



How to access Excel and Delimited Files in SharePoint from Denodo

Revision 20210608

NOTE

This document is confidential and proprietary of **Denodo Technologies**.
No part of this document may be reproduced in any form by any means without prior
written authorization of **Denodo Technologies**.

Copyright © 2021
Denodo Technologies Proprietary and Confidential

CONTENTS

1 CONTENT.....	3
2 CREATING AN APP IN THE AZURE PORTAL.....	4
3 CONFIGURING ACCESS TO SHAREPOINT.....	6
4 API PERMISSIONS.....	7
5 CLIENT SECRET KEY GENERATION.....	8
6 CREATING AN EXCEL DATA SOURCE.....	9
7 REFERENCES.....	14

1 CONTENT

In this document we explain how to create an Excel data source in Denodo that retrieves an Excel file stored in SharePoint 365 using the Sharepoint API. The same steps can be performed for Delimited Files as well.

It is also possible to access SharePoint files using OData following the steps described in [How to connect Denodo to Sharepoint Online using OData with OAuth authentication](#).

Unless you need to access SharePoint using OData the procedure explained in this document is the recommended approach.

2 CREATING AN APP IN THE AZURE PORTAL

For the SharePoint application to be accessible, API permissions have to be configured. A client application token has to be generated for the access token. In order to generate the token the following has to be done in the Azure Portal:

- Login to the Azure portal
- Click on App registrations > New registration.
- Select the desired supported account types. The type of the application support depends on your requirement. For the purpose of this documentation we are using **Accounts in this organizational directory only (Default Directory only - Single tenant)**.
- Provide a **Redirect URI**. The redirect URI does not have to be a valid one. But it is recommended to be a HTTPS URL.

[Home](#) > [App registrations](#) >

Register an application ...

* Name

The user-facing display name for this application (this can be changed later).

sharepointDenodo ✓

Supported account types

Who can use this application or access this API?

- Accounts in this organizational directory only (Denodo only - Single tenant)
- Accounts in any organizational directory (Any Azure AD directory - Multitenant)
- Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)
- Personal Microsoft accounts only

[Help me choose...](#)

Redirect URI (optional)

We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.

Web ✓

By proceeding, you agree to the [Microsoft Platform Policies](#)

[Register](#)

3 CONFIGURING ACCESS TO SHAREPOINT

Once the app is registered we will need to grant permissions for clients to access SharePoint using the app:


- In the newly registered application, click on **Integration assistant** (on the left side menu) and select **Daemon** for **What application types are you building** and select **Evaluate my registration**.
- Once the evaluation is done, the Recommended configurations are shown.

