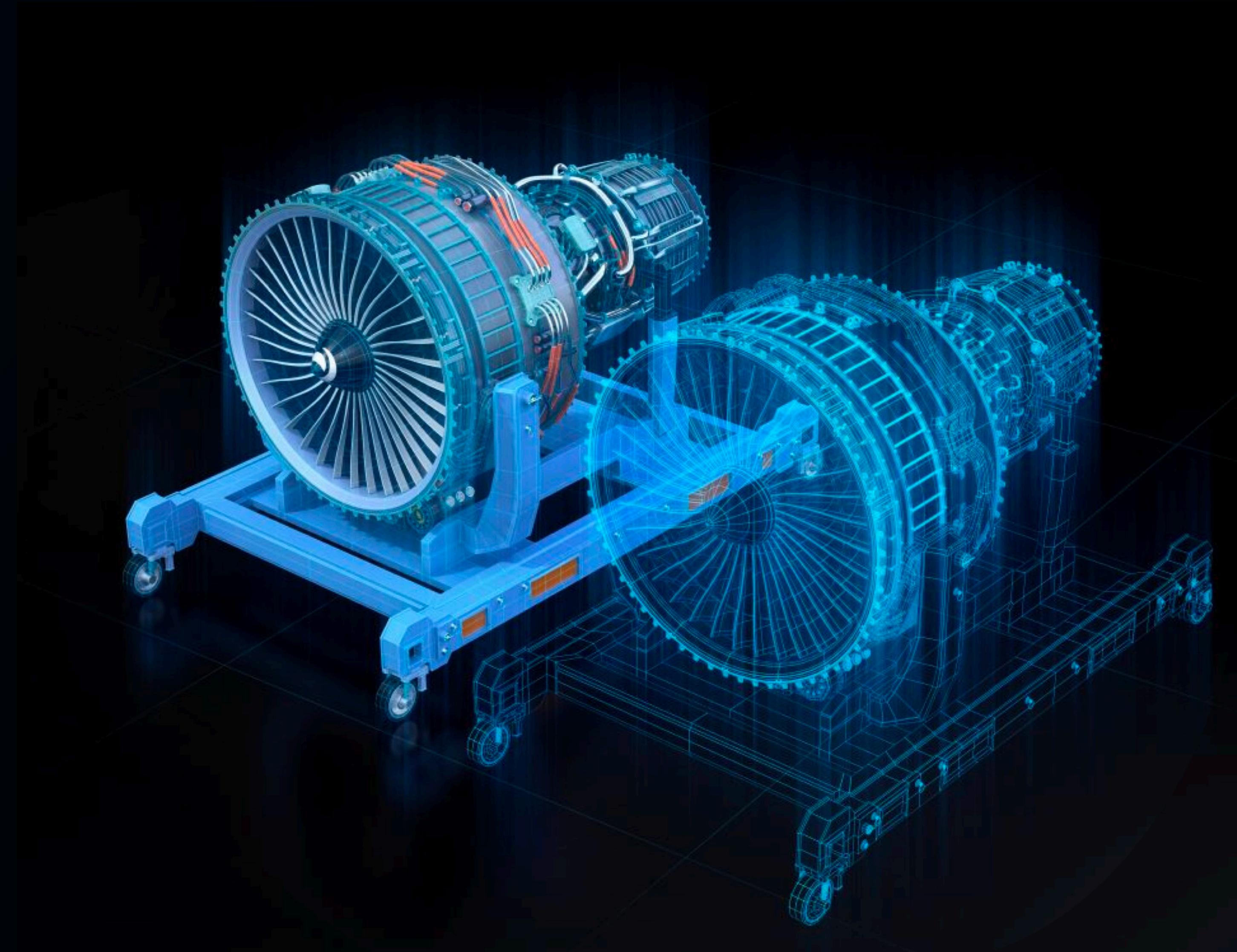


Digital Twin

Rewolucja IoT w Biznesie

Marek Serba 23.03.2023



Marek Serba

Pracuje w IT od przeszło 14 lat

Azure Cloud Platform Manager

w firmie:



Links:

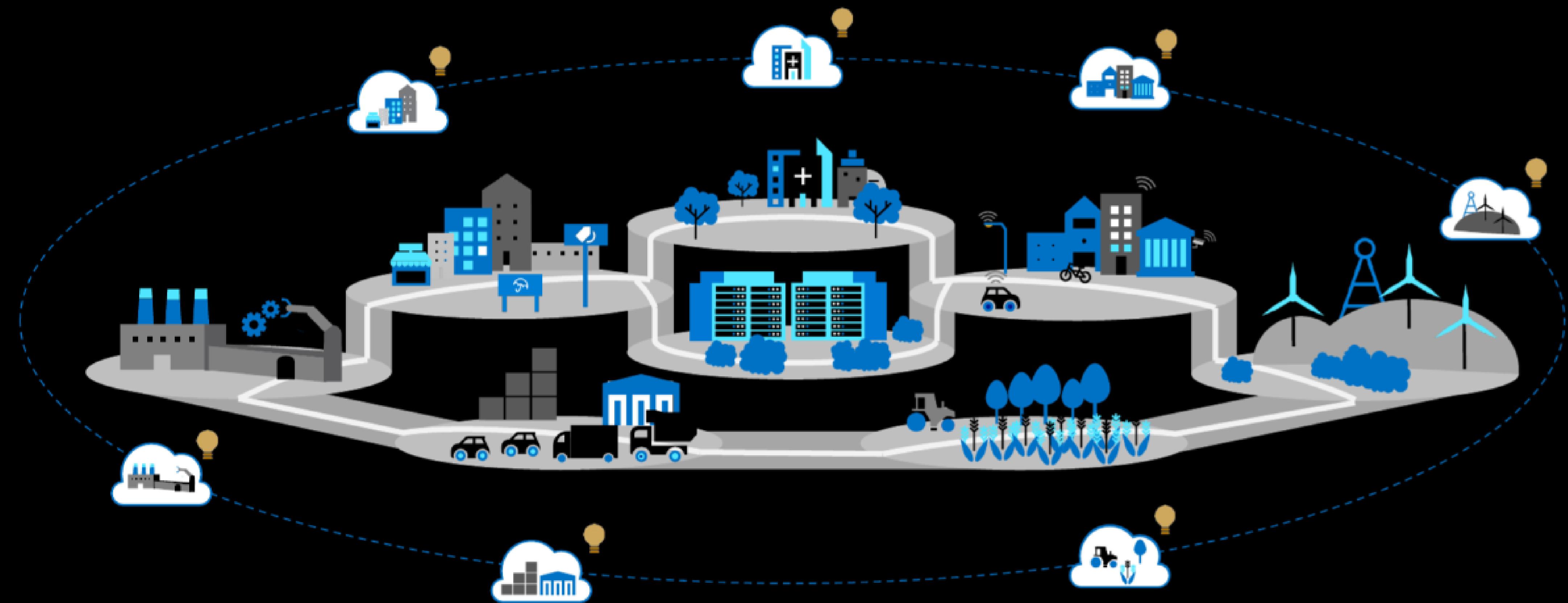
- linkedin.com/in/techfellow
- github.com/technicalflow
- twitter.com/technicalflow
- mysmall.cloud



