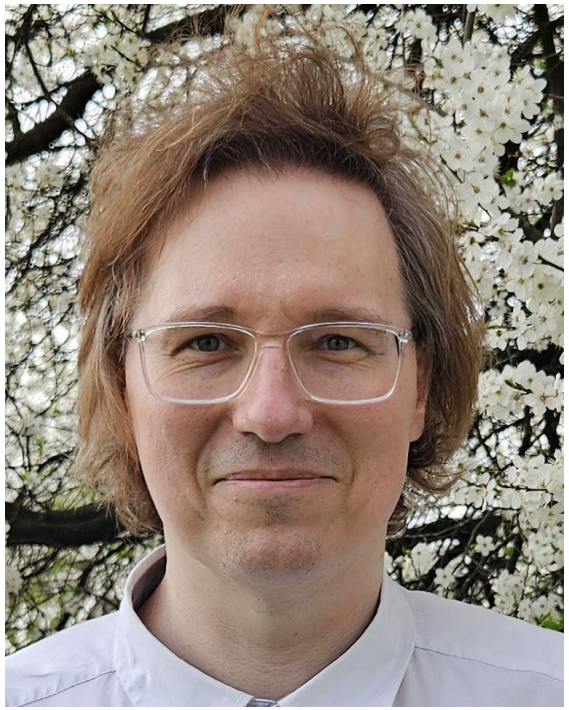
# (Azure) Open AI - how to work with information, not with data warehouse(s)

Tomasz Kopacz

Architect, Microsoft



SQL Day

# 16 edycja konferencji SQLDay

Data Community

13-15 maja 2024, WROCŁAW + ONLINE

partner platynowy

lingaro

partner złoty

DXC TECHNOLOGY

VOLVO

software one

partner srebrny

accenture

Nordcloud an IBM Company

Capgemini

TECHNOLOGY INNOVATION DATA KNOWLEDGE hdk

elitmind think bright

C&F

# Case: Doctors, CRM, opinions…

EMR = Forms, boring, structure, next, next

**Free Text in doctor notes = real value**

SALES/CRM Systems

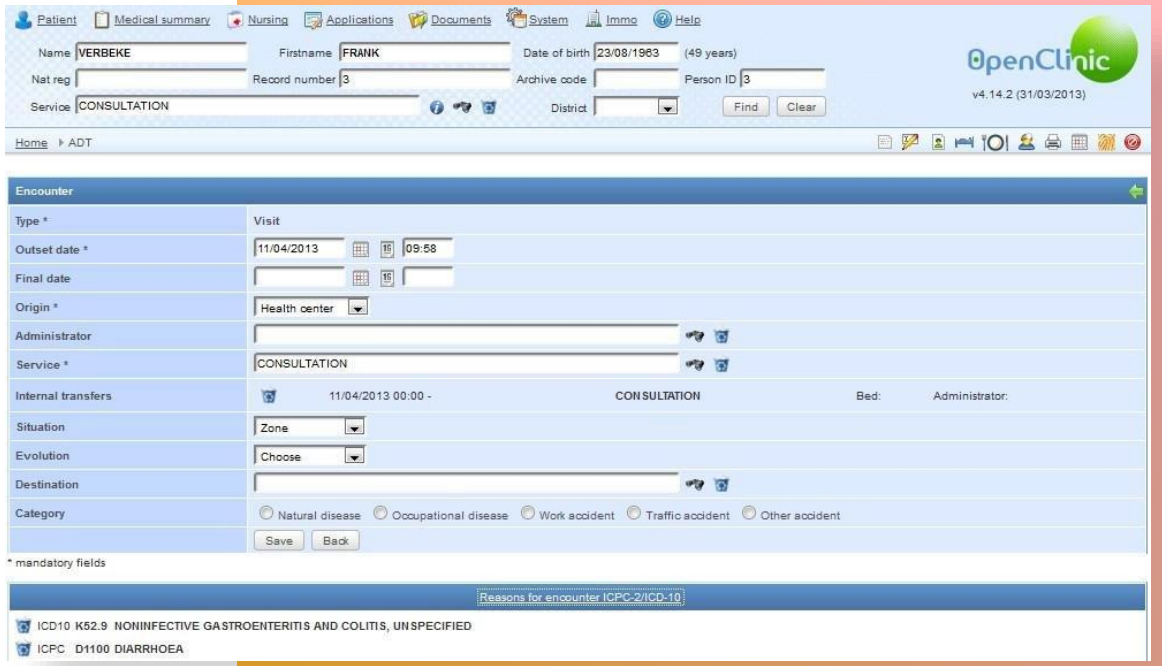
Records… yes

Notes/comments/discussions/team(s) chat – what's REALLY going on.

Conversation with suppliers

Measure of success / opinion on comments vs asking, "rate from 1 to 5 how you like us". How to extract area of complain?

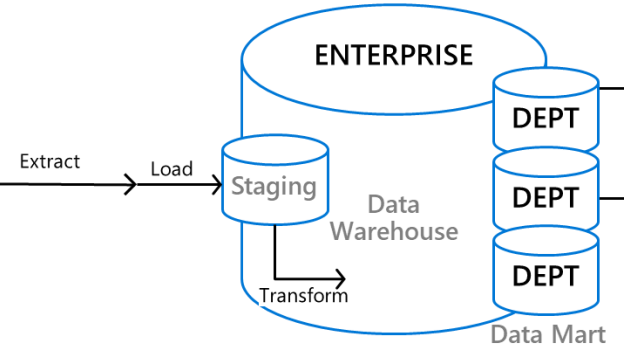
Financial Reports – tables, graphs, but – who will understand them? CEO needs recommendation!



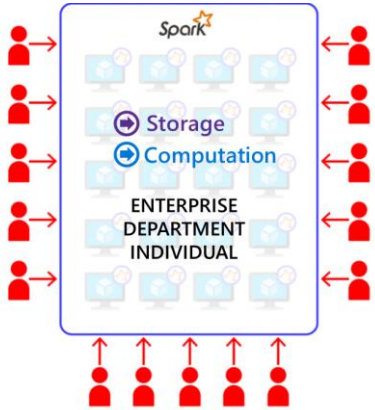
# But, for last 40+ years:

# High Level Data architecture patterns and benefits

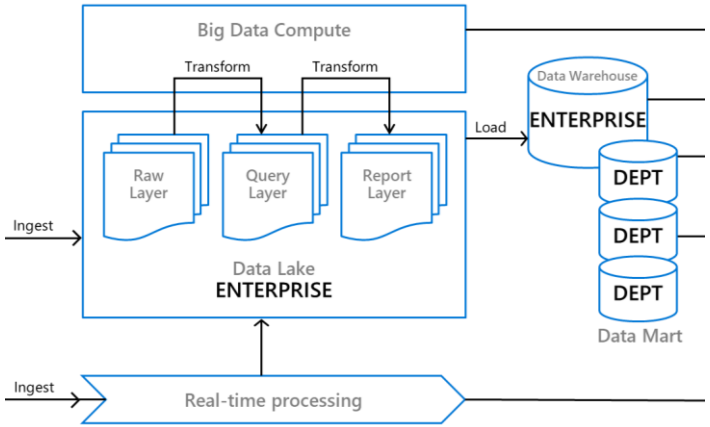
## Late 1980s
Data Warehouse



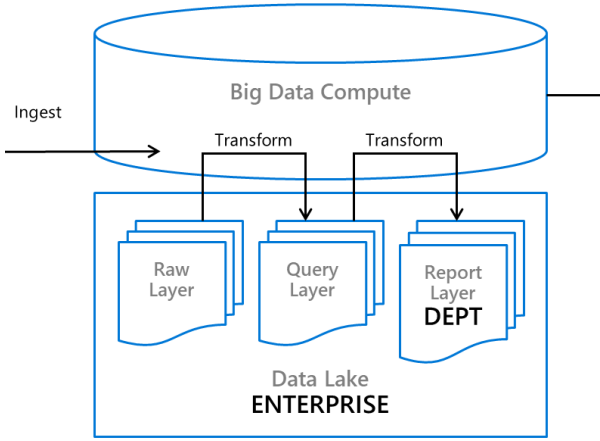

## Late 2000s
Data Lake




## Mid 2010s
Cloud Data Platform
DWH **and** Data Lake



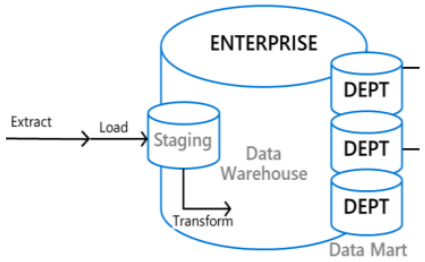

## Early 2020s
Data Lake House
(OneLake)



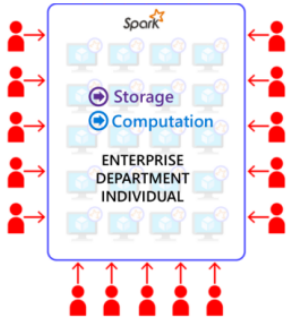

*The right architecture and services within a platform depend on use cases, personas, workloads, regulatory requirements, data standards, security, interoperability, etc. There is no "good" or "bad" architecture, only architectures that are less popular during specific timeframes.*

# And… Data Mesh!

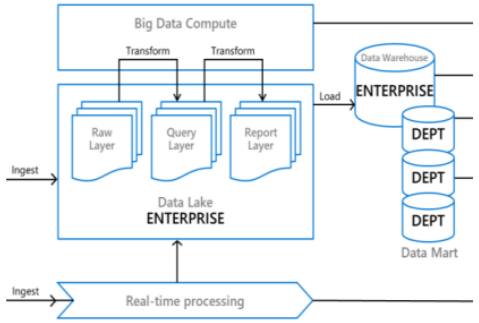
**Late 1980s**
Data Warehouse



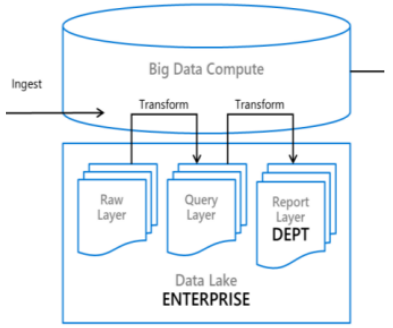

**Late 2000s**
Data Lake



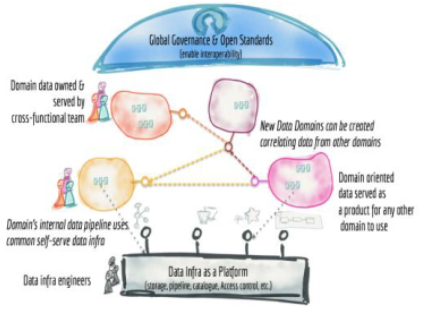

**Mid 2010s**
Cloud Data Platform




**2020**
Data Lakehouse




**2021**
**Data Mesh??**




Centralization

Decentralization

Awareness of Data Mesh is high in the Health Service sector and other industries.

*How to approach that "free text"?*

# Large Language Models

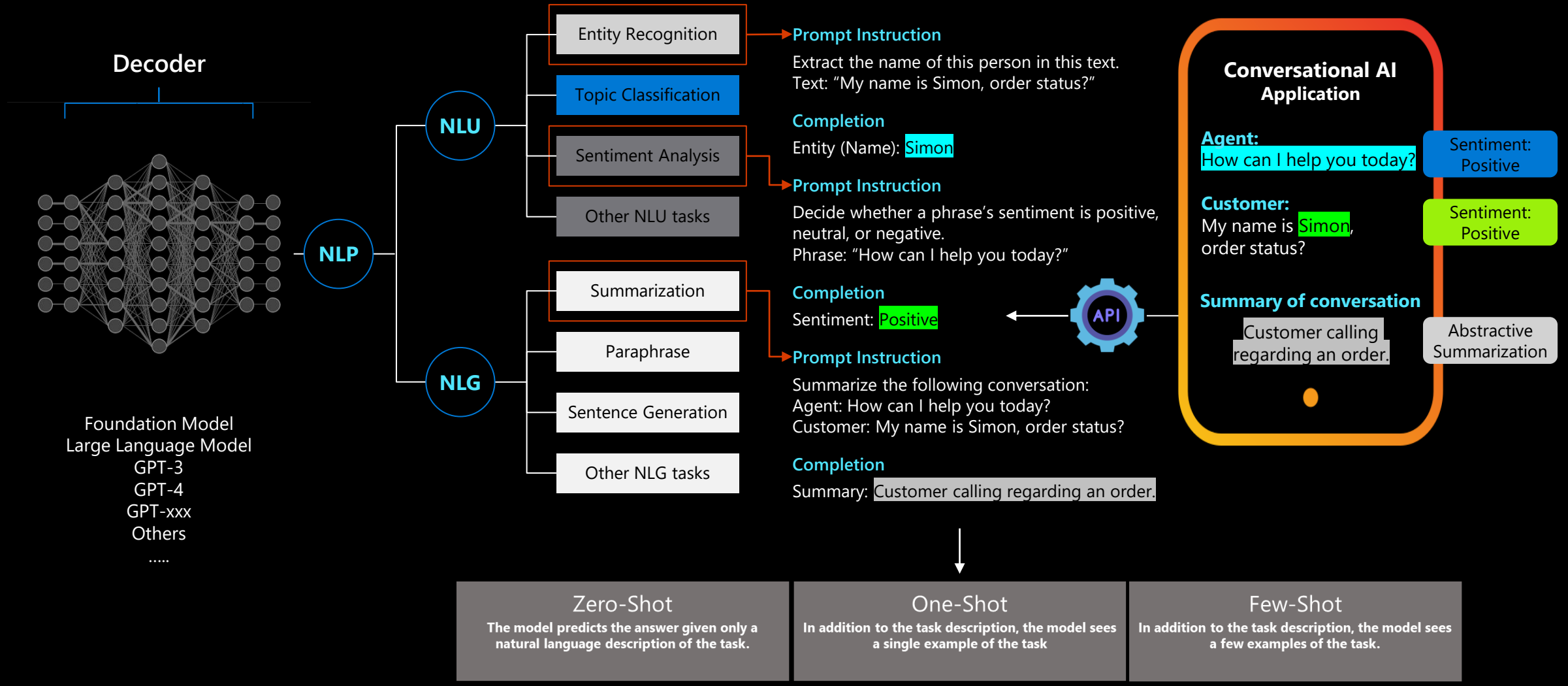
**Transformer Architecture**: Utilizes self-attention mechanisms to understand context, improving natural language processing capabilities.

**Scalability**: Enhances performance with increased parameters, supported by distributed computing to handle complex tasks and large datasets. (big hammer!)

**Pre-training and Fine-tuning**: Initially trained on broad datasets to learn language patterns, then fine-tuned on specific tasks to improve task-specific performance. Can be fine-tuned more, but in many cases – not needed!

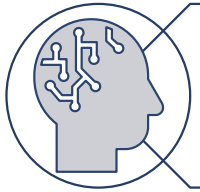
# All you need is a Prompt

## Model use out of the box—prompting, in-context learning

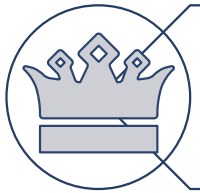



Command written in HUMAN language to analyze TEXT!