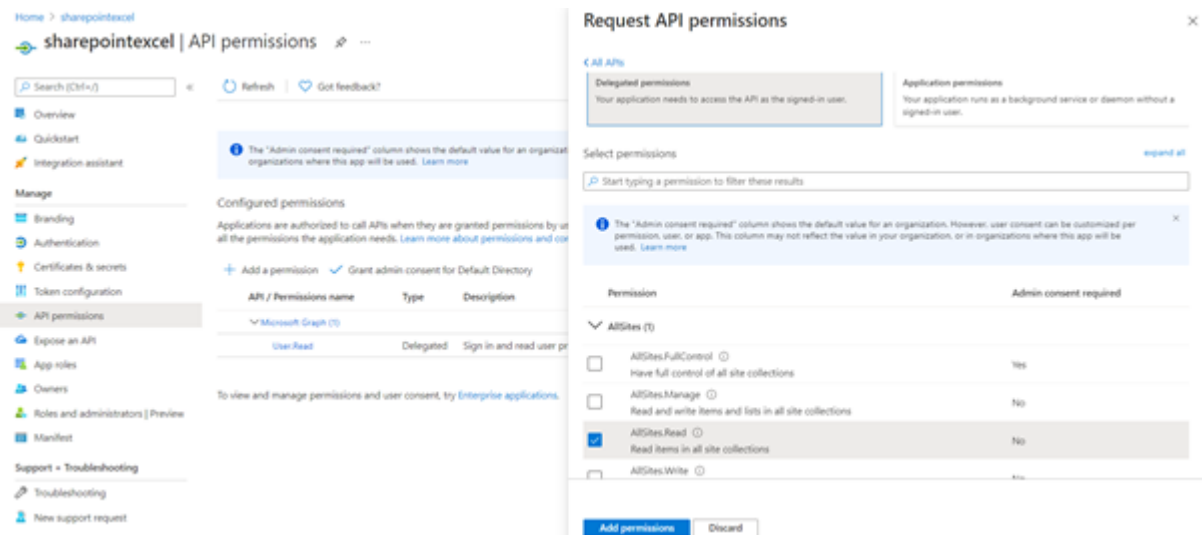
Recommended configurations

Item	Status
Configure API permissions.	 Action required ***
Configure a valid credential.	 Action required ***

There are 2 needed configurations that have to be performed to allow the Denodo Platform to access the data in SharePoint: API permissions and client secret generation.

4 API PERMISSIONS

- Click on the  icon on Configure API permissions and select Go to page.
- Select Add Permissions. In the Request API permissions pop-up select SharePoint.
- Select Delegated Permissions and select AllSites.Read as the permissions



The screenshot displays the Azure AD API permissions configuration page. On the left, the navigation pane is open to 'API permissions'. The main content area shows 'Configured permissions' with a table listing existing permissions:

API / Permissions name	Type	Description
Microsoft Graph (1)		
User.Read	Delegated	Sign in and read user profile

Below the table, there is a section for 'Request API permissions' which is currently open as a modal dialog. The dialog shows 'All APIs' selected. Under 'Select permissions', the following permissions are listed:

Permission	Admin consent required
AllSites.FullControl (0)	Yes
AllSites.Manage (0)	No
AllSites.Read (0)	No
AllSites.Write (0)	No

The 'AllSites.Read' permission is selected, and the 'Admin consent required' column shows 'No' for this permission.

5 CLIENT SECRET KEY GENERATION

After the API permissions are set we have to generate a new client secret key. In order to generate the key:

- In the Integration assistance section, select Configure a valid credential.
- In the Certificates & secrets page select the option New client secret.
- After generating the Client secret, copy the **Value**.

| Certificates & secrets ✎ ...

<< [Got feedback?](#)

Credentials enable confidential applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.

Certificates

— Certificates can be used as secrets to prove the application's identity when requesting a token. Also can be referred to as public keys.

[↑ Upload certificate](#)

Thumbprint	Start date	Expires	Certificate ID
No certificates have been added for this application.			

Client secrets

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

[+ New client secret](#)

Description	Expires	Value	Secret ID
client secret	11/27/2021	[REDACTED]	[REDACTED]

The API access is now set up to connect Denodo to SharePoint.

In the App Registration page in the Azure portal, select the app that has been registered for the SharePoint integration with Denodo:

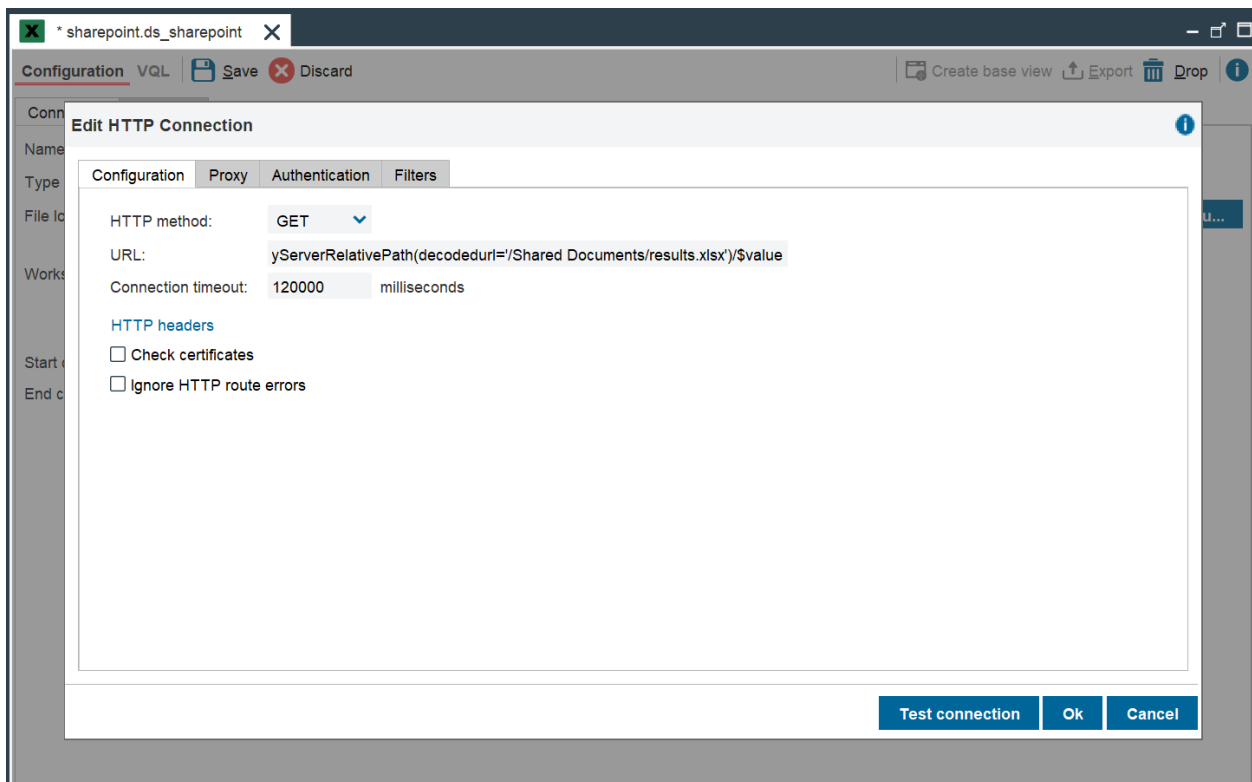
- Copy the **Application (client) ID** this will be the **Client identifier** in the data source configuration.
- Copy the **Value of the Client Secret** generated for the new application.

6 CREATING AN EXCEL DATA SOURCE

In this section we will explain how to create an Excel data source in Denodo to read Excel files from SharePoint.

