Text) - Required

Description : Provide the ID of the message to get.

Output - Dictionary

Unset

```
{
  success : true(Boolean)
  messages : { (List of Dictionary) - 1 item
    {
      attachments : { (List of Dictionary) - 1 item
        {images : (List of Dictionary) - 1 item
          {
            id :
              "external_349026222359922%2Fd7cKAEs4
              hi4Ca5QA%2BJduA%3D%3D"(Text)
            dimension : "600"(Text)
          }
          size : 49077(Number (Integer))
          name : "spiderman.png"(Text)
          id :
            "external_349026222359922%2Fd65KWUwayE%2BEE%2B
            Ni6Y95Tg%3D%3D"(Text)
          }
        }
      }
      stream : { (Dictionary)
        streamType : "IM"(Text)
        streamId : "wLw4ywEhr4IOVmXbDPK-Hn___nmIGDjzdA"(Text)
      }
      messageId : "mi699vKczWsuagoZEryjln___nljGF7UbQ"(Text)
      userAgent :
        "Agent-22.12.2-658-Linux-4.14.275-207.503.amzn2.x86_64"(Text)
      originalFormat : "com.symphony.messageml.v2"(Text)
      message : "<div data-format='PresentationML' data-version='2.0'>Hello
      World!</div>"(Text)
      user : { (Dictionary)
        displayName : "Dev Cert Bot 2780"(Text)
        userId : 3.490262e+14(Number (Decimal))
        email : "devcertbot2780@bots.symphony.com"(Text)
        username : "devcertbot2780"(Text)
      }
    }
  }
}
```

```

timestamp : 1.67767e+12(Number (Decimal))
sid : "19b59c91-0c94-4275-a483-f5ab5b369e7d"(Text)
}
}}

```

• Suppress Message

Suppress (Remove) a message from the conversation, preventing its contents from being displayed to users.

Rule Input Name	Expression	Value
messageId (Text)	1 "JazoGpRmMGN9D3dosEksCX__n!oPro0bQ"	"JazoGpRmMGN9D3dosEksCX__n!oPro0bQ"

[Set as default test values](#)

Result Request Response

Success!

Time
337 ms
Prepare: 1 ms - Execute: 336 ms (Send/Wait/Receive: N/A) - Transform: < 1 ms

Value: Result

- Dictionary
 - suppressionDate 0 (Number (Integer))
 - messageId "JazoGpRmMGN9D3dosEksCX__n!oPro0bQ" (Text)
 - suppressed true (Boolean)

TEST REQUEST

Input

Message ID (Text) - Required

Description : Provide the ID of the message to suppress (remove).

Output - Dictionary

Unset

```
{
```

```
suppressionDate : 0(Number (Integer))  
messageId : "JazoGpRmMGN9D3dosEksCX___nloPRo0bQ"(Text)  
suppressed : true(Boolean)  
}
```

• List Messages

List messages from an existing stream (IM, MIM, chatroom, wall post). Additionally returns any attachments associated with the message.

Rule Input Name	Expression	Value
SID (Text)	1 "wLw4ywEhr4IOvmXbDPK-Hn___nmiGDjzda"	"wLw4ywEhr4IOvmXbDPK-Hn___nmiGDjzda"
skip (Number (Integer))	1	

Time
1,386 ms
Prepare: < 1 ms - Execute: 1,386 ms (Send) / Wait: N/A - Receive: N/A - Transform: < 1 ms

Value

- Dictionary
 success true (Boolean)
- result Dictionary
 messages List of Dictionary - 50 items
 Dictionary
 stream Dictionary
 streamType "IM" (Text)
 streamid "wLw4ywEhr4IOvmXbDPK-Hn___nmiGDjzda" (Text)
 messageId "qRwormjPJL14hIAbw1Zan___njMV2rbQ" (Text)
 userAgent "Agent-22.12.2-658-Linux-4.14.275-207.503.amzn2.x86_64" (Text)
 originalFormat "com.symphony.messageml.v2" (Text)
 message "<div data-format='PresentationML' data-version='2.0'>Sent</div>" (Text)
- user Dictionary
 displayName "Dev Cert Bot 2780" (Text)
 userid 3.490262e+14 (Number (Decimal))
 email "devcertbot2780@bots.symphony.com" (Text)
 username "devcertbot2780" (Text)
 timestamp 1.677668e+12 (Number (Decimal))
 sid "7180e1fe-c14a-49c2-8f56-a5e549d6b526" (Text)
- Dictionary
 stream Dictionary
 streamType "IM" (Text)
 streamid "wLw4ywEhr4IOvmXbDPK-Hn___nmiGDjzda" (Text)

Inputs

Stream ID (Text) - Required

Description : Provide the Stream ID (conversation ID of IM, MIM or room) to get the messages.