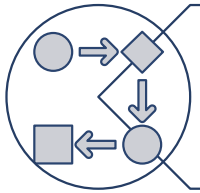
# Building LLM based product, we need to think:



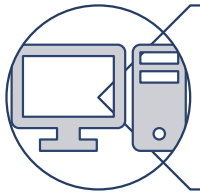
Models will change. Prompts will change



Strategy for larger context/documents/tasks



Orchestrated call**s** to AOI to get PROPER answer



Storage for vectors and tool for similarity search

Test prompts & Feedback loop

Red Team of experts

# Our tool: Semantic Kernel?

**NEVER, EVER** call AOI and forward directly response to UI

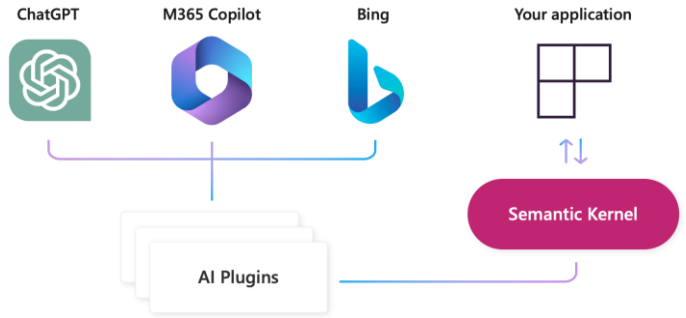
Semantic Kernel – lightweight **Pipeline** Builder for semantic programming with **plugins***(ex skills)* and **memory** (**embeddings**)

Also: Microsoft.DeepDev.TokenizerLib

Also: Kernel Memory (RAG pattern)

(and – personal – EXCELLENT binding and integration with for C#)

# Plugins? *(aka functions, skills, …)*



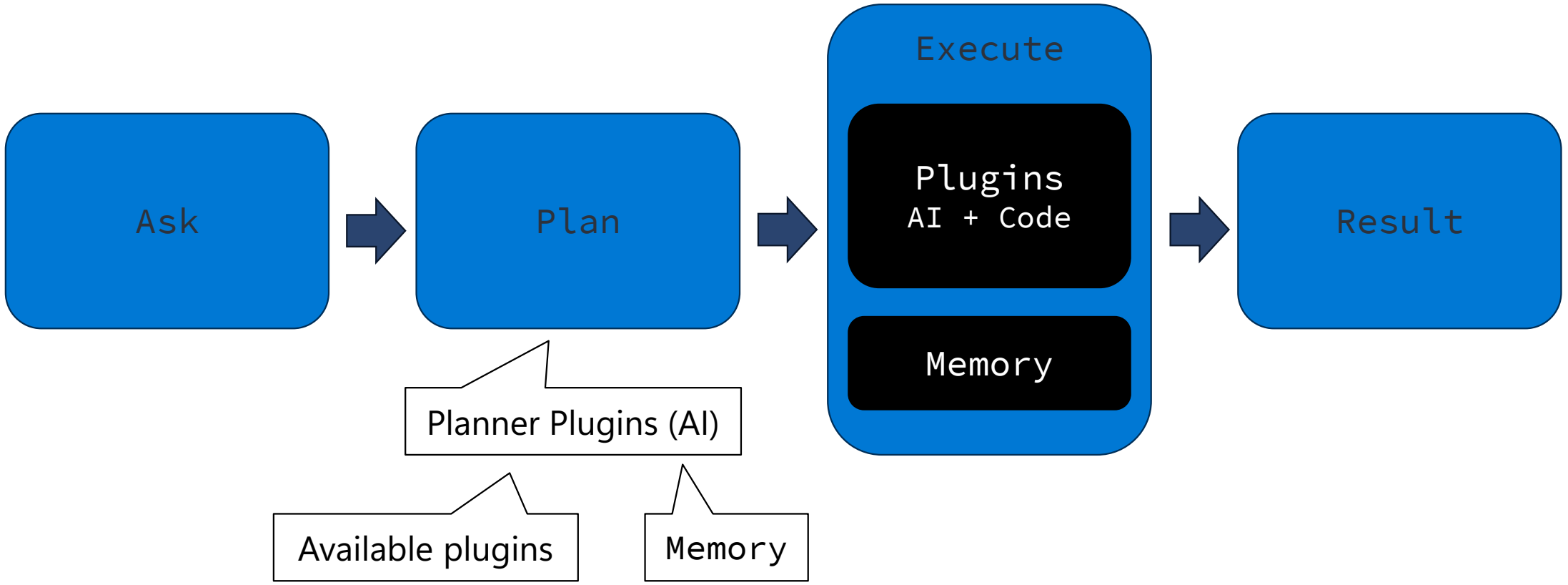

Plugins are building blocks of the Semantic Kernel.
Encapsulate AI capabilities into a single unit of functionality.
OpenAI plugin specification = plugins will be usable in ChatGPT, Bing, M365 etc.

You can invoke functions either manually (chaining functions) or automatically with a planner.

**Semantic functions** represent the ears and mouth (input and outputs) of your AI Application. They allow your AI app to listen to users asks and respond back with a natural language response.

**Native** (code) **functions** are used BEFORE / AFTER calling model REST API to give broader context

# Simplification: Compositions & pipelines in SK



# OK – LLM and „traditional" data warehouses

LLM can understand relations/structure – also DB schema!

LLM can execute task

(but task needs to be very ACCURATE, SPECIFIC and CLEAR and …)

*(when there are many tasks, the model is usually confused)*

LLM can respond in JSON / Text

**So – LLM can write query!**

# Demo – TextToSql, SqlToText etc

Z:\A\tkopacz\AzFY23-AI\OpenAI\fy24-aoi-promptToSQL-and-sqltodesc\fy24-aoi-promptToSQL-and-sqltodesc.csproj

# Comments – do not forget….

Imagine SQL Injection

- Because of prompt injection
  - Because of user few words
  - Being part of final context
  - ….

Define clear and specific task(in human natural language) is HARD

- (therefore, we HAVE **S**tructured **Q**uery **L**anguage)
  - (or any programming language in fact!)

If you look for product – check current/next copilot for PowerBI!

Description(s) for various audiences – priceless!

# Why not  application as data source?

## Data to application

- In general – Object, relationship of object
- So – for databases:
  - Relational: ORM (Entity Framework, (N)Hibernate, Dapper, LLBLGen Pro, …), Java Persistence API, JDO
  - NoSQL (Mongo, Cosmos, …) – ONDM (but – less popular) – usually JSON is nice

## Data from application … From object logic back to database

- Imagine: Complex state in network of Actors in Orleans / Service Fabric / Akka /…

## What if QUERY to Application (Data Mesh!) So:

- REST – like Graph API from M365, any custom API, easy to expose app-based dataset
- ODATA (ISO/IEC 20802) – structured (records) based data, like Dataverse, SAP, (Sharepoint)
- GraphQL (Facebook) – graph in general

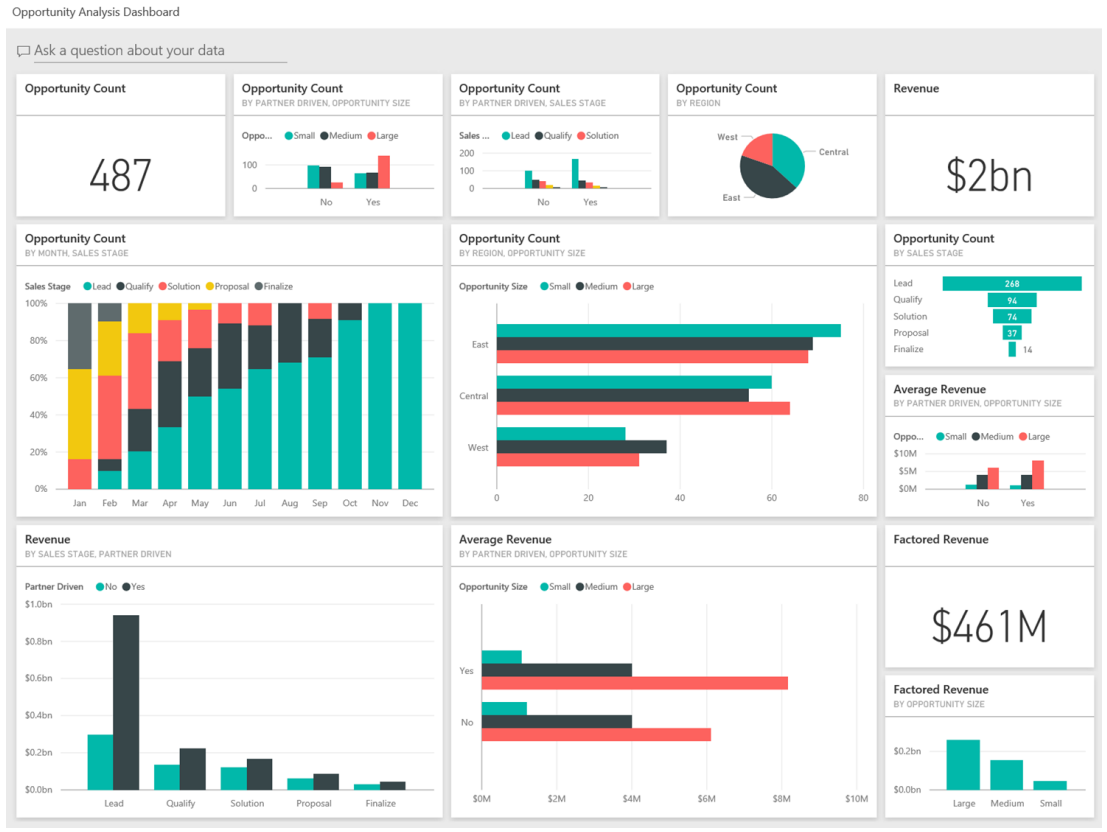
Microsoft

# Remember title:

Information not WAREHOUSE

# Many scenario

## What we are doing: Reports




## What is needed: One Pager

**One-Pager: Boosting Furniture Sales via E-Commerce**

**Date:** April 16, 2024

**Prepared by:** Marketing and Sales Department

Executive Summary

- **Objective:** Increase furniture sales by expanding our e-commerce presence in response to stagnant physical store sales and growing online market demand.

Situation Overview

- **Current Issue:** Physical store sales are stagnant; however, online sales in the industry have grown by 20% annually.
- **Opportunity:** Expanding our e-commerce capabilities to capture this growing market segment.

Recommendation

- **Action:** Develop and launch a comprehensive e-commerce platform.
- **Key Steps:**
  1. Partner with a tech firm for platform development.
  2. Implement a targeted digital marketing campaign.
- **Investment:** Estimated initial cost of $500,000.

Expected Outcomes

- **Benefits:** Anticipate a 20% increase in total sales within the first year.
- **Risk Management:** Begin with a pilot in select regions, adjusting based on performance and feedback.

Conclusion

- Embracing e-commerce will align us with consumer trends and drive growth in a competitive market.

SymphonyAI

CAPITAL MARKETS

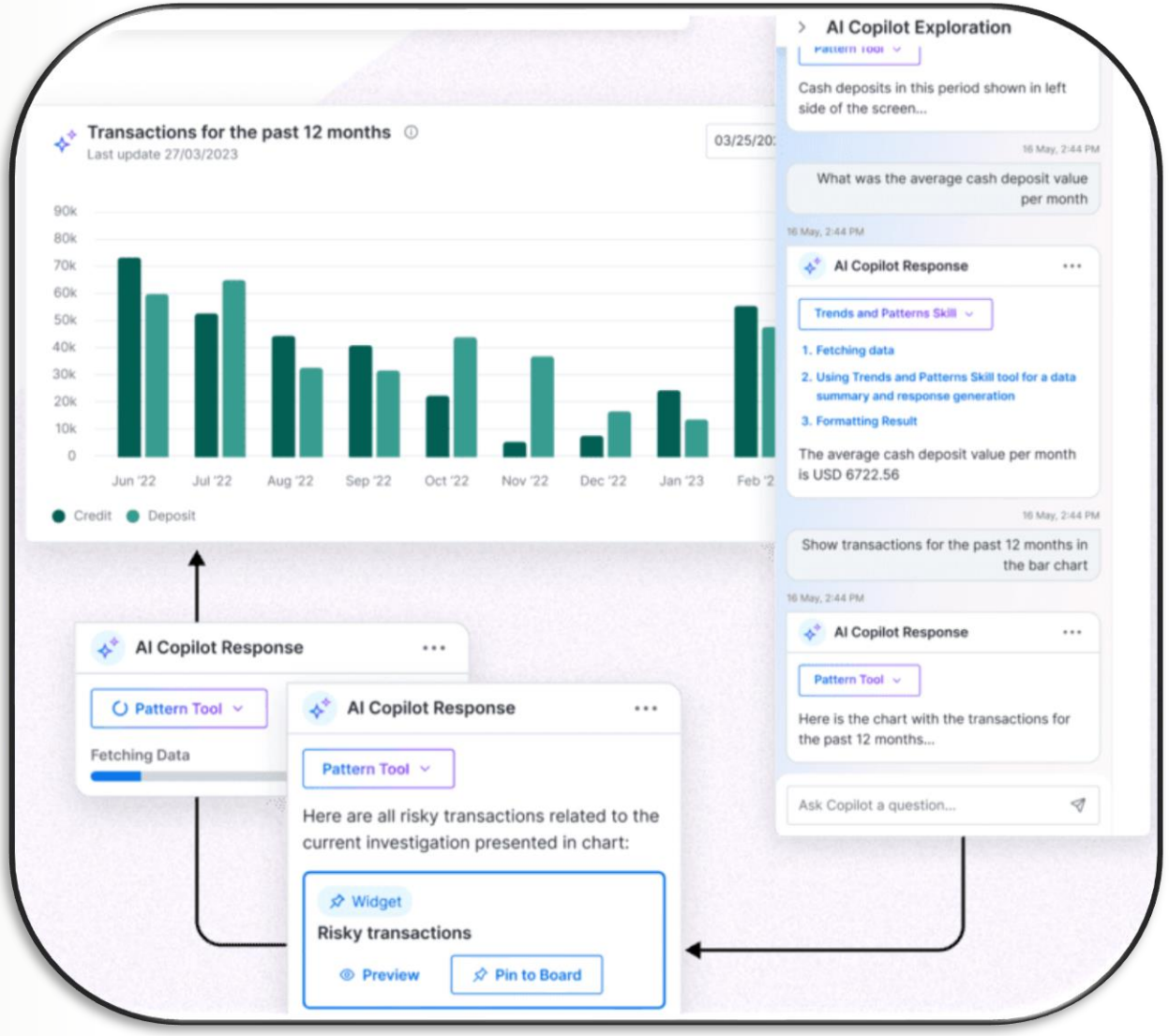
# SymphonyAI

**Financial Crime investigation solution**

By addressing challenges at the core of your business, SymphonyAI's tailored predictive and generative AI applications deliver rapid, real, relevant results to transform how the world works.

SymphonyAI's generative AI applications empower users with deep insights and productivity boosts to enhance workflows and decision making. The architecture combines generative and predictive AI, offering quick, accurate responses while maintaining data privacy and access through a natural language interface.

Forecast what's next for your business with predictive AI tuned for your industry. SymphonyAI applications use the right supervised, semi-supervised, or unsupervised machine learning tuned and optimized to address specific industry use cases.




Accelerate financial crime investigations by up to 70% and instantly generate summary reports.

[Enterprise AI SaaS for the future of business | SymphonyAI]

# Capabilities of LLM to process different data

(0)
- (sales/project/edu) meeting notes to CRM/DevOps/Jira

1
- Many documents, short summary

2
- Document to record(s)

3
- Records (tables) to description

4
- Images (graphs/diagrams) to records

5
- Changes in document/comparison

# Demo – Semantic Kernel and document processing (few scenarios)

Z:\A\tkopacz\AzFY23-AI\OpenAI\fy24-AOI-MiscDocumentProcessingScenarios\fy24-AOI-MiscDocumentProcessingScenarios.sln

Z:\A\tkopacz\AzFY23-AI\OpenAI\fy24-gpt4-vision (python)

# Forms vs Bot vs “discussion”

How to fill form without filling the form?

Why? Form = record = DB Structure = ….

Statement: Users HATES forms | bots | Call Center  Scripts …

Maybe – they will “talk”

LLM likes “ONE” task at the time

But – we can change “system prompt” during conversation

(later – Agents – Agent1: question + Agent2: EVALUATION!)

# Demo – Semantic Kernel and questions

Z:\A\tkopacz\AzFY23-AI\OpenAI\fy24-chatToGetAnswers\fy24-chatToGetAnswers.sln

# Challenge – on how to extract „action" from reviews

(or from comments online)

(or from posts in Teams)

(or from any discussion about topic)

(or categorize transaction, complains, messages)

Goal – avoid building apps around question – *rate us from 1 to 5*....

and get **clear** guidance on what to fix.

and get single category

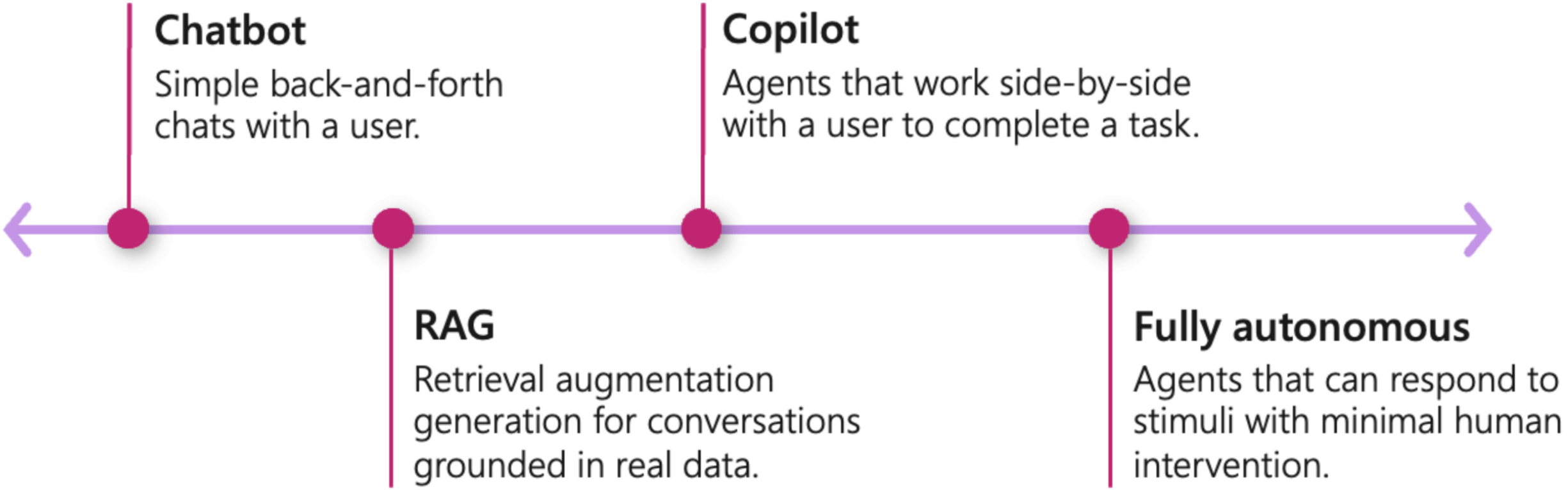
and...

