Microsoft Azure

Azure Synapse Analytics - Workspace 技術編

日本マイクロソフト株式会社
カスタマーサクセス事業本部
データ & クラウド AI アーキテクト統括本部

Senior Cloud Solution Architect

宮原 誠治 (*Doku*)

Doku の経歴

Doku
@dokums

1991年 **大手 SIer に入社**
- メインフレーム全盛時代、ニューラル ネットワーク や オブジェクト指向について独学

1992年 **24時間×365日稼働の生産管理システム再構築**
- 「生産ラインを 10分止めると、1,000万円の損失だ！」と脅される

1999年 **国内最大規模の Oracle Parallel Server DWH 構築**

2002年 **世界最大規模の .NET & SQL Server 基幹システム構築の PM 兼アーキテクト**
- 2003年に、MS Corp. から国内企業として初めて WPC Award 受賞
- 「アーキテクチャ仮想化技術」を考案し、日米で特許取得

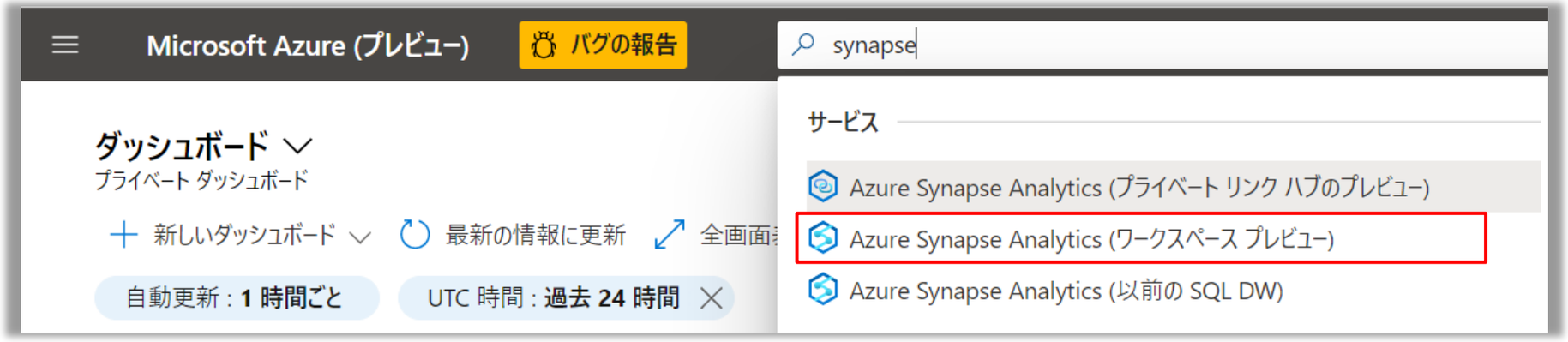
2006年 **MSKK にアーキテクトエバンジェリストとして入社**

～ 日立アライアンス部門 担当本部長、パートナーアーキテクト室長、などを経て

2015年 **Cloud Solution Architect (7人目)**

Synapse Analytics とは

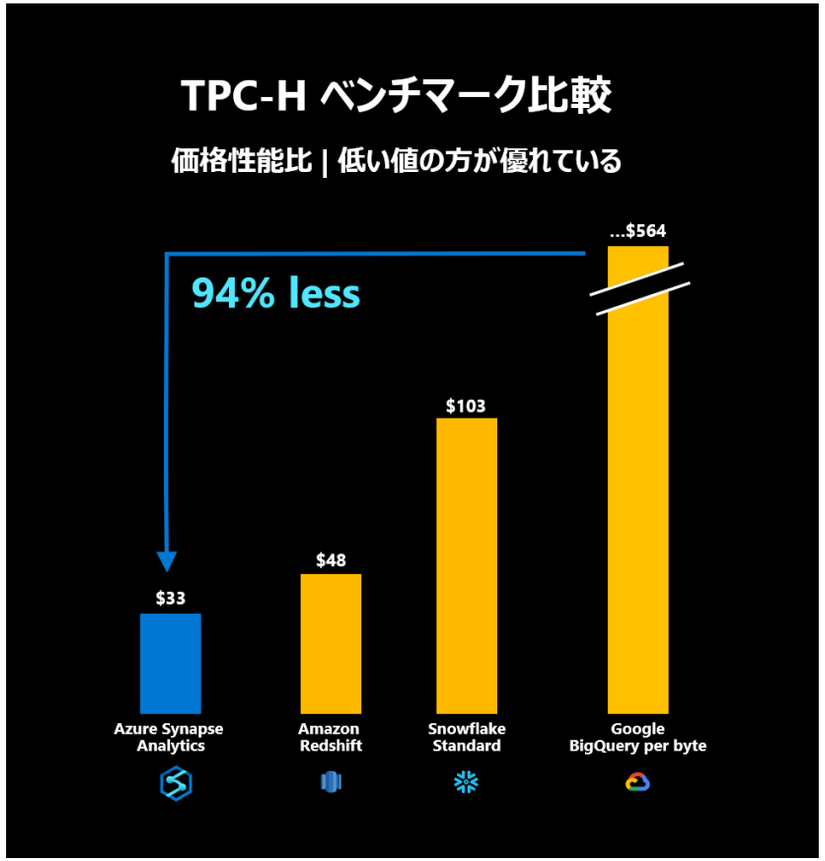
Synapse と検索すると。。。

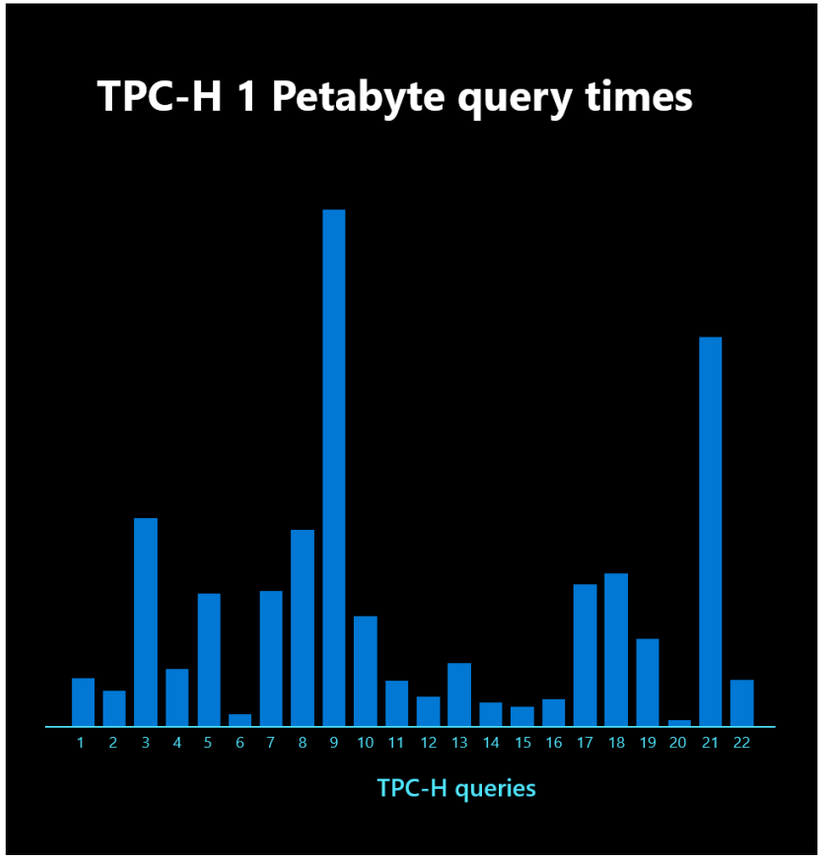


Azure SQL Data Warehouse としての進化

Azure Synapse Analytics (Formerly SQLDW) のこれまでの目標

- 性能効率の追求、スケーラビリティの拡大
- データの大容量化・無制限化

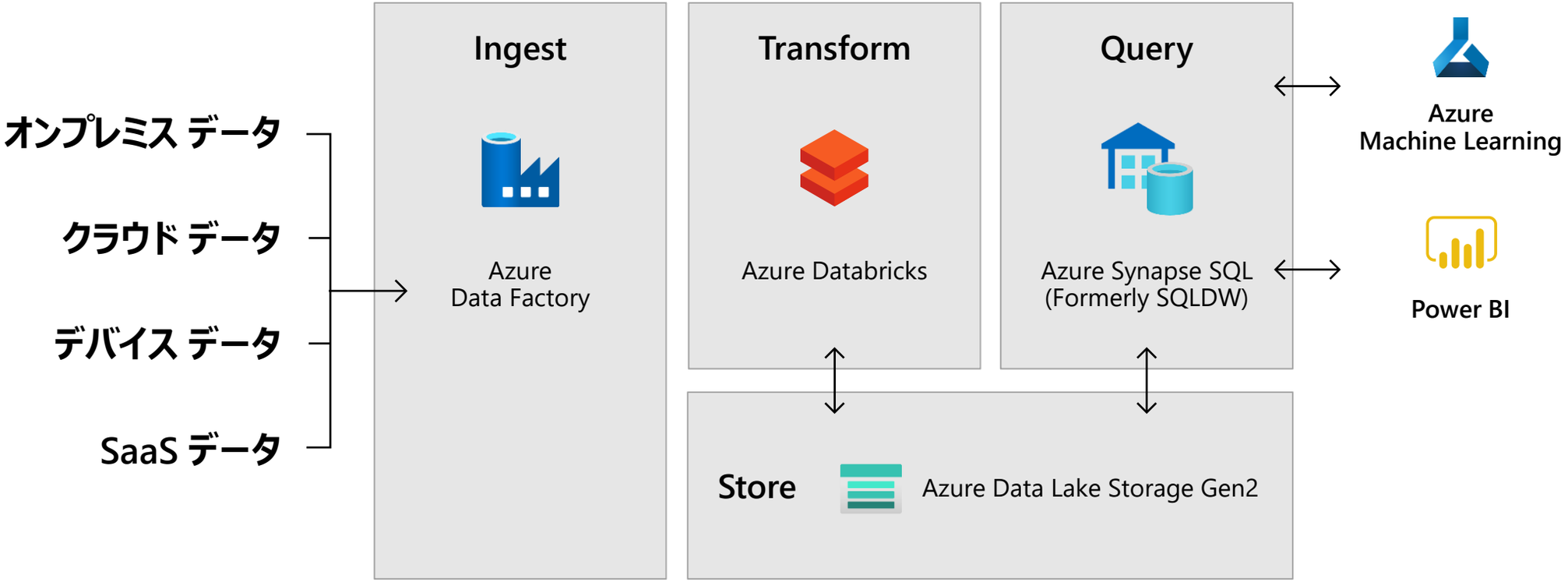




ペタバイト スケールにおける全ての TPC-H クエリを実行できた最初で、かつ、唯一のクラウド分析プラットフォーム

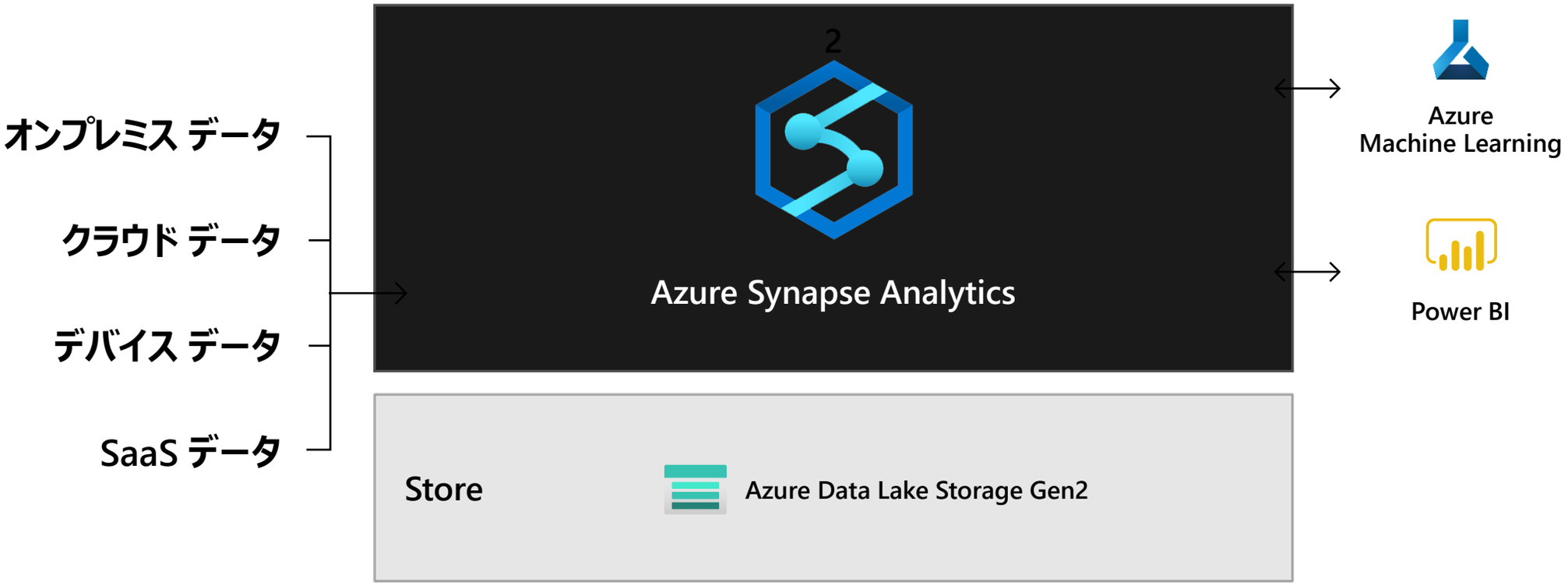
GigaOm TPC-H benchmark report, January 2019, “GigaOm report: Data Warehouse in the Cloud Benchmark”

新しい Azure Synapse Analytics としてのイノベーション



新しい Azure Synapse Analytics としてのイノベーション

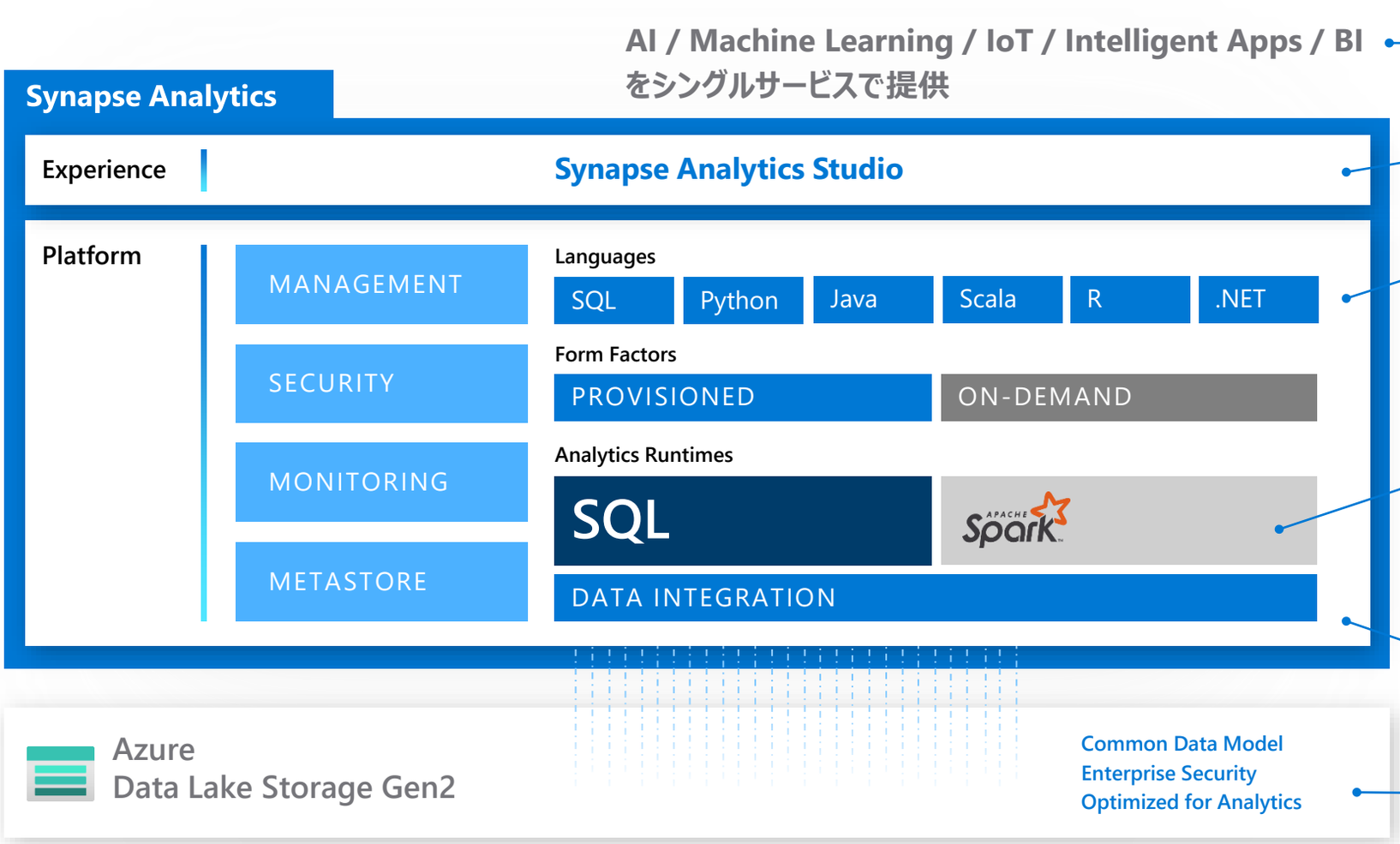
データ ウェアハウス と ビッグデータ分析システムを**融合させ**、
あらゆるデータから、驚異的なスピードでインサイトを提供する
無制限の分析サービス



Synapse Analytics のアーキテクチャ

Azure Synapse Analytics の論理アーキテクチャ

BI, AI と継続的なインテリジェンスの為の統合データプラットフォーム



あらゆるスケールの分析ワークロードを処理できるように設計

コードフリー & コードファーストの SaaS 開発者エクスペリエンスを提供

様々な分析ワークロードに適した**多様な**言語をサポート

統合された Analytics Runtime をプロビジョニング型とサーバーレス型 (オンデマンド) で提供

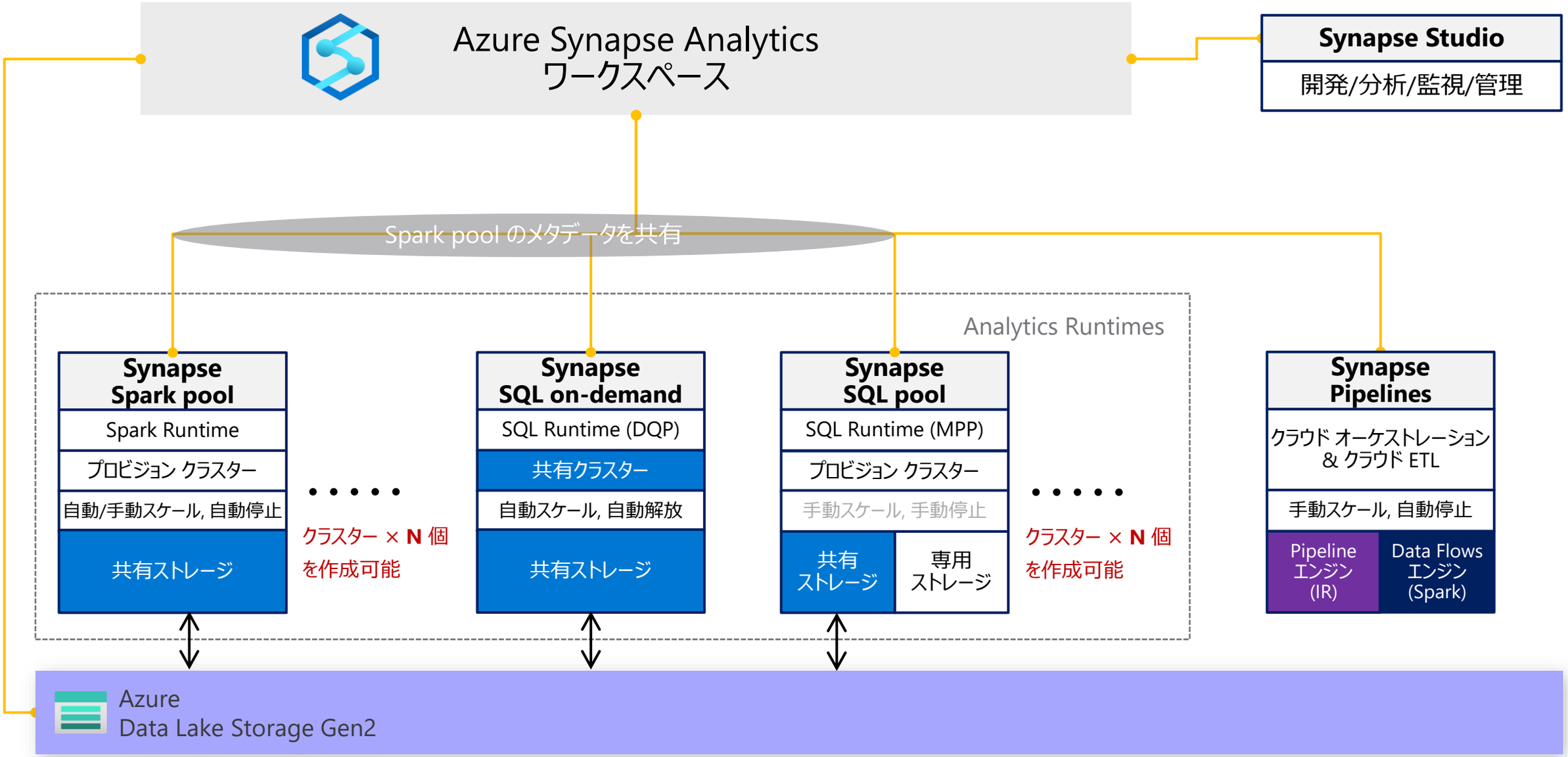
SQL Analytics : T-SQL バッチ, ストリーミング, インタラクティブ処理に適応

Spark : Python, Scala, R, .NET によるビッグデータ処理に適応

マネジメント, セキュリティ, モニタリング, メタストアに対して**統合プラットフォーム**を提供

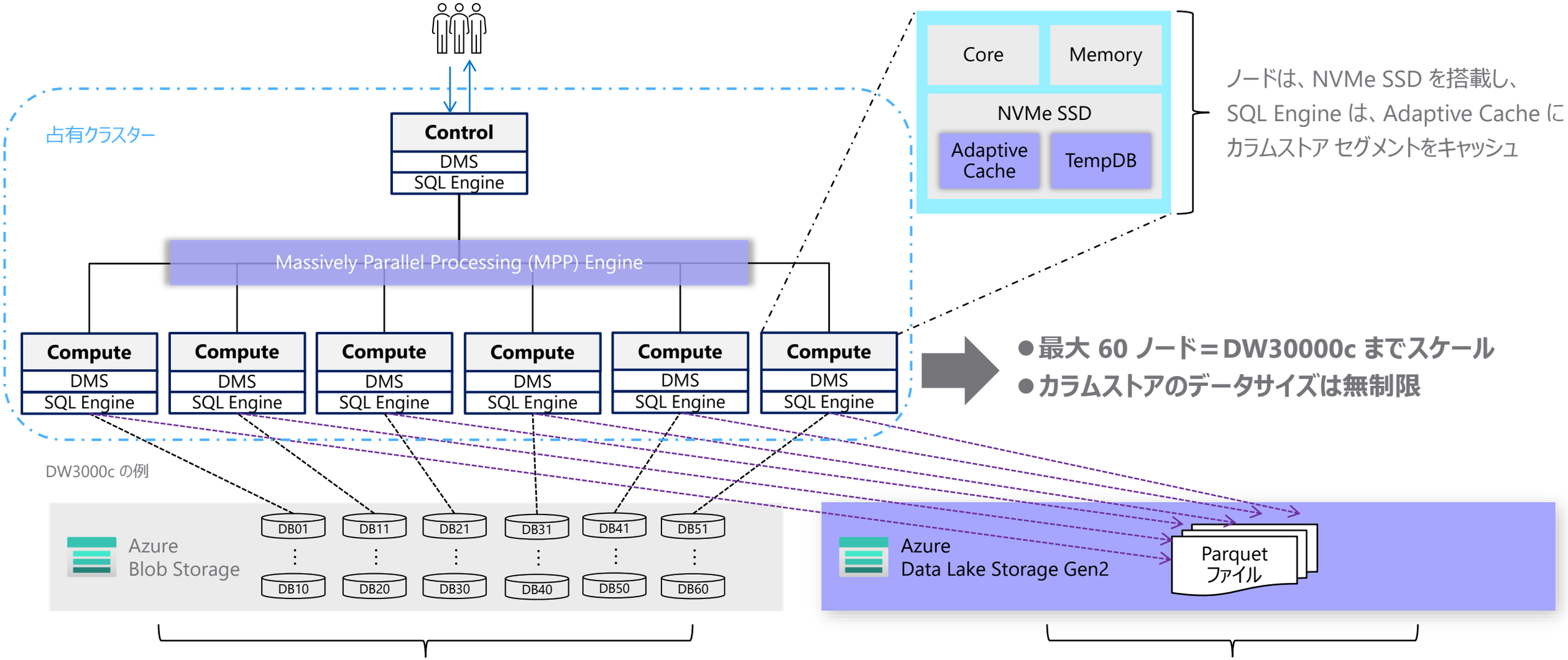
Data Lake Storage との統合、
Common Data Model の採用

Azure Synapse Analytics の物理アーキテクチャ



※現在、Spark pool から SQL Analytics にメタデータ共有可能なのは、Parquet ファイル形式のマネージドテーブル/外部テーブルで、SQL pool への共有は Gen3 で実現予定

Synapse SQL pool のアーキテクチャ



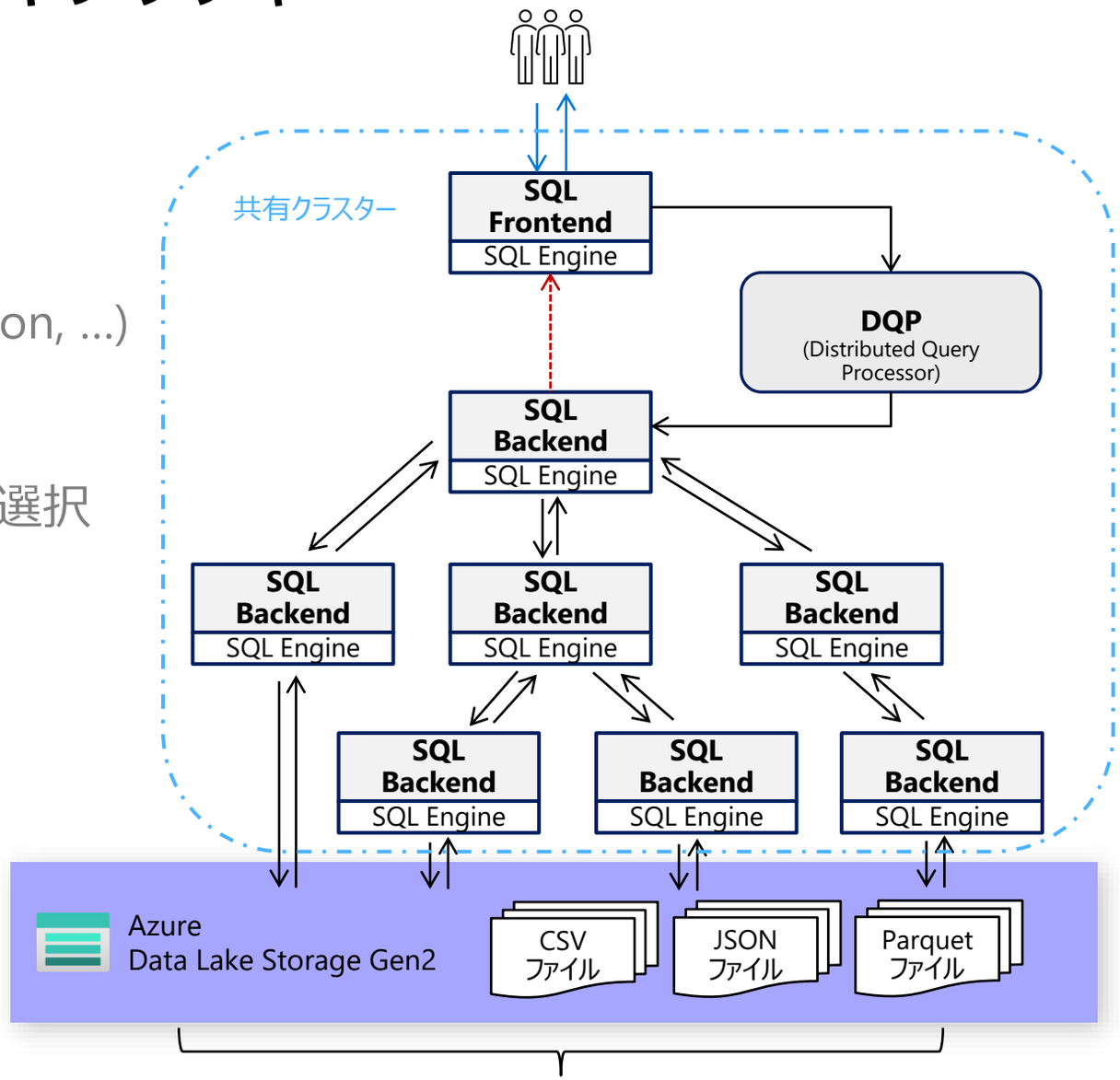
Azure Storage 上ではあるが、SQL pool の占有ストレージとして
内部的に 60 個のデータベースを保持し、Compute Node が均等に分担する

