



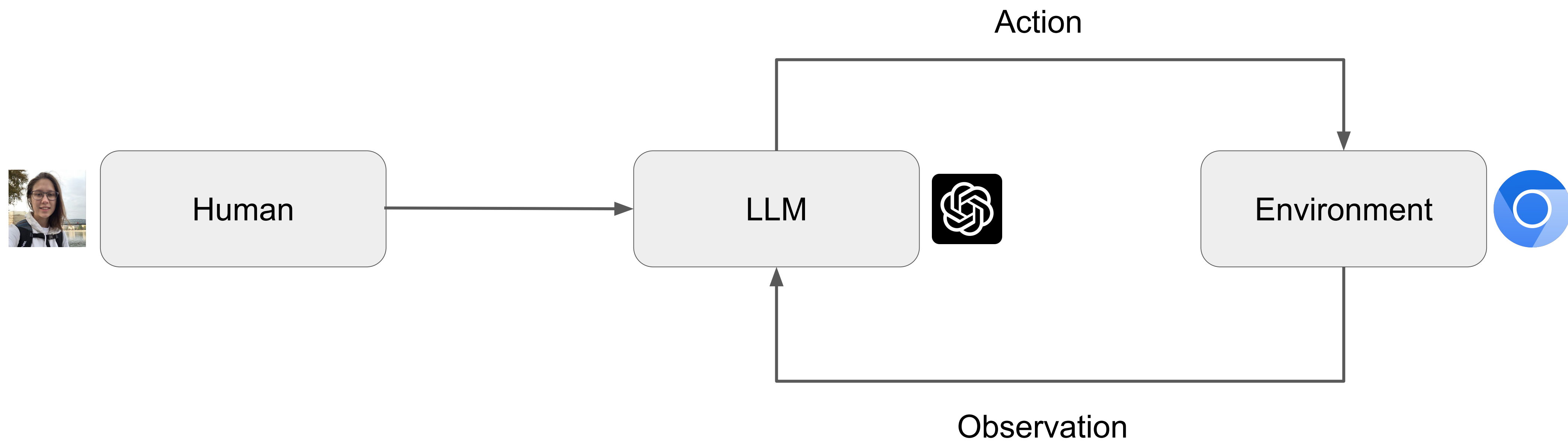
Kickstarting AI Agents Development Using Koog Framework