# Demo – Semantic Kernel and transactions analysis and categorization

Z:\A\tkopacz\AzFY23-AI\OpenAI\fy24-SKTextAndCategorizationAndAction\fy24-SKTextAndCategorizationAndAction.sln

# Planners and Function and Agents and...

# „Evolution" of AOI apps (usage) – what is an agent?



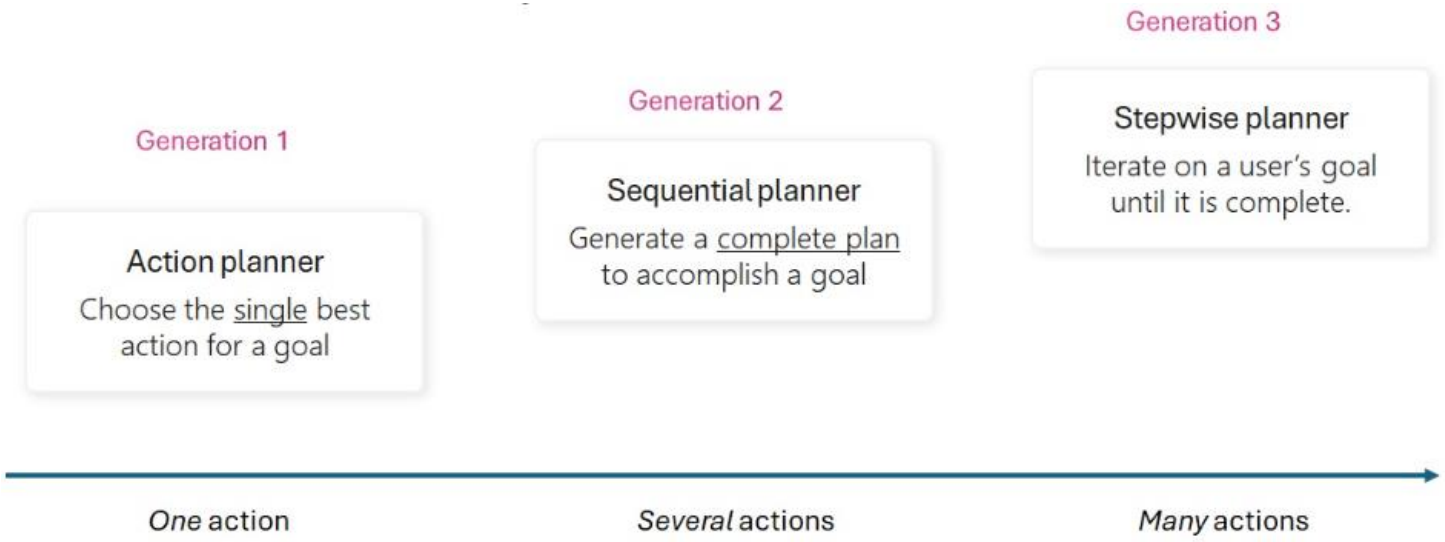

***https://github.com/microsoft/semantic-kernel/tree/main/dotnet/src/Experimental/Agents***

# Demo – SK + (manual) custom orchestration

Z:\A\tkopacz\AzFY23-AI\OpenAI\fy24-20240101SKV1Demos\TK101SemanticKernel\fy24-101SemanticKernel.csproj
Demo05b - TYLKO

# Planners? Azure OpenAi Function calling?

Function that takes a user's ask and returns a plan on how to accomplish the request.
It does so by using AI to mix-and-match the plugins registered in the kernel so that it can recombine them into a series of steps that complete a goal.




Concepts; pre v.1

V1: 2024

1. Handlebars
*(preview as may 2024 but EXTREAMLY POWERFULL)*

2. Function calling stepwise planner
*(preview, but ok, simple).*

# Handlebars - example

```
## Start
Now take a deep breath and accomplish the task:
1. Keep the template short and sweet. Be as efficien
2. Do not make up helpers or functions that were not
3. If you can't fully accomplish the goal with the a
4. Always start by identifying any important values
5. The template should use the \{{json}} helper at l
6. Don't forget to use the tips and tricks otherwise
7. Don't close the ``` handlebars block until you're
```

```
{{#each functions}}
### `{{doubleOpen}}{{PluginName}}{{../nameDelimiter}}{{Name}}{{doubleClose}}`
Description: {{Description}}
Inputs:
  {{#each Parameters}}
    - {{Name}}:
    {{~#if ParameterType}} {{ParameterType.Name}} -
    {{~else}}
        {{~#if Schema}} {{getSchemaTypeName this}} -{{/if}}
    {{~/if}}
    {{~#if Description}} {{Description}}{{/if}}
    {{~#if IsRequired}} (required){{else}} (optional){{/if}}
  {{/each}}
Output:
{{~#if ReturnParameter}}
  {{~#if ReturnParameter.ParameterType}} {{ReturnParameter.ParameterType.Name}}
  {{~else}}
    {{~#if ReturnParameter.Schema}} {{getSchemaReturnTypeName ReturnParameter}}
    {{else}} string{{/if}}
  {{~/if}}
  {{~#if ReturnParameter.Description}} - {{ReturnParameter.Description}}{{/if}}
{{/if}}
{{/each}}
```

```
[AVAILABLE FUNCTIONS]

### `{{MathPlugin-Add}}`
Description: Add two numbers
Inputs:
  - number1 double - The first number to add (required)
  - number2 double - The second number to add (required)
Output: double

### `{{MathPlugin.Divide}}`
Description: Divide two numbers
Inputs:
  - number1: double - The first number to divide from (required)
  - number2: double - The second number to divide by (required)
Output: double
```

```
Plugins.MathSolver: Information: Plan: {{!-- Step 1: Set the initial investment amount --}}
{{set "initialInvestment" 2130.23}}

{{!-- Step 2: Calculate the increase percentage --}}
{{set "increasePercentage" 0.23}}

{{!-- Step 3: Calculate the final amount after the increase --}}
{{set "finalAmount" (MathPlugin-Multiply (get "initialInvestment") (MathPlugin-Add 1 (get "incr

{{!-- Step 4: Output the final amount --}}
{{json (get "finalAmount")}}
```

# Demo – SK + “auto” function calling + planners

Z:\A\tkopacz\AzFY23-AI\OpenAI\fy24-20240101SKV1Demos\TK101SemanticKernel\fy24-101SemanticKernel.csproj
Demo6,7- TYLKO

Z:\A\tkopacz\AzFY23-AI\OpenAI\fy24-SKTextAndCategorizationAndAction\fy24-SKTextAndCategorizationAndAction.sln – await tKDemo01.Demo02()

# Copilot and Plugins

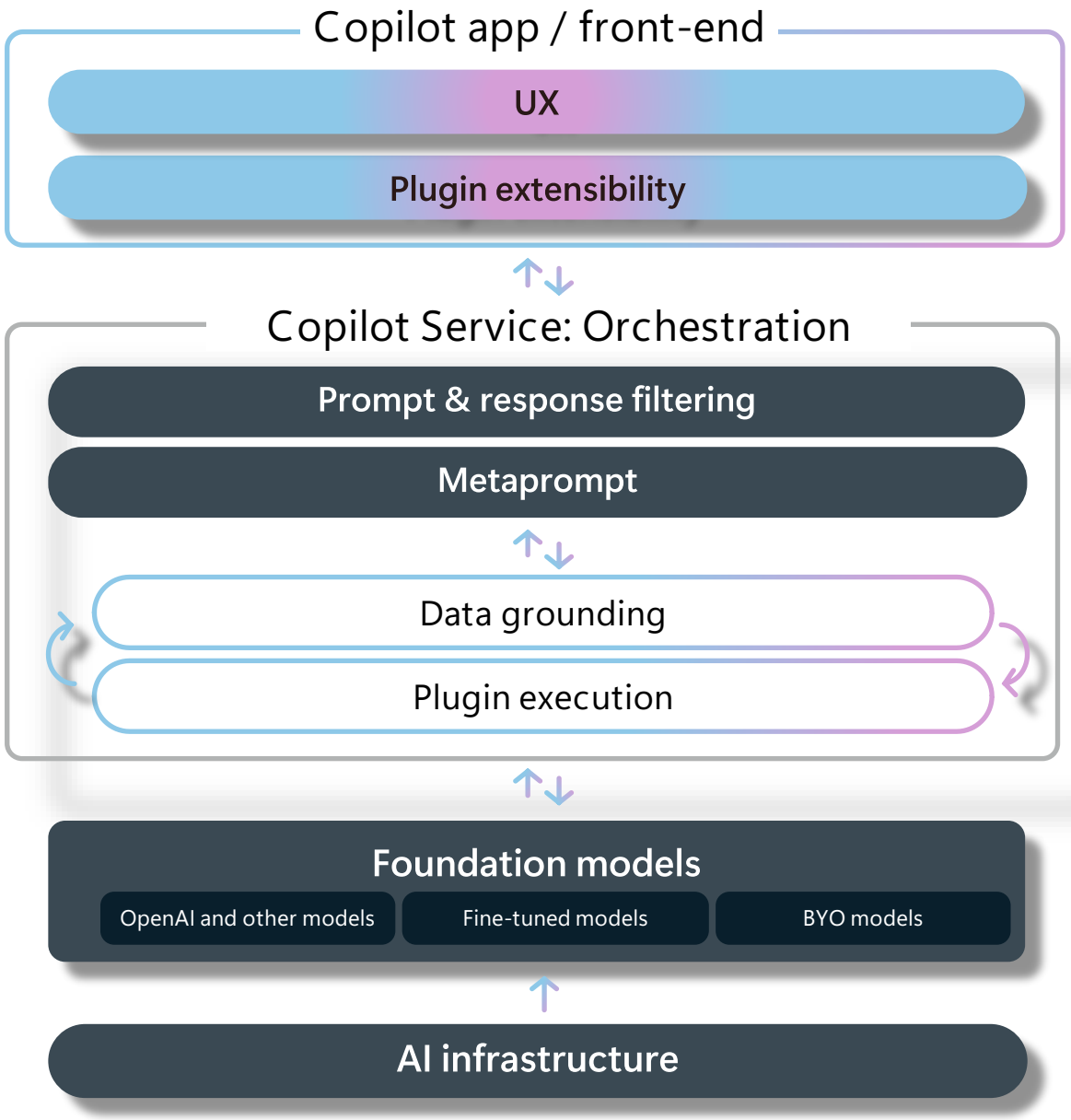
# What is in the Copilot stack?

## Copilot app

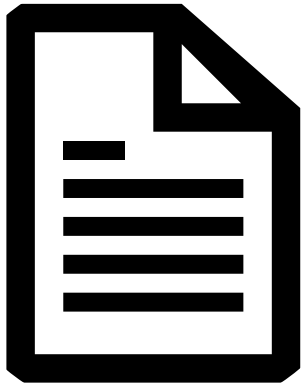
- Interface for user interaction – often chat-centric
- Extensible through plugins

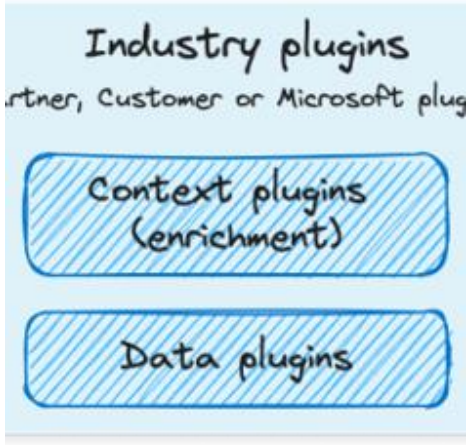
## Copilot service: orchestration

- LLM is the core 'operating system' driving the Copilot
- Orchestration is responsible for routing, data grounding and executing code and LLM-based plugins
- Note: a copilot service can be invoked as a plugin by another copilot!



# Three essential copilot building blocks







## RAG Store

The primary knowledge store to ground the large language model, usually comprised of unstructured data / documents to start

## Plugins

Expand the reach of the chat conversation by interacting with data from an array of external sources.

Short: OUR OWN API

## Orchestrator

Orchestrate the conversation across an array of plugins and send instructions via APIs to complete actions

# Digression: Developers programs (as per May 2024!)

## Public Preview :

Plugins for M365 (Microsoft 365 Chat and Microsoft Teams)

Plugins for Microsoft Copilot (Bing, Microsoft Edge)

Plugins are available in Public Preview to users of Copilot in Bing (desktop, mobile), and to users of Microsoft Edge (desktop, mobile).

## Private Preview - Plugins for Microsoft Copilot (Private Preview)

Plugin sideloading is only available to select users.
See Test and debug a plugin (Private Preview).

Publishing plugins is only available to select users, for initial feedback.
See Publish a plugin (Private Preview).

# OpenAI Plugins

Common for: (ChatGPT), M365 Copilot, Bing, ….

Goal: way to provide additional grounding / context for LLM context

2023: https://devblogs.microsoft.com/semantic-kernel/skills-to-plugins-fully-embracing-the-openai-plugin-spec-in-semantic-kernel/

Spec: https://platform.openai.com/docs/actions/introduction

Short: Manifest + OpenAPI (Swagger) + import JSON + REST API

# Demo – SK + OpenAI plugin (format)

*(part – pre 1.0 SK)*

Z:\A\tkopacz\AzFY23-AI\OpenAI\fy24-CreateChatGptPlugin1\sk-csharp-chatgpt-plugin\azure-function\sk-chatgpt-azure-function.sln

# RAG? And “information retrieval”

# LLM and “token window” and “speed”

LLM are slow (lack of engineering capabilities!)

Read only = REST Call are independent

Maximum size of “window” = “single” rest calls

Token: Assume 1000 tokens is equivalent to 750 characters (PL – less)

Examples:

GPT 2: (different, pad right; ca 1K) | GPT 3, GPT 3.5: 4K

GPT 3.5 Turbo: Input: 16K, Output: 4K

GPT 4-32K: 32K | GPT 4 Turbo, GPT 4-v: Input: 128K, Output: 4K

Mistral: 32000

LLAMA3: 2K

…

Embedding text-embedding-ada-002, text-embedding-3-large, text-embedding-3-small: 8191

# Tasks and challenges

Divide large documents (part ca 6000 characters)

Generate embedding (delays!)

Find IMPORTANT/RELEVANT documents for context

# Find: Azure Search – searching for information

## Filtering and plain text

Raw context, verbatim from source document, supporting filters/pattern matching

## Full-text search

Retrieval that matches on plain text stored in an index; keyword search; simple/full Lucene, Office dictionaries. Text is analyzed and tokenized

## Vector Search

Search based on vector indexes (embedding)

## Hybrid search **<- start with THAT**

Combination of Full Text and Vector search. Useful, because user input is not always “task” or sentence – we are looking for KEYWORDS

## Retrieval Augmented Generation (RAG)

## +SKILLS!

# Remarks – skills in Azure Search

BUILT-IN SKILLS

+ CUSTOM!