従来の Polybase のように一度 TempDB に読み込む方式とは異なり、
Fast Parquet 機能※による高速アクセスを実現

※Fast Parquet は、現在、Private Preview 段階。この機能がリリースされれば、メタデータ共有も SQL pool に拡張される予定

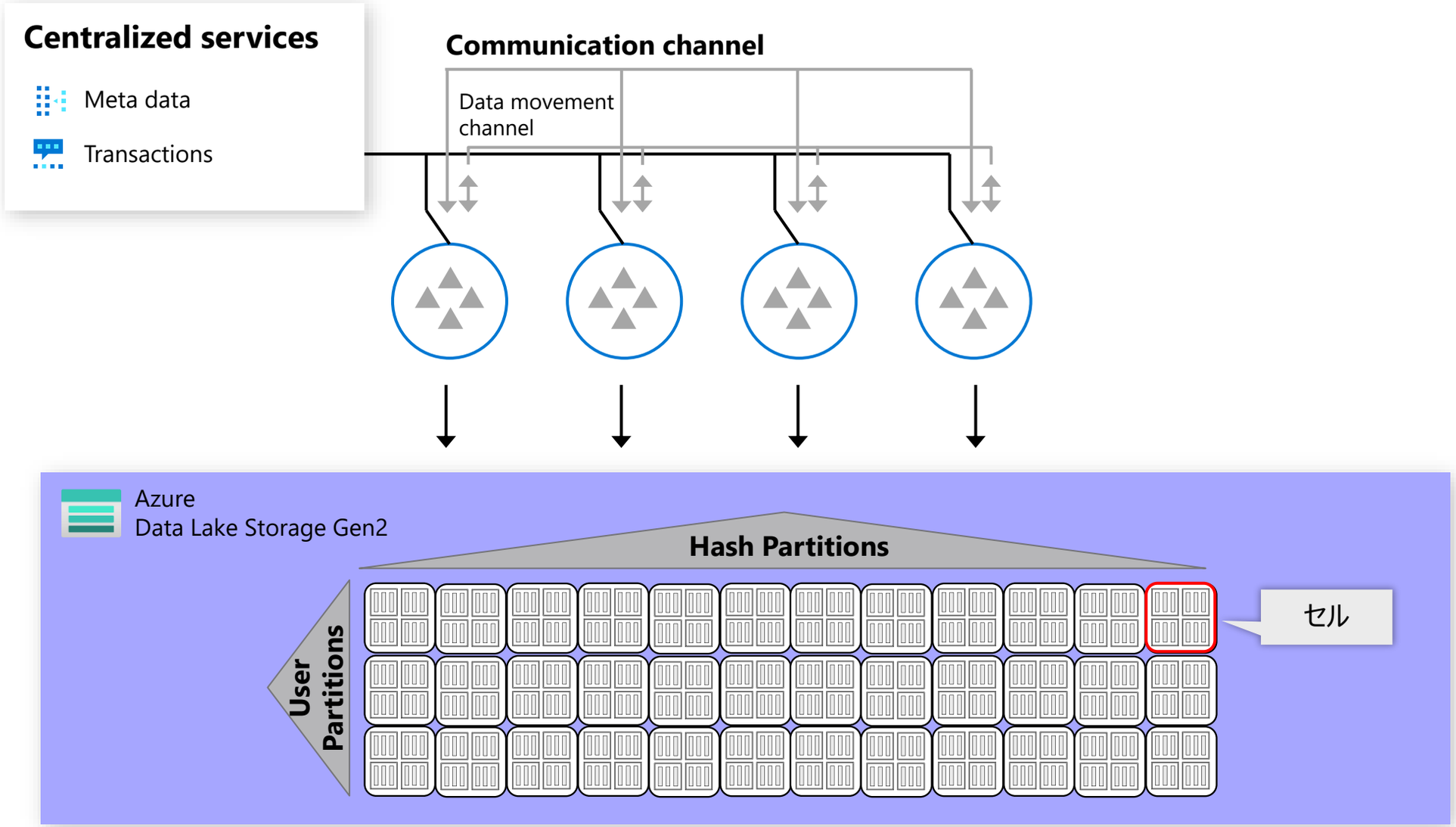
Synapse SQL on-demand のアーキテクチャ

- SQL Frontend
  - メタデータ
  - セキュリティ
  - クエリの単純化 (filter pushdown, partition elimination, ...)
- DQP (Distributed Query Processor)
  - 分散実行プランを計算し、最も低い推定コストの 1 つを選択
  - ユーザークエリを T-SQL フラグメント (タスク) に分解
  - フォルト トレラント
  - オートスケール
- SQL Backend - 完全なステートレス
  - タスクの実行
  - 結果をペアレントに返却

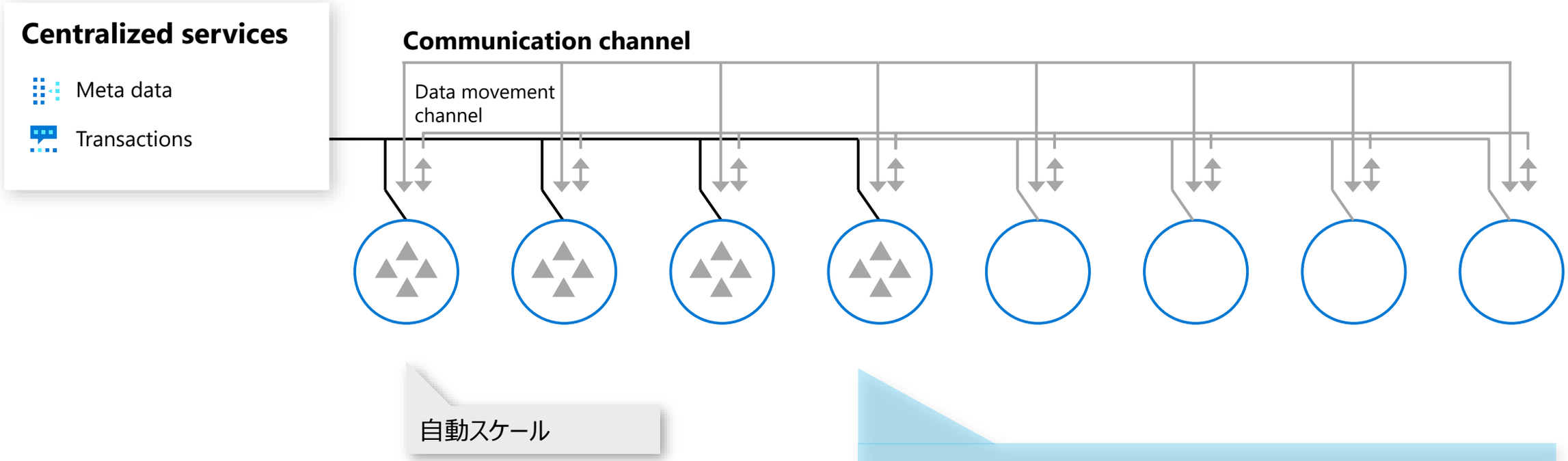


Azure Data Lake Storage Gen2 上にあるファイル (現在、CSV, JSON, Parquet をサポート) を対象としており、SQL pool のような占有ストレージではない

DQP：状態とコンピュートの分離

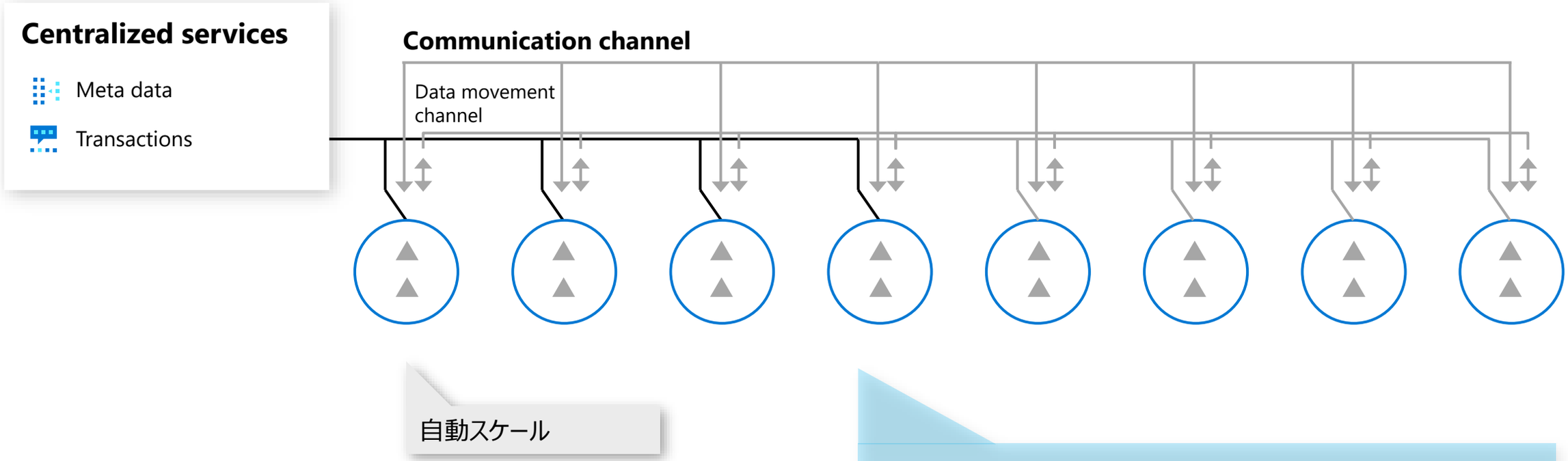


DQP：自動スケール



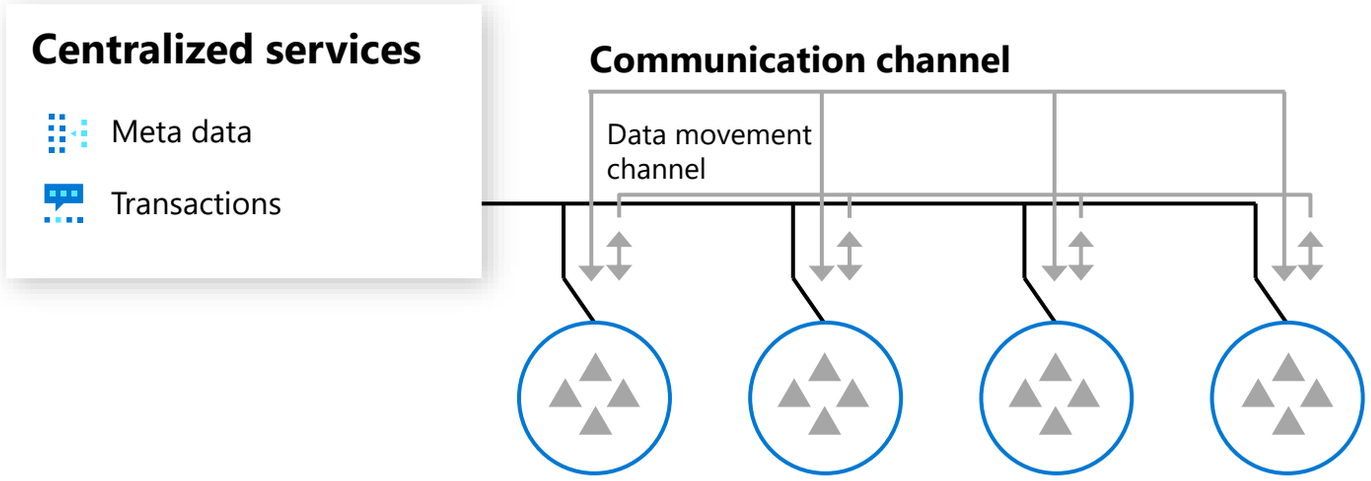
- 増分単位は、ノード
- タスクは以前のトポロジーで継続実行
- スケールは、セルの数と同じタスク数に制限

DQP：自動スケール

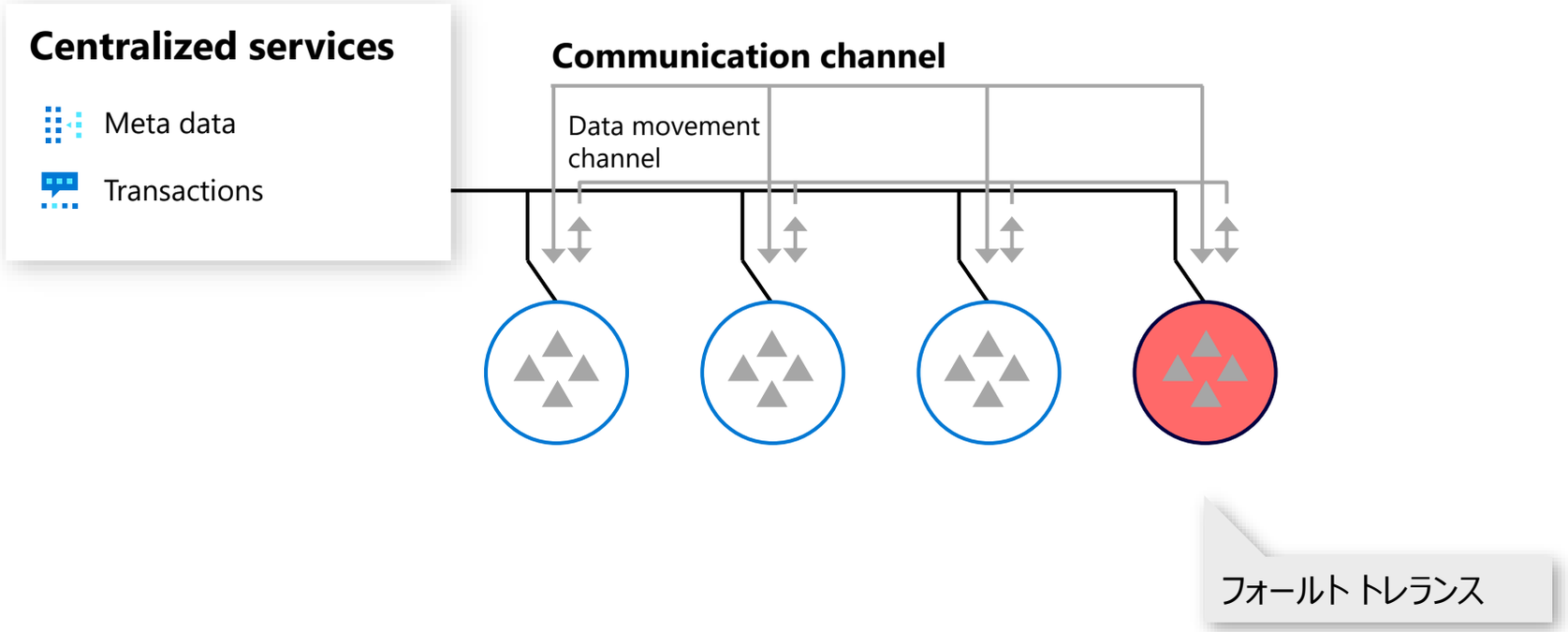


- 増分単位は、ノード
- タスクは以前のトポロジーで継続実行
- スケールは、セルの数と同じタスク数に制限

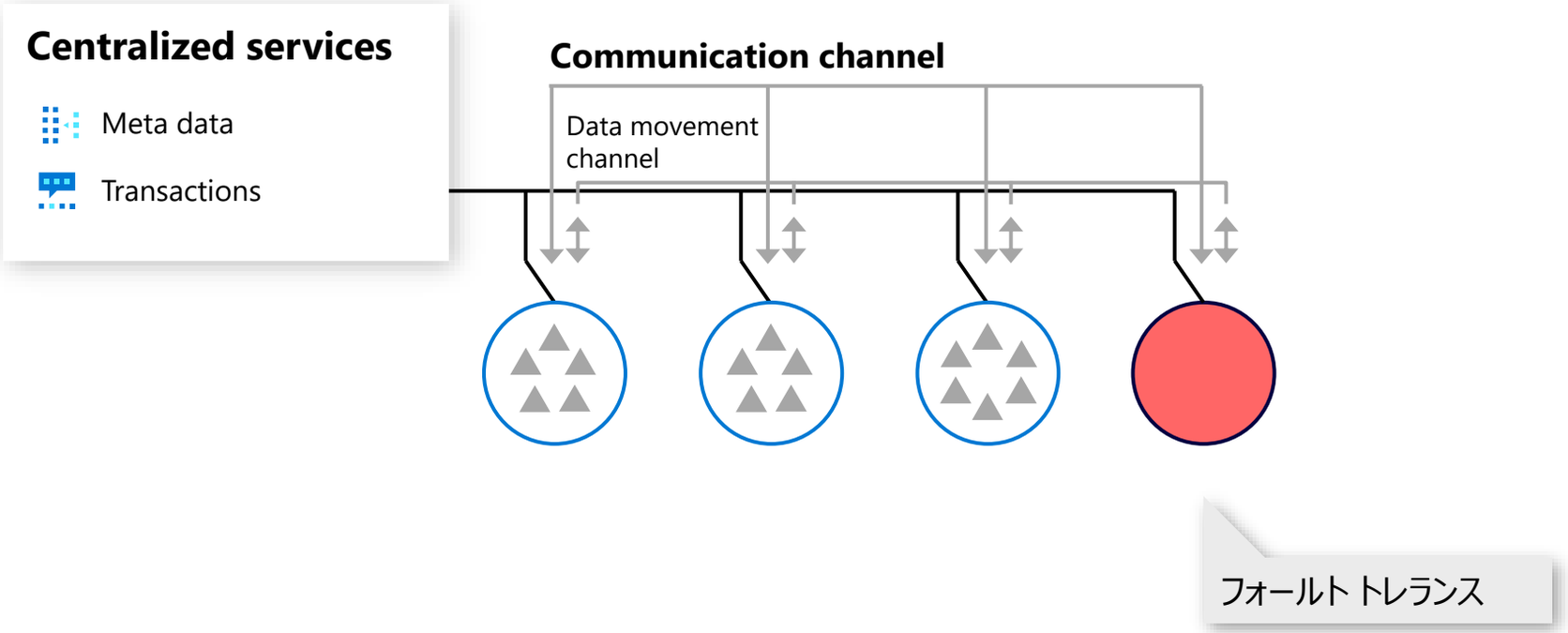
DQP : フォールト トレランス



DQP : フォールト トレランス



DQP : フォールト トレランス

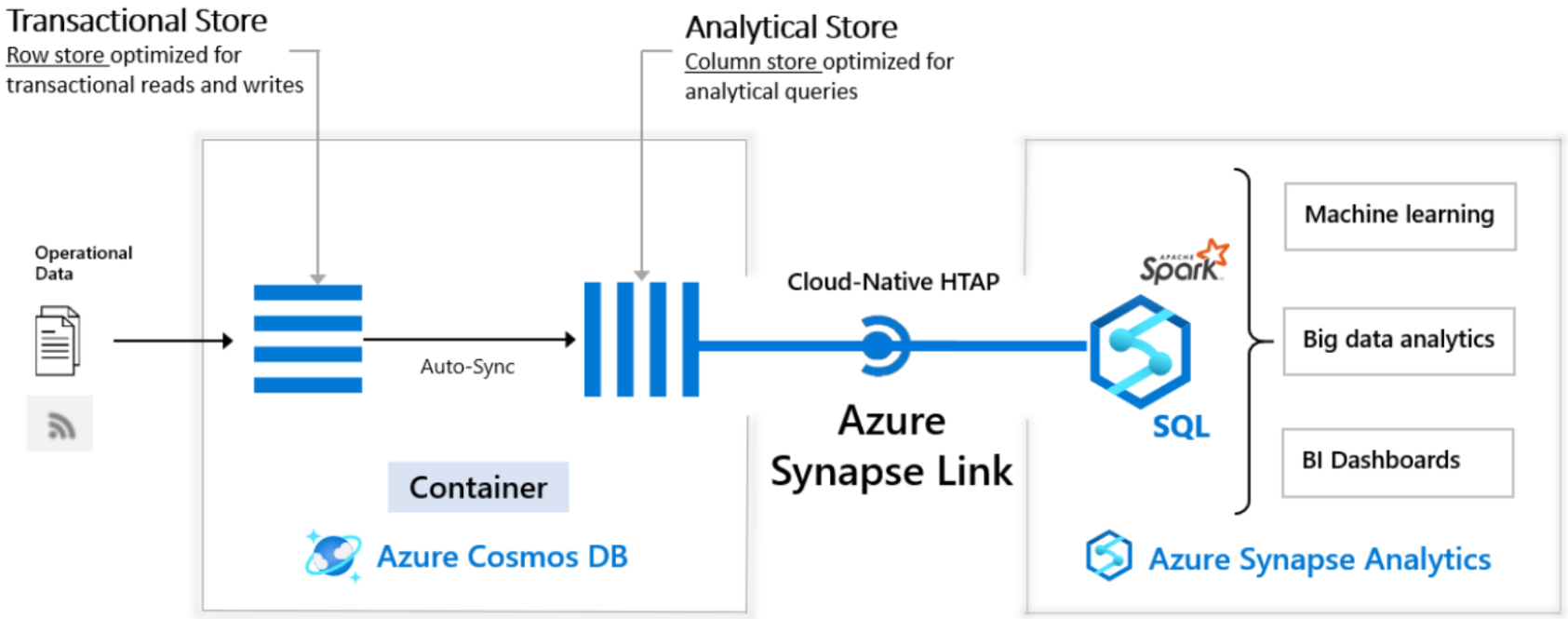


- SLA の向上
- ペタバイト スケールの性能

Azure Synapse Link

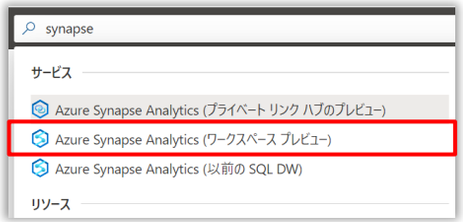
Azure Synapse Analytics をより強固にする Cloud Native HTAP (Hybrid Transactional & Analytical Processing) に対応した新しい機能

- Transactional ストア (OLTP DB) の機能として、カラムストア型の Analytical ストアに自動同期
- ETL 無し (データの 2 重持ちが不要) で Synapse に統合
- Azure Cosmos DB (プレビュー), Azure SQL Database (予定), Azure Database for PostgreSQL / MySQL (予定)



Synapse Workspace & Synapse Studio

Synapse Workspace の作成



推奨の設定

- 「ネットワーク」タブの「マネージド仮想ネットワークの有効化」をチェックしておく（セキュリティ上重要）

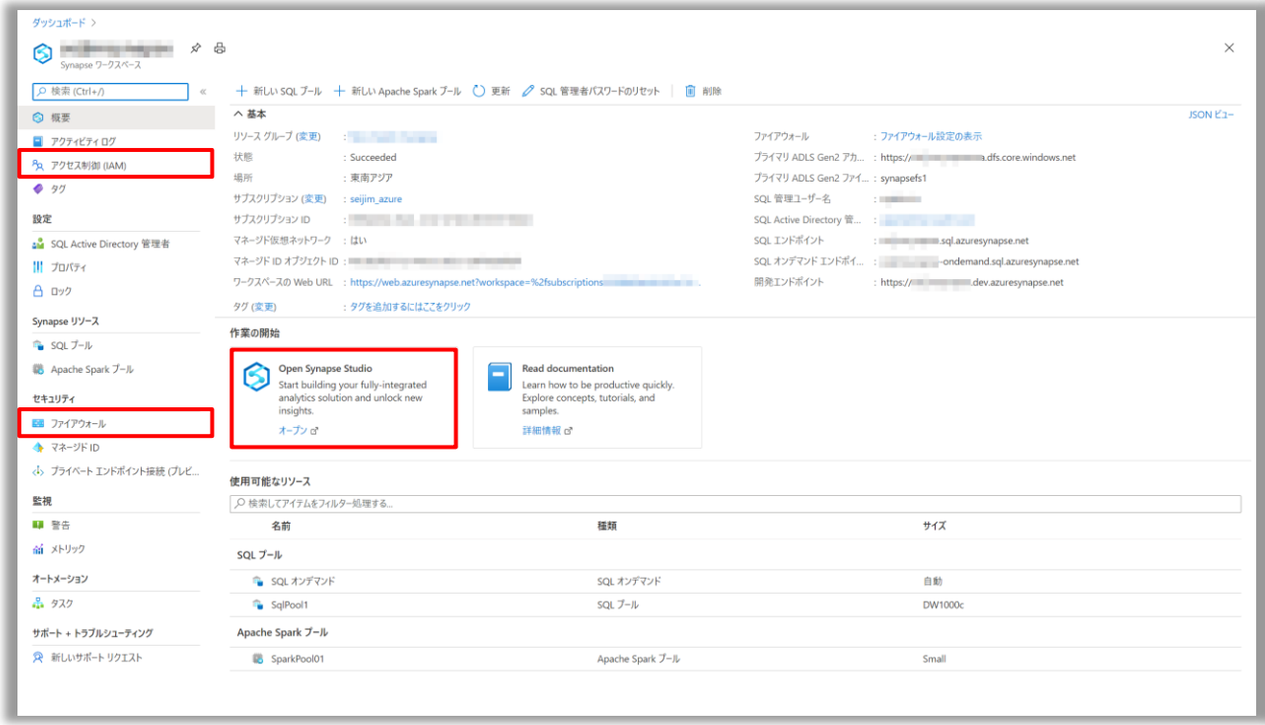
Azure Portal

Synapse Workspace と Synapse Studio

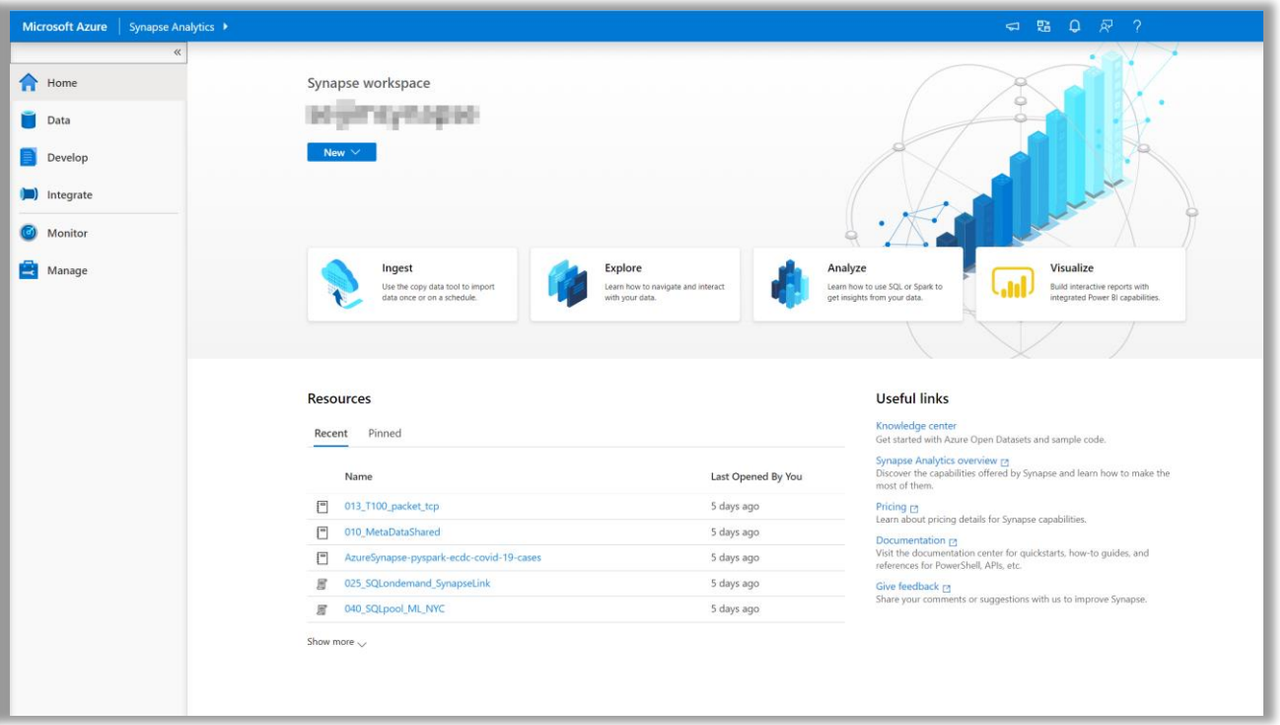
Synapse Workspace 設定の注意点

- 「アクセス制御 (IAM) 」で、ワークスペースにアクセス可能な AAD グループ/ユーザーを設定
  - ⇒ SQL pool や Spark pool を Synapse Studio で新規作成したい場合：ワークスペースの「所有者」/「共同作成者」の権限が必要
- 「ファイアウォール」で、Synapse Studio にアクセスするデバイスを許可

