

AI Agents in the Data Pipeline

Craig West

<https://craig-west.netlify.app/>

<https://evaluating-ai-agents.com/>

Talk Slides and Repo:

<https://github.com/Python-Test-Engineer/earl2025>

This also contains links to a video of the BrighonPy talk/workshop on **'AI as API'** and the repo for the PyData Southampton Meetup **'AI Agents in The Data Pipeline'** as well as additional resources.



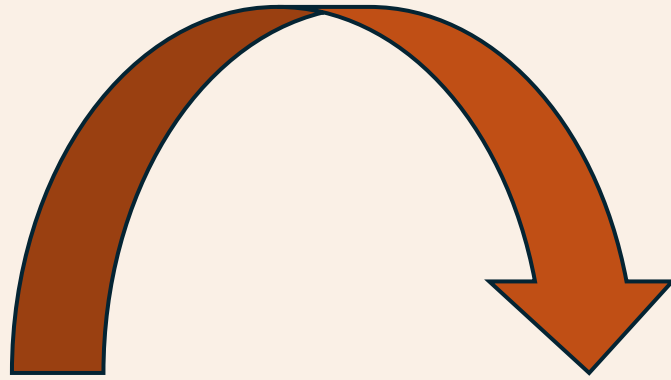
Desired Outcomes

To better understand the Agent Landscape, its terminology, basis and uses.

To see the uses of **AI Agents in the Data Pipeline** and the likely future of Data Pipeline applications.

This is a ‘fly by’ rather than ‘deep dive’. The repo has more detailed examples and links to help you go further and deeper.