**Key Phrase extraction**

Organization entity extraction

Face detection

Text Utilities

**Location entity extraction**

Persons entity extraction

Celebrity recognition

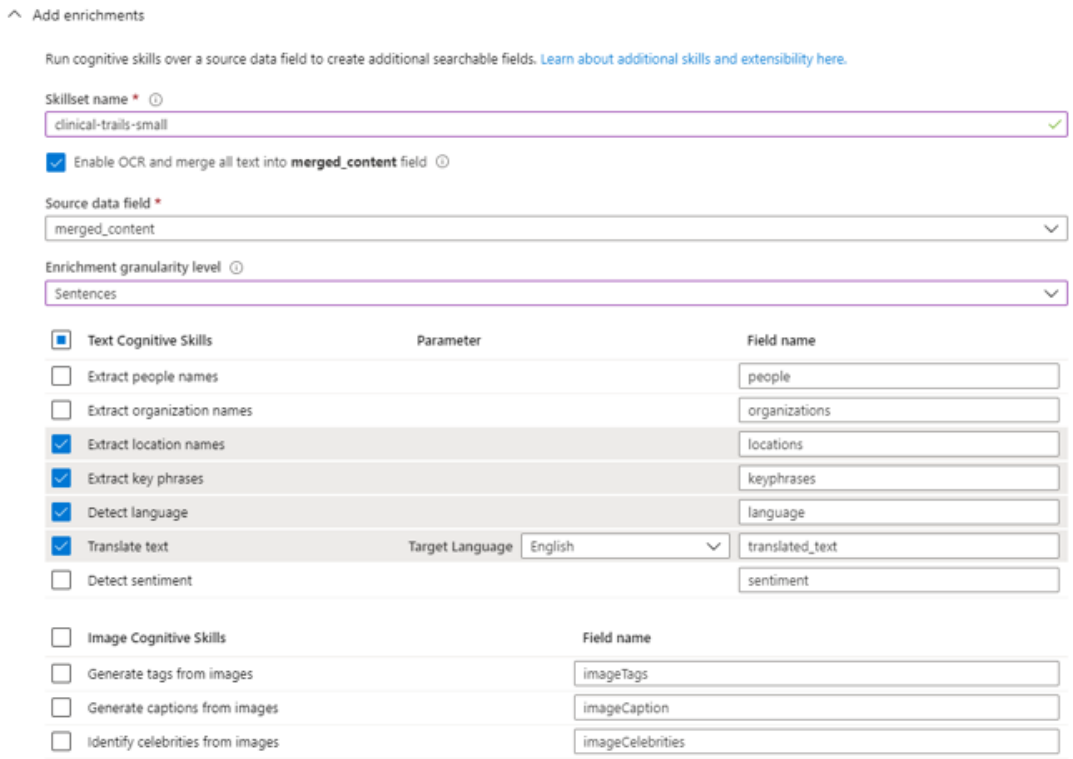
Landmark detection

**Sentiment analysis**

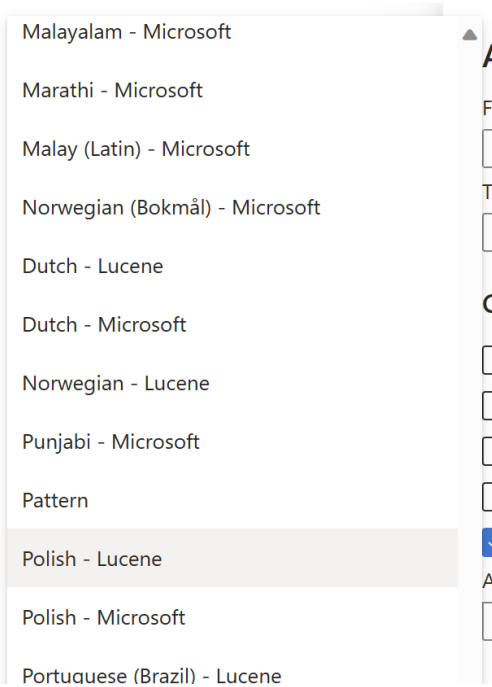
Language detection

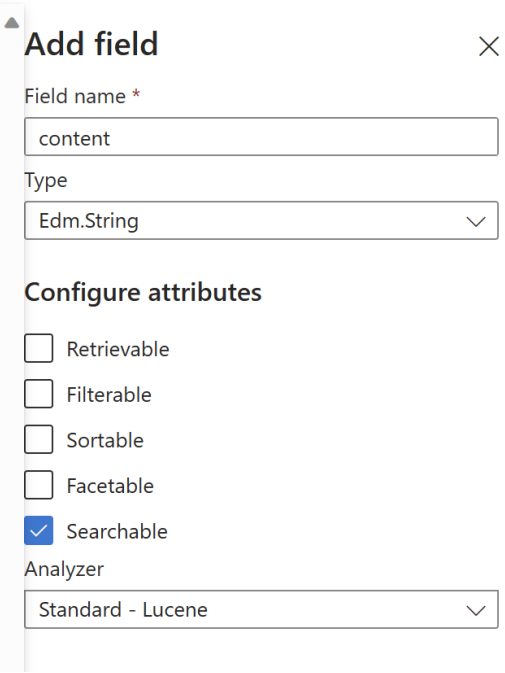
Image tag extraction

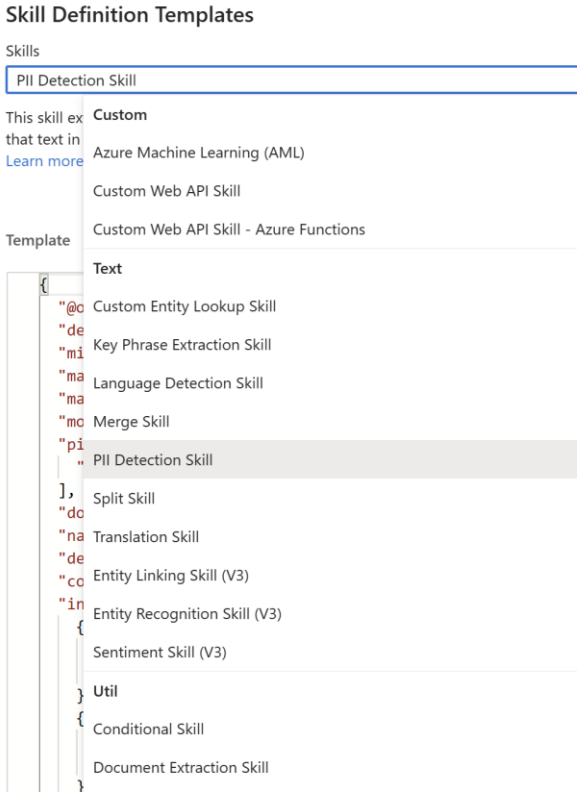
Printed text recognition

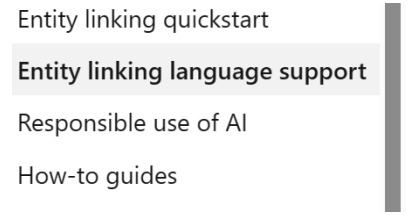


# Polish?

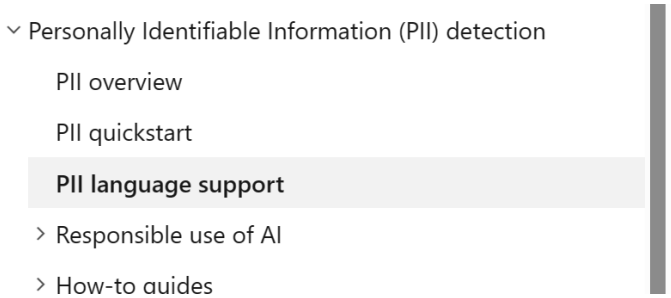


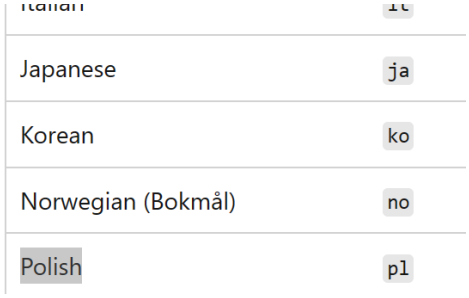




| Language | Language code |
|---|---|
| English | en |
| Spanish | es |

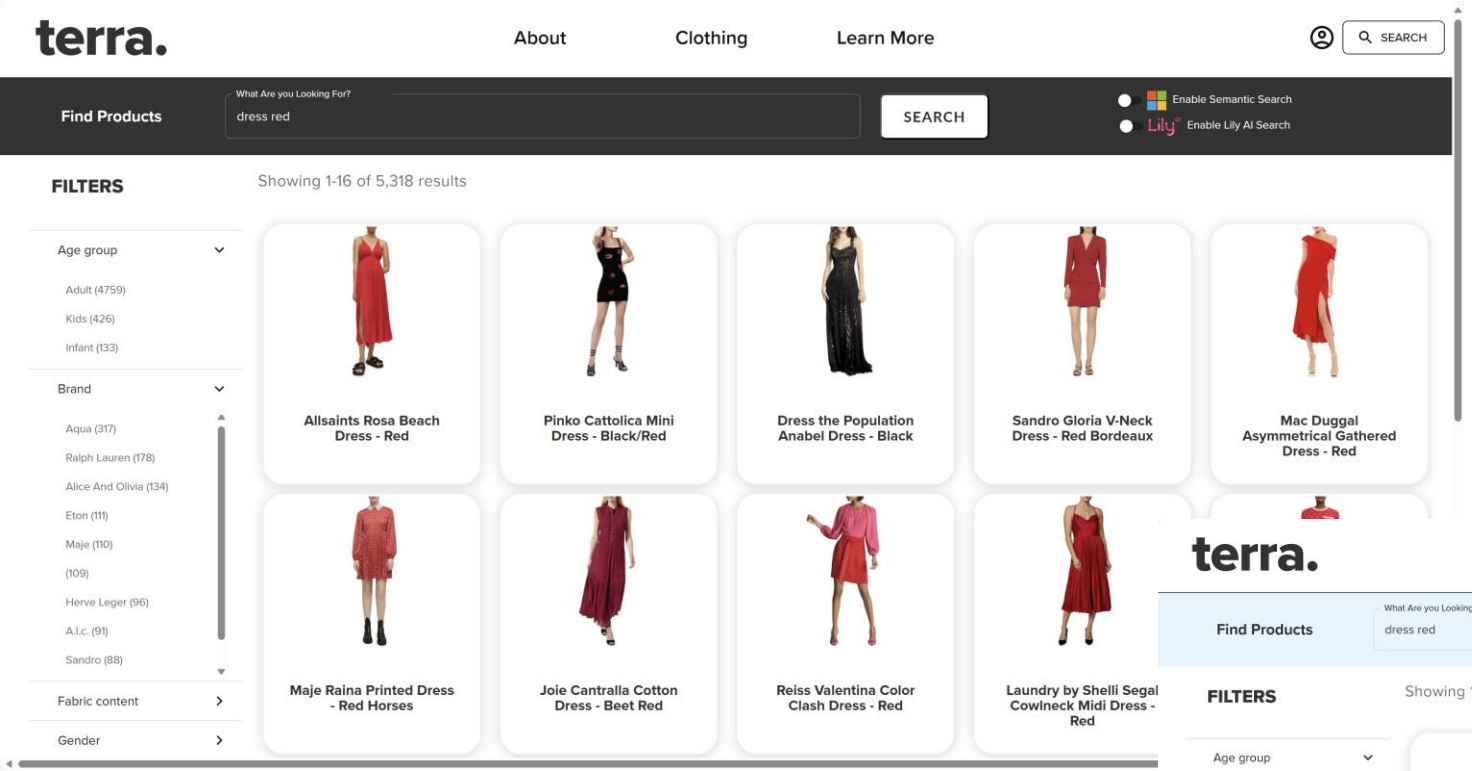


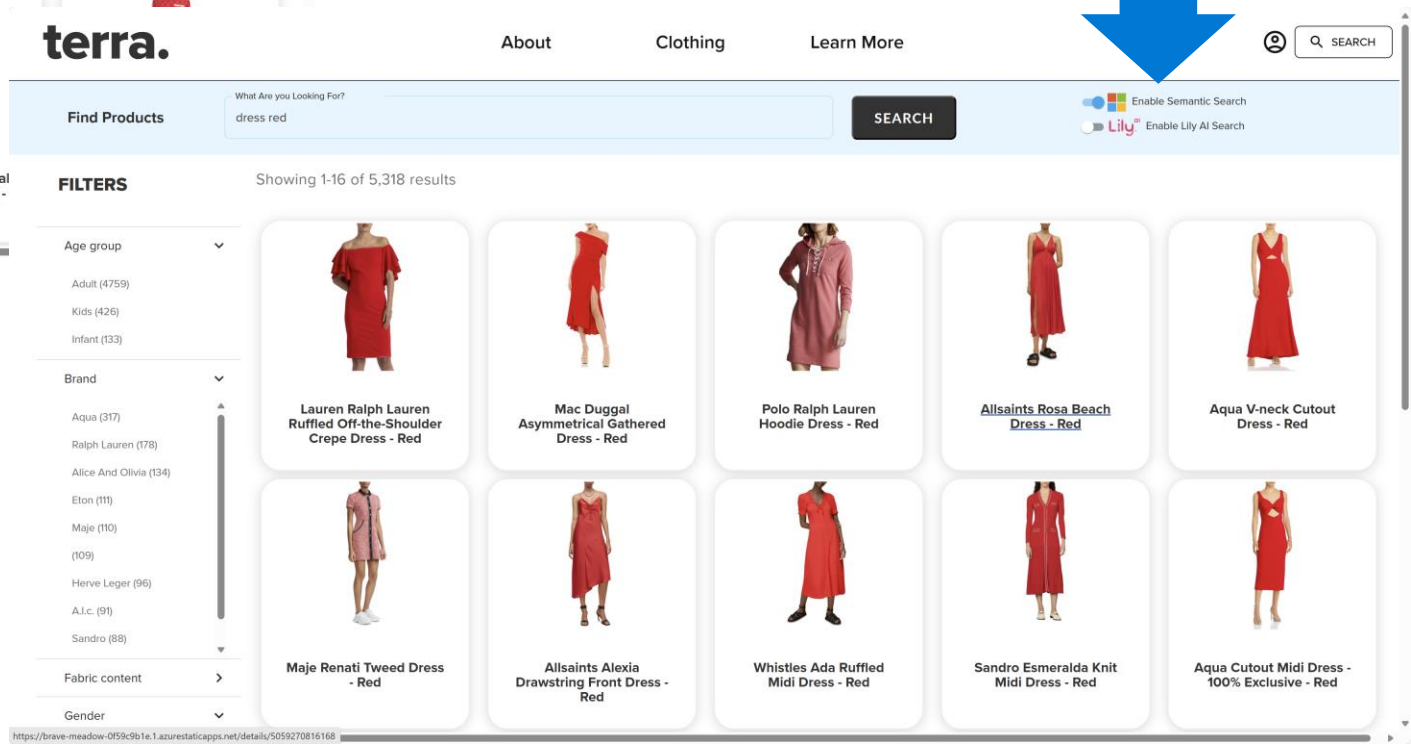





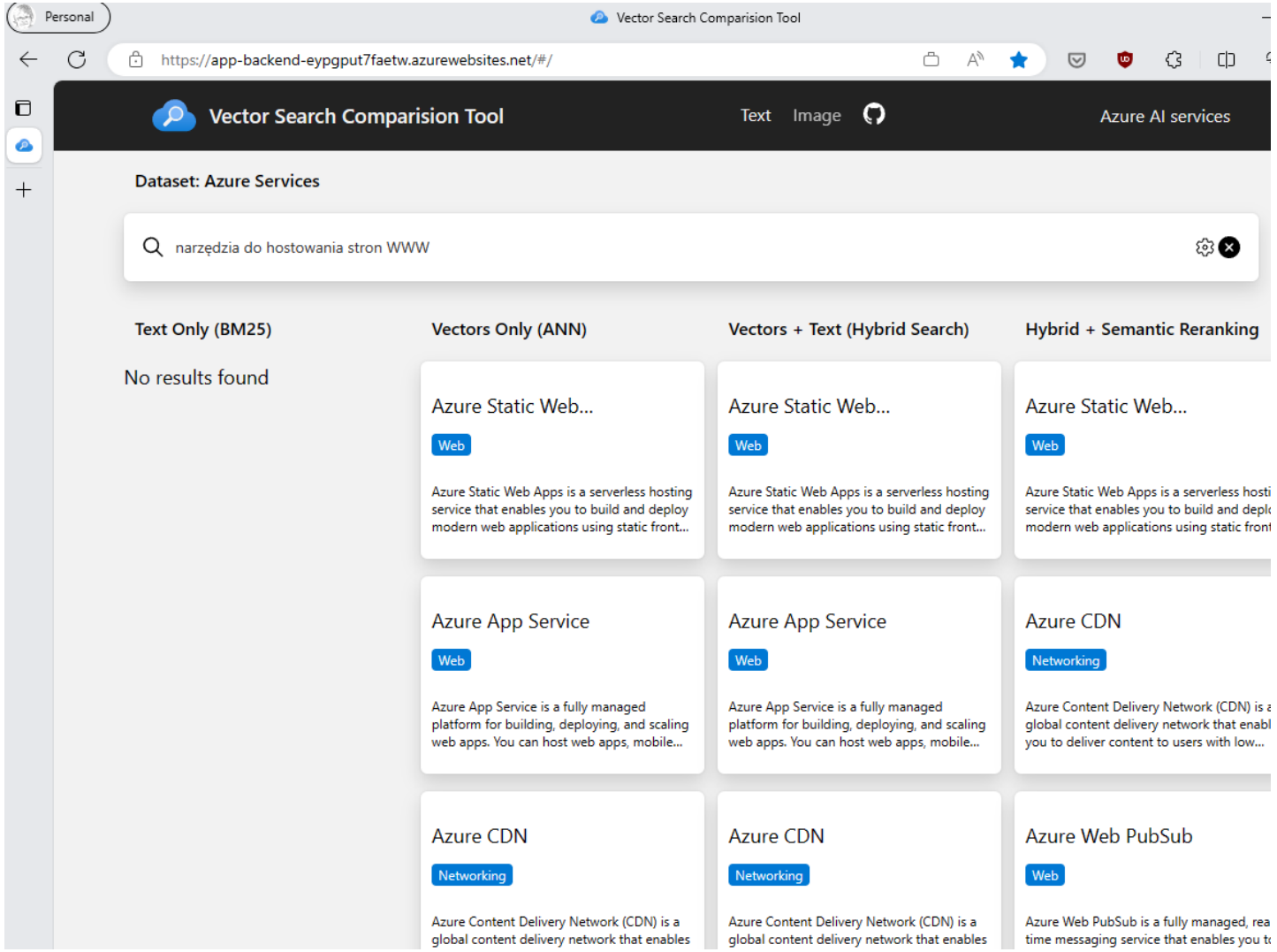

Language support for ....