Azure Portal



Synapse Studio



Synapse Studio
Demo

Synapse Spark pool

Synapse Spark の特徴

特徴

- ビッグデータ処理機能
- スケールアウト型の超並列データ分析エンジン
- Spark との完全互換
- Spark クラスター間、SQL Analytics へのメタデータ共有

利点

- プロビジョニング型 (コンピューティング課金)
- 自動スケール/自動停止機能を提供
- データは Data Lake に存在し、データロード不要

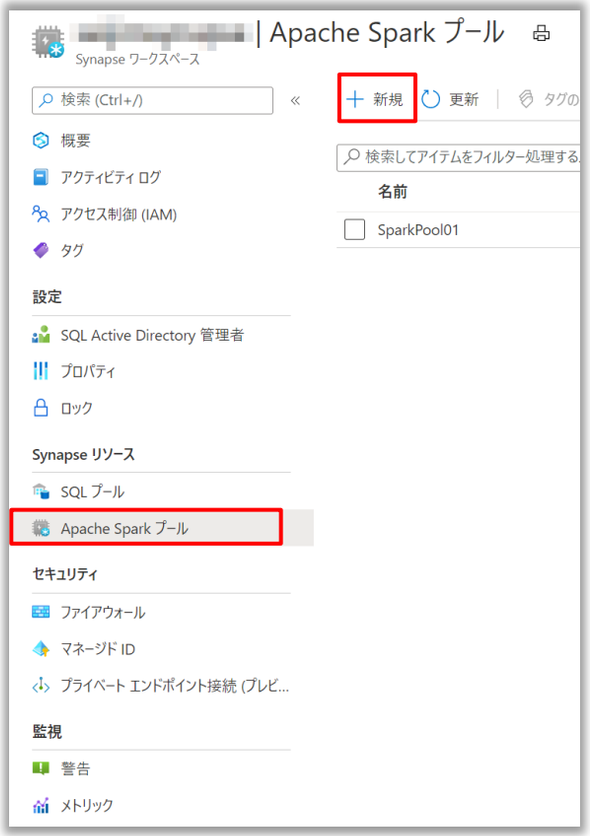
適用領域

- Data Lake 上のデータに対する大規模な加工処理
- データ サイエンティスト向けの分析・加工・機械学習/AI プラットフォーム

Synapse Spark pool の作成

Azure Portal、もしくは、Synapse Studio から Spark pool を作成

Azure Portal

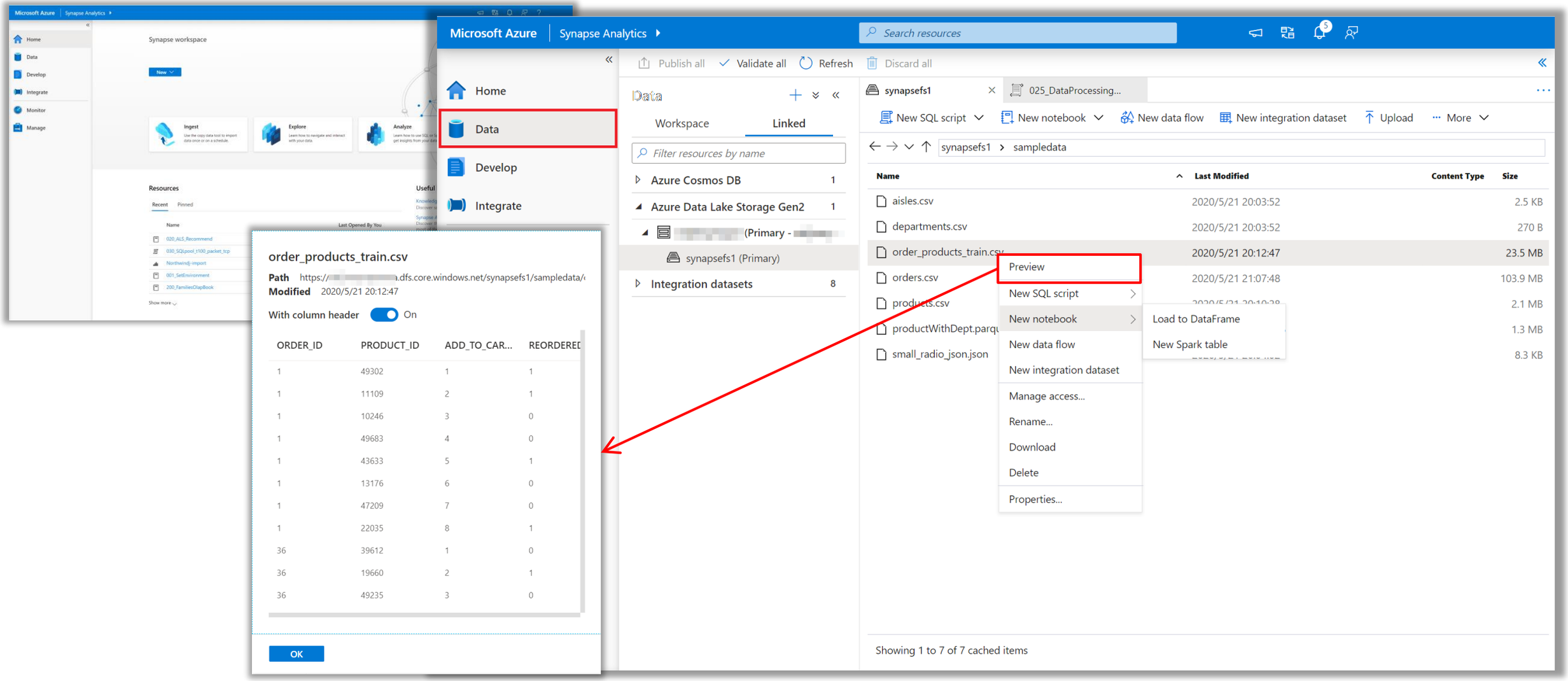


ADLS Gen2 上のファイルを可視化・分析

ファイルデータをクイックに探索・参照して、次のアクションへ

Synapse Studio を起動

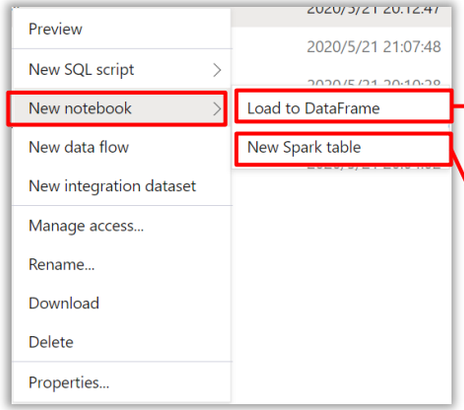
ADLS Gen2 ファイルシステム上のファイルを選択 ⇒ 右クリックで、すぐにアクション可能



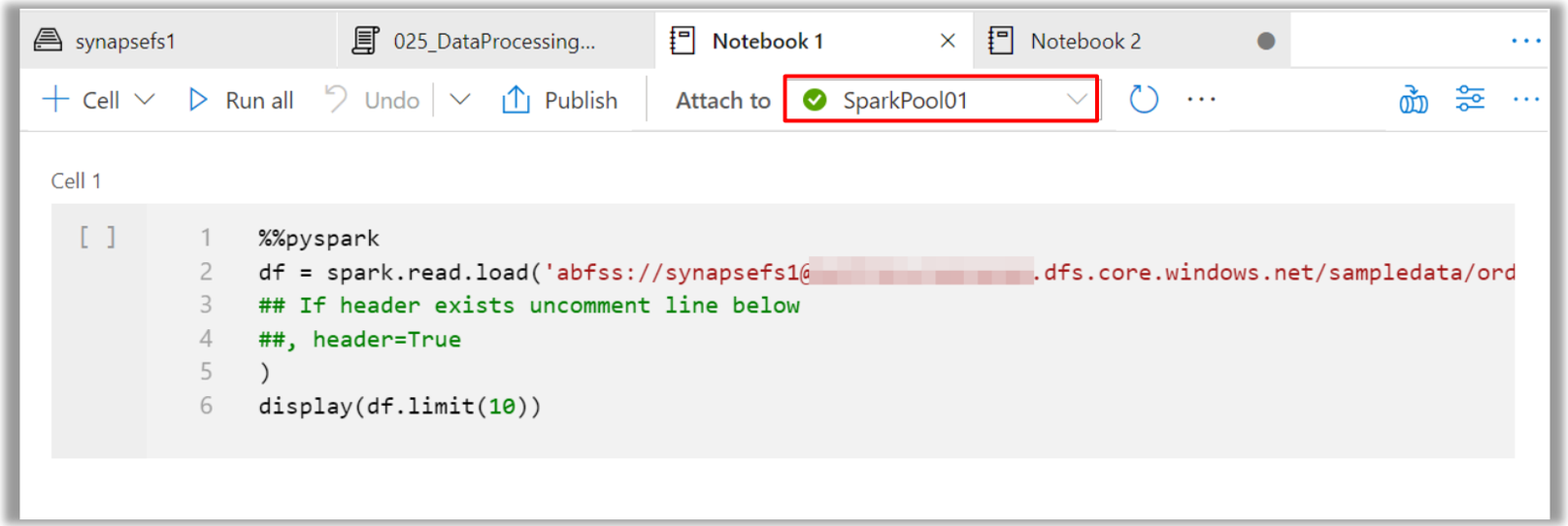
ADLS Gen2 上のファイルを可視化・分析

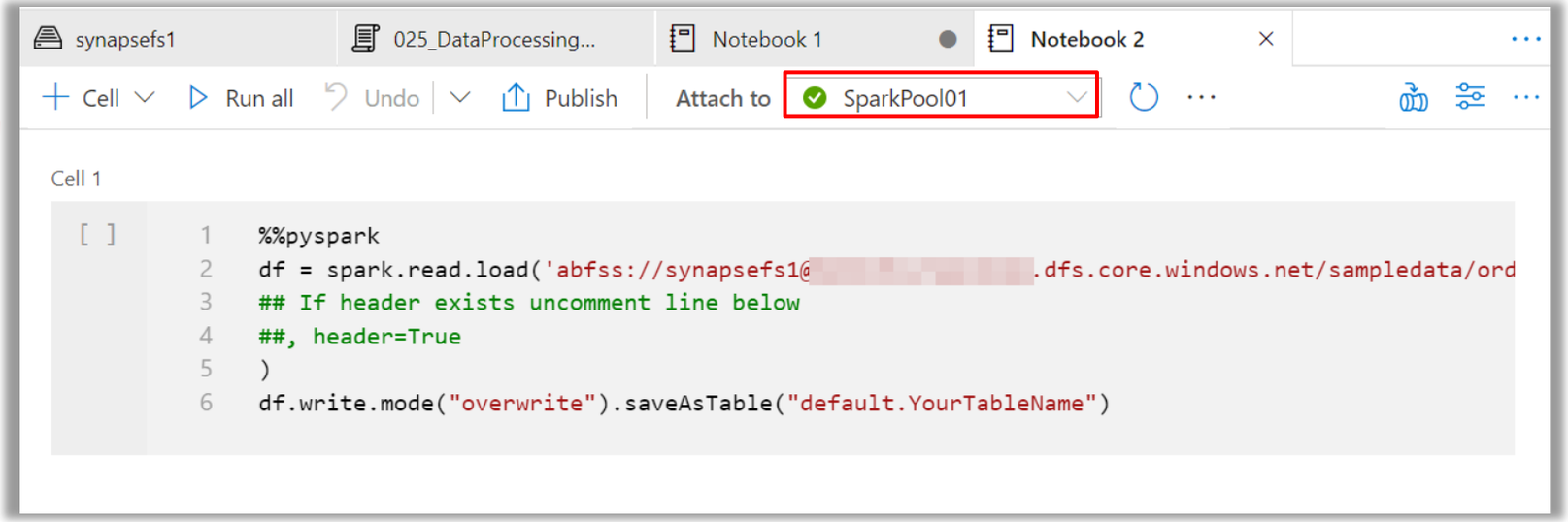
ファイルデータを可視化・分析する為の作業を徹底的に自動化・省力化

右クリック メニュー



Notebook のデータアクセス
コードを自動生成



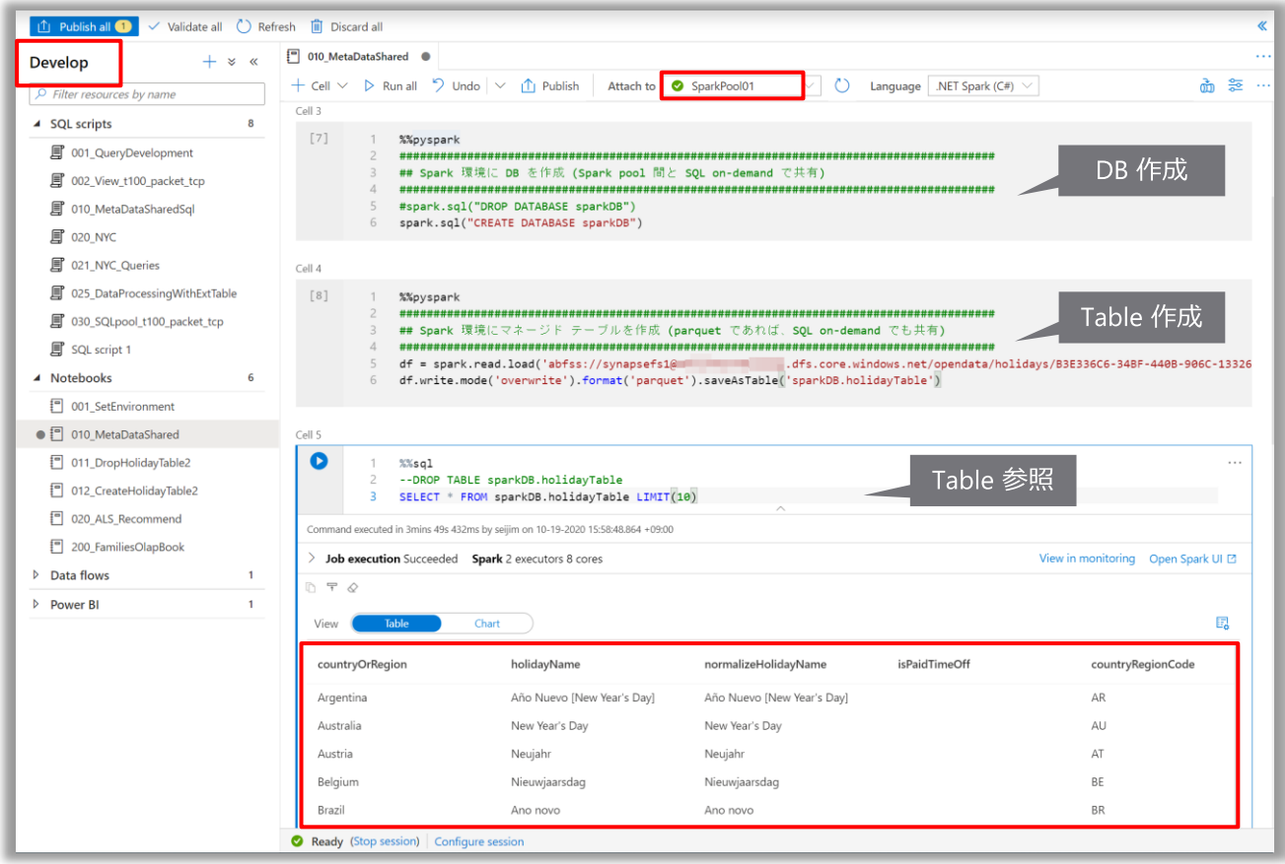


メタデータを Spark クラスター間、Synapse SQL に共有

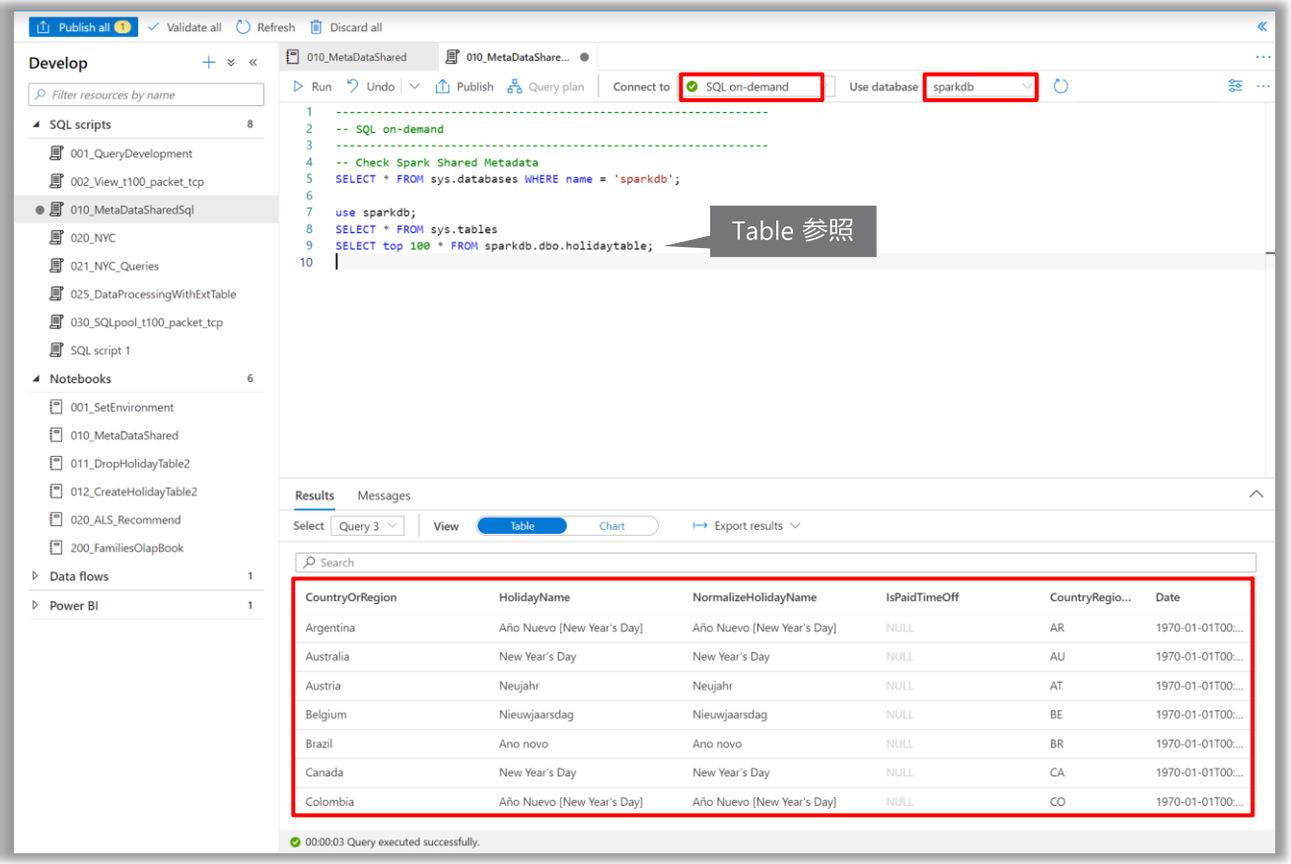
既定 (default) および作成されたデータベースとテーブルのメタデータは、Spark クラスター間で共有され、Synapse SQL on-demand に対してもデータベースと Parquet 形式のテーブルが共有される

Synapse Studio - Develop Hub

Spark pool 上での操作



SQL on-demand 上での操作

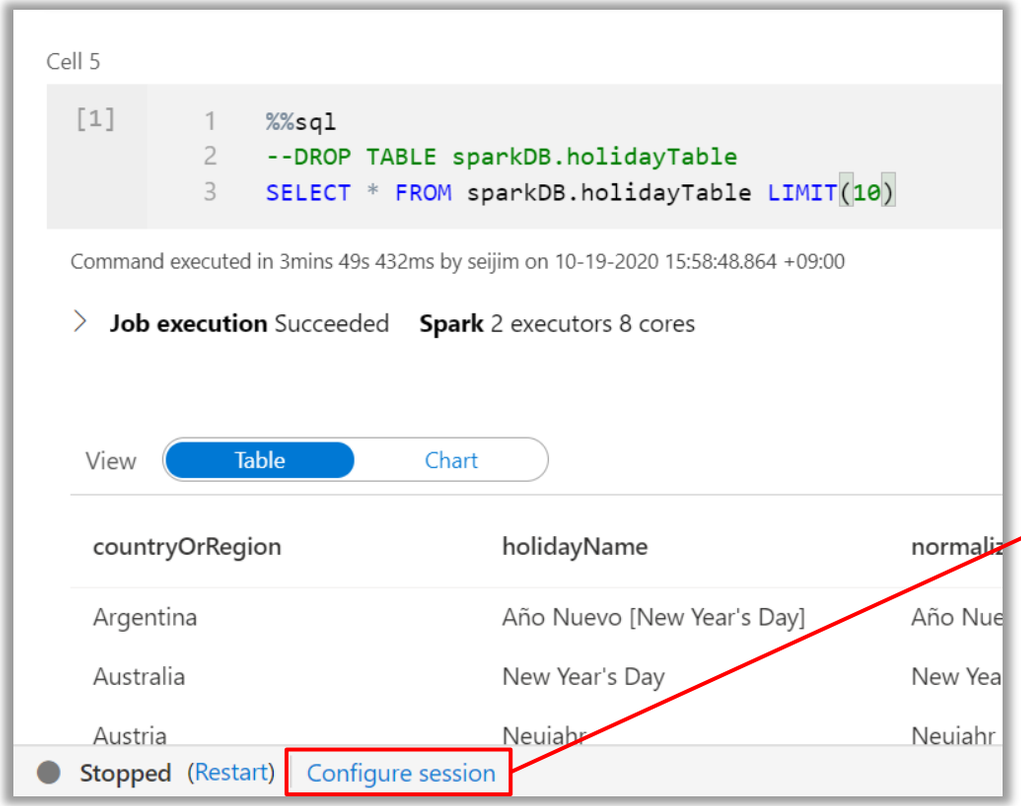


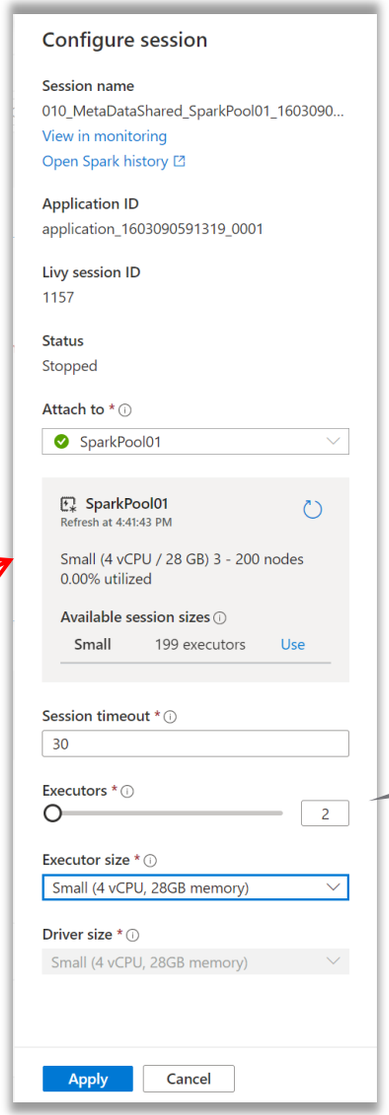
Notebook 実行における Spark session の設定

Notebook (Cell) を実行する Spark session の設定を個別に実施可能

Synapse Studio - Develop Hub

Spark pool 上での操作





Executor 数を変更して
性能調整を行うことが可能

クラスター規模による処理性能

Spark ML の ALS (協調フィルタリング) アルゴリズムを適用し、1 億件のトランザクション データを使ったモデル トレーニングについて、クラスター規模 (Executor 数) による処理性能結果を下記に示す

010_MetaDataShared | 010_MetaDataShared... | 020_ALS_Recommend

Cell | Run all | Undo | Publish | Attach to SparkPool01 | Language PySpark (Python)

Cell 4

[3]

```
%%pyspark
##########################################
## Load to DataFrame from internal table
##########################################
tranDf = spark.sql("select * from tran_table_int")

#######################################################################################
## Build the recommendation model using ALS on the training data
## Note we set cold start strategy to 'drop' to ensure we don't get NaN evaluation metrics
#######################################################################################
from pyspark.ml.recommendation import ALS

(training, test) = tranDf.randomSplit([0.8, 0.2])
als = ALS(maxIter=5, regParam=0.01, userCol="int_userId", itemCol="int_itemId", ratingCol="rating",
          coldStartStrategy="drop")
model = als.fit(training)
```

トレーニング データとしては、
80% (8,000万件) を使用

Azure の他の Spark サービスよりも
高速な結果となった

Command executed in 4mins 43s 340ms by seijim on 10-19-2020 17:14:33.788 +09:00

2 executors 8 cores
4 min 43 s

Job execution Succeeded Spark 2 executors 8 cores | View in monitoring | Open Spark UI

ID	Description	Status	Stages	Tasks	Submission Time	Duration
Job 5	isEmpty at ALS.scala:919	Succeeded	1/1	1/1 succeeded	5:09:50 PM, 10/19/20	18 sec
Job 6	count at ALS.scala:932	Succeeded	3/3	26/26 succeeded	5:10:08 PM, 10/19/20	54 sec
Job 7	count at ALS.scala:939	Succeeded	2/2	18/18 succeeded	5:11:02 PM, 10/19/20	27 sec
Job 8	count at ALS.scala:1031	Succeeded	12/12	120/120 succeeded	5:11:30 PM, 10/19/20	2 min 48 sec
Job 9	count at ALS.scala:1033	Succeeded	1/1	10/10 succeeded	5:14:19 PM, 10/19/20	11 sec

Command executed in 2mins 25s 230ms by seijim on 10-19-2020 17:24:41.644 +09:00

4 executors 16 cores
2 min 25 s

Job execution Succeeded Spark 4 executors 16 cores | View in monitoring | Open Spark UI

ID	Description	Status	Stages	Tasks	Submission Time	Duration
Job 8	isEmpty at ALS.scala:919	Succeeded	1/1	1/1 succeeded	5:22:16 PM, 10/19/20	7 sec
Job 9	count at ALS.scala:932	Succeeded	3/3	42/42 succeeded	5:22:24 PM, 10/19/20	26 sec
Job 10	count at ALS.scala:939	Succeeded	2/2	26/26 succeeded	5:22:50 PM, 10/19/20	12 sec
Job 11	count at ALS.scala:1031	Succeeded	12/12	120/120 succeeded	5:23:03 PM, 10/19/20	1 min 28 sec
Job 12	count at ALS.scala:1033	Succeeded	1/1	10/10 succeeded	5:24:32 PM, 10/19/20	7 sec

Command executed in 1mins 51s 94ms by seijim on 10-19-2020 17:45:45.897 +09:00

8 executors 32 cores
1 min 51 s

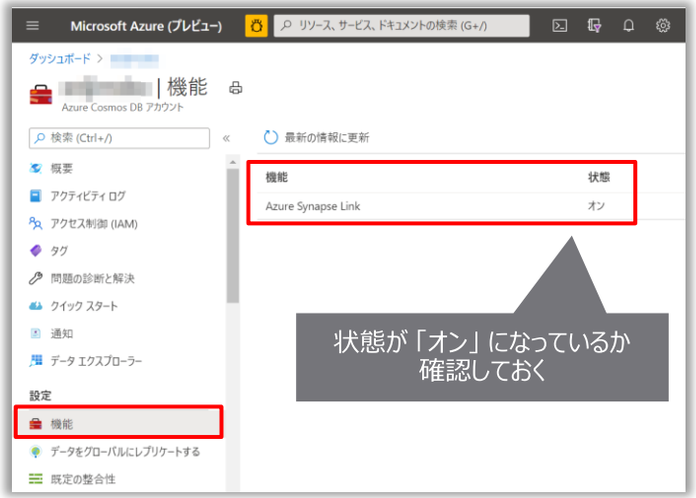
Job execution Succeeded Spark 8 executors 32 cores | View in monitoring | Open Spark UI

