

medepia

Dropbox Adapter for Cloud Platform Integration User Guide

Version	Release Date	Description
1.0	29.10.2018	First release of the document

Topics covered in this document:

- Configuring Dropbox account for the integration
- Sender adapter features and configuration examples
- Receiver adapter features and configuration examples
- Use cases with Integration Flow examples

Table of Contents

- 1. Configuring Dropbox Account 4
 - 1.1. Creating an app for the integration 4
 - 1.2. Creating a new credential with Dropbox access token..... 6
 - 1.3. Revoking an access token 7
- 2. Sender Adapter Features and Configuration 9
 - 2.1. Basic Example..... 9
 - 2.2. File name pattern matching 9
 - 2.3. Action after read.....10
 - 2.4. Scheduler options 11
 - 2.5. CamelFileName header 11
- 3. Receiver 12
 - 3.1. Basic Example..... 12
 - 3.2. CamelFileName header 12
 - 3.3. File name addition..... 13
 - 3.4. Dropbox behaviour 13
 - 3.5. Example configuration..... 13
- 4. Example Integration Flows 14
 - 4.1. SFTP to Dropbox integration 14
 - 4.2. Dropbox to SFTP integration 14
 - 4.3. Dropbox to Dropbox integration..... 14
 - 4.4. Logging incoming files to Dropbox..... 15
- 5. Exception Handling FAQ 15

1. Configuring Dropbox Account

1.1. Creating an app for the integration

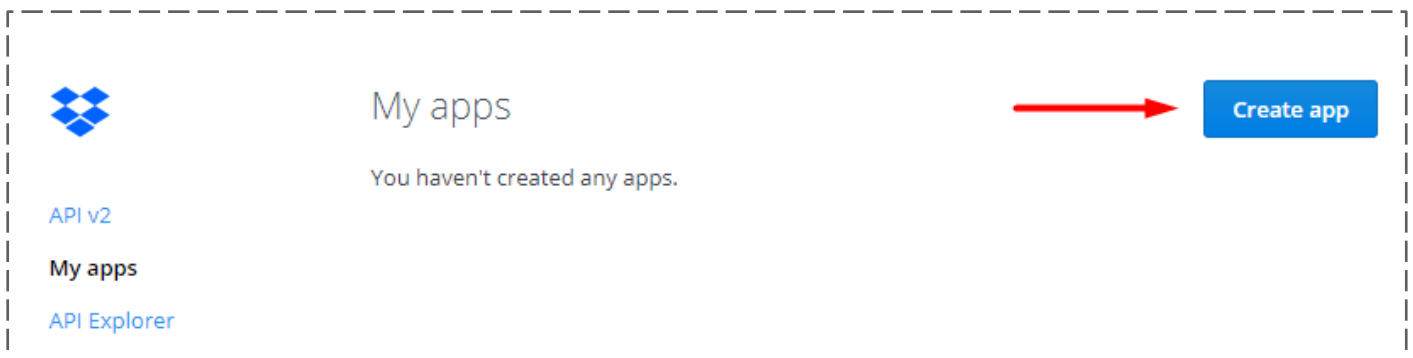
Go to Dropbox developers web site:

<https://www.dropbox.com/developers>

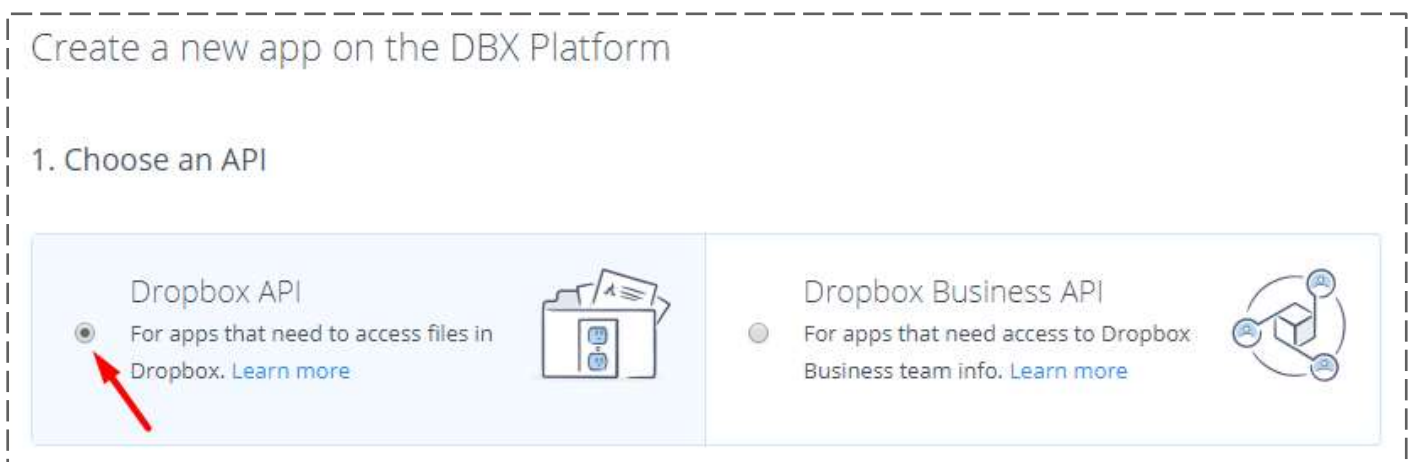
Go to "My apps" and log in if necessary:



Click "Create app":



Dropbox API is enough for the integration:



For test and logging purposes it is recommended to use "App folder" access. If you have a specific business account you may consider "Full Dropbox" access.

2. Choose the type of access you need

[Learn more about access types](#)

- App folder – Access to a single folder created specifically for your app.
- Full Dropbox – Access to all files and folders in a user's Dropbox.

Choose a unique name for your application and click Create app:

3. Name your app

com.medepia.example ←

→ Create app

You should generate and copy access token. This access token does not expire and requires manual action for invalidation/revocation. Please note that access token is private information:

App folder name: com.medepia.example

App key: yhtmrpb75os55q2

App secret: Show

OAuth 2

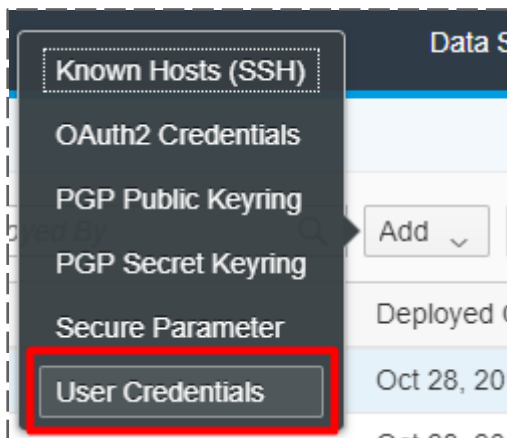
Redirect URIs: https:// (http allowed for localhost)

Allow implicit grant: Allow

Generated access token: **Generate**

1.2. Creating a new credential with Dropbox access token

Go to your tenant's operational view. Add "User Credentials" from top right:



You should enter your credential to Password fields twice. "User" field is not important for the adapter you can enter the same value as the "Name" field. We will use the "Name" field while configuring the adapter. Then click "Deploy".

Add User Credentials

*Name: medepia_dropbox_credential

Description:

*User: medepia_dropbox_credential

Password:

Repeat Password:

SuccessFactors

Deploy Cancel

1.3. Revoking an access token

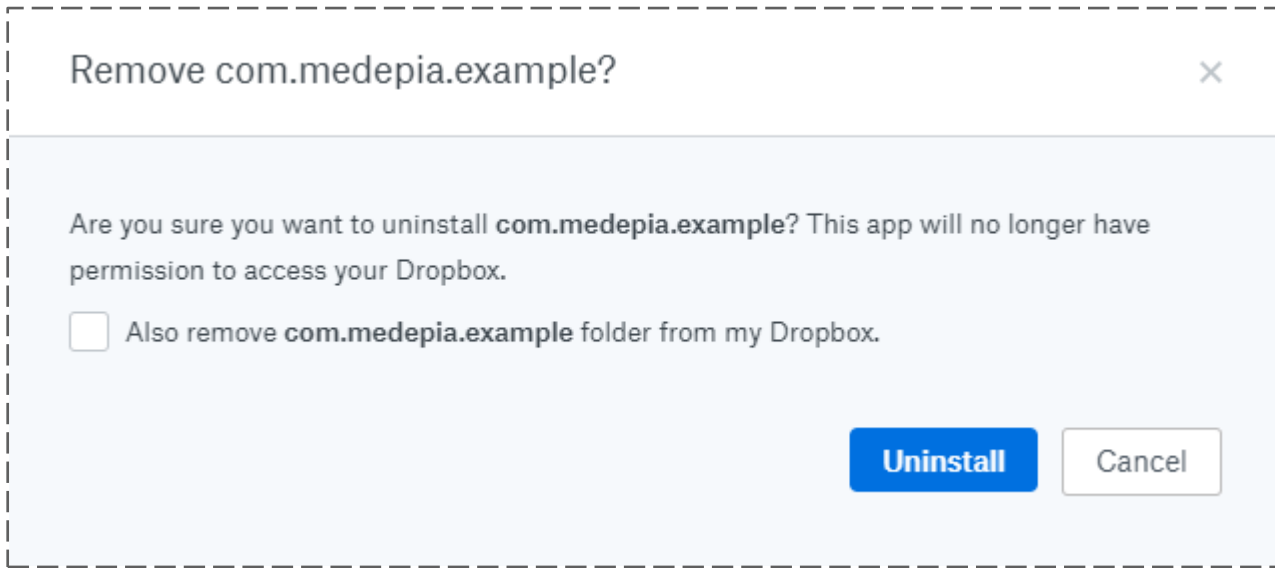
Access tokens for the Dropbox API don't expire. You can read more about tokens here:

<https://www.dropbox.com/developers/support#token-expiration>

For manually revoking token. Go to: https://www.dropbox.com/account/connected_apps

Linked apps
You've given these apps access to your personal Dropbox account.

App name	Publisher	Access type
com.medepia.example	com.medepia.example	App folder



Alternatively you can remove App folder from your Dropbox to revoke issued access tokens.

2. Sender Adapter Features and Configuration

2.1. Basic Example

You can see general features in this example.

- We have provided the access token
- The adapter will only read files ending with ".txt"
- Reading directory is "/test/source"
- The adapter will archive files under "/test/source/archive" after a successful read

Dropbox

General **Parameters** Scheduler

ACCESS TOKEN CREDENTIAL

Access Token Credential:

FILE

File Name:

File Directory:

Action After Read:

Archive Path:

2.2. File name pattern matching

File Name parameter supports regex pattern matching:

Example	Result
".*txt"	Only read files ending with "txt" e.g. "Data.txt", "Datatxt"
".*\.*txt"	Only read files ending ".txt" e.g. "Data.txt"
".*\.*[tT][xX][tT]"	Only read files ending ".txt", ".TXT" or ".tXt" etc.
".*\.*pdf"	Only read files ending ".pdf"
"F.*na.*me\.*txt"	It matches "Filename.txt", "FilenameReadme.txt" etc.
".*"	Read every file in the folder.

2.3. Action after read

Action after read parameter supports "Test", "Archive", and "Delete"



- "Test" option only reads file and does nothing afterwards
- "Archive" option archives file after reading. ".archive" means "/reading_folder/archive" as in the standard SFTP adapter
- "Delete" option deletes file after a successful read.

2.4. Scheduler options

Scheduler option simply starts the process given the configured interval:

The screenshot shows the 'Dropbox' configuration window with the 'Scheduler' tab active. The 'TIMER' section is visible, and the 'Every:' dropdown menu is open, displaying a list of intervals: 10 sec, 30 sec, 1 min, 5 min, 10 min, 15 min, 20 min, 30 min, 1 hr, 2 hr, 3 hr, and 4 hr. The '10 sec' option is currently selected.

Example	Result
Every 10 seconds	Starts the process every 10 seconds starting with 00. e.g. "05:19:00", "05:19:10", "05:19:20", "05:19:30"...
Every 15 minutes	Starts the process every 15 minutes starting with 00. e.g. "05:00:00", "05:15:00", "05:30:00", "05:45:00"...
Every 2 hours	Starts the process every 2 hours starting with 00:00. e.g. "00:00:00", "02:00:00", "04:00:00", "06:00:00"...
Every 6 hours	Starts the process every 6 hours starting with 00:00. e.g. "00:00:00", "06:00:00", "12:00:00", "18:00:00"...