# https://brave-meadow-0f59c9b1e.1.azurestaticapps.net/search?q=red dress

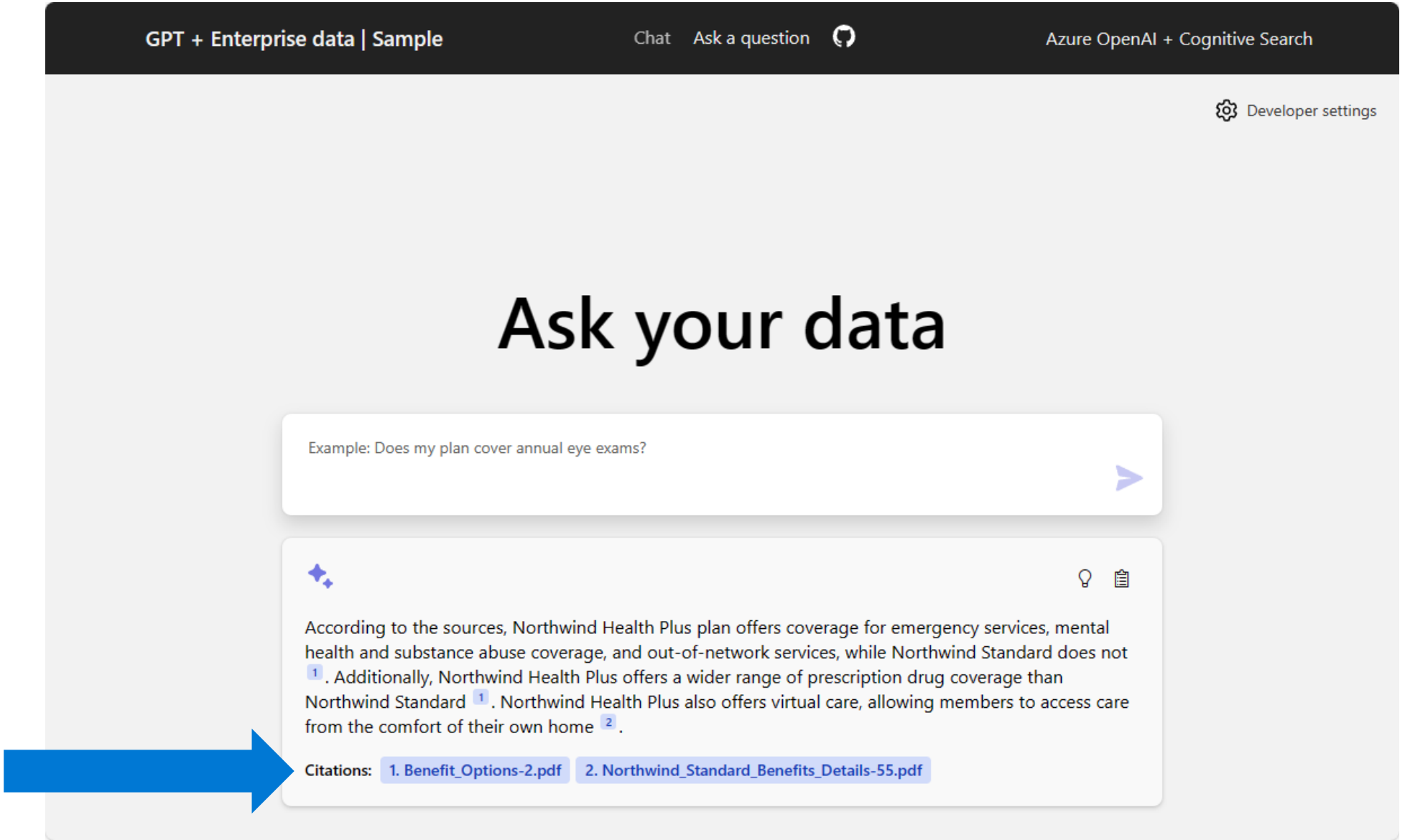


# Vector Search Comparision Tool



# Wait… so what it will bring to LLM product?

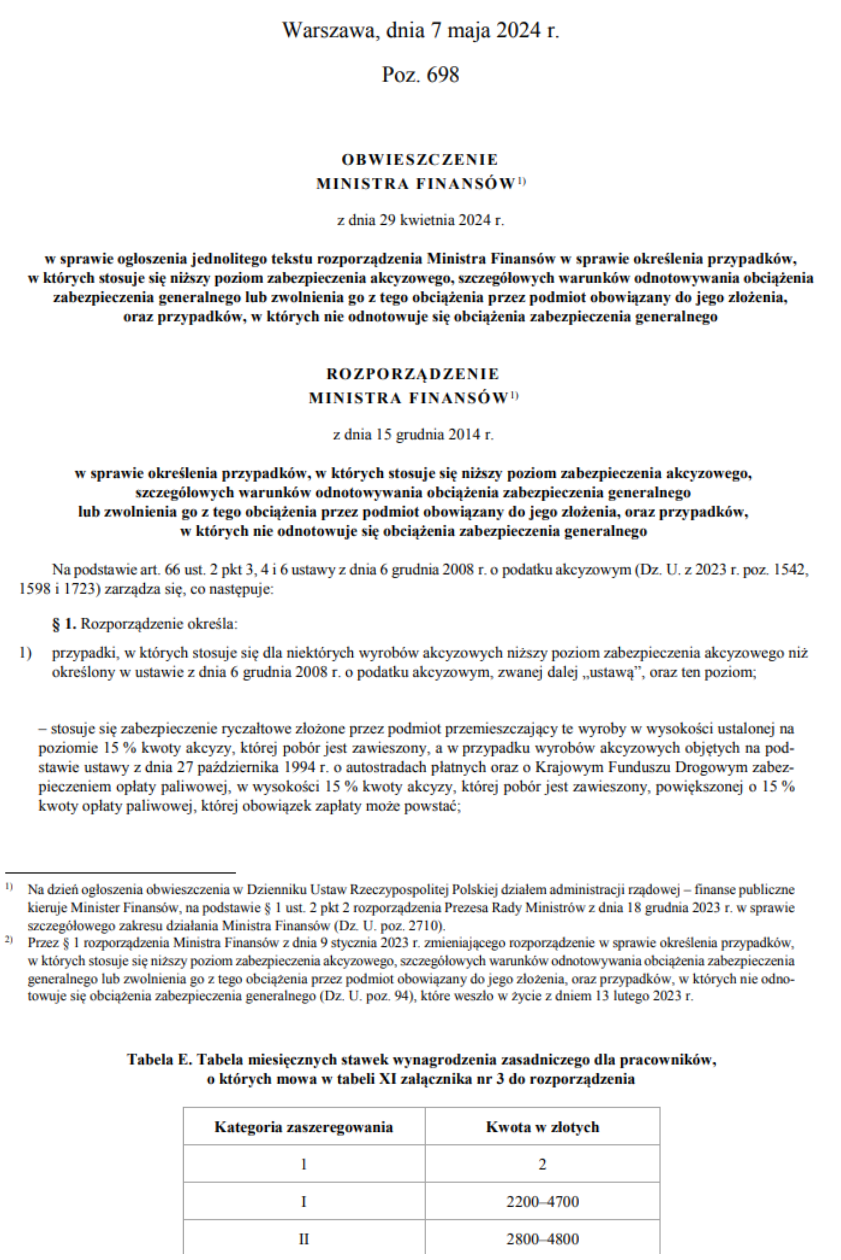


# Divide documents...

1. Overlap (simple, and effective)

   9000 = 4000 + 4000 + 4000
2. Repeat some parts (legal, agreements etc)
3. Generate embedding for summaries
4. Generate embedding for summaries AND for overlapping pieces

   Strategy – find document and then piece IN document
5. Semi-manual with tests
6. PDF: OCR + Layout detection



Warszawa, dnia 7 maja 2024 r.

Poz. 698

**OBWIESZCZENIE**
**MINISTRA FINANSÓW**[1)]

z dnia 29 kwietnia 2024 r.

**w sprawie ogłoszenia jednolitego tekstu rozporządzenia Ministra Finansów w sprawie określenia przypadków, w których stosuje się niższy poziom zabezpieczenia akcyzowego, szczegółowych warunków odnotowywania obciążenia zabezpieczenia generalnego lub zwolnienia go z tego obciążenia przez podmiot obowiązany do jego złożenia, oraz przypadków, w których nie odnotowuje się obciążenia zabezpieczenia generalnego**

**ROZPORZĄDZENIE**
**MINISTRA FINANSÓW**[1)]

z dnia 15 grudnia 2014 r.

**w sprawie określenia przypadków, w których stosuje się niższy poziom zabezpieczenia akcyzowego, szczegółowych warunków odnotowywania obciążenia zabezpieczenia generalnego lub zwolnienia go z tego obciążenia przez podmiot obowiązany do jego złożenia, oraz przypadków, w których nie odnotowuje się obciążenia zabezpieczenia generalnego**

Na podstawie art. 66 ust. 2 pkt 3, 4 i 6 ustawy z dnia 6 grudnia 2008 r. o podatku akcyzowym (Dz. U. z 2023 r. poz. 1542, 1598 i 1723) zarządza się, co następuje:

**§ 1.** Rozporządzenie określa:

1) przypadki, w których stosuje się dla niektórych wyrobów akcyzowych niższy poziom zabezpieczenia akcyzowego niż określony w ustawie z dnia 6 grudnia 2008 r. o podatku akcyzowym, zwanej dalej „ustawą", oraz ten poziom;

– stosuje się zabezpieczenie ryczałtowe złożone przez podmiot przemieszczający te wyroby w wysokości ustalonej na poziomie 15 % kwoty akcyzy, której pobór jest zawieszony, a w przypadku wyrobów akcyzowych objętych na podstawie ustawy z dnia 27 października 1994 r. o autostradach płatnych oraz o Krajowym Funduszu Drogowym zabezpieczeniem opłaty paliwowej, w wysokości 15 % kwoty akcyzy, której pobór jest zawieszony, powiększonej o 15 % kwoty opłaty paliwowej, której obowiązek zapłaty może powstać;

[1)] Na dzień ogłoszenia obwieszczenia w Dzienniku Ustaw Rzeczypospolitej Polskiej działem administracji rządowej – finanse publiczne kieruje Minister Finansów, na podstawie § 1 ust. 2 pkt 2 rozporządzenia Prezesa Rady Ministrów z dnia 18 grudnia 2023 r. w sprawie szczegółowego zakresu działania Ministra Finansów (Dz. U. poz. 2710).

[2)] Przez § 1 rozporządzenia Ministra Finansów z dnia 9 stycznia 2023 r. zmieniającego rozporządzenie w sprawie określenia przypadków, w których stosuje się niższy poziom zabezpieczenia akcyzowego, szczegółowych warunków odnotowywania obciążenia zabezpieczenia generalnego lub zwolnienia go z tego obciążenia przez podmiot obowiązany do jego złożenia, oraz przypadków, w których nie odnotowuje się obciążenia zabezpieczenia generalnego (Dz. U. poz. 94), które weszło w życie z dniem 13 lutego 2023 r.

**Tabela E. Tabela miesięcznych stawek wynagrodzenia zasadniczego dla pracowników, o których mowa w tabeli XI załącznika nr 3 do rozporządzenia**

| Kategoria zaszeregowania | Kwota w złotych |
|---|---|
| 1 | 2 |
| I | 2200–4700 |
| II | 2800–4800 |

# Read

**API version:** 2023-07-31 (3.1 General Availability) **Service resource:** pltkfy23ext-cs-form

Drag & drop file here or Browse for files or Fetch from URL

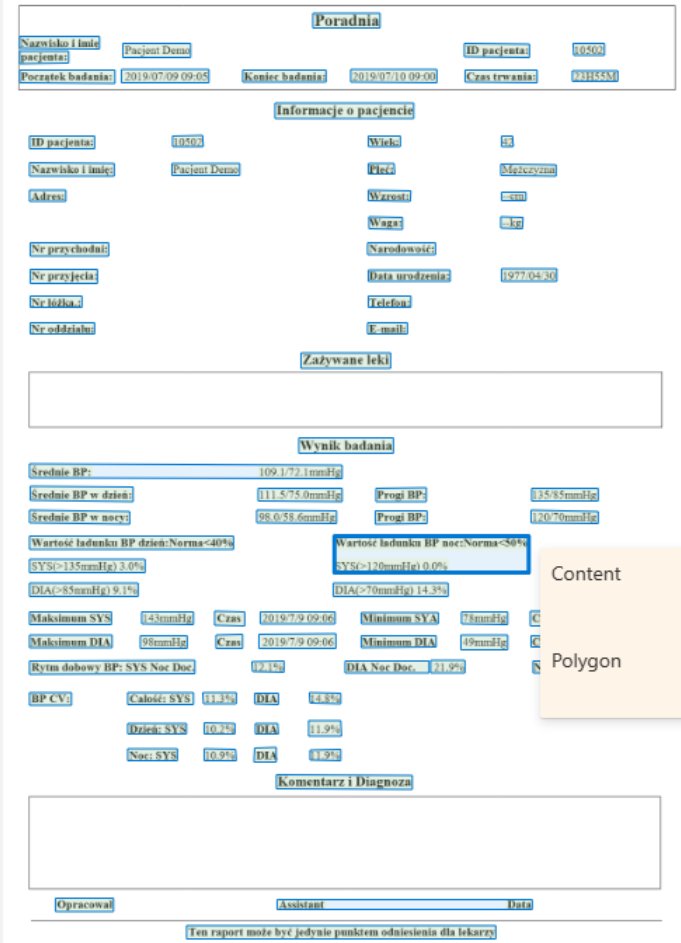
ABPM50-prz... ia.pdf

pdf-92740-7...70.pdf

Ulotka-2-20... 12.pdf

Sample

Run analysis | Analyze options




Content: Wartość ładunku BP noc:Norma<50% SYS(>120mmHg) 0.0%

Polygon: 3.934, 6.568, 6.2487, 6.568, 6.2487, 6.9943, 3.934, 6.9943

1 of 5

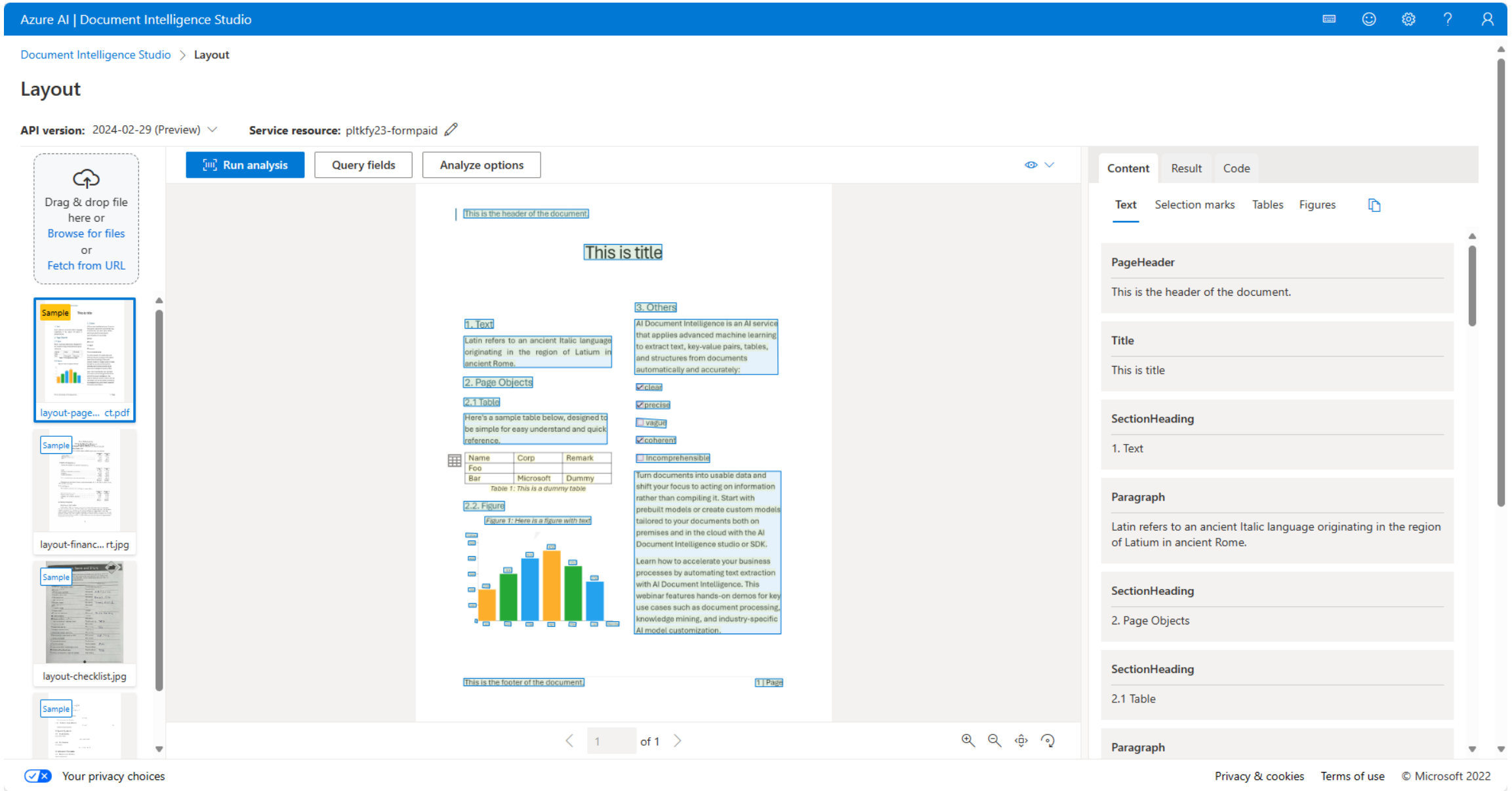
Content | Result | Code

JSON


```
34307            "length": 35
34308          }
34309        ],
34310        "boundingRegions": [
34311          {
34312            "pageNumber": 1,
34313            "polygon": [
34314              0.3147,
34315              6.5629,
34316              2.736,
34317              6.5629,
34318              2.736,
34319              6.7202,
34320              0.3147,
34321              6.7202
34322            ]
34323          }
34324        ],
34325        "content": "Wartość ładunku BP dzień: Norma<4
34326      },
34327      {
34328        "spans": [
34329          {
34330            "offset": 597,
34331            "length": 18
34332          }
34333        ],
34334        "boundingRegions": [
34335          {
34336            "pageNumber": 1,
34337            "polygon": [
34338              0.3147,
34339              6.837,
34340              1.6853,
34341              6.837,
34342              1.6853,
34343              6.9994,
```

