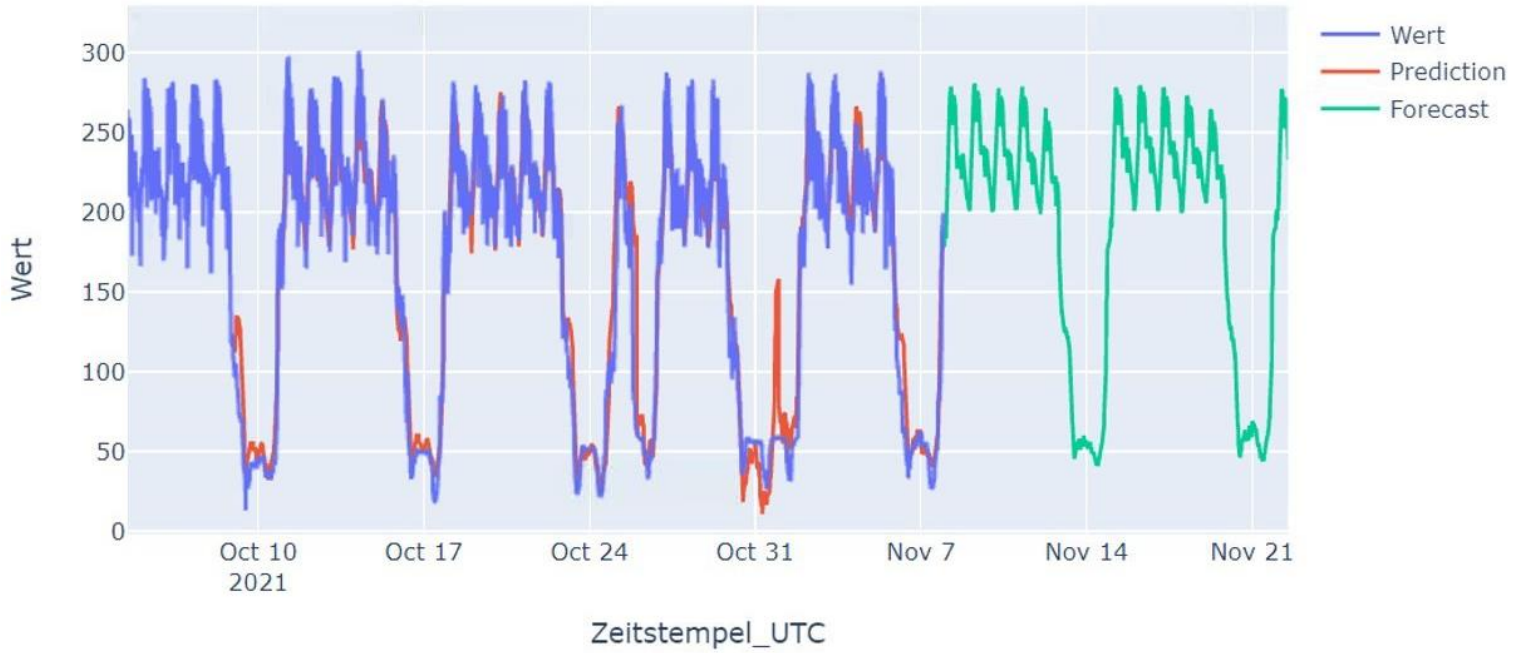
ACP CUBIDO
Digital Solutions

From Synapse to Fabric
Insights of a Customer POV

IT for innovators.
Analytics & Development

Use Case: Energy Consumption Prediction





IT for innovators.
Analytics & Development



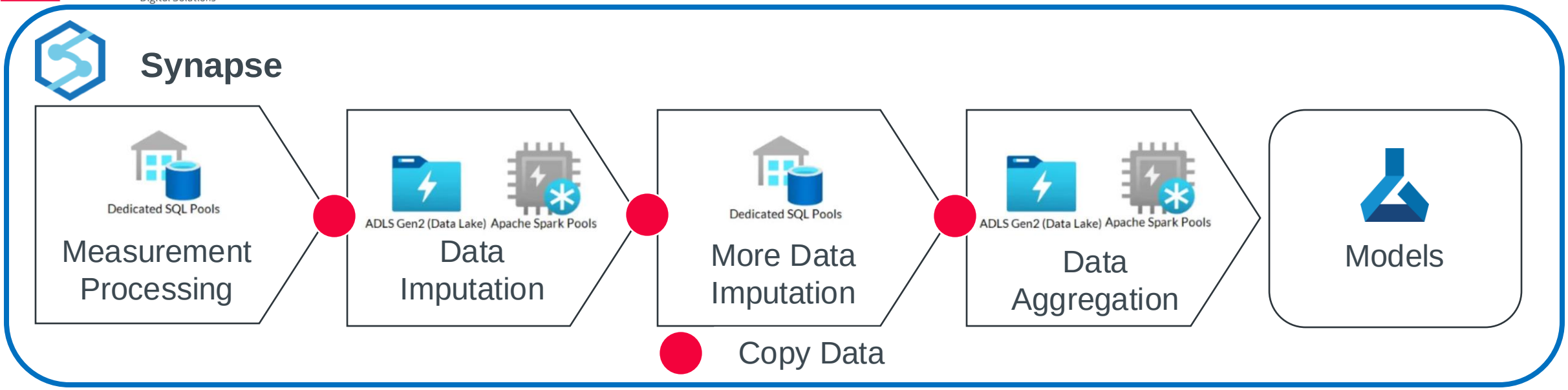
IT for innovators.
Analytics & Development



IT for innovators.
Analytics & Development

Synapse

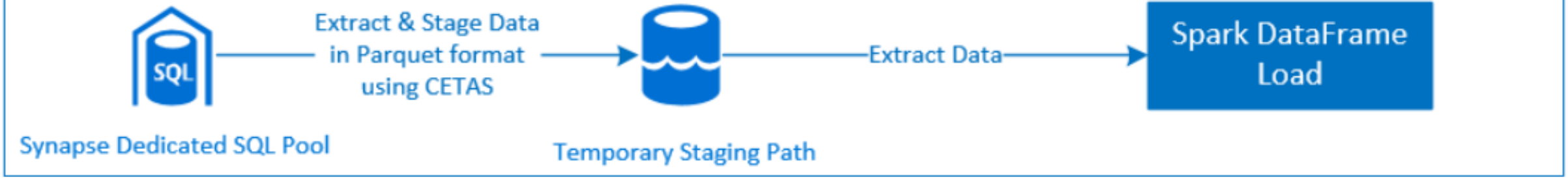

Dedicated SQL Pools
Measurement Processing
ADLS Gen2 (Data Lake)
Apache Spark Pools
Data Imputation
Dedicated SQL Pools
More Data Imputation
ADLS Gen2 (Data Lake)
Apache Spark Pools
Data Aggregation
Models