2.5. CamelFileName header

Receiver channel will add "CamelFileName" header containing the original name of the file. You can automatically use this feature with the standard SFTP receiver.

3. Receiver

3.1. Basic Example

You can see general features in this example.

- We have provided the access token
- Writing directory is "/test/target"
- File name pattern will be "ReceivedFromCPI.txt"
- The adapter will add timestamp before file extension. e.g.: " ReceivedFromCPI-2018-10-29T16-11-18.321.txt"

Dropbox

General Parameters

ACCESS TOKEN CREDENTIAL

Access Token Credential: medepia_dropbox_credential

FILE

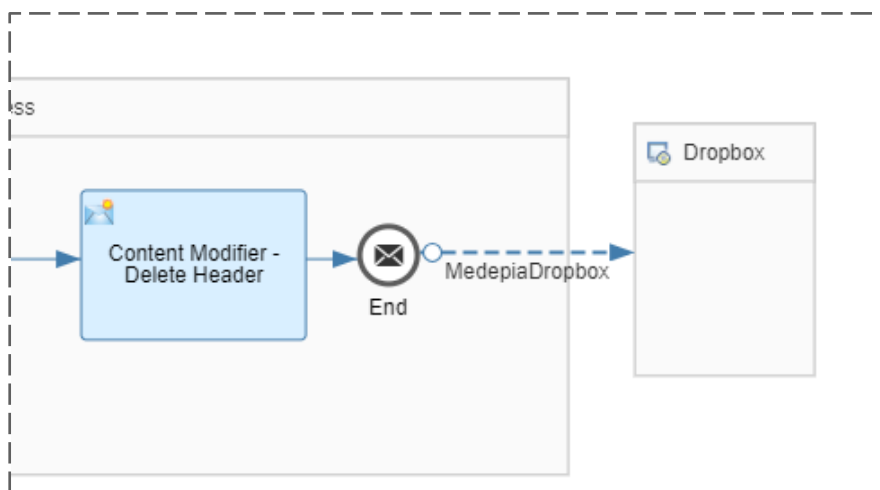
Writing File Directory: /test/target

File name to write: ReceivedFromCPI.txt

File Name Addition: Add timestamp

3.2. CamelFileName header

The adapter has preference for CamelFileName header over manual configuration. If you want to explicitly use manual configuration, you should delete CamelFileName header before receiver.



3.3. File name addition

File name addition option can add a timestamp or UUID to enforce uniqueness.



3.4. Dropbox behaviour

If receiver adapter sends two files with the same name. Dropbox renames the second file to resolve the conflict

If you delete a file manually on your computer and adapter sends same file name with the same contents in seconds. Dropbox may think it is a sync problem and doesn't create the new file.

3.5. Example configuration

CamelFileName	File name	File Name Addition	Result
"MyDocument.pdf"	"Doc.pdf"	Add timestamp	"MyDocument.pdf"
""	"Doc.pdf"	Add timestamp	"MyDocument-2018-10-29T16-11-18.321.pdf"
"LogProps-v1.txt"	"FromCPI.txt"	Add UUID	"LogProps-v1.txt"
""	"FromCPI.txt"	Add UUID	"FromCPI-592f8f7c-5821-4853-a5af-7e131b00a590.txt"
""	"FromCPI.txt"	Do not add anything	"FromCPI.txt"
""	"FromCPI.txt"	Do not add anything	"FromCPI (1).txt" Here Dropbox renames the file to resolve the conflict

