Analytics Vidhya

Overview of Design Patterns

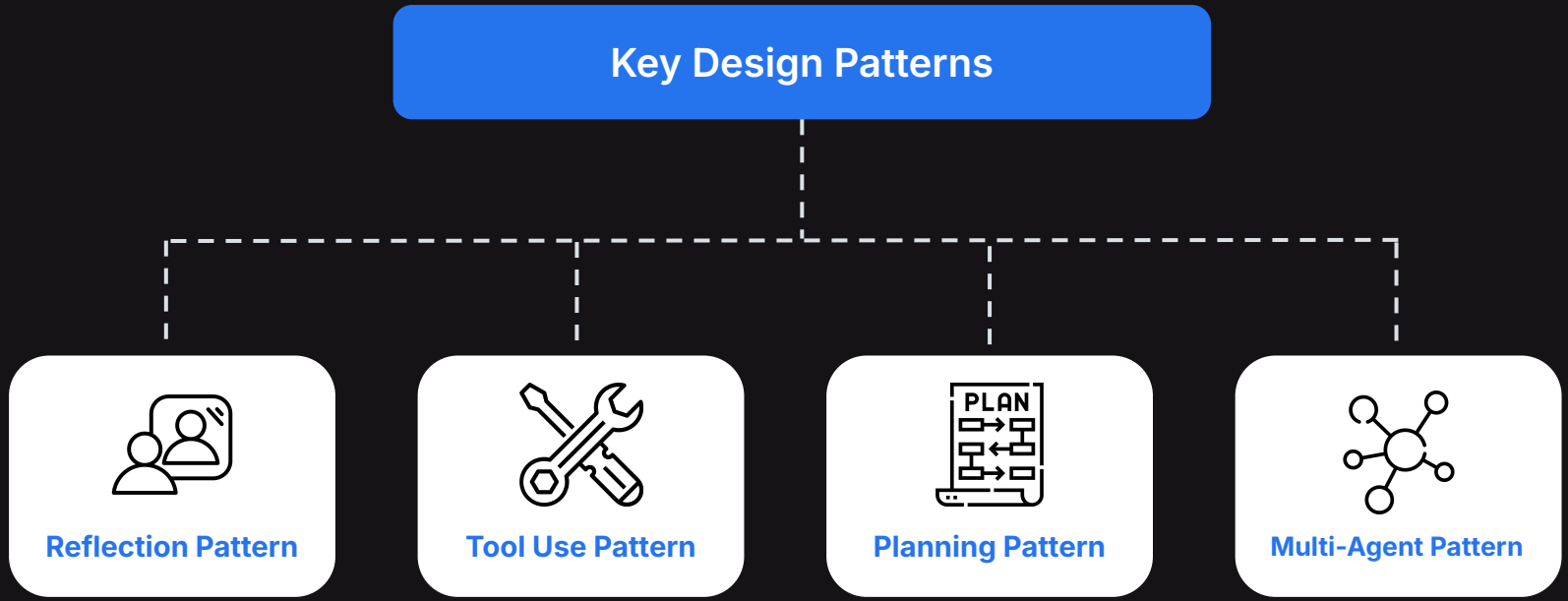
Created by:
Eleni Verteouri
Gen AI Tech Lead @ UBS

Created & Narrated by:
Dipanjan Sarkar
Head of Community & Principal AI Scientist @ Analytics Vidhya
Google Developer Expert - ML & Cloud Champion Innovator
Published Author

Key Design Patterns for Agentic AI

Almost a year ago, Andrew Ng defined four design patterns recognizable in Agentic AI Systems

Key Design Patterns for Agentic AI

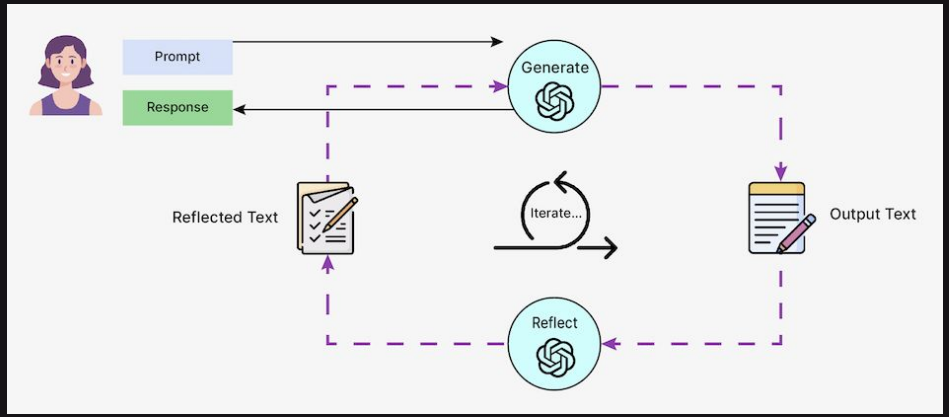


The Reflection Pattern

The reflection pattern enables AI systems to alternate between generate and critique for iterative improvement of the generated response.