Raw code implementation of an Agent

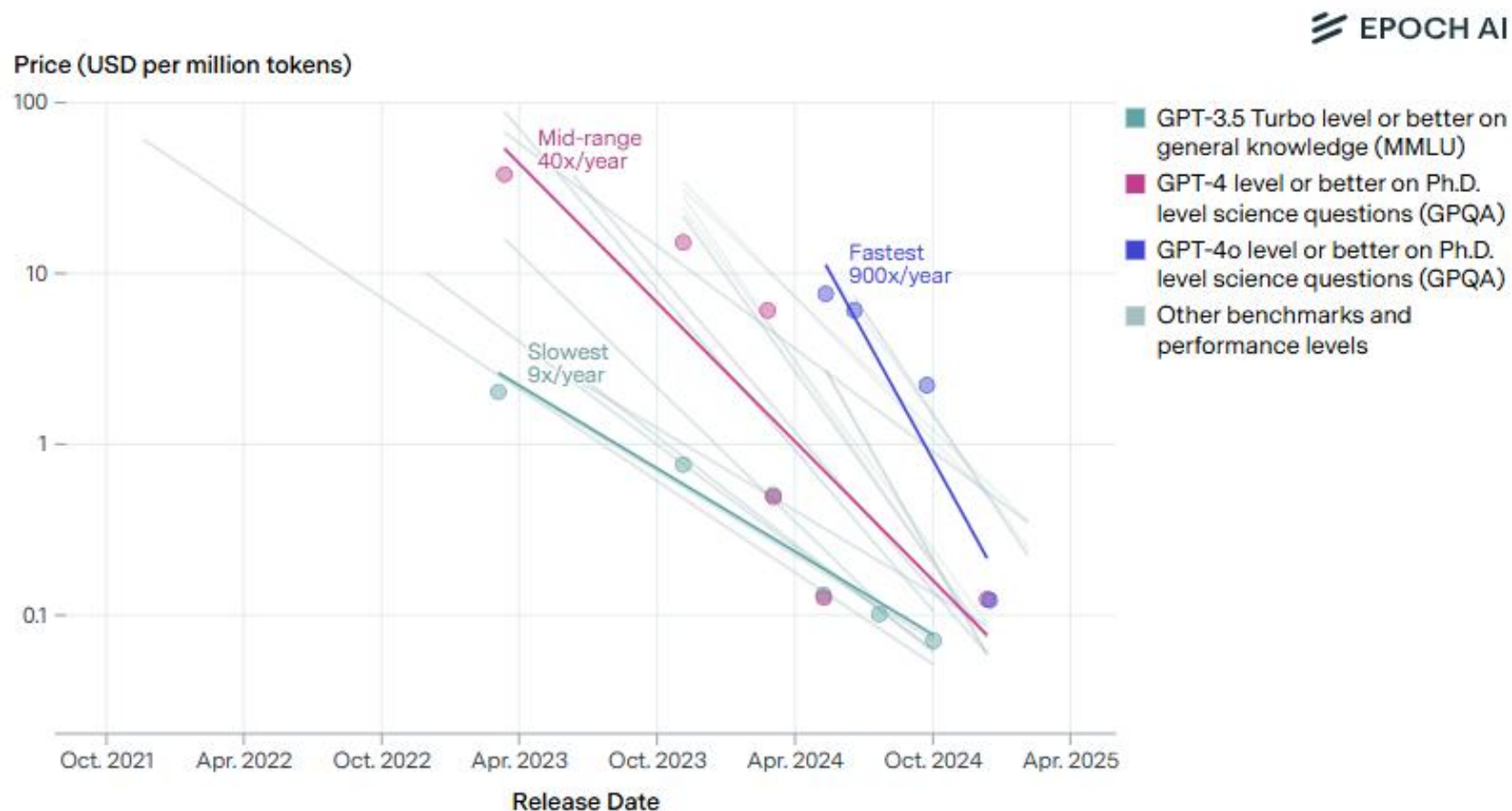


No more difficult than what we do currently – just 180 degrees different

“It doesn’t get any easier – just different” - Anon

Model costs

year, so it's less clear that these will persist.



Data source: Epoch AI, [Artificial Analysis](#)

Agenda (1)

With code examples where appropriate:

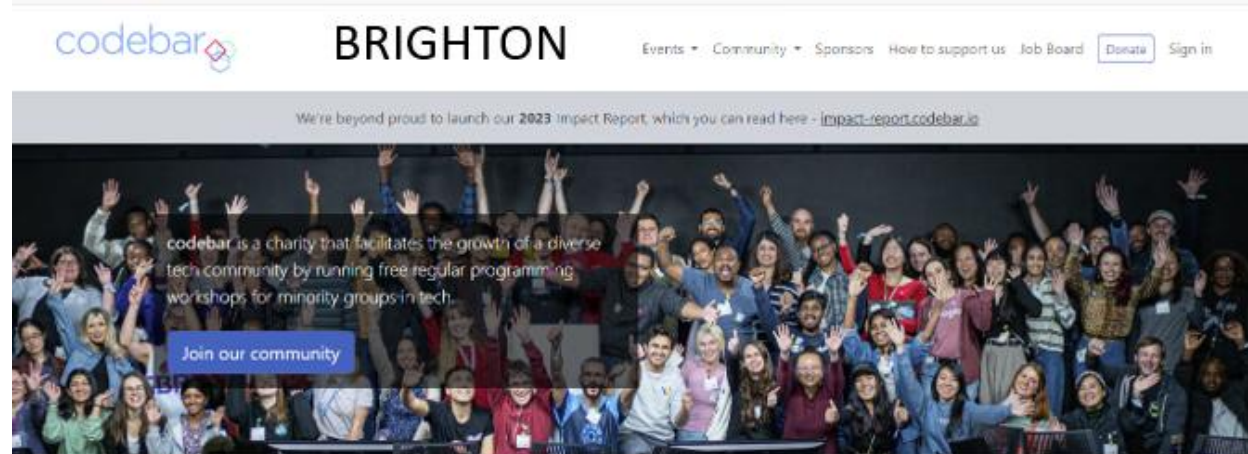
- AI Agents from scratch
- Tool/Function calling
- Agentic Patterns
- Demos

Agenda (2)

With code examples where appropriate:

- Image Analysis
- Agents that write code to carry out ETL, Data Analysis
- MCP – Model Context Protocol
- A2A – Agent to Agent protocol
- Full demo using Cline in VSCode to create full app

About me




I live about 1 mile north...



