



Lab 02: Governance

Introduction

In this lab, you will start by creating a new workspace, a Warehouse and Lakehouse. Then you will assign users to the various workspace roles and warehouse security features.

Objectives

After completing this lab, you will be better able to:

- Endorsements and Certifying items
- Data Lineage and Impact Analysis
- Scan Fabric Metadata using APIs using Power Shell

Estimated time to complete this lab

60 minutes

Lab Prerequisites

- A Fabric capacity or Fabric trial

Contents

Lab 02: Governance1

 Introduction1

 Objectives.....1

 Task 1: Enabling certification on your tenant.....2

 Task 2: Creating a new workspace.....3

 Task 3: Endorsement and certifying items.....5

 Taks 3.1 - Promoting an item5

 Taks 3.2 – Certifying an item7

 Task 4: Data Linage.....9

 Task 5 - View the Impact Analysis 11

 Task 6 – Scan Microsoft Fabric Metadata using API 13

 Task 6.1 – Incremental scans (Optional).....16

Task 1: Enabling certification on your tenant

In this task as a Fabric admin, you're responsible for enabling and setting up the certification process for your organization.

Upskilling on Microsoft Fabric Governance & Security

In the Admin portal, go to Tenant settings.

Under the **Export and sharing settings** section, expand the Certification section

Set the toggle to Enabled.

Admin portal

The screenshot shows the Microsoft Fabric Admin portal. On the left, the 'Tenant settings' menu item is highlighted. The main content area shows the 'Certification' section, which is expanded. The 'Certification' section has a toggle switch set to 'Enabled'. Below the toggle, there is a text input field for 'Specify URL for documentation page'. There are also radio buttons for 'Apply to:' with 'The entire organization' selected. At the bottom, there is an 'Apply' button and a 'Cancel' button.

You can share the URL of your organization's certification policy. This link will be used as the "Learn more" link in the certification section of the endorsement settings dialog. If you don't provide a link, users interested in certification will be directed to contact their Fabric administrator.

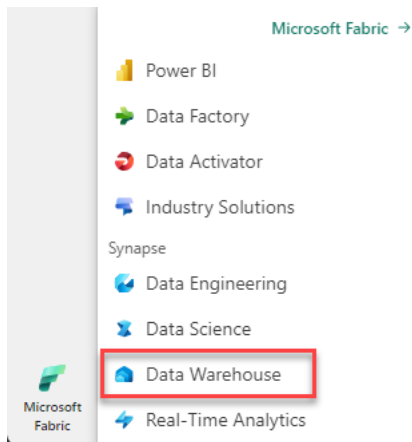
Select Apply.

Task 2: Creating a new workspace

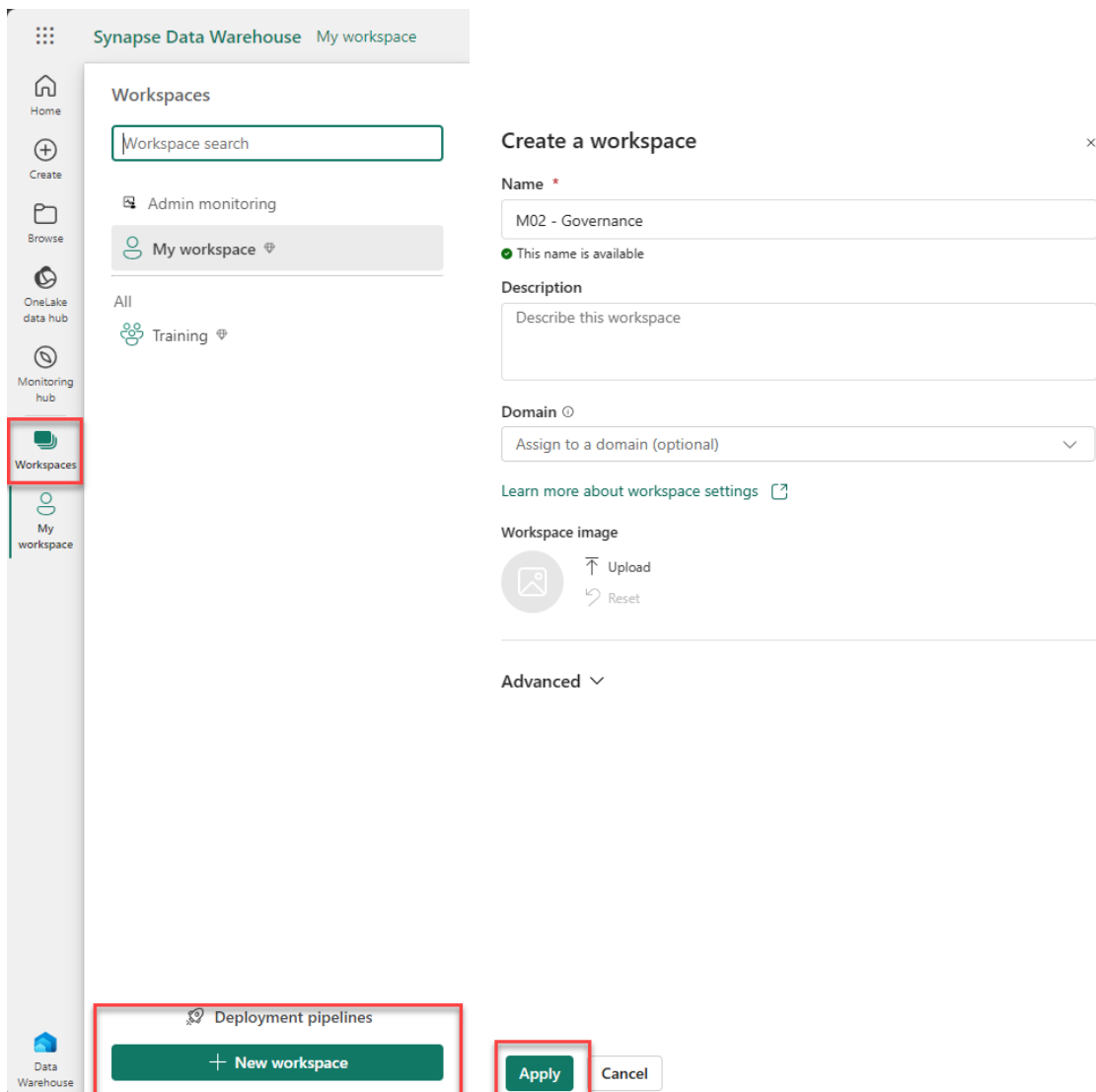
In this task, you will create a new workspace to be used in the remainder of the lab.

