Open Edu Analytics (OEA) Solution Guide

Published: August, 2021

Introduction 1

1) Setup of base OEA architecture 2

2) Walking through the included example 5

3) More about Synapse Studio 6

4) Power BI dashboard examples 7

5) Connect Power BI workspace 9

6) Privacy and Security 10

Introduction

This document provides step by step instructions for the setup of the OEA reference architecture as well as information on how to deploy modules and packages on the base architecture.

OEA is an open source modern data estate solution for education, built on Synapse Analytics and the powerful set of Azure platform data services.
For a set of brief introductory videos on Synapse Analytics see: Azure Synapse Analytics demo videos.
For a step-by-step guide through Synapse Analytics, see: Get Started with Azure Synapse Analytics.
For a detailed e-book on analytics in Azure, see: Cloud Analytics with Microsoft Azure

All scripts and documentation for OEA can be found at: https://github.com/microsoft/OpenEduAnalytics

The OEA framework is comprised of:

1) The core OEA framework architecture – an Azure storage account, a Synapse workspace, an Application Insights instance, an Azure Key Vault instance, an Apache Spark Pool, and a set of AAD security groups
2) modules – Apache Spark notebooks for the processing of source data from a specific source system. Data modules can be seen as data silos, bringing in data from a single system, with no dependencies.
3) packages – a package of assets such as Apache Spark notebooks for provisioning a comprehensive view over multiple data sets, Power BI reports, and Machine Learning models. A solution package utilizes one or more data modules for providing the source data utilized in the solution.

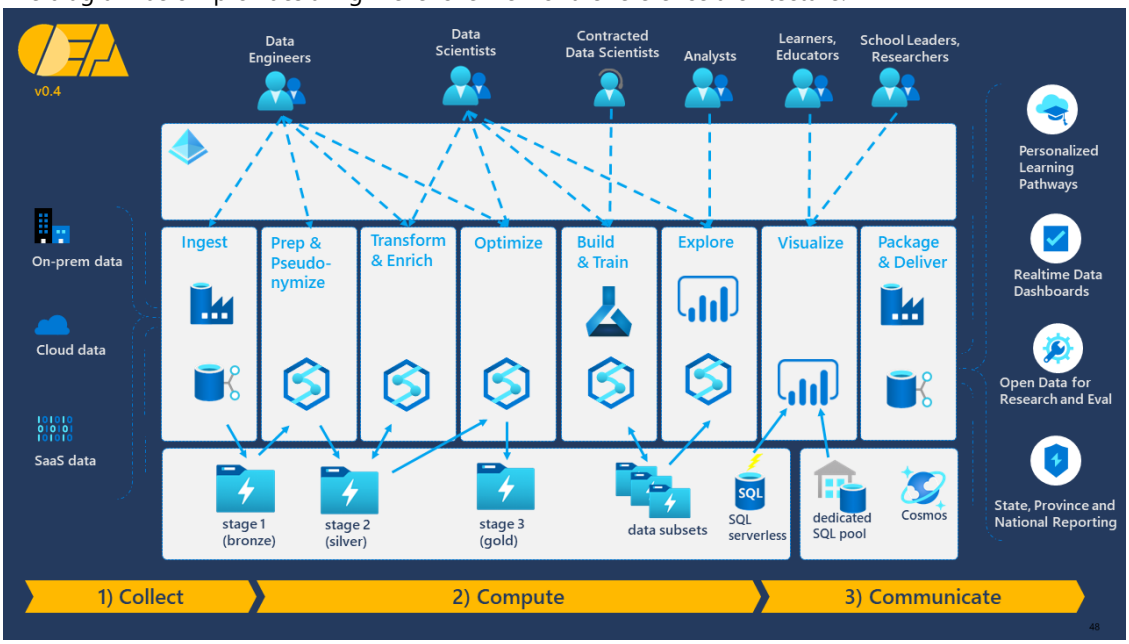
Modules and packages in OEA can contain the same set of assets – the main distinction between the two is that modules are self-contained while packages have dependencies on one or more modules.

Modules and packages have the following standard structure:

1. a readme.md for basic documentation
2. a setup.sh script to be used for automated deployment from cloud shell
3. a notebook folder for Synapse notebooks
4. a powerbi folder for Power BI assets (this is optional)

In order to begin the setup of the OEA solution, all you need is an Azure subscription. See the following section for detailed setup instructions.