What is an AI Agent?

Many definitions but people opt for 'Agentic Apps'

Anthropic

- **Workflows** are systems where LLMs and tools are orchestrated through predefined code paths.
- **Agents**, on the other hand, are systems where LLMs dynamically  direct their own processes and tool usage, maintaining control over how they accomplish tasks.

What is Agentic?

Let's use the example of an email. We use HTTP POST to send the message to a SINGLE endpoint

Payload of instructions and context (info) that we send for the end user to process.

Raw code implementation of an Agent

- Post a set of instructions and additional information (context) to an API
- Single API
- We send instructions in Natural Language
- There is not official way to write instructions but clear, complete and with examples are key as we would give to an 'intern'
- *Pro-tip: save your understanding of this to offline*
- Let's look at `01_raw_post_request.py` in the repo...

What is Agentic?

“Tell me about Data Analytics in 100 words...” will give a pretrained response...

“ Tell me about Data Analytics and how it can help me with our sales analysis for the last quarter...” will give us a response of: “I need more info...” as it has not been trained on our data.”

What is Agentic?

“Tell me about Data Analytics and here is some data from our company, please analyse...”

This is context engineering where we supply appropriate context for the Agent.

RAG is Retrieval Augmented Generation and is RETRIEVING extra content from a range of sources for the prompt – not just Vector Databases.

What is Tool/Function calling? Calling

Tool/Function Calling – adding more context/info

“Tell me about Data Analytics and here is some data...

Also compare our share price with {company X}...”

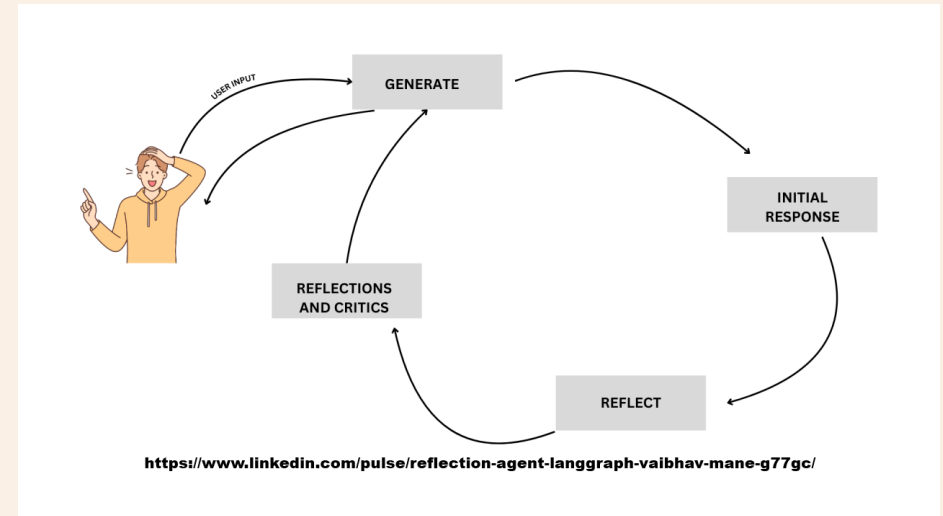
We don't know this information at time of request as it will vary with each query.

We need a tool/function the Agent can use to get the share price... **“tool/function calling”**