Core components:

- Generation phase
- Reflection phase
- Iteration cycles
- Response generation



Benefits:

- Iterative improvement
- Enhanced adaptability to complex tasks

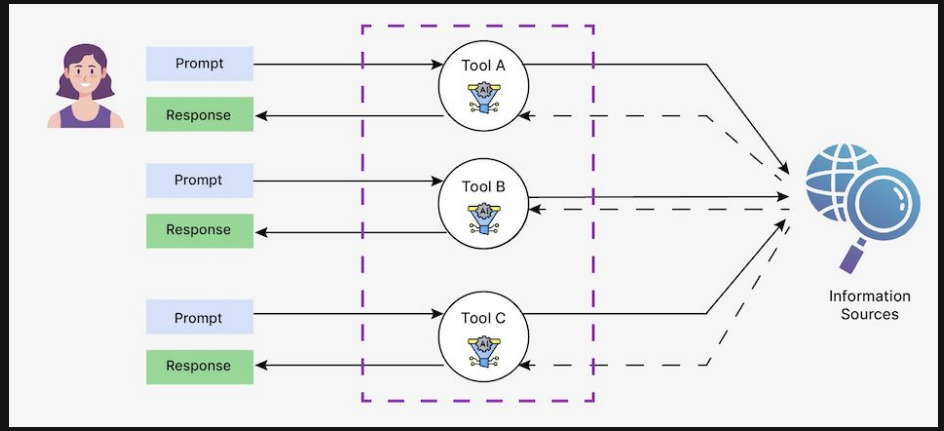
Analytics Vidhya

The Tool Use Pattern

The tool use pattern empowers AI agents to interact with external tools, APIs, and resources for enhanced functionality and additional information to aid its reasoning.

Core components:

- Tool repository
- Tool calling
- Feedback integration
- Response generation



Benefits:

- Real-time data & knowledge integration
- Overcome static knowledge limitations

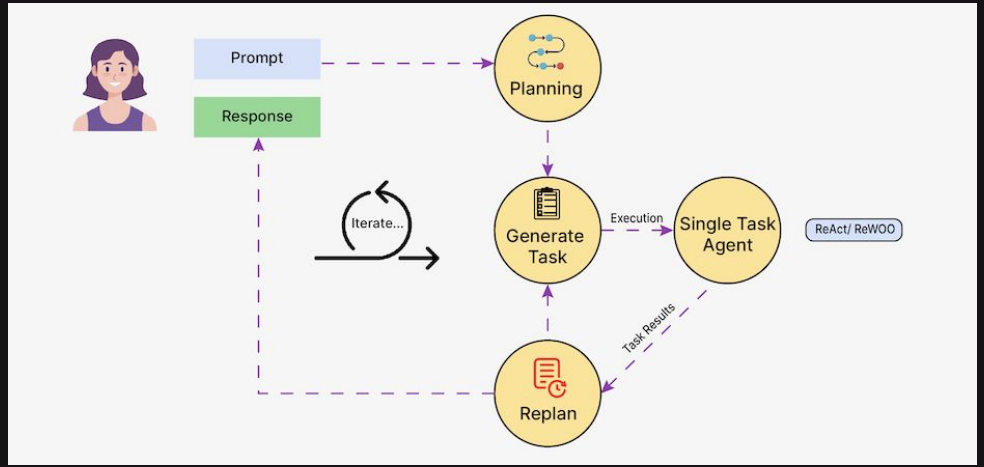
Analytics Vidhya

The Planning Pattern

The planning pattern structures and executes multi-step tasks through reasoning, acting and iterative adjustment.

Core components:

- Task planning
- Task execution
- Replanning
- Response generation



Benefits:

- Dynamic adaptability
- Optimized task execution
- Handle complex multi-step tasks

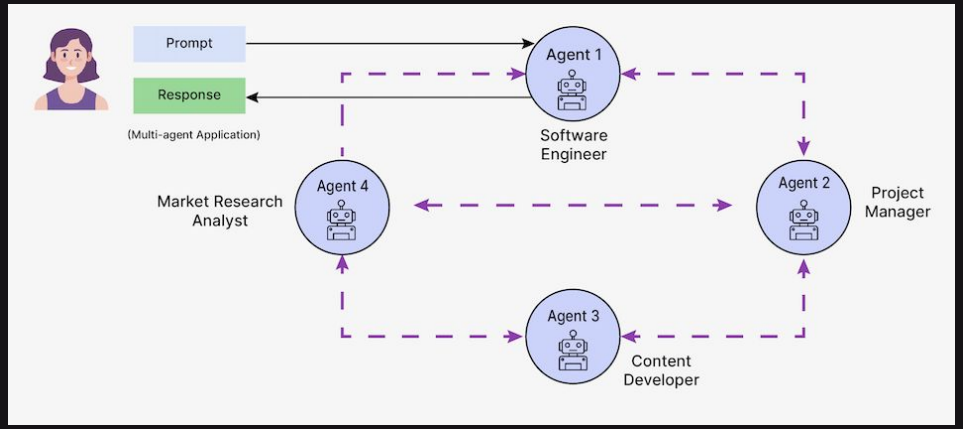
Analytics Vidhya

The Multi-Agent Pattern

Enables multiple AI agents to solve complex problems through communication and coordination.