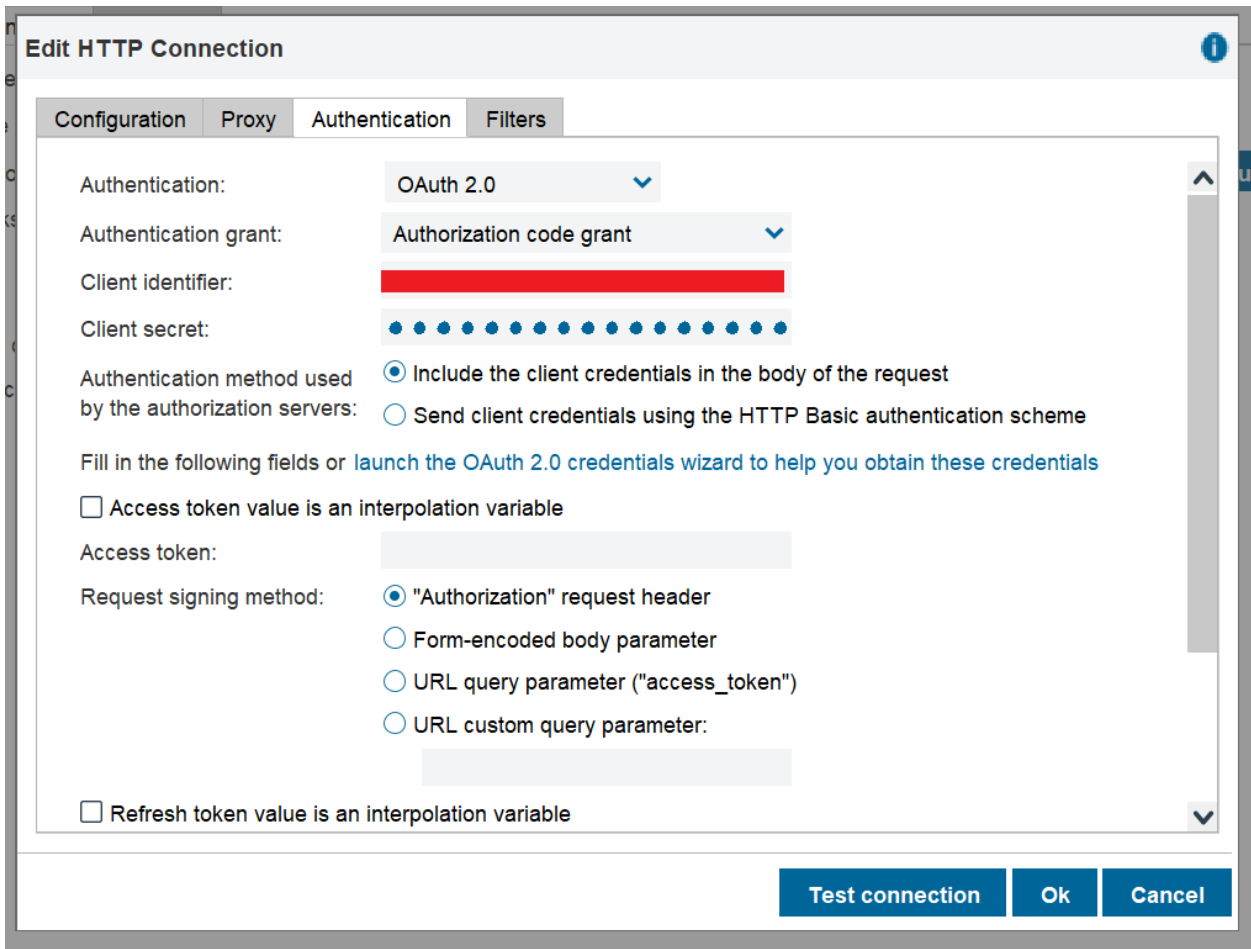
- From the Virtual DataPort Administration Tool or Design Studio, create an Excel Data Source.
- In the data source configuration, select the **File Location** as **HTTP Client** and select **Configure**.
- In the **Edit HTTP Configuration** dialog box set the HTTP Method to GET and provide the URL with the following format:

```
https://<mysite>.sharepoint.com/_api/Web/GetFileByServerRelativePath(decodedurl='<location of an Excel file in the sharepoint>')/$value
```



- In the Authentication section, select **OAuth 2.0** for Authentication.

- Paste the **Application (client) ID** from the newly created application to the **Client Identifier** and the **Value of the Client Secret** to the **Client Secret**.