ID	Description	Status	Stages	Tasks	Submission Time	Duration
Job 10	isEmpty at ALS.scala:919	Succeeded	1/1	1/1 succeeded	5:43:54 PM, 10/19/20	4 sec
Job 11	count at ALS.scala:932	Succeeded	3/3	74/74 succeeded	5:43:58 PM, 10/19/20	18 sec
Job 12	count at ALS.scala:939	Succeeded	2/2	42/42 succeeded	5:44:16 PM, 10/19/20	9 sec
Job 13	count at ALS.scala:1031	Succeeded	12/12	120/120 succeeded	5:44:26 PM, 10/19/20	1 min 10 sec
Job 14	count at ALS.scala:1033	Succeeded	1/1	10/10 succeeded	5:45:37 PM, 10/19/20	5 sec

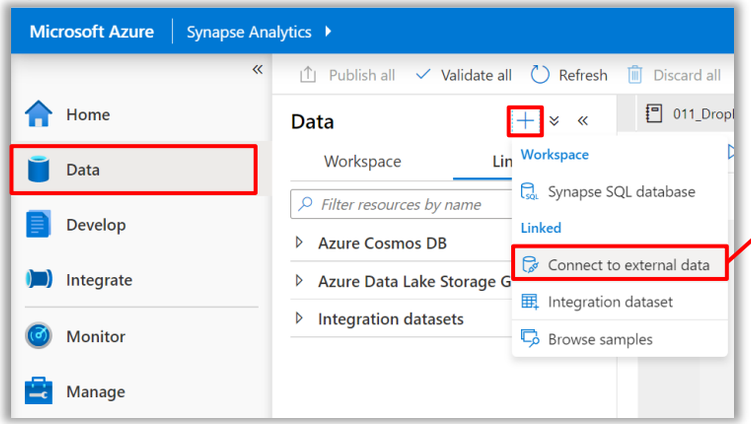
Synapse Link : Cosmos DB Analytical Store へのアクセス

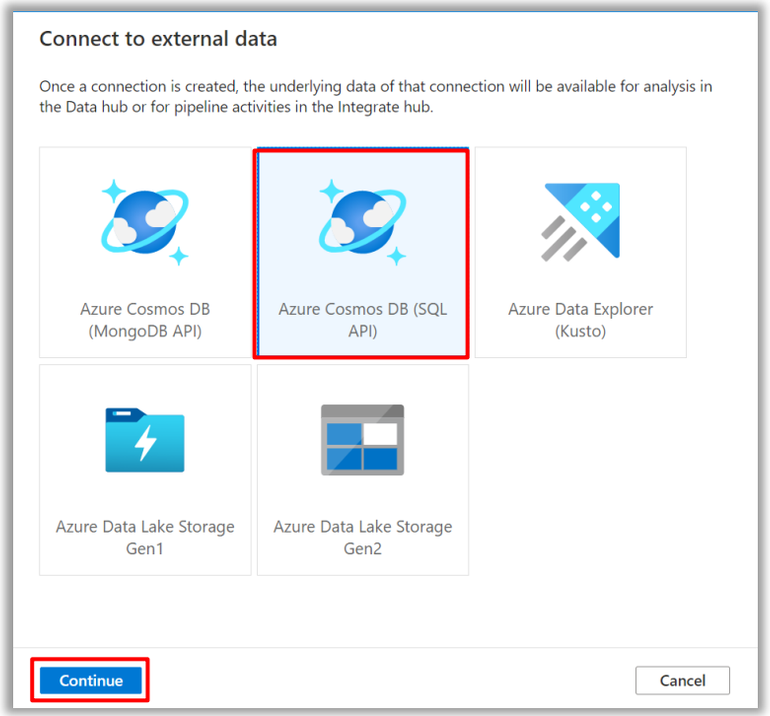
Cosmos DB Analytical Store への Linked サービスを作成

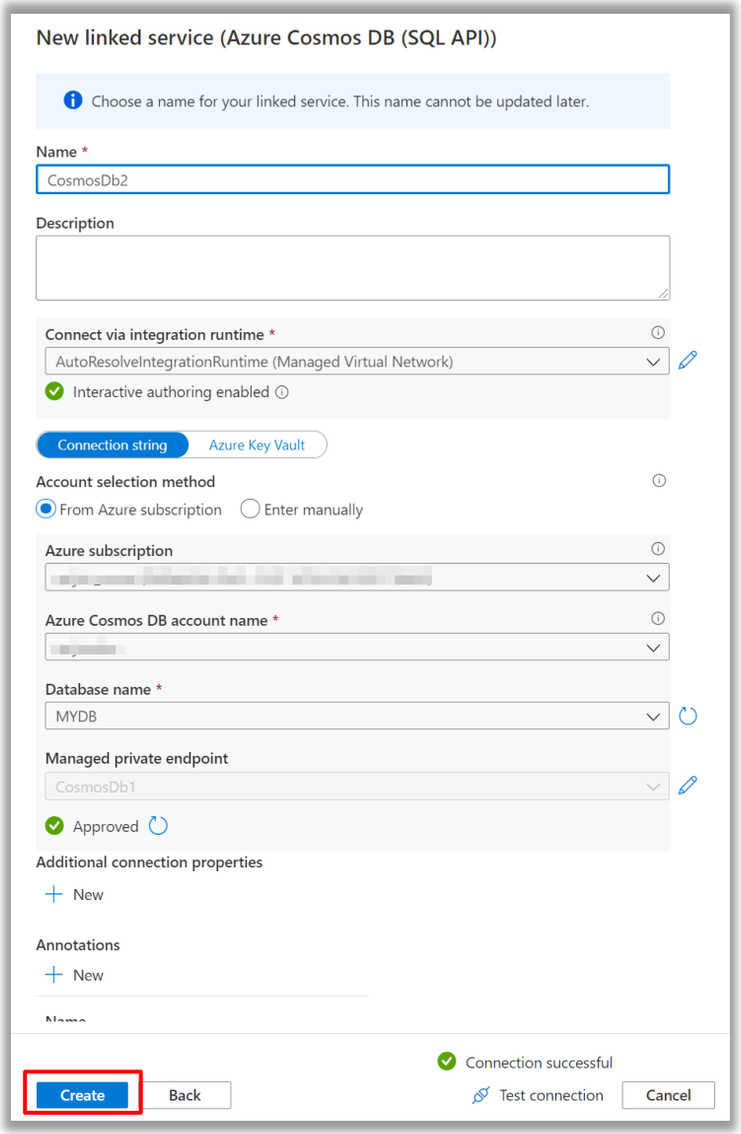
Azure Portal - Cosmos DB アカウント



Synapse Studio - Data Hub



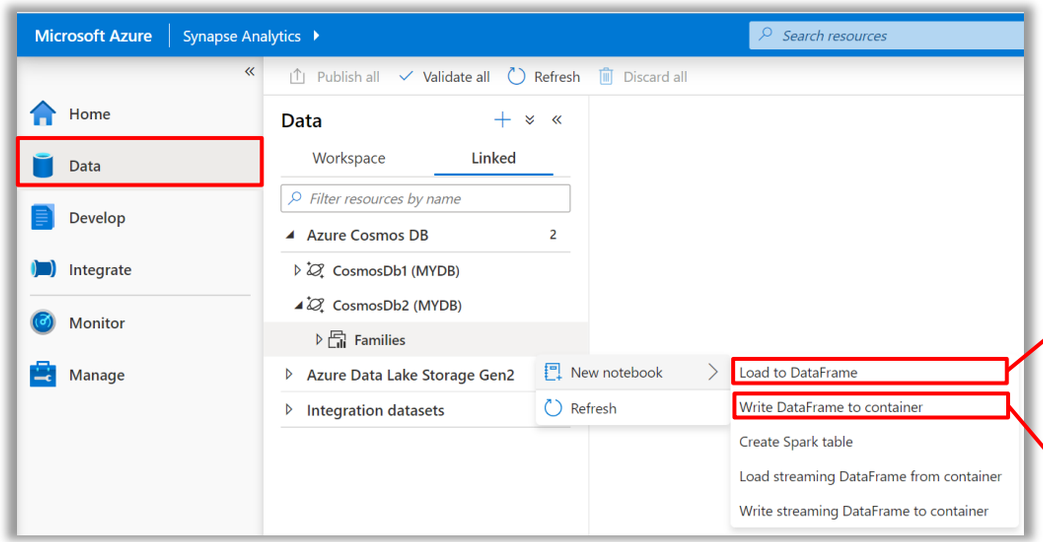


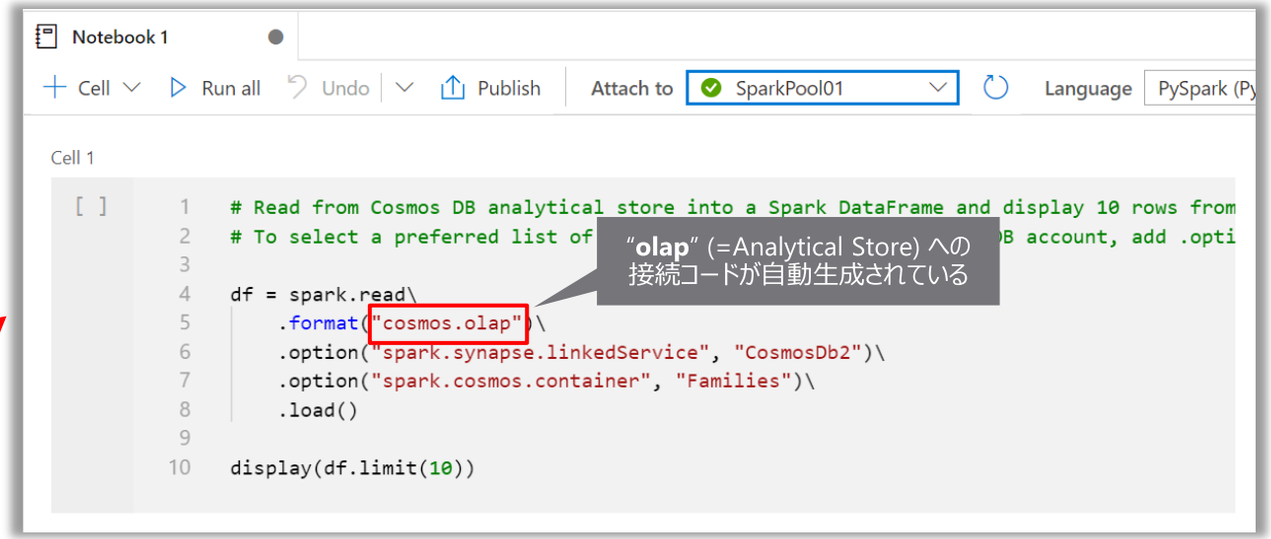


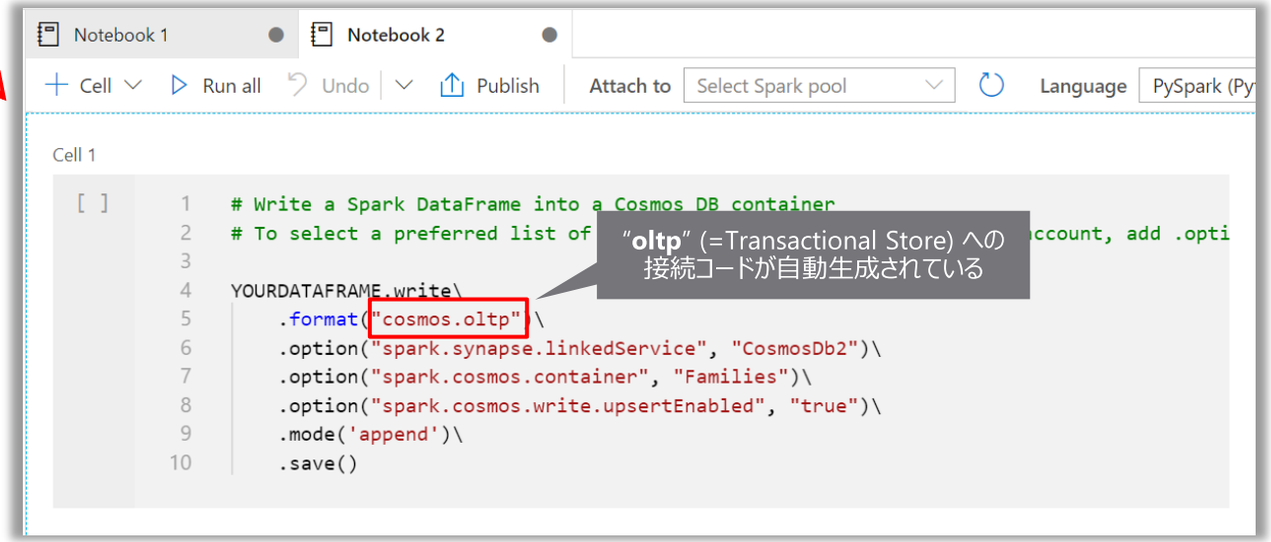
Synapse Link : Cosmos DB Analytical Store へのアクセス

Synapse Link を使った Notebook コードを自動生成

Synapse Studio - Data Hub







Synapse SQL on-demand

Synapse SQL on-demand の特徴

特徴

- データ ウェアハウス機能
- スケールアウト型の超並列データ分析エンジン
- T-SQL のサポート (SQL Server との高い互換性)
- Spark pool とのメタデータ共有

利点

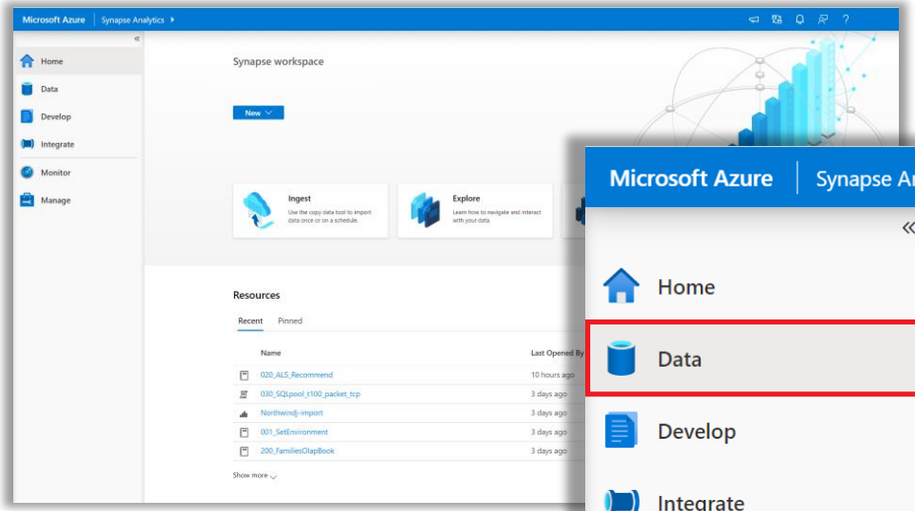
- サーバーレス クエリ (データ処理量課金)
- 管理オーバーヘッドが無い
- データは Data Lake に存在し、データロード不要

適用領域

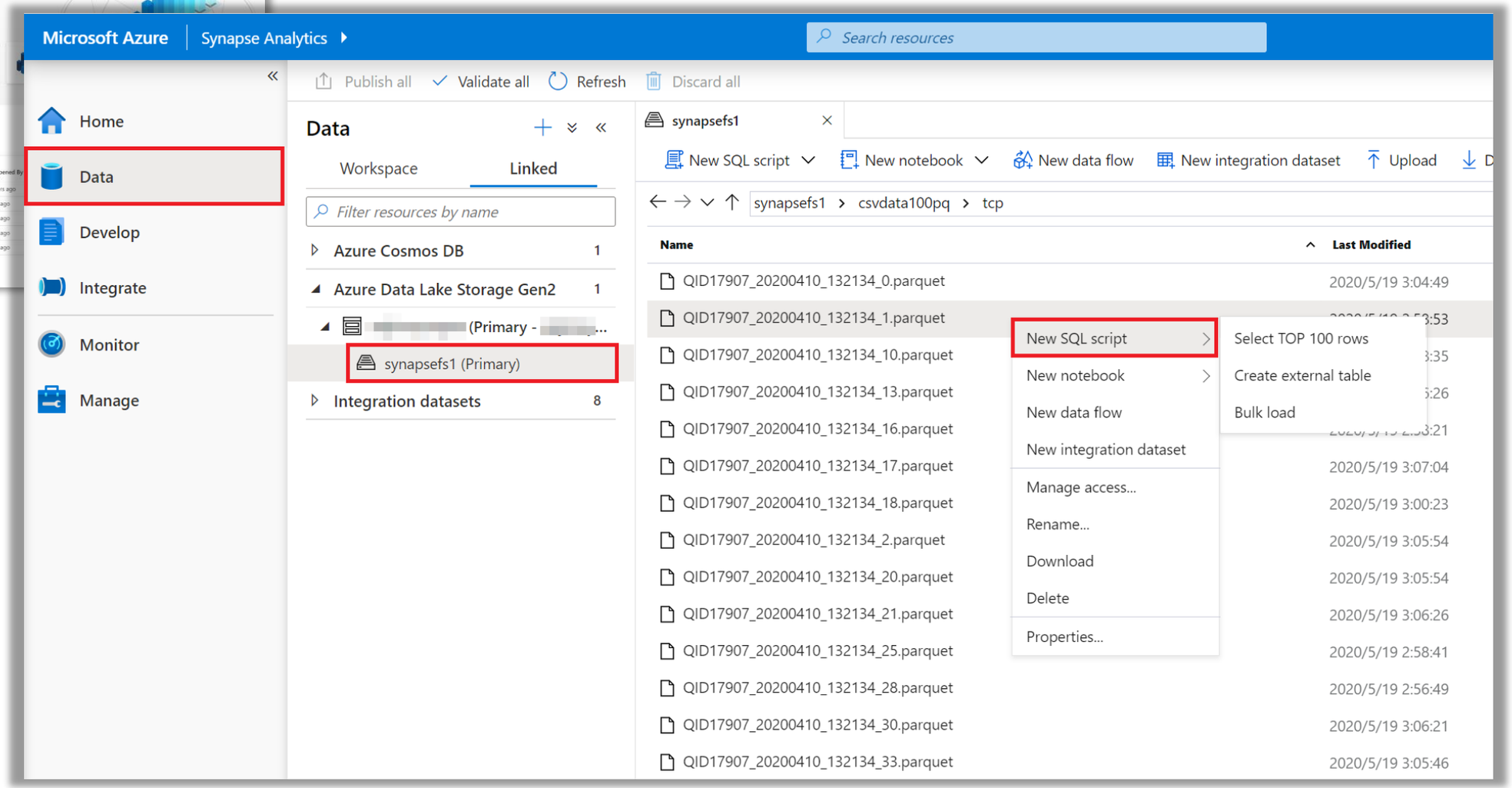
- Data Lake 上のデータに対する大規模な加工処理
- データ サイエンティスト向けの分析・加工プラットフォーム

ADLS Gen2 上のファイルを可視化・分析

Synapse Studio を起動



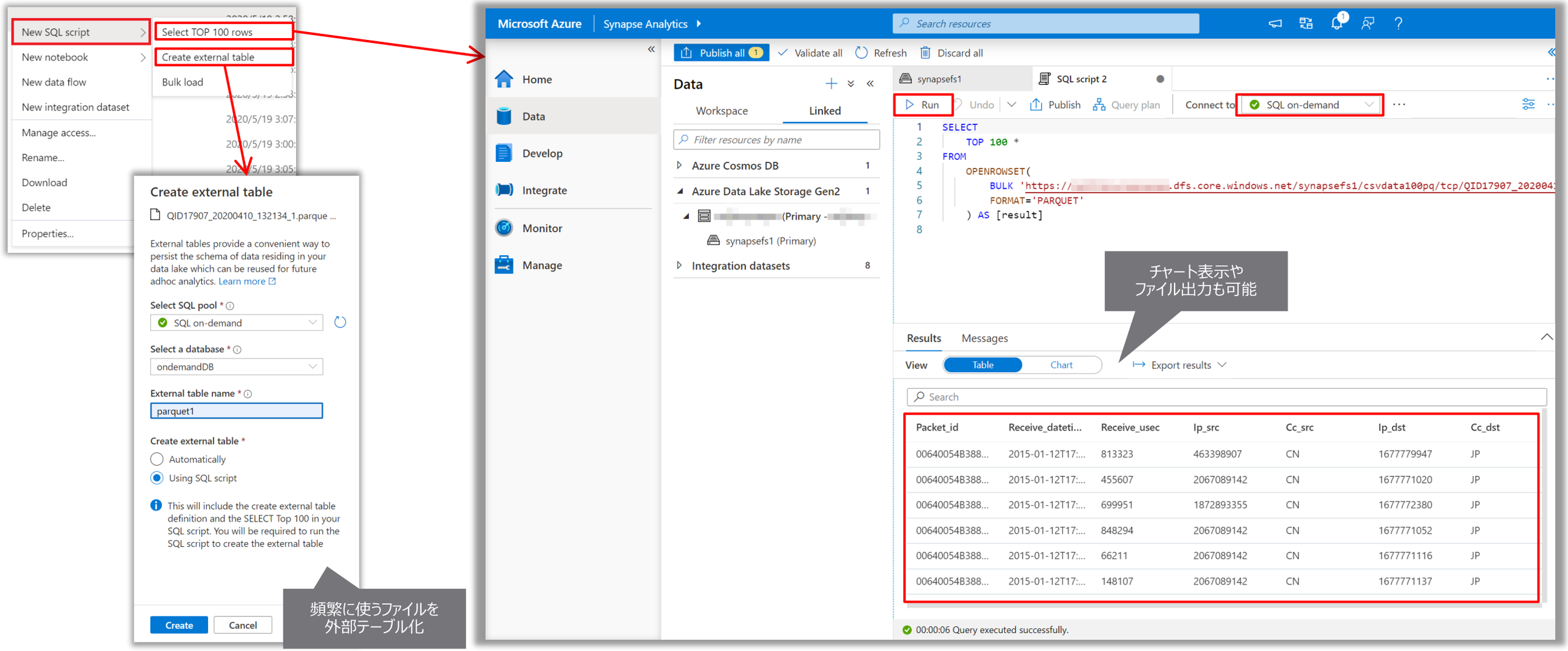
ADLS Gen2 ファイルシステム上のファイルを選択 ⇒ 右クリックで、すぐにアクション可能



ADLS Gen2 上のファイルを可視化・分析

ファイルデータを可視化・分析する為の作業を徹底的に自動化・省力化

Synapse Studio - Data Hub



ADLS Gen2 上のファイルをデータ加工＆ファイル出力

CREATE EXTERNAL TABLE AS SELECT (CETAS) を使用することで、Spark 以外にも、ビッグデータ処理 (加工処理) を行う選択肢が増えた

Synapse Studio - Data Hub

Microsoft Azure | Synapse Analytics

Search resources

Publish all | Validate all | Refresh | Discard all

Home
Data
Develop
Integrate
Monitor
Manage

Data

Workspace | Linked

Filter resources by name

Azure Cosmos DB 1
Azure Data Lake Storage Gen2 1
(Primary -)
synapsefs1 (Primary)
Integration datasets 8

synapsefs1 | 025_DataProcessing...

Properties (Ctrl+` / F4)

Run | Undo | Publish | Query plan | Connect to | SQL on-demand

```
----------------------------------------------------
-- CTAS を利用してデータ加工＆ファイル出力
----------------------------------------------------
IF OBJECT_ID(N'parquet2_exp') IS NOT NULL
    DROP EXTERNAL TABLE parquet2_exp
GO

CREATE EXTERNAL TABLE parquet2_exp
    WITH (
    LOCATION = 'csvdata100pq/export/parquet2_exp/',
    DATA_SOURCE = [synapsefs1_ _dfs_core_windows_net],
    FILE_FORMAT = [SynapseParquetFormat]
    )
AS SELECT TOP 10000 * FROM parquet1
GO
```

外部テーブル化で自動生成した
クエリを CETAS に書き換え

Results | Messages

17:37:17 Started executing query at Line 1

Total execution time: 00:00:01.090

00:00:01 Query executed successfully.

ADLS Gen2 ファイルシステム上に出力されていることを確認

synapsefs1 | 025_DataProcessing...

New SQL script | New notebook | New data flow | New integration dataset | Upload | More

synapsefs1 > csvdata100pq > export > parquet2_exp

Name	Last Modified	Content Type	Size
0FB54EA9-2286-44DB-A746-90ACF3BF34B2_41_0-1.parquet	2020/10/16 17:25:57		689.9 KB
_	2020/10/16 17:25:55		

クエリ分析性能：Spark, SQL pool との比較

18 億件超、18 カラムの Parquet ファイル (28 ファイルで、合計 66.2 GB) に対する集計クエリの実行

synapsefs1 › csvdata100pq › tcp

Name	Last Modified	Content Type	Size
QID17907_20200410_132134_0.parquet	2020/5/19 3:04:49		2.2 GB
QID17907_20200410_132134_1.parquet	2020/5/19 2:58:53		2.2 GB
QID17907_20200410_132134_10.parquet	2020/5/19 2:58:35		2.2 GB
QID17907_20200410_132134_13.parquet	2020/5/19 3:06:26		2.2 GB
QID17907_20200410_132134_16.parquet	2020/5/19 2:58:21		2.2 GB
QID17907_20200410_132134_17.parquet	2020/5/19 3:07:04		2.2 GB
QID17907_20200410_132134_18.parquet	2020/5/19 3:00:23		3.3 GB
QID17907_20200410_132134_2.parquet	2020/5/19 3:05:54		2.2 GB
QID17907_20200410_132134_20.parquet	2020/5/19 3:05:54		2.2 GB
QID17907_20200410_132134_21.parquet	2020/5/19 3:06:26		2.2 GB
QID17907_20200410_132134_25.parquet	2020/5/19 2:58:41		2.2 GB
QID17907_20200410_132134_28.parquet	2020/5/19 2:56:49		2.2 GB
QID17907_20200410_132134_30.parquet	2020/5/19 3:06:21		2.2 GB
QID17907_20200410_132134_33.parquet	2020/5/19 3:05:46		2.2 GB
QID17907_20200410_132134_34.parquet	2020/5/19 3:04:13		2.2 GB
QID17907_20200410_132134_35.parquet	2020/5/19 2:58:34		2.2 GB
QID17907_20200410_132134_37.parquet	2020/5/19 2:59:57		2.2 GB
QID17907_20200410_132134_39.parquet	2020/5/19 2:58:38		3.4 GB
QID17907_20200410_132134_4.parquet	2020/5/19 3:00:14		3.3 GB
QID17907_20200410_132134_41.parquet	2020/5/19 2:56:49		2.2 GB
QID17907_20200410_132134_45.parquet	2020/5/19 2:58:53		2.2 GB
QID17907_20200410_132134_46.parquet	2020/5/19 3:06:30		2.2 GB
QID17907_20200410_132134_48.parquet	2020/5/19 3:00:42		3.4 GB

Showing 1 to 28 of 28 cached items

SQL on-demand

```
-- Performance Comparison
SELECT COUNT(*) FROM (
SELECT ROUND((DATEDIFF(SECOND,'1970-01-01',receive_datetime)+32400)/300,0,1)*300-32400 AS d,
        port_src AS pts,
        cc_src AS ccs,
        COUNT(DISTINCT ip_src) AS ips
FROM View_t100_packet_tcp
WHERE receive_datetime >= '2014-08-01' AND receive_datetime < '2014-08-06'
GROUP BY
        ROUND((DATEDIFF(SECOND,'1970-01-01',receive_datetime)+32400)/300,0,1)*300-32400,
        port_src,
        cc_src
) t1
GO
```

00:00:16 Query executed successfully.