Connect to the Microsoft Fabric environment and switch to Data Warehouse experience

Upskilling on Microsoft Fabric Governance & Security



Now, select "Workspaces" in the left side menu and click "+ New workspace"



Provide the name as **M02 - Governance**.

Click apply.

Task 3: Endorsement and certifying items

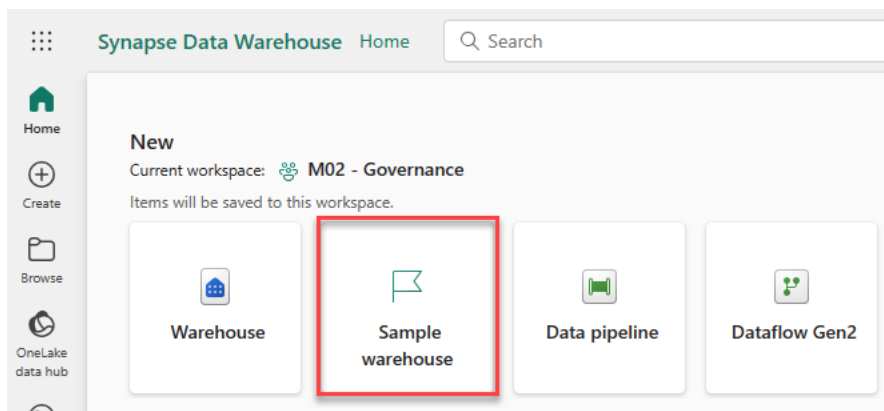
In this task, you will learn how the endorsement and certification features work in Microsoft Fabric and the tenant level settings that are needed for these features.

Let's create some items first.

In the **M02 - Governance** workspace, switch to Data Warehouse experience

In the data warehouse experience, create a new warehouse named: governance_wh

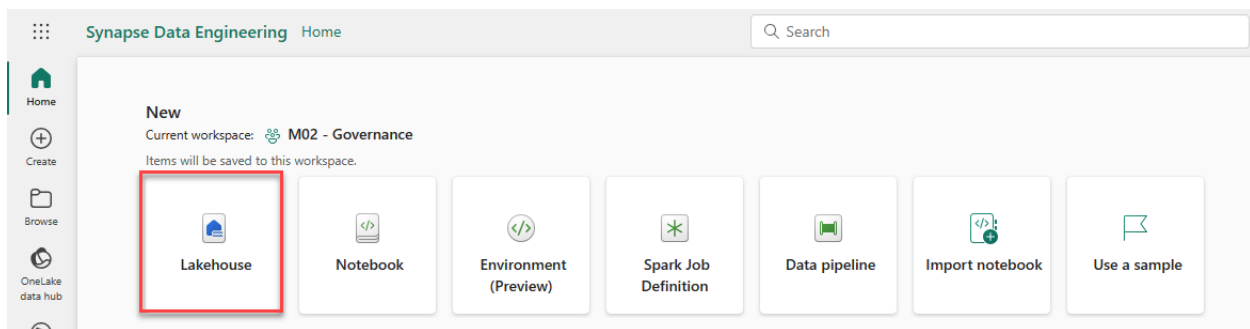
On the left side menu, click "Home", then click "Sample warehouse" as shown below



Provide a name for the new warehouse: governance_wh

In the **M02 - Governance** workspace, switch to Data Engineering experience

On the left side menu, click "Home", click in "Lakehouse" and provide a name: governance_lh