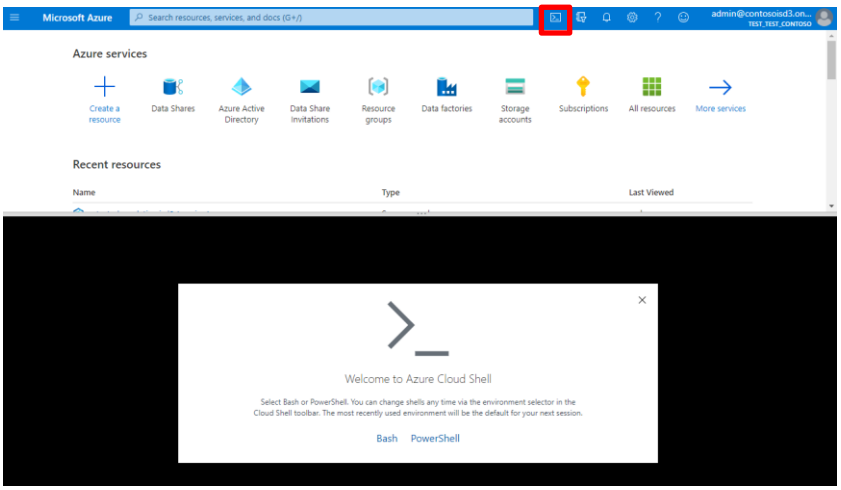
The diagram below provides a high-level overview of the reference architecture.



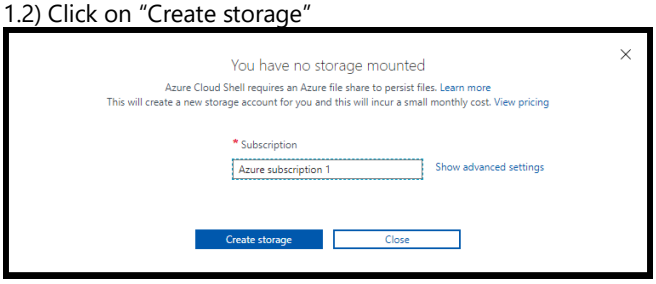
1) Setup of base OEA architecture

In this section you will use a script to provision the Azure resources that comprise the core of this solution, as well as an example solution package that provides example datasets and notebooks to use for further exploring the capabilities of Synapse Analytics.

1.1) In Azure portal, click on the Cloud Shell icon, then select “Bash”.

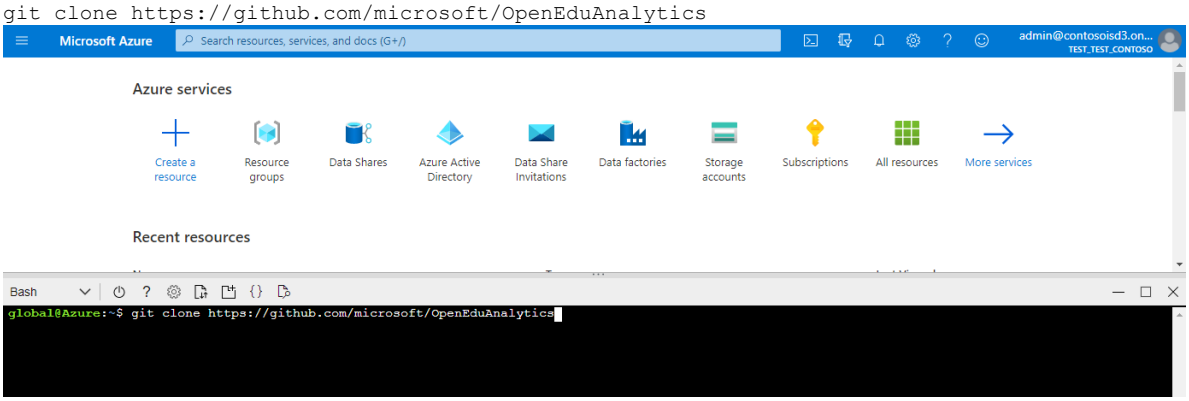


1.2) Click on “Create storage”



1.3) At the bash shell prompt, enter the following commands to download the contents of the OpenEduAnalytics repository to your Azure cloud drive.

```
cd clouddrive
git clone https://github.com/microsoft/OpenEduAnalytics
```



1.4) Now run the setup script found in the root directory of OpenEduAnalytics by running the following commands. Note that for <unique_suffix> in the command below, you should enter an ID for your org which will be used as a suffix of Azure resources that must be unique. For example, a school district named Contoso Independent School District might choose an org ID of “CISD” or “ContosoISD”.

```
cd OpenEduAnalytics
./setup.sh <unique_suffix>
```

The Azure resources will be created in the East US region by default. In order to have the resources created in a different location, specify the desired location as the second argument to the script:

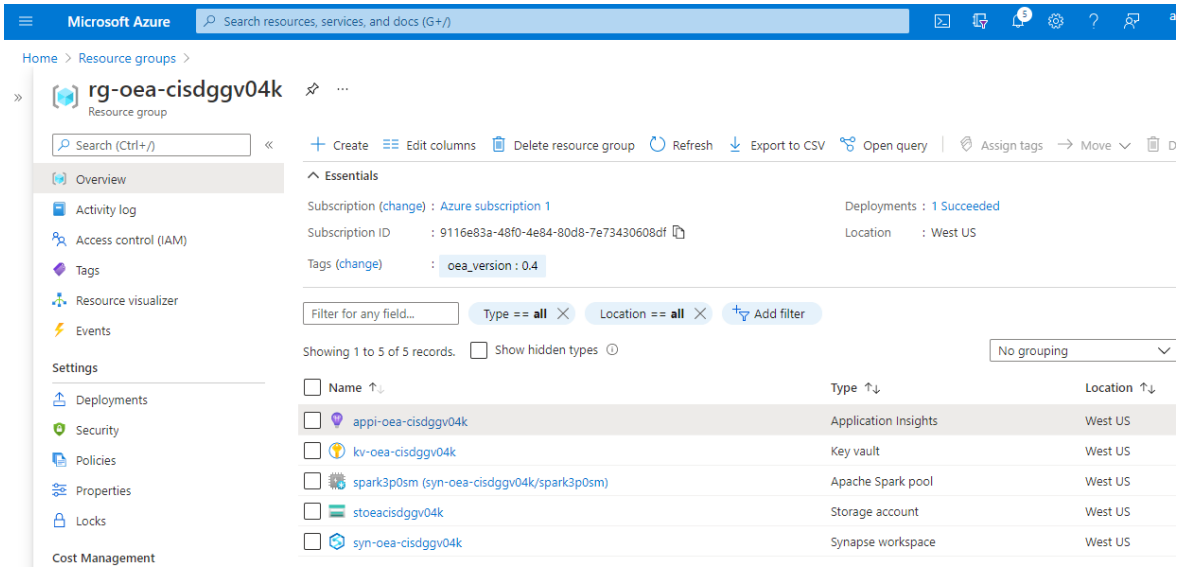
```
./setup.sh <unique_suffix> <location>
```

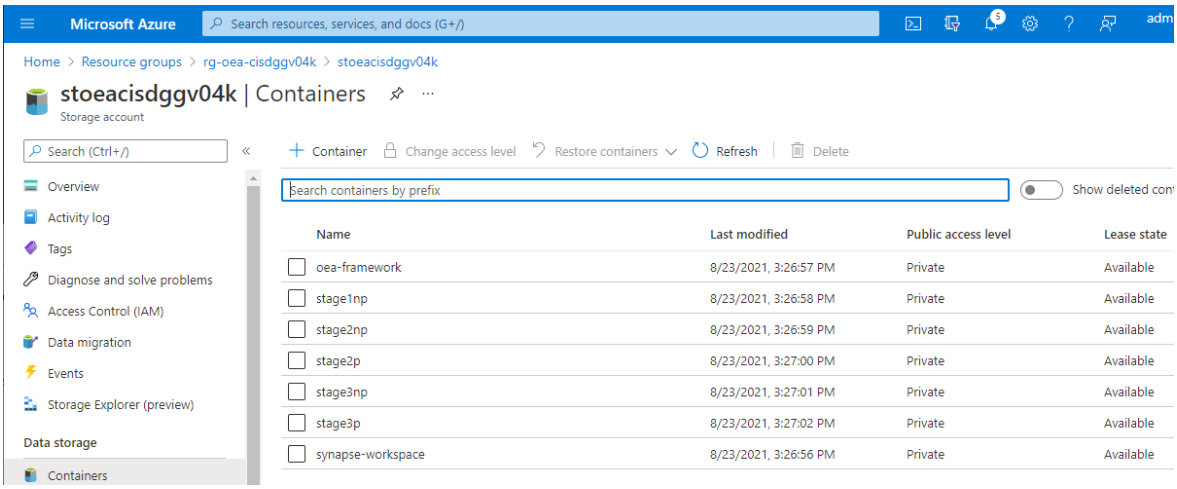
For a list of available locations, run this command on your bash command line: `az account list-locations`

The installation script will then take several minutes to complete, as it provisions the following Azure resources:

1. a resource group (which serves as a logical container for the rest of the resources created)
2. a storage account with 7 storage containers (stage1np, stage2np, stage2p, stage3np, stage3p, oea-framework, synapse)
3. an Azure Synapse workspace
4. an Apache Spark pool
5. a key vault
6. an Application Insights instance