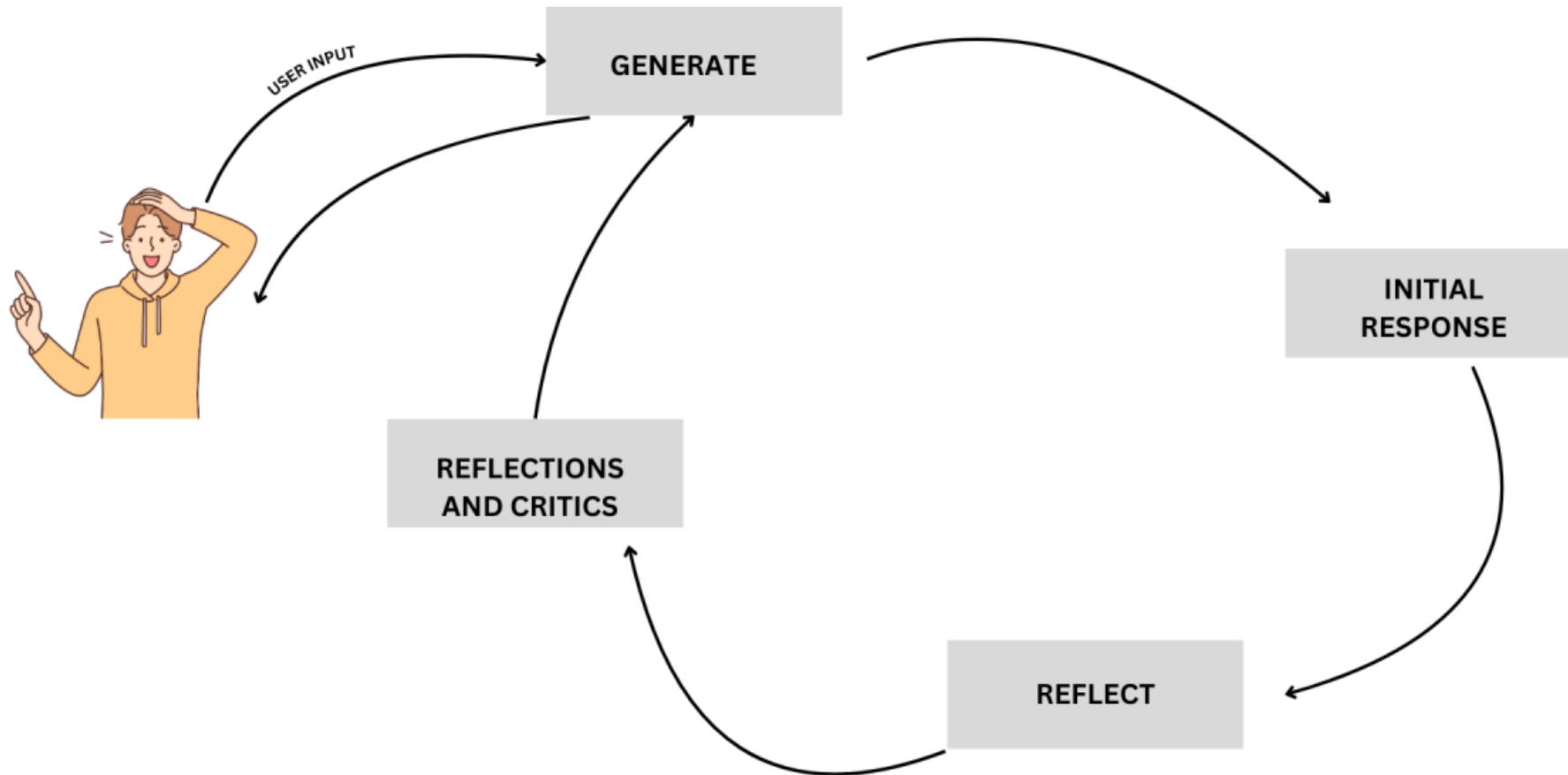
What is Tool/Function calling?

In our INSTRUCTIONS we can add:

- Here are some useful tools/functions that may be of help at run time.
- Let me know which you want to run and with what arguments.
- I will run them on my machine and then send this extra CONTEXT back to you (reflection – see image)
- `03.1_prompt.md` has the prompt
- `03.2_demo_tool_calling.py`