Spark

[2]

```
%%sql
-- Performance Comparison
SELECT COUNT(*) row_count FROM (
    SELECT ROUND((unix_timestamp(receive_datetime)+32400)/300,0)*300-32400 AS d,
        port_src AS pts,
        cc_src AS ccs,
        COUNT(DISTINCT ip_src) AS ips
    FROM sparkDB.T100_packet_tcp
    WHERE receive_datetime >= '2014-08-01' AND receive_datetime < '2014-08-06'
    GROUP BY
        d,
        pts,
        ccs
)
```

Command executed in 2mins 34s 541ms by seijim on 10-26-2020 15:11:02.158 +09:00

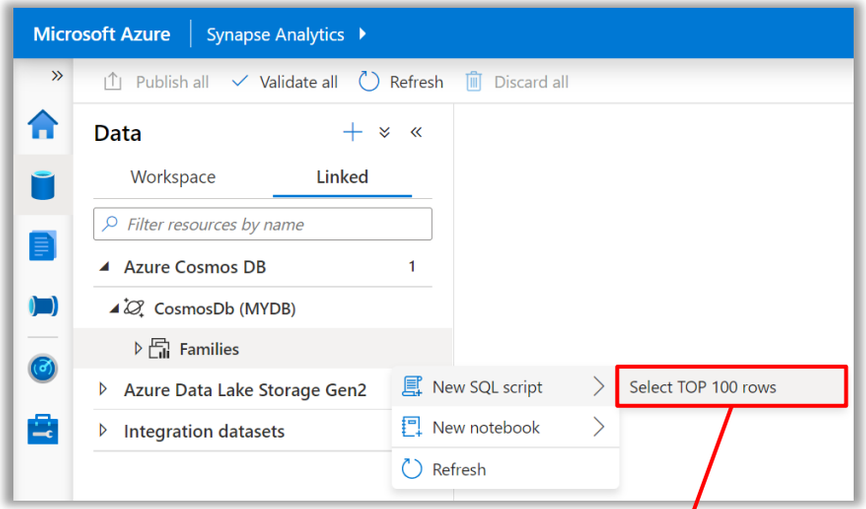
- Synapse Spark (4core × 8node) : 4 分 14 秒
- Synapse Spark (8core × 8node) : 2 分 34 秒
- Synapse SQL on-demand : 16 秒 (最遅 51 秒)
- Synapse SQL pool (DW500c) ※ : 10 秒

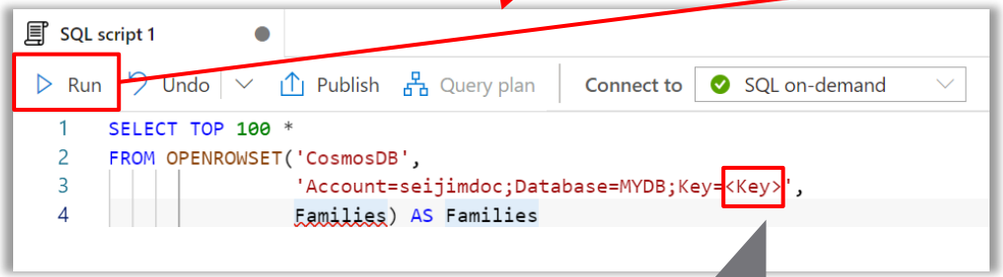
※Synapse SQL pool は、クラスター化列ストアインデックスを利用 (物理サイズ=54 GB)

Synapse Link : Cosmos DB Analytical Store へのアクセス

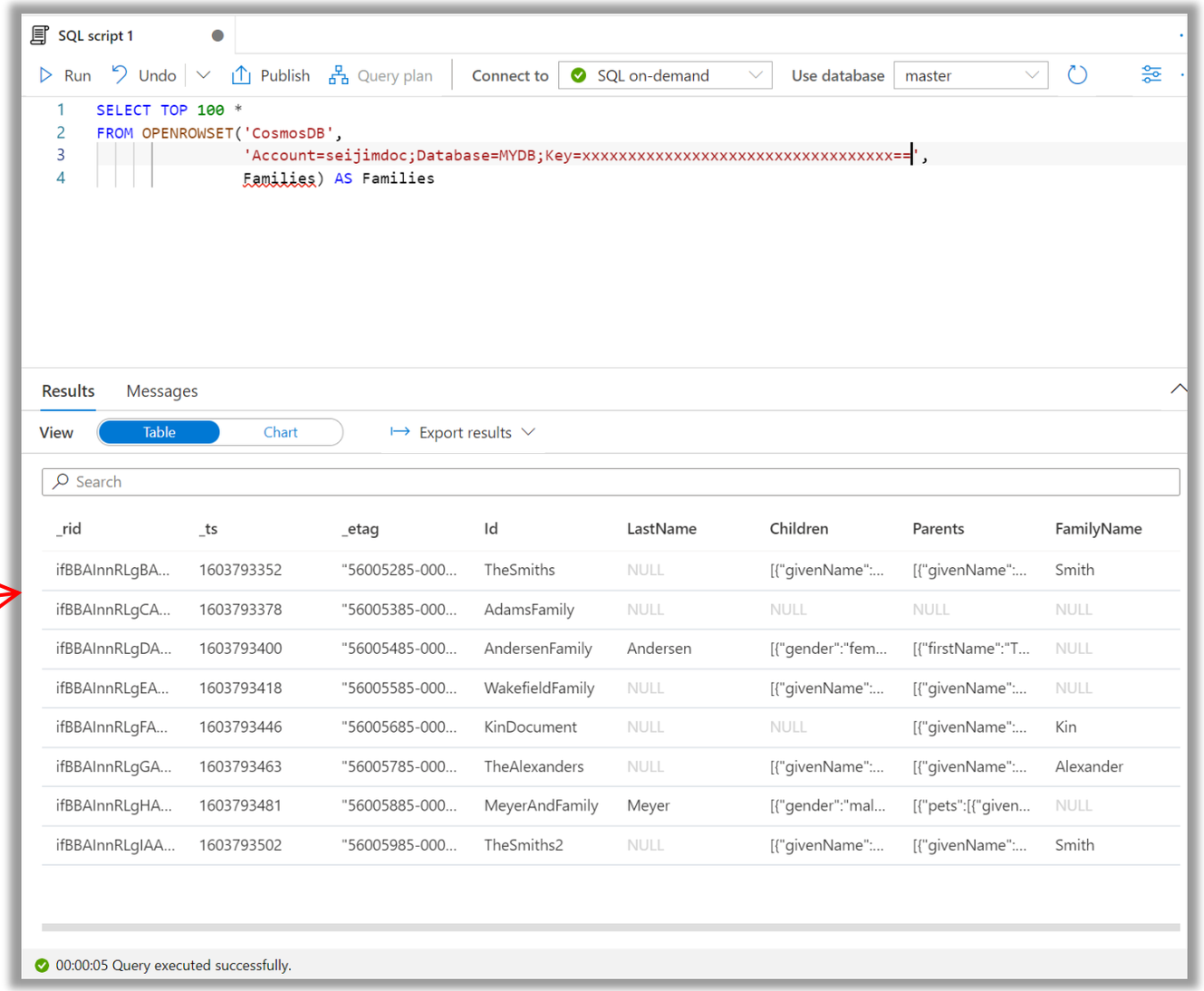
Synapse Link を使った SQL クエリの自動生成

Synapse Studio - Data Hub





アクセスキーを置き換え
Run をクリック



_rid	_ts	_etag	Id	LastName	Children	Parents	FamilyName
ifBBAInnRLgBA...	1603793352	"56005285-000...	TheSmiths	NULL	[{"givenName":...	[{"givenName":...	Smith
ifBBAInnRLgCA...	1603793378	"56005385-000...	AdamsFamily	NULL	NULL	NULL	NULL
ifBBAInnRLgDA...	1603793400	"56005485-000...	AndersenFamily	Andersen	[{"gender":"fem...	[{"firstName":"T...	NULL
ifBBAInnRLgEA...	1603793418	"56005585-000...	WakefieldFamily	NULL	[{"givenName":...	[{"givenName":...	NULL
ifBBAInnRLgFA...	1603793446	"56005685-000...	KinDocument	NULL	NULL	[{"givenName":...	Kin
ifBBAInnRLgGA...	1603793463	"56005785-000...	TheAlexanders	NULL	[{"givenName":...	[{"givenName":...	Alexander
ifBBAInnRLgHA...	1603793481	"56005885-000...	MeyerAndFamily	Meyer	[{"gender":"mal...	[{"pets":[{"given...	NULL
ifBBAInnRLgIAA...	1603793502	"56005985-000...	TheSmiths2	NULL	[{"givenName":...	[{"givenName":...	Smith

Synapse SQL pool

Synapse SQL pool の特徴

特徴

- データ ウェアハウス機能
- スケールアウト型の超並列データ分析エンジン
- T-SQL のサポート (SQL Server との高い互換性)
- Spark pool とのメタデータ共有

利点

- プロビジョニング型 (コンピューティング課金)
- 先進的なワークロード マネジメント (異なるワークロードをリソース分離して最適化可能)
- ロードデータに対するガバナンスが有効 (堅牢なセキュリティ、様々なセキュリティ機能)

適用領域

- エンドユーザー向けのインタラクティブな分析プラットフォーム

Synapse SQL pool 新機能 - Microsoft Ignite 発表 (2019/11) の最新状況

GA 済みの機能 :

- **性能 :** Resultset Caching
- **性能 :** Materialized View
- **性能 :** Ordered ColumnStore
- **データタイプ :** JSON サポート
- **セキュリティ :** Dynamic Data Masking
- **セキュリティ :** Private Link のサポート
- **CI & CD :** SSDT (SQL Server Data Tools) サポート
- **SQL :** Read committed snapshot isolation

- **ワークロード管理 :** Workload Isolation
- **データ取込み :** COPY ステートメント
- **データ取込み :** Streaming Ingestion & Analytics

プレビュー機能 :

- **データ共有 :** Azure Data Share によるデータ共有
- **ビルトイン ML** : ネイティブな予測/スコアリング
- **SQL :** 更新可能分散カラム
- **SQL :** JOIN を伴う FROM 句
- **SQL :** 複数カラムによる分散

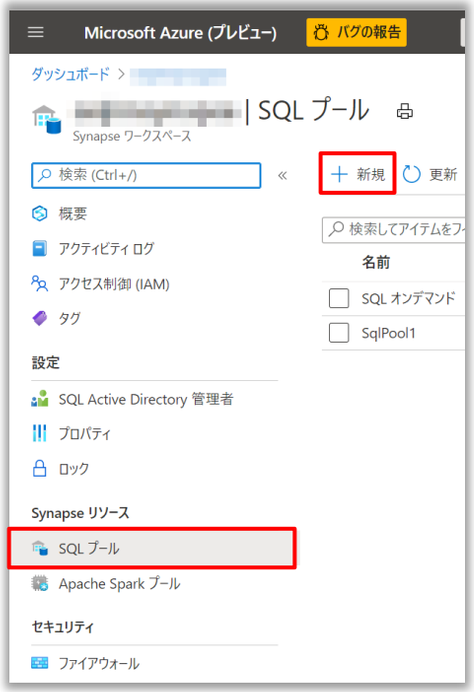
- **Fast Parquet** : Parquet ファイルへのダイレクトクエリ

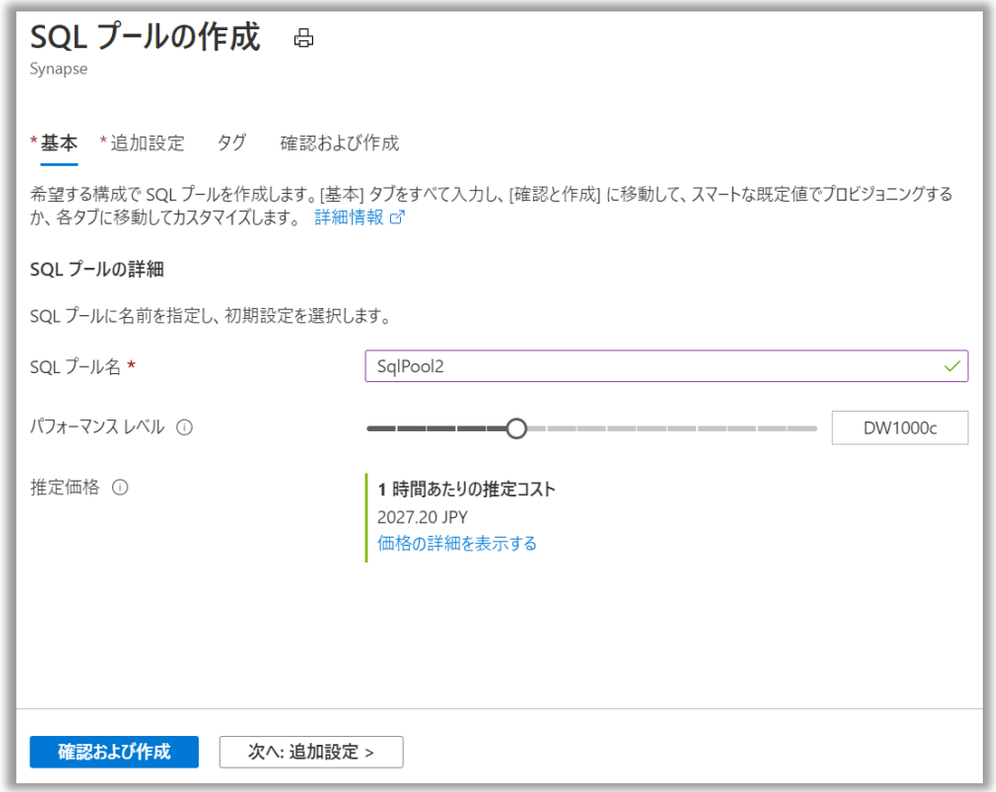
Next Gen (Gen3) ：革新的なプラットフォームを開発中

Synapse SQL pool の作成

Azure Portal、もしくは、Synapse Studio から SQL pool を作成

Azure Portal

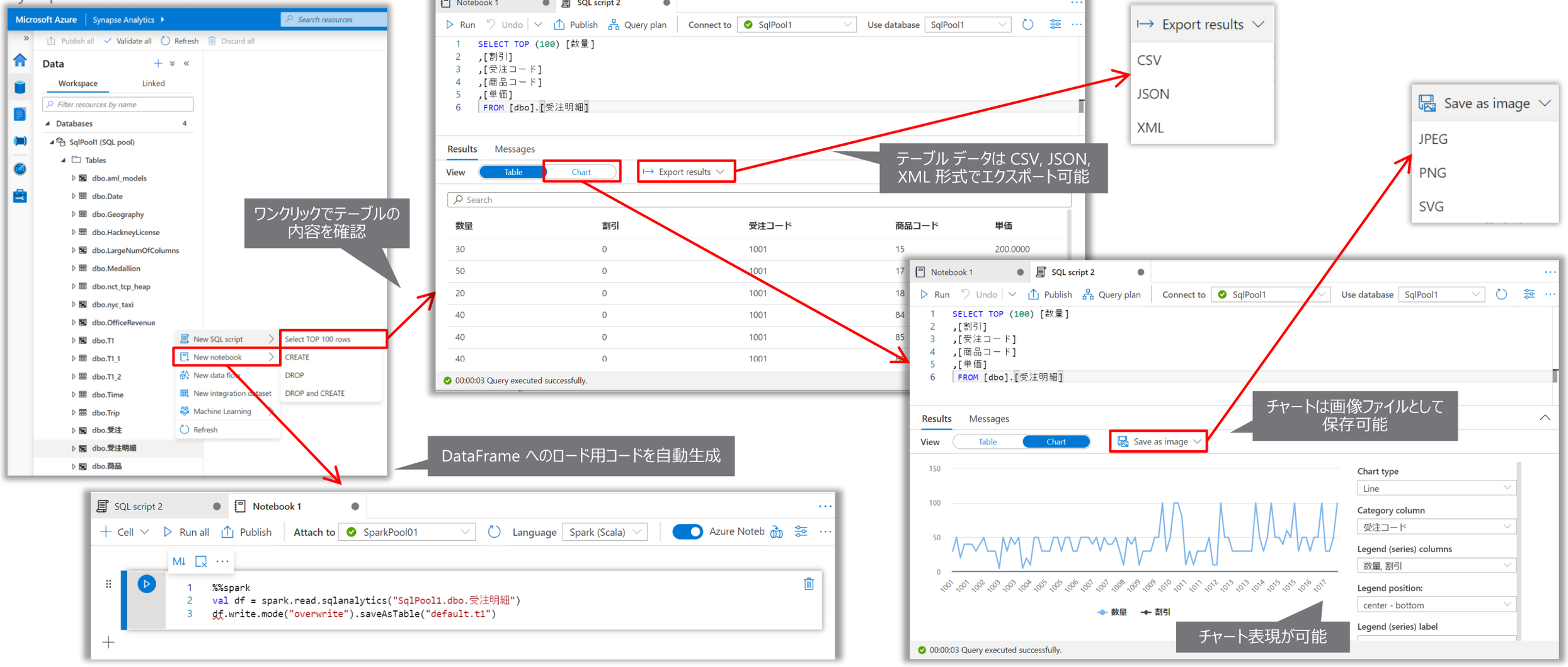




構造化データの分析

Synapse Studio では、SQL pool テーブルからのクエリ生成、Notebook コード (Spark) の生成、Data Flow の生成など、データ処理アプリケーションをすぐに作成することができる

Synapse Studio - Data Hub

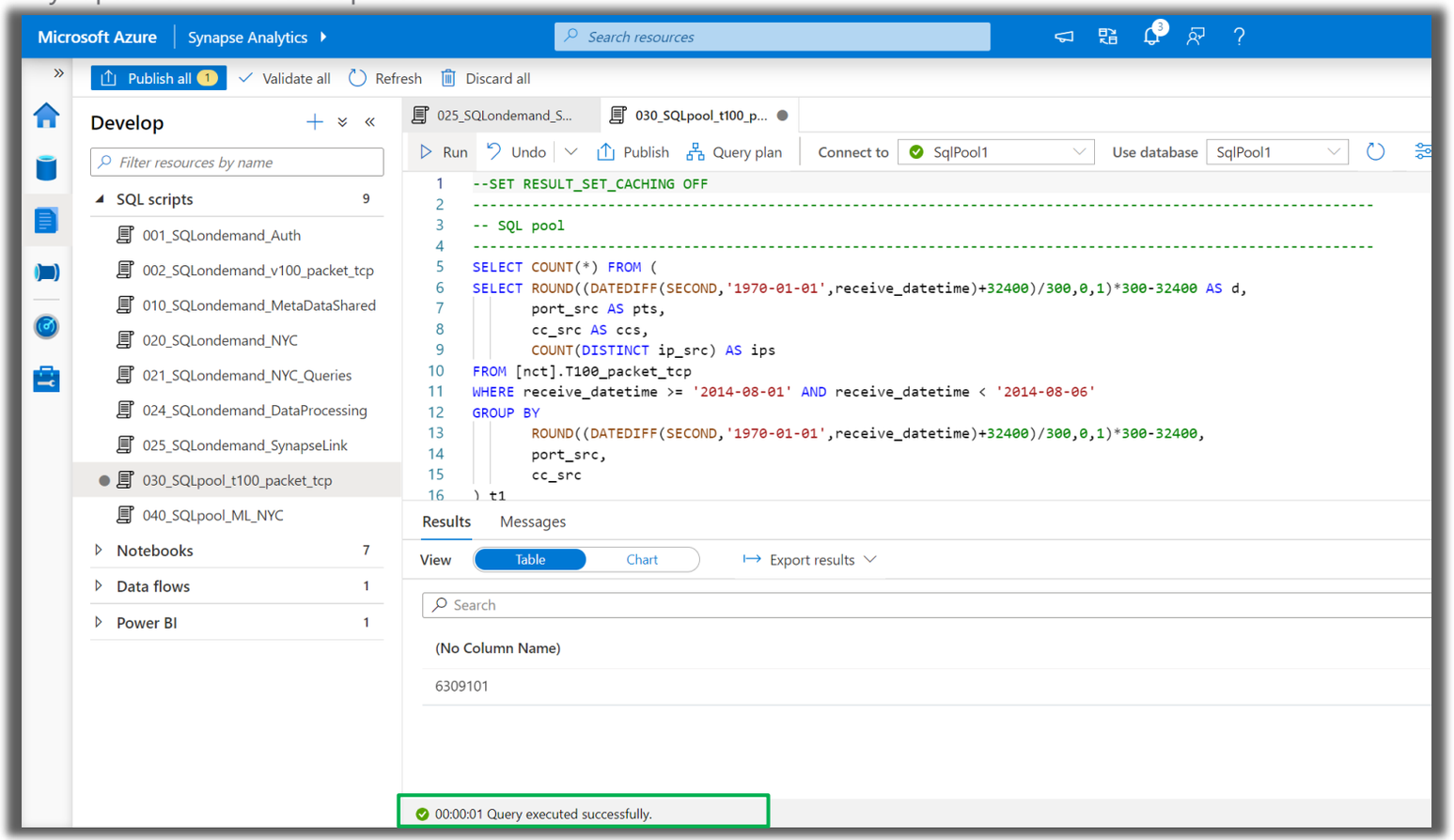


クエリ分析性能：結果セットのキャッシュ (Result-set caching)

Result-set caching を有効化すると、結果セットを Premium Storage にキャッシュし、Control ノードから直接結果を返却するようになる為、性能の向上と Compute ノードの負荷軽減を実現できる

18 億件超、18 カラムのクラスター化列ストア テーブル (54 GB) に対する集計クエリ

Synapse Studio - Develop Hub



- Synapse SQL pool (DW500c) : 10 秒
- Result-set Caching = ON : 1 秒

クエリ分析性能：ワークロードの分離

ワークロード分離は、リソースクラスとは異なり、コンピューティング リソース (CPU/メモリ) を分離する為、クエリが複数同時実行している環境下でのパフォーマンスを向上させることができる

同時多重の負荷継続下で、18 億件超、18 カラムのクラスター化列ストア テーブル (54 GB) に対する集計クエリを実行

```
-- ワークロードグループ
CREATE WORKLOAD GROUP wg1
 WITH
 (
    MIN_PERCENTAGE_RESOURCE = 25
,   CAP_PERCENTAGE_RESOURCE = 100
,   REQUEST_MIN_RESOURCE_GRANT_PERCENT = 25
 );

-- ワークロード分類子
CREATE WORKLOAD CLASSIFIER wc_wg1 WITH
(
    WORKLOAD_GROUP = 'wg1'
,   MEMBERNAME = 'seijimw1'
,   WLM_LABEL      = 'wg1'
 --,WLM_CONTEXT  = 'wg1ctx’
);

-- テスト用（ワークロードグループ適用あり） -- seijimw1
SET RESULT_SET_CACHING OFF;

SELECT COUNT(*) FROM (
    SELECT ROUND((DATEDIFF(SECOND,'1970-01-01',receive_datetime)+32400)/300,0,1)*300-32400 AS d,
            port_src AS t,
            cc_src AS o,
            COUNT(DISTINCT ip_src) AS c
    FROM [nct].T100_packet_tcp
    WHERE receive_datetime >= '2014-08-01' AND receive_datetime < '2014-08-06'
    GROUP BY
            ROUND((DATEDIFF(SECOND,'1970-01-01',receive_datetime)+32400)/300,0,1)*300-32400,
            port_src,
            cc_src
) t OPTION(LABEL='wg1’);
```

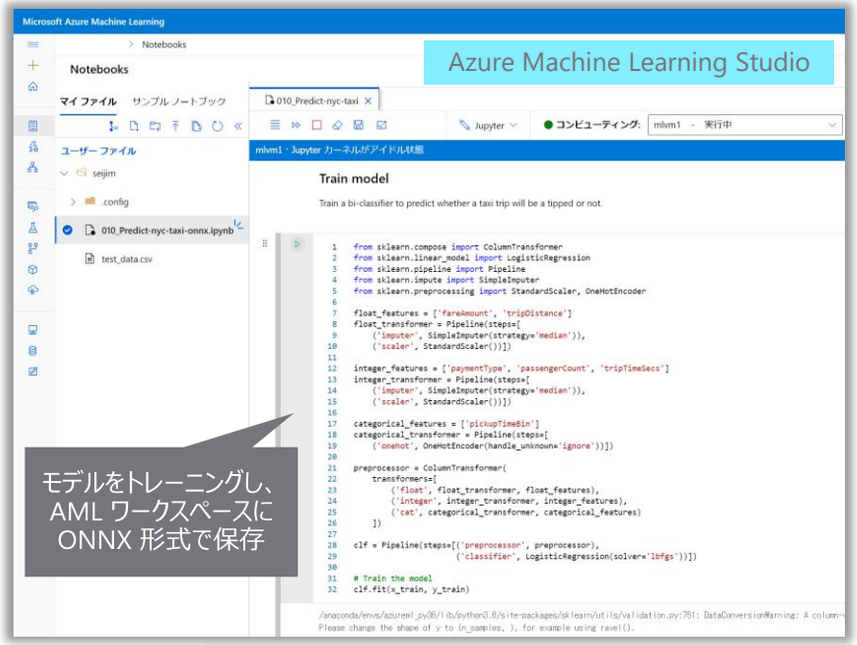
```
-- ワークロードグループが適用されているかをチェック
SELECT
    request_id,
    session_id,
    [status],
    total_elapsed_time,
    command,
    group_name,
    classifier_name
FROM sys.dm_pdw_exec_requests
WHERE [label] is not null
  and request_id in ('QID155230','QID155253’);
```

	request_id	session_id	status	total_elapsed_time	command	group_name	classifier_name
1	QID155230	SID19423	Completed	14640	SELECT COUNT(*) FROM (SELECT ROUND((DATEDIFF(...	wg1	wc_wg1
2	QID155253	SID19417	Completed	20203	SELECT COUNT(*) FROM (SELECT ROUND((DATEDIFF(...	smallrc	NULL

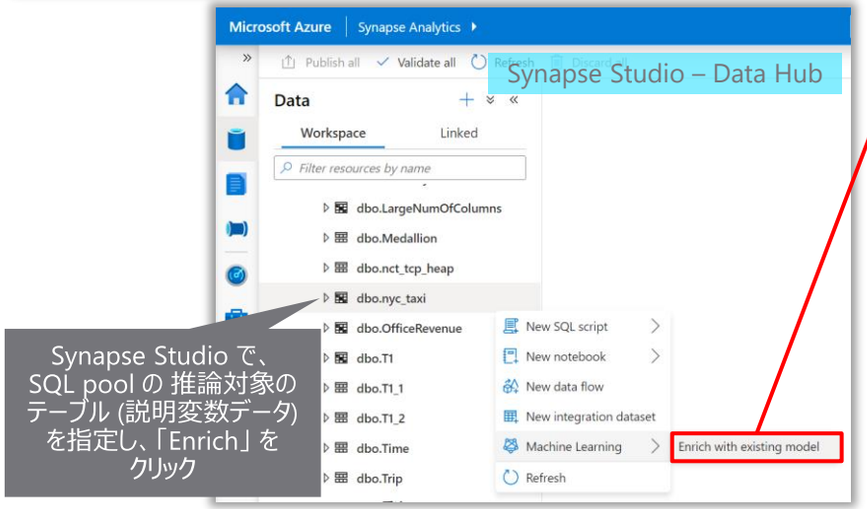
- ワークロード分離なし : 20 秒
- ワークロード分離あり : 15 秒

Azure Machine Learning (AML) との連携

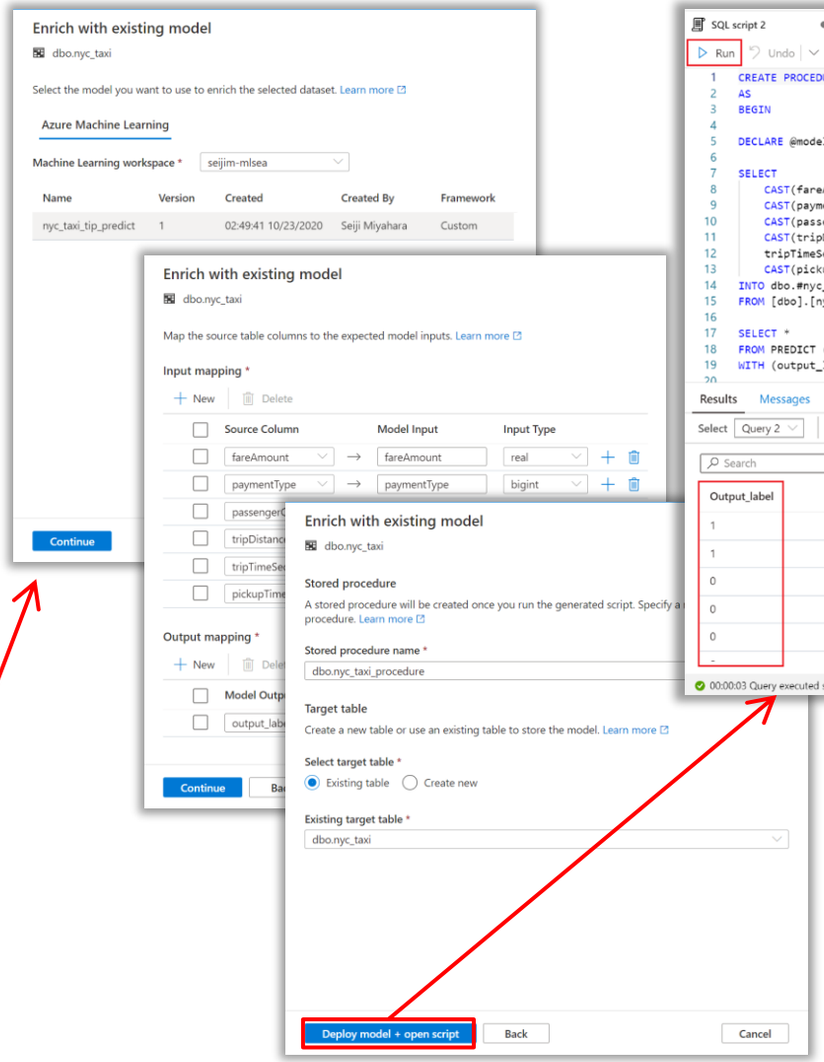
AML トレーニング済みモデルを SQL pool に格納し、クエリ実行時に推論 (予測) を行う