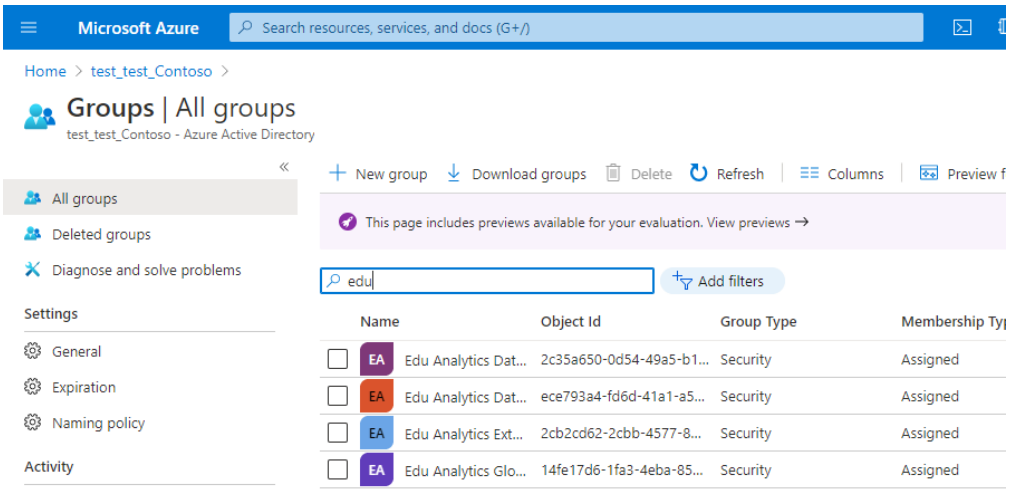
Pictured below are screenshots of the created resources:





Note too that the automated setup installs test datasets in the storage container named oea-framework. See the section "Walking through the example" for more details on how to use this data to run example notebooks and learn more about Synapse Analytics.
You can also choose to have the script create security groups to facilitate the use of role based access control to the data lake.
If you are running the setup for an environment in which you have Global Admin permissions on the tenant, and you want to have security groups provisioned, you can invoke the setup script like this:

```
./setup.sh <unique_suffix> <location> true
```



Debugging

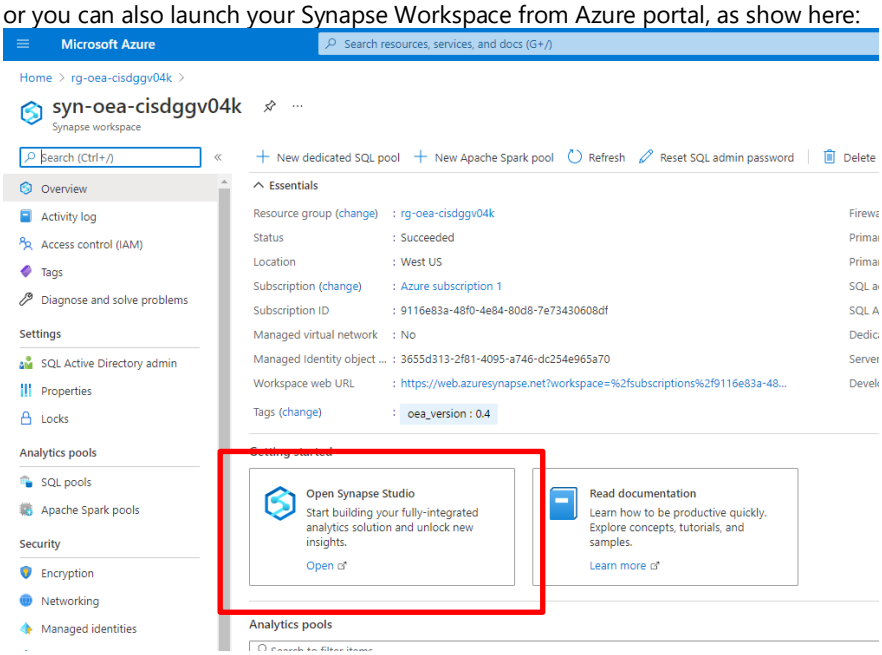
Refer to the OEA wiki for the most up to date debugging info: [Debugging issues in OEA]

