TECHNISCHE UNIVERSITÄT DRESDEN

DRESDEN concept

Faculty of Environmental Sciences
Chair of Geoinformatics

Agents for Conversational Geodata Search

AGILE 2025 Tutorial, Dresden, Germany

Simeon Wetzel, Auriol Degbelo, Stephan Mäs

Agenda

- Key features of an Agent
- Introduction to the 🤗 smolagents Framework

Why Agents

- LLMs only are not enough to solve complex problems
- They are complemented with different **tools**
- Agents
  - … take over the control logic within the application
  - … are useful to automate complex workflows (keyword: automated orchestration)

What is an Agent?

Definition of an Agent

A hardware or software-based computer system that exhibits the following characteristics:

- Autonomy
- Social ability
- Reactivity
- Proactivity

[Wooldridge and Jennings 1995]

Features of an agent

- **Autonomy**: agents operate without the direct intervention of humans, and have some kind of control over their actions and internal state.

- **Social ability**: Agents interact with other agents (and possibly with humans) using some form of agent communication language.

- **Reactivity**: Agents perceive their environment - whether it's the physical world, a user via a graphical interface, a collection of other agents, the Internet, or a combination of these - and respond to changes.

- **Proactivity**: Agents do not merely react; they exhibit goal-directed behavior by taking independent initiative.

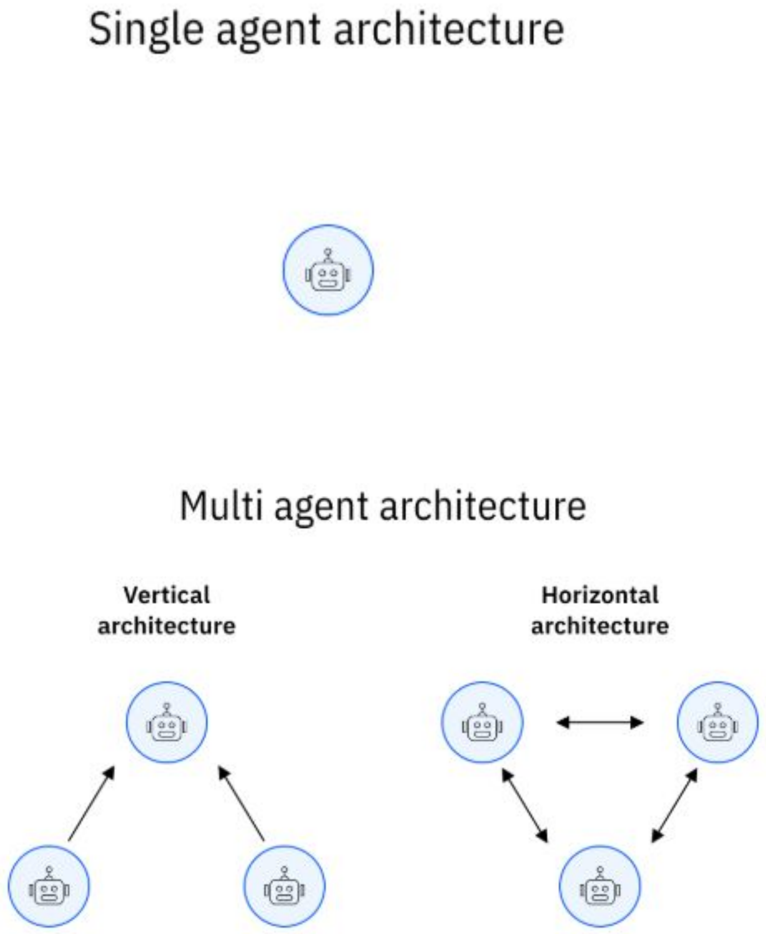
[Wooldridge and Jennings 1995]

Features of an agent (mentioned occasionally)...