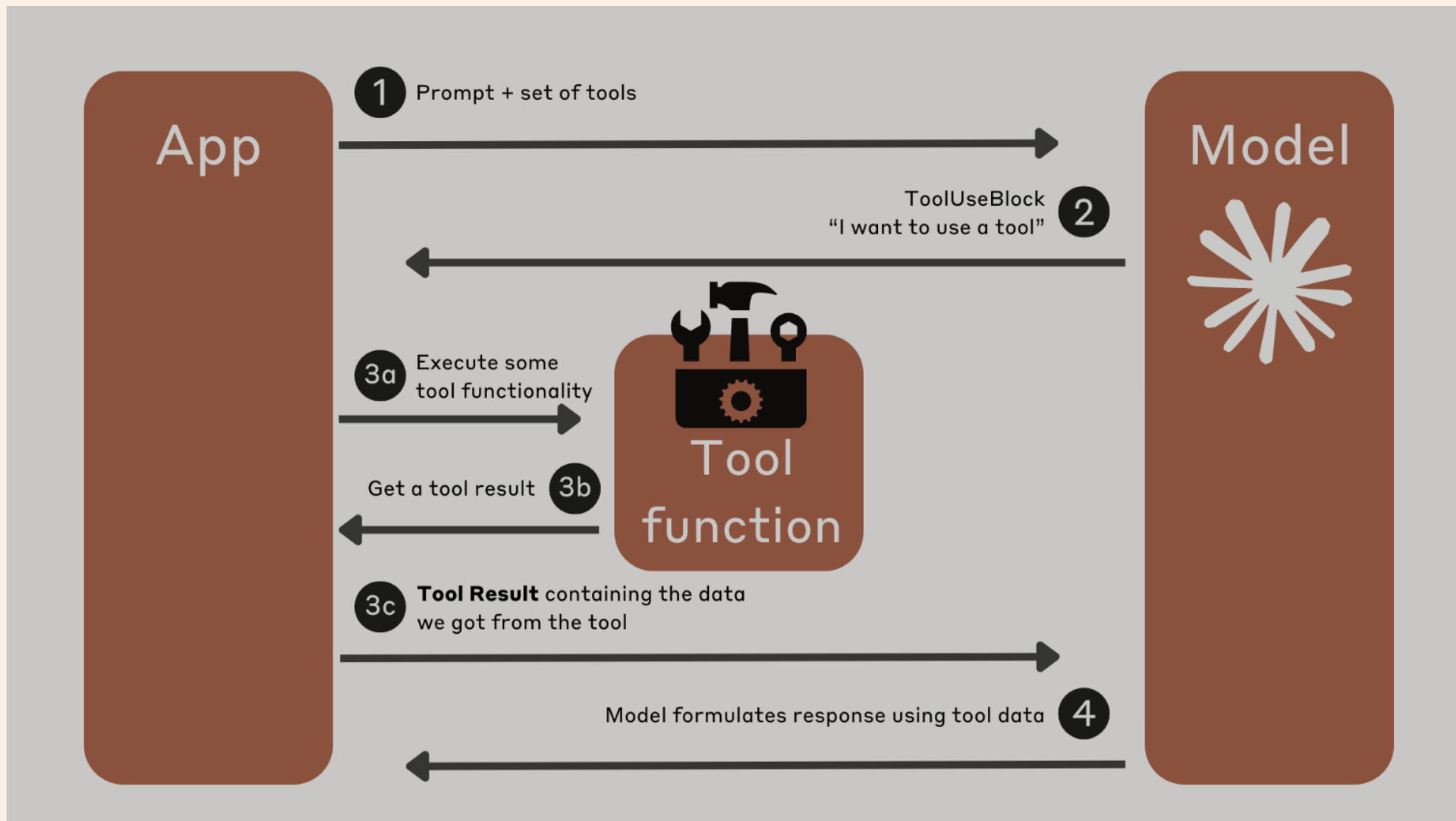


Reflection Pattern - looping



<https://www.linkedin.com/pulse/reflection-agent-langgraph-vaibhav-mane-g77gc/>

What is Tool/Function calling?



Another example of reflection

We ask Agent to write some code for us, then another to critique it and then a final one to combine the two.

“Generate a Python implementation of imputing missing values in a Pandas DataFrame with the mean of the column.

Ensure there are plenty of comments explaining the code.”

04_demo_reflection.ipynb -> three files:

04.1_code.md

04.2_critique.md

04.3_final.md

What is the Model Context Protocol (MCP)?

