

Open Battery Database

Team members: Yao-Yu Li, Robin Lin, Margherita Taddei, Oliver Zhu
Department of Chemistry and Chemical Engineering, University of Washington

Project sponsor: Dr. Jian Chen
Senior Software Development Engineer, Microsoft



Background and overview

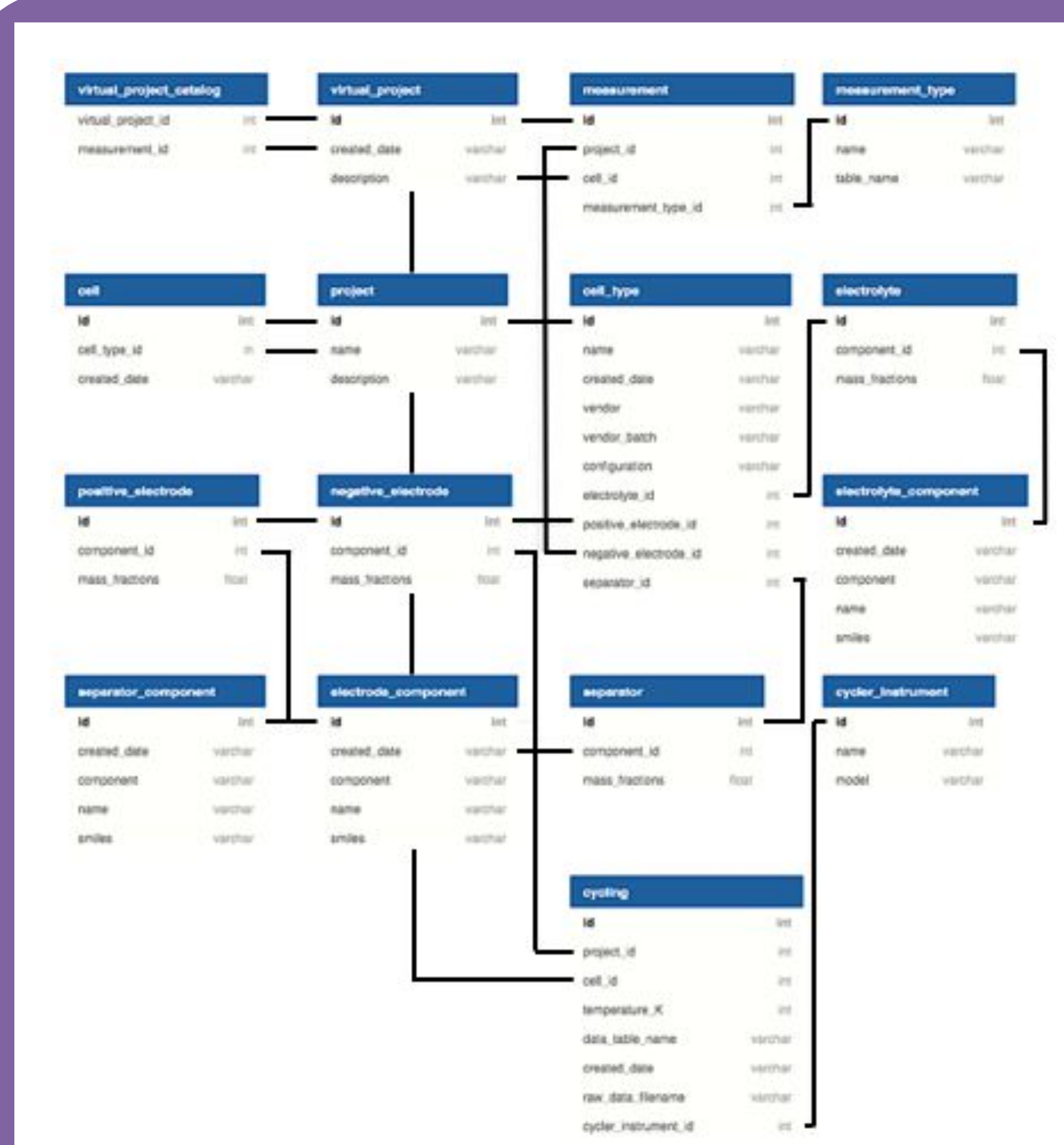
The battery market is expected to grow by more than 12% in the next few years and to meet this high increase in demand it's crucial to improve battery performance.