“ Measurement is the first step... Measurement is the first step that leads to control and eventually to improvement. If you can't measure something, you can't understand it.”

H. James Harrington

Digital Twins = Digital Replica of real-world things, places, business processes, and people.



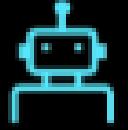
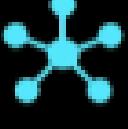
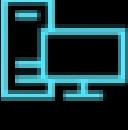
Co daje nam cyfrowe odzwierciedlenie świata?

3 Kluczowe aspekty które dają nam pozwalają lepiej wykorzystać dostępne narzędzia to:

- Metryki i Alerty
- Analiza i wizualizacja danych
- Modelowanie ML na danych



Updated Capabilities with New Version of Azure Digital Twins

	PREVIOUS VERSION (2018 Preview)	NEW VERSION (Generally Available)
 Flexible Modeling	Smart Spaces use cases focused	Create custom models with Digital Twins Definition Language (DTDL)
 Flexible Topology	Trees / hierarchies tailored to Smart Spaces use cases	Flexible general graph topology
 Richer Compute	Only Java script and limited debugging with User Defined Functions (UDFs)	Open compute model using external compute for event and data processing
 Accessible Hub	IoT Hub within Azure Digital Twins	Bring your own IoT Hub to integrate IoT data with Azure Digital Twins
 Scalability	Scale limitations for devices, messages, graphs, scale units	New architecture with improved scalability

Czym jest Azure Digital Twin ?

Usługa Azure Digital Twins to oferta typu platforma jako usługa (PaaS), która umożliwia tworzenie grafów bliźniaczych na podstawie modeli cyfrowych całych środowisk, które mogą być budynkami, fabrykami, farmami, sieciami energetycznymi, kolejami, stadionami i nie tylko – nawet całymi miastami.

Te modele cyfrowe mogą służyć do uzyskiwania szczegółowych informacji, które umożliwiają lepsze produkty, zoptymalizowane operacje, mniejsze koszty i przełomowe środowiska klientów.

Usługa Azure Digital Twins może służyć do projektowania architektury cyfrowej reprezentacji bliźniaczej, która reprezentuje rzeczywiste urządzenia IoT w szerszym rozwiązaniu w chmurze i łączy się z bliźniaczymi reprezentacjami urządzeń IoT Hub w celu wysyłania i odbierania danych na żywo.

Czym jest Azure Digital Twin pod spodem ?

Azure Digital Twin Platform (ADT)

The diagram illustrates the Azure Digital Twin Platform (ADT) architecture, structured into several layers:

- SDKs & Azure CLI:** This layer includes icons for Microsoft .NET (blue 'N' logo), Golang (gopher icon), Python (Python logo), Java (Java logo), JavaScript (JS logo), and PowerShell (greater than sign icon).
- REST API:** Represented by a hexagonal icon.
- Twin Definitions:** Represented by a gear-like icon.
- Twin Schema (DTDL):** Represented by a JSON file icon.
- Graph Database:** Represented by a network graph icon.

Co wnoszą Microsoft do gry ?

Open Modeling Language - DTDL

Digital Twin Definition Language

... JSON

```
1 [  
2 {  
3     "@id": "dtmi:demo:Factory;1",  
4     "@type": "Interface",  
5     "@context": "dtmi:dtdl:context;2",  
6     "displayName": "Factory Interface Model",  
7     "contents": [  
8         {  
9             "name": "Temperature",  
10            "@type": "Property",  
11            "schema": "double"  
12        },  
13        {  
14            "name": "Humidity",  
15            "@type": "Property",  
16            "schema": "double"  
17        }]  
18    ]  
19 }]  
20 ]
```

DIGITAL TWIN MODELING IN DTDL

PROPERTIES

TELEMETRY

COMPONENTS

RELATIONSHIPS



QUERY EXPLORER

Saved Queries

SELECT * FROM digitaltwins

Overlay results

Run Query



TWINS

MODELS

TWIN GRAPH

MODEL GRAPH



Search

Base Consumer ...

dtmi:example:grid:consumer:base...

Base Plant ...

dtmi:example:grid:plants:basePlan...

Base Receiver ...

dtmi:example:grid:transmission:ba...

City Plant ...

dtmi:example:grid:plants:cityPlant;1

Coal Plant ...

dtmi:example:grid:plants:coalPlant;1

Delivery SubStation ...

dtmi:example:grid:transmission:de...

Domestic Consumer ...