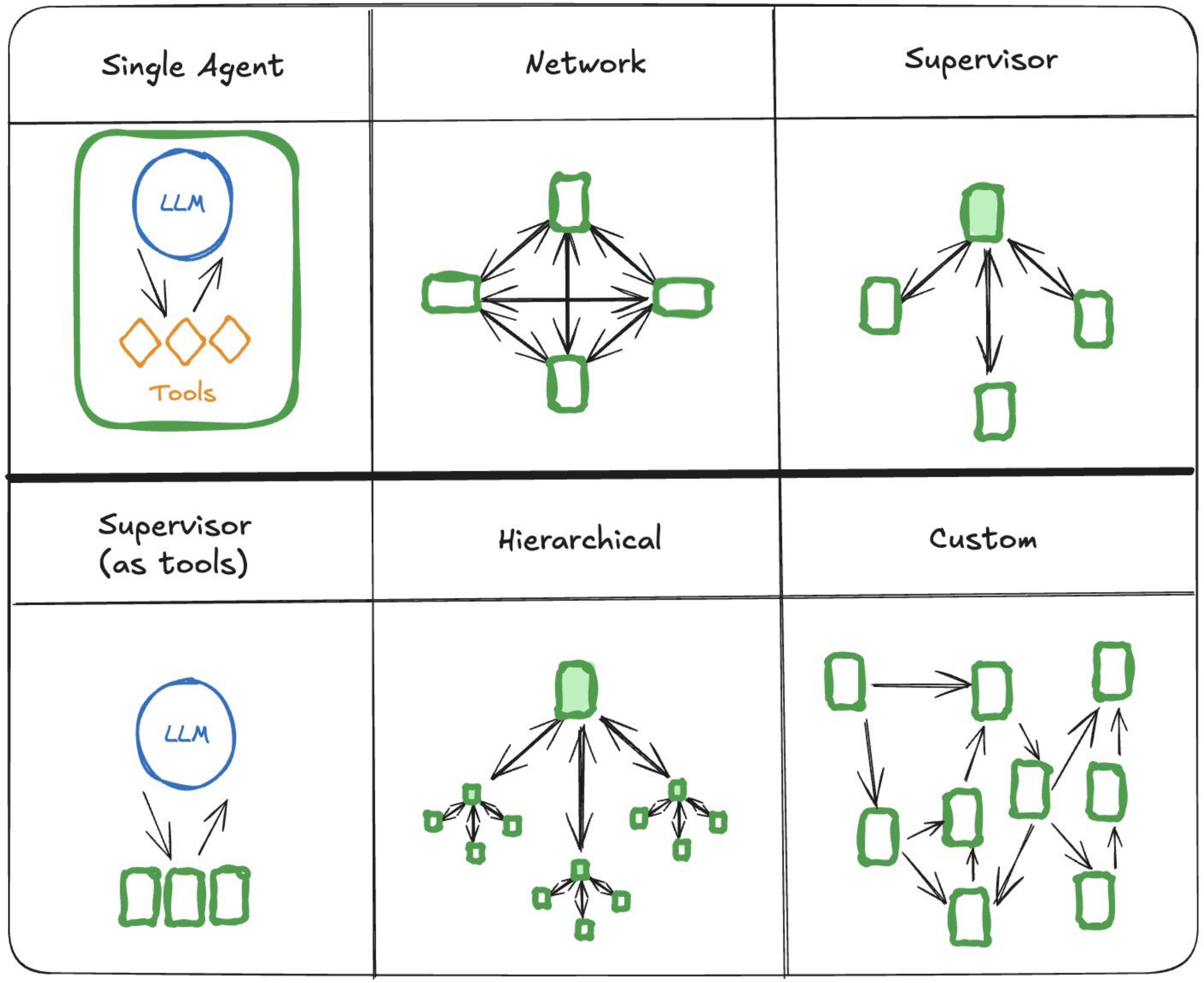
- **Mobility**: The ability of an agent to move within an (electronic) network.
- **Veracity**: The assumption that an agent will not knowingly communicate false information.
- **Benevolence**: The assumption that agents do not pursue conflicting goals and therefore always strive to do what is required of them.
- **Rationality**: (crude) assumption that an agent will act in order to achieve its goals, an will not act in such a way as to prevent its goals being achieved - at least within the limits of its beliefs.