by Maria Tigina, Developer @ Koog Team, JetBrains

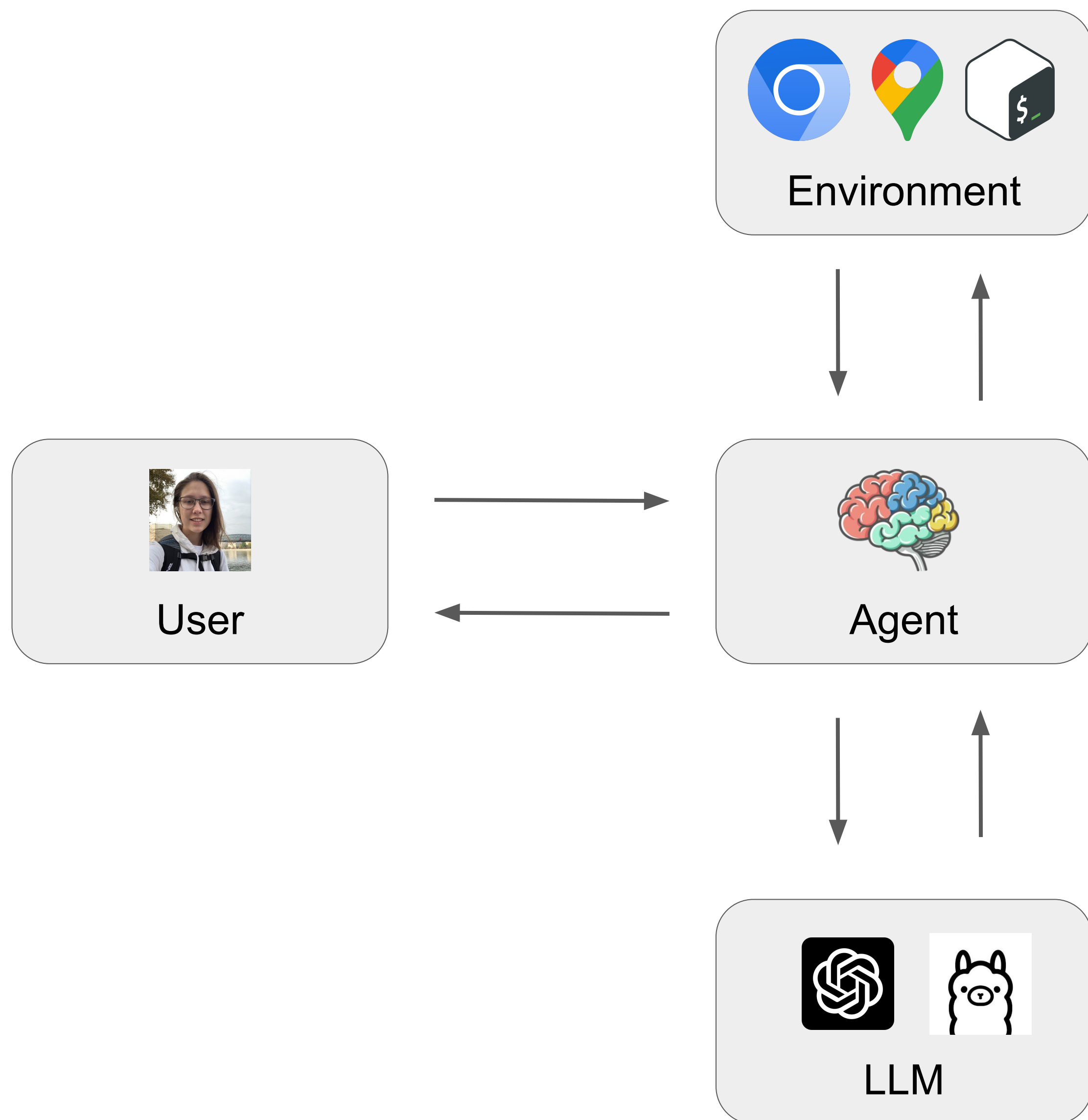
Plan for today

1. AI Agent
2. Koog AI Agent
3. Prompt and PromptExecutor
4. Tools and ToolRegistry
5. Strategy (including singleRunStrategy)
6. Node (including nodeCompressHistory)
7. Subgraph (including subgraphWithTask)

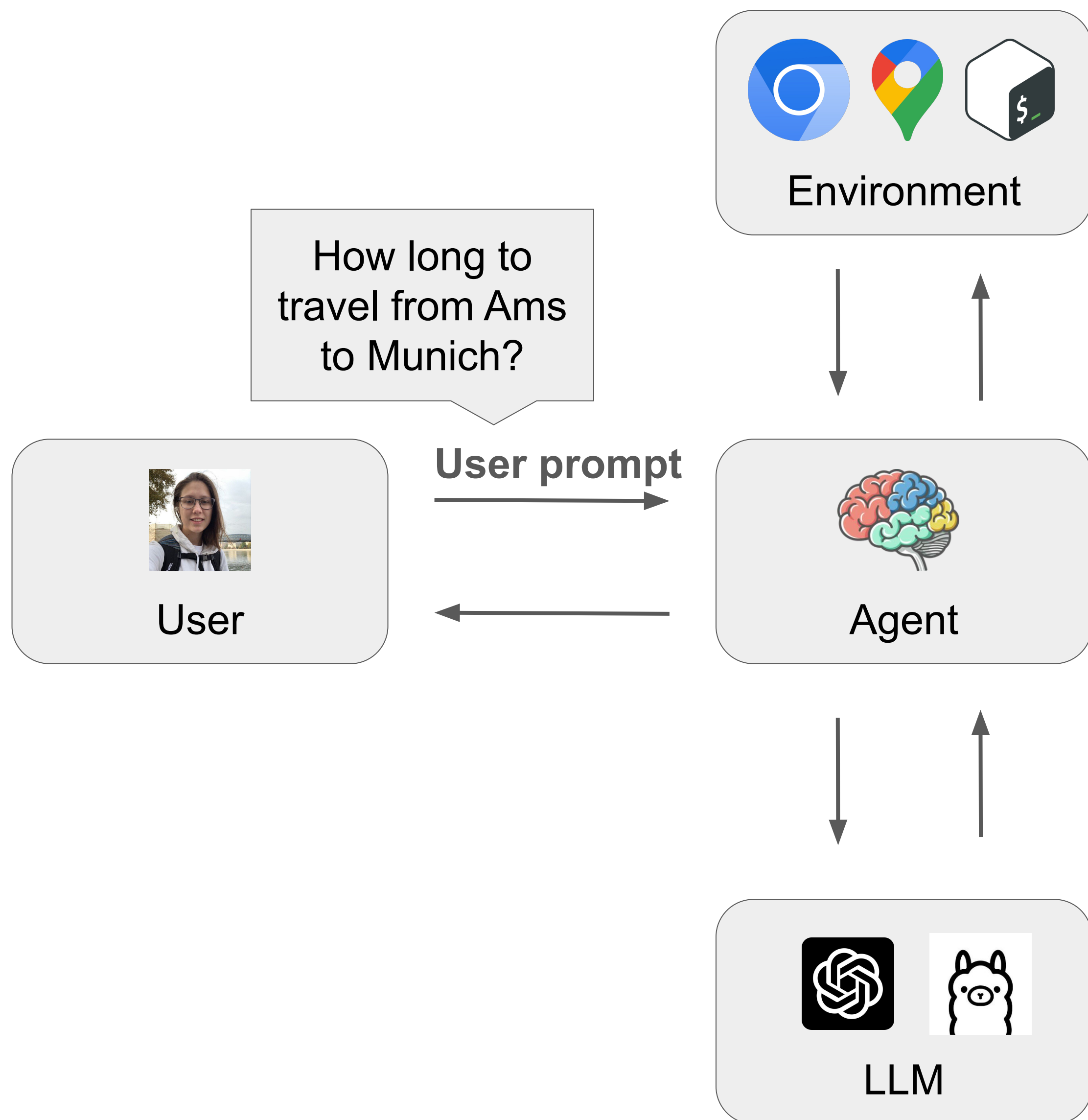
What is an Agent?



