



# Microsoft Outlook Adapter for SAP Integration Suite

Version 1.0.0 – February 2026

# Contents

1	Introduction.....	3
1.1	Objective.....	3
1.2	Coding Samples .....	3
1.3	Internet Hyperlinks .....	3
1.4	Overview.....	3
1.5	Features.....	4
1.6	Quick Start.....	4
2	Installation and Configuration.....	6
2.1	Prerequisites .....	6
2.2	Procedure.....	6
2.2.1	Adapter Installation by creating a New Integration Flow.....	6
2.2.2	Adapter Installation without creating a New Integration Flow .....	7
3	Getting Started: Microsoft Outlook Adapter .....	9
3.1	Architecture Overview .....	9
3.1.1	Sender Adapter.....	9
3.1.2	Receiver Adapter .....	9
3.2	Application Configuration.....	10
3.2.1	Account Creation .....	10
3.2.2	App Setup .....	10
3.3	Authentication.....	10
3.3.1	Creating OAuth2 Client Credentials for Application Permission .....	10
3.3.2	OAuth2 Authorization Code for Delegated permission.....	13
4	Microsoft Outlook Adapter Configuration.....	17
4.1	Sender Adapter .....	17
4.1.1	Connection.....	17
4.1.2	Processing .....	18
4.2	Receiver Adapter .....	20
4.2.1	Connection .....	20
4.2.2	Processing.....	21

5	Microsoft Outlook Supported Operations .....	25
5.1	Sender Adapter .....	25
5.1.1	Read Mail Message .....	25
5.2	Receiver Adapter .....	27
5.2.1	Create draft message.....	27
5.2.2	Send Message.....	28
5.2.3	Get Message.....	30
5.2.4	Add message attachment .....	31
5.2.5	Create Mail Folder.....	32
5.2.6	List Calendar Events.....	33
5.2.7	Get Next Results Page.....	34
5.2.8	Find Meeting Times.....	35
5.2.9	Get Message Delta.....	36
5.2.10	Add event attachment.....	37
6	Support.....	40
6.1	Troubleshooting .....	40
7	Appendix.....	42

# 1 Introduction

## 1.1 Objective

This is the official guide for the Microsoft Outlook Adapter for SAP Integration Suite. This guide covers all relevant information for integration developers to start working with the Microsoft Outlook adapter. Read this guide carefully before using the Adapter.

## 1.2 Coding Samples

Any software coding and/or code lines/strings ("Code") included in this documentation are only examples and are not intended to be used in a productive system environment. The Code is only intended to better explain and visualize the syntax and phrasing rules of certain coding. The correctness and completeness of the Code given herein are not guaranteed.

## 1.3 Internet Hyperlinks

The documentation may contain hyperlinks to the Internet. These hyperlinks are intended to serve as a hint about where to find related information. The availability and the correctness of this related information or the ability of this information to serve a particular purpose are not warranted.

## 1.4 Overview

Microsoft Outlook offers secure email communication along with calendar, scheduling, task capabilities etc. Microsoft Outlook adapter for SAP Integration Suite enables seamless integration between Microsoft Outlook and SAP Integration Suite by improving communication and automating workflows between the two platforms.

This adapter allows you to access the APIs relating to email communication, calendar management, and task organization etc. in Microsoft Outlook. Simplify and accelerate your email-based integrations using the Microsoft Outlook Adapter.

## 1.5 Features

- Sender and Receiver adapter to read and manage Microsoft Outlook resources such as **Calendar**, **Mail Folders**, **Attachments**, and **Messages** respectively.
- Supports authentication methods like **OAuth2 Authorization Code (Delegated Permission)** for user-level access and **OAuth2 Client Credentials (Application Permission)** for organization/admin-level access.
- Enables fetching emails directly by specifying **folder names** or **folder ID**.
- Sender adapter allows retrieval of **all messages, only unread**, or messages fulfilling custom filter expression.
- Sender adapter provides flexible post-processing actions, such as **Mark as Read**, **Delete**, **Archive**, and **Archive & Mark as Read**.
- Receiver adapter supports **pagination** via the **Get next page result** operation.
- Receiver adapter supports **querying** resources to retrieve desired data.
- Receiver adapter supports various content types such as **Application/JSON**, **Text/Plain**, and **MIME**.

## 1.6 Quick Start



This Quick Start section outlines minimal steps required to demo adapter usage. This does not cover all configuration scenarios, always refer to the complete documentation for complete and accurate guidance.


1. Prerequisites:
  - a) Microsoft Outlook account (see [Create a Microsoft Outlook account](#)).
  - b) SAP Integration Suite tenant with Standard License.
2. Now you must configure a few things on your application side. To Create an Application on Azure, add a redirect URI and create Client ID and Client Secret, see [3.2.2 App Setup](#).
3. Create **OAuth2 Client Credentials** in SAP Integration Suite:
  - a) Navigate to **Monitor > Integrations and APIs**.
  - b) Go to **Manage Security > Security Material**.
  - c) Create an **OAuth2 Client Credentials** using
    - i. Use **Client ID**, **Client Secret** and **Redirect URI** created in step 2.

- ii. Set **Token Service URL ID** as  
`https://login.microsoftonline.com/<<tenantID>>/oauth2/v2.0/token`
    - iii. Set **Scope** as `https://graph.microsoft.com/.default`
    - iv. Set **Content Type** as `application/x-www-form-urlencoded`
  4. Add Microsoft Outlook Adapter to an Integration Flow:
    - a) Go to the **Design** workspace and create or open an Integration Flow.
    - b) Add **MicrosoftOutlook** Adapter as a Receiver.
  5. Configure **Connection** Tab:
    - a) **Address:** `https://graph.microsoft.com`
    - b) **Credential Name:** Enter **OAuth2 Client Credentials** artifact created in step 2c.
    - c) Keep default timeout values.
  6. Configure **Processing** :
    - a) Perform [5.2.2 Send Message](#) operation.
  7. Deploy & Run:
    - a) **Save** and deploy the Integration Flow.
    - b) Monitor execution under **Integrations and APIs**.

# 2 Installation and Configuration

This section details the prerequisites to install and configure the Microsoft Outlook adapter.


## 2.1 Prerequisites

 The Microsoft Outlook adapter is available as part of your Standard license for SAP Integration Suite. For more information, see [SAP Note](#).

Before you start working with the adapter, you must deploy it to your SAP Integration Suite tenant.

## 2.2 Procedure

You can deploy the adapter using the following methods:

 The following installation procedure is compatible with Apache Camel 3x environment, and Edge Integration Cell (EIC) platform.

### 2.2.1 Adapter Installation by creating a New Integration Flow



The Microsoft Outlook adapter is available for selection in the receiver adapter list and can be deployed in the **Design** tab directly as you use it in an Integration flow.

#### Purpose

To install an adapter for use in your Integration flow.

#### Procedure

Go to the **Design** workspace and select the integration package where you want to create a new Integration flow.

1. Click **Edit** to make the package editable.
2. Go to the **Artifacts** tab. Click **Add** and select **Integration Flow**.
3. Enter **Name** and **ID** for your flow. Additionally, select **Runtime Profile** from the drop-down and choose **Sender** and **Receiver** systems from the list . Finally, click **Add** to create the integration flow.
4. Go to the newly created integration flow and click **Edit** to make it editable.
5. In the integration flow, click **End** to add a **Connector**  between the **End** and the **Receiver** Box.

A drop-down with the available adapters appears. The **MicrosoftOutlook** adapter should show up in the list.

6. Select the **MicrosoftOutlook** adapter from the list. The adapter is now imported which *triggers* an adapter deployment. Once the Microsoft Outlook Adapter is deployed, a success message is displayed.

After the above steps are done, the Microsoft Outlook Adapter is successfully deployed in your Design workspace of the SAP Integration Suite tenant.

## 2.2.2 Adapter Installation without creating a New Integration Flow



The following procedure explains how the Microsoft Outlook adapter is migrated from the **Discover** workspace to the **Design** workspace of the SAP Integration tenant.

This method is useful for scenarios where integration flow packages are migrated from development to a higher environment such as Production.

The Microsoft Outlook adapter can be imported into the Design workspace without creating an integration flow. Use the Transport Management Service (TMS) to import/transport the Microsoft Outlook adapter to a higher environment. Alternatively, if the TMS is not available in the landscape, the adapter package can be imported to the Design workspace by copying it from the Discover workspace.

### Purpose

To copy the integration package from the Discover workspace and import the Microsoft Outlook adapter to the Design workspace, follow these steps:

### Procedure

1. Go to **Discover** workspace.
2. In the search box, search for **MicrosoftOutlook adapter for SAP Integration Suite** package.
3. Select the package and click **Copy**. This copies the package from the Discover workspace to the Design workspace.
4. Go to Design workspace and select the copied **MicrosoftOutlook adapter for SAP Integration Suite** package.
5. In the **Actions** tab of the selected package, click **Deploy**. This completes the adapter deployment to the Design workspace.