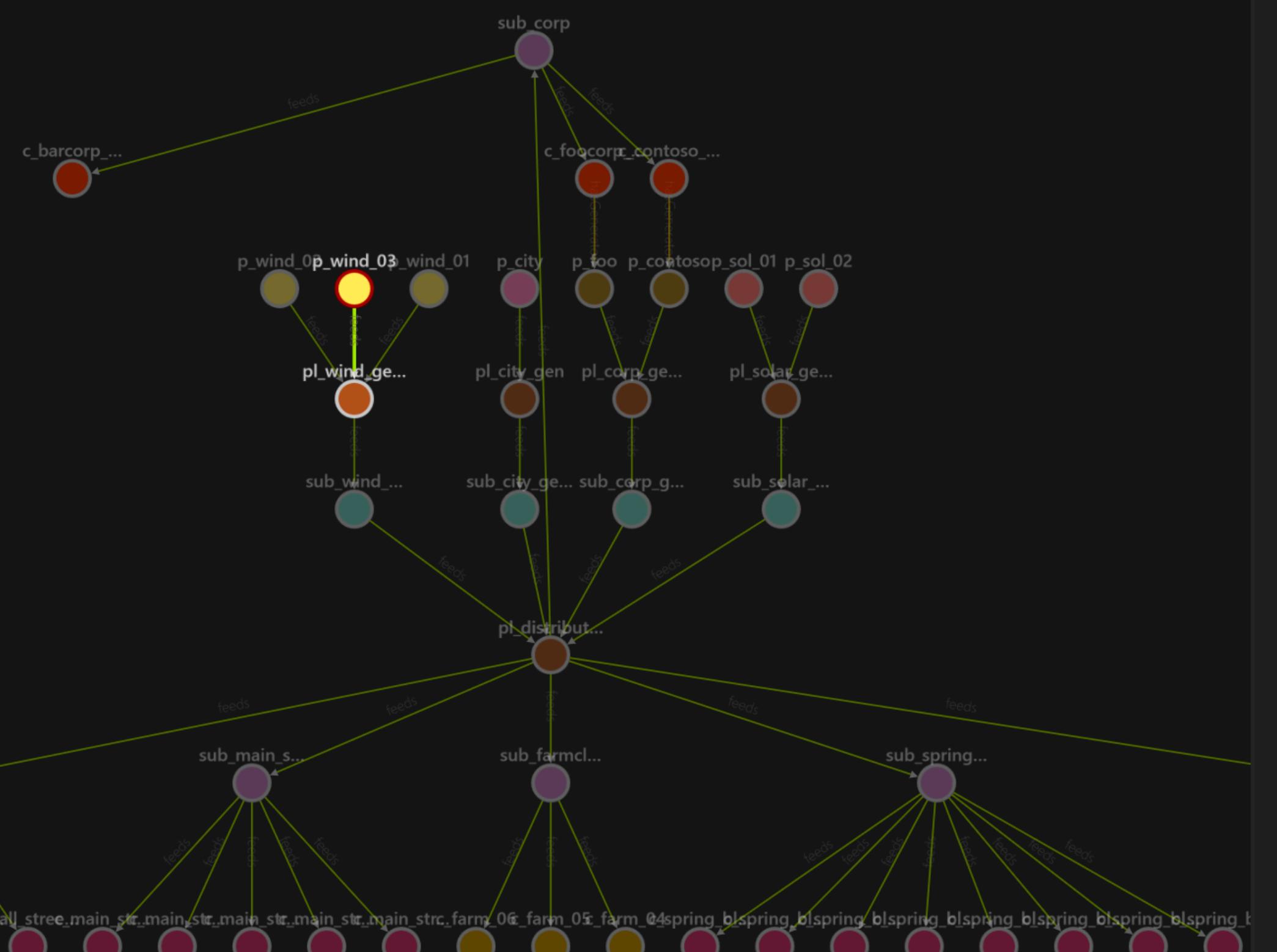
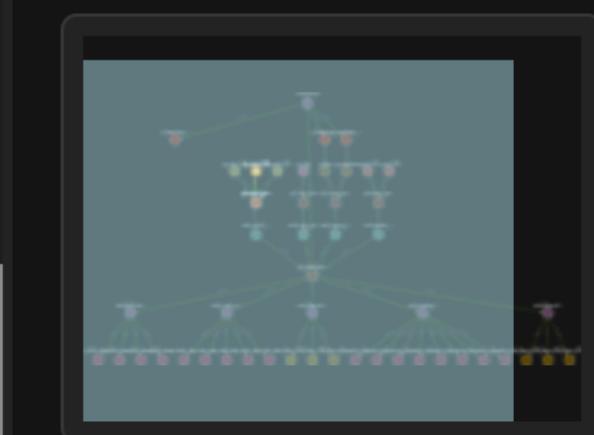
dtmi:example:grid:plants:domesti...

Farm Consumer ...

dtmi:example:grid:plants:farmCon...

Floor ...

dtmi:example:Floor;1



TWIN PROPERTIES

\$dtId: p_wind_03

Output: 3000 X

EmissionType: -- (not set)

> ManufacturerInfo: (not set)