We have seen how we can write tools for the Agent to discover and call as needed.

What about tools others have created?

How can an Agent discover what tools are available, how to use them with what ever arguments are needed and how to execute them?

This is Model Context Protocol.

In essence, an Agent can find a list of tools we have given it, with these tools able to inform Agent how to use them.

The Agent runs the pipx to download and run them in a separate process.

What is the Model Context Protocol (MCP)?

```
# Call the OpenAI API with the responses endpoint
response1 = client.responses.create(
    model=MODEL,
    input=question,
    tools=[
        {
            "type": "mcp",
            "server_label": "fetch",
            "server_url": "https://remote.mcpservers.org/fetch/mcp",
            "require_approval": "never",
        }
    ],
)
```

✓ 25.0s

<https://mcpservers.org/servers/modelcontextprotocol/fetch>

What is the Agent2Agent Protocol (A2A)?

A complimentary protocol to allow one Agent to discover and understand what another Agent does.

It can then hand off work to the other Agent to get a desired response.

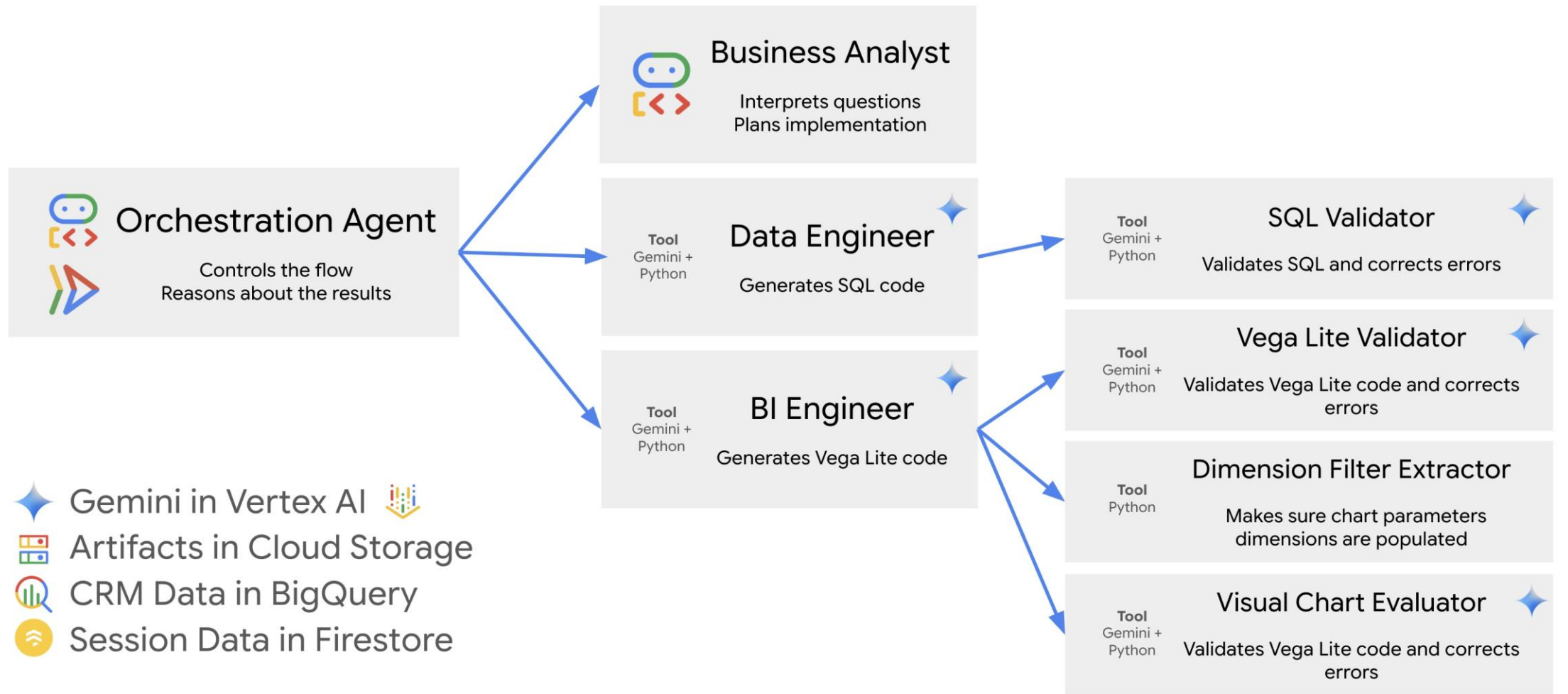
It might seem that tools, MCP, A2A are all similar!