Synapse
Dedicated SQL Pools
Measurement Processing
ADLS Gen2 (Data Lake)
Apache Spark Pools
Data Imputation
Dedicated SQL Pools
More Data Imputation
ADLS Gen2 (Data Lake)
Apache Spark Pools
Data Aggregation
Models
Copy Data

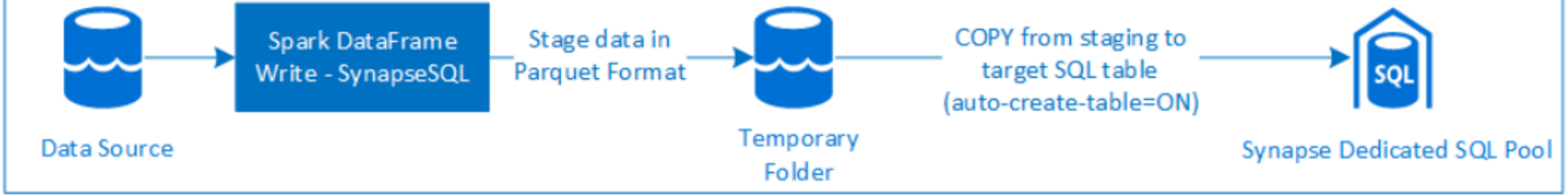
Synapse

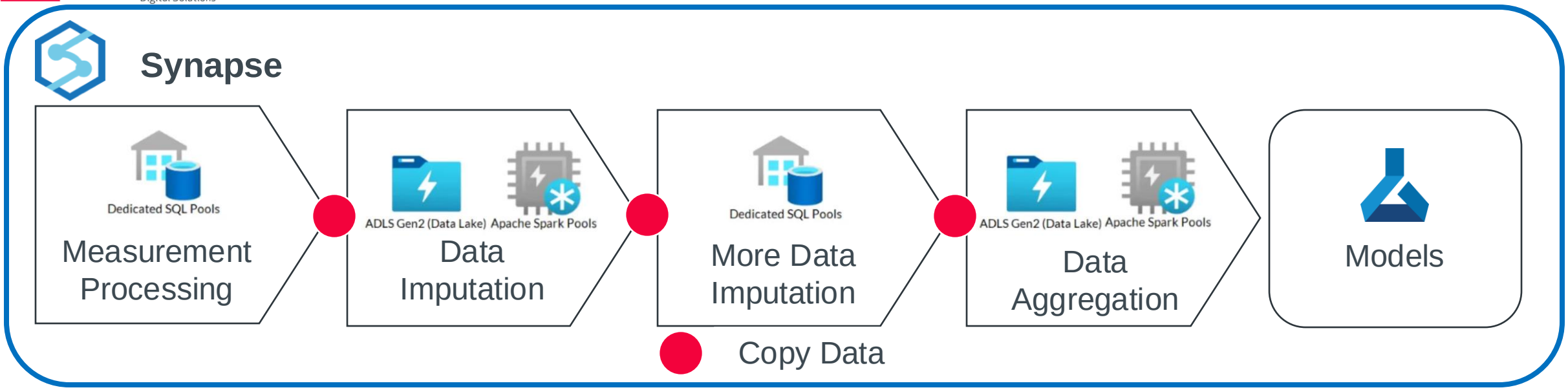


Read from an Internal Table



Write to an Internal Table Type




Synapse
Dedicated SQL Pools
Measurement Processing
ADLS Gen2 (Data Lake)
Apache Spark Pools
Data Imputation
Dedicated SQL Pools
More Data Imputation
ADLS Gen2 (Data Lake)
Apache Spark Pools
Data Aggregation
Models
Copy Data

Fabric Data Storage - Lakehouse and Warehouse



Lakehouse

- Type of data:
  - Unstructured, semi-structured, structured
- Dev languages:
  - Spark (Scala, PySpark, Spark SQL, R)
- Data organized by:
  - Folders and files, databases, and tables
- Dev interface:
  - Spark notebooks

Warehouse

- Type of data:
  - semi-structured (JSON), structured
- Dev languages:
  - SQL
- Data organized by:
  - Databases, schemas, and tables
- Dev interface:
  - SQL scripts