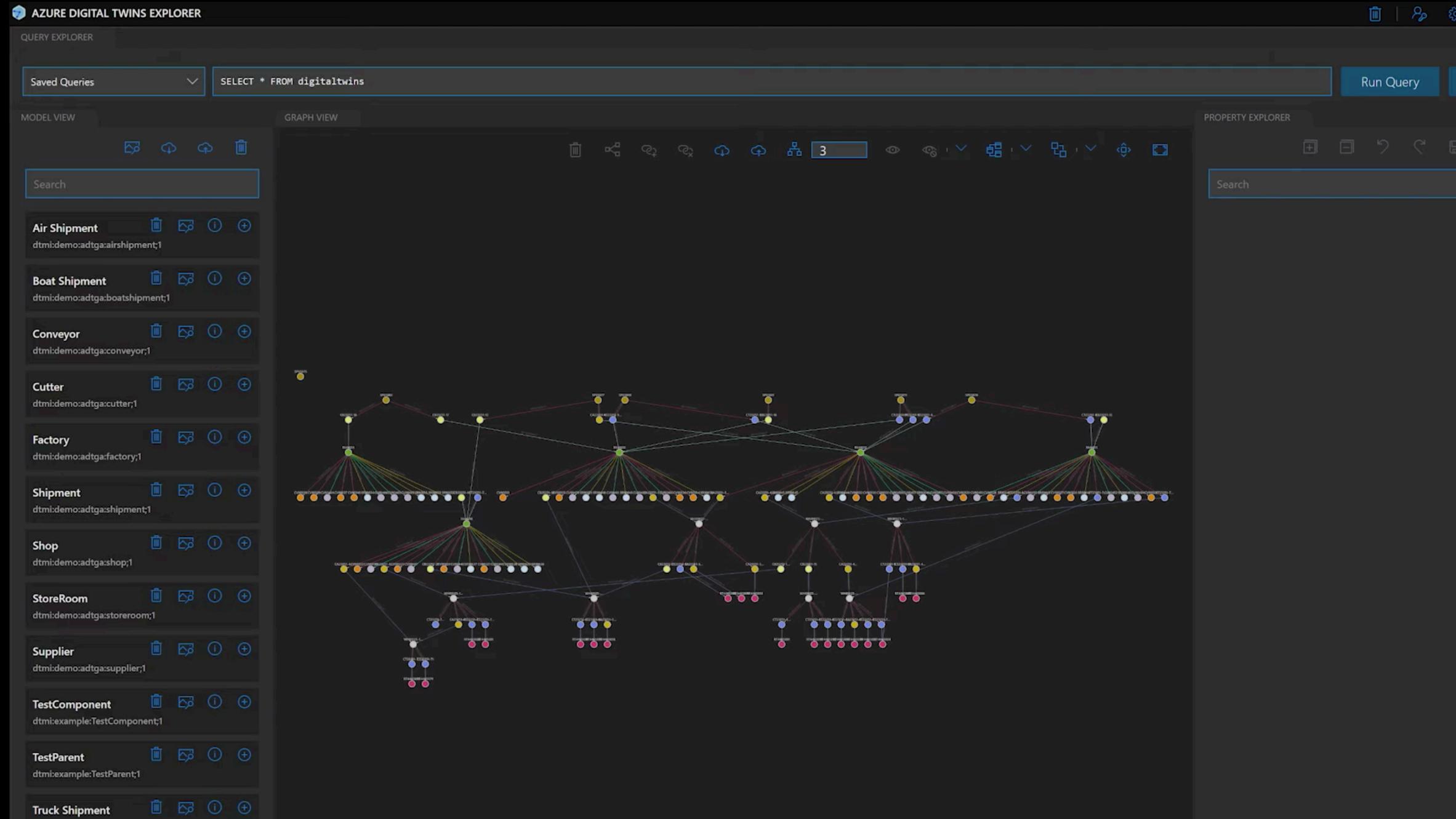
\$etag: W/"c93e0b41-5929-4b67-9550-71993b1d3b58"

> \$metadata:



Azure Digital Twin Explorer (Preview)