# But first – extract layout



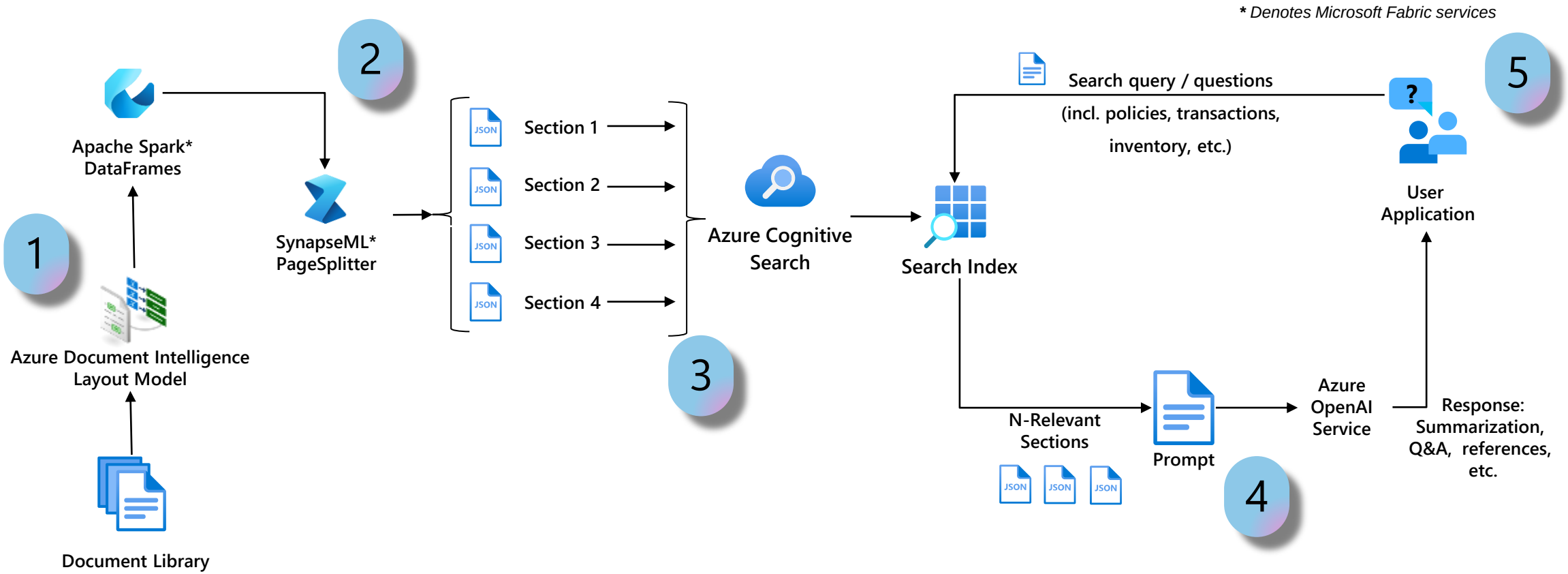
# Ready to use OCR + structure recognizer

## Pretrained document-analysis models

| Document type | Example | Data to extract | Your best solution |
|---|---|---|---|
| **A generic document.** | A contract or letter. | You want to primarily extract written or printed text lines, words, locations, and detected languages. | Read OCR model |
| **A document that includes structural information.** | A report or study. | In addition to written or printed text, you need to extract structural information like tables, selection marks, paragraphs, titles, headings, and subheadings. | Layout analysis model |
| **A structured or semi-structured document that includes content formatted as fields and values.** | A form or document that is a standardized format commonly used in your business or industry like a credit application or survey. | You want to extract fields and values including ones not covered by the scenario-specific prebuilt models **without having to train a custom model.** | General document model |

+ Custom Models with Layouts/Extraction rules

# Fabric is excellent tool for RAG!




## 1. Data Ingestion

Assemble a library of business documents

## 2. Chunking

Employ Fabric's token management logic

## 3. Indexing

Generate embeddings and store vectors to setup semantic search

## 4. Prompting

Tools, techniques and strategies of prompting

## 5. User Interface

Surface data via APIs, plugins, or copilots

https://blog.fabric.microsoft.com/en-us/blog/unleashing-the-power-of-synapseml-and-microsoft-fabric-a-guide-to-qa-on-pdf-documents-2/

```
# Import required libraries from SynapseML
from synapse.ml.featurize.text import PageSplitter

ps = (PageSplitter()
    .setInputCol("output_content")
    .setMaximumPageLength(4000)
    .setMinimumPageLength(3000)
    .setOutputCol("chunks"))

splitted_df = ps.transform(analyzed_df)
```

```
# Import required libraries from SynapseML
from synapse.ml.featurize.text import PageSplitter
from synapse.ml.cognitive import OpenAIEmbedding

embedding = (
    OpenAIEmbedding()
    .setSubscriptionKey(aoai_key)
    .setDeploymentName(aoai_deployment_name_embeddings)
    .setCustomServiceName(aoai_service_name)
    .setTextCol("chunk")
    .setErrorCol("error")
    .setOutputCol("embeddings")
)

df_embeddings = embedding.transform(exploded_df)
```

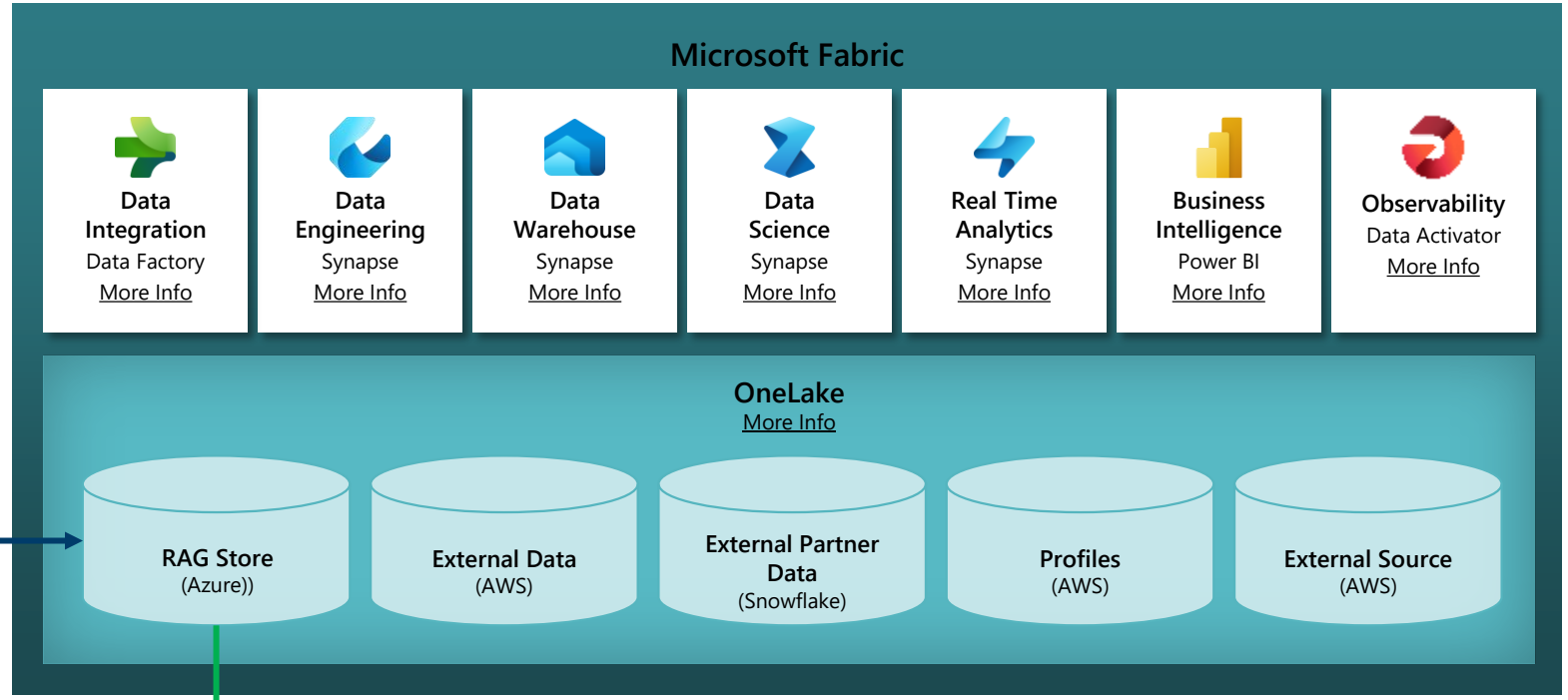
```
EMBEDDING_LENGTH = 1536 # length of the embedding vector (OpenAI generates embedding

# Create Index for Cognitive Search with fields as id, content, and contentVector

url = f"https://{cogsearch_name}.search.windows.net/indexes/{cogsearch_index_name}?a
payload = json.dumps(
    {
        "name": cogsearch_index_name,
        "fields": [
            {"name": "id", "type": "Edm.String", "key": True, "filterable": True},
            {
                "name": "content",
                "type": "Edm.String",
                "searchable": True,
                "retrievable": True,
            },
            {
                "name": "contentVector",
                "type": "Collection(Edm.Single)",
```

# Challenge – how to control user access / permission?

1
Aggregate data via Fabric short cuts




2
Form RAG Store and integrate with GenAI Apps

3
Enforce data protection policies and access controls

Copilots

*Purview Compliance Manager*

# But – not only “embedding” or Azure Search Goal to give CONTEXT

# Demo – SK + Custom Bing Search + REST Search in CT …

Z:\A\tkopacz\AzFY23-AI\OpenAI\fy24-sk-and-searchfunctions\fy24-sk-and-searchfunctions.sln

# Challenge: throughput, PTU, TPMs, large scale systems

(Retry– mandatory, but….)

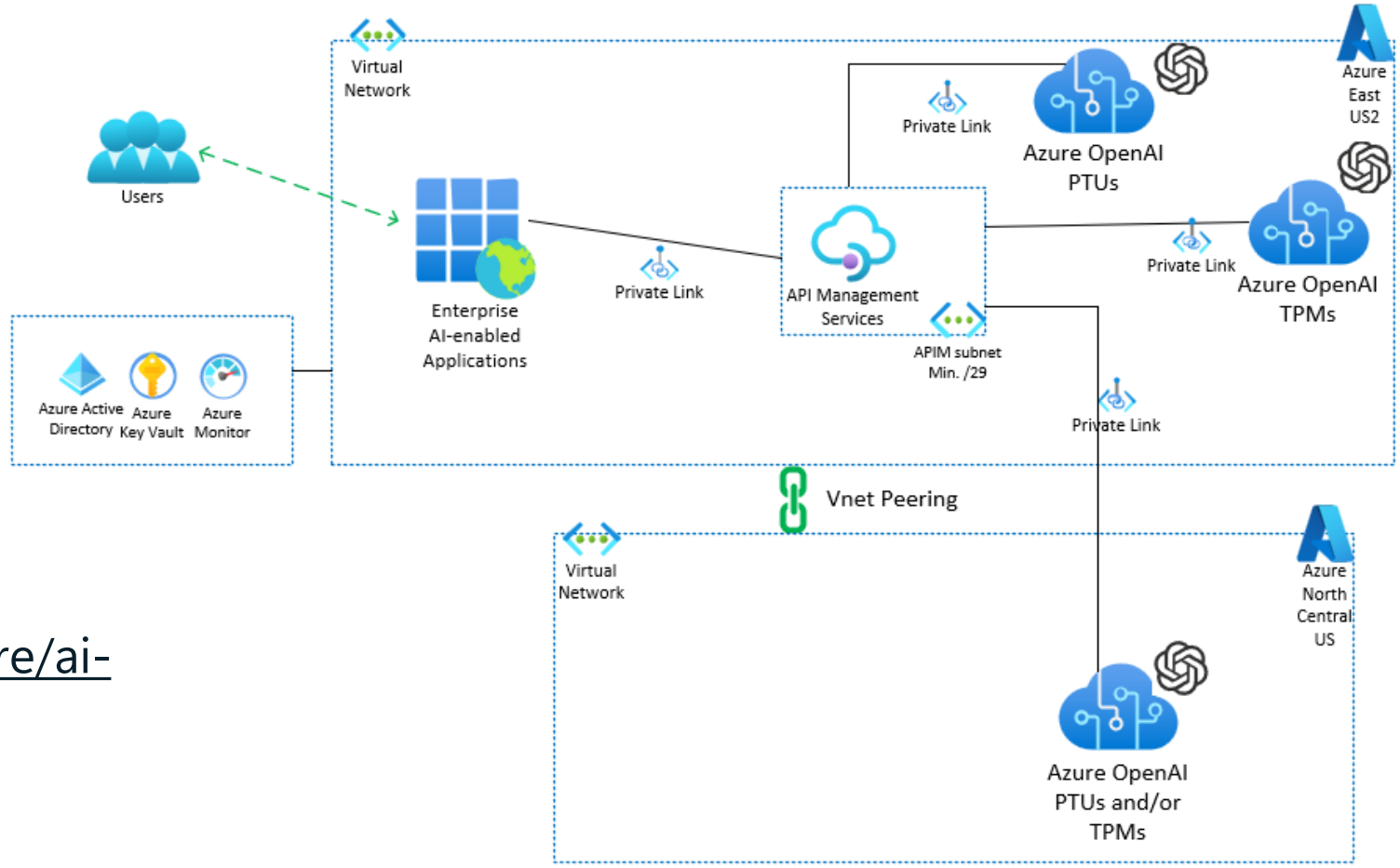
**Scale Out:**

https://github.com/Azure/aoai-apim
https://github.com/azure-samples/azure-openai-apim-load-balancing

**Current limits:**

https://learn.microsoft.com/en-us/azure/ai-services/openai/quotas-limits

**QUEUE – RECOMMENDED!**



# Timings...

| Input token | Output token | AOAI 0125 | OAI gpt-4-turbo | AOAI | AOAI 2 | AOAI 3 | AOAI 4 | AOAI 5 | AOAI AVG | AOAI vs OAI | AOAI vs OAI % |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1457 | 73 | 5.04 | 4.24 | 2.73 | 2.37 | 2.07 | 2.07 | 2.38 | 2.32 | 1.86 | 80% |
| 49 | 804 | 52.87 | 43.32 | 20.81 | 22.48 | 25.09 | 27.95 | 23.66 | 24.00 | 19.66 | 82% |
| 41 | 292 | 24.70 | 15.97 | 7.50 | 8.05 | 5.45 | 5.84 | 11.83 | 7.73 | 4.14 | 53% |
| 43 | 714 | 41.35 | 26.40 | 20.30 | 24.54 | 25.69 | 25.54 | 19.11 | 23.04 | 7.29 | 32% |
| 54 | 593 | 44.57 | 29.10 | 14.06 | 14.34 | 20.27 | 12.39 | 16.56 | 15.52 | 12.54 | 81% |
| 57 | 663 | 37.97 | 25.73 | 18.17 | 20.78 | 16.61 | 15.56 | 15.78 | 17.38 | 9.94 | 57% |
| 36 | 113 | 21.06 | 10.93 | 3.81 | 6.02 | 4.55 | 7.41 | 4.28 | 5.21 | 6.65 | 128% |
| 41 | 653 | 39.03 | 30.20 | 20.22 | 17.48 | 28.11 | 22.86 | 26.44 | 23.02 | 3.76 | 16% |
| 43 | 170 | 30.76 | 10.04 | 5.64 | 9.56 | 4.78 | 6.49 | 7.60 | 6.81 | 2.44 | 36% |
| 49 | 695 | 93.72 | 29.47 | 20.49 | 25.75 | 25.51 | 22.43 | 23.04 | 23.44 | 6.44 | 27% |

*© Piotr Bubacz, time in seconds*

# "Traditional" Databases?

SQL – TPC-C (tpm-C – orders / sec) – 1.2mln (in 2005!), now – 814mln (dedicated cluster)

Cosmos DB - 1000xxx RU/s (reads 4K "jsons")