2) Walking through the included example

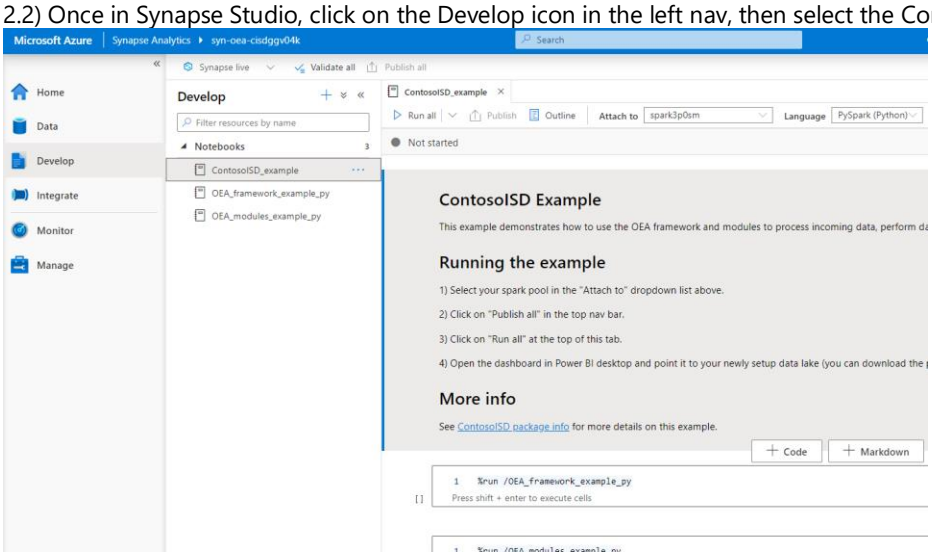
2.1) Open your new Synapse Workspace by clicking on the url at the end of the setup script

```
--> Setting up OEA (logging detailed setup messages to oea_setup_20210823_192620.log)
--> Setting up the OEA base architecture.
--> 1) Creating resource group: rg-oea-cisdggv04k
--> 2) Creating storage account: stoeacisdggv04k
--> 3) Creating Synapse Workspace: syn-oea-cisdggv04k (this is usually the longest step - it may take 5 to 10 minutes to complete)
--> 4) Creating key vault: kv-oea-cisdggv04k
--> 5) Creating security groups in Azure Active Directory.
--> Setting up the example OEA package.
--> OEA setup is complete. Click on this url to work with your new Synapse workspace (via Synapse Studio): https://web.azuresynapse.net?work
Synapse%2fworkspaces%2fsyn-oea-cisdggv04k
    Once in Synapse Studio, click on Develop, select the notebook called ContosoISD_example, and follow the directions shown there.
```

or you can also launch your Synapse Workspace from Azure portal, as show here:



2.2) Once in Synapse Studio, click on the Develop icon in the left nav, then select the ContosoISD_example notebook



2.3) Now follow the directions in that notebook to land test data into your data lake, process that data, and view the data in desktop Power BI.

3) More about Synapse Studio

OEA is built to leverage the power of Azure Synapse, and the central tool for working in Azure Synapse is Synapse Studio.

Here's a brief online lesson on the basics of Synapse Studio: Explore Azure Synapse Studio

1100 XP

Explore Azure Synapse Studio

43 min • Module • 10 Units

4.6 (194) Rate it

Beginner | Data Engineer | Synapse Analytics

Take a tour of the core application used to interact with the various components of Azure Synapse Analytics.

Learning objectives

In this module, you will:

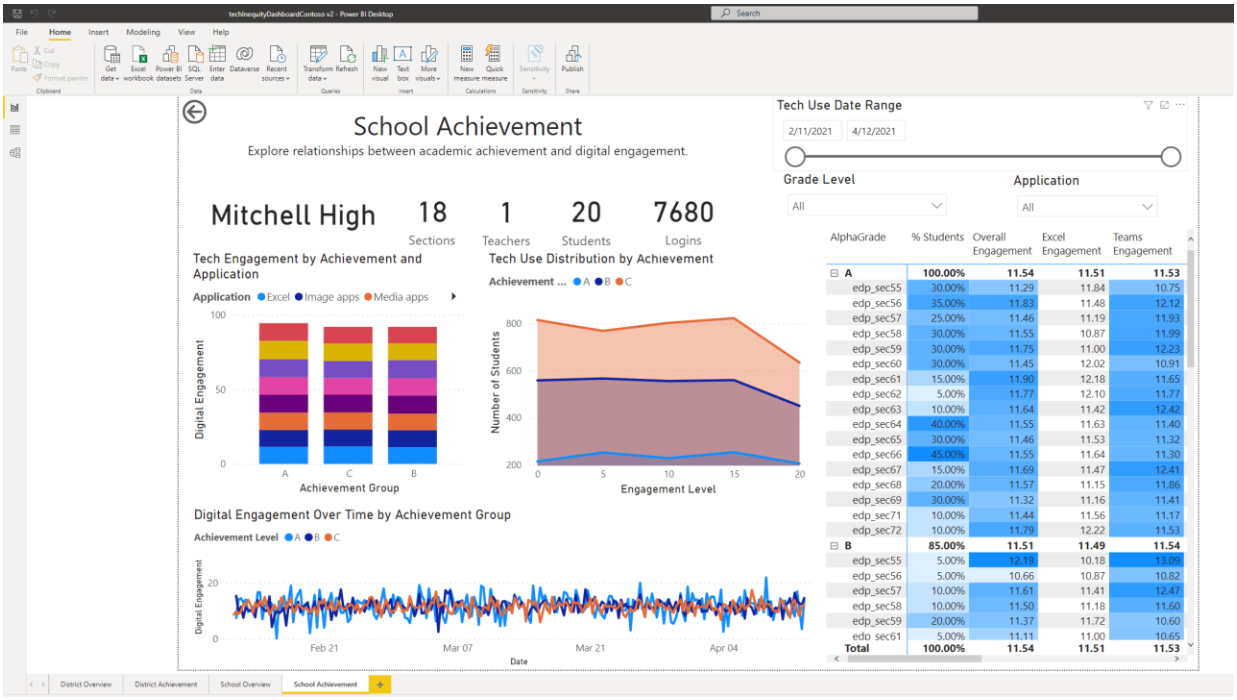
- Use Azure Synapse Studio
- Understand the Azure Synapse Analytical processes
- Explore the Data hub
- Explore the Develop hub
- Explore the Integrate hub
- Explore the Monitor hub
- Explore the Manage hub