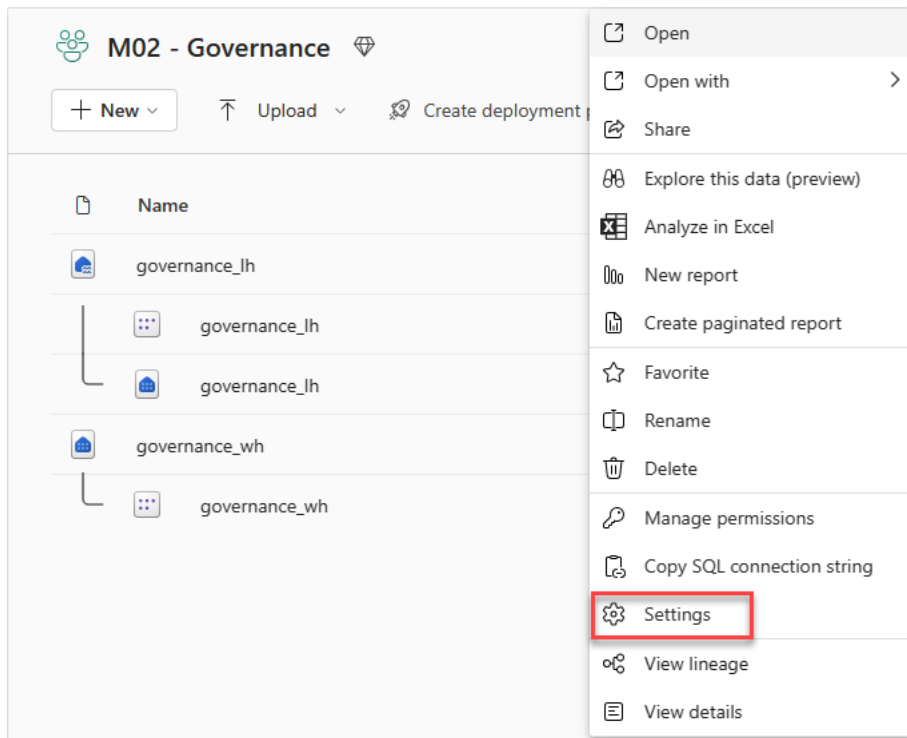
Taks 3.1 - Promoting an item

Any content owner or member with write permissions for the item can promote it when they believe it's ready for sharing.

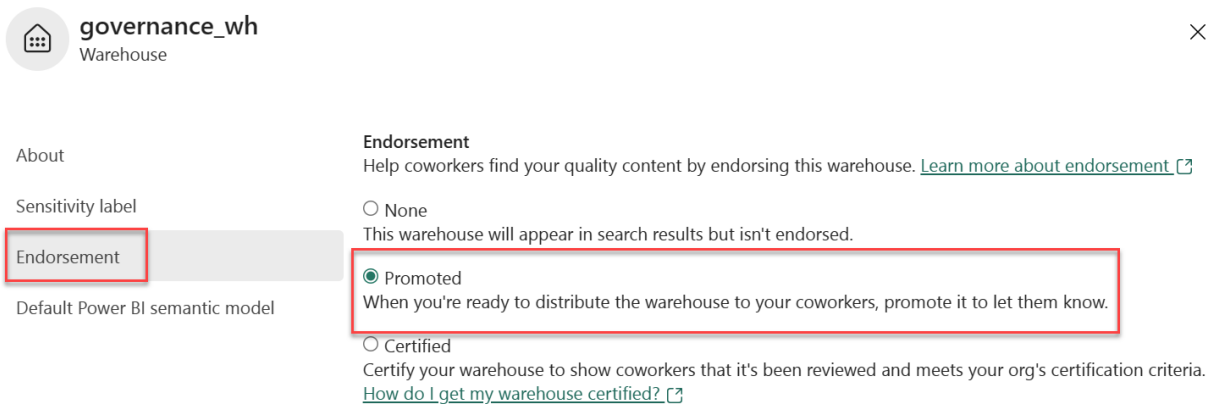
Upskilling on Microsoft Fabric Governance & Security

To promote an item, you need to have write permissions for that specific item.

In governance_wh warehouse, click the ellipsis and from the menu, click “Settings”



Click “Endorsement” and then promote it selecting the radio button “Promoted”.



Now, let's promote a semantic model.

In governance_lh lakehouse, click the ellipsis and from the menu, click “Settings”

Expand the “**Endorsement and discovery**”

Upskilling on Microsoft Fabric Governance & Security

The screenshot shows the 'Settings for governance_lh' page in the Microsoft Fabric Governance console. The 'Semantic models' tab is selected. In the left sidebar, 'governance_lh' and 'governance_wh' are listed. The main content area shows settings for 'governance_lh'. The 'Endorsement and discovery' section is highlighted with a red box. It includes options for 'None', 'Promoted' (selected), and 'Certified'. The 'Make discoverable' checkbox is also checked and highlighted with a red box. Below this, a yellow banner states: 'This dataset will be made discoverable. Others in your org will be able to find it by such details as name, tables, columns, etc. Learn more'. At the bottom, the 'Apply' button is highlighted with a red box.

Select the radio button “Promoted” to promoting the item.

If “**Make discoverable**” checkbox is selected, this implies that users who don't have access to the semantic model can discover it.

Go back to “**M02 – Governance**” workspace and you'll notice a tag has been added to those items.

The screenshot shows the 'M02 - Governance' workspace. A table lists several items, with two rows highlighted by red boxes to show the 'Promoted' endorsement status.