Since (Text) - Optional

Description: Timestamp of the earliest possible date of the first message returned. A long integer value representing milliseconds since January 1, 1970.

Limit (Number (Integer)) - Optional

Description: Maximum number of messages to return. If not specified, the default limit will be 50.

Output - Dictionary

Unset

```
{
  success : true(Boolean)
  messages : { (List of Dictionary) - 1 item
    {
      attachments : { (List of Dictionary )- 1 item
        {
          images : (List of Dictionary )- 1 item
            {
              id :
                "external_349026222359922%2Fd7clKAes4
                hi4Ca5QA%2BJduA%3D%3D"(Text)
              dimension : "600"(Text)
            }
          size : 49077(Number (Integer))
          name : "spiderman.png"(Text)
          id :
            "external_349026222359922%2Fd65KWUwayE%2BEE%2B
            Ni6Y95Tg%3D%3D"(Text)
          }
        }
      }
    }
  stream : { (Dictionary)
    streamType : "IM"(Text)
    streamId : "wLw4ywEhr4IOVmXbDPK-Hn___nmIGDjzdA"(Text)
  }
}
```

```
    }
    messageId : "mi699vKczWsuagoZEryjln___nljGF7UbQ"(Text)
    userAgent :
    "Agent-22.12.2-658-Linux-4.14.275-207.503.amzn2.x86_64"(Text)
    originalFormat : "com.symphony.messageml.v2"(Text)
    message : "<div data-format='PresentationML' data-version='2.0'>Hello
    World!</div>"(Text)
    user : { (Dictionary)
        displayName : "Dev Cert Bot 2780"(Text)
        userId : 3.490262e+14(Number (Decimal))
        email : "devcertbot2780@bots.symphony.com"(Text)
        username : "devcertbot2780"(Text)
    }
    timestamp : 1.67767e+12(Number (Decimal))
    sid : "19b59c91-0c94-4275-a483-f5ab5b369e7d"(Text)
},
{
    stream : { (Dictionary)
        streamType : "IM"(Text)
        streamId : "wLw4ywEhr4IOVmXbDPK-Hn___nmIGDjzdA"(Text)
    }
    messageId : "-qRwormjPjL14hlAbw1Zan___nljMV2rbQ"(Text)
    userAgent: "Agent-22.12.2-658-Linux-4.14.275-207.503.amzn2.x86_64"(Text)
    originalFormat : "com.symphony.messageml.v2"(Text)
    message : "<div data-format='PresentationML' data-version='2.0'>Sent</div>"(Text)
    user : { (Dictionary)
        displayName : "Dev Cert Bot 2780"(Text)
        userId : 3.490262e+14(Number (Decimal))
        email : "devcertbot2780@bots.symphony.com"(Text)
        username : "devcertbot2780"(Text)
    }
    timestamp : 1.677668e+12(Number (Decimal))
    sid : "7180e1fe-c14a-49c2-8f56-a5e549d6b526"(Text)
}
}
}
```

• Download Attachment

Downloads the attachment body and saves in the Appian folder by the attachment ID, stream ID, and message ID.

Rule Input Name	Expression	Value
SID (Text)	"wLw4ywEhr4IOvmxbDPK-Hn__nmIGDjzdA"	"wLw4ywEhr4IOvmxbDPK-Hn__nmIGDjzdA"
messageId (Text)	"qyvffmK/f7nUHSVP46q/D3///nmIDYCVbQ=="	"qyvffmK/f7nUHSVP46q/D3///nmIDYCVbQ=="
fileId (Text)	"internal_349026222359922%2FGS Bx11DagLphPmW1%2BIO%2FA%3D%3D"	"internal_349026222359922%2FGS Bx11DagLphPmW1%2BIO%2FA%3D%3D"
contentType (Text)	"image/png"	"image/png"
outputFileName (Text)	"TEST"	"TEST"

Result | Request | Response

Success!

Time
946 ms
Prepare: < 1 ms - Execute: 945 ms (Send/Wait/Receive: N/A) - Transform: 1 ms

Value: Result

- Dictionary
 - success true (Boolean)
 - document 698096 - TEST.png (Document)

Inputs

Stream ID (Text) - Required

Description : Provide the Stream ID (conversation ID of IM, MIM or room) of the attachment.

Message ID (Text) - Required

Description : Provide the ID of the associated message of the attachment.

File ID (Text) - Required

Description: Provide the File ID of the attachment to download.

File Name (Text) - Required

Description: Provide the name of the attachment with extension. Example :
image.png.

Save to Folder (Folder) - Required

Description: Provide the target save to folder where the output file needs to be saved.

Output - Dictionary

Unset

```
{  
  success: true,  
  document: documentId - Output file name  
}
```