Synapse



Measurement Processing




Data Imputation



More Data Imputation




Data Aggregation



Models

Copy Data



Fabric



Measurement Processing



Data Imputation



More Data Imputation



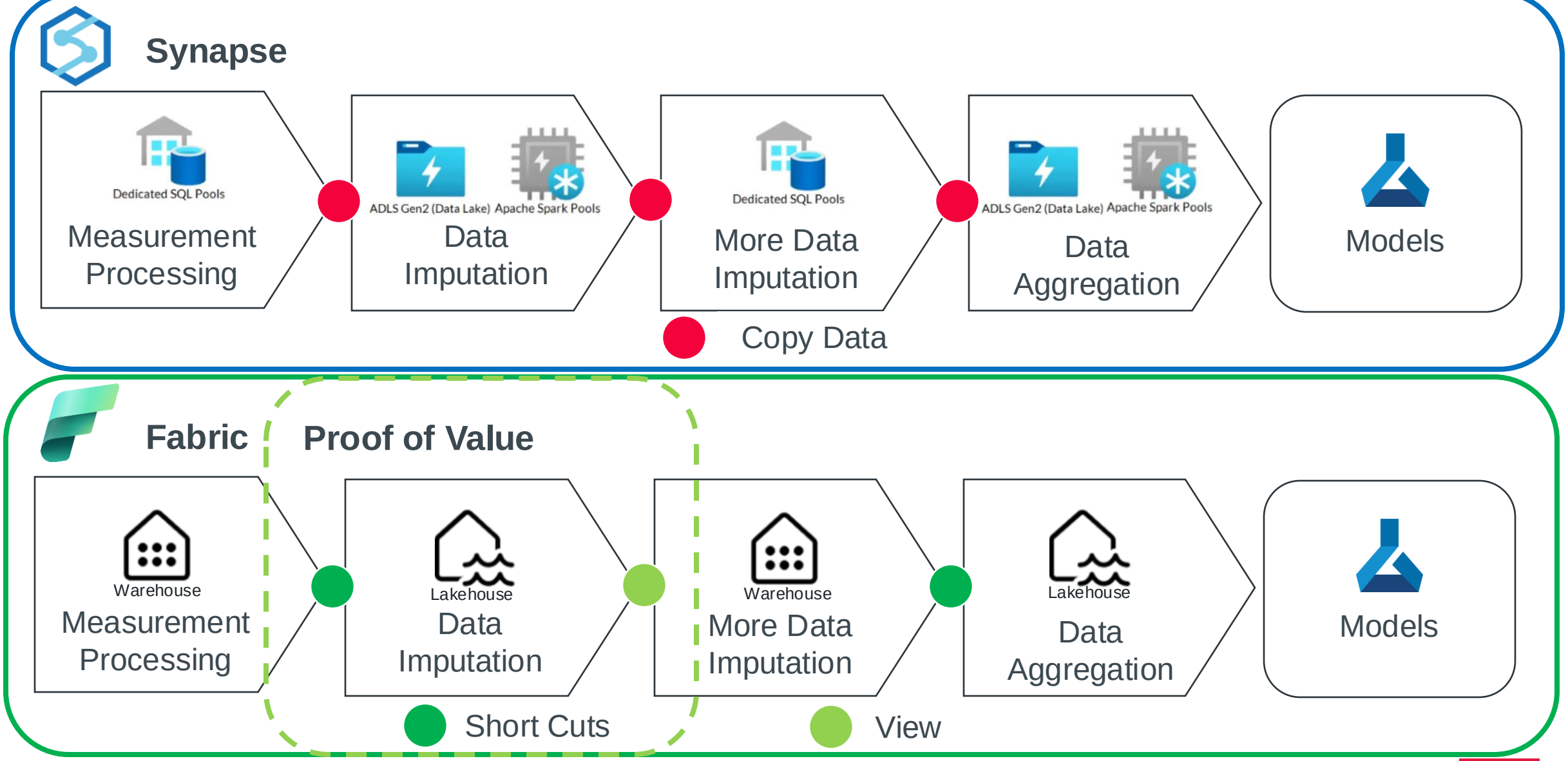
Data Aggregation

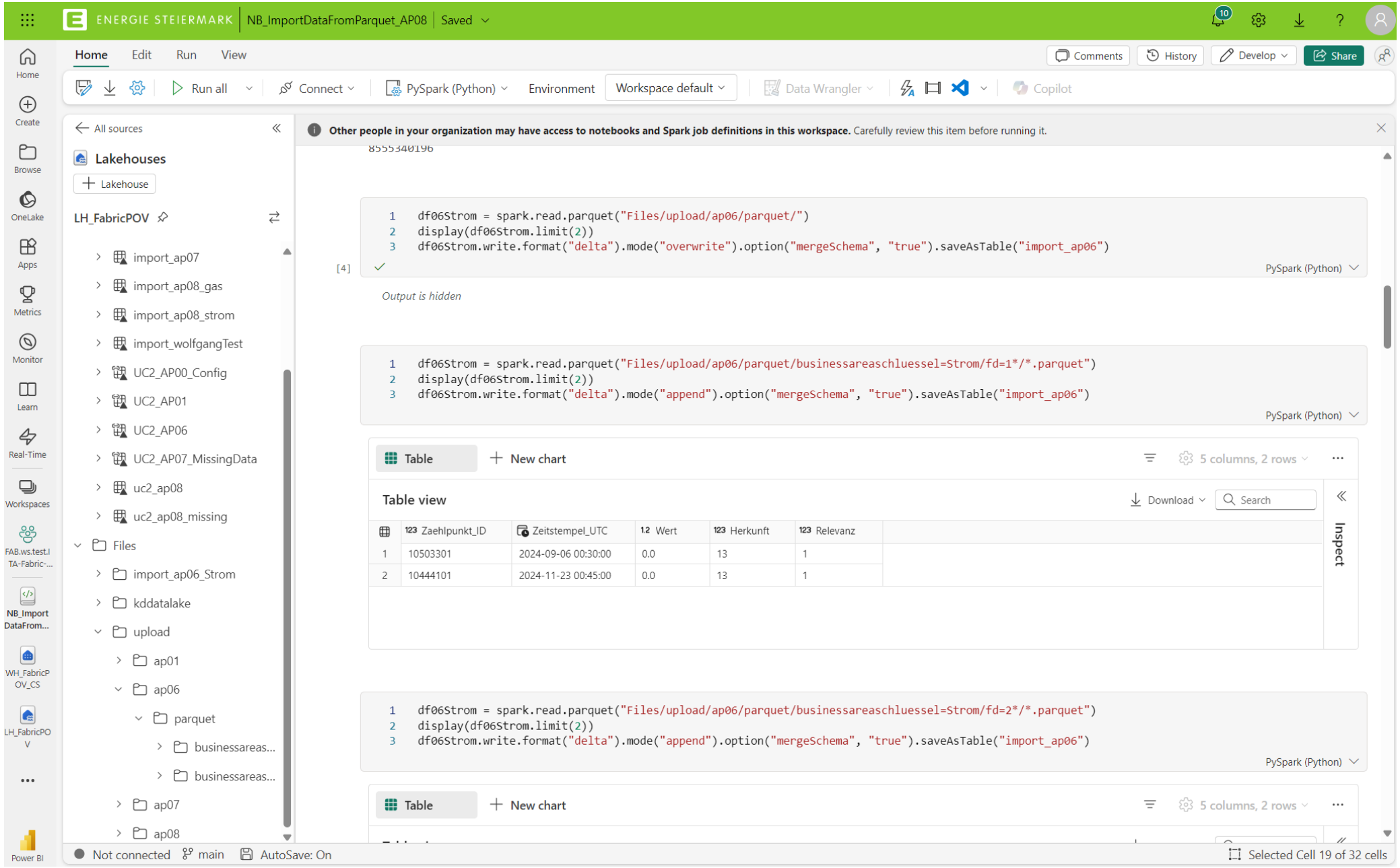


Models

Short Cuts

View


Synapse
Dedicated SQL Pools
Measurement Processing
ADLS Gen2 (Data Lake)
Apache Spark Pools
Data Imputation
Dedicated SQL Pools
More Data Imputation
ADLS Gen2 (Data Lake)
Apache Spark Pools
Data Aggregation
Models
Copy Data
Fabric
Proof of Value
Warehouse
Measurement Processing
Lakehouse
Data Imputation
Warehouse
More Data Imputation
Lakehouse
Data Aggregation
Models
Short Cuts
View