Name	Type	Owner	Refreshed	Next refresh	Endorsement	Sensitivity
governance_lh	SQL analytics end...	M02 - Governance	—	N/A	—	—
governance_lh	Semantic model (...)	M02 - Governance	3/4/24, 6:36:18 PM	N/A	Promoted	—
governance_lh	Lakehouse	System Administr...	—	—	—	—
governance_wh	Warehouse	System Administr...	3/4/24, 6:28:42 PM	N/A	Promoted	—
governance_wh	Semantic model (...)	M02 - Governance	3/4/24, 6:27:53 PM	N/A	—	—

Taks 3.2 – Certifying an item

Make sure you have completed task 1 before starting this task.

To certify an item, you need to have write permissions for that specific item.

In governance_lh warehouse, click the ellipsis and from the menu, click “Settings”

The screenshot shows the 'M02 - Governance' workspace in Microsoft Fabric. At the top, there are buttons for '+ New', 'Upload', 'Create deployment pipeline', and 'Create app'. Below is a table of Lakehouses:

Name	Type	Owner
governance_lh	Lakehouse	System Administr...
governance_lh		
governance_lh		
governance_wh		
governance_wh		

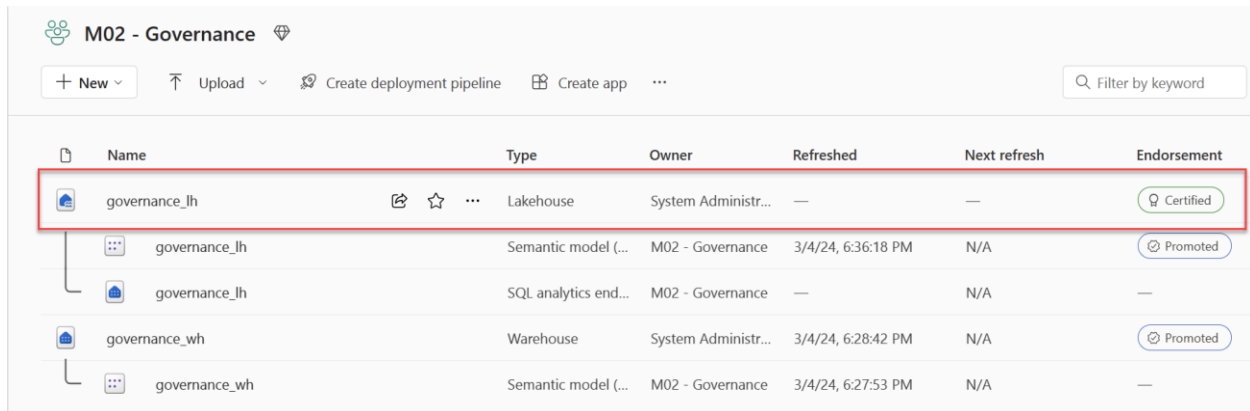
A context menu is open for the first 'governance_lh' Lakehouse. The menu options are: Open, Delete, Settings (highlighted with a red box), Add to Favorites, View lineage, View details, Manage permissions, Share, and Recent runs.

Click “Endorsement” and then promote it selecting the radio button “Certified”.

The screenshot shows the details page for the 'governance_lh' Lakehouse. On the left, there are tabs for 'About', 'Sensitivity label', and 'Endorsement' (highlighted with a red box). The 'Endorsement' section shows the following options:

- ☐ None: This Lakehouse is not endorsed.
- ☐ Promoted: This Lakehouse is recommended for others to use.
- ☒ Certified (Certified by [redacted] on March 5, 2024.)
This Lakehouse is certified by your org as a trusted source.
[How do I get content certified?](#)

Go back to “**M02 – Governance**” workspace and you'll notice a tag has been added to the Lakehouse.



Name	Type	Owner	Refreshed	Next refresh	Endorsement
governance_lh	Lakehouse	System Administr...	—	—	Certified
governance_lh	Semantic model (...)	M02 - Governance	3/4/24, 6:36:18 PM	N/A	Promoted
governance_lh	SQL analytics end...	M02 - Governance	—	N/A	—
governance_wh	Warehouse	System Administr...	3/4/24, 6:28:42 PM	N/A	Promoted
governance_wh	Semantic model (...)	M02 - Governance	3/4/24, 6:27:53 PM	N/A	—

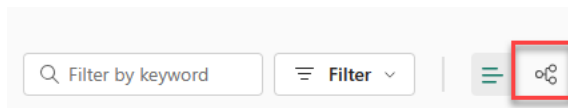
Task 4: Data Lineage

This task allows you to view the lineage between all the items in a workspace. With this feature, you can answer questions such as "What happens if I change this data?" or "Why isn't this report up to date?" These questions can be challenging to address without Fabric's lineage view.