[Wooldridge and Jennings 1995]

Agent-based architectures

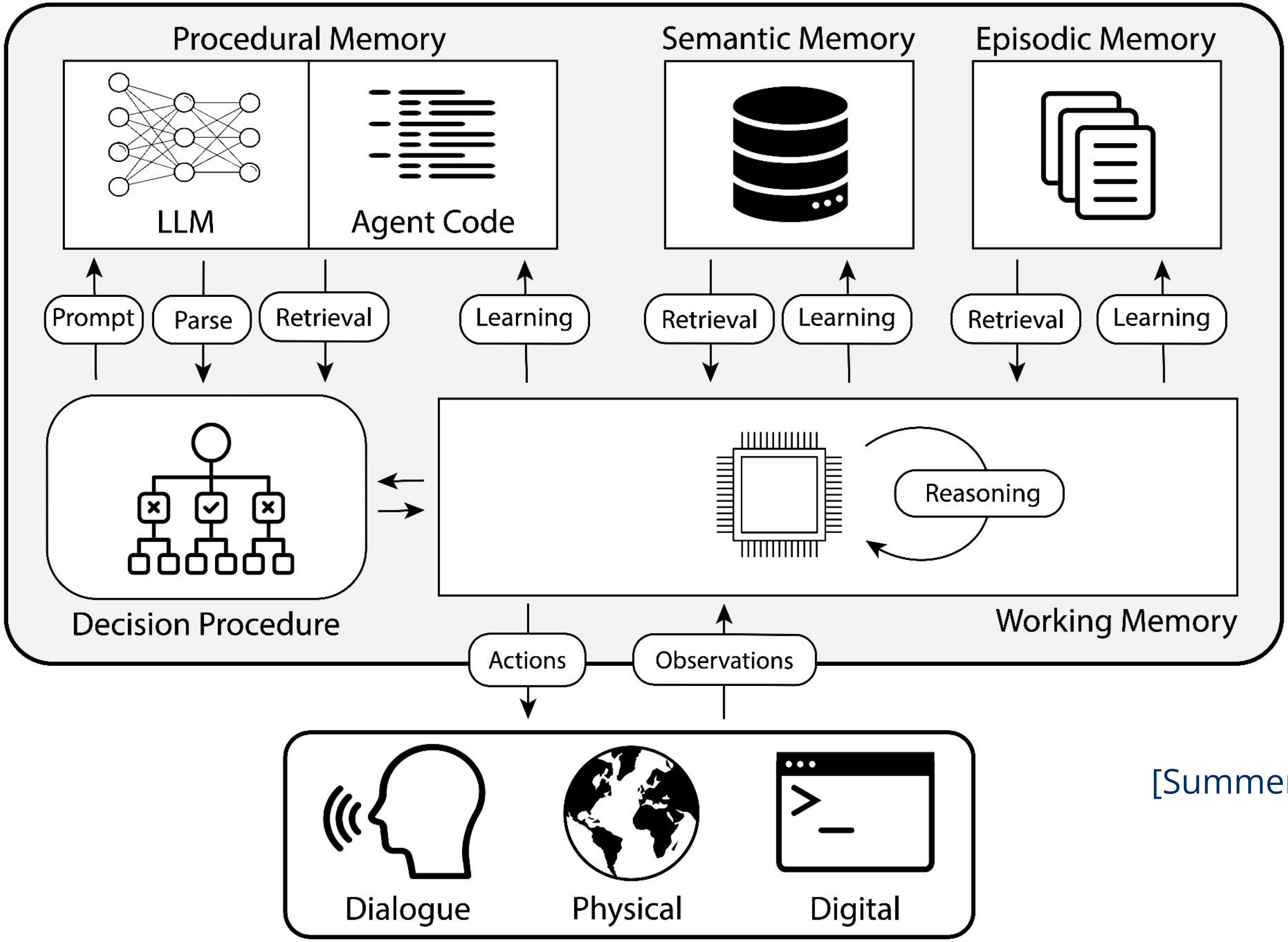


https://www.ibm.com/think/topics/ai-agent-types



https://langchain-ai.github.io/langgraph/concepts/multi_agent/

Architecture: single agent



[Summers et al 2024]