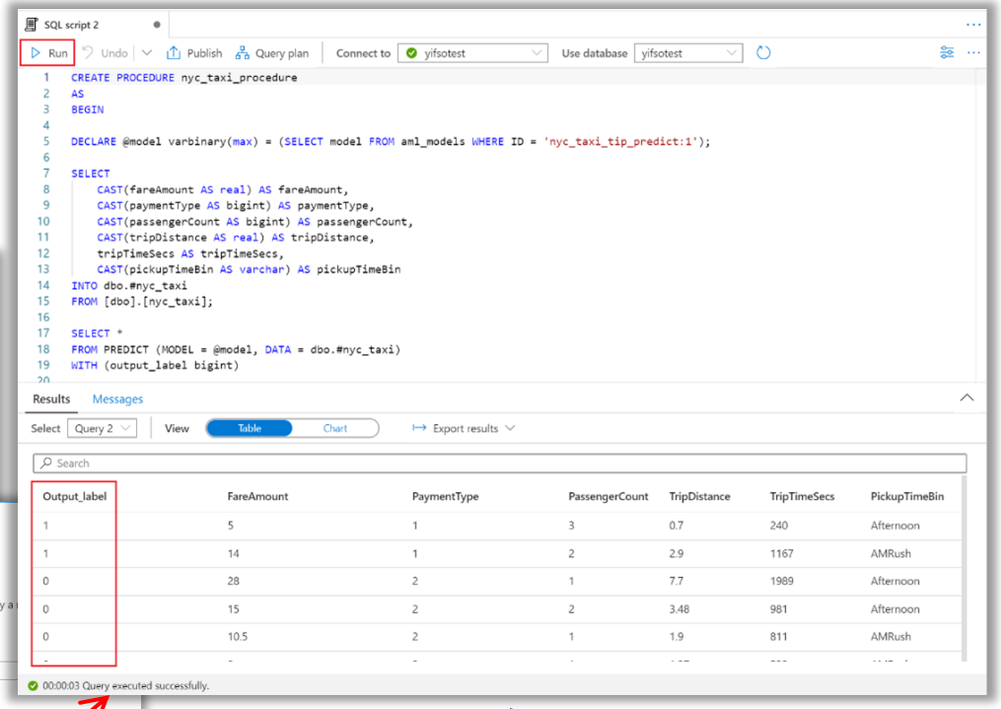


モデルをトレーニングし、AML ワークスペースに ONNX 形式で保存



Synapse Studio で、SQL pool の 推論対象のテーブル (説明変数データ) を指定し、「Enrich」をクリック





生成されたストアドプロシージャを利用して、推論 (予測) を実行

Synapse Pipelines & Data Flows

Synapse Pipelines & Data Flows の特徴

特徴

- クラウド オーケストレーター / クラウド ETL
- ビッグデータ処理機能

利点

- 90 種類のコネクター (Azure 各種サービス、他社クラウドサービス、オンプレ環境)
- GUI によるコードフリーのオーケストレーション / ETL (Pipelines)
- GUI によるコードフリーの Spark 処理 (Data Flows)
- イベント or スケジュール ベースのトリガー
- Auto-resolve IR (既定) がマネージド仮想ネットワークに組み込まれる為、セキュリティ担保目的で、Azure 内に Self-hosted IR を追加で構成する必要がない

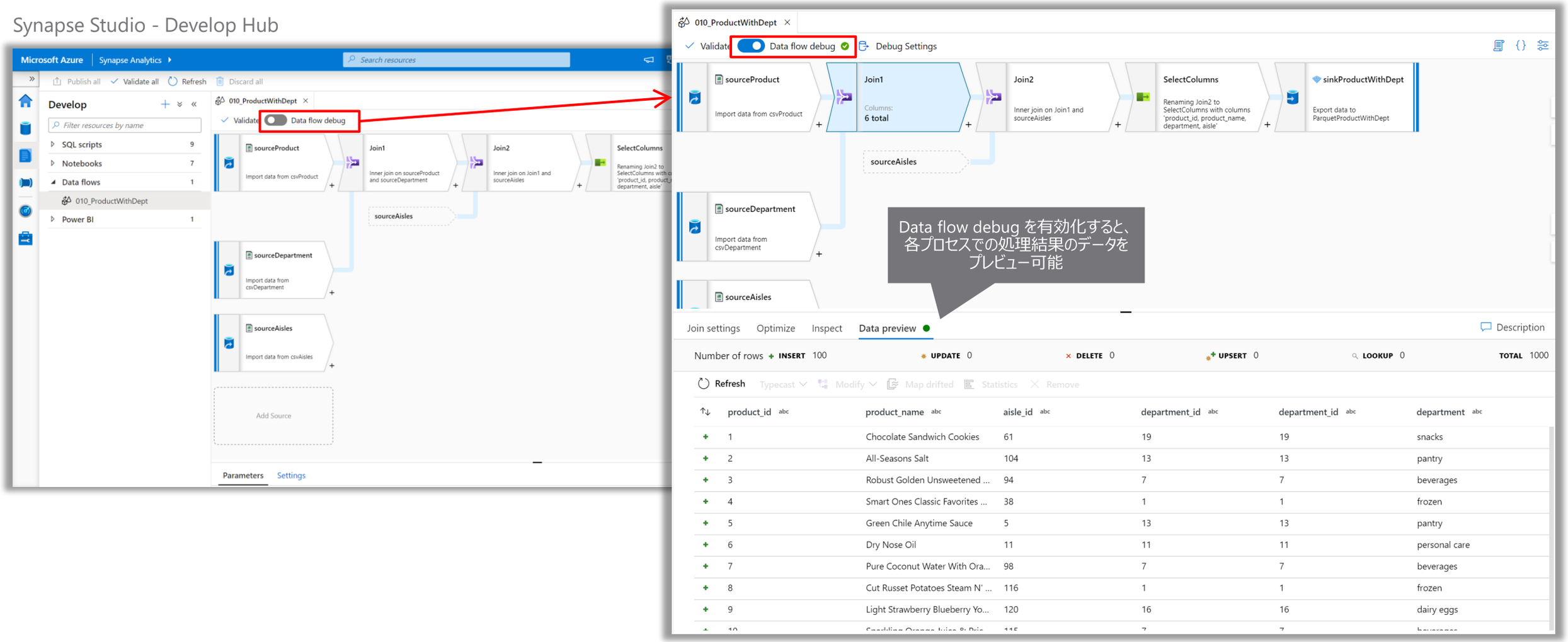
適用領域

- データの分析・加工、機械学習/AI モデル生成やバッチ推論の自動化

Synapse Data Flows

GUI によるコードフリーで、Spark ベースのビッグデータ処理ロジックを構築することができ、Synapse Pipelines の Data Flow アクティビティで実行制御する

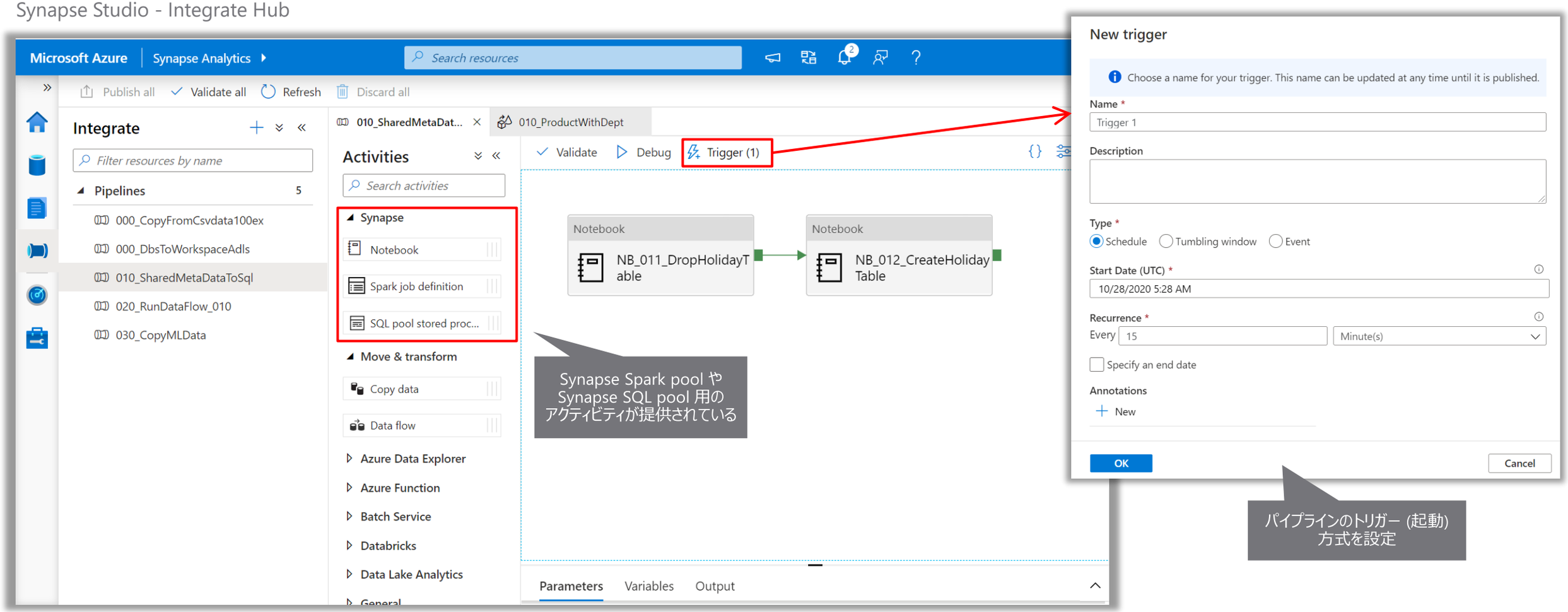
Synapse Studio - Develop Hub



Synapse Pipelines

GUI によるコードフリーで、クラウド ETL 処理や、様々な PaaS サービスを組み合わせたクラウド
オーケストレーションを実現でき、スケジュール/タンブリング ウィンドウ/イベント ベースの起動が可能

Synapse Studio - Integrate Hub



Power BI in Synapse Workspace

Power BI in Synapse の特徴

特徴

- Power BI Pro を Synapse Studio から利用

利点

- Synapse Studio でソースデータを見ながら、PBI レポートを編集できる
- Synapse Studio から PBI レポートを発行できる
- Synapse Studio 上での高度なビジュアル化を PBI が担う

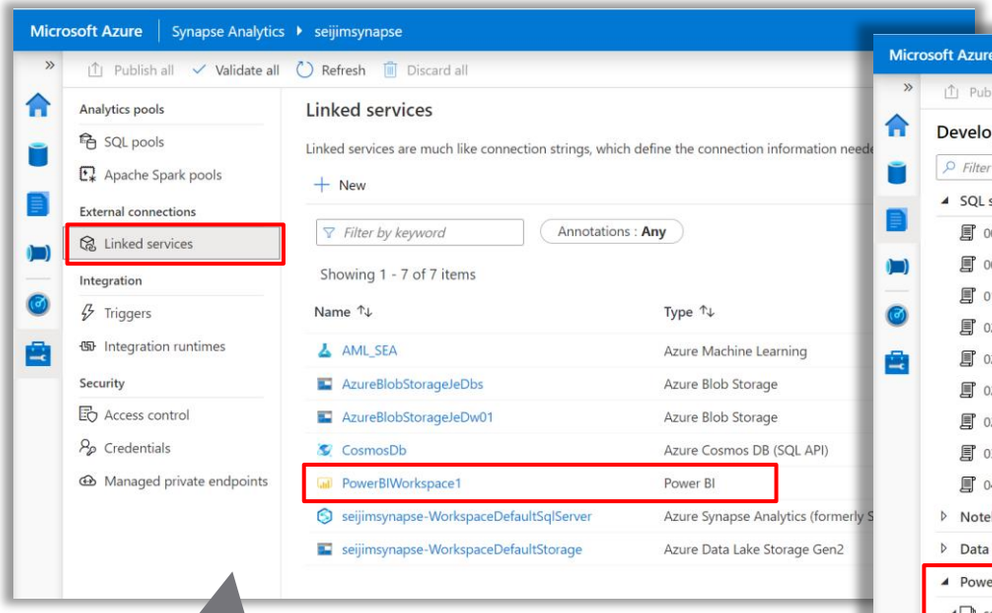
適用対象者

- データ サイエンティスト
- 分析レポート開発者

Power BI Report in Synapse

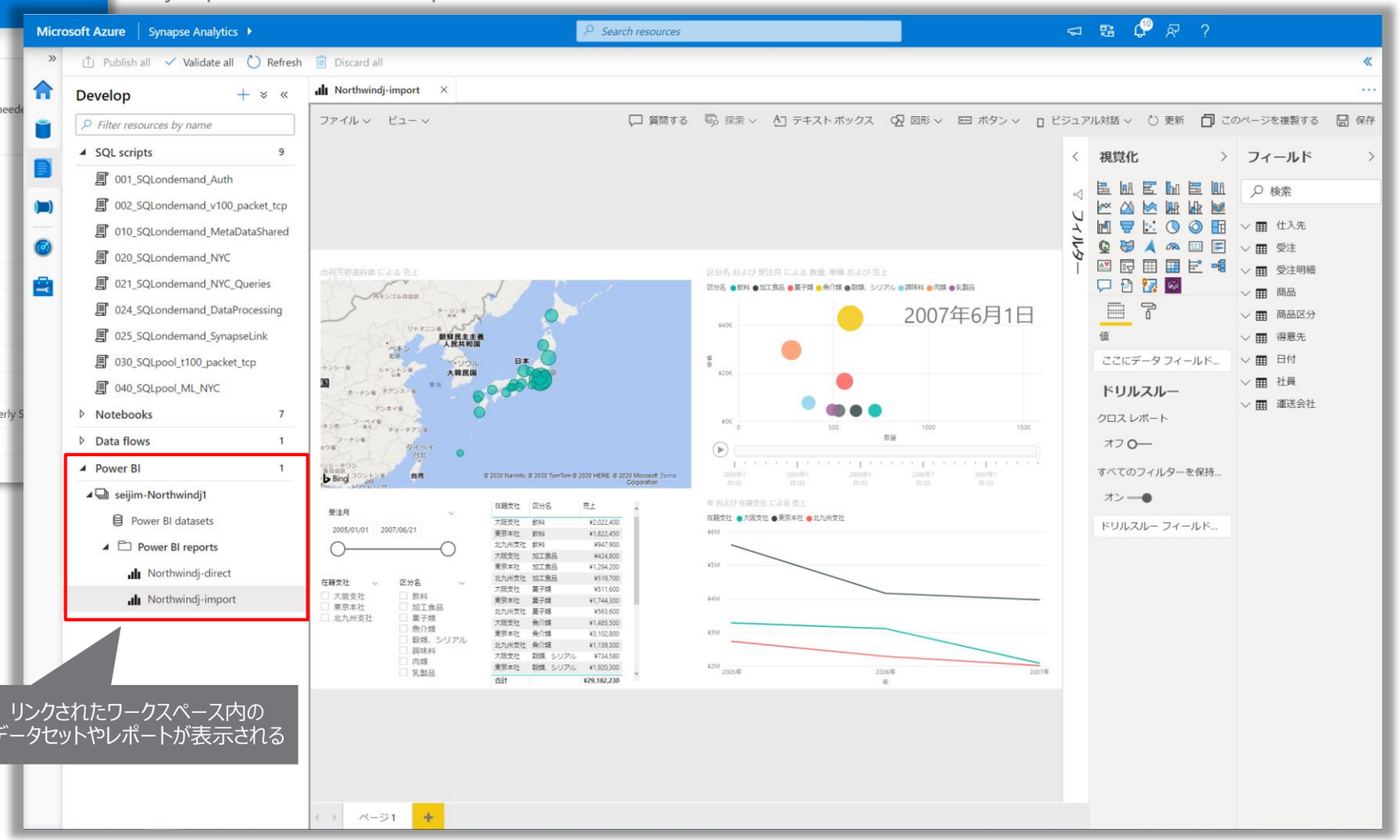
Power BI サービス上のワークスペースを Synapse Studio に統合し、レポートの編集・発行が可能

Synapse Studio - Manage Hub



事前に Power BI サービスの
ワークスペースとリンクしておく
(Linked サービスを作成)

Synapse Studio - Develop Hub



リンクされたワークスペース内の
データセットやレポートが表示される

Synapse Analytics のもたらす価値

Synapse Analytics のもたらす価値

- ペタバイトをはるかに超える巨大データを処理可能なスケーラビリティ
- データ収集から、非構造化/構造化データの加工・分析・可視化の容易性
- 複数の PaaS を組み合わせる複雑性を排除した、統合的なセキュリティと優れた運用性の実現

アンケートご回答のお願い

Azure Synapse Analytics (Workspace) 技術編

本セッションについて、日本マイクロソフト株式会社によるアンケートへご回答お願いいたします。
アンケート内にてご希望の方へは、後日セッション資料（PowerPoint形式）を電子メールにてお送りさせていただきます。

https://aka.ms/dbts38

※アンケートURL 有効期限 : 2020/12/11

触って理解するデータ活用と AI

ウェビナー　ハンズオン / ワークショップ　PoC / アセスメント

http://aka.ms/aiwda

ウェビナー/
ワークショップ情報

お客様事例

お役立ち
コンテンツ

日本マイクロソフトは、**クラウドを活用したデータ分析基盤**、**データを活用するための分析モデルの構築**、**AIの活用**をご検討のお客様向けに、課題に合わせた様々なテーマごとにウェビナーやワークショップなどをご用意しております。詳細はこちらのページをご確認ください。

Appendix
統合的なセキュリティ・運用

Azure Synapse セキュリティ全体像

- **ネットワーク セキュリティ**
  - ✓ ワークスペースの IP ファイアウォール規則
  - ✓ ワークスペースのマネージド仮想ネットワーク (Spark pool, Pipelines がデプロイされる)
  - ✓ ワークスペースのマネージド プライベート エンドポイント
- **認証とパススルー**
  - ✓ Azure AD 認証 (ワークスペースへの認証と各機能へのパススルー)
  - ✓ SQL 認証 (SQL pool と SQL on-demand のエンドポイントへ直接接続する時に利用可能)
- **アクセス制御**
  - ✓ ADLS Gen2 ストレージは、ファイルやフォルダーに Azure AD ユーザー/グループで権限設定
  - ✓ SQL pool と SQL on-demand は、DB ユーザー (Azure AD 認証/SQL 認証) 単位の権限設定
  - ✓ ワークスペースのマネージド ID (パイプラインの Linked サービスの認証で利用可能)
- **暗号化**
  - ✓ ADLS Gen2 のストレージ暗号化と SQL pool の透過的暗号化 (両方とも既定で有効)

Azure 構成要素

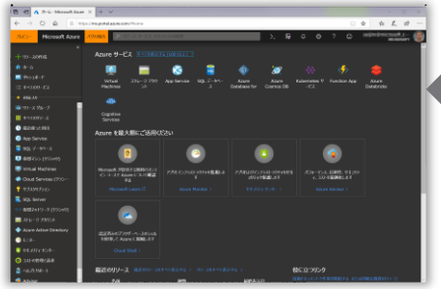
Azure Active Directory

xxxxx.onmicrosoft.com
(別名: **xxxxx.co.jp**)
ディレクトリ

Azure AD ディレクトリは、
任意のサブスクリプションで
作成可能

サブスクリプションは、
必ずどこかの Azure AD
ディレクトリへの関連付けが
必要となる
(CSP は、変更不可)

サブスクリプション
(Azure Portal)



Azure Portal は、Azure のすべての
サービスをデプロイ & 管理できる

Azure SaaS / API

DevOps や AI 系 (Cognitive Services) などの
SaaS 型 / API 型のサービス

DevOps Services
Cognitive Services
・・・・・ etc.

Azure PaaS (Multi-tenant)

Service Fabric Cluster (コンテナー & サービス凝集型) などをベースとした PaaS で、サービス毎に Firewall 機能、VM ベースのゲートウェイ機能を持ったマルチテナント型アーキテクチャ

Private Link
Blob Storage
Queue Storage
File Storage
Cosmos DB
Event Hubs
・・・ etc.

F/W 機能
SQL Data Warehouse
SQL Database
DB for PostgreSQL
DB for MySQL
Functions (従量課金)

Data Factory
Analysis Services
(G/W 型)
・・・ etc.

Azure IaaS

仮想マシン (VM, VM Scale-set), NIC, 管理ディスク, 仮想ネットワーク (VNet), Load Balancer が主たる構成要素

Load Balancer
VM (100 種類超の Spec)
サブネット / NIC に対して F/W 設定 (サービスタグ or IP)
NSG
VNet (1,000 超のサブネットを構成可能)

VNet Service Endpoint
or
Private Endpoint

Azure PaaS (Dedicated)

VM や VM Scale-set をベースとした PaaS で、VNET への組み込みが 可能な占有型アーキテクチャ

VNet に組み込み

Databricks
HDInsight
・・・・・ etc.

App Service Env / 統合
Service Fabric
AKS (K8s)
Batch (HPC)
SQL Database Managed Instance

Azure Resource Manager (ARM)

Azure リソースのデプロイ & 管理機能 (Azure Portal, Azure CLI / PowerShell, ARM REST API)

ARM テンプレートによるデプロイ

・JSON による一貫性のあるデプロイ
・繰り返して利用可能

リソース グループ

複数のリソースをまとめて管理

Role based Access Control (RBAC)

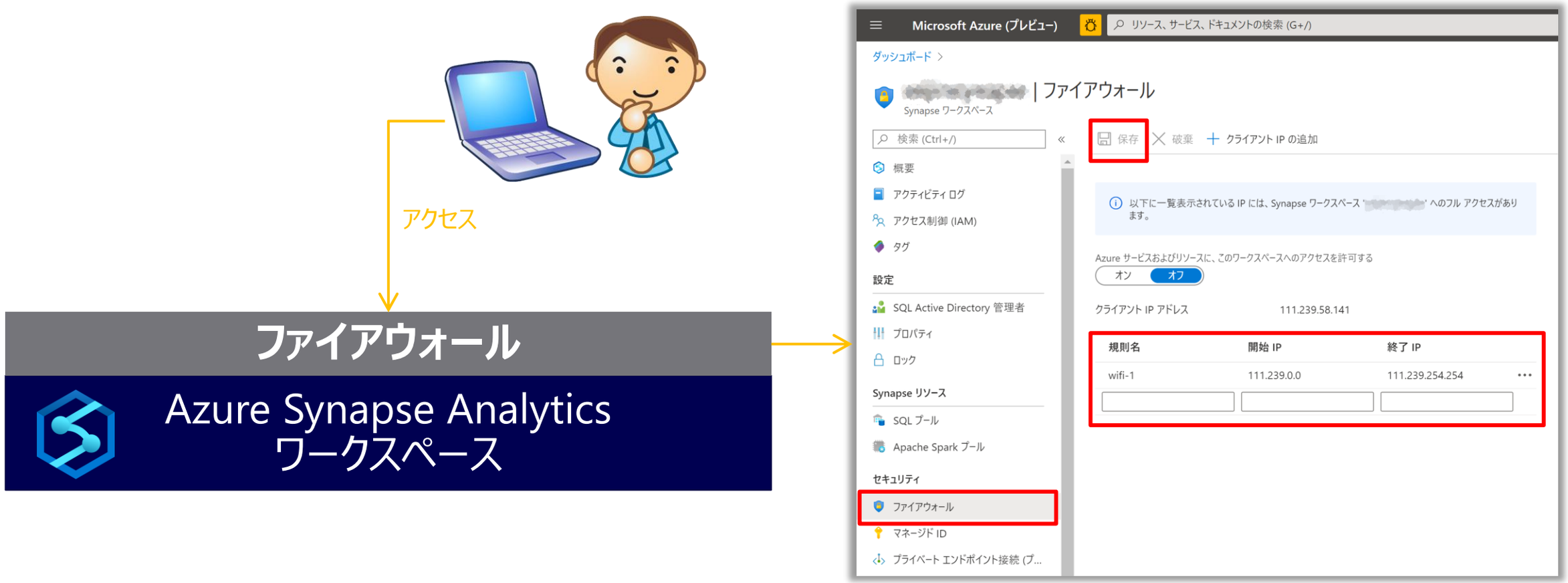
リソース グループやリソース単位でのアクセス制御

タグ や ロック

・リソースに任意のタグを追加
・リソースの変更または削除を禁止

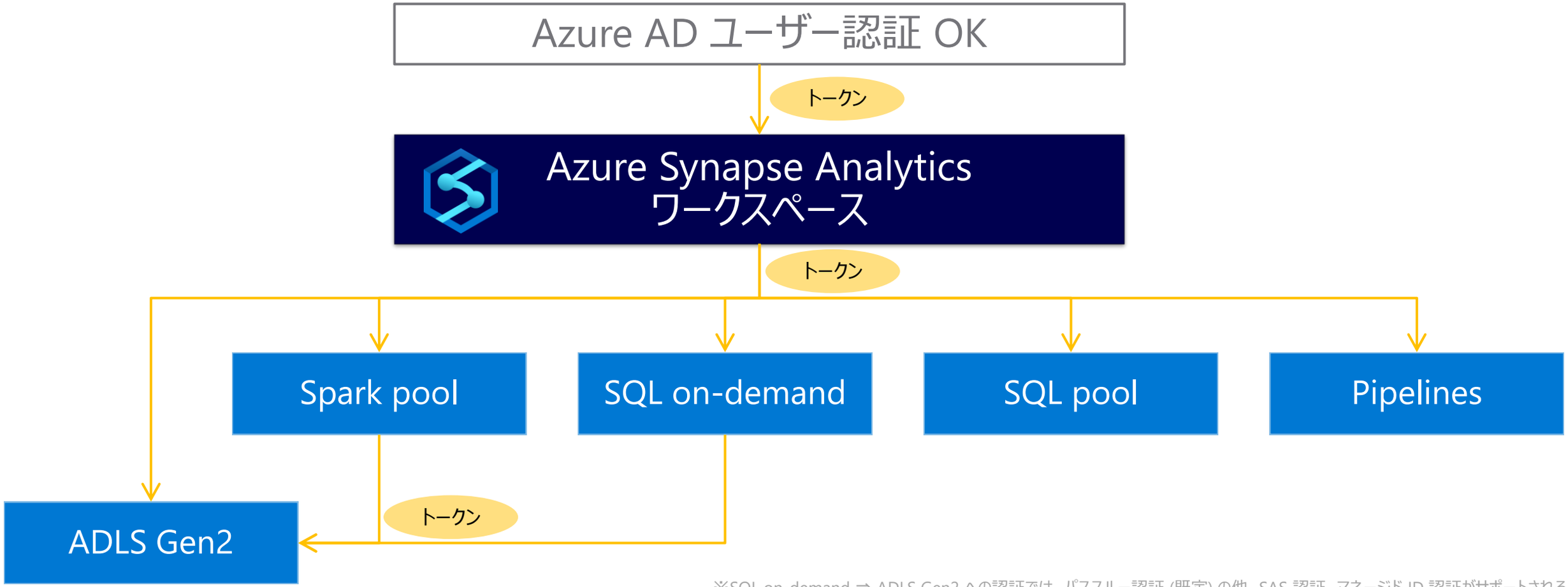
ワークスペースのファイアウォール規則

- Azure Portal の Synapse ワークスペースのメニューで設定
- ワークスペースへのログイン以前の為、Synapse Studio では設定できない
- Synapse Studio / SQL on-demand / SQL pool へのパブリック IP アクセスを制限