4) Power BI dashboard examples

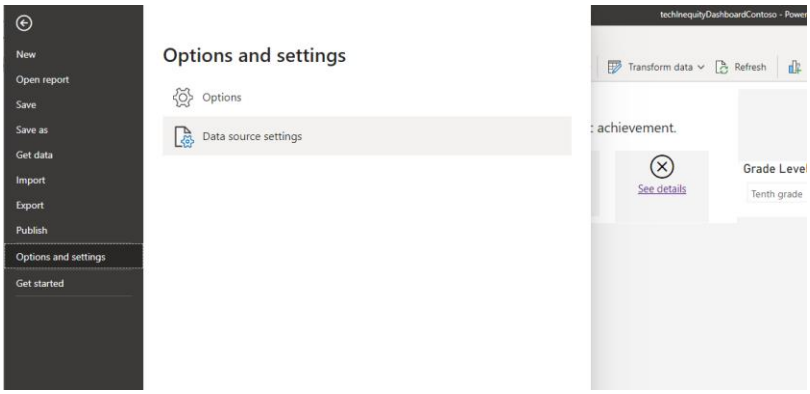
The previous section demonstrated the steps needed for a complete setup with a test environment and test data.

This section will demonstrate how to open the example Power BI dashboards in Power BI desktop and connect to the data lake in your test environment via SQL On-Demand. You will need to have Power BI Desktop installed on your computer to complete this section (Power BI Desktop is free to download and free to use – it can be downloaded from here).

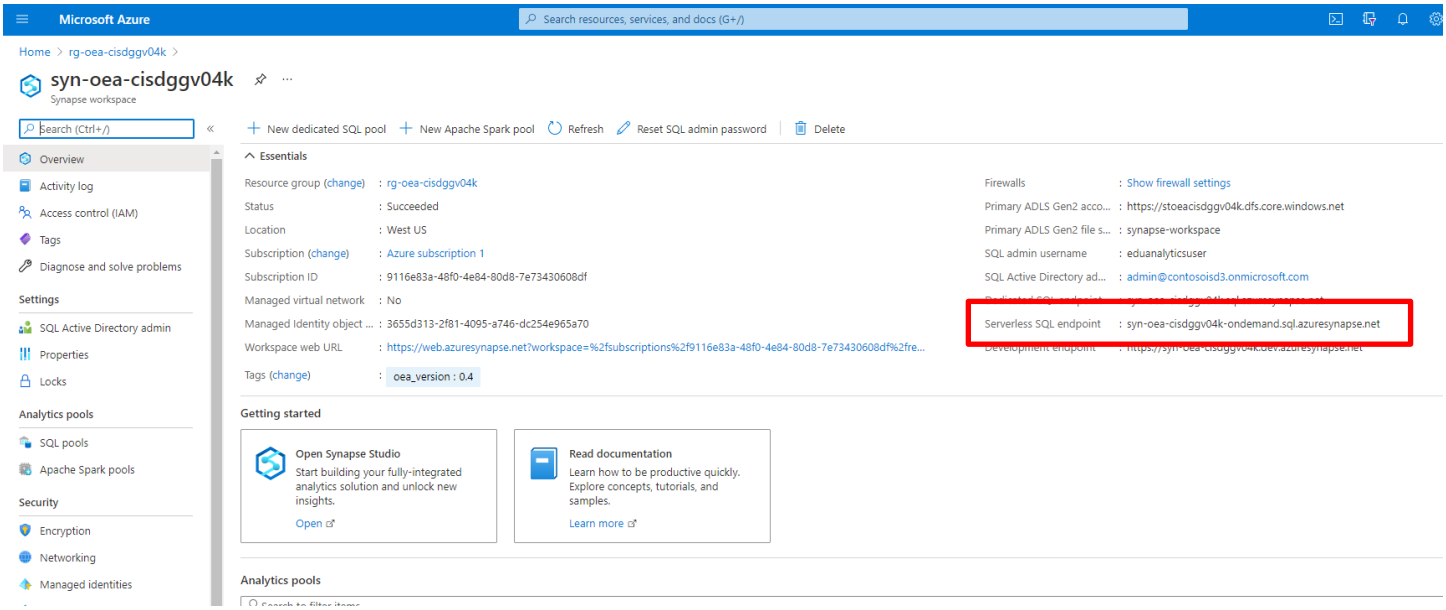
3.1) Download and then open this pbix file from the OEA github repo: techInequityDashboardContoso v2.pbix



