

Recap of Agentic AI Design Patterns

Miguel Otero Pedrido

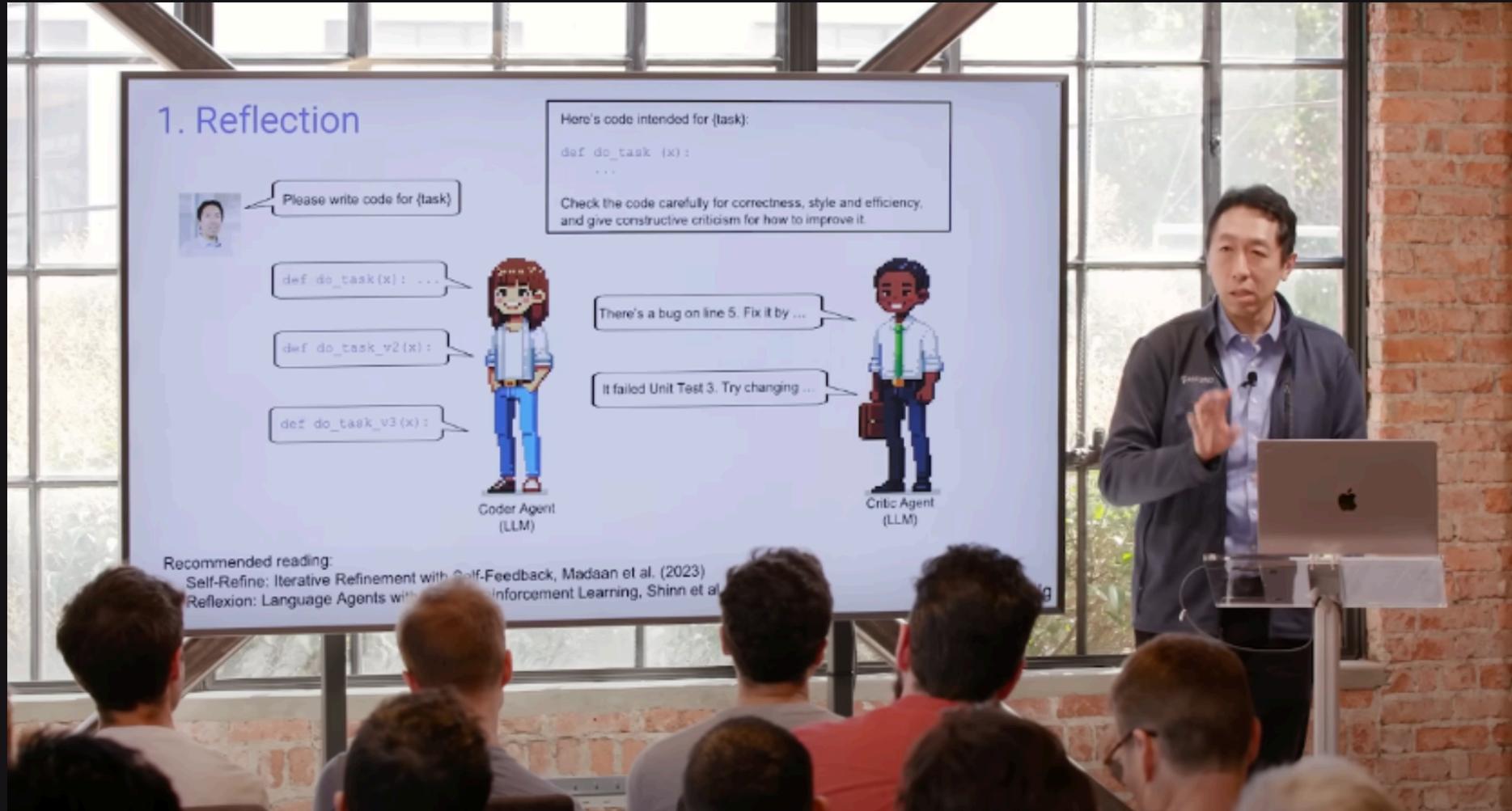
Senior ML / AI Engineer at Dressipi

Founder @ The Neural Maze



Agentic AI Design Patterns

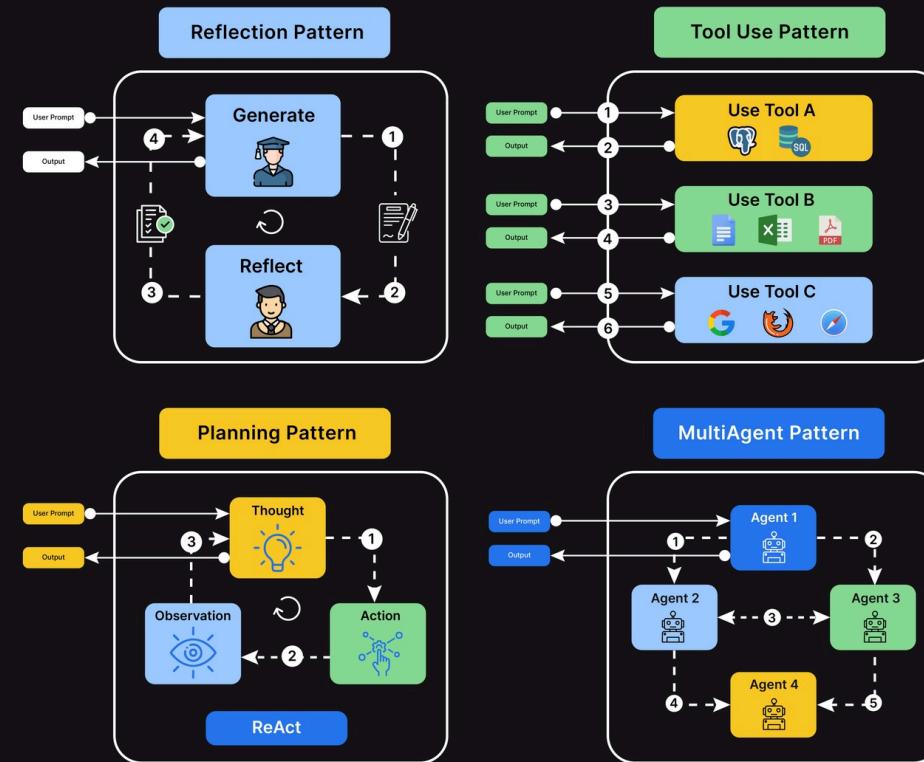
Almost a year ago, Andrew Ng defined four patterns recognizable in any application of LLMs/Agents.



Agentic AI Design Patterns

Patterns are as follows:

- Reflection Pattern
- Tool Use Pattern