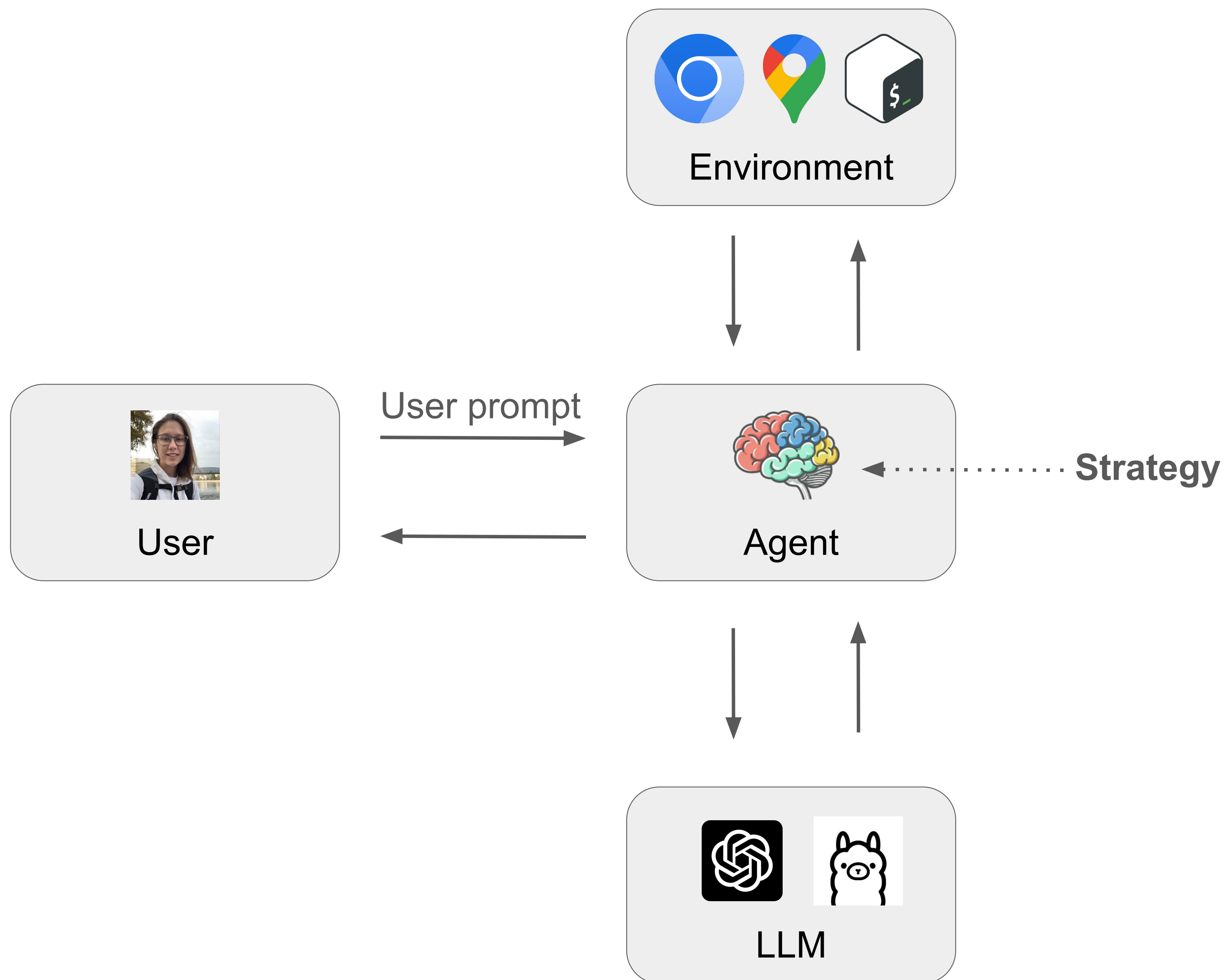
What is an Agent in **Koog**?