ENERGIE STEIERMARK
NB_ImportDataFromParquet_AP08 | Saved
Home
Edit
Run
View
Comments
History
Develop
Share
Run all
Connect
PySpark (Python)
Environment
Workspace default
Data Wrangler
Copilot
All sources
Lakehouses
Lakehouse
LH_FabricPOV
import_ap07
import_ap08_gas
import_ap08_strom
import_wolfgangTest
UC2_AP00_Config
UC2_AP01
UC2_AP06
UC2_AP07_MissingData
uc2_ap08
uc2_ap08_missing
Files
import_ap06_Strom
kddatalake
upload
ap01
ap06
parquet
businessareas...
businessareas...
ap07
ap08
Other people in your organization may have access to notebooks and Spark job definitions in this workspace. Carefully review this item before running it.
8555340196
1 df06Strom = spark.read.parquet("Files/upload/ap06/parquet/")
2 display(df06Strom.limit(2))
3 df06Strom.write.format("delta").mode("overwrite").option("mergeSchema", "true").saveAsTable("import_ap06")
[4]
PySpark (Python)
Output is hidden
1 df06Strom = spark.read.parquet("Files/upload/ap06/parquet/businessareaschluessel=Strom/fd=1*/*.parquet")
2 display(df06Strom.limit(2))
3 df06Strom.write.format("delta").mode("append").option("mergeSchema", "true").saveAsTable("import_ap06")
PySpark (Python)
Table
New chart
5 columns, 2 rows
Table view
Download
Search
Inspect
123 Zaehlpunkt_ID
Zeitstempel_UTC
1.2 Wert
123 Herkunft
123 Relevanz
1 10503301 2024-09-06 00:30:00 0.0 13 1
2 10444101 2024-11-23 00:45:00 0.0 13 1
1 df06Strom = spark.read.parquet("Files/upload/ap06/parquet/businessareaschluessel=Strom/fd=2*/*.parquet")
2 display(df06Strom.limit(2))
3 df06Strom.write.format("delta").mode("append").option("mergeSchema", "true").saveAsTable("import_ap06")
PySpark (Python)
Table
New chart
5 columns, 2 rows
Not connected
main
AutoSave: On
Selected Cell 19 of 32 cells

Fabric Warehouse Collation

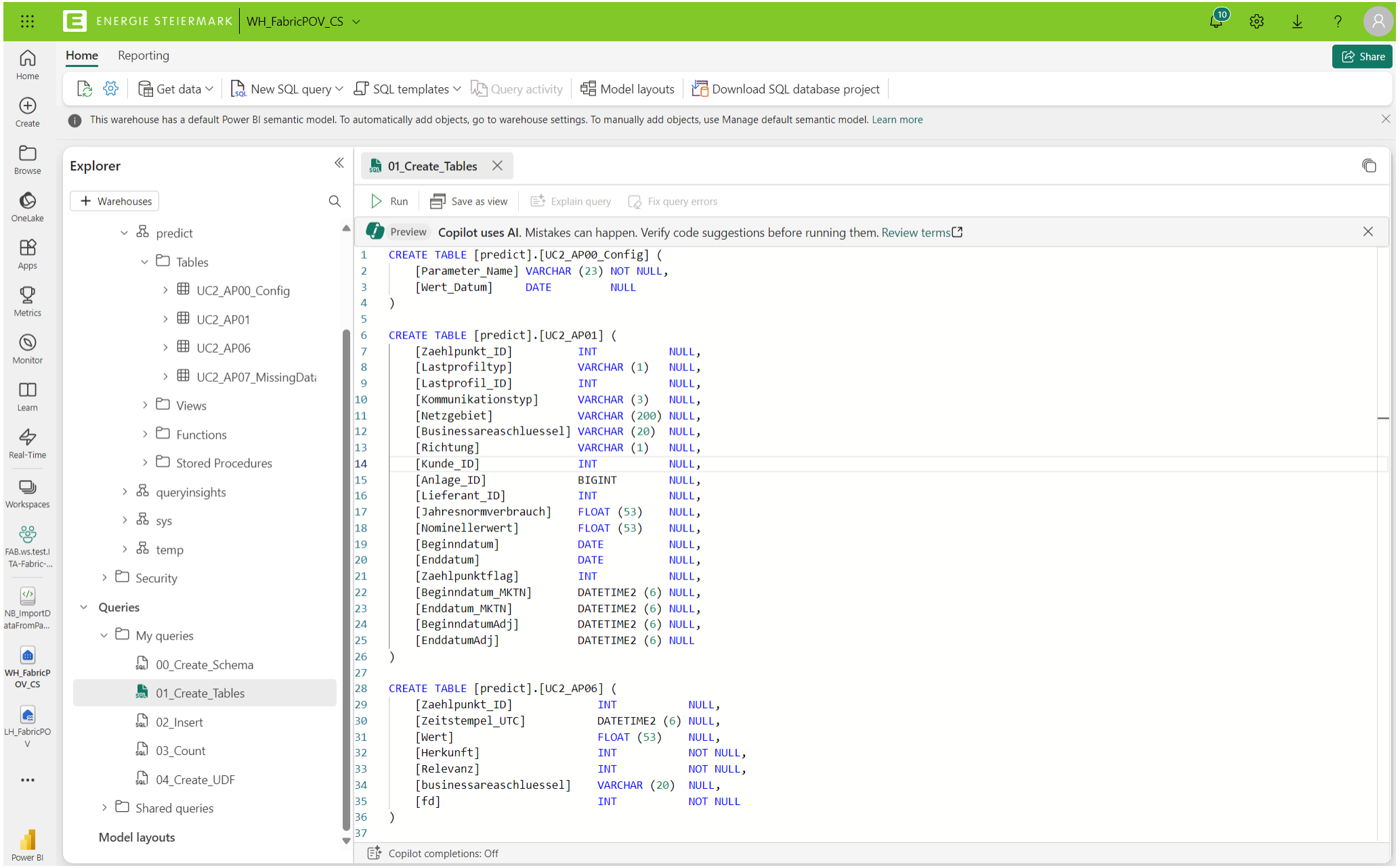
- Default Collation: Latin1_General_100_BIN2_UTF8
- Case Insensitive: Latin1_General_100_CI_AS_KS_WS_SC_UTF8

Fabric Warehouse Collation

- Default Collation: Latin1_General_100_BIN2_UTF8
  - compatible with parquet string sorting rules
  - optimized filtering