Rozwiązanie OpenSource do wizualizacji czujników i połączeń



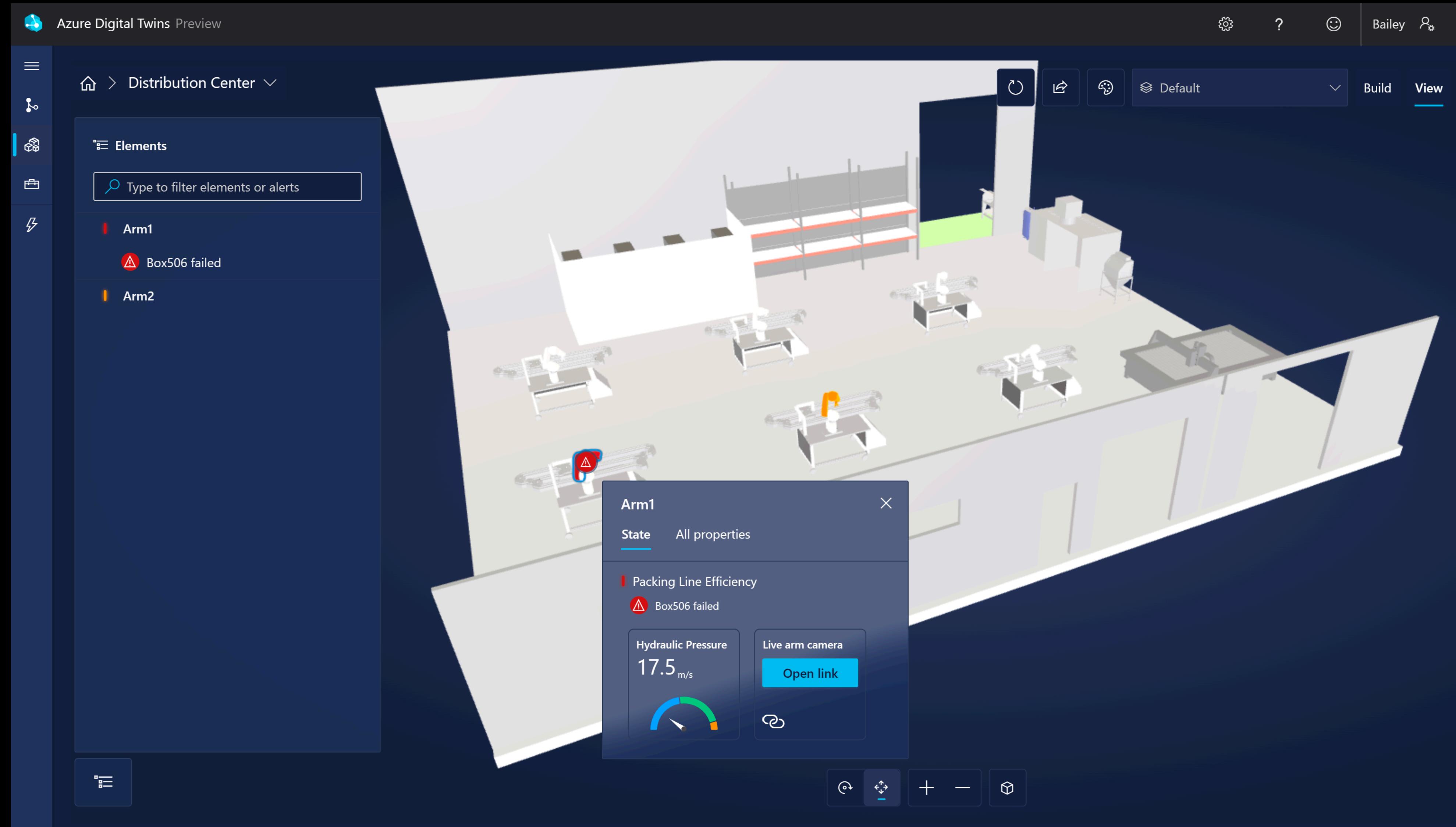
Azure Digital Twin Explorer

Azure Digital Twins Explorer jest narzędziem do wizualnej reprezentacji danych zgromadzonych w formie grafowej w usłudze Azure Digital Twin.

Możemy użyć tego narzędzie do:

- Podglądu danych
- Zapytań do bazy
- Wdrożenia modeli DTDL
- Edycji modeli
- Edycji połączeń

3D Scenes Studio



Jak działa model kosztowy ?

Region:	Currency:
West Europe	United States – Dollar (\$) USD
Message	Price
Message	\$1.30 per million messages
Operation	Price
Operation	\$3.25 per million operations
Query unit	Price
Query unit	\$0.65 per million query units

Customers only pay for what they consume.

Message - message send to ADT

Operation - API Call to Control Plane or Data Plane

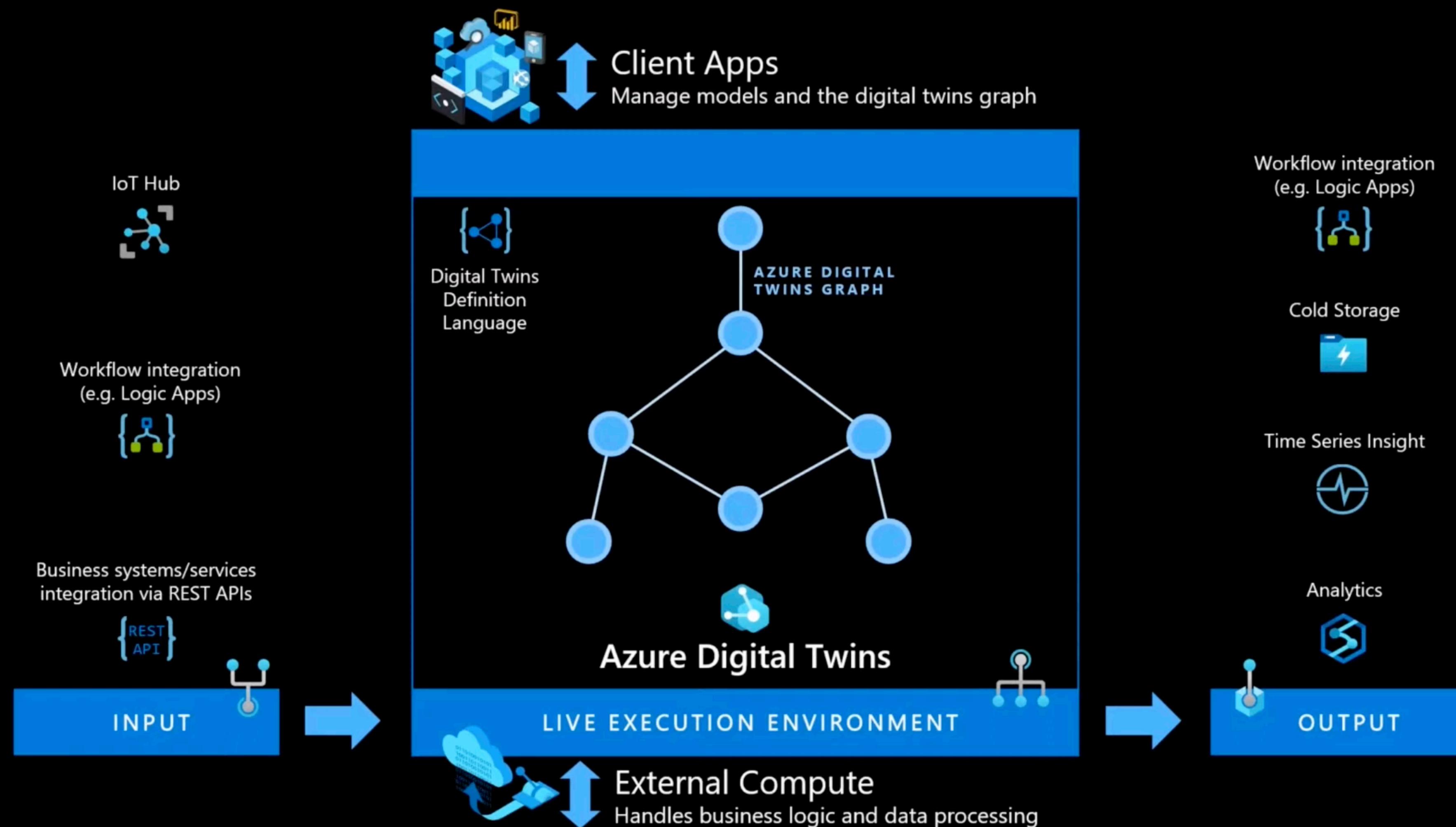
Query unit - Unified metric for CPU, MEM, IOPS

What is a Query unit in Azure Digital Twins?