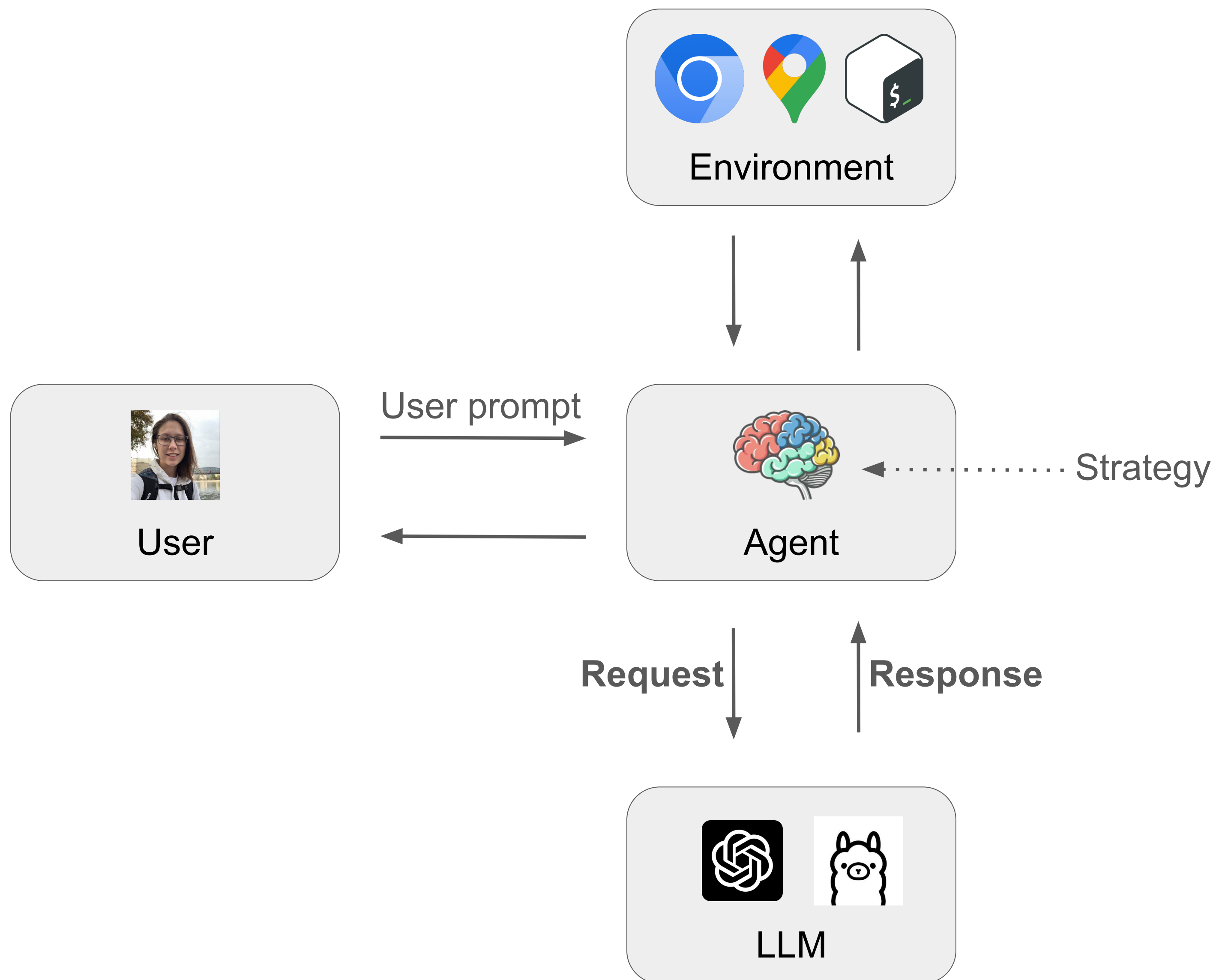
What is an Agent in Koog?