- Case Insensitive: Latin1_General_100_CI_AS_KS_WS_SC_UTF8

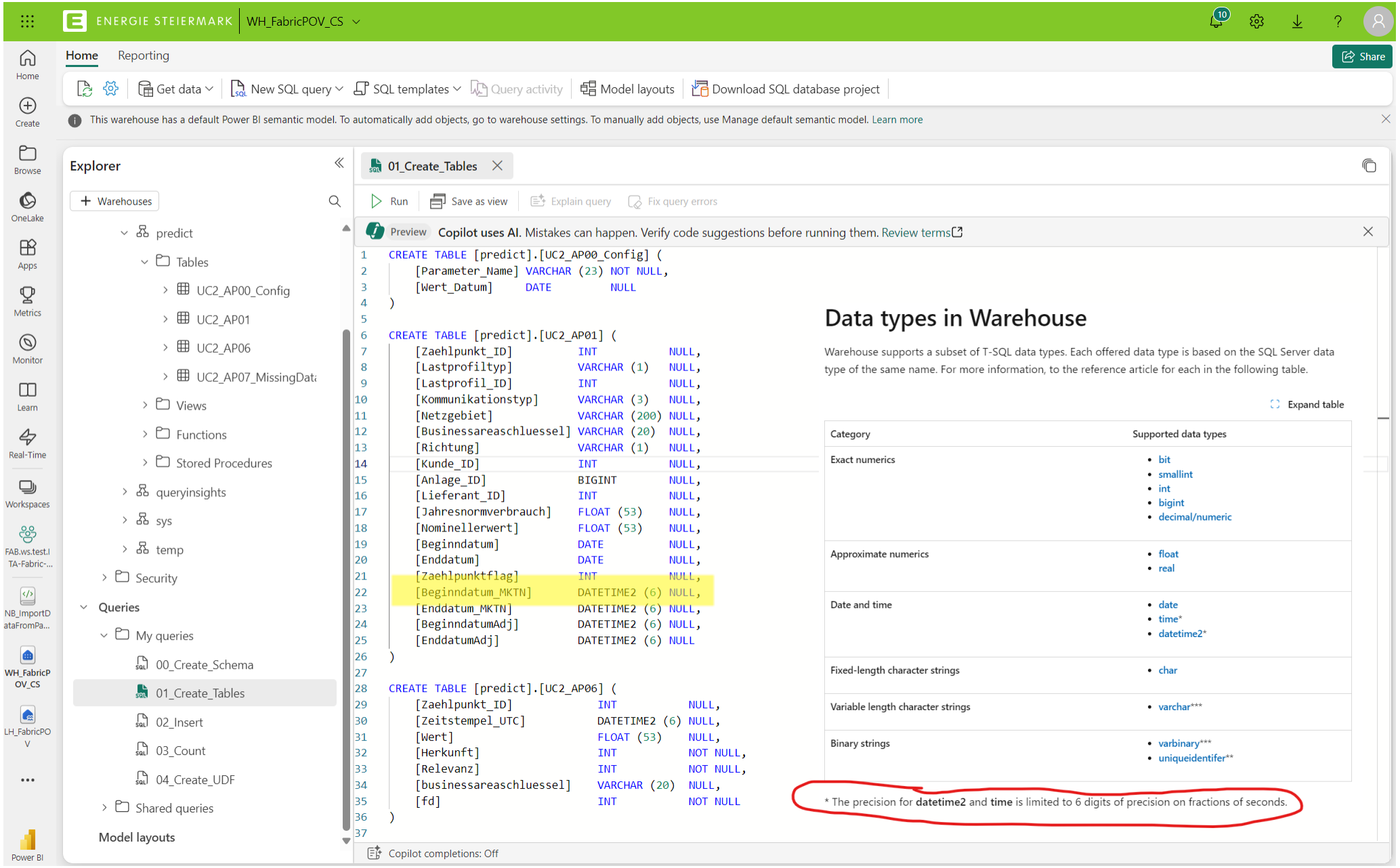
Currently, the only method available for creating a case-insensitive data warehouse is via REST API. This article provides a step-by-step guide on how to create a warehouse with case-insensitive collation through the REST API. It also explains how to use Visual Studio Code with the REST Client extension to facilitate the process.

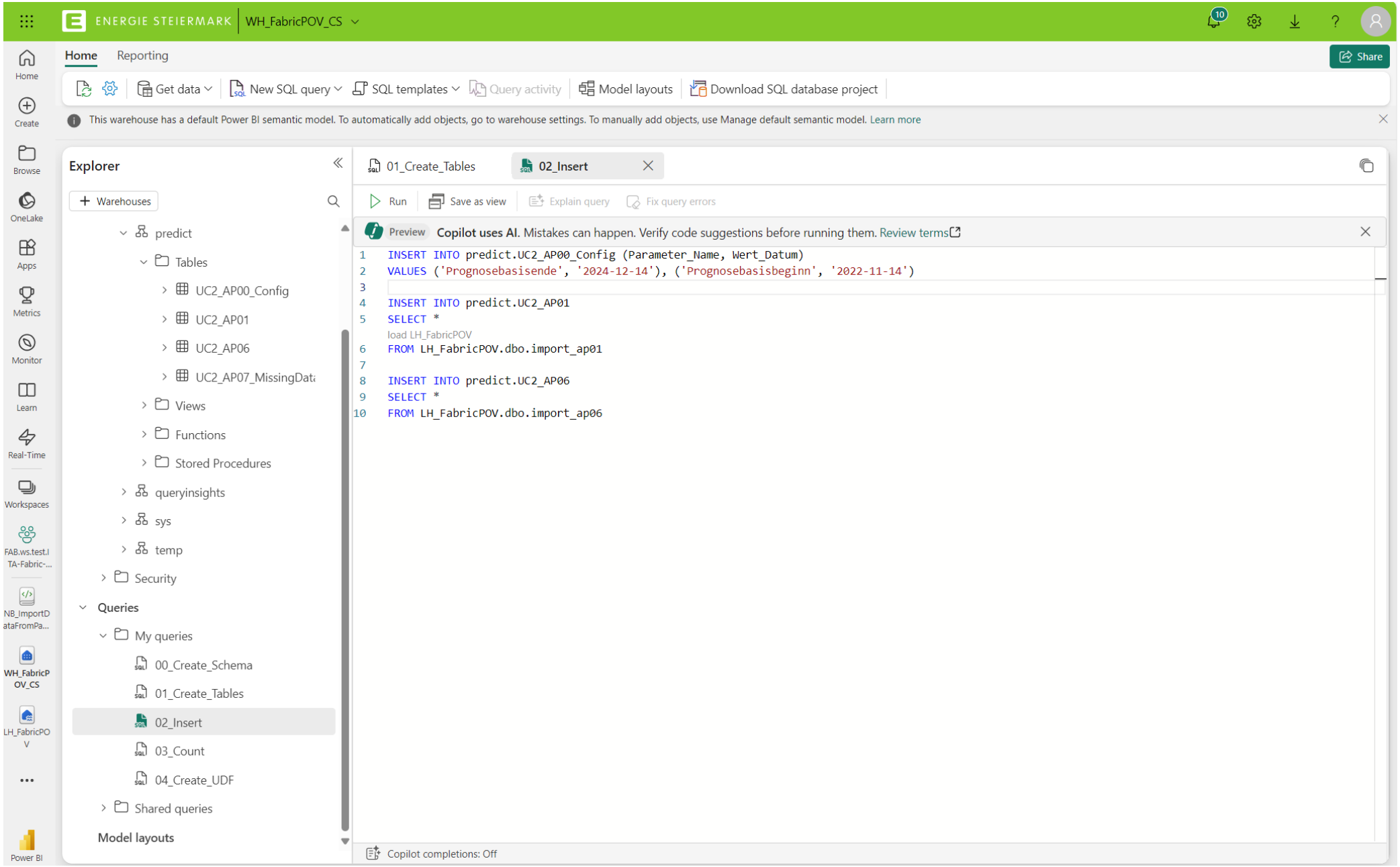



```
CREATE TABLE [predict].[UC2_AP00_Config] (
    [Parameter_Name] VARCHAR (23) NOT NULL,
    [Wert_Datum]     DATE         NULL
)

CREATE TABLE [predict].[UC2_AP01] (
    [Zaehlpunkt_ID]          INT           NULL,
    [Lastprofiltyp]          VARCHAR (1)   NULL,
    [Lastprofil_ID]          INT           NULL,
    [Kommunikationstyp]      VARCHAR (3)   NULL,
    [Netzgebiet]             VARCHAR (200) NULL,
    [Businessareaschluessel] VARCHAR (20)  NULL,
    [Richtung]               VARCHAR (1)   NULL,
    [Kunde_ID]               INT           NULL,
    [Anlage_ID]              BIGINT        NULL,
    [Lieferant_ID]           INT           NULL,
    [Jahresnormverbrauch]    FLOAT (53)    NULL,
    [Nominellerwert]         FLOAT (53)    NULL,
    [Beginndatum]            DATE          NULL,
    [Enddatum]               DATE          NULL,
    [Zaehlpunktflag]         INT           NULL,
    [Beginndatum_MKTN]       DATETIME2 (6) NULL,
    [Enddatum_MKTN]          DATETIME2 (6) NULL,
    [BeginndatumAdj]         DATETIME2 (6) NULL,
    [EnddatumAdj]            DATETIME2 (6) NULL
)

CREATE TABLE [predict].[UC2_AP06] (
    [Zaehlpunkt_ID]          INT           NULL,
    [Zeitstempel_UTC]        DATETIME2 (6) NULL,
    [Wert]                   FLOAT (53)    NULL,
    [Herkunft]               INT           NOT NULL,
    [Relevanz]               INT           NOT NULL,
    [businessareaschluessel] VARCHAR (20)  NULL,
    [fd]                     INT           NOT NULL
)
```