After the adapter deployment is complete, you can check the status in the **Monitor** section.

## Purpose

To check the status of the deployed adapter.

## Procedure

1. Under the **Monitor** tab, click **Integrations and APIs**. This opens the **Overview** page.
2. On the **Overview** page, go to **Manage Integration Content** section and click **All**. This opens the **Integration Content** page with a list of all the deployed adapters.
3. Here, you can check and confirm the deployment status of your adapter.

The screenshot shows the 'Integration Content (392)' page. On the left, a search bar contains 'MicrosoftOutlook' and a table lists the adapter with a 'Started' status. The main panel displays 'MicrosoftOutlook' details, including deployment time, version (1.0.0), and a success message: 'The Integration Adapter is deployed successfully.'

Name	Status
MicrosoftOutlook Integration Adapter	Started

**MicrosoftOutlook** [Undeploy](#)

Deployed On: Jan 02, 2026, 15:47:01 ID: [redacted]  
Deployed By: [redacted] Version: 1.0.0  
Package: [redacted]

**Status Details**

✓ The Integration Adapter is deployed successfully.

# 3 Getting Started: Microsoft Outlook Adapter

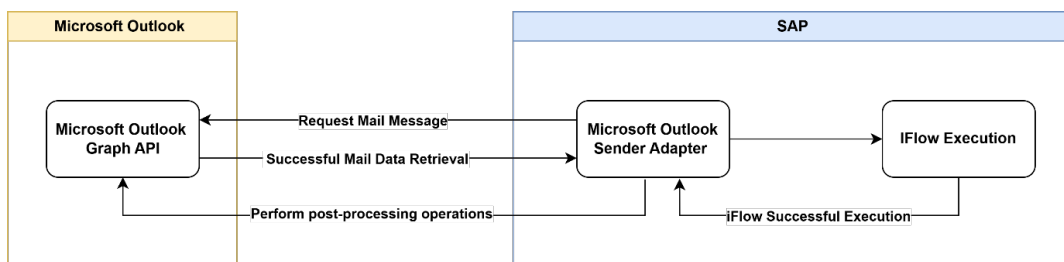
This section contains information to help you get started with the Microsoft Outlook Adapter, along with relevant setup and configurations on the application side of Microsoft Outlook. You can also find information about adapter architecture and the creation of security artifacts.

## 3.1 Architecture Overview

The Microsoft Outlook adapter (as the name suggests) is designed process data from the integration flow and communicate with an external Outlook application. It is designed to function as a sender and receiver adapter.

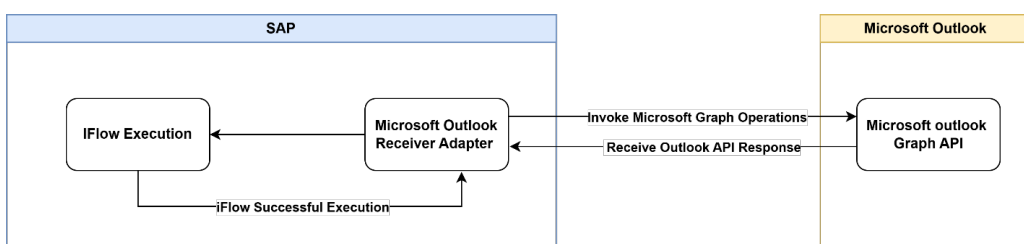
### 3.1.1 Sender Adapter

- The Microsoft Outlook Sender adapter allows connection to the Microsoft Outlook (external) system to fetch or receive incoming data.
- It then triggers iFlow execution based on the adapter configuration and finally handles post-processing actions.



### 3.1.2 Receiver Adapter

- The Microsoft Outlook Receiver adapter is designed to receive and process data from an integration flow in SAP Integration Suite and facilitate communication with Microsoft Outlook application (external) using Graph APIs.
- SAP Integration Suite acts as the initiator of the calls and sends a request to Microsoft Outlook (this is the receiver system) using the adapter. It invokes the Microsoft Outlook server to perform operations like Send Message, Create Calendar, etc. Additionally, the adapter also handles responses and logs execution status and errors before returning them to SAP Integration Suite tenant.



## 3.2 Application Configuration

The following resources can help you set up your Microsoft Outlook application before you start using the adapter in an integration flow.

### 3.2.1 Account Creation

To create a Microsoft Outlook account, see [Create a Microsoft Outlook account](#).

### 3.2.2 App Setup

1. To register an application, [Register an application in Microsoft Entra ID](#).
2. To set up redirect URI for authentication, [Add a redirect URI](#).
3. To create Client ID and Client Secret, see [Add Credentials](#).
4. For permissions, see [Add permissions to access Microsoft Graph](#).

## 3.3 Authentication



Before setting up the authentication, ensure you have finished the required steps as directed in [Application Configuration](#).

This section explains the procedure to create security artifacts that are supported by the Microsoft Outlook Adapter in the SAP Integration Suite.

The adapter supports standard security artifacts: **OAuth2 Client Credentials** and **OAuth2 Authorization Code**. These security artifacts are stored in **Security Material** and can be accessed in the adapter by referencing the alias name.

For setting up the authentication, you must create the required security artifacts in the Security Material of the SAP Integration Suite.

### 3.3.1 Creating OAuth2 Client Credentials for Application Permission

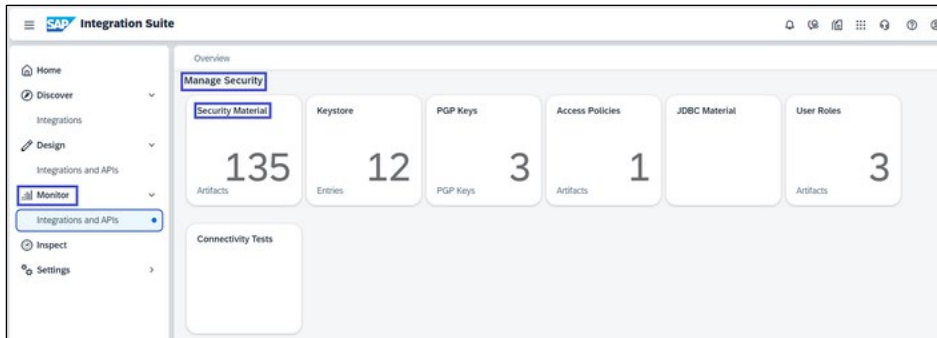
#### Purpose

To create credentials in Security Material for **OAuth2 Client Credentials**.

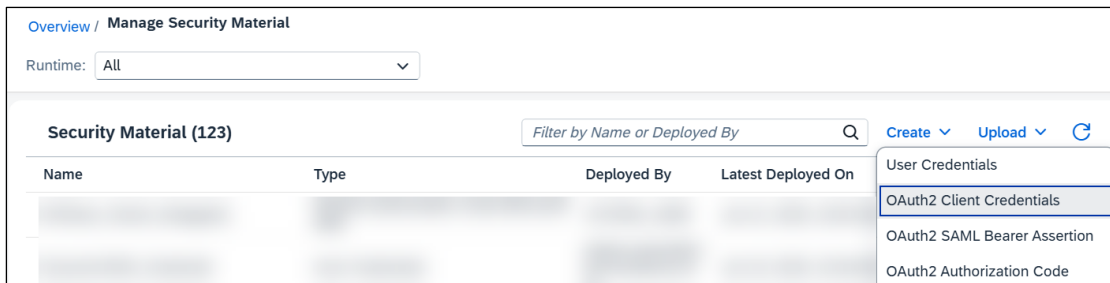
#### Procedure

1. In SAP Integration Suite, navigate to **Monitor > Integrations and APIs**. This opens the **Overview** page.

- On the **Overview** page, go to **Manage Security** section and click **Security Material**.



- On **Manage Security Material** page, click **Create** to select **OAuth2 Client Credentials** from the dropdown.






- In the **Create OAuth2 Client Credentials** popup, provide the below details.

The screenshot shows the 'Edit OAuth2 Client Credentials' form. The fields are as follows:

- Name: MSOutlook\_OAuth\_AppPermissions
- Description: AppPermissions
- Runtimes: Cloud Integration
- Token Service URL: https://login.microsoftonline.com/29aed333-2840-4e...
- Grant Type: Send as Part of Body
- Client ID: [Redacted]
- Client Secret: [Redacted]
- Client Authentication: Send as Body Parameter
- Scope: https://graph.microsoft.com/default
- Content Type: application/x-www-form-urlencoded
- Resource: [Empty]
- Audience: [Empty]

Below the form is a 'Custom Parameters' section with 'Add' and 'Delete' buttons. A table with columns 'Key', 'Value', and 'Send as Part of' is shown, currently containing 'No data'. At the bottom right are 'Deploy' and 'Cancel' buttons.