Query units are the currency for query execution in Azure Digital Twins. Each query execution in Azure Digital Twins consumes CPU, memory and IOPs resources. These resources incur charges which are expressed in Query Units (QUs). Look for "query-charge" in the Query API response to track the number of QUs consumed by the query. To learn more, please visit [Query Units in Azure Digital Twins](#) page.

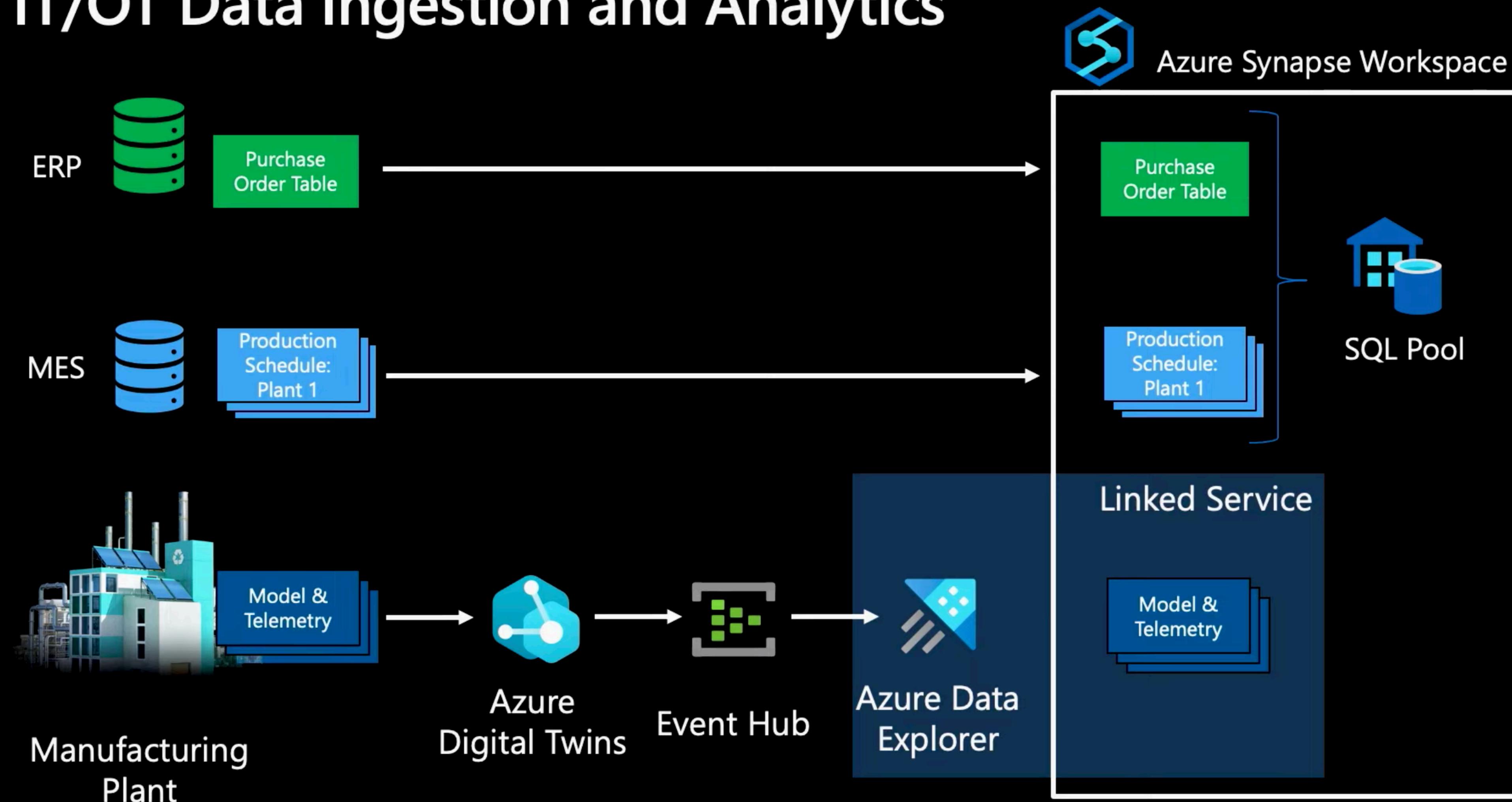
Data flow

Create next generation IoT solutions that model the real world



Machine Learning i symulacja przyszłych zachowań

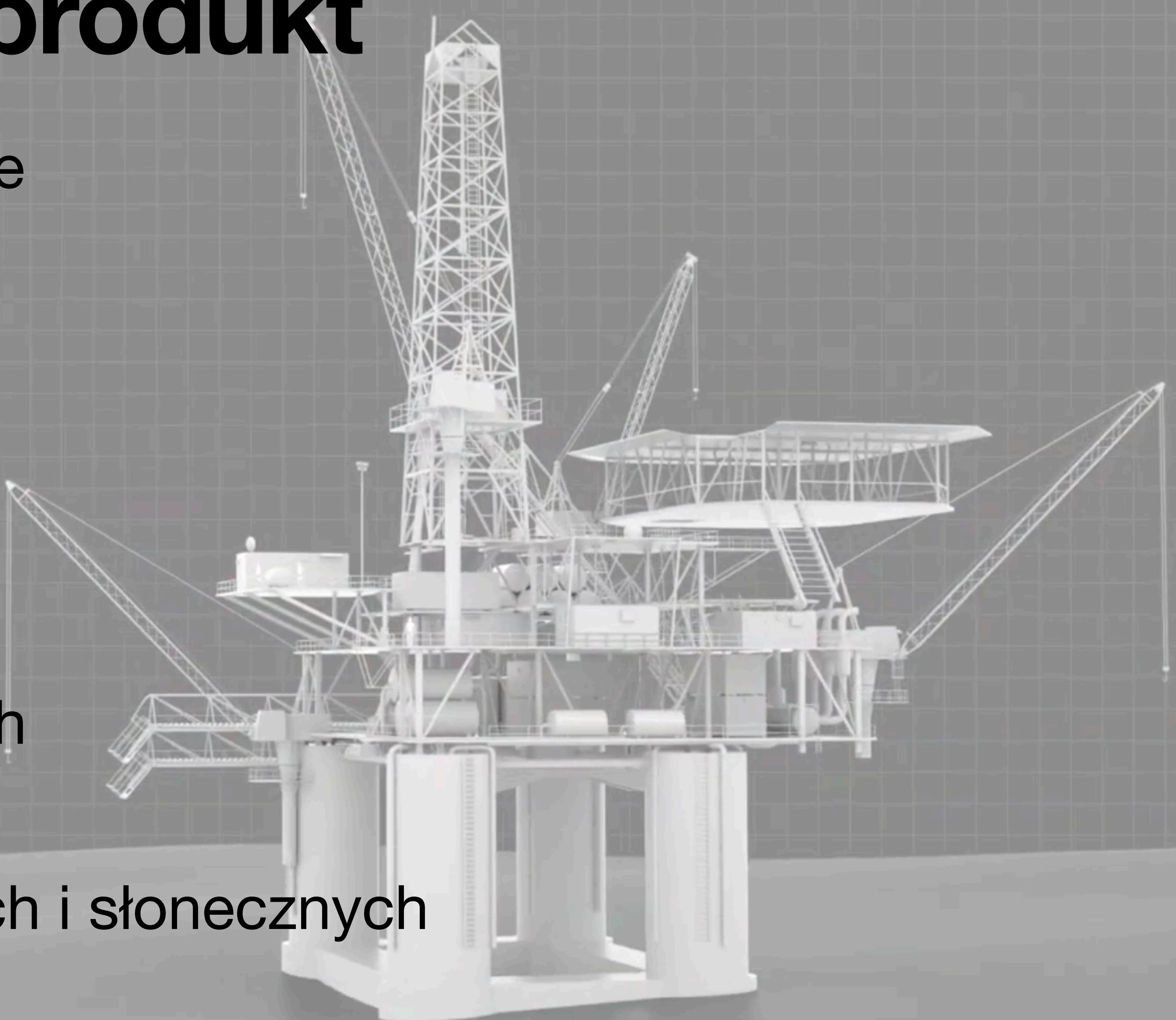
IT/OT Data Ingestion and Analytics





Dla kogo jest ten produkt

- Rafinerie i platformy wiertnicze
- Fabryki
- Transport
- Sterowanie ruchem
- Kolej
- Centra danych
- Elementy sieci energetycznych
- Elektrownie
- Optymalizacja farm wiatrowych i słonecznych
- Rolnictwo
- I wiele, wiele więcej zastosowań ...



Digital Twin w Lotos

Najnowocześniejszy projekt w Polsce
i top projektów w Europie