# So - queue (and Service Bus)

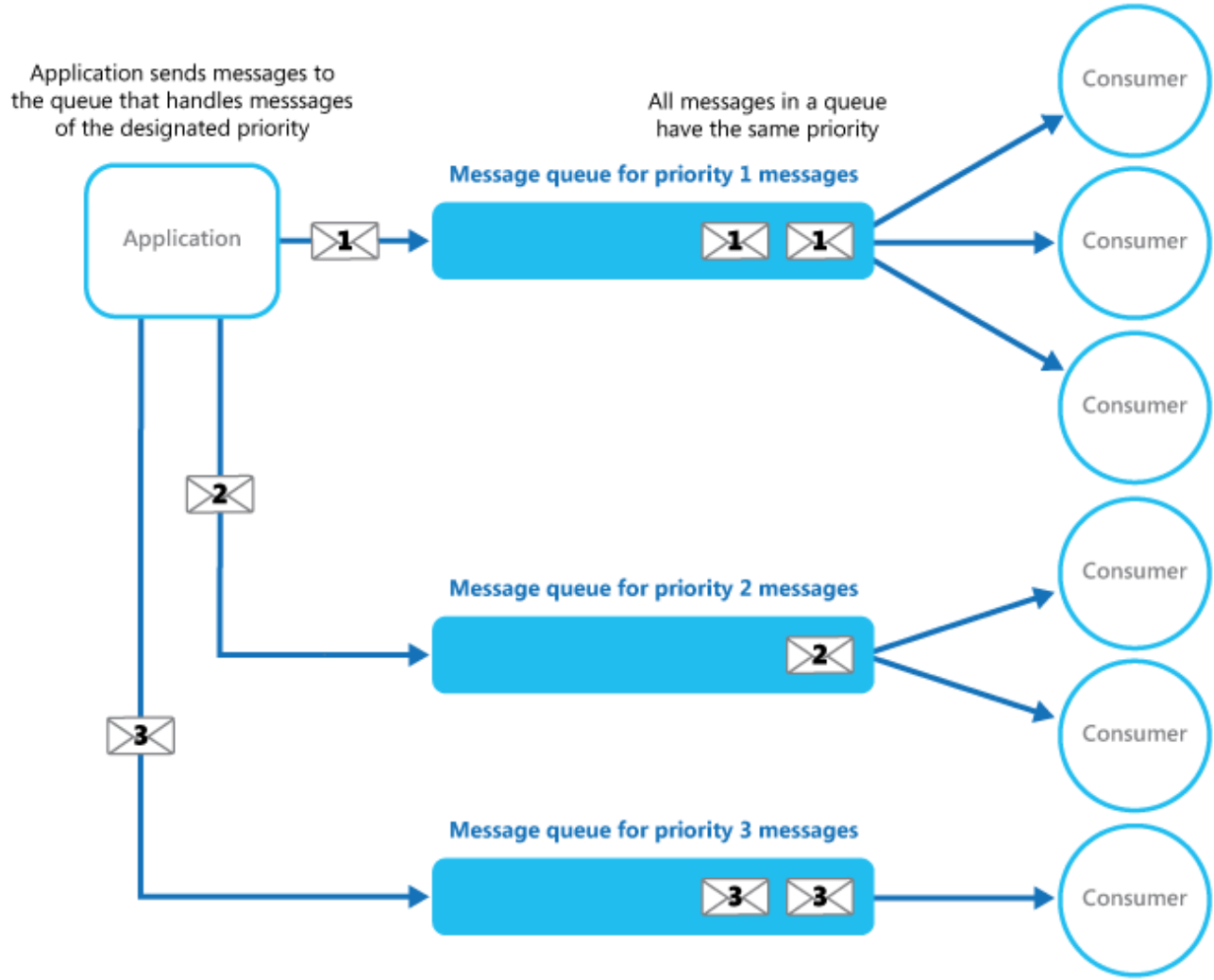
Topics and subscription
Automatic dead-lettering
Message deferral
Transactions

Can implement priority queue pattern!
Can choose model based on price/performance/capabilities/…

Exposed API:
- Send “order” | Get “order” result

Processing:
- pick message, route it, execute actions

*Ps. Concepts like https://openrouter.ai/. But read **first**: https://openrouter.ai/privacy*



# Demo – Semantic Kernel and QUEUE (Service Bus)
# And – “EMBEDDING”

Z:\A\tkopacz\AzFY23-AI\OpenAI\fy24-sb-aoi-blazor\fy24-sb-aoi-blazor.sln
Embedding,
Long/short

# Demo – memory
# (and .NET Notebooks)

# Building Semantic Memory with Embeddings

So far, we've mostly been treating the kernel as a stateless orchestration engine. We send text into a model API and receive text out.

In a previous notebook, we used `context variables` to pass in additional text into prompts to enrich them with more context. This allowed us to create a basic chat experience.

However, if you solely relied on context variables, you would quickly realize that eventually your prompt would grow so large that you would run into a the model's token limit. What we need is a way to persist state and build both short-term and long-term memory to empower even more intelligent applications.

To do this, we dive into the key concept of `Semantic Memory` in the Semantic Kernel.

```
#r "nuget: Microsoft.SemanticKernel, 1.0.0-beta1"
#r "nuget: System.Linq.Async, 6.0.1"

#!import config/Settings.cs

using Microsoft.SemanticKernel;
using Microsoft.SemanticKernel.SemanticFunctions;
using Microsoft.SemanticKernel.Orchestration;

var kernelBuilder = new KernelBuilder();

// Configure AI backend used by the kernel
var (useAzureOpenAI, model, azureEndpoint, apiKey, orgId) = Settings.LoadFromFile();

if (useAzureOpenAI)
    kernelBuilder.WithAzureChatCompletionService(model, azureEndpoint, apiKey);
else
    kernelBuilder.WithOpenAIChatCompletionService(model, apiKey, orgId);

var kernel = kernelBuilder.Build();
```

csharp - C# Script Code

In order to use memory, we need to instantiate the Memory Plugin with a Memory Storage and an Embedding backend. In this example, we make use of the `VolatileMemoryStore` which can be thought of as a temporary in-memory storage (not to be confused with Semantic Memory).

This memory is not written to disk and is only available during the app session.

When developing your app you will have the option to plug in persistent storage like Azure Cosmos Db, PostgreSQL, SQLite, etc. Semantic Memory allows also to index external data sources, without duplicating all the information, more on that later.

# Kernel Memory (OSS project)

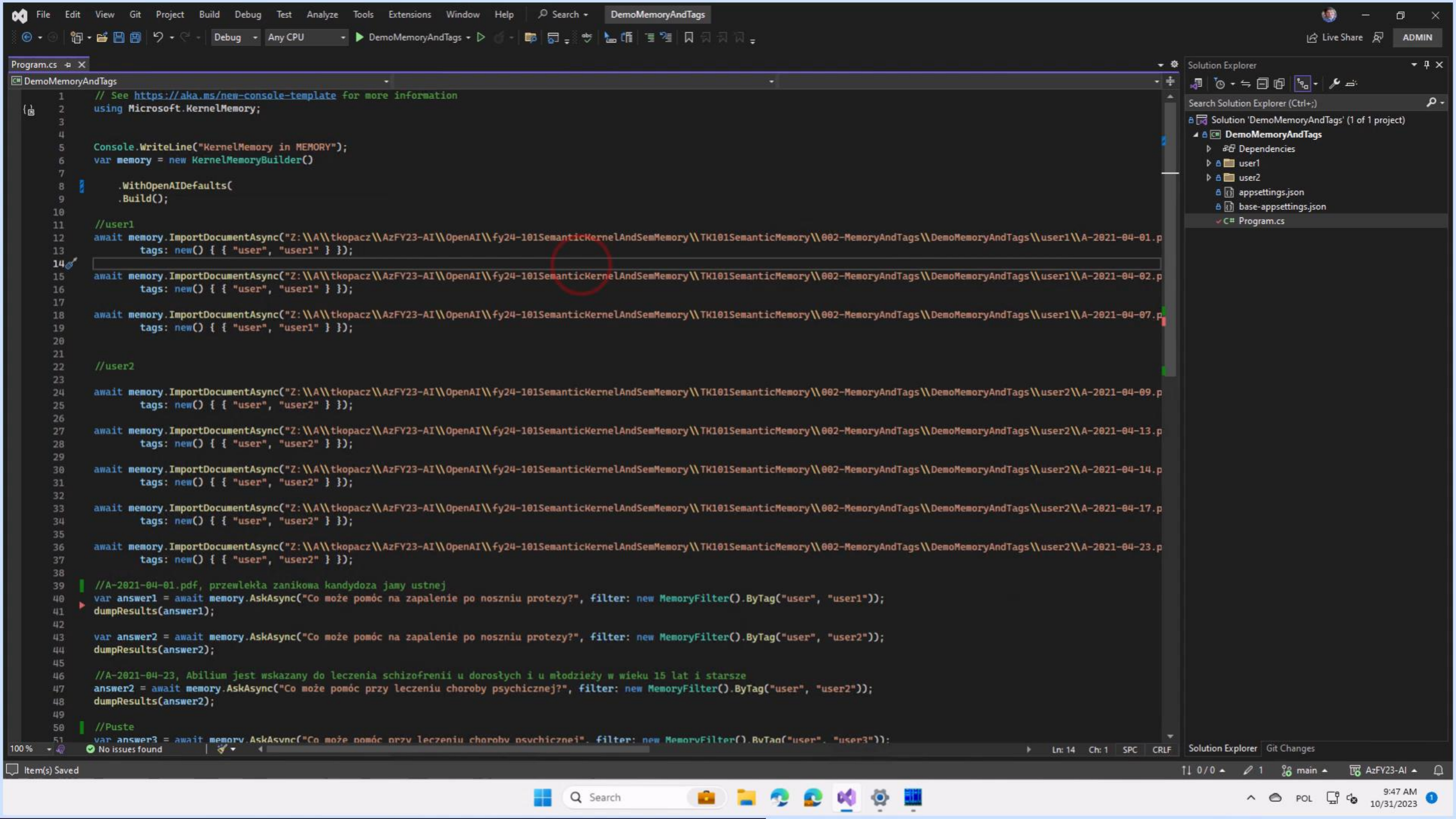
**Kernel Memory** (KM) is an open-source service and plugin specialized in the efficient indexing of datasets through custom continuous data hybrid pipelines.

Available as: InProc, WebApi

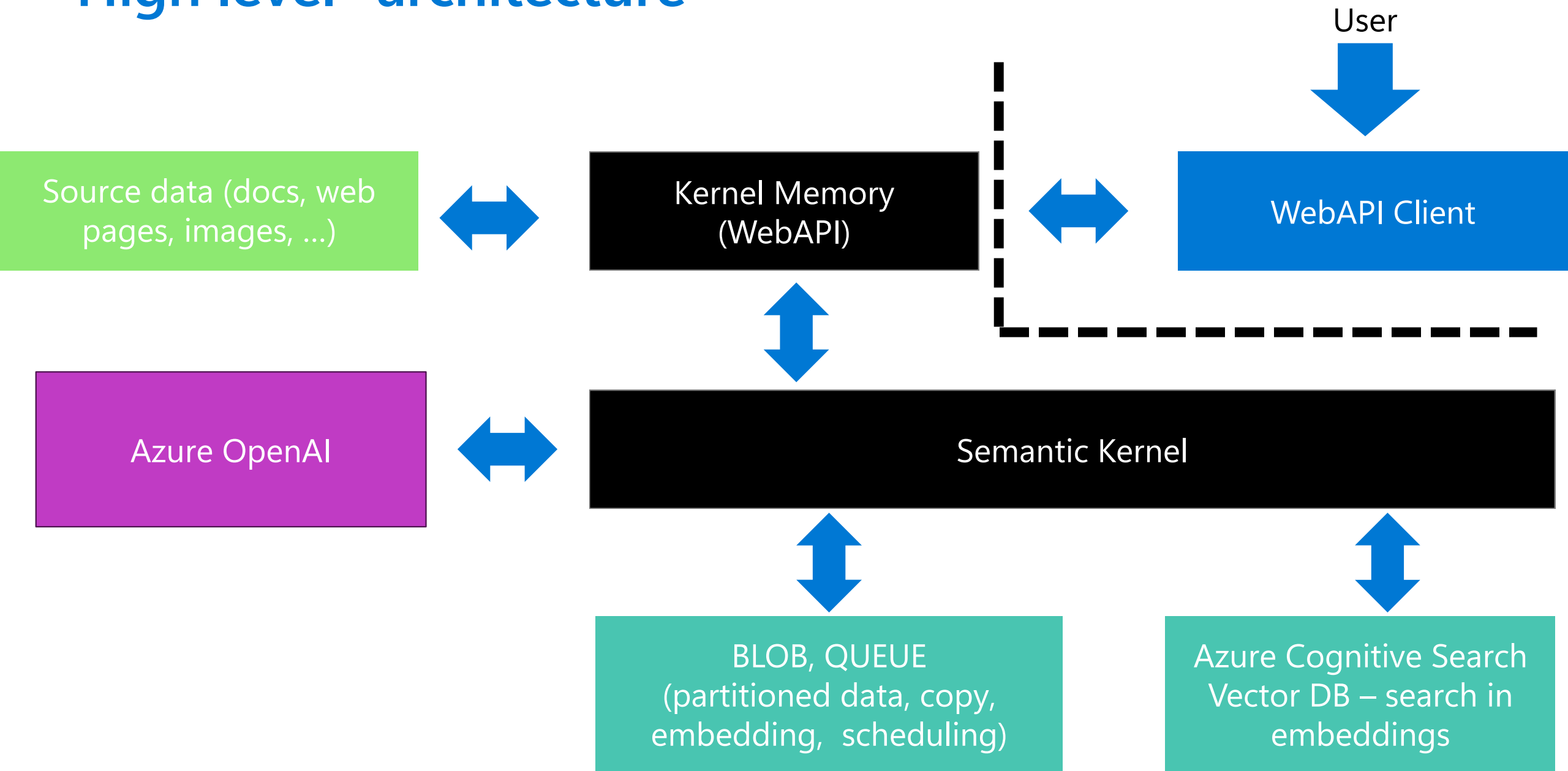
Using SEMANTIC KERNEL (under the hood – see source code)

```
000-notebooks
001-dotnet-WebClient
002-dotnet-SemanticKernel-plugin
002-dotnet-Serverless
003-dotnet-SemanticKernel-plugin
003-dotnet-Serverless
004-dotnet-serverless-custom-pipeline
005-dotnet-async-memory-custom-pipeline
006-curl-calling-webservice
101-dotnet-custom-Prompts
102-dotnet-custom-partitioning-options
103-dotnet-custom-EmbeddingGenerator
104-dotnet-custom-LLM
105-dotnet-serverless-llamasharp
106-dotnet-retrieve-synthetics
107-dotnet-SemanticKernel-TextCompletion
108-dotnet-custom-content-decoders
109-dotnet-custom-webscraper
111-dotnet-azure-ai-hybrid-search
200-dotnet-nl2sql
201-dotnet-serverless-custom-handler
202-dotnet-custom-handler-as-a-service
203-dotnet-using-core-nuget
204-dotnet-ASP.NET-MVC-integration
205-dotnet-extract-text-from-docs
206-dotnet-configuration-and-logging
207-dotnet-expanding-chunks-on-retrieval
208-dotnet-lmstudio
301-discord-test-application
```