Key operations

- **Retrieval**: read information from long-term memory into working memory.
- **Reasoning**: process content from working memory to generate new information
- **Learning**: write information to long-term memory (e.g. update episodic memory with experience, update semantic memory with knowledge, update LLM parameters)

[Summers et al 2024]

Types of memories

Short-term

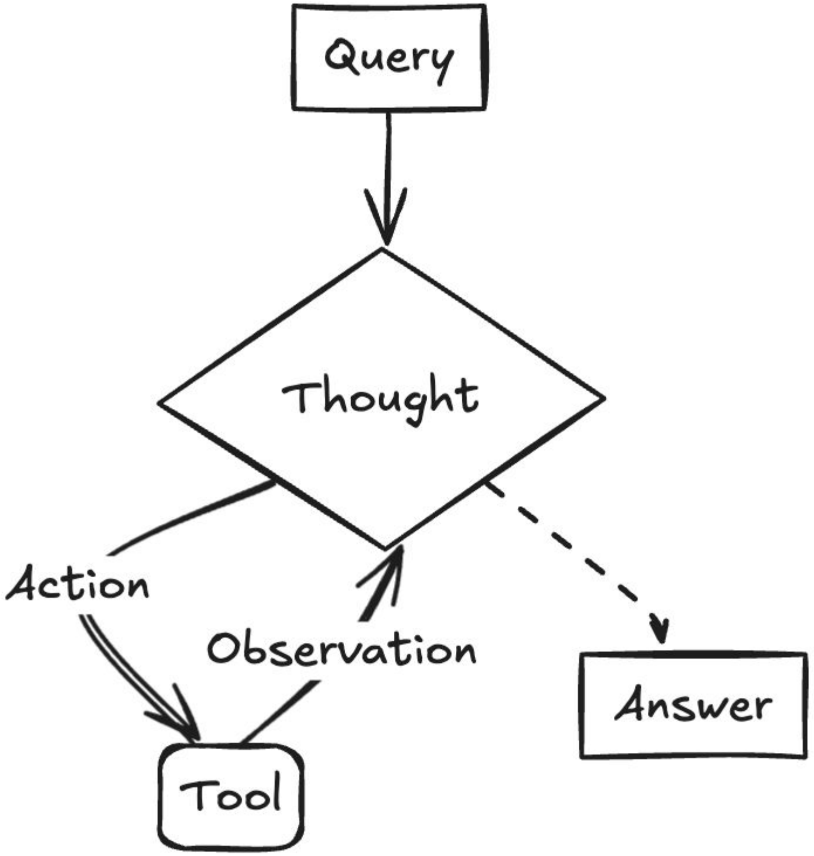
- **Working Memory**: maintains active and immediately available information for the current decision cycle, e.g. perception inputs or information from the previous decision cycle.

Long-term

- **Episodic Memory**: stores experiences from previous decision cycles, e.g. training data in the form of input-output pairs, sequences of historical events.
- **Semantic Memory:** stores an agent's knowledge about the world and about itself.
- **Procedural Memory**: this has two forms: *Implicit* knowledge, stored in the weights of the language model; *explicit* knowledge, stored in the agent's code. The agent's code can be further subdivided into: *procedures* that implement actions (reasoning, retrieval, grounding, and learning) and *procedures* that implement decision-making.