Parameter	Description
<b>Name</b>	Specify the name for the credentials.
<b>Description</b>	Specify the description for the credentials.
<b>Token Service URL</b>	Specify the URL of the OAuth2 authorization server that issues the access token.  Example: <code>https://login.microsoftonline.com/⟨tenantID⟩/oauth2/v2.0/token</code>
<b>Client ID</b>	Specify the ID of the client to which you are connecting.   Ensure that the Azure application setup is completed to obtain the Tenant ID, Client ID, and Client Secret. For more information, see <a href="#">Application Configuration</a> .
<b>Client Secret</b>	Specify the Secret key of the client to which you are connecting.   OAuth2 uses a multiple-step authentication pattern: Client credentials (Client ID and Client Secret, as specified in the artifact) are used by the client application to initially request an access token. The access token is then used to authorize the client (for as long as the token is valid) to access the server's resources (for example, the resources that are used in the associated integration flow). In many OAuth2 scenarios, the access token is issued (or generated) by an authorization server.
<b>Client Authentication</b>	Select the Client Authentication which allows you to access an application using Client ID and Client Secret.  By default, the <b>Send as Body Parameter</b> is selected, this option sends the Client ID and Client Secret as a JSON content to the authentication server in the request body.  If you select the <b>Send as Request Header</b> , then the Client ID and Client Secret are encoded, and send to the server as an Authorization header.
<b>Scope</b>	Specify the OAuth2 scope information to be included in the request body.  Example: <code>https://graph.microsoft.com/.default</code>
<b>Content Type</b>	Select content type to indicate the media type.   To use application permissions in the adapter, deploy an OAUTH2 Client Credential security artifact and ensure Content Type is set to <b>'application/x-www-form-urlencoded'</b> :

- Click **Deploy** to complete the process.

When you refresh the **Manage Security Material** page, the new artifact is displayed (with Type **OAuth2 Client Credentials**) in the artifact table.

### 3.3.2 OAuth2 Authorization Code for Delegated permission

#### Purpose

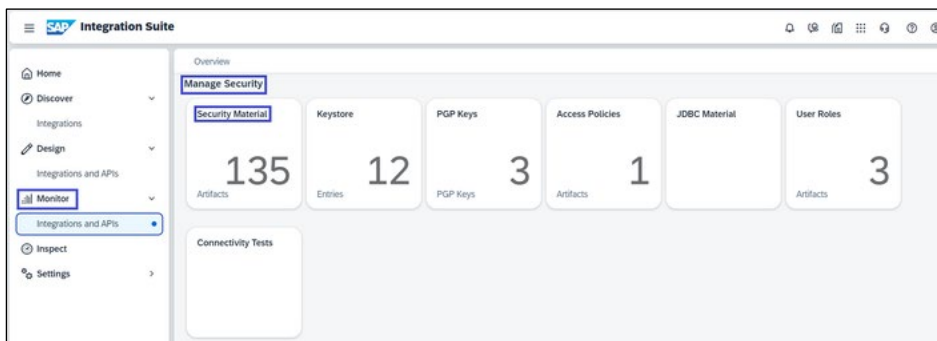
To create credentials in Security Material for **OAuth2 Authorization Code**.



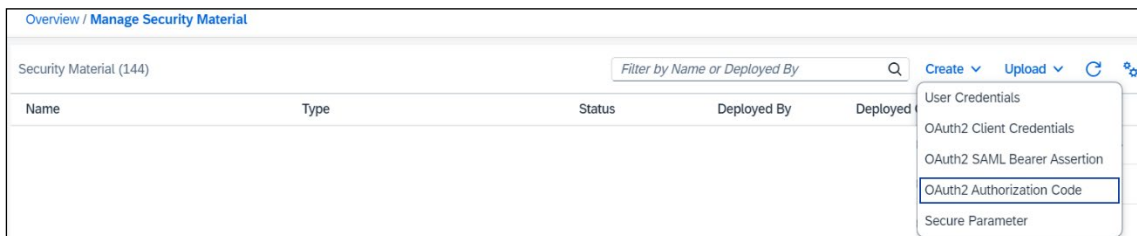
The OAuth2 Authorization Code security artifact for Delegated permissions is not supported for deployment to the Edge Integration Cell (EIC).

#### Procedure




- In SAP Integration Suite, navigate to **Monitor > Integrations and APIs**. This opens the **Overview** page.
- On the **Overview** page, go to **Manage Security** section and click **Security Material**.



- On Manage Security Material page, click **Create** to select **OAuth2 Authorization Code** from the dropdown.

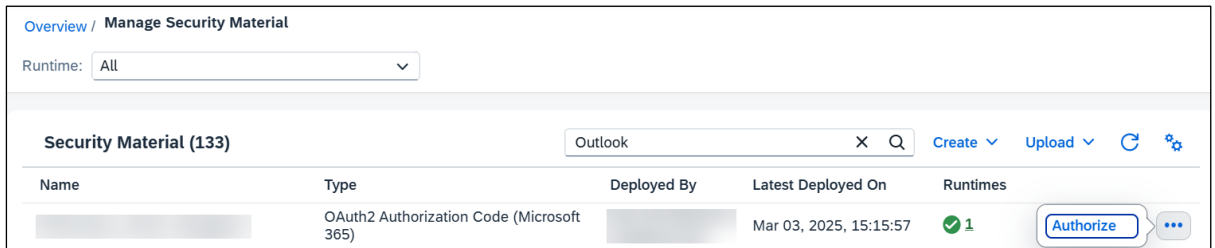




Parameter	Description
<b>Authorization URL</b>	Specify the Authorization URL for authorizing the OAuth client to access resources of a user.  Example: <code>https://login.microsoftonline.com/&lt;&lt;tenantId&gt;&gt;/oauth2/v2.0/authorize</code>
<b>Token Service URL</b>	Specify the Address of the token service that issues the access token.  Example: <code>https://login.microsoftonline.com/&lt;&lt;tenantId&gt;&gt;/oauth2/v2.0/token</code>
<b>Redirect URL</b>	Displays the URL you need, when creating the OAuth Clients/App in OAuth Authorization Server/Token Server.   Ensure that the Redirect URL is added to the Azure application.
<b>Client ID</b>	Specify the ID of the client you want to connect to.   Ensure that the Azure application setup is completed to obtain the Tenant ID, Client ID, and Client Secret. For more information, see <a href="#">Application Configuration</a> .
<b>Client Secret</b>	Specify the Secret key of the client to which you are connecting.
<b>Send As</b>	Select as Body Parameter which sends the Client ID and Client Secret in the request body when calling the Authorization URL or Token Service URL.
<b>User Name</b>	Specify the name of the user whose resources the OAuth2 client gets access to.
<b>Scope</b>	Specify the OAuth2 scopes protecting the access to the resources.  The scope value is added to the Authorization URL (see above) as query parameter and is also added to the Token Service URL, in case of the "authorization_code" grant type request; but is not added to the Token Service URL in case of the "refresh_token" grant type request (see above).   If you add more than 1 scope, you need to separate your scopes by a blank space.

5. Click **Deploy**.

6. After deployment, click **Authorize** to complete the process.



When you refresh the **Manage Security Material** page, the new artifact is displayed (with Type **OAuth2 Authorization Code**) in the artifact table.

# 4 Microsoft Outlook Adapter Configuration

This section defines the configurations for the Microsoft Outlook adapter. You need to configure the **Connection** and **Processing** tabs. A description and example usage for every field has been added.

## 4.1 Sender Adapter

In this section, you will learn how to configure the Microsoft Outlook sender adapter.

### 4.1.1 Connection

The Connection tab contains connection and authentication parameters required to connect to Microsoft Outlook.

The Security artifacts created in the previous section ([Creating Credentials in Security Material](#)) are referenced in the **Connection tab** of the adapter as shown in the figure below.

The connection tab contains the following fields:

Parameter	Description
<b>Address</b>	Specify the address for the Microsoft Outlook service. Default: <code>https://graph.microsoft.com</code>
<b>Authentication</b>	Select the authentication method and permission type: 'Delegated' for user-level access, 'Application' for app-level access. <ul style="list-style-type: none"><li><b>OAuth2 Client Credentials (Application Permissions)</b></li><li><b>OAuth2 Authorization Code (Delegated Permissions)</b></li></ul>

Parameter	Description
<b>Credential Name</b>	Specify the security artifact for the selected authentication type to connect to the Microsoft Outlook account.
<b>Polling Interval (in ms)</b>	Specify the poll interval in milliseconds. Example: 600000
<b>Reuse Connection</b>	Enable this property to reuse the connection.
<b>Connection Timeout (in ms)</b>	Specify the maximum waiting time (in milliseconds) for the connection to be established. Example: 60000
<b>Response Timeout (in ms)</b>	Specify the maximum waiting time (in milliseconds) for a response message to be received. Example: 60000




## 4.1.2 Processing

This section lists fields under the **Processing** tab configurations for the Microsoft Outlook Sender Adapter.