ENERGIE STEIERMARK
WH_FabricPOV_CS
Home
Reporting
Share
Get data
New SQL query
SQL templates
Query activity
Model layouts
Download SQL database project
This warehouse has a default Power BI semantic model. To automatically add objects, go to warehouse settings. To manually add objects, use Manage default semantic model. Learn more
Explorer
Warehouses
predict
Tables
UC2_AP00_Config
UC2_AP01
UC2_AP06
UC2_AP07_MissingDat
Views
Functions
Stored Procedures
queryinsights
sys
temp
Security
Queries
My queries
00_Create_Schema
01_Create_Tables
02_Insert
03_Count
04_Create_UDF
Shared queries
Model layouts
01_Create_Tables
Run
Save as view
Explain query
Fix query errors
Preview
Copilot uses AI. Mistakes can happen. Verify code suggestions before running them. Review terms
1 CREATE TABLE [predict].[UC2_AP00_Config] (
2 [Parameter_Name] VARCHAR (23) NOT NULL,
3 [Wert_Datum] DATE NULL
4)
5
6 CREATE TABLE [predict].[UC2_AP01] (
7 [Zaehlpunkt_ID] INT NULL,
8 [Lastprofiltyp] VARCHAR (1) NULL,
9 [Lastprofil_ID] INT NULL,
10 [Kommunikationstyp] VARCHAR (3) NULL,
11 [Netzgebiet] VARCHAR (200) NULL,
12 [Businessareaschluessel] VARCHAR (20) NULL,
13 [Richtung] VARCHAR (1) NULL,
14 [Kunde_ID] INT NULL,
15 [Anlage_ID] BIGINT NULL,
16 [Lieferant_ID] INT NULL,
17 [Jahresnormverbrauch] FLOAT (53) NULL,
18 [Nominellerwert] FLOAT (53) NULL,
19 [Beginndatum] DATE NULL,
20 [Enddatum] DATE NULL,
21 [Zaehlpunktflag] INT NULL,
22 [Beginndatum_MKTN] DATETIME2 (6) NULL,
23 [Enddatum_MKTN] DATETIME2 (6) NULL,
24 [BeginndatumAdj] DATETIME2 (6) NULL,
25 [EnddatumAdj] DATETIME2 (6) NULL
26)
27
28 CREATE TABLE [predict].[UC2_AP06] (
29 [Zaehlpunkt_ID] INT NULL,
30 [Zeitstempel_UTC] DATETIME2 (6) NULL,
31 [Wert] FLOAT (53) NULL,
32 [Herkunft] INT NOT NULL,
33 [Relevanz] INT NOT NULL,
34 [businessareaschluessel] VARCHAR (20) NULL,
35 [fd] INT NOT NULL
36)
37
Data types in Warehouse
Warehouse supports a subset of T-SQL data types. Each offered data type is based on the SQL Server data type of the same name. For more information, to the reference article for each in the following table.
Expand table
Category
Supported data types
Exact numerics
bit
smallint
int
bigint
decimal/numeric
Approximate numerics
float
real
Date and time
date
time*
datetime2*
Fixed-length character strings
char
Variable length character strings
varchar***
Binary strings
varbinary***
uniqueidentifer**
* The precision for datetime2 and time is limited to 6 digits of precision on fractions of seconds.
Copilot completions: Off