- Planning Pattern
- MultiAgent Pattern

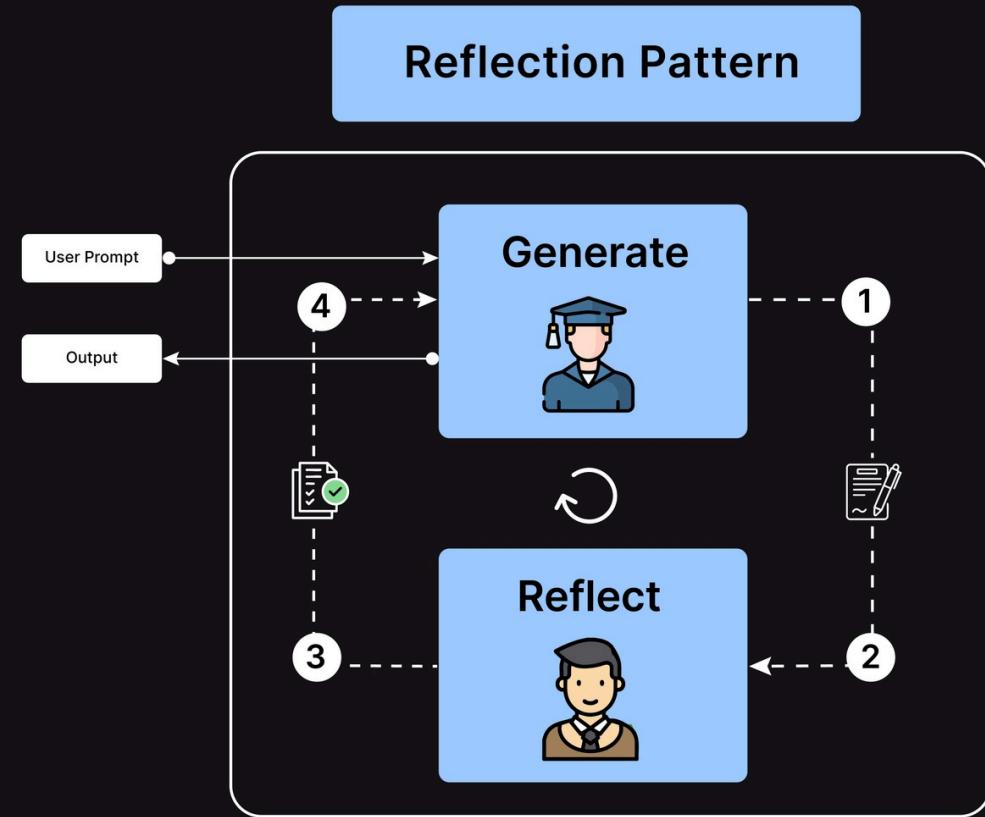


Source: <https://theneuralmaze.substack.com/>

Agentic AI Design Patterns

Reflection Pattern

- A very basic pattern but, despite its simplicity, it provides surprising performance gains for the LLM response.
- It allows the LLM to reflect on its results, suggesting modifications, additions, improvements in the writing style, etc.