The screenshot shows the configuration interface for the Microsoft Outlook Sender Adapter, specifically the **Processing** tab. The interface includes the following fields and options:

- User ID:** john.doe@outlook.com
- Folder Identifier:** Name (dropdown menu)
- Folder Name:** Inbox
- Selection:** Only Unread (dropdown menu)
- Max Messages per Node:** 2
- Lock Timeout (in ms):** 120000
- Post-Processing:** Mark as Read (dropdown menu)
- Response Type:** Application/JSON (dropdown menu)

Parameter	Description
<b>User ID</b>	Specify the user principal name (email address) or ID to perform the operation. Example: john.doe@outlook.com or b393b406-9cb1-4b5d-5332-b7d6be49c81b
<b>Folder Identifier</b>	Select the mode to identify the folder: <ul style="list-style-type: none"> <li><b>Name</b></li> </ul>

Parameter	Description
	<ul style="list-style-type: none"> <li><b>ID</b></li> </ul>
<b>Folder Name</b>  (Only available when the Folder Identifier is set to <b>Name</b> )	Specify the Folder Name that contains the emails to be read. Folder names are not valid for child folders, use the mail folder ID to specify them instead.  Example: <code>Inbox</code>   If left blank, emails from all folders will be read.
<b>Mail Folder ID</b>  (Only available when the Folder Identifier is set to <b>ID</b> )	Specify the Mail Folder ID that contains the emails to be read.  Example: <code>AQMkAGY5NzFiZjE3LTgwNDgtNDBlYS1jM2Q4LTI3ZTQ1QjE3NGEzODkzLgAAAYtmMD5sTKHsQXpqNcMHR1CAP06ZHfBE8eMu0QQS_GmeY9BBBICDAAAAA==</code>   If left blank, emails from all folders will be read.
<b>Selection</b>	Specify which mails are to be processed (all mails or only unread ones). This field is editable, allowing you to use a filter expression to fetch mails.  <ul style="list-style-type: none"> <li><b>All</b></li> <li><b>Only Unread</b></li> </ul>  This field currently supports only filter queries.  Example: <code>sentDateTime lt 2025-11-28T12:00:00Z</code>
<b>Max Messages per Node</b>	Specify the maximum number of mails to be retrieved per worker node.  Example: <code>2</code>
<b>Lock Timeout (in ms)</b>	Specify the maximum duration (in milliseconds) when the lock is active during the polling process.  Example: <code>120000</code>
<b>Post-Processing</b>	Select the action to be performed after successful iflow processing: <ul style="list-style-type: none"> <li><b>Archive</b></li> <li><b>Archive and Mark as Read</b></li> <li><b>Delete</b></li> <li><b>Mark as Read</b></li> </ul>

Parameter	Description
<b>Archive Folder Identifier</b>  (Only available when the <b>Post-Processing</b> is set to <b>Archive</b> or <b>Archive and Mark as Read</b> )	Select the mode to identify the archive folder. <ul style="list-style-type: none"> <li>• <b>Name</b></li> <li>• <b>ID</b></li> </ul>
<b>Archive Folder Name</b>  (Only available when the Archive Folder Identifier is set to <b>Name</b> )	Specify the name of the archive folder where the file should be archived.  Example: <code>Archive</code>
<b>Archive Mail Folder ID</b>  (Only available when the Archive Folder Identifier is set to <b>ID</b> )	Specify the ID of the archive folder where the file should be archived.
<b>Response Type</b>	Select the content type for the response payload to be returned. <ul style="list-style-type: none"> <li>• <b>Application/JSON</b></li> <li>• <b>Content Only</b></li> <li>• <b>MIME</b></li> </ul>

## 4.2 Receiver Adapter

In this section, you will learn how to configure the Microsoft Outlook receiver adapter. On selecting the Microsoft Outlook adapter from the list of adapters, you must configure the **Connection**, and **Processing** tabs.

### 4.2.1 Connection

The Connection tab contains connection and authentication parameters for Microsoft Outlook.

The Security artifact created in the previous section ([Creating Credentials in Security Material](#)) should be used in the **Connection tab** of the Adapter as shown in the figure below.

The screenshot shows the 'MicrosoftOutlook' application window with the 'Connection' tab selected. The 'CONNECTION DETAILS' section contains the following fields:

- Address:** A text input field containing 'https://graph.microsoft.com'.
- Authentication:** A dropdown menu set to 'OAuth2 Authorization Code (Delegated Permissions)'.
- Credential Name:** A text input field with a greyed-out placeholder.
- Reuse Connection:** An unchecked checkbox.
- Connection Timeout (in ms):** A text input field containing '60000'.
- Response Timeout (in ms):** A text input field containing '60000'.

The connection tab contains the following fields:

Parameter	Description
<b>Address</b>	Specify the address for the Microsoft Outlook service. Default: <code>https://graph.microsoft.com</code>
<b>Authentication</b>	Select the authentication method and permission type: 'Delegated' for user-level access, 'Application' for app-level access: <ul style="list-style-type: none"> <li>• <b>OAuth2 Client Credentials (Application Permissions)</b></li> <li>• <b>OAuth2 Authorization Code (Delegated Permissions)</b></li> </ul>
<b>Credential Name</b>	Specify the security artifact for the selected authentication type to connect to the Microsoft Outlook account.
<b>Reuse Connection</b>	Enable this property to reuse the connection.
<b>Connection Timeout (in ms)</b>	Specify the maximum waiting time (in milliseconds) for the connection to be established.
<b>Response Timeout (in ms)</b>	Specify the maximum waiting time (in milliseconds) for a response message to be received.

## 4.2.2 Processing

The Processing tab lists all the operations that can be performed through the adapter.

MicrosoftOutlook Externalize ?

General Connection **Processing**

PROCESSING DETAILS

Resource: Message

Operation: List messages

User ID: \* b393b406-9cb1-4b5d-5332-b7d6be49c81b

Folder Identifier: Name

Folder Name: Inbox

Query Parameters:

HEADER DETAILS

Request Headers:

Response Headers: \*

Parameter	Description
<b>Processing Details</b>	
<b>Resource</b>	Select the resource on which to perform an operation. <ul style="list-style-type: none"> <li>• <b>Mail Folder</b></li> <li>• <b>Message</b></li> <li>• <b>Attachment</b></li> <li>• <b>Calendar</b></li> </ul>
<b>Operation</b>	Select the operation to be performed. Example: List Messages
<b>User ID</b>	Specify the user principal name (email address) or ID to perform the operation. Example: b393b406-9cb1-4b5d-5332-b7d6be49c81b
<b>Group ID</b>	Specify the Group ID to perform the operation. Example: b991a872-b048-43ae-9074-4246a246745e
<b>Folder Identifier</b>	Select the mode to identify the folder using its ID or name.
<b>Folder Name</b>	Specify the name of the folder where the operation will be performed.
<b>Mail Folder ID</b>	Specify the ID of the mail folder where the operation will be performed.
<b>Calendar Group ID</b>	Specify the ID of the Calendar Group.

<b>Parameter</b>	<b>Description</b>
<b>Calendar ID</b>	Specify the Calendar ID to perform the operation.
<b>Mail Folder ID</b>	Specify the Mail Folder ID to perform the operation.
<b>Mail Message ID</b>	Specify the Mail Message ID to perform the operation.
<b>Attachment ID</b>	Specify the Attachment ID associated with a message, event, etc.
<b>Child Folder ID</b>	Specify the Child Folder ID associated with the subfolder on which this operation will be performed.
<b>Thread ID</b>	Specify the Thread ID. This ID represents a conversation thread within a group.
<b>Event ID</b>	Specify the Event ID of the calendar event.
<b>Conversation ID</b>	Specify the Conversation ID. This ID groups related messages or posts into a thread.
<b>Post ID</b>	Specify the Post ID. This ID represents an individual message or post within a thread.
<b>Start Date Time</b>	Specify the date and time from when the calendar view should be fetched. Example: <code>2026-09-08T15:00:00</code>
<b>End Date Time</b>	Specify the date and time until when the calendar view should be fetched. Example: <code>2026-09-08T16:00:00</code>
<b>Request Format</b>	Select the payload format of the request body. Example: <b>Application/JSON</b>
<b>Response Format</b>	Select the payload format of the response body. Example: <b>Text/Plain</b>

Parameter	Description
<b>Next Link URL</b>	<p>Specify the Next Link URL to perform the operation.</p> <p>Example:  <code>https://graph.microsoft.com/v1.0/users/{{userId}}/mailFolders?%24select=id%2cdisplayName%2cisHidden&amp;%24top=8&amp;%24skip=8</code></p> <p>Replace <code>{{userId}}</code> with the actual user ID.</p>
<b>Delta Link (URL)</b>	<p>Specify the Delta Link (URL) to perform the operation; leave blank for the initial request.</p>
<b>Query Parameters</b>	<p>Specify the query parameter. When using multiple parameters, ensure they are separated by "&amp;".</p> <p>Example: <code>\$top=20&amp;\$expand=members</code></p>
<b>Header Details</b>	
<b>Request Headers</b>	<p>Enter a list of custom headers, separated by a pipe ( ), to be sent to the target system. Use an asterisk (*) to send all custom headers to the target system. All Camel-specific headers and HTTP protocol headers except "date" are excluded by default even if you specify them. Note that this value can also be read dynamically using an exchange header or property.</p>
<b>Response Headers</b>	<p>Enter a list of headers, separated by a pipe ( ), coming from the target system's response to be received in the message. Use an asterisk (*) to receive all the headers from the target system, which is also the default value. All Camel-specific headers and HTTP protocol headers except "date" are excluded by default even if you specify them. Note that this value can also be read dynamically using an exchange header or property.</p>

# 5 Microsoft Outlook Supported Operations

This section lists and describes some of the operations supported by the Microsoft Outlook adapter. For detailed information on any particular operation, see [Microsoft Outlook documentation](#).