Core components:

- Multi-Agent architecture (Supervisor, Collaborative etc.)
- Agent specific task allocation
- Task execution
- Response generation



Benefits:

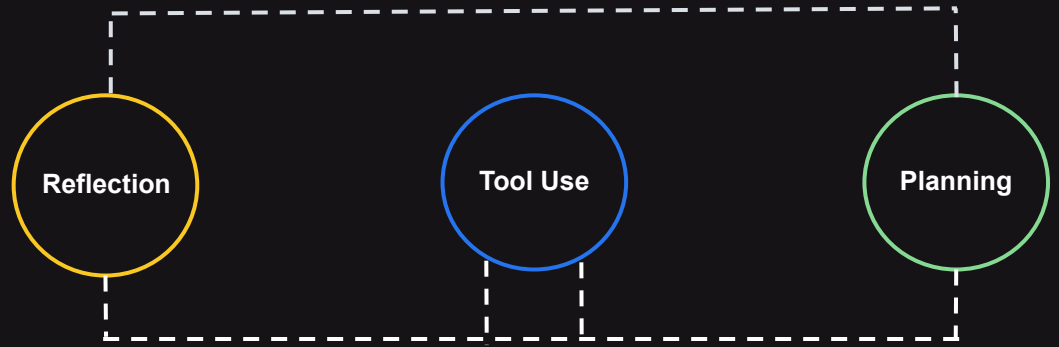
- Scalability through distributed problem solving
- Fault tolerance with decentralized control

Analytics Vidhya

How do Design Patterns Work Together?

Reflection ensures that plans are dynamically refined and optimized. It strengthens the adaptability of planning by incorporating a feedback loop. Tool Use is a basic requirement of any Agentic AI System.

Reflection benefits from evaluating and critiquing generated outputs to improve them.

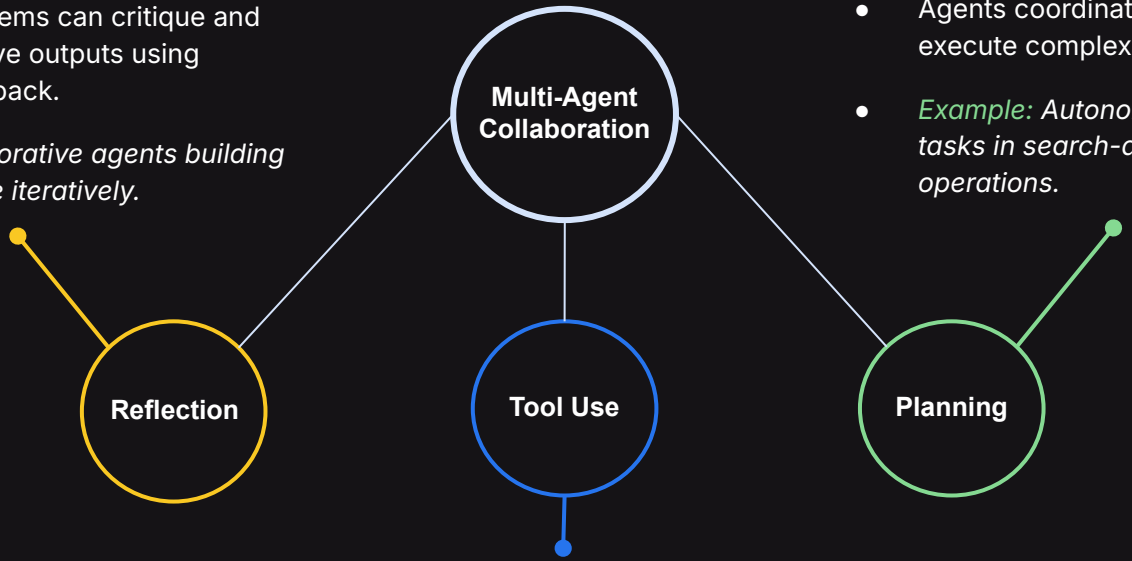


Planning often relies on external tools to execute subtasks or gather real-time data.

Tool use expands the functionality of agents.

Multi-Agent Collaboration with Other Patterns

- Multi-agent systems can critique and improve collective outputs using distributed feedback.
- *Example: Collaborative agents building and testing code iteratively.*



- Agents coordinate to decompose and execute complex task collaboratively.
- *Example: Autonomous drones sharing tasks in search-and-rescue operations.*

- Multi-agent systems often rely on specialized agents that access and use tools collaboratively.
- *Example: A team of e-commerce bots where one agent check inventory while others updates stock and delivery status.*

Analytics Vidhya

Thanks!

Analytics Vidhya