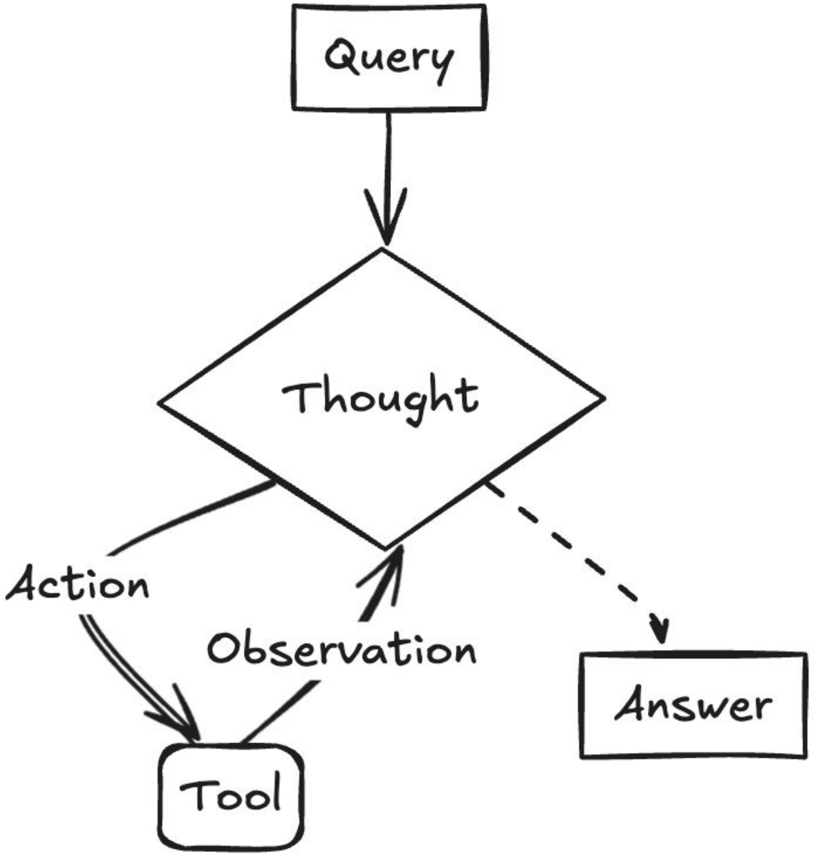
[Summers et al 2024]

ReAct: Reasoning + Act [Yao et al 2023]