## 5.1 Sender Adapter

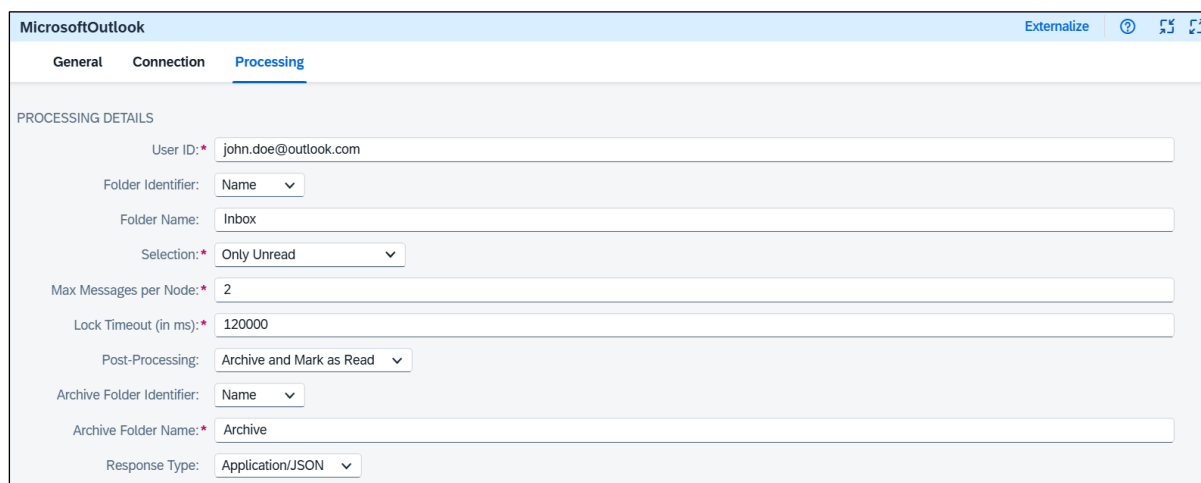
The Outlook sender adapter enables reading email messages from the mailbox and managing post-processing actions.

### 5.1.1 Read Mail Message

In this scenario, the adapter reads email messages from the specified folder by using the folder name. It fetches only unread emails and allows you to specify the maximum number of messages per node to process, as well as the lock timeout for the polling process.

Once a message is successfully retrieved, it is archived to the Archive folder by specifying the folder name. The email messages are then marked as read, and the successful response is returned in Application/JSON format.

A sample screenshot is provided below.



The screenshot shows the 'MicrosoftOutlook' configuration window with the 'Processing' tab selected. The 'PROCESSING DETAILS' section contains the following fields:

- User ID: john.doe@outlook.com
- Folder Identifier: Name (dropdown)
- Folder Name: Inbox
- Selection: Only Unread (dropdown)
- Max Messages per Node: 2
- Lock Timeout (in ms): 120000
- Post-Processing: Archive and Mark as Read (dropdown)
- Archive Folder Identifier: Name (dropdown)
- Archive Folder Name: Archive
- Response Type: Application/JSON (dropdown)

For the complete list of descriptions about the fields below, refer to the [Processing Tab](#).

Parameter	Value
User ID	Specify the User ID as john.doe@outlook.com
Folder Identifier	Select the folder identifier as <b>Name</b> .
Folder Name	Specify the name as <b>Inbox</b> .
Selection	Specify the selection as <b>Only Unread</b> .

Parameter	Value
<b>Max Messages per Node</b>	Set as 2.
<b>Lock Timeout (in ms)</b>	Set as 120000.
<b>Post-Processing</b>	Select the post-processing to <b>Archive and Mark as Read</b> .
<b>Archive Folder Identifier</b>	Select the archive folder identifier as <b>Name</b> .
<b>Archive Folder Name</b>	Specify the name as <i>Archive</i> .
<b>Response Type</b>	Select as <b>Application/JSON</b> .

## 5.2 Receiver Adapter



For a complete list of all available operations and the corresponding permissions required in the Microsoft Outlook Adapter, see [Appendix](#).

### 5.2.1 Create draft message

The create a new message draft enables you to compose an email, including defining the recipients, subject, body, and attachments, without sending it immediately. The created draft can be saved and later sent by the user.

To create a draft message, you can use either **JSON format** or **MIME format**:

- **JSON format:** Allows you to include attachments and later update the draft to modify the body or other properties.
- **MIME format:** Requires base64-encoded Internet message headers and MIME content, including any attachments and S/MIME properties.

Both formats enable the creation of a message draft, with JSON being simpler for structured data and MIME providing more flexibility for advanced scenarios.

A sample screenshot is provided below.

The screenshot shows the 'Processing' tab in the Microsoft Outlook adapter configuration. It contains the following fields:

- Resource: Message (dropdown)
- Operation: Create draft message (dropdown)
- User ID: b393b406-9cb1-4b5d-5332-b7d6be49c81b (text input)
- Folder Identifier: Name (dropdown)
- Folder Name: (text input)
- Query Parameters: (text input)
- Request Format: Application/JSON (dropdown)

For the complete list of descriptions about the fields below, refer to the [Processing Tab](#).

Parameter	Value
Resource	Select the resource as <b>Message</b> .
Operation	Select the operation as <b>Create draft message</b> .
User ID	Specify the User ID as <b>b393b406-9cb1-4b5d-5332-b7d6be49c81b</b>
Request Format	Select the request format as <b>Application/JSON</b>



The Mail Folder ID can be configured if the draft needs to be created in a specific folder.



For more information about query parameters etc. that can be used in this operation, see [Message API documentation](#).

Sample request body to create a new message draft using JSON format:

```
{
  "subject": "Did you see last night's game?",
  "importance": "Low",
  "body": {
    "contentType": "HTML",
    "content": "They were <b>awesome</b>!"
  },
  "toRecipients": [
    {
      "emailAddress": {
        "address": "AdeleV@contoso.com"
      }
    }
  ]
}
```

## 5.2.2 Send Message

The **Send message** operation in Microsoft Outlook allows you to send an email directly without creating a draft first and then sending it via send draft message operation. You can provide a request body including subject, body, and recipients.

When using JSON format, you can include a [file attachment](#) in the same **sendMail** action call.

When using MIME format:

- Provide the applicable [Internet message headers](#) and the [MIME content](#), all encoded in **base64** format in the request body.
- Add any attachments and S/MIME properties to the MIME content.

A sample screenshot is provided below.

MicrosoftOutlook Externalize ?

General Connection **Processing**

PROCESSING DETAILS

Resource: Message

Operation: Send message

User ID: \* b393b406-9cb1-4b5d-5332-b7d6be49c81b

Query Parameters:

Request Format: Application/JSON

For the complete list of descriptions about the fields below, refer to the [Processing Tab](#).

Parameter	Value
<b>Resource</b>	Select the resource as <b>Message</b> .
<b>Operation</b>	Select the operation as <b>Send Message</b> .
<b>User ID</b>	Specify the User ID as b393b406-9cb1-4b5d-5332-b7d6be49c81b
<b>Request Format</b>	Select the request format as <b>Application/JSON</b> .



For more information about query parameters for this operation , see [Send Mail API documentation](#).