ENERGIE STEIERMARK
WH_FabricPOV_CS
Home
Reporting
Share
Get data
New SQL query
SQL templates
Query activity
Model layouts
Download SQL database project
This warehouse has a default Power BI semantic model. To automatically add objects, go to warehouse settings. To manually add objects, use Manage default semantic model. Learn more
Explorer
Warehouses
predict
Tables
UC2_AP00_Config
UC2_AP01
UC2_AP06
UC2_AP07_MissingDat
Views
Functions
Stored Procedures
queryinsights
sys
temp
Security
Queries
My queries
00_Create_Schema
01_Create_Tables
02_Insert
03_Count
04_Create_UDF
Shared queries
Model layouts
01_Create_Tables
02_Insert
Run
Save as view
Explain query
Fix query errors
Preview
Copilot uses AI. Mistakes can happen. Verify code suggestions before running them. Review terms
1 INSERT INTO predict.UC2_AP00_Config (Parameter_Name, Wert_Datum)
2 VALUES ('Prognosebasisende', '2024-12-14'), ('Prognosebasisbeginn', '2022-11-14')
3
4 INSERT INTO predict.UC2_AP01
5 SELECT *
load LH_FabricPOV
6 FROM LH_FabricPOV.dbo.import_ap01
7
8 INSERT INTO predict.UC2_AP06
9 SELECT *
10 FROM LH_FabricPOV.dbo.import_ap06
Copilot completions: Off

ACP CUBIDO Digital Solutions

Synapse (DW1000c)

PipelineRunID	Command	StartTimestamp	StartDate_ID	LoadDurationMinutes
88ce08ba-2e49-41a8-8f43-ff1844d8c7b2	[predict].[load_UC2_AP07_MissingData]	2025-01-05 08:20:23.5143399	2025-01-05	110.966666
3e3d9cf4-c14e-4a5f-8a7f-a4d29e0b21e3	[predict].[load_UC2_AP07_MissingData]	2024-12-29 09:29:25.2698165	2024-12-29	71.083333
046956db-2e38-4402-84e2-db05481f2556	[predict].[load_UC2_AP07_MissingData]	2024-12-22 08:08:34.2755376	2024-12-22	74.066666
6589a894-3dd0-4629-9736-752e772e2baa	[predict].[load_UC2_AP07_MissingData]	2024-12-15 09:47:15.1103937	2024-12-15	112.366666
a3cc218f-36d0-4f4e-8497-b42be87b54fc	[predict].[load_UC2_AP07_MissingData]	2024-12-08 07:49:26.2322011	2024-12-08	100.433333

Fabric (F64)

UID PipelineRunID	ABC Command	StartTimestamp	StartDate_ID	1.2 LoadDurationMinutes
00000000-0000-0000-0000-000000000000	[predict].[load_UC2_AP07_MissingData]	2025-01-10 08:38:24.106323	2025-01-10	10.083333
00000000-0000-0000-0000-000000000000	[predict].[load_UC2_AP07_MissingData]	2025-01-09 16:34:38.409632	2025-01-09	9.450000
00000000-0000-0000-0000-000000000000	[predict].[load_UC2_AP07_MissingData]	2025-01-09 13:16:41.868417	2025-01-09	10.333333

ACP CUBIDO Digital Solutions

Synapse (DW1000c)

PipelineRunID	Command	StartTimestamp	StartDate_ID	LoadDurationMinutes
88ce08ba-2e49-41a8-8f43-ff1844d8c7b2	[predict].[load_UC2_AP07_MissingData]	2025-01-05 08:20:23.51	025-01-05	110.966666
3e3d9cf4-c14e-4a5f-8a7f-a4d29e0b21e3	[predict].[load_UC2_AP07_MissingData]	2024-12-29 09:29:	-12-29	71.083333
046956db-2e38-4402-84e2-db05481f2556	[predict].[load_UC2_AP07_MissingData]	2024-12-22 0	2024-12-22	74.066666
6589a894-3dd0-4629-9736-752e772e2baa	[predict].[load_UC2_AP07_MissingData]	2024-12	937 2024-12-15	112.366666
a3cc218f-36d0-4f4e-8497-b42be87b54fc	[predict].[load_UC2_AP07_MissingData]	20	.2322011 2024-12-08	100.433333

7-10 Times Faster

Fabric (F64)

UID PipelineRunID	ABC Command	StartTimestamp	StartDate_ID	1.2 LoadDurationMinutes
00000000-0000-0000-0000-000000000000	[predict].[load_UC2_AP07_MissingData]	2025-01-10 08:38:24.106323	2025-01-10	10.083333
00000000-0000-0000-0000-000000000000	[predict].[load_UC2_AP07_MissingData]	2025-01-09 16:34:38.409632	2025-01-09	9.450000
00000000-0000-0000-0000-000000000000	[predict].[load_UC2_AP07_MissingData]	2025-01-09 13:16:41.868417	2025-01-09	10.333333

IT for innovators. Analytics & Development

ACP CUBIDO Digital Solutions

Synapse (DW1000c)

PipelineRunID	Command	StartTimestamp	StartDate_ID	LoadDurationMinutes
88ce08ba-2e49-41a8-8f43-ff1844d8c7b2	[predict].[load_UC2_AP07_MissingData]	2025-01-05 08:20:23.51	025-01-05	110.
3e3d9cf4-c14e-4a5f-8a7f-a4d29e0b21e3	[predict].[load_UC2_AP07_MissingData]	2024-12-29 09:29:	-12-29	
046956db-2e38-4402-84e2-db05481f2556	[predict].[load_UC2_AP07_MissingData]	2024-12-22	2024-12-22	
6589a894-3dd0-4629-9736-752e772e2baa	[predict].[load_UC2_AP07_MissingData]	2024-12	2024-12-	6666
a3cc218f-36d0-4f4e-8497-b42be87b54fc	[predict].[load_UC2_AP07_MissingData]	20 .2322011	202	00.433333

7-10 Times Faster

No Copy (~2h)