For this reason, our team created OpenBattery, the first **open, high-performance, relational** battery database, that using a set of software tools can convert battery cycling data from different instruments into a standard format dataframe.

Users would be able to easily **upload, download and query** data on the open-source Openbattery database and use it to develop ML-based models to design highly performant batteries.

Thanks to the Microsoft Azure, a cloud computing service created by Microsoft for building, testing, deploying, and managing applications and services through Microsoft managed data centers, provide us a ready-to-use platform we can build our battery database easier.

A Schema



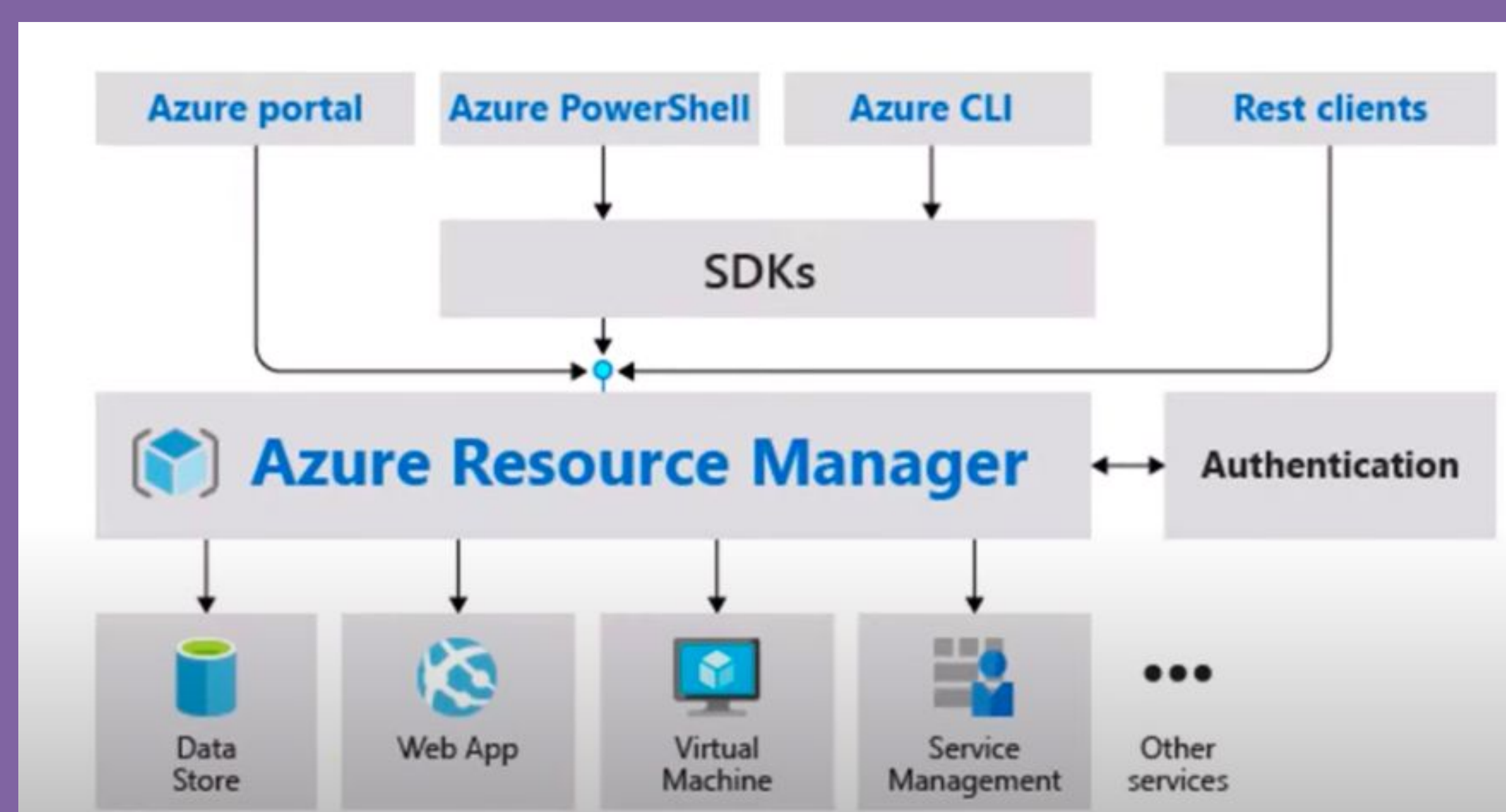
```
> dbo.cell
> dbo.cell_type
> dbo.cycler_instrument
> dbo.cycling
> dbo.electrode_component
> dbo.electrolyte
> dbo.electrolyte_component
> dbo.measurement
> dbo.measurement_type
> dbo.negative_electrode
> dbo.positive_electrode
> dbo.project
> dbo.separator
> dbo.separator_component
> dbo.virtual_project
> dbo.virtual_project_catalog
```