Sample payload for sending a Message with an Attachment:

```
{
  "message": {
    "subject": "Meet for lunch?",
    "body": {
      "contentType": "Text",
      "content": "The new cafeteria is open."
    },
    "toRecipients": [
      {
        "emailAddress": {
          "address": "meganb@contoso.com"
        }
      }
    ],
    "attachments": [
      {
        "@odata.type": "#microsoft.graph.fileAttachment",
        "name": "attachment.txt",
        "contentType": "text/plain",
        "contentBytes": "SGVsbG8gV29ybGQh"
      }
    ]
  }
}
```

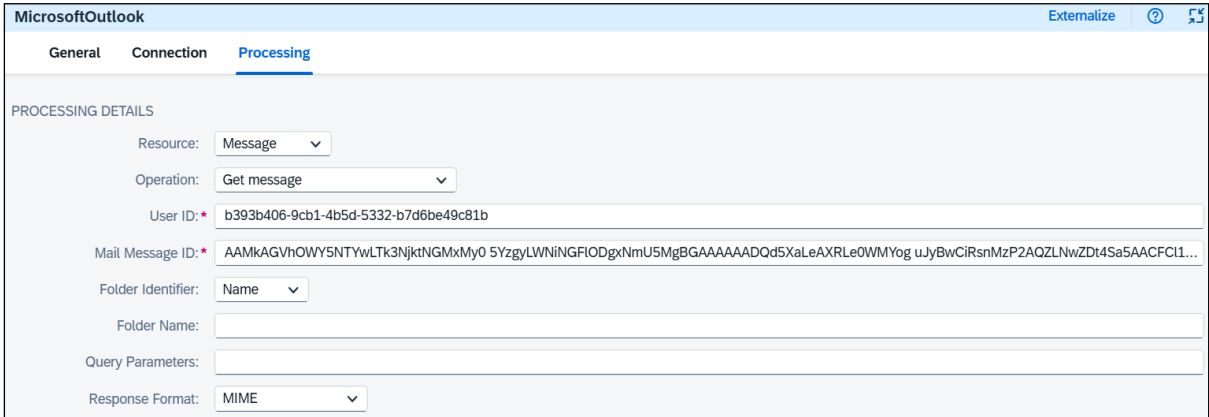
```
}
}
```

### 5.2.3 Get Message

The **Get message** operation is used to retrieve the details of a specific email message from a mailbox. You can fetch messages from the signed-in user’s mailbox or another user’s mailbox using their user ID or user principal name.

In addition to retrieving the message in JSON format (default), you can also request the MIME content by specifying the response format as **Application/JSON** or **MIME**.

A sample screenshot is provided below.



For the complete list of descriptions about the fields below, refer to the [Processing Tab](#).


Parameter	Value
<b>Resource</b>	Select the resource as <b>Message</b> .
<b>Operation</b>	Select the operation as <b>Get message</b> .
<b>User ID</b>	Specify the User ID as <code>b393b406-9cb1-4b5d-5332-b7d6be49c81b</code>
<b>Mail Message ID</b>	Specify the Message ID as <code>AAMkAGVhOWY5NTYwLTk3NjktNGMxMy05YzgyLWNiNGFIODgxNmU5MgBGAAAAAADQd5XaLeAXRLe0WMyog uJyBwCiRsnMzP2AQZLNwZDt4Sa5AACFC11UAACiRsnMzP2AQZLNwZDt4Sa5AACFCpAAA=AE6AACiRsnMzP2AQZLNwZDt4Sa5BBBDouhkBAA=</code>
<b>Response Format</b>	Select the response format as <b>MIME</b> .

**i** For more information about query parameters for this operation , see [Get Message API documentation](#).

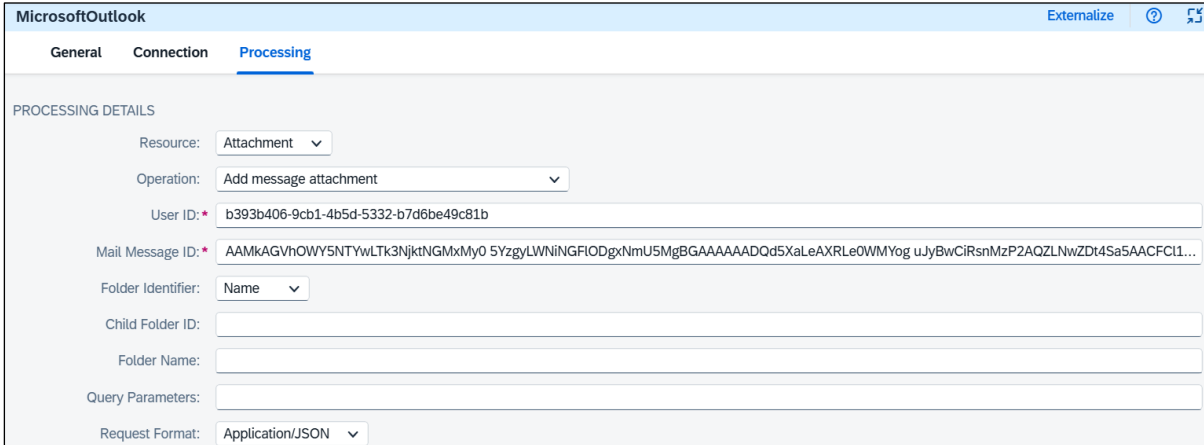
**i** The Mail Folder ID can be configured to retrieve the message from any specific folder.

## 5.2.4 Add message attachment

The **Add message attachment** operation allows you to add attachments like files (file Attachment), or items (item Attachment) to an existing message or a message being created.

 The official limit for attachment size is 3 MB; the API may support larger attachments depending on the context and configuration. It's recommended to test with the intended file size to ensure compatibility.

A sample screenshot is provided below.




The screenshot shows the 'Processing Details' tab in Microsoft Outlook. It contains the following fields:

- Resource: Attachment (dropdown)
- Operation: Add message attachment (dropdown)
- User ID: b393b406-9cb1-4b5d-5332-b7d6be49c81b
- Mail Message ID: AAMkAGVhOWY5NTYwLTk3NjktNGMxMy05YzgyLWNiNGF1ODgxNmU5MgBGAAAAAADQd5XaLeAXRLe0WMMYog uJyBwCiRsnMzP2AQZLNwZDt4Sa5AACFC1...
- Folder Identifier: Name (dropdown)
- Child Folder ID: (empty text box)
- Folder Name: (empty text box)
- Query Parameters: (empty text box)
- Request Format: Application/JSON (dropdown)

For the complete list of descriptions about the fields below, refer to the [Processing Tab](#).

Parameter	Value
<b>Resource</b>	Select the resource as <b>Attachment</b> .
<b>Operation</b>	Select the operation <b>Add message attachment</b> .
<b>User ID</b>	Specify the User ID as <code>b393b406-9cb1-4b5d-5332-b7d6be49c81b</code> .
<b>Mail Message ID</b>	Specify the Message ID as <code>AABkAGVhOWY5NTYwLTk3NjktNGMxMy05YzgyLWNiNGF1ODgxNmU5MgBGAAAAAADQd5XaLeAXRLe0WMMYoguJyBwCiRsnMzP2AQZLNwZDt4Sa5AXSAAAE6AACiRsnMzP2AQZLNwZDt4Sa5BBBDouhkBAA=</code> .
<b>Request Format</b>	Select as request format as <b>Application/JSON</b> .

 The Mail Folder ID and Child Folder ID can be configured if the attachment needs to be added to a message located in a specific folder.

Sample Request payload (Item attachment):

```
{
  "@odata.type": "#microsoft.graph.itemAttachment",
  "name": "Holiday event",
  "item": {
```

```

"@odata.type": "microsoft.graph.event",
"subject": "Discuss gifts for children",
"body": {
  "contentType": "HTML",
  "content": "Let's look for funding!"
},
"start": {
  "dateTime": "2016-12-02T18:00:00",
  "timeZone": "Pacific Standard Time"
},
"end": {
  "dateTime": "2016-12-02T19:00:00",
  "timeZone": "Pacific Standard Time"
}
}
}
}

```



For more information about query parameters for this operation, see [Message Attachments API documentation](#).

## 5.2.5 Create Mail Folder

The **Create Mail folder** operation can be used to create a new mail folder in the user's mailbox.

A sample screenshot is provided below.

The screenshot shows the 'Processing' tab in Microsoft Outlook. Under 'PROCESSING DETAILS', the 'Resource' is set to 'Mail Folder', the 'Operation' is 'Create mail folder', and the 'User ID' is 'b393b406-9cb1-4b5d-5332-b7d6be49c81b'. There is a 'Query Parameters' field at the bottom which is currently empty.

For the complete list of descriptions about the fields below, refer to the [Processing Tab](#).

Parameter	Value
<b>Resource</b>	Select the resource as <b>Mail Folder</b> .
<b>Operation</b>	Select the operation as <b>Create mail folder</b> .
<b>User ID</b>	Specify the Team ID as b393b406-9cb1-4b5d-5332-b7d6be49c81b

Sample Request Body:

```
{
  "displayName": "Clutter",
  "isHidden": true
}
```



For more information about query parameters for this operation, see [Create Mail Folder API documentation](#).

## 5.2.6 List Calendar Events

The List Events operation in allows you to retrieve a collection of events from a calendar. It helps you view all upcoming events, including details like start and end times, locations, and descriptions. This operation is useful for retrieving and managing events on a calendar. You can filter and sort the events based on parameters like date range or event type, providing flexibility in how you view calendar data.

A sample screenshot is provided below.