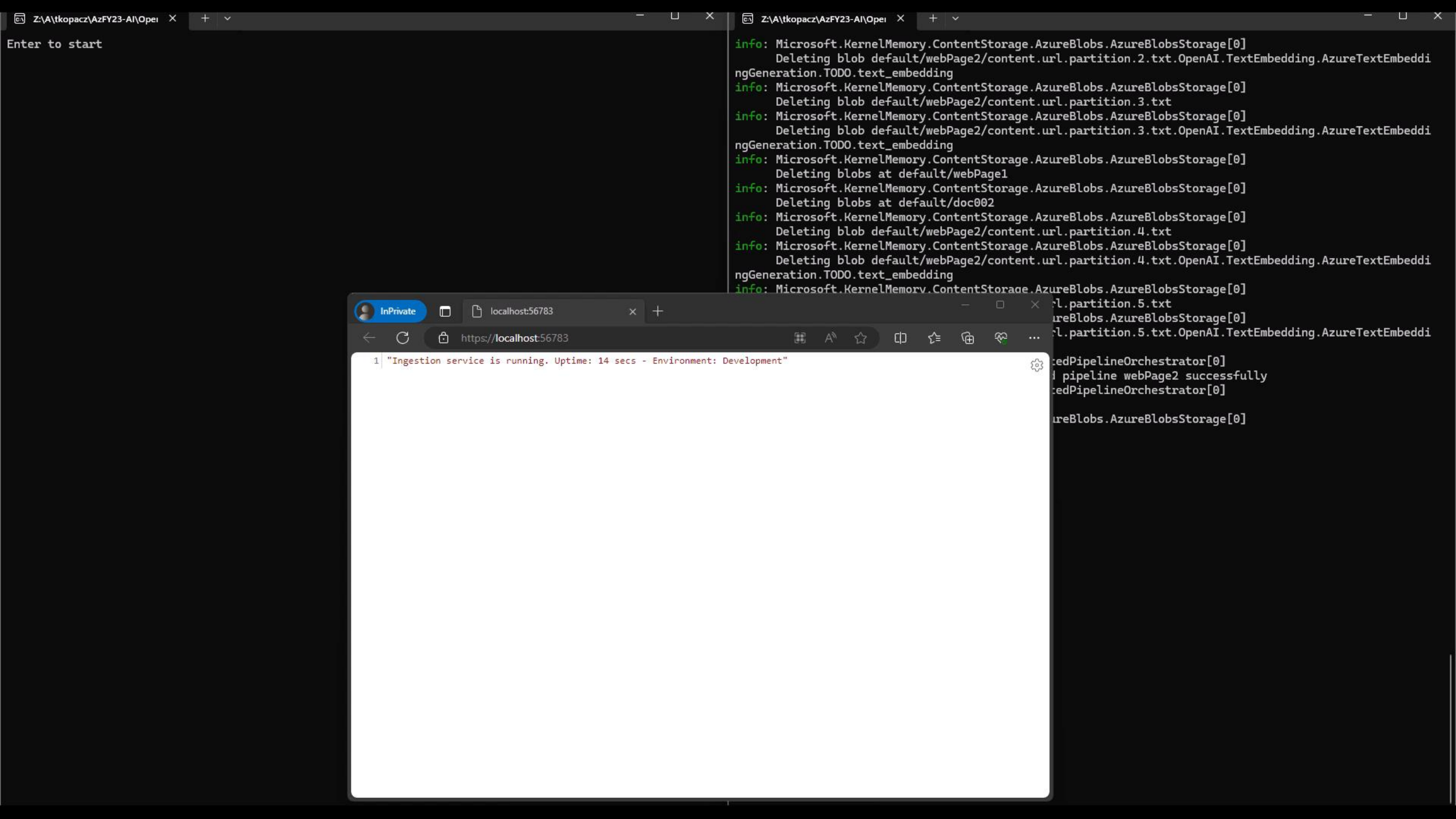
# Demo – KM and Tags - InMemory


File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help
Search
DemoMemoryAndTags
Debug
Any CPU
DemoMemoryAndTags
Live Share
ADMIN
Program.cs
DemoMemoryAndTags
// See https://aka.ms/new-console-template for more information
using Microsoft.KernelMemory;
Console.WriteLine("KernelMemory in MEMORY");
var memory = new KernelMemoryBuilder()
.WithOpenAIDefaults(
.Build();
//user1
await memory.ImportDocumentAsync("Z:\\A\\tkopacz\\AzFY23-AI\\OpenAI\\fy24-101SemanticKernelAndSemMemory\\TK101SemanticMemory\\002-MemoryAndTags\\DemoMemoryAndTags\\user1\\A-2021-04-01.p
tags: new() { { "user", "user1" } });
await memory.ImportDocumentAsync("Z:\\A\\tkopacz\\AzFY23-AI\\OpenAI\\fy24-101SemanticKernelAndSemMemory\\TK101SemanticMemory\\002-MemoryAndTags\\DemoMemoryAndTags\\user1\\A-2021-04-02.p
tags: new() { { "user", "user1" } });
await memory.ImportDocumentAsync("Z:\\A\\tkopacz\\AzFY23-AI\\OpenAI\\fy24-101SemanticKernelAndSemMemory\\TK101SemanticMemory\\002-MemoryAndTags\\DemoMemoryAndTags\\user1\\A-2021-04-07.p
tags: new() { { "user", "user1" } });
//user2
await memory.ImportDocumentAsync("Z:\\A\\tkopacz\\AzFY23-AI\\OpenAI\\fy24-101SemanticKernelAndSemMemory\\TK101SemanticMemory\\002-MemoryAndTags\\DemoMemoryAndTags\\user2\\A-2021-04-09.p
tags: new() { { "user", "user2" } });
await memory.ImportDocumentAsync("Z:\\A\\tkopacz\\AzFY23-AI\\OpenAI\\fy24-101SemanticKernelAndSemMemory\\TK101SemanticMemory\\002-MemoryAndTags\\DemoMemoryAndTags\\user2\\A-2021-04-13.p
tags: new() { { "user", "user2" } });
await memory.ImportDocumentAsync("Z:\\A\\tkopacz\\AzFY23-AI\\OpenAI\\fy24-101SemanticKernelAndSemMemory\\TK101SemanticMemory\\002-MemoryAndTags\\DemoMemoryAndTags\\user2\\A-2021-04-14.p
tags: new() { { "user", "user2" } });
await memory.ImportDocumentAsync("Z:\\A\\tkopacz\\AzFY23-AI\\OpenAI\\fy24-101SemanticKernelAndSemMemory\\TK101SemanticMemory\\002-MemoryAndTags\\DemoMemoryAndTags\\user2\\A-2021-04-17.p
tags: new() { { "user", "user2" } });
await memory.ImportDocumentAsync("Z:\\A\\tkopacz\\AzFY23-AI\\OpenAI\\fy24-101SemanticKernelAndSemMemory\\TK101SemanticMemory\\002-MemoryAndTags\\DemoMemoryAndTags\\user2\\A-2021-04-23.p
tags: new() { { "user", "user2" } });
//A-2021-04-01.pdf, przewlekła zanikowa kandydoza jamy ustnej
var answer1 = await memory.AskAsync("Co może pomóc na zapalenie po noszniu protezy?", filter: new MemoryFilter().ByTag("user", "user1"));
dumpResults(answer1);
var answer2 = await memory.AskAsync("Co może pomóc na zapalenie po noszniu protezy?", filter: new MemoryFilter().ByTag("user", "user2"));
dumpResults(answer2);
//A-2021-04-23, Abilium jest wskazany do leczenia schizofrenii u dorosłych i u młodzieży w wieku 15 lat i starsze
answer2 = await memory.AskAsync("Co może pomóc przy leczeniu choroby psychicznej?", filter: new MemoryFilter().ByTag("user", "user2"));
dumpResults(answer2);
//Puste
var answer3 = await memory.AskAsync("Co może pomóc przy leczeniu choroby psychicznej", filter: new MemoryFilter().ByTag("user", "user3"));
100 %
No issues found
Ln: 14 Ch: 1 SPC CRLF
Solution Explorer
Search Solution Explorer (Ctrl+;)
Solution 'DemoMemoryAndTags' (1 of 1 project)
DemoMemoryAndTags
Dependencies
user1
user2
appsettings.json
base-appsettings.json
Program.cs
Solution Explorer
Git Changes
Item(s) Saved
0 / 0
1
main
AzFY23-AI
Search
POL
9:47 AM
10/31/2023

# High level “architecture”



# Demo – KM as a SERVICE


Z:\A\tkopacz\AzFY23-AI\Opei
Enter to start
Z:\A\tkopacz\AzFY23-AI\Opei
info: Microsoft.KernelMemory.ContentStorage.AzureBlobs.AzureBlobsStorage[0]
Deleting blob default/webPage2/content.url.partition.2.txt.OpenAI.TextEmbedding.AzureTextEmbeddingGeneration.TODO.text_embedding
info: Microsoft.KernelMemory.ContentStorage.AzureBlobs.AzureBlobsStorage[0]
Deleting blob default/webPage2/content.url.partition.3.txt
info: Microsoft.KernelMemory.ContentStorage.AzureBlobs.AzureBlobsStorage[0]
Deleting blob default/webPage2/content.url.partition.3.txt.OpenAI.TextEmbedding.AzureTextEmbeddingGeneration.TODO.text_embedding
info: Microsoft.KernelMemory.ContentStorage.AzureBlobs.AzureBlobsStorage[0]
Deleting blobs at default/webPage1
info: Microsoft.KernelMemory.ContentStorage.AzureBlobs.AzureBlobsStorage[0]
Deleting blobs at default/doc002
info: Microsoft.KernelMemory.ContentStorage.AzureBlobs.AzureBlobsStorage[0]
Deleting blob default/webPage2/content.url.partition.4.txt
info: Microsoft.KernelMemory.ContentStorage.AzureBlobs.AzureBlobsStorage[0]
Deleting blob default/webPage2/content.url.partition.4.txt.OpenAI.TextEmbedding.AzureTextEmbeddingGeneration.TODO.text_embedding
InPrivate
localhost:56783
https://localhost:56783
1 "Ingestion service is running. Uptime: 14 secs - Environment: Development"

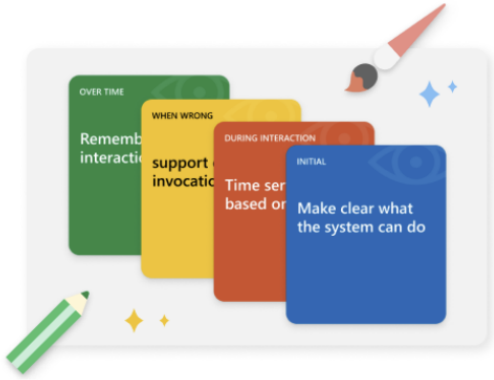
# Last – but THE MOST IMPORTANT

*PROPERLY DESIGN INTERACTION (not only UI)*

# HAX Toolkit

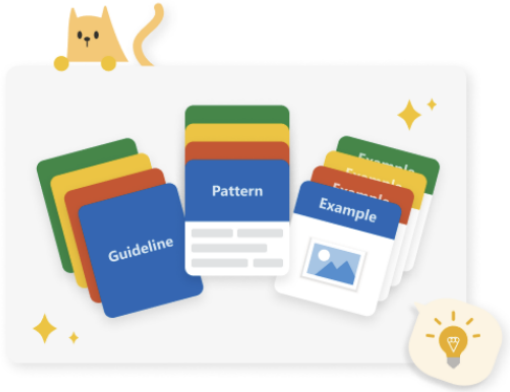
https://www.microsoft.com/en-us/haxtoolkit/

The HAX Toolkit is for teams building user-facing AI products. It helps you conceptualize what the AI system will do and how it will behave. Use it early in your design process.




[Guidelines for Human-AI Interaction]

Best practices for how AI systems should behave during interaction. Use them to guide your AI product planning.

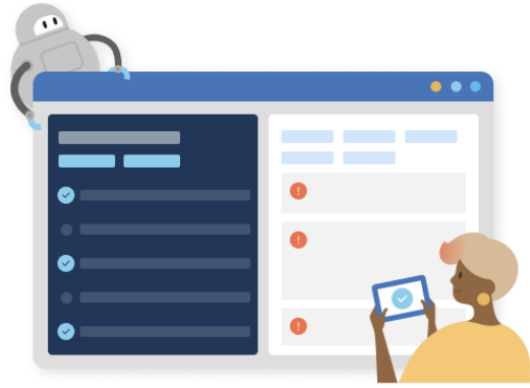



[HAX Design Library]

Learn the Guidelines for Human-AI Interaction and how to apply them, using patterns and examples.

[HAX Workbook]

Work together with your team to prioritize which Guidelines to implement in your product.

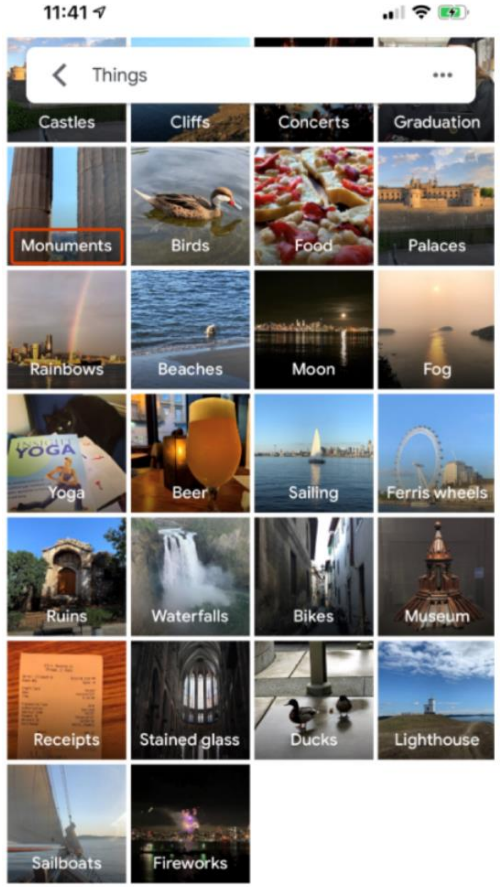


[HAX Playbook]

For applications using natural language processing, identify common failures so you can plan for mitigating them.

# Find good examples based on fav services


Google Photos sets expectations for what the system can do (Guideline 1) by providing tappable labels that let the user know the system can automatically categorize images based on their content (Pattern 1C). *Image captured July 2020.*




Copilot in Outlook makes clear how well the system can do what it can do (Guideline 2) by using latency moments as opportunities to educate users. Here, it reminds the user to review the email draft before sending it.

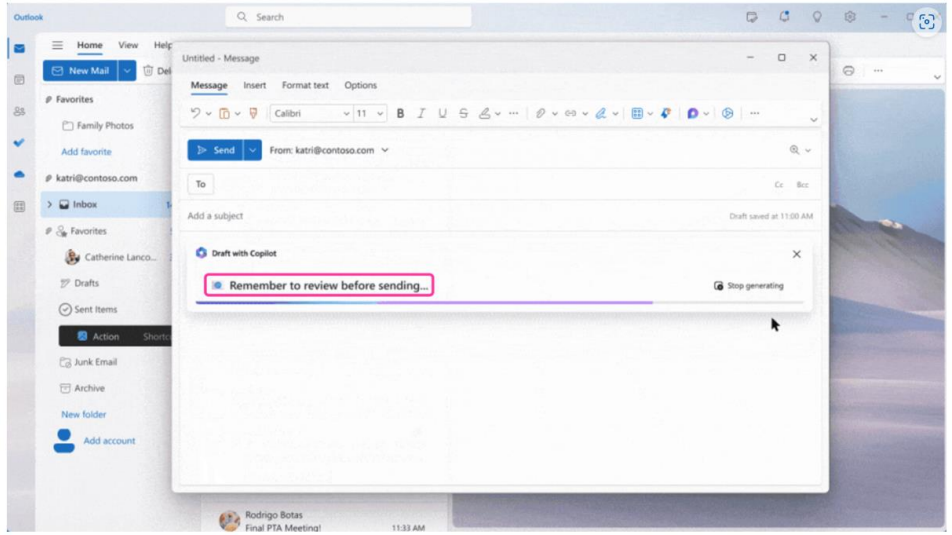



*Image captured March 2023 (source)*

G2: Make clear how well the system can do what it can do. | Email | Productivity | Writing and editing | Generative AI | Large Language Model | Natural language processing (Text) | Text generation

# Schillace Laws of Semantic AI and Responsible AI tools

**Don’t write code if the model can do it; the model will get better, but the code won’t**

**Trade leverage for precision; use interaction to mitigate**

**Code is for syntax and process; models are for semantics and intent**

**The system will be as brittle as its most brittle part**

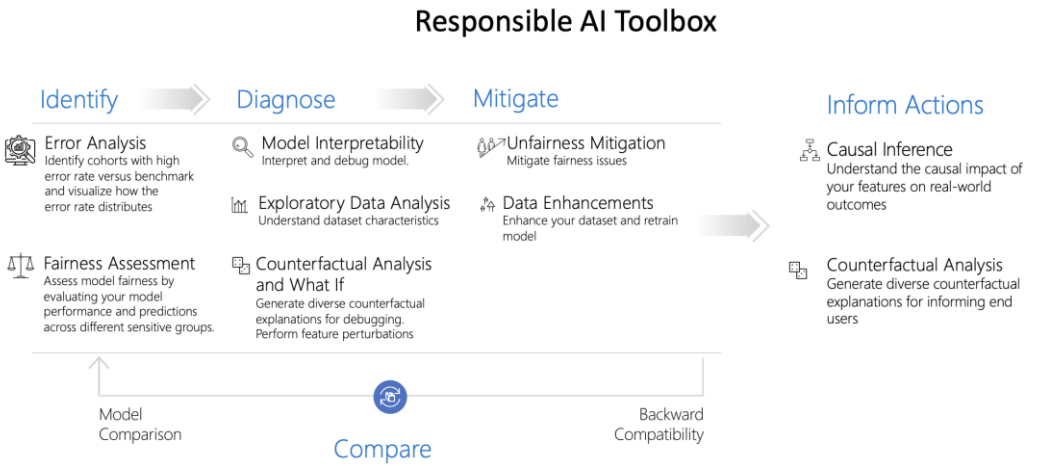
**Ask Smart to Get Smart**

**Uncertainty is an exception throw**

**Text is the universal wire protocol**

**Hard for you is hard for the model**

**Beware "pareidolia of consciousness"; the model can be used against itself.**



Microsoft

# Thank you!

Tomasz Kopacz
tkopacz@microsoft.com

**Links:**

https://github.com/microsoft/semantic-kernel
https://github.com/microsoft/kernel-memory
https://devblogs.microsoft.com/semantic-kernel/

https://github.com/microsoft/semantic-kernel-starters
https://devblogs.microsoft.com/semantic-kernel/category/samples/

https://www.microsoft.com/en-us/haxtoolkit/
https://github.com/microsoft/responsible-ai-toolbox
https://learn.microsoft.com/en-us/semantic-kernel/overview/
https://devblogs.microsoft.com/semantic-kernel/

**Future of Semantic Kernel:**
**https://devblogs.microsoft.com/semantic-kernel/spring-2024-roadmap-for-semantic-kernel/**
**https://devblogs.microsoft.com/semantic-kernel/semantic-kernel-office-hours-recordings/**