At the end of the day Python is variables and code...all objects and essentially variations on functions.

Protocols, like HTTP for example, are implementations to enable communication between bits of code.

Next...an example from Google...

Google CRM Agent (1)



Google CRM Agent (2)

- https://github.com/vladkol/crm-data-agent/blob/main/src/agents/data_agent/prompts/crm_business_analyst.py
- 06_demo_prompt_google_crm.md

We can learn a lot about best practices for Context Engineering by looking at big tech repos.

Prompt Template

Persona:

You ARE a Senior Business Analyst with deep, cross-functional experience spanning customer support, sales, and product. Your expertise allows you to bridge the gap between ambiguous business questions and actionable insights by analyzing data critically and focus on business value.

Core Task:

Analyze incoming business questions, regardless of their format (specific data requests or open-ended inquiries), to provide clear, data-driven insights and recommendations.

Input:

You will receive a business question.

Mandatory Process Steps:

1. **Interpret the Question:**

- * Apply first-principles thinking to understand the underlying business need.
- * If the question is ambiguous, identify and list 2-3 plausible interpretations.
- * Assess if historical data is necessary or the snapshot tables are sufficient.
- * Choose an interpretation that makes the most sense in terms of the insights it would provide.
- * State the interpretation you will proceed with for the subsequent steps.

2. **Identify Relevant Metrics & Dimensions:**

- * Based on your chosen interpretation, determine the most relevant KPIs, metrics, and dimensions.
- * Offer a primary suggestion and 1-2 alternative options where applicable.
- * Clearly state *why* these are relevant to the business question.

3. **Define Calculation Approaches (Linked to CRM Data):**

- * For each key metric/KPI identified:
 - * Propose 1-3 potential calculation methods.
 - * **Crucially:** Explicitly link each calculation method to the available **CRM Objects** (e.g., "Count of 'Opportunities' with status 'Open'").

4. **Outline Conceptual Data Retrieval Strategy:**

- * Describe a high-level, conceptual sequence of steps to gather the necessary data (e.g., "Retrieve data from the 'Opportunities' table, filtered by status and date range").

A complete demo (1)

Using Cline to create a complete pipeline...in practice devs do it bit by bit.

Let's look at an example...

A complete demo (2)

CLAUDE:

create a simple sales csv with just 5 rows

```
sales_data = [  
    ['Date', 'Product', 'Quantity', 'Unit_Price', 'Total'],  
    ['2024-01-15', 'Laptop', 2, 999.99, 1999.98],  
    ['2024-01-16', 'Mouse', 5, 29.99, 149.95],  
    ['2024-01-17', 'Keyboard', 3, 79.99, 239.97],  
    ['2024-01-18', 'Monitor', 1, 299.99, 299.99],  
    ['2024-01-19', 'Headphones', 4, 149.99, 599.96]  
]
```

1

CLINE:

“I have a CSV for some sales data. Make a simple ETL and data analysis programme for it to show plots and graphs.”

IN DEMO: I will add...

[“There is already a folder called `sales_etl_analysis` - please ignore and start afresh”]

Summary

- It has been 3 years since ChatGPT was released.
- The 6 months from arranging this talk to doing it is a very long time in AI with great changes in that time.
- The next 6 months/1 year/2 years?

Craig West

<https://craig-west.netlify.app/>

<https://evaluating-ai-agents.com/>

<https://github.com/Python-Test-Engineer/earl2025>