The screenshot shows the 'Processing' tab in Microsoft Outlook. Under 'PROCESSING DETAILS', the 'Resource' is set to 'Calendar', the 'Operation' is 'List calendar events', the 'User ID' is 'b393b406-9cb1-4b5d-5332-b7d6be49c81b', and the 'Query Parameters' are '\$top=2&\$select=id'.

For the complete list of descriptions about the fields below, refer to the [Processing Tab](#).

Parameter	Value
Resource	Select the resource as <b>Calendar</b> .
Operation	Select the operation as <b>List calendar events</b> .
User ID	Specify the User ID as b393b406-9cb1-4b5d-5332-b7d6be49c81b
Query Parameter	Specify the query parameter as \$top=2&\$select=id.

Sample Response Body:

```
{
  "@odata.context": "https://graph.microsoft.com/v1.0/$metadata#users(b393b406-9cb1-4b5d-5332-b7d6be49c81b)/calendar/events(id)",
  "value": [
```

```

{
  "@odata.etag": "W/\\"okbJzMz9gEGSzcGQ7eEmuQAAQ5FVMg==\",
  "id":
  "ABMkAGVhONA5NTYwLTk3NjktNGMxMy05YzgyLWNiNGFIODgxNmU5MgBGAAAAAADQd5XaLeA
  XRLe0WMYoguJyBwCiRsnMzP2AQZLNwZDt4Sa5AAAAAENAACiRsnMzP2AQZLNwZDt4Sa5AABDor
  SNCCA="
},
{
  "@odata.etag": "W/\\"okbJzMz9gEGSzcGQ7eEmuQAAQ5FTQQ==\",
  "id":
  "ASRkAGVhOWY5NTYwLTk3NjktNGMxMy05YzgyLWNiNGFIODgxNmU5MgBGAAAAAADQd5XaLeA
  XRLe0WMYoguJyBwCiRsnMzP2AQZLNwZDt4Sa5AAAAAENAACiRsnMzP2AQZLNwZDt4Sa5AABDor
  MSAAA="
}
],
"@odata.nextLink": "https://graph.microsoft.com/v1.0/users/b393b406-9cb1-4b5d-5332-
b7d6be49c81b/calendar/events?select=id&%24top=2&%24skip=2"
}

```



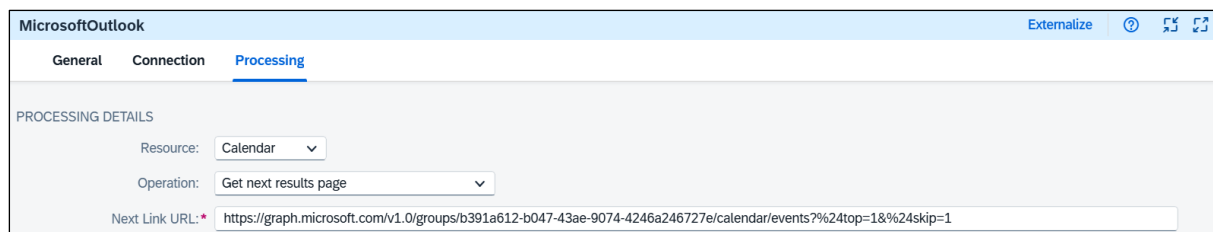
For more information about query parameters for this operation, see [Calendar Events API documentation](#).

## 5.2.7 Get Next Results Page

The **Get Next Results** page can be used for pagination across all resource types. As mentioned in the previous operation, when data is filtered using the query parameter, only a subset of the data is returned.

To retrieve the remaining data, provide the Next Link URL found in the response body of the operation for pagination. (Refer to the Response body of the **List calendar events** operation)

A sample screenshot is provided below.



For the complete list of descriptions about the fields below, refer to the [Processing Tab](#).

Parameter	Value
<b>Resource</b>	Select the resource as <b>Calendar</b> .
<b>Operation</b>	Select the operation as <b>Get next results page</b> .
<b>Next Link URL</b>	Specify the <b>Next Link URL</b> to perform the operation.

## 5.2.8 Find Meeting Times

The **Find Meeting Times** operation helps suggest the best meeting times for a group of attendees. By considering the availability of all participants, it provides optimal time slots that work for everyone. This feature is especially useful for scheduling meetings with multiple people, as it helps avoid conflicts and ensures a time that fits within the participants' available windows.

A sample screenshot is provided below.

The screenshot shows the 'Processing' tab in Microsoft Outlook. Under 'PROCESSING DETAILS', the 'Resource' is set to 'Calendar', the 'Operation' is 'Find meeting times', and the 'User ID' is '6ad943cb-2cfd-4295-9e07-57c5e4d57af3'. There is also a 'Query Parameters' field which is currently empty.

For the complete list of descriptions about the fields below, refer to the [Processing Tab](#).

Parameter	Value
<b>Resource</b>	Select the resource as <b>Calendar</b> .
<b>Operation</b>	Select the operation as <b>Find meeting times</b> .
<b>User ID</b>	Specify the Team ID as <b>6ad943cb-2cfd-4295-9e07-57c5e4d57af3</b>
<b>Query Parameter</b>	Specify the query parameter.

### Request Body:

```
{
  "attendees": [
    {
      "type": "required",
      "emailAddress": {
        "name": "Alex Wilbur",
        "address": "alexw@contoso.com"
      }
    }
  ],
}
```

```

"locationConstraint": {
  "isRequired": false,
  "suggestLocation": false,
  "locations": [
    {
      "resolveAvailability": false,
      "displayName": "Conf room Hood"
    }
  ]
},
"timeConstraint": {
  "activityDomain": "work",
  "timeSlots": [
    {
      "start": {
        "dateTime": "2019-04-16T09:00:00",
        "timeZone": "Pacific Standard Time"
      },
      "end": {
        "dateTime": "2019-04-18T17:00:00",
        "timeZone": "Pacific Standard Time"
      }
    }
  ]
},
"isOrganizerOptional": "false",
"meetingDuration": "PT1H",
"returnSuggestionReasons": "true",
"minimumAttendeePercentage": 100
}

```



For more information about query parameters for this operation, see [Find Meeting Times API documentation](#).

## 5.2.9 Get Message Delta

The Delta query operation allows you to track changes (such as additions, deletions, and updates) to messages in a user's mailbox. By using this operation, you can retrieve only the changes made since the last query, rather than fetching all the data again. This is useful for efficiently syncing email data or keeping track of recent updates.

A successful initial response returns either an @odata.nextLink (containing a skipToken) for continued pagination, or an @odata.deltaLink (containing a deltaToken) once all changes have been retrieved.

After completing pagination, the delta link should be used in the **Delta Link** field on the UI to fetch future incremental changes.

A sample screenshot is provided below.

The screenshot shows the 'Processing' tab in the Microsoft Outlook interface. It contains the following fields:

- Resource:** Message (dropdown)
- Operation:** Get message delta (dropdown)
- User ID:** b393b406-9cb1-4b5d-5332-b7d6be49c81b (text input)
- Folder Identifier:** Name (dropdown)
- Folder Name:** Inbox (text input)
- Delta Link (URL):** https://graph.microsoft.com/v1.0/users/b393b406-9cb1-4b5d-5332-b7d6be49c81b/messages?%24top=10&%24skip=10 (text input)
- Query Parameters:** (empty text input)

For the complete list of descriptions about the fields below, refer to the [Processing Tab](#).

Parameter	Value
<b>Resource</b>	Select the resource as <b>Message</b> .
<b>Operation</b>	Select the operation as <b>Get message delta</b> .
<b>User ID</b>	Specify the User ID as <code>b393b406-9cb1-4b5d-5332-b7d6be49c81b</code> .
<b>Folder Identifier</b>	Select the folder identifier as <b>Name</b> .
<b>Folder Name</b>	Specify the Folder Name as <code>Inbox</code> .
<b>Delta Link (URL)</b>	Specify the Delta Link URL as <code>https://graph.microsoft.com/v1.0/users/b393b406-9cb1-4b5d-5332-b7d6be49c81b/messages?%24top=10&amp;%24skip=10</code>

## 5.2.10 Add event attachment

The Add event attachment operation allows you to add attachments such as files (`fileAttachment`), or items (`itemAttachment`), to an existing event.



The official limit for attachment size is 3 MB, the API may support larger attachments depending on the context and configuration. It's recommended to test with the intended file size to ensure compatibility.

A sample screenshot is provided below.

The screenshot shows the 'Processing' tab in Microsoft Outlook. Under 'PROCESSING DETAILS', the following fields are visible:

- Resource: Attachment (dropdown)
- Operation: Add event attachment (dropdown)
- User ID: 6ad943cb-2cfd-4295-9e07-57c5e4d57af3
- Event ID: AAMkAGVh0WY5NTYwLTk3NjktNGMxMy05YzgyLWNiNGF1ODgxNmU5MgBGAAAAAADQd5XaLeAXRLe0WMyYoguJyBwCiRsnMzP2AQZLNwZDt4Sa5AAAAAEN...
- Calendar Group ID: (empty text box)
- Calendar ID: (empty text box)
- Query Parameters: (empty text box)
- Request Format: Application/JSON (dropdown)

For the complete list of descriptions about the fields below, refer to the [Processing Tab](#).