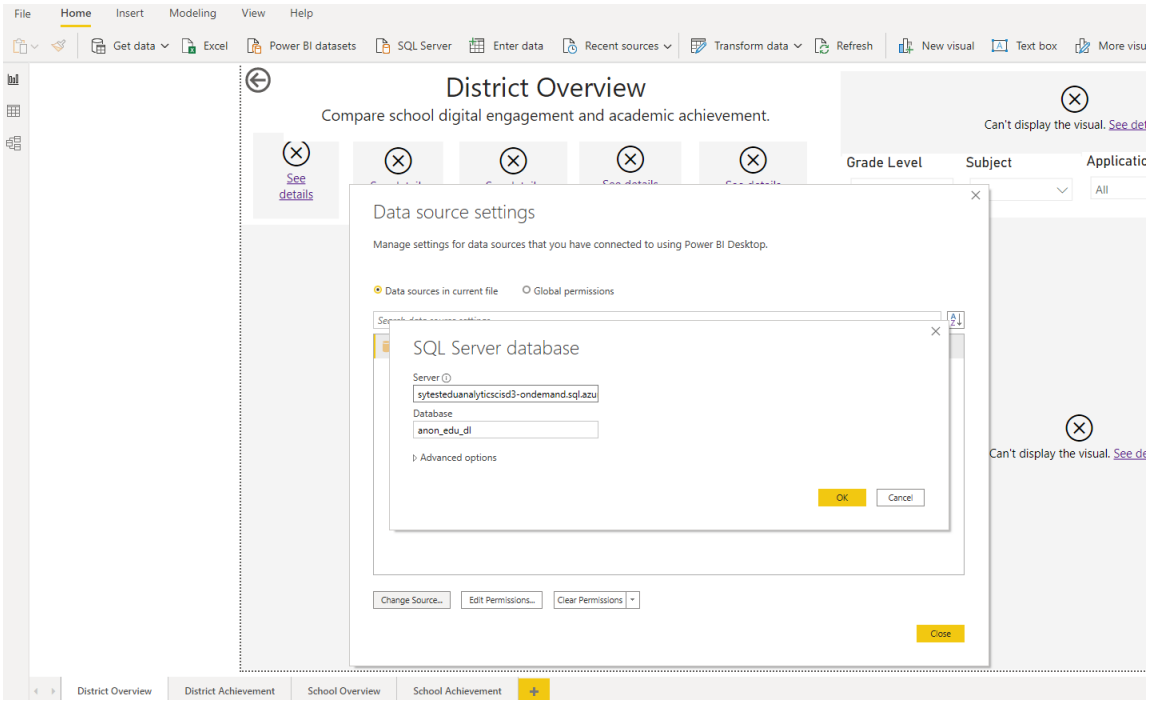
3.2) The data you see was included with the report. Now click on File -> Options and Settings -> Data source settings, and on the next screen click on "Change Source"



3.3) In order to get the right url for your server, go to portal.azure.com and navigate to your Synapse instance. You need to copy the value for “Serverless SQL endpoint”.



3.5) Enter the value you retrieved in the previous step in the textbox for “Server”, and for “Database” enter “s2_contosoisd”, then click “Edit Permissions”



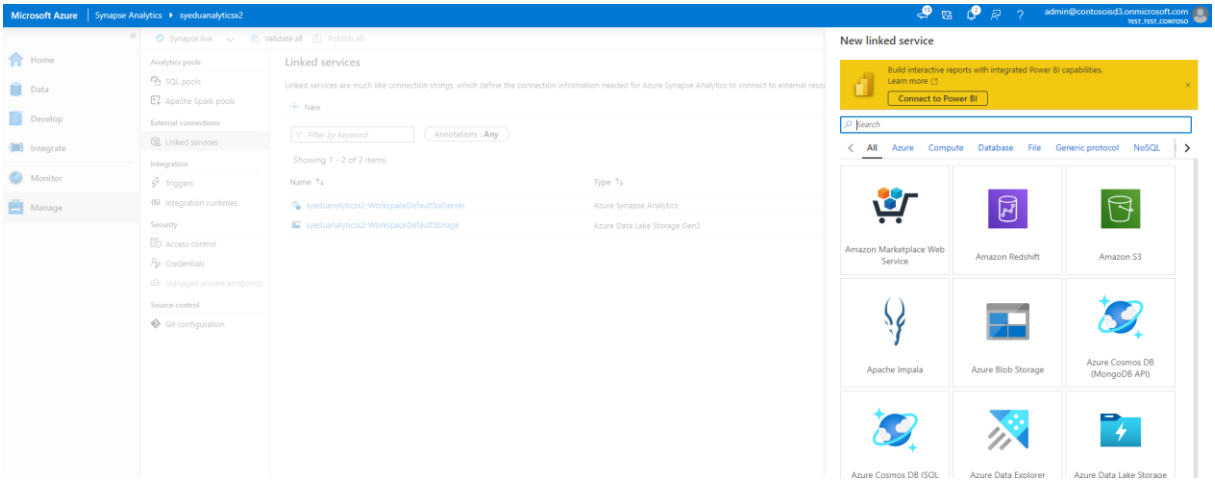
3.6) Click “Edit”, and in the next window click “Microsoft account”, then click “Sign in”, and complete the sign in process with the credentials for the user that has access to the Synapse workspace.