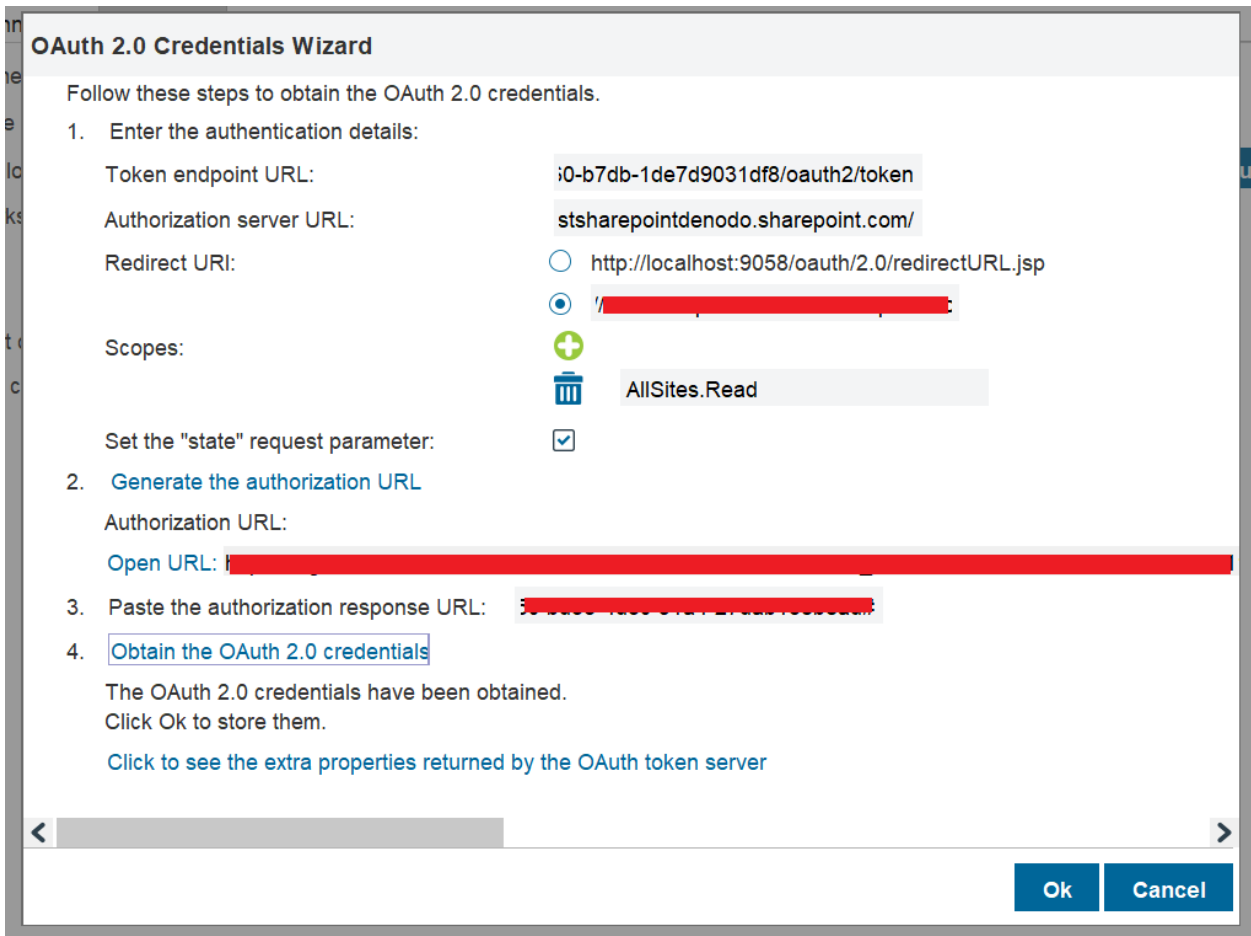
- Then select the **Launch the Oauth 2.0 credentials wizard**. The Oauth 2.0 credentials wizard will pop up.
- In this dialog, provide **EndPoint URL** from the Azure portal to the **Token endpoint URL** as **https://login.microsoftonline.com/common/oauth2/token**
- Provide the Authorization URL

<https://login.microsoftonline.com/common/oauth2/authorize?resource=https://<mysite>.sharepoint.com/>