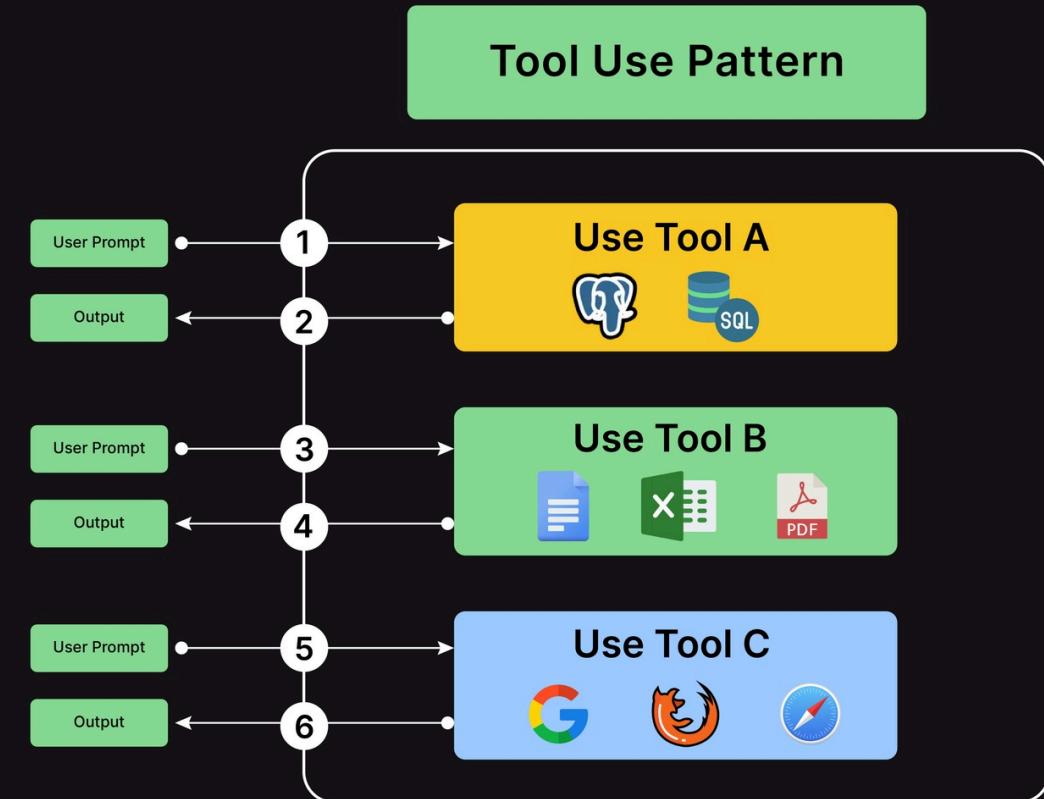


Source: <https://theneuralmaze.substack.com/>

Agentic AI Design Patterns

Tool Use Pattern

- The information stored in the LLM weights is (usually) not enough to give accurate and insightful answers to our questions
- That's why we need to provide the LLM with ways to access the outside world 🌎
- Tools are the way to do this!