Then click on “Save”, followed by “OK”, followed by “Close”, and then click on “Apply changes”.

5) Connect Power BI workspace

If you have a Power BI Premium license, you have the option of connecting a cloud based Power BI workspace to Synapse.

To connect your Power BI workspace to Synapse so that it is accessible from with Synapse studio, login to Synapse studio and click on "Manage", then select "Linked services", then click on "Connect to Power BI" and complete the form with the connection info to your Power BI workspace.



For more details see: [Linking a Power BI workspace to a Synapse workspace]

For more information about what Power BI licenses are needed for a given scenario, see: [Power BI Premium FAQ - Power BI | Microsoft Docs]

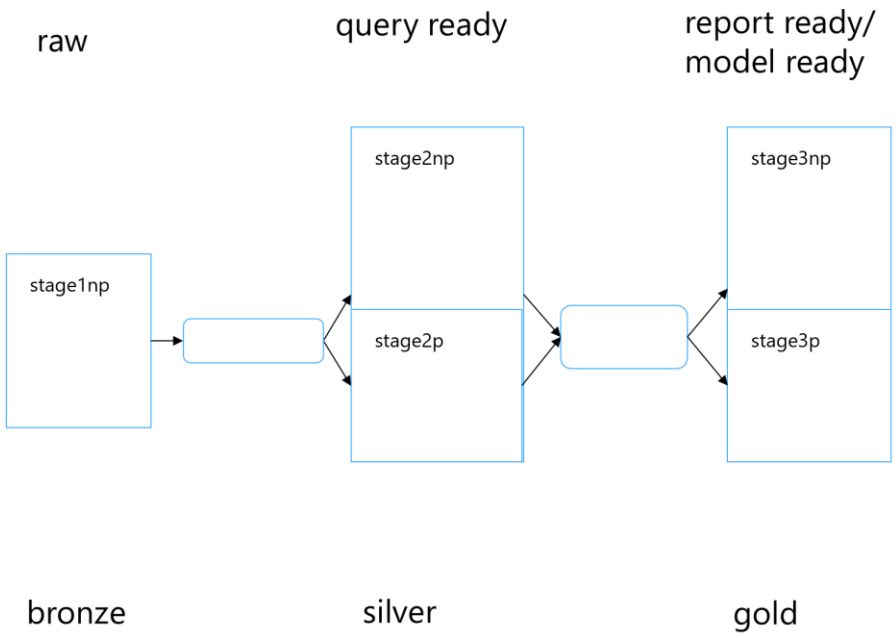
6) Privacy and Security

At the storage level, data protection comes from Azure's automatic data encryption. Data in the data lake is automatically encrypted as it is written to storage using 256-bit AES encryption, and automatically decrypted as it is read from storage from an authorized source.

Security of access is provided at the storage level through the use of Security Groups within Azure Active Directory, allowing the Global Admin to grant the minimum access necessary for specific groups of users to specific zones within the data lake based on the access needed for a given use case. Through this use of role-based access control (RBAC) at the storage level, access is controlled regardless of the tools used to query or analyze the data. Furthermore, using RBAC to set up the minimum access necessary for a new group of users or for a specific use case is straightforward and easily maintained. Additional permissions can be set at the SQL level for finer-grained control over access. See Securing access to ADLS files using Synapse SQL permission model for more info.

In the OEA architecture, data privacy is guarded by first reducing what data is made available - that is, reducing the data set to that which is needed for a given use case. In addition, the data is pseudonymized to protect personally identifiable information (PII). Pseudonymization is defined in the GDPR as "the processing of personal data in such a manner that the personal data can no longer be attributed to a specific data subject without the use of additional information, provided that such additional information is kept separately". Additional info regarding best practices on the use of pseudonymization in a data lake see: Best practices: GDPR and CCPA compliance using Delta Lake

With the OEA framework, the data lake is structured to have 3 conceptual stages – which reflect a common data lake architecture in which the first stage is for raw data, the second stage is for query-ready data, and the third stage is for report ready data. Each stage in the data lake is comprised of Azure storage containers. Containers with the "p" suffix signify that they contain pseduonymized data, and containers with the "np" suffix signify that they contain non-pseudonymized data.



The process of pseudonymization is performed through the pseudonymize method within the OEA framework.