4. Example Integration Flows

4.1. SFTP to Dropbox integration



4.2. Dropbox to SFTP integration



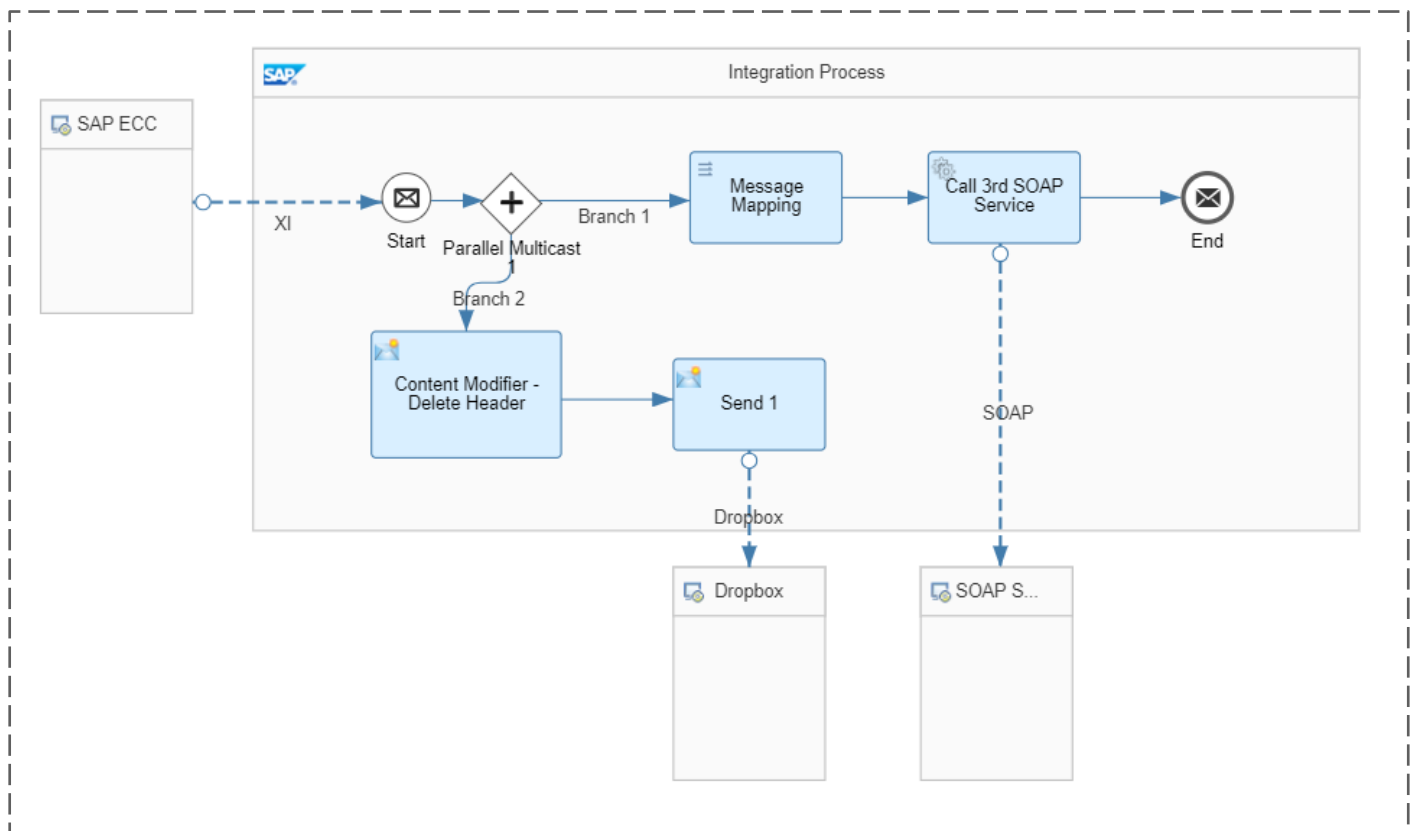
4.3. Dropbox to Dropbox integration

Here sender adapter sets "CamelFileName" header, so you will see the same filename in the writing folder without making any other configuration.



4.4. Logging incoming files to Dropbox

Here is a typical scenario SAP on premise system connects to an 3rd party SOAP Web Service. We want to log requests to our Dropbox account for testing purposes. You can set CamelFileName in a "content modifier".



5. Exception Handling FAQ

Q1: What would happen if my access token is wrong?

A1: If access token is wrong on the sender adapter you should check tracing logs. If access token is wrong on the receiver adapter your messages will fail and you can see them in the monitoring view.

Q2: The message seems successful in the monitoring, however I can not see on my Windows client.

A2: Please check in the browser. Some characters are not permitted on Windows and they will not be synced.

Q3: How can I check if my regex in the sender adapter is correct?

A3: Regex functionality uses the standard Java string matching. You can also use online tools.