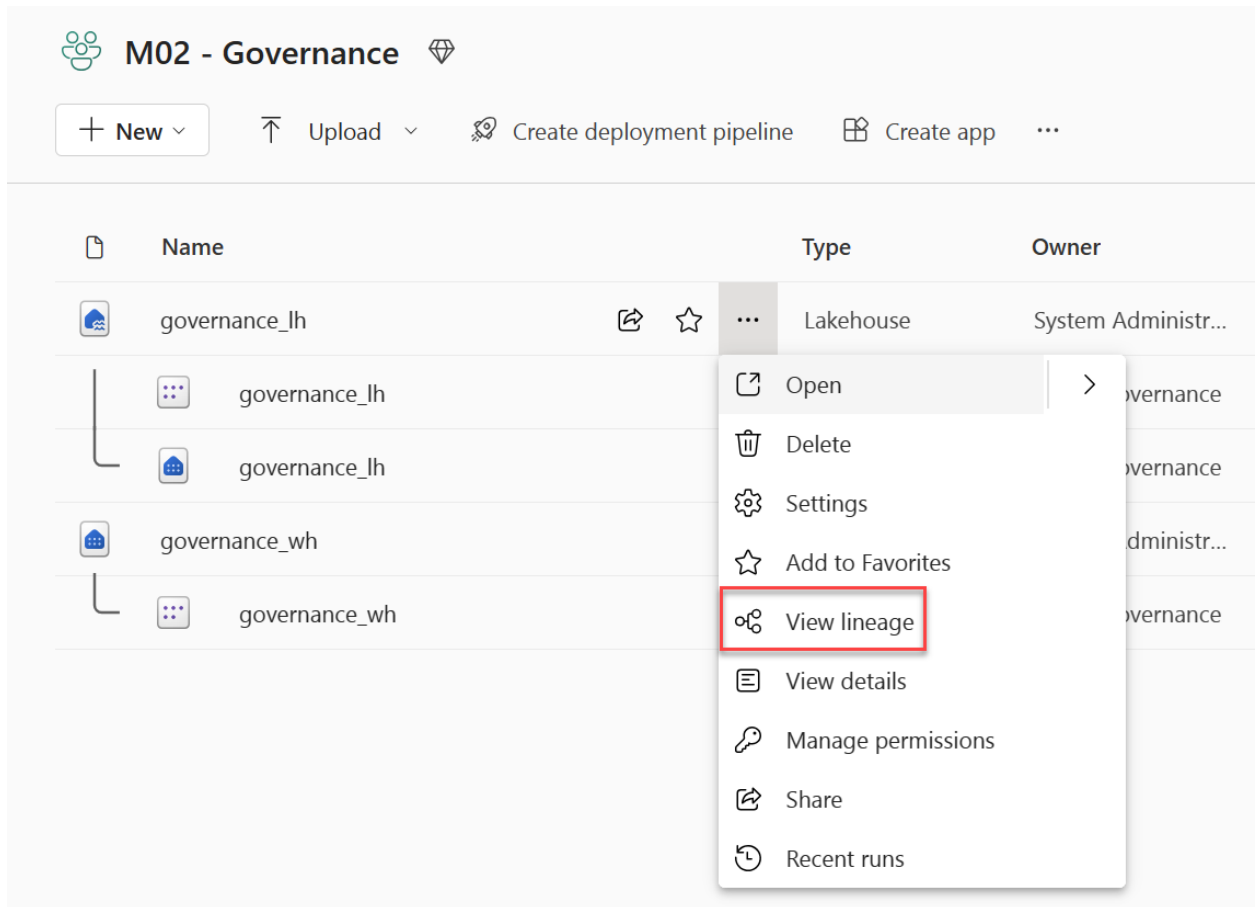
Before seeing the lineage, let's create a report using the default semantic model.

Lineage is accessible from multiple locations. Typically, you can get to it:

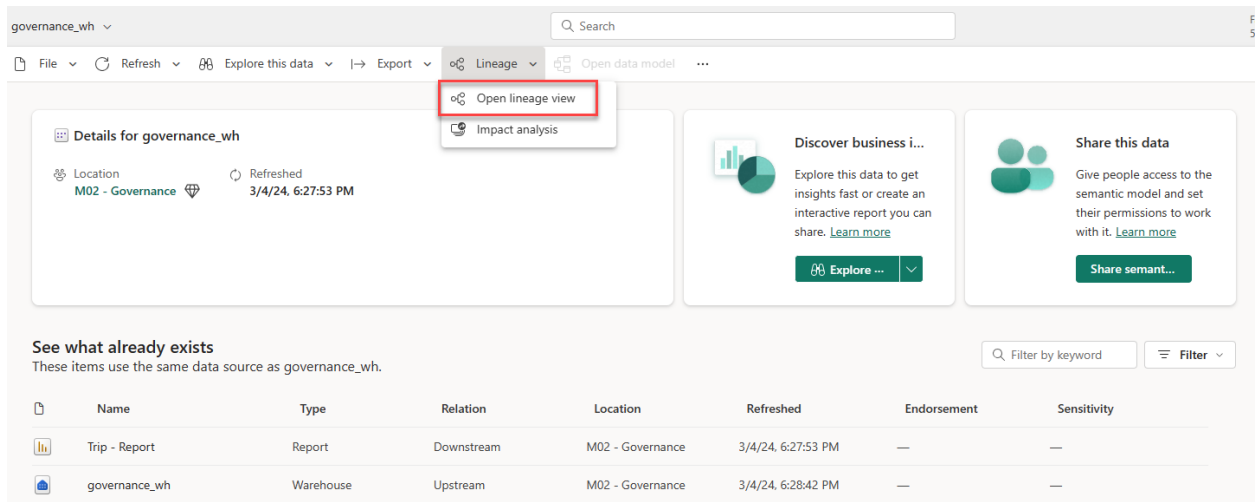
- From the workspace toolbar



- From an item's option menu

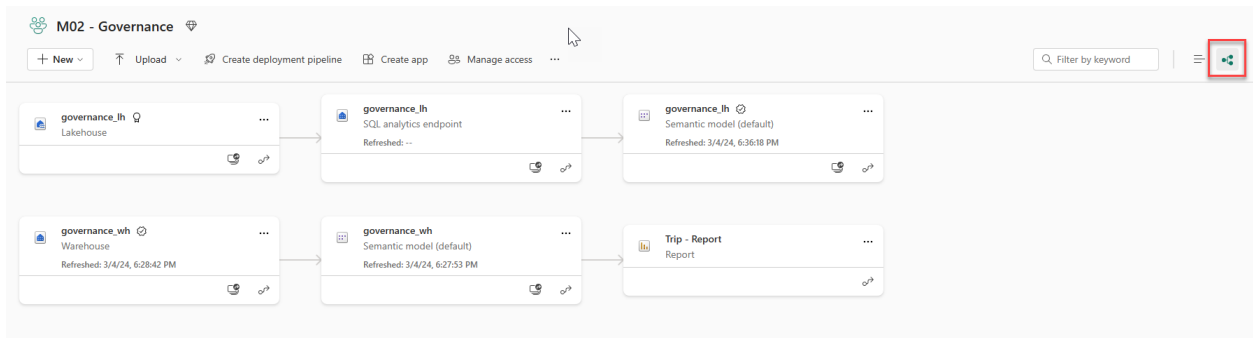


- From the menu items at the top of the item's details page

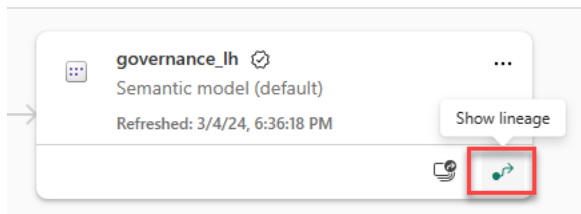


When you access the lineage view of an item, you'll observe the connections between all items within the workspace where the item is situated. The displayed view is a sample and may vary from the lineage view of your workspace.

Upskilling on Microsoft Fabric Governance & Security



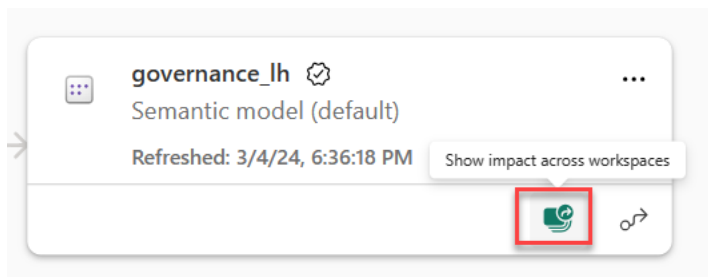
- Explore the lineage view to understand various tiles cards, use the “Show lineage” within a card to highlight lineage



- Explore the zoom and Full Screen feature that is available in the right hand bottom corner of the lineage view

Task 5 - View the Impact Analysis


To view the impact analysis, click “Show impact across workspaces” on the card



Examine the impact analysis to comprehend the potential effects the item may have on dependent items.

You have the option to switch between Child Items and All downstream items. Additionally, you can navigate by Item Type or by Workspace.


Impacted by this Lakehouse

 governance_lh

Child Items All downstream items


1


Impacted child items ⓘ


 1

Workspaces


Browse by workspace

⋮ 

✓  M02 - Governance

 governance_lh


Impacted by this Lakehouse

 governance_lh

Child Items **All downstream items**


2


Items impacted in total ⓘ


 1


Workspaces

Browse by item type



✓  SQL analytics endpoint 1

 governance_lh

>  Semantic model 1

Task 6 – Scan Microsoft Fabric Metadata using API

Metadata scanning facilitates governance of your organization's Microsoft Fabric data by making it possible to catalog and report on all the metadata of your organization's Fabric items. It accomplishes this using a set of Admin REST APIs that are collectively known as the scanner APIs.

In this task, you will use the REST APIs along with PowerShell (using MicrosoftPowerBIMgmt module) to scan and extract metadata for your workspace(s) using Power Shell ISE.

- Pre-requisites:
 - Windows PowerShell v3.0 and up with .NET 4.7.1 or above.
 - PowerShell Core (v6) and up on any OS platform supported by PowerShell Core.

Prior to scanning Fabric using API's there are two tenant settings control metadata scanning that need to be enabled:

- **Enhance admin APIs responses with detailed metadata:** This setting turns on Model caching and enhances API responses with low-level dataset metadata (for example, name and description) for tables, columns, and measures.
- **Enhance admin APIs responses with DAX and mashup expressions:** This setting allows the API response to include DAX expressions and Mashup queries. This setting can only be enabled if the first setting is also enabled.