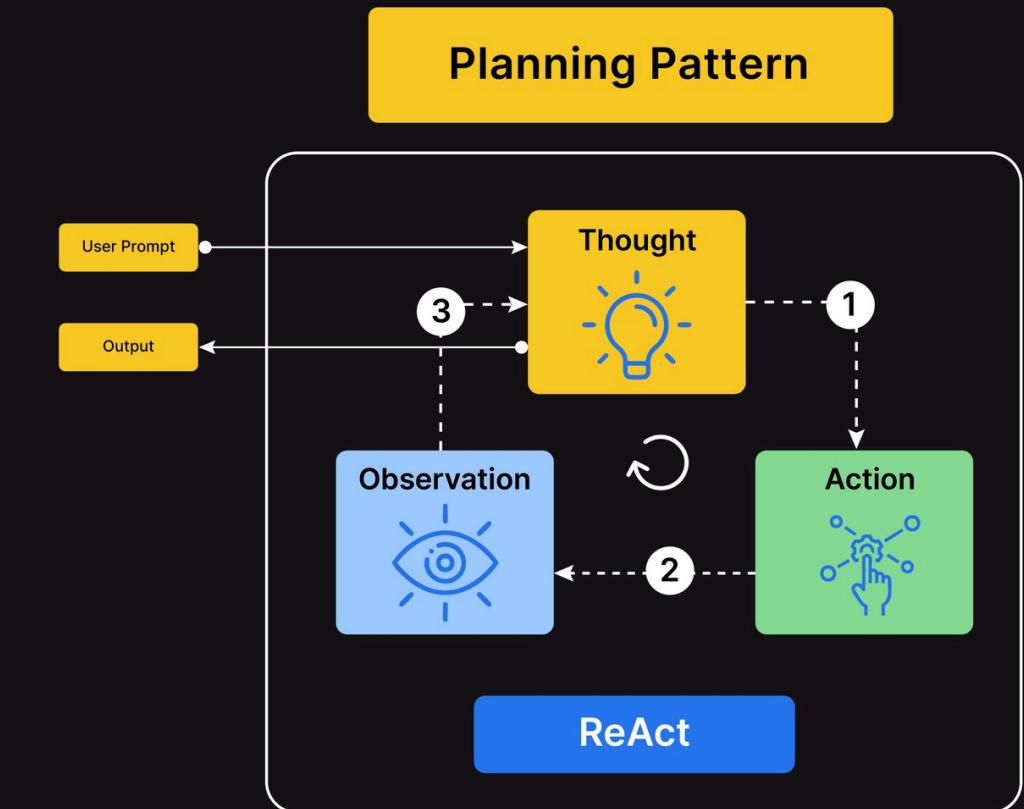


Source: <https://theneuralmaze.substack.com/>

Agentic AI Design Patterns

Planning Pattern

- These patterns allow the agents to decide what sequence of steps to follow to accomplish a large task
- The most paradigmatic example of the planning pattern is the ReAct technique

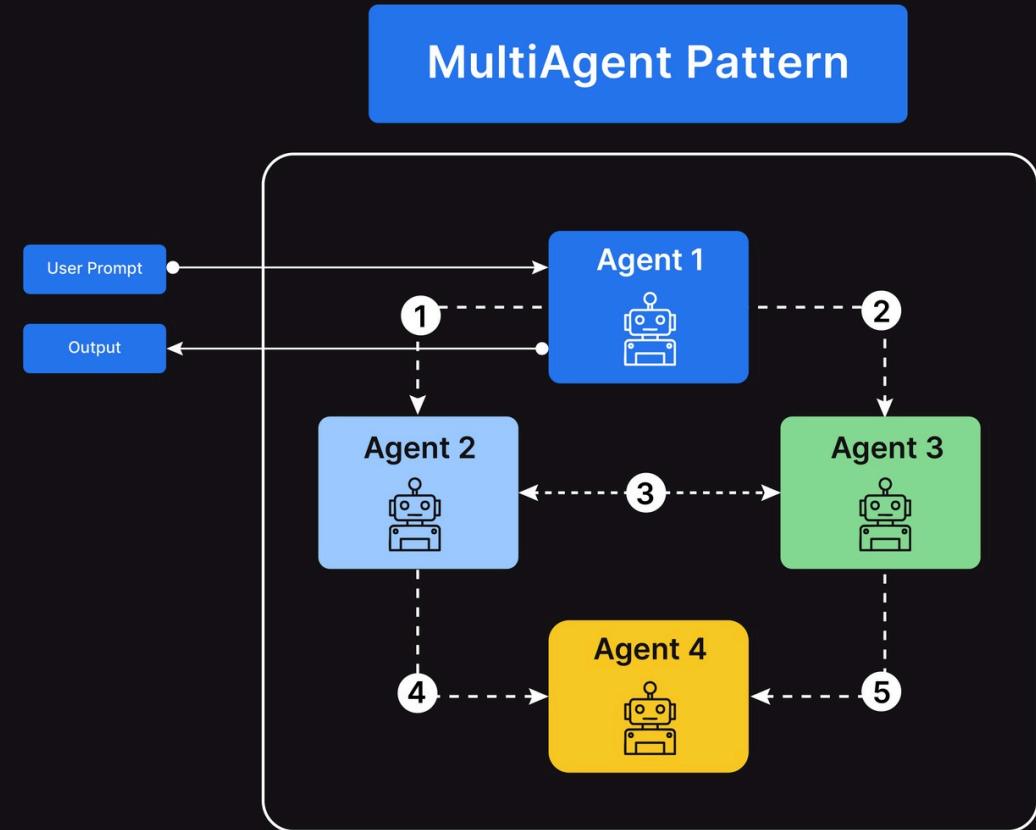


Source: <https://theneuralmaze.substack.com>

Agentic AI Design Patterns

MultiAgent Pattern

- Frameworks like crewAI and AutoGen enable multi-agent applications.
- They implement variations of the multi-agent pattern.
- Tasks are divided into smaller subtasks.
- Agents handle subtasks based on specific roles
e.g. one agent can be a software engineer, another a project manager, etc.



Source: <https://theneuralmaze.substack.com/>

Thanks