https://www.philschmid.de/langgraph-gemini-2-5-react-agent

ReAct: Thought

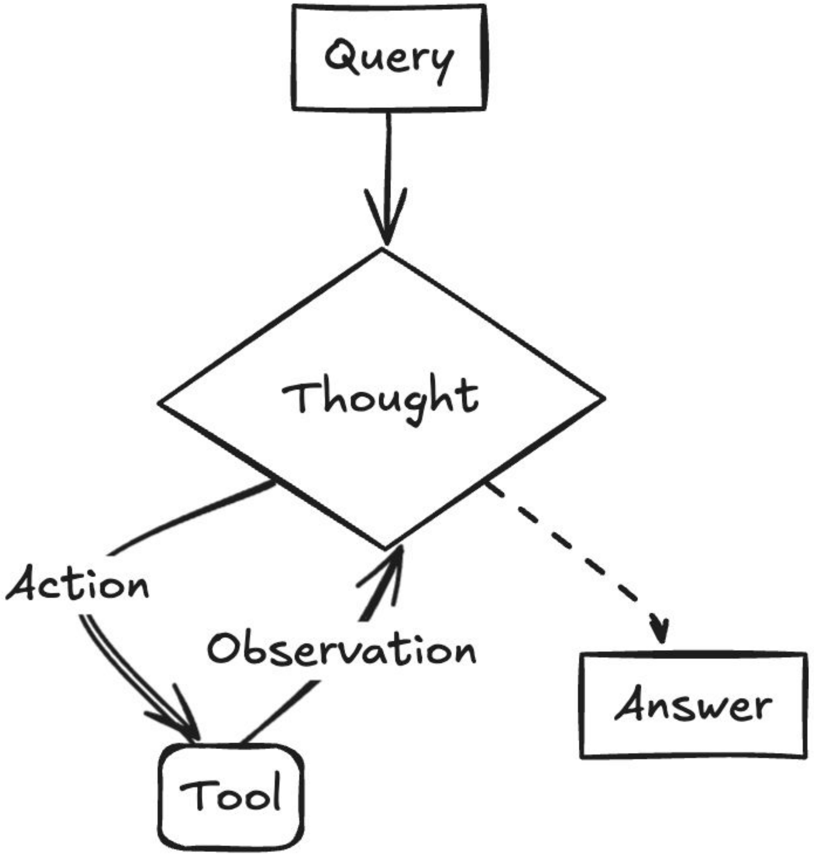


"Thoughts represent the **Agent's internal reasoning and planning processes** to solve the task".

Type of Thought	Example
Planning	"I need to break this task into three steps: 1) gather data, 2) analyze trends, 3) generate report"
Analysis	"Based on the error message, the issue appears to be with the database connection parameters"
Decision Making	"Given the user's budget constraints, I should recommend the mid-tier option"
Problem Solving	"To optimize this code, I should first profile it to identify bottlenecks"
Memory Integration	"The user mentioned their preference for Python earlier, so I'll provide examples in Python"
Self-Reflection	"My last approach didn't work well, I should try a different strategy"
Goal Setting	"To complete this task, I need to first establish the acceptance criteria"
Prioritization	"The security vulnerability should be addressed before adding new features"

https://huggingface.co/learn/agents-course/unit1/thoughts

ReAct: Action



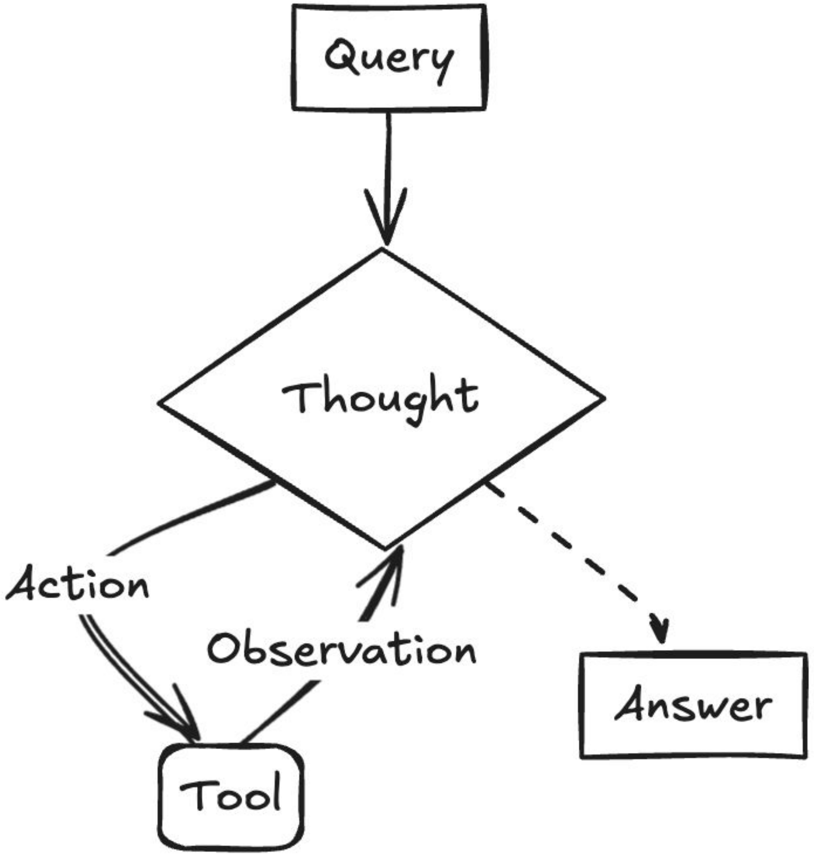
“Actions are the concrete steps an **AI agent takes to interact with its environment”**.