We used Structured Query Language (SQL) to create the backbone of our relational OpenBattery database which is called "schema".

A database in SQL Server is made up of a collection of tables that stores a specific set of structured data. A table contains a collection of rows, also referred to as records or tuples, and columns, also referred to as attributes. Each column in the table is designed to store a certain type of information, for example, the type of electrode, electrolyte and separator of a battery.

B Introduction of Azure



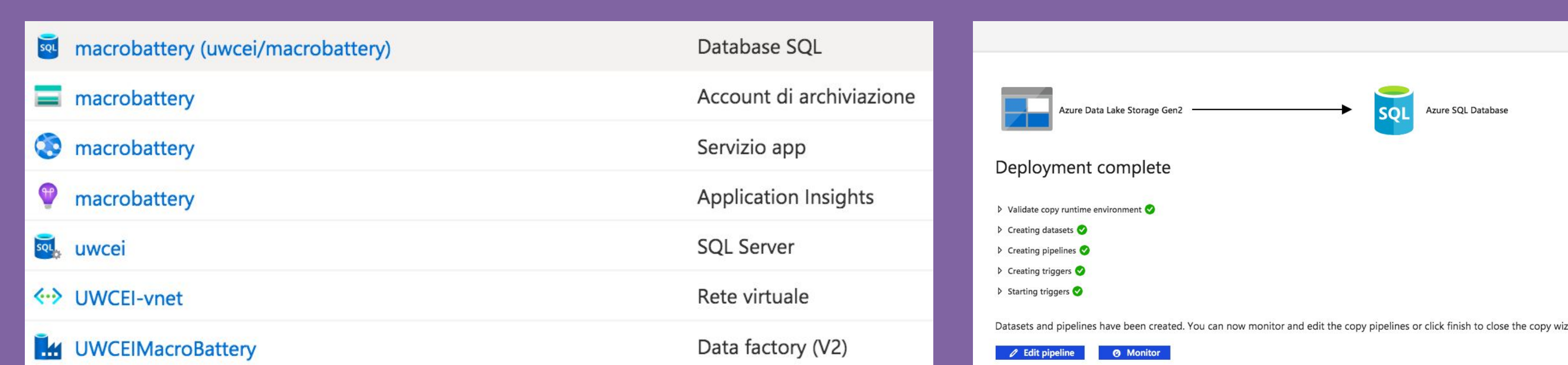
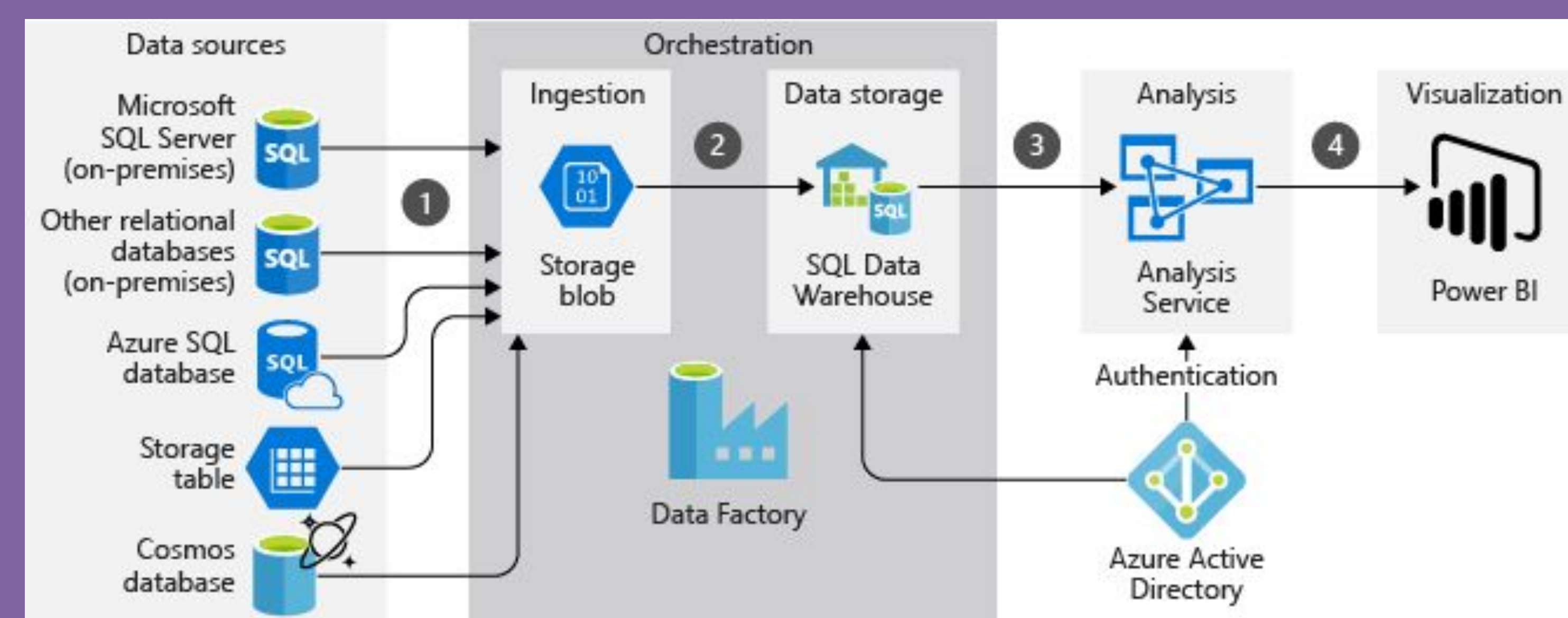
Shown is Azure Architecture can be divided into 3 main components, Front Ends, Middle Ware, Services.

The Azure services includes but not limited in Compute, Networking, File Storage, Database, AI + Machine Learning, Identity and management.

For Example, the Computing service contains the Azure **Virtual Machines, Function App, App Service**, Azure Kubernetes Services.

In this project, the most used service are **Blob Storage, SQL database, SQL server, Data factory, and Web service.**

C Data Factory and Azure SQL Database



On Azure we created our own storage account, Azure SQL Database, Data Factory and Web App service.

The Azure storage account enabled us to put large amount of data and create containers where we upload csv files containing the information of each column of the schema.

Then using the Data Factory service, we were able to transfer the data from the csv files to the Azure SQL Database.

Azure SQL database is an intelligent, scalable, cloud database service that provides the broadest SQL Server engine compatibility.

Azure Data Factory works as a glue connecting many of the other Azure data services. This allows the development of data-flow pipelines that are capable of ingesting, transforming, analyzing, and even publishing data and insights. Moreover, these pipelines can be developed, scheduled, and monitored all from within the Azure Portal.

D Web App

Future

In the near future we will continue to enrich OpenBattery with new battery data and the SQL relational database and Web App service will be used by UWCEI for further development

References

<https://portal.azure.com/#home>
<https://github.com/OpenBattery/OpenBattery>
<https://uwemilab.github.io/>
<https://web.calce.umd.edu/batteries/data.htm#INR18650-test>

Acknowledgments