To enable these settings, go to Admin portal > Tenant settings > Admin API settings

Enhance admin APIs responses with detailed metadata
Enabled for the entire organization

Users and service principals allowed to call Power BI admin APIs may get detailed metadata about Power BI items. For example, responses from GetScanResult APIs will contain the names of dataset tables and columns. [Learn More](#)

Note: For this setting to apply to service principals, make sure the tenant setting allowing service principals to use read-only admin APIs is enabled. [Learn More](#)

☒ Enabled

Apply to:

☒ The entire organization

☐ Specific security groups

☐ Except specific security groups

Apply Cancel

Upskilling on Microsoft Fabric Governance & Security

Enhance admin APIs responses with DAX and mashup expressions

Enabled for the entire organization

Users and service principals eligible to call Power BI admin APIs will get detailed metadata about queries and expressions comprising Power BI items. For example, responses from GetScanResult API will contain DAX and mashup expressions. [Learn More](#)

Note: For this setting to apply to service principals, make sure the tenant setting allowing service principals to use read-only admin APIs is enabled. [Learn More](#)

☒ Enabled

Apply to:

☒ The entire organization

☐ Specific security groups

☐ Except specific security groups

Apply

Cancel

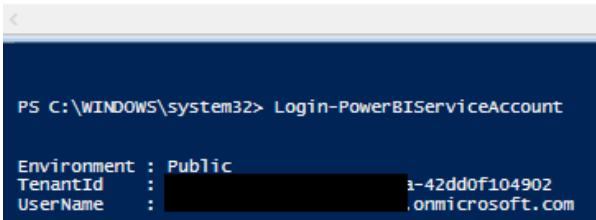
Follow the instructions below and run the commands.

1. Open Power Shell ISE ("Run as Administrator" is optional)
2. Install the MicrosoftPowerBIMgmt module using the below commands

```
#Install MicrosoftPowerBIMgmt module
Install-Module -Name MicrosoftPowerBIMgmt

#Login
Login-PowerBIServiceAccount
```

```
1 #Install MicrosoftPowerBIMgmt module
2 Install-Module -Name MicrosoftPowerBIMgmt
3
4 #Login
5 Login-PowerBIServiceAccount
6
```



```
PS C:\WINDOWS\system32> Login-PowerBIServiceAccount

Environment : Public
TenantId    : i-42dd0f104902
UserName    : [redacted]@onmicrosoft.com
```

3. Build URL to exclude Personal Workspaces and get the list of workspaces

```
#Build URL to exclude Personal workspaces
$url =
"https://api.powerbi.com/v1.0/myorg/admin/workspaces/modified?excludePersonalWorkspaces=True"
#Get list of workspaces
Invoke-PowerBIRestMethod -url $url -Method Get | ConvertFrom-Json
```

```

7 #Build URL to exclude Personal Workspaces
8 $url = "https://api.powerbi.com/v1.0/myorg/admin/workspaces/modified?excludePersonalWorkspaces=True"
9 #Get list of workspaces
10 Invoke-PowerBIRestMethod -Url $url -Method Get | ConvertFrom-Json
11
PS C:\WINDOWS\system32> #Build URL to exclude Personal Workspaces
$url = "https://api.powerbi.com/v1.0/myorg/admin/workspaces/modified?excludePersonalWorkspaces=True"
#Get list of workspaces
Invoke-PowerBIRestMethod -Url $url -Method Get | ConvertFrom-Json

id
--
-055df2a28f41
-301d2196a90e
-2173cf098ad5
-464fce210465
-2c09621fb2d5
-b9883eb5b2eb
-4f1322852c45
-ff178c0630a5
-af09f2138c65
-4f91803d3c6d
-e84ac71aee12
-16261dedf86c
-0ad9fbcc75d1
-da6cf01e83af
-dc9da8e8e8fd
-6ed78f3acba1
-bd2ec9b59e2b

```

4. Build URL to start a scan against a workspace or set of workspaces. Replace the workspace ID in the \$body variable from the result of the above command. You can use one workspace ID or multiple separated by commas

```

#Build URL to start a scan against a workspace or set of workspaces. Replace
the workspace ID in the $body variable from the result of the above command
$url1 = "https://api.powerbi.com/v1.0/myorg/admin/workspaces/getInfo"
$body = @"
{
  "workspaces": [
    "xxxxxxxx-xxxx-xxxx-xxxxxxxxxxxx", "xxxxxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxxxx"
  ]
}
"@

```

#Trigger a scan

```
Invoke-PowerBIRestMethod -Url $url1 -Body $body -Method Post | ConvertFrom-Json
```

```

12 #Build URL to start a scan against a workspace or set of workspaces. Replace the workspace ID in the $body variable from the result of the above command
13 $url1 = "https://api.powerbi.com/v1.0/myorg/admin/workspaces/getInfo"
14 $body = @"
15 {
16   "workspaces": [
17     "xxxxxxxx-xxxx-xxxx-xxxxxxxxxxxx"
18   ]
19 }
20 "@
21
22 #Trigger a scan
23 Invoke-PowerBIRestMethod -Url $url1 -Body $body -Method Post | ConvertFrom-Json
24
PS C:\WINDOWS\system32> #Build URL to start a scan against a workspace or set of workspaces. Replace the workspace ID in the $body variable from the result of the above command
$url1 = "https://api.powerbi.com/v1.0/myorg/admin/workspaces/getInfo"
$body = @"
{
  "workspaces": [
    "xxxxxxxx-xxxx-xxxx-xxxxxxxxxxxx"
  ]
}
"@
#Trigger a scan
Invoke-PowerBIRestMethod -Url $url1 -Body $body -Method Post | ConvertFrom-Json

id                      createdDateTime          status
--                      -
ad844b03-bb07-490b-8db8-132978b99334 2023-11-03T07:58:51.6778162Z NotStarted