Type of Agent	Description
JSON Agent	The Action to take is specified in JSON format.
Code Agent	The Agent writes a code block that is interpreted externally.
Function-calling Agent	It is a subcategory of the JSON Agent which has been fine-tuned to generate a new message for each action.

Type of Action	Description
Information Gathering	Performing web searches, querying databases, or retrieving documents.
Tool Usage	Making API calls, running calculations, and executing code.
Environment Interaction	Manipulating digital interfaces or controlling physical devices.
Communication	Engaging with users via chat or collaborating with other agents.

https://huggingface.co/learn/agents-course/unit1/actions

ReAct: Observation



"Observations are **how an Agent perceives the consequences of its actions**".

Type of Observation	Example
System Feedback	Error messages, success notifications, status codes
Data Changes	Database updates, file system modifications, state changes
Environmental Data	Sensor readings, system metrics, resource usage
Response Analysis	API responses, query results, computation outputs
Time-based Events	Deadlines reached, scheduled tasks completed

https://huggingface.co/learn/agents-course/unit1/observations

What is smolagents?

Smolagents

- Recent framework (released 2025)
- Built by **HuggingFace** developers
- Simple **python framework** for building agents
- Supports **any LLM**
  - from HuggingFace
  - or other Inference Providers (OpenAI, Groq, Anthropic, etc.)
- First-class support for **Code Agents**

Smolagents - Different levels of agency

Agency Level	Description	Short name	Example Code
☆☆☆	LLM output has no impact on program flow	Simple processor	`process_llm_output(llm_response)`
★☆☆	LLM output controls an if/else switch	Router	`if llm_decision(): path_a() else: path_b()`
★★☆	LLM output controls function execution	Tool call	`run_function(llm_chosen_tool, llm_chosen_args)`
★★☆	LLM output controls iteration and program continuation	Multi-step Agent	`while llm_should_continue(): execute_next_step()`
★★★	One agentic workflow can start another agentic workflow	Multi-Agent	`if llm_trigger(): execute_agent()`
★★★	LLM acts in code, can define its own tools / start other agents	Code Agents	`def custom_tool(args): ...`

https://huggingface.co/docs/smolagents/conceptual_guides/intro_agents

Smolagents

CodeAgents: *Executable Code Actions Elicit Better LLM Agents (Wang et al. 2024, DOI)*

Smolagents

- **Composability**: JSON actions cannot be nested or pre-defined for later use
- **Object management**: with JSON it is harder to store outputs of complex objects (e.g. images)
- **Generality**: code is built to express simply anything you can have a computer do
- **Representation in LLM training data**: plenty of quality code actions are already included in LLMs' training data which means they're already trained for this!

References

- Sumers, T.R., Yao, S., Narasimhan, K. and Griffiths, T.L. (2024) 'Cognitive architectures for language agents', *Transactions on Machine Learning Research*. Available at: https://doi.org/10.48550/arXiv.2309.02427.
- Wang, X., Chen, Y., Yuan, L., Zhang, Y., Li, Y., Peng, H., & Ji, H. (2024). Executable Code Actions Elicit Better LLM Agents. Proceedings of Machine Learning Research, 235(Llm), 50208–50232. Available at: https://doi.org/10.48550/arXiv.2402.01030
- Wooldridge, M. and Jennings, N.R. (1995) 'Intelligent agents: theory and practice', The Knowledge Engineering Review, 10(2), pp. 115–152. Available at: https://doi.org/10.1017/S0269888900008122.
- Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K.R. and Cao, Y. (2023) 'ReAct: Synergizing reasoning and acting in language models', in The Eleventh International Conference on Learning Representations (ICLR 2023). Kigali, Rwanda.