ワークスペースの Azure AD 認証とパススルー

ワークスペースへの Azure AD ユーザー認証のログインが成功すると、ユーザー トークンがそれぞれの機能に渡され、そのユーザーの資格情報が検証される

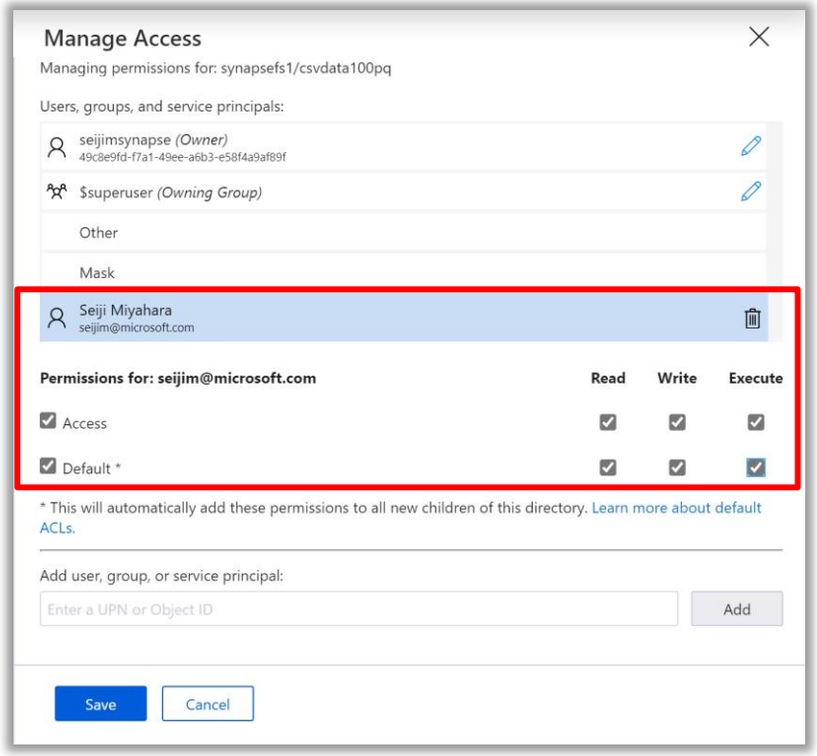


※SQL on-demand ⇒ ADLS Gen2 への認証では、パススルー認証 (既定) の他、SAS 認証、マネージド ID 認証がサポートされる

ADLS Gen2 へのアクセス権限追加

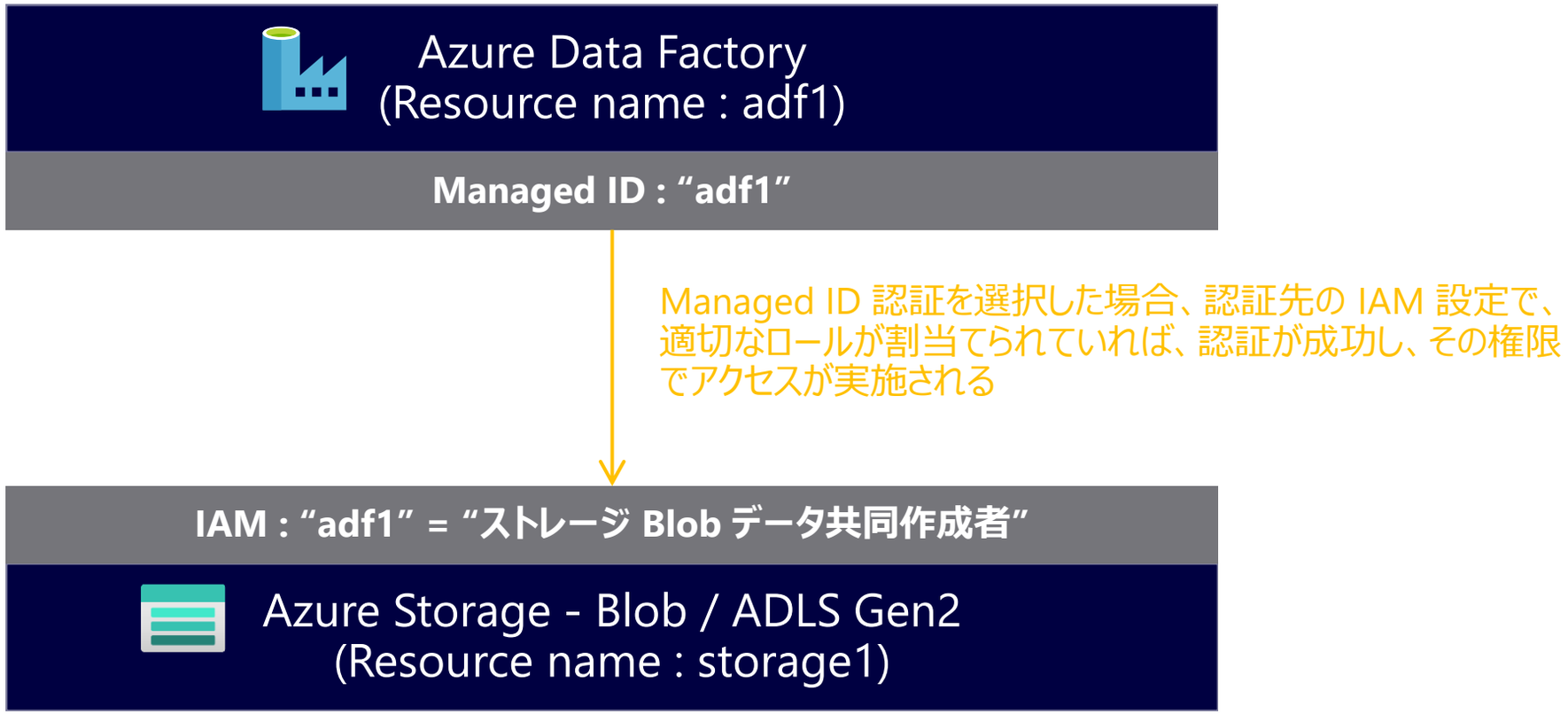
Spark pool や SQL on-demand から ADLS Gen2 に対して、パススルー認証が行われる為、事前にアクセス権限を付与する必要がある (※ワークスペース作成者も必要)

- Azure Portal で対象ストレージの [アクセス制御 (IAM)] で "ストレージ Blob データ共同作成者" ロールに自分の Azure AD ユーザー プリンシパル (user_name@xxxxx.co.jp など) や所属するセキュリティ グループを追加する
- Azure Portal もしくは Synapse Studio で、フォルダーやファイルに対して、自分の Azure AD ユーザー プリンシパル (user_name@xxxxx.co.jp など) や所属するセキュリティ グループを追加し、Read/Write/Execute の権限を設定する



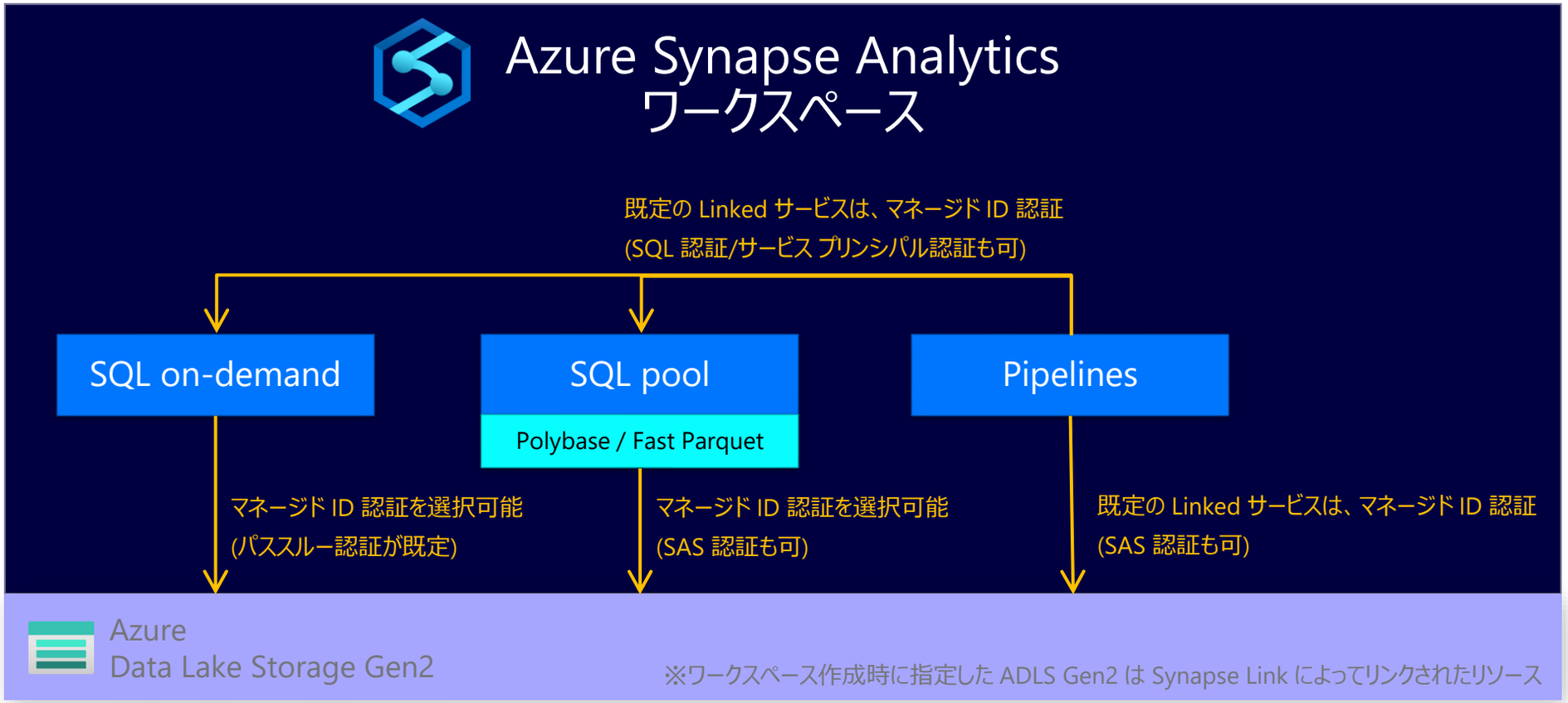
マネージド ID とは

- マネージド ID とは、Azure AD で自動的に管理される ID を Azure サービスに提供する機能
- マネージド ID は、Azure リソース間の認証/認可で利用
- マネージド ID を利用することで、ID やパスワードの管理から解放される



ワークスペースのマネージド ID

- ワークスペースのマネージド ID は、ワークスペース名と**同名**が割り当てられる
- ワークスペースのマネージド ID 認証は、ワークスペース内やリンクされたリソース、ワークスペース外のリソースへのアクセスで利用可能



ワークスペース外の Azure リソースに対しても、マネージド ID 認証が可能

SQL on-demand : マネージド ID 認証利用 (例1)

- 既定のパススルー認証をやめ、マネージド ID 認証に変更する
- 既定の ADLS Gen2 であれば、ワークスペース名のマネージド ID に対して "ストレージ Blob データ共同作成者" ロールが割り当て済み

```
-- パススルー認証を特定のユーザー「xxxxxxxxxxx」に対して無効化する
DENY REFERENCES ON CREDENTIAL::[UserIdentity] TO [xxxxxxxxxxx];
GO
-- パススルー認証をすべて無効化する
DROP CREDENTIAL [UserIdentity];
GO
-- 既定の ADLS Gen2 ファイルシステムにマネージド ID 認証を適用する
CREATE CREDENTIAL [https://adlsgen2xxxxx.dfs.core.windows.net/synapsefs1]
WITH IDENTITY='Managed Identity';
GO

-- パススルー認証を再度有効化する場合 (ワークスペース作成時に流されるスクリプト)
CREATE CREDENTIAL [UserIdentity] WITH IDENTITY = 'User Identity';
GO
GRANT REFERENCES ON CREDENTIAL::[UserIdentity] TO [public];
GO
```

SQL on-demand : マネージド ID 認証利用 (例2)

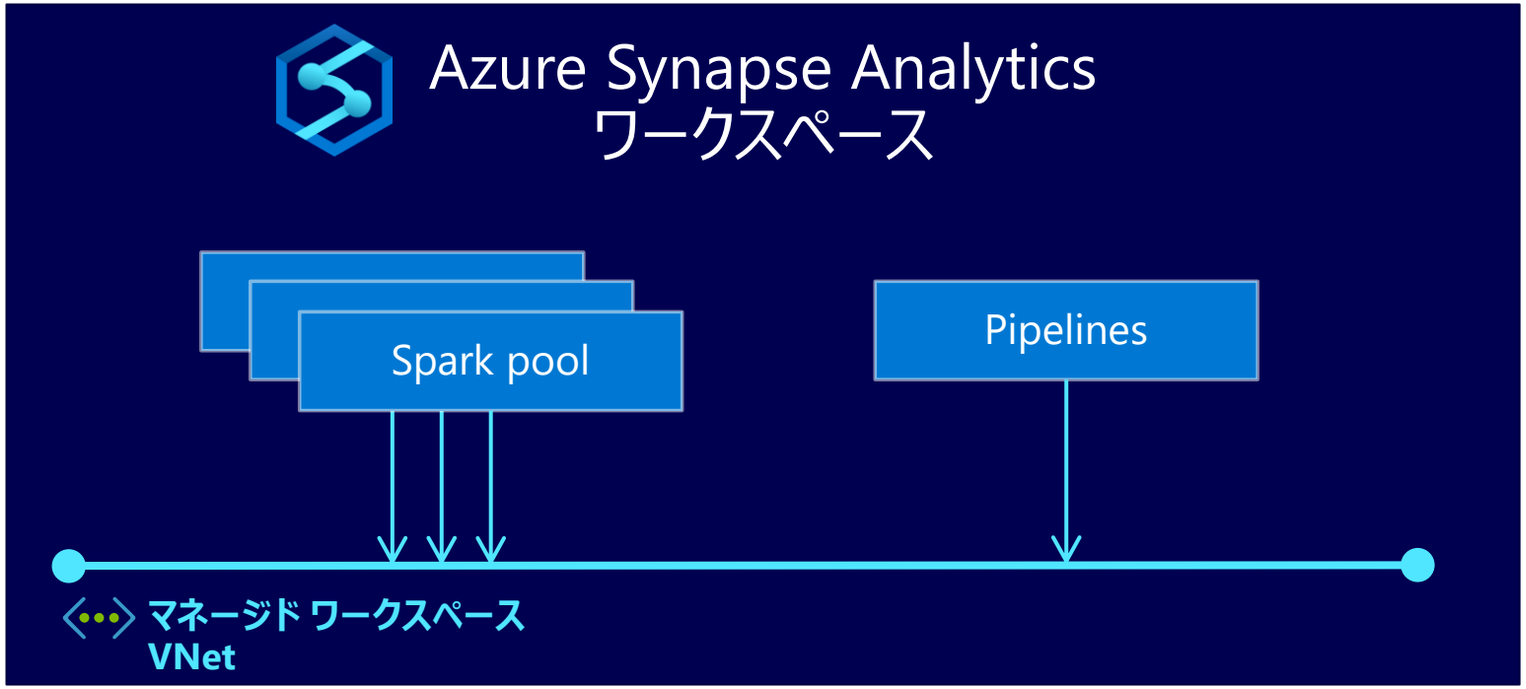
- 特定のバッチ処理 (ADLS Gen2 への出力) に適用する前提で、ワークスペースのマネージド ID を利用
- 既定の ADLS Gen2 であれば、ワークスペース名のマネージド ID に対して "ストレージ Blob データ共同作成者" ロールが割り当て済み

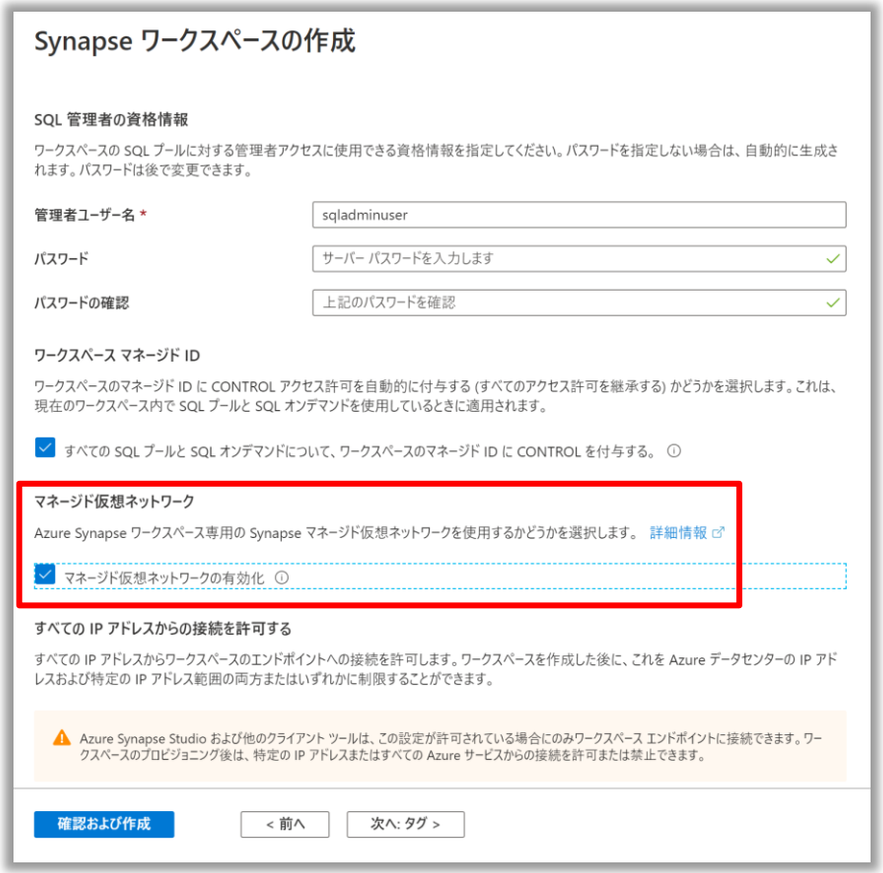
```
-- Create Database
CREATE DATABASE ondemandDB;
GO
-- Master key
USE ondemandDB;
CREATE MASTER KEY;
GO
-- Credential
CREATE DATABASE SCOPED CREDENTIAL MyAdls
WITH IDENTITY='Managed Identity'
GO
-- External Data Source
CREATE EXTERNAL DATA SOURCE MyAdlsSource
WITH (
    LOCATION = 'https://adlsgen2xxxxx.dfs.core.windows.net/synapsefs1',
    CREDENTIAL = MyAdls
);
GO
-- Format
CREATE EXTERNAL FILE FORMAT ParquetFormat
WITH (
    FORMAT_TYPE = PARQUET,
    DATA_COMPRESSION = 'org.apache.hadoop.io.compress.SnappyCodec'
);
GO
```

```
-- ADLS Gen2 ストレージに 2016 年の NYC データを出力
USE ondemandDB;
CREATE EXTERNAL TABLE OpenData_NYC
WITH (
        LOCATION='opendata/nyc/',
        DATA_SOURCE = MyAdlsSource,
        FILE_FORMAT = ParquetFormat
)
AS
SELECT * FROM
    OPENROWSET(
        BULK 'https://azureopendatastorage.blob.core.windows.net
/nyctlc/yellow/puYear=2016/puMonth=*/*.parquet',
        FORMAT='PARQUET'
    ) AS [nyc]
GO
```

ワークスペースのマネージド仮想ネットワーク

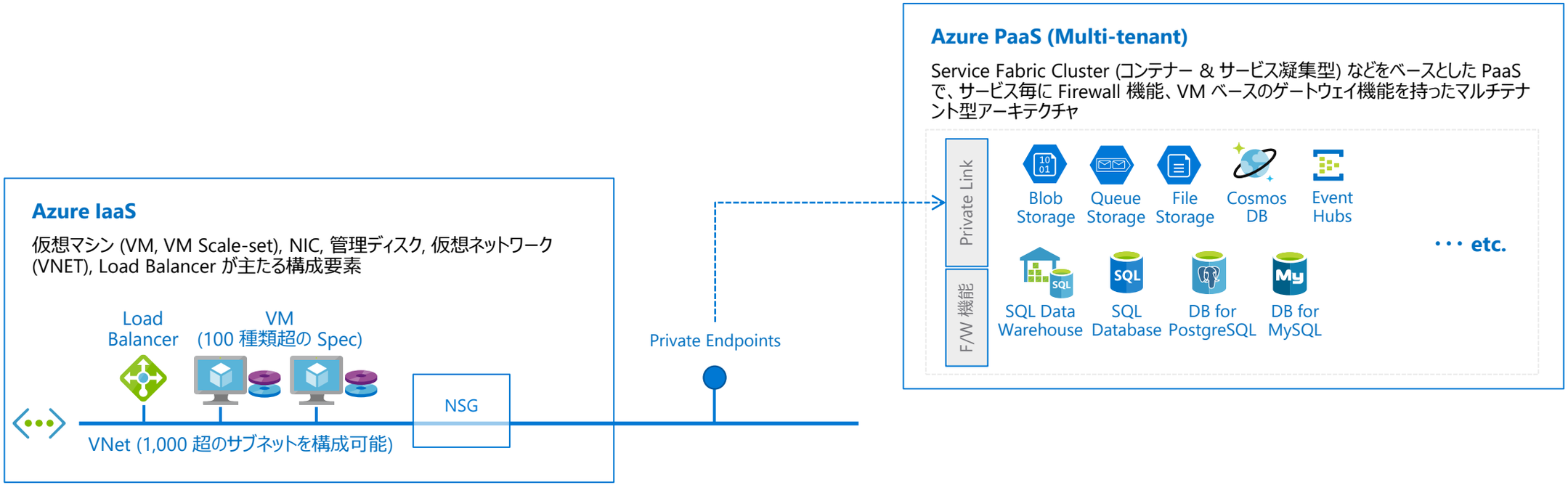
- マネージド仮想ネットワークは自動管理される VNet で、ユーザーは触ることができない
- マネージド仮想ネットワークが有効な場合、Spark pool と Pipelines が自動的に配置される
- SQL on-demand / SQL pool は、マルチテナント型 PaaS の為、仮想ネットワークへの配置はされない
- マネージド仮想ネットワークは、ワークスペース作成時のみに構成可能





プライベート エンドポイント / プライベート リンクとは

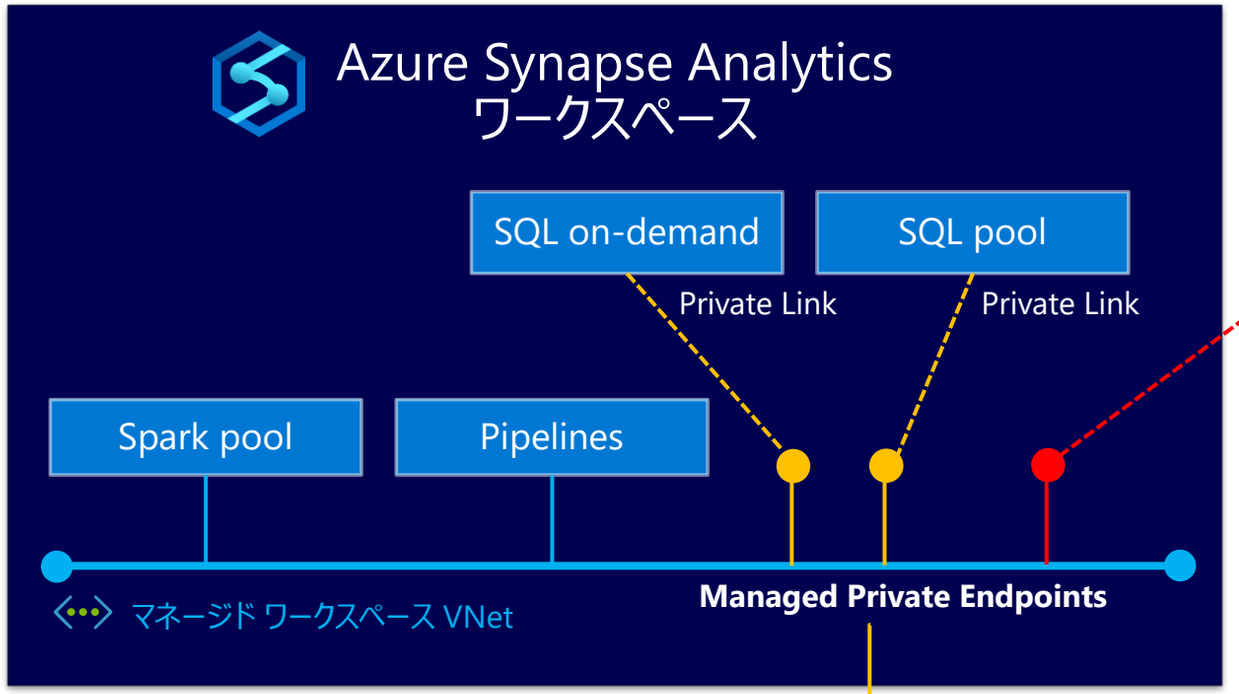
- プライベート エンドポイントは、マルチテナント型 PaaS リソースを VNet に仮想的に組み込み、プライベート IP で通信する機能
- 既定では、プライベート DNS ゾーンも展開される為、従来の FQDN をそのまま利用可能
- プライベートリンクは、PaaS 側で提供するプライベート エンドポイントとリンクする機能
- 1 つの PaaS リソース (プライベートリンク) は、複数のプライベート エンドポイントにリンク可能



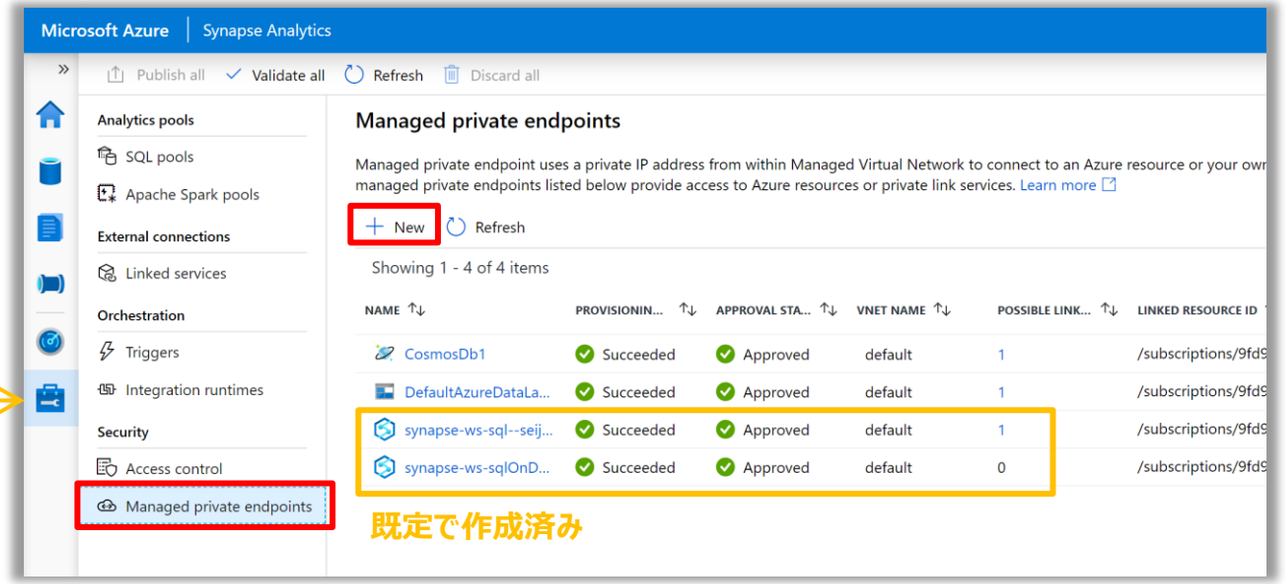
ワークスペースのマネージド プライベート エンドポイント

Spark pool / データ統合機能から以下のリソースに対するプライベート接続可能

- ワークスペース内の SQL on-demand / SQL pool へのプライベート接続 (既定で作成済み)
- ワークスペース外の Azure リソースへのプライベート接続 (追加で作成要)