Parameter	Value
<b>Resource</b>	Select the resource as <b>Attachment</b> .
<b>Operation</b>	Select the operation as <b>Add event attachment</b> .
<b>User ID</b>	Specify the User ID as 6ad943cb-2cfd-4295-9e07-57c5e4d57af3
<b>Event ID</b>	Specify the Event ID as AAMkAGVh0WY5NTYwLTk3NjktNGMxMy05YzgyLWNiNGF1ODgxNmU5MgBGAAAAAADQd5XaLeAXRLe0WMyYoguJyBwCiRsnMzP2AQZLNwZDt4Sa5AAAAAEN...NAACiRsnMzP2AQZLNwZDt4Sa5AACFCrchAAA=
<b>Request Format</b>	Select the request format as <b>Application/JSON</b> .

**i** The **Calendar Group ID** and **Calendar ID** can be configured to add an attachment to an event located within a specific calendar and calendar group. Ensure that all required field values are provided to execute the correct and valid endpoint operation. For more information, refer to the official documentation.

Request payload for item attachment:

```
{
  "@odata.type": "#microsoft.graph.itemAttachment",
  "name": "Holiday event",
  "item": {
    "@odata.type": "microsoft.graph.event",
    "subject": "Discuss gifts for children",
    "body": {
      "contentType": "HTML",
      "content": "Let's look for funding!"
    },
    "start": {
      "dateTime": "2016-12-02T18:00:00",
      "timeZone": "Pacific Standard Time"
    },
    "end": {
      "dateTime": "2016-12-02T19:00:00",
      "timeZone": "Pacific Standard Time"
    }
  }
}
```



For more information about query parameters for this operation, see [Event Attachments API documentation](#).

# 6 Support

## 6.1 Troubleshooting

Behaviour/Error Message	Possible Reason	Possible Solution
<p>Emails are not delivered to the recipient.</p> <p><b>Error message:</b> Your message to <code>john.doe@outlook.com</code> couldn't be delivered.</p> <div style="background-color: #e6f2ff; padding: 10px; border: 1px solid #d9e1f2;"> <p>The iFlow execution is successful as the message is handed off by SAP Integration Suite. Any email delivery failure caused by external factors is reported in the Outlook UI and does not impact the iFlow status.</p> </div>	<p>The email was not delivered due to one or more of the following reasons:</p> <ul style="list-style-type: none"> <li>• Invalid or incorrect recipient email address</li> <li>• Recipient mailbox is full or inactive</li> <li>• Recipient mail server temporarily unavailable</li> <li>• Email blocked by spam or security filters</li> </ul>	<p>Verify that the recipient email address is correct and active. If the issue persists, check the email server or spam filter settings, or contact your system administrator.</p>
<p>The adapter reads the same email messages repeatedly on every polling cycle.</p>	<p>This behaviour occurs when the adapter configuration is set to retrieve all emails, while the post-processing action is configured to mark messages as read. If the emails are already marked as read before processing, the adapter does not update its state and repeatedly fetches the same messages during each polling cycle.</p>	<p>To resolve this issue, increase the maximum number of messages per node or change the post-processing action to an option other than <i>mark as read</i>, ensuring that processed messages are not picked up again in subsequent polls.</p>

Behaviour/Error Message	Possible Reason	Possible Solution
<p><b>Error message:</b> An error occurred while resolving the processing details., Reason: The property/header '<code>\${header.ABC}</code>' does not exist in the exchange. Properties/Headers are case sensitive.</p>	<p>An optional field related to required IDs is configured using a dynamic expression (for example, <code>\${header.ABC}</code>), but the referenced header or property is not present in the message exchange, or its key/value is not populated. Although the field is optional, the adapter validates the dynamic expression at runtime.</p>	<p>When configuring dynamic expressions in optional ID-related fields, ensure that the corresponding header or property key and value are explicitly set in the message exchange. Alternatively, avoid configuring dynamic expressions for these fields if the value may be unavailable at runtime.</p>
<p>Inconsistent Data Retrieval while Pagination</p>	<p>Server-side pagination does not always return the expected response in the initial request, leading to inconsistent data retrieval in certain cases.</p>	<p>It is recommended to implement additional checks, such as handling next-page tokens or retry mechanisms, to ensure complete data retrieval.</p> <p>For more details, see <a href="#">Paging Microsoft Graph Data</a>.</p>
<p>Query Parameter Handling with Parentheses</p>	<p>Parentheses in query parameters may not be correctly interpreted by SAP CI during deployment, leading to incorrect structuring.</p> <p>Example:  <code>\$filter=startswith(displayName,'abc')</code>  <code>&amp;\$select=displayName,description</code>  may cause an error during iFlow deployment.</p>	<p>It is recommended to handle the query parameters dynamically to avoid iFlow deployment issues.</p>
<p>Microsoft Graph Throttling Specifications</p>	<p>Microsoft Graph API enforces limits and throttling for performance optimization</p>	<p>For more information about this, refer to <a href="#">Throttling Specifications</a>.</p>

# 7 Appendix

This section lists and describes all the operations supported by the Microsoft Outlook adapter. For detailed information on any particular operation, see [Microsoft Outlook API Documentation](#).

The following table provides information regarding various operations available in the Microsoft Outlook Receiver Adapter.

Resource	Operation	Permissions Allowed
Calendar	List calendars	Delegated and Application
Calendar	Create calendar	Delegated and Application
Calendar	Get default calendar	Delegated and Application
Calendar	Get default calendar of a group	Delegated
Calendar	Update default calendar	Delegated and Application
Calendar	Update default calendar of a group	Delegated
Calendar	Delete calendar	Delegated and Application
Calendar	List default calendar view	Delegated and Application
Calendar	List calendar events	Delegated and Application
Calendar	List calendar events of a group	Delegated
Calendar	Create calendar event	Delegated and Application
Calendar	Create calendar event in a group	Delegated
Calendar	Get free/busy schedule	Delegated and Application
Calendar	Find meeting times	Delegated
Calendar	List calendar groups	Delegated and Application
Calendar	Create calendar group	Delegated and Application
Calendar	Get calendar group	Delegated and Application
Calendar	Update calendar group	Delegated and Application
Calendar	Delete calendar group	Delegated and Application
Calendar	Get next results page	Delegated and Application
MailFolder	List mail folders	Delegated and Application
MailFolder	Get mail folder	Delegated and Application
MailFolder	Create mail folder	Delegated and Application
MailFolder	List child folders	Delegated and Application
MailFolder	Create child folder	Delegated and Application
MailFolder	Update mail folder	Delegated and Application

<b>Resource</b>	<b>Operation</b>	<b>Permissions Allowed</b>
MailFolder	Delete mail folder	Delegated and Application
MailFolder	Copy mail folder	Delegated and Application
MailFolder	Get mail folder delta	Delegated and Application
MailFolder	Move mail folder	Delegated and Application
MailFolder	Get next results page	Delegated and Application
Message	List messages	Delegated and Application
Message	Create draft message	Delegated and Application
Message	Get message	Delegated and Application
Message	Update message	Delegated and Application
Message	Delete message	Delegated and Application
Message	Copy message	Delegated and Application
Message	Create draft to forward message	Delegated and Application
Message	Create draft to reply	Delegated and Application
Message	Create draft to reply-all	Delegated and Application
Message	Get message delta	Delegated and Application
Message	Forward message	Delegated and Application
Message	Move message	Delegated and Application
Message	Reply to a message	Delegated and Application
Message	Reply-all to a message	Delegated and Application
Message	Send draft message	Delegated and Application
Message	Send message	Delegated and Application
Message	Get next results page	Delegated and Application
Attachment	List message attachments	Delegated and Application
Attachment	Add event attachment	Delegated and Application
Attachment	Add message attachment	Delegated and Application
Attachment	Add post attachment	Delegated
Attachment	List event attachments	Delegated and Application
Attachment	List post attachments	Delegated
Attachment	Delete event attachment	Delegated and Application
Attachment	Delete event attachment in specific calendar	Delegated and Application
Attachment	Delete event attachment in default calendar group	Delegated and Application
Attachment	Delete event attachment in specific calendar group	Delegated and Application

<b>Resource</b>	<b>Operation</b>	<b>Permissions Allowed</b>
Attachment	Delete message attachment	Delegated and Application
Attachment	Delete message attachment in top level folder	Delegated and Application
Attachment	Delete message attachment in child folder	Delegated and Application
Attachment	Delete post attachment	Delegated
Attachment	Delete post attachment in a specific conversation	Delegated
Attachment	Get next results page	Delegated and Application
Attachment	List message attachments	Delegated and Application
Attachment	Add message attachment	Delegated and Application