„Bardziej trafne przewidywanie awarii i odpowiednie zarządzanie ryzykiem potwierdzi, że wydłużenie okresu eksploatacji między poszczególnymi postojami remontowymi jest realizowane zgodnie z najwyższymi standardami bezpieczeństwa. Dla rafinerii każdy dzień dłuższego wykorzystania pełnych mocy przetwórczych ma olbrzymie korzyści i przynosi konkretne oszczędności finansowe mierzone w milionach złotych. Jeśli posiadamy narzędzia, które pozwolą na przewidywanie zmian, będziemy mogli podejmować bardziej trafne decyzje. Dodatkowo wykorzystanie takich wysokospecjalistycznych, cyfrowych narzędzi uwiarygodni nas przed naszym partnerem Urzędem Dozoru Technicznego, który decyduje o dopuszczeniu naszych urządzeń do eksploatacji” – przekonuje Tomasz Branicki, dyrektor ds. techniki w spółce Rafineria Gdańska Sp. z o.o., należącej do Grupy LOTOS (dawniej: Lotos Asfalt).

Digital Twin w Lotos

Problemy rafinerii

- Badanie sprzętu wydobywczego dwa razy do roku
- Wstrzymanie prac na około 2 tygodnie
- Koszt przestoju przekraczają znacząco koszt części
- UDT musi nadzorować pracę i tworzyć raporty