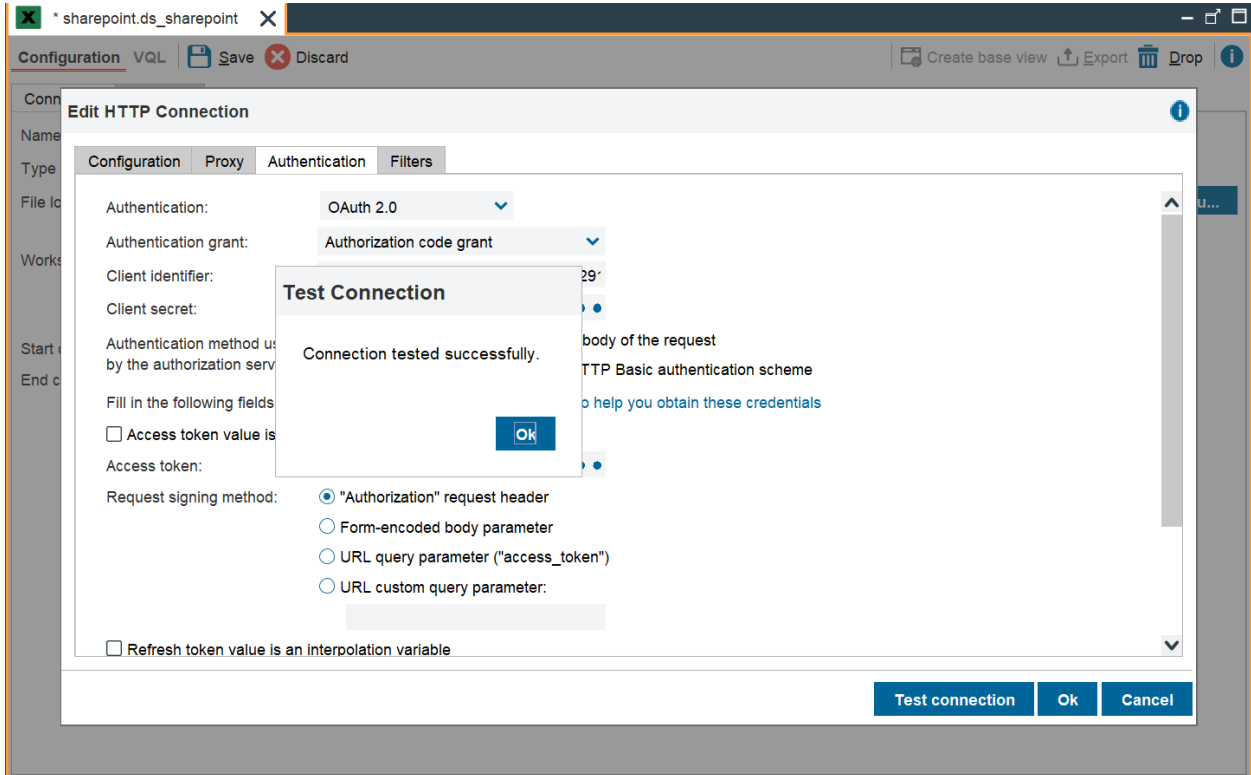
- Select the **Redirect URI**, provide the redirect URI that you used to register an application in the first step.
- For the **Scopes** option provide the value **AllSites.Read**.
- After providing all the URL's click on the option **Generate Authorization URL**.
- When the Authorization URL is generated, click on the **open URL**.
- Once the URL is opened on your desired browser, provide the permission, and once the permission is given a URL with code would be generated. The format of the URL will be:

```
https://<mysite>.sharepoint.com/?code=<>&state=<>&session_state=<>#
```

- Paste the same on the **Paste the authorized response URL** and select **Obtain the OAuth 2.0 credentials**.
- Select **Ok**.



Once the OAuth 2.0 Credentials are obtained , test the connection to check if the connection can be established.



Now you will be able to create a new Base view from this Excel data source.

sharepoint.ds_sharepoint | sharepoint.bv_sharepoint

Summary Edit Options VQL

Database: sharepoint

View type: Base

Schema:

Field Name	Field Type	Description
empno	int	
firstname	text	
midinit	text	
lastname	text	

Owner: admin Last modifier: admin

Creation: Jun 2, 2021 9:52:06 AM Last modification: Jun 2, 2021 9:58:43 AM

Swap status: default Cache status: off

Folder: /1-datasource

Description:

Execute Query Results

Results Execution Trace

Query: SELECT * FROM bv_sharepoint LIMIT 150 CONTEXT ('i18n='us_pst', 'cache_wait_for_load'=true) TRACE

Total rows received: 42 (shown 42)

empno	firstname	midinit	lastname	workdept	phoneno	hiredate	job	edlevel	sex	birthdate	salary	boi
10	CHRISTINE	I	HAAS	A00	3978	1/1/95	PRES	18	F	8/24/63	152750	100
20	MICHAEL	L	THOMPSON	B01	3476	10/10/03	MANAGER	18	M	2/2/78	94250	800
30	SALLY	A	KWAN	C01	4738	4/5/05	MANAGER	20	F	5/11/71	98250	800
50	JOHN	B	GEYER	E01	6789	8/17/79	MANAGER	16	M	9/15/55	80175	800
60	IRVING	F	STERN	D11	6423	9/14/03	MANAGER	16	M	7/7/75	72250	500
70	EVA	D	PULASKI	D21	7831	9/30/05	MANAGER	16	F	5/26/03	96170	700

7 REFERENCES

[Quickstart: Register an application with the Microsoft identity platform](#)

[Excel Sources](#)