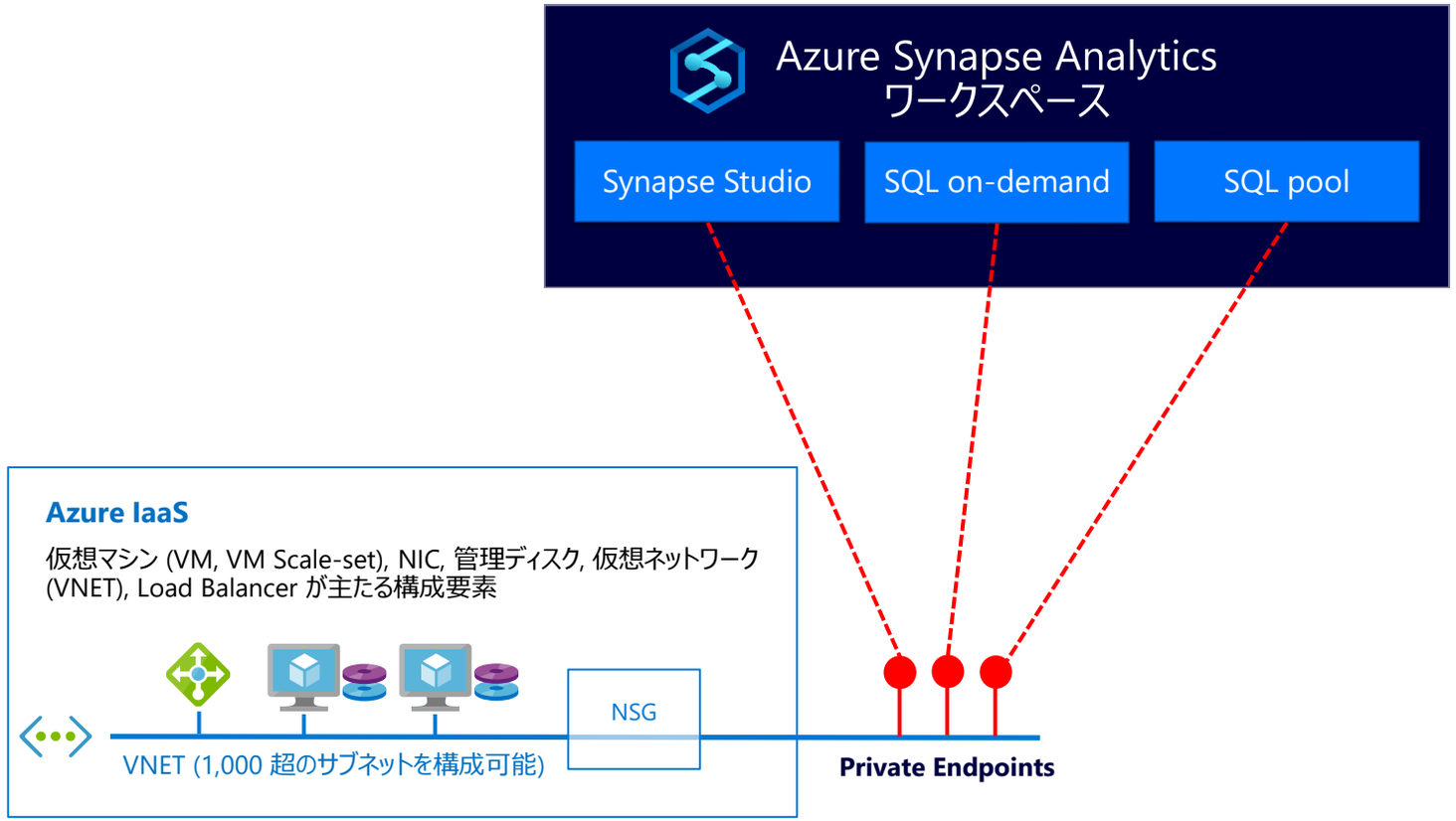
Synapse Studio

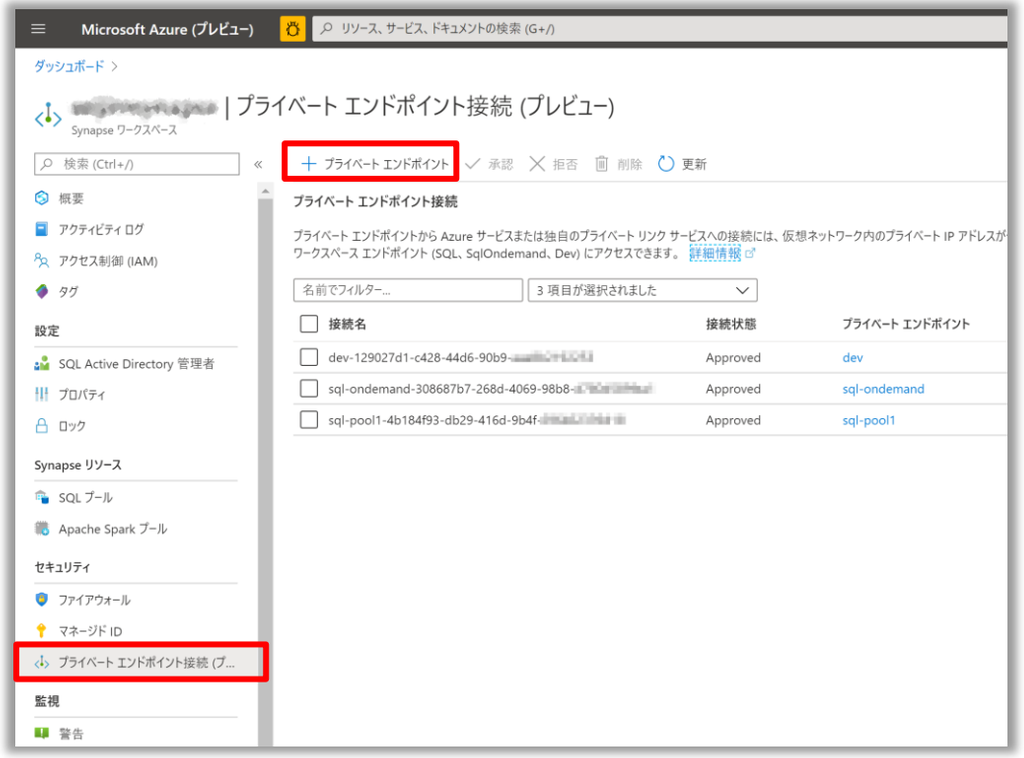


ワークスペース リソースへのプライベート エンドポイント接続

他のサービスからワークスペース内の以下のリソースにプライベート接続可能

- Synapse Studio
- SQL on-demand
- SQL pool

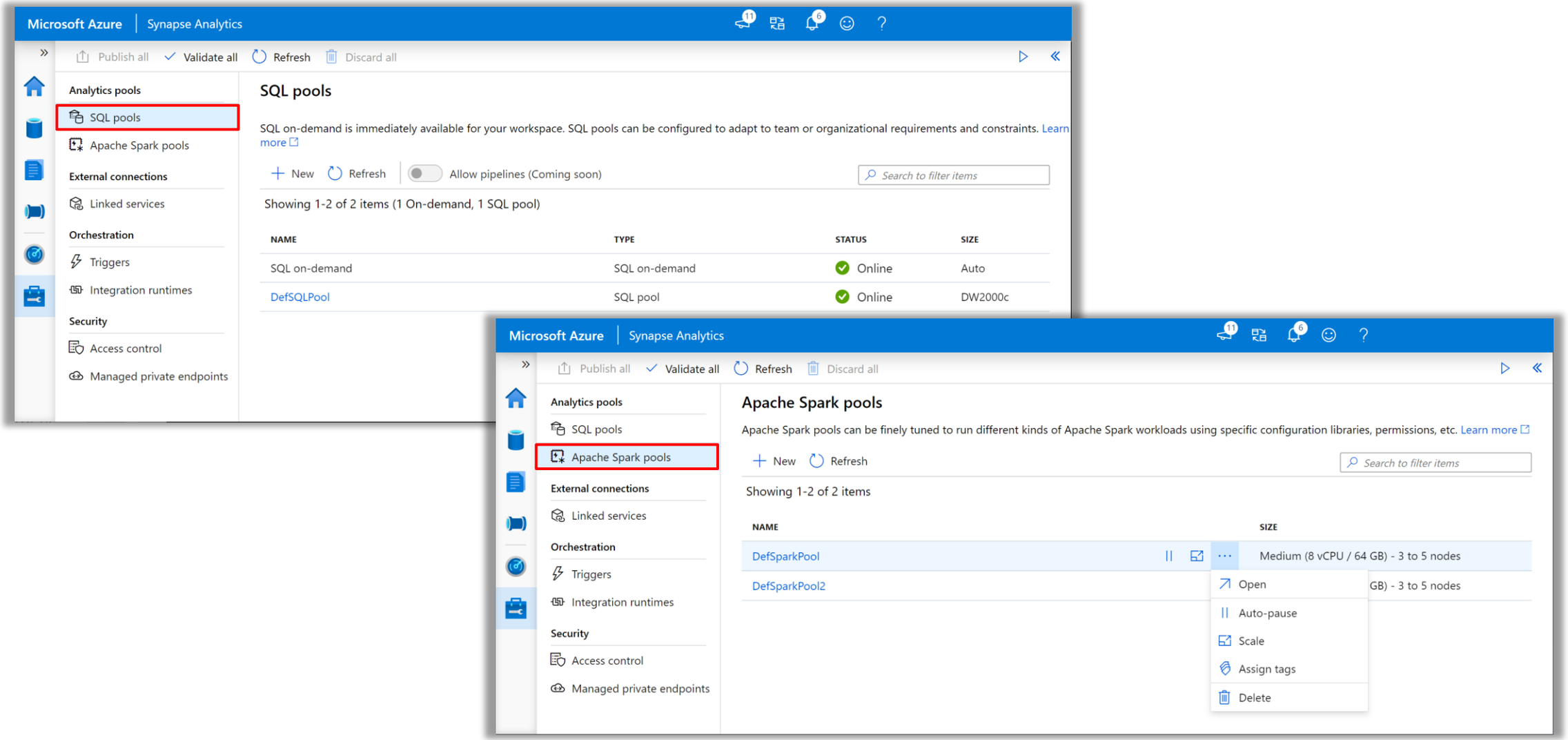




Synapse Studio
Manage Hub

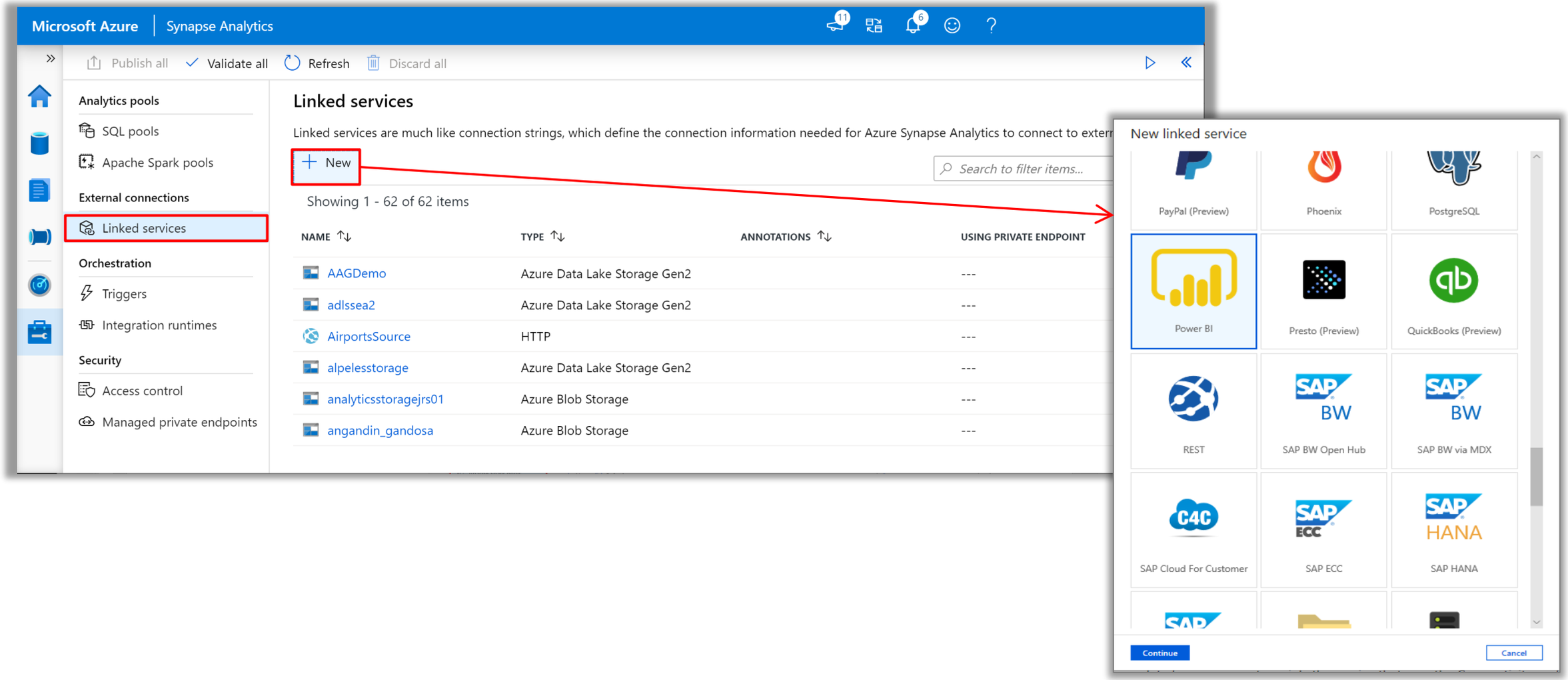
Manage Hub - Analytics プール

SQL pool や Spark pool の管理機能 (作成・削除・設定変更など)



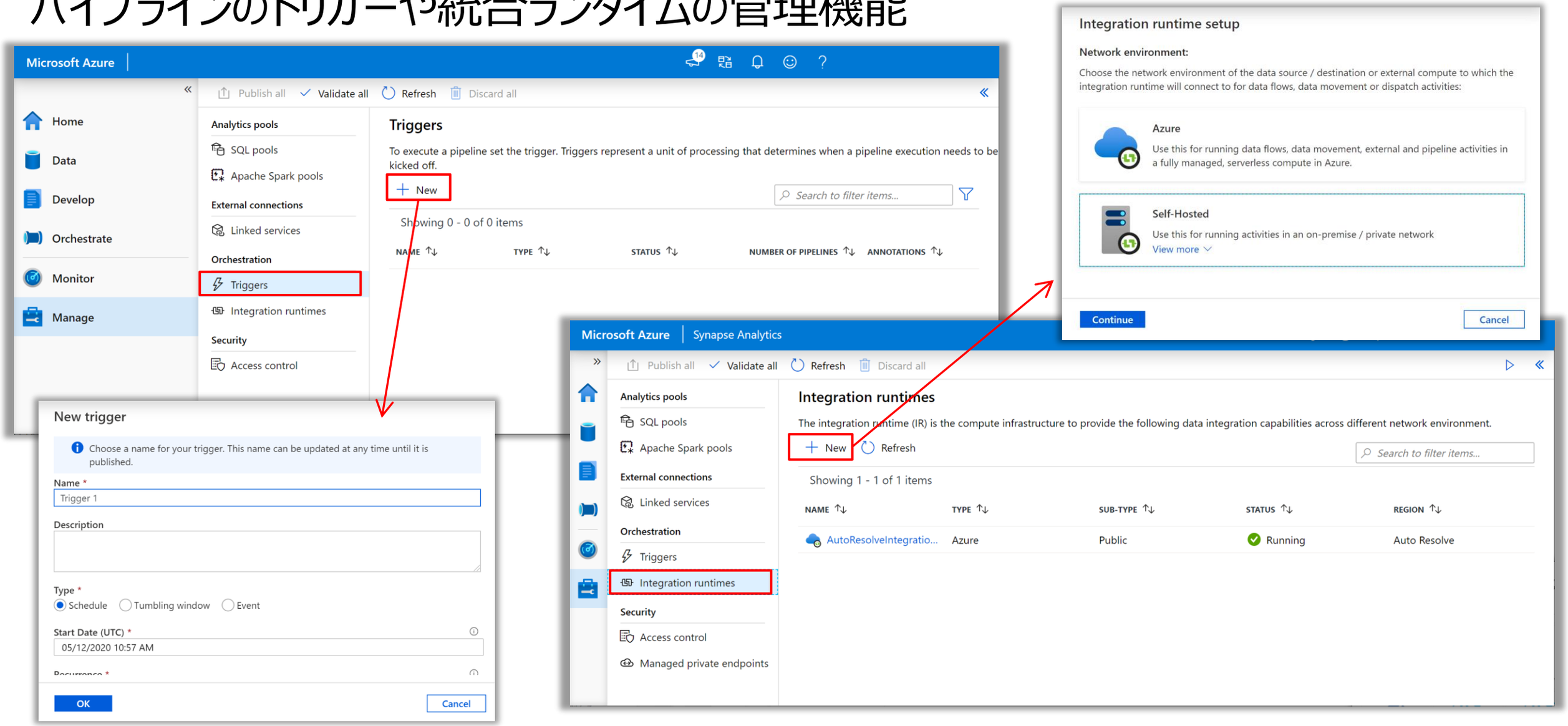
Manage Hub - Linked サービス

パイプラインの外部接続定義の管理機能



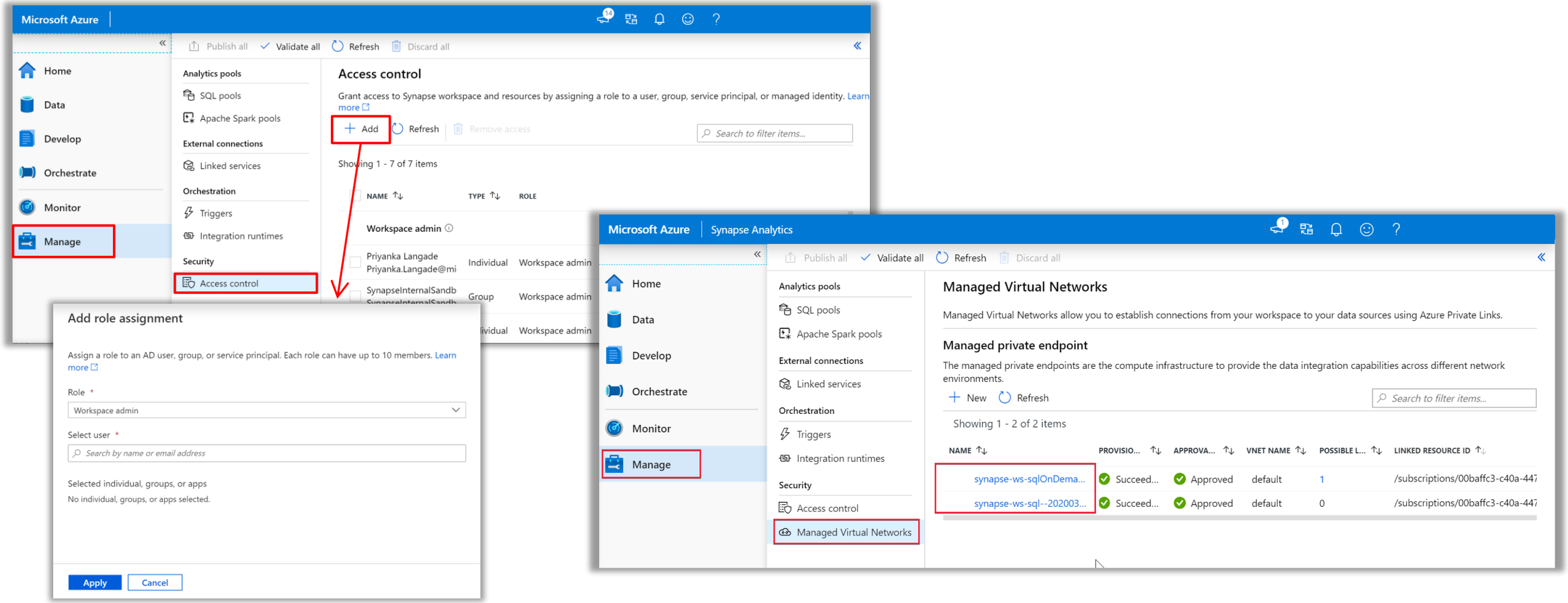
Manage Hub - オーケストレーション

パイプラインのトリガーや統合ランタイムの管理機能



Manage Hub - セキュリティ

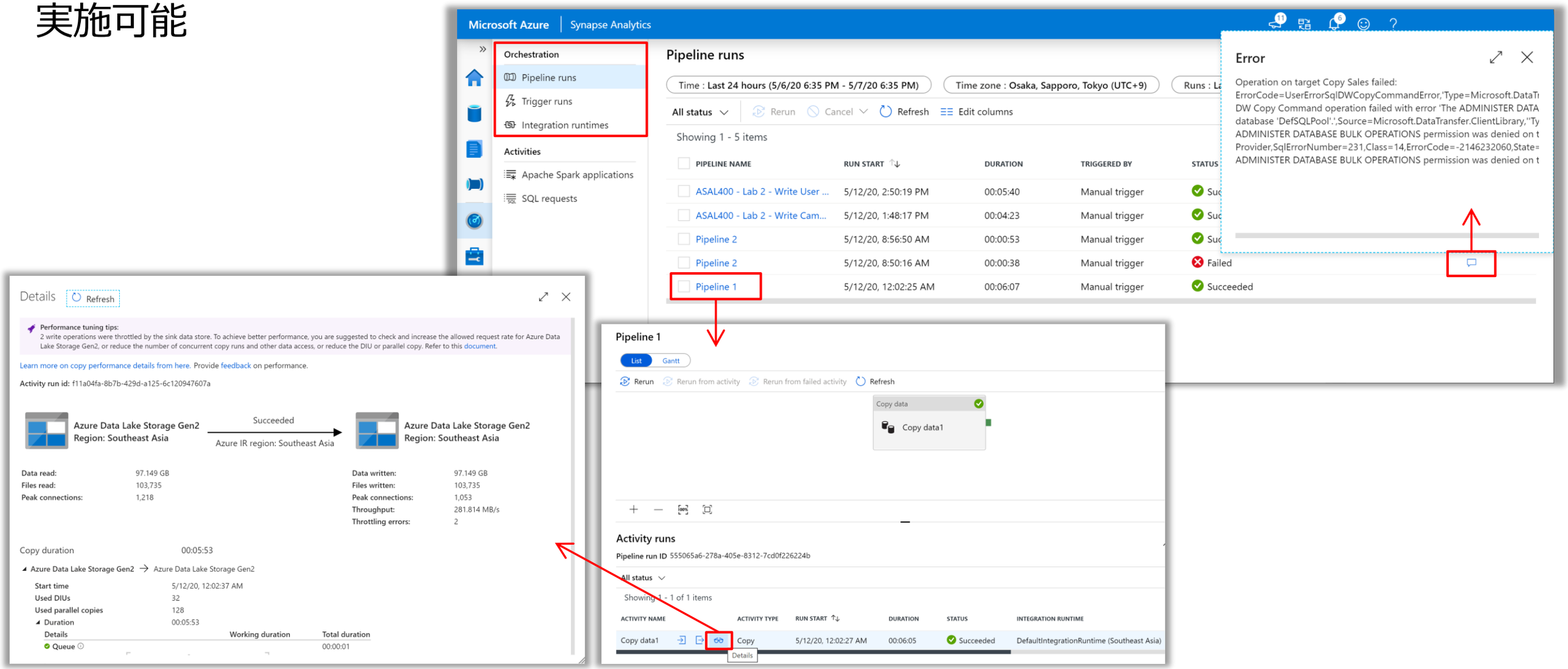
- ワークスペース リソースやアーティファクトに対するアクセス制御機能
- マネージド仮想ネットワーク に配置するプライベート エンドポイントの管理機能



Synapse Studio
Monitor Hub

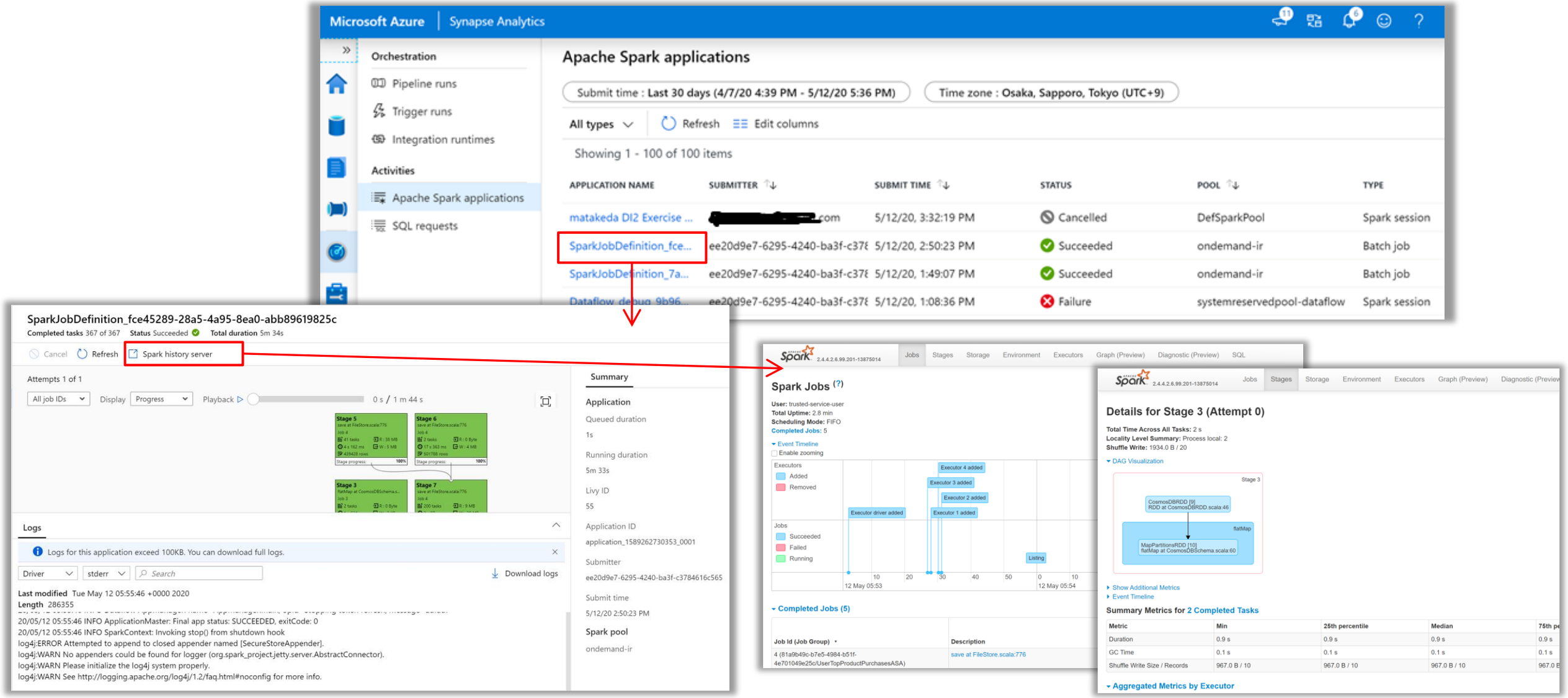
Monitor Hub - オーケストレーション

パイプラインやパイプライン アクティビティの実行詳細/履歴の確認、失敗原因の調査、リランなどを実施可能



Monitor Hub - Spark アプリケーション

Spark アプリケーションの実行詳細/履歴の確認、失敗原因の調査、リランなどを実施可能



■ 本書に記載した情報は、本書各項目に関する発行日現在の Microsoft の見解を表明するものです。Microsoftは絶えず変化する市場に対応しなければならないため、ここに記載した情報に対していかなる責務を負うものではなく、提示された情報の信憑性については保証できません。
■ 本書は情報提供のみを目的としています。 Microsoft は、明示的または暗示的を問わず、本書にいかなる保証も与えるものではありません。
■ すべての当該著作権法を遵守することはお客様の責務です。Microsoftの書面による明確な許可なく、本書の如何なる部分についても、転載や検索システムへの格納または挿入を行うことは、どのような形式または手段（電子的、機械的、複写、レコーディング、その他）、および目的であっても禁じられています。
これらは著作権保護された権利を制限するものではありません。
■ Microsoftは、本書の内容を保護する特許、特許出願書、商標、著作権、またはその他の知的財産権を保有する場合があります。Microsoftから書面によるライセンス契約が明確に供給される場合を除いて、本書の提供はこれらの特許、商標、著作権、またはその他の知的財産へのライセンスを与えるものではありません。
© 2020 Microsoft Corporation. All rights reserved.
Microsoft, Windows, その他本文中に登場した各製品名は、Microsoft Corporation の米国およびその他の国における登録商標または商標です。
その他、記載されている会社名および製品名は、一般に各社の商標です。