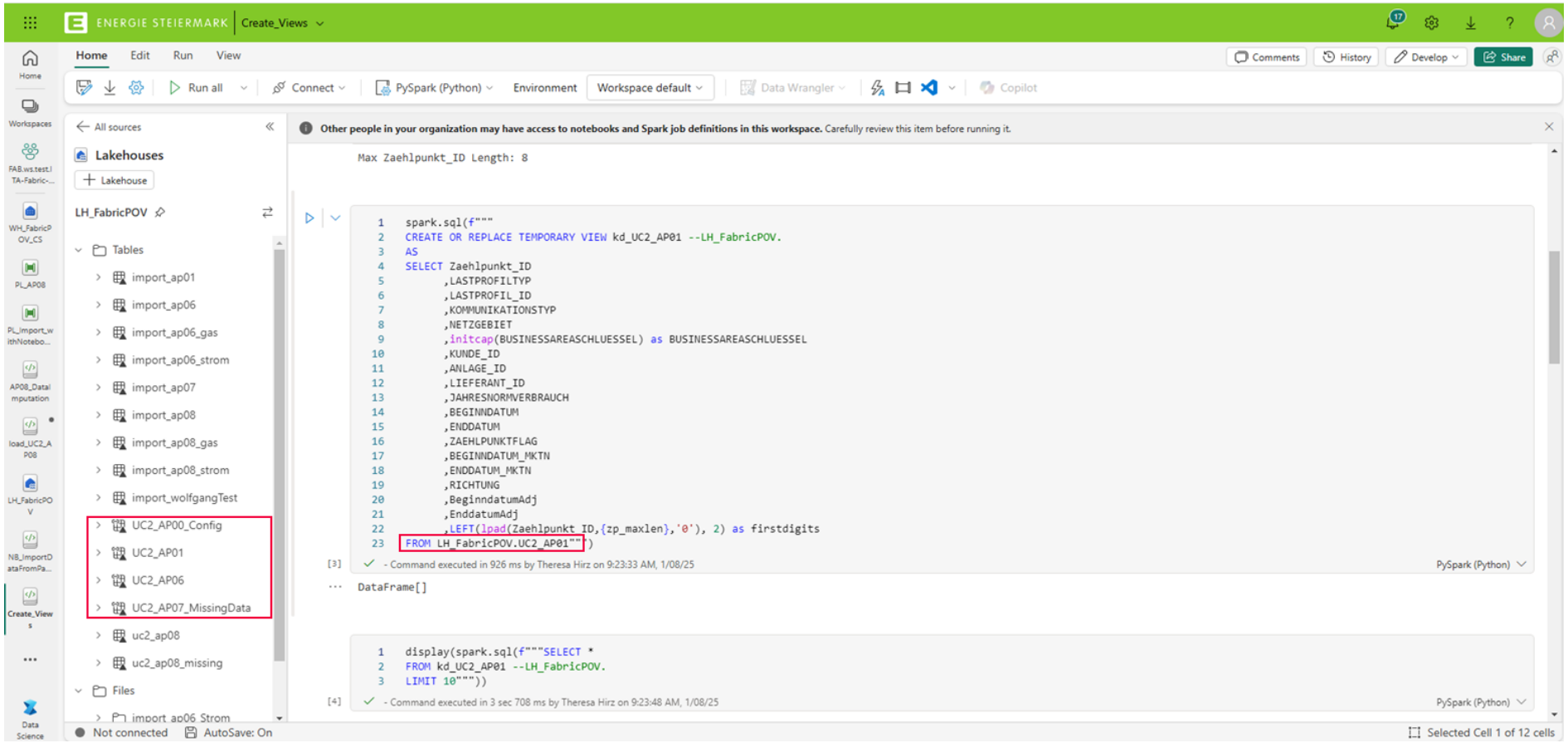
Fabric (F64)

UID PipelineRunID	ABC Command	StartTimestamp	StartDate_ID	1.2 LoadDurationMinutes
00000000-0000-0000-0000-000000000000	[predict].[load_UC2_AP07_MissingData]	2025-01-10 08:38:24.106323	2025-01-10	10.083333
00000000-0000-0000-0000-000000000000	[predict].[load_UC2_AP07_MissingData]	2025-01-09 16:34:38.409632	2025-01-09	9.450000
00000000-0000-0000-0000-000000000000	[predict].[load_UC2_AP07_MissingData]	2025-01-09 13:16:41.868417	2025-01-09	10.333333

IT for innovators. Analytics & Development

Data Imputation in Lakehouse

- Learnings we had when moving Synapse Spark Code for Data Imputation to Fabric



IT for innovators.
Analytics & Development

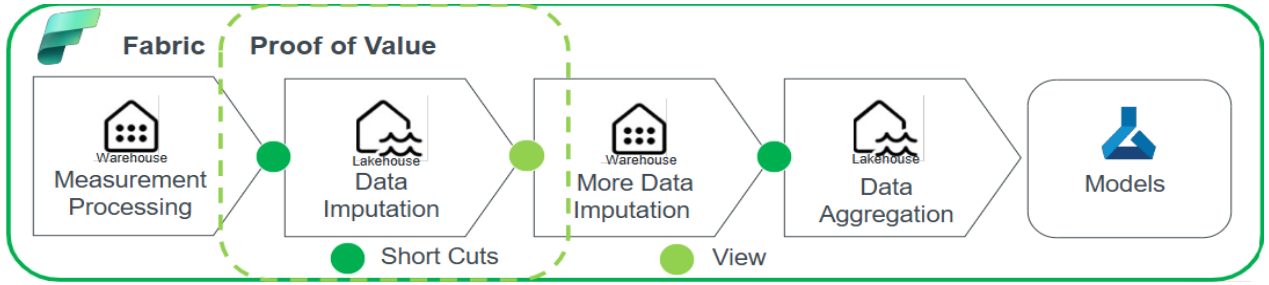
Synapse vs. Fabric Pipeline Runtime – Data Imputation

Azure Synapse

- Spark Pool:
  - Medium Node Size
  - 8 vCores
  - 64GB Memory
  - 20 Executors
- Runtime: 1h 54m 9s

Microsoft Fabric

- Spark Pool:
  - Medium Node Size
  - 8 vCores
  - 56GB Memory
  - 15 Executors
- Runtime: 2h 31m 53s



-> Runtime in both systems is the same

Environments and Bursting

- Environments:
  - Needed to run notebooks on custom spark pool

    ->size and number of nodes for custom spark pool depends on Fabric capacity
  - Select Env in Spark Notebook
- Bursting:
  - borrow capacity units from a future time interval (e.g. idle time)
  - Allows use of 3x more vCores than vCores purchased
    - e.g. start 3 jobs in parallel on the same spark pool
    - Starting a 4th job would raise **TooManyRequestsForCapacity** error

Fabric Take Aways

- Using synapse spark notebooks does not require any major code changes
- Shortcuts are easy to use and are created quickly
- Spark Views in Lakehouse do not work the way as expected (maybe improvements will come in future) - temporary views are quick fix for now
- Lakehouse schemas are available but are in public preview
- Environments are necessary when to use custom pools

ACP CUBIDO
Digital Solutions



IT for innovators.
Analytics & Development


No Copy Activities
Easy to Work With
Faster = Cheaper
SUCCESS



IT for innovators.
Analytics & Development

THANK YOU

Data Community Austria Day 2025

Awesome Partner

Microsoft

Platinum

redgate

ACP IT for innovators.

Gold

b.telligent
smart data. smart decisions.

Lucient

Bronze

Tabular Editor

paiqo
The Platform & AI Company

Power BI Camp
www.linearis.at

Evaluations, evaluations...



https://evals.datagrillen.com/evals_vienna.aspx