```

Notice from the above output that we submitted a scan and got the id, CreateDateTime in the output. Copy this output as we will use these in the later steps. Also note that the status is reported was "NotStarted". Meaning the scanning is yet to start

5. We will poll using the scan ID that we got from the above step to check the scan status

```
$poll =  
"https://api.powerbi.com/v1.0/myorg/admin/workspaces/scanStatus/<workspace id>"  
Invoke-PowerBIRestMethod -Url $poll -Method Get | ConvertFrom-Json
```

```
25 #Poll to check if the scan shows status as Succeeded
26 $poll = "https://api.powerbi.com/v1.0/myorg/admin/workspaces/scanStatus/50d92da8-1cee-4c4d-a3a8-8ffa365acc7d"
27 Invoke-PowerBIRestMethod -Uri $poll -Method Get | ConvertFrom-Json
28
```

```
PS C:\WINDOWS\system32> $poll = "https://api.powerbi.com/v1.0/myorg/admin/workspaces/scanStatus/50d92da8-1cee-4c4d-a3a8-8ffa365acc7d"
Invoke-PowerBIRestMethod -Uri $poll -Method Get | ConvertFrom-Json
```

id	createdDateTime	status
50d92da8-1cee-4c4d-a3a8-8ffa365acc7d	2023-11-03T08:39:44.993	Succeeded

6. When the scan status shows as Succeeded, it means that the scan completed and we can consume the results of the scan in the next step
7. Get the result of the scan

```
$scanresult =  
"https://api.powerbi.com/v1.0/myorg/admin/workspaces/scanResult/50d92da8-1cee-  
4c4d-a3a8-8ffa365acc7d"  
Invoke-PowerBIRestMethod -Url $scanresult -Method Get
```

```
29 # Get the result of the scan
30 $scanresult = "https://api.powerbi.com/v1.0/myorg/admin/workspaces/$scanResult/Sd93da8-icce-4c4d-a3ab-affa365acc0d"
31 Invoke-PowerBIRestMethod -Uri $scanresult -Method Get
32
33
34 PS C:\Windows\system32> # Get the result of the scan
35 $scanresult = "https://api.powerbi.com/v1.0/myorg/admin/workspaces/$scanResult/Sd93da8-icce-4c4d-a3ab-affa365acc0d"
36 Invoke-PowerBIRestMethod -Uri $scanresult -Method Get
37 {
38   "workspaces": [
39     {
40       "id": "Sd93da8-icce-4c4d-a3ab-affa365acc0d",
41       "name": "PowerBI Desktop",
42       "type": "Workspace",
43       "state": "Active",
44       "isoredacted": false,
45       "capacity": true,
46       "capacityId": null,
47       "description": null,
48       "tags": "ACTIVE",
49       "lastModifiedDate": "2023-08-08T13:36:41Z",
50       "createdDate": "2023-08-08T13:36:41Z",
51       "defaultDatasetStorageFormat": "Small",
52       "eventstream": "[{"id": "9316ad7b-b648-4c4d"}]"
53     }
54   ]
55 }
```

The output above has been clipped but, this output can be consumed by various tools for cataloging

Task 6.1 – Incremental scans (Optional)

Incremental scans can be performed by using `createdDateTIme` from the output of the command that triggered the scan in step 4 above (the output of which you should have saved). But prior to that, you need to make some changes (like adding new items) to the workspace(s) you would like to perform an incremental scan and wait 30 minutes. Then run the below:

```
#Build URL to exclude look for modified workspaces and also exclude Personal
workspaces
$WorkspacesModifiedSince =
"https://api.powerbi.com/v1.0/myorg/admin/workspaces/modified?modifiedSince=<DateCreated value from step 3>&excludePersonalworkspaces=True"
Invoke-PowerBIRestMethod -Url $WorkspacesModifiedSince -Method Get |
ConvertFrom-Json
```

```
36 #Build URL to exclude look for modified workspaces and also exclude Personal Workspaces
37 $workspacesmodifiedSince = "https://api.powerbi.com/v1.0/myorg/admin/workspaces/modified?modifiedSince=2023-11-01T15:37:35.0000000Z&excludePersonalWorkspaces=True"
38 Invoke-PowerBIRestMethod -Url $workspacesmodifiedSince -Method Get | ConvertFrom-Json
```

```
PS C:\WINDOWS\system32> #Build URL to exclude look for modified workspaces and also exclude Personal Workspaces
$workspacesModifiedSince = "https://api.powerbi.com/v1.0/myorg/admin/workspaces/modified?modifiedSince=2023-11-01T15:37:35.0000000Z&excludePersonalWorkspaces=True"
Invoke-PowerBIRestMethod -Url $workspacesModifiedSince -Method Get | ConvertFrom-Json

id
--
b98a3eb5b2eb
dc9da8e8e8fd
6ed78f3acba1
bd2ec9b59e2b
4f921803d3ced
8f43ac718ee12
```


The output would provide a list of workspaces that changed since the last full scan. You can then use the workspaces in this list to trigger a new scan (starting from step3).

This is the end of the lab. Congratulations for finishing the lab!