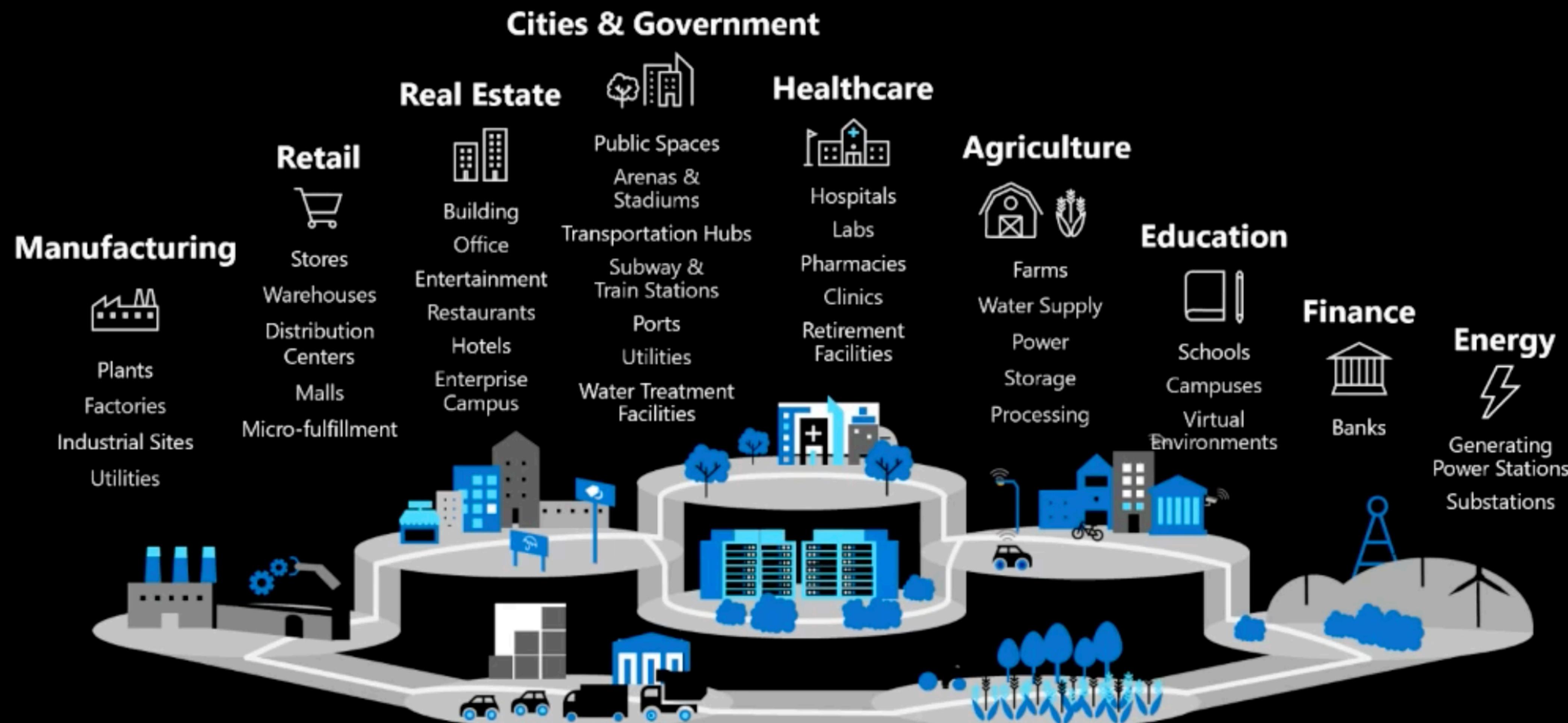
Największy projekt tego typu w Polsce

Digital Twin w Lotos

Dzięki Digital Twin:

- Najbardziej aktualny monitoring zużycia komponentów
- Zaawansowane metryki całego środowiska
- Analiza i optymalizacja na bieżąco procesu wydobycia
- Możliwość przewidywania przestoju i zmian
- Możliwość przewidywania stopnia zużycia części
- Lepsze zarządzanie ryzykiem
- Zwiększenie bezpieczeństwa

Connected Environments



Dziękuję za uwagę :)

LINKI:

Badanie Gartnera

<https://www.gartner.com/smarterwithgartner/prepare-for-the-impact-of-digital-twins>

DTDL:

Like schema

<https://www.youtube.com/watch?v=iPIrjQGDvmM>

Azure Digital Twin Explorer demo:

<https://explorer.digitaltwins.azure.net/3DScenes/demo>

IOT Blog

<https://techcommunity.microsoft.com/t5/internet-of-things-blog/how-can-digital-twins-make-smart-grids-a-reality/ba-p/3749640>

Elements of a model

Within a model definition, the top-level code item is an [interface](#). This type encapsulates the entire model, and the rest of the model is defined within the interface.

A DTDL model interface may contain zero, one, or many of each of the [following fields](#):

- **Property** - Properties are data fields that represent the state of an entity (like the properties in many object-oriented programming languages). Properties have backing storage and can be read at any time. For more information, see [Properties and telemetry](#) below.
- **Telemetry** - Telemetry fields represent measurements or events, and are often used to describe device sensor readings. Unlike properties, telemetry isn't stored on a digital twin; it's a series of time-bound data events that need to be handled as they occur. For more information, see [Properties and telemetry](#) below.
- **Relationship** - Relationships let you represent how a digital twin can be involved with other digital twins. Relationships can represent different semantic meanings, such as *contains* ("floor contains room"), *cools* ("hvac cools room"), *isBilledTo* ("compressor is billed to user"), and so on. Relationships allow the solution to provide a graph of interrelated entities. Relationships can also have properties of their own. For more information, see [Relationships](#) below.
- **Component** - Components allow you to build your model interface as an assembly of other interfaces, if you want. An example of a component is a *frontCamera* interface (and another component interface *backCamera*) that are used in defining a model for a *phone*. First define an interface for *frontCamera* as though it were its own model, and then reference it when defining *Phone*.

Digital Twins Definition Language

- Interface
 - Digital twin type
- Property
 - State of a digital twin
 - Read only or writable
 - Implies synchronization for remote entities
- Telemetry
 - Measurement of a digital twin
 - Implies timestamped stream of measurements
- Command
 - Method on a digital twin
 - Implies "time to start" and "time to complete" for remote entities or async operations
- Relationship
 - Reference type containment in a digital twin
 - 1:n collection
 - Circular references allowed
 - Relationships support properties
- Component
 - Value type containment in a digital twin
 - No circular references
- Schema
 - Data type definition
 - Standard primitive schemas provided for strings, numbers, dateTimes, etc.
 - Complex schema types for arrays, enums, objects, and maps
- Inheritance
 - Interface subclassing
 - Twin instances have exactly one interface type
 - Interfaces can extend multiple other interfaces
- Semantic Type
 - Annotations on DTDL elements (additional "semantics")
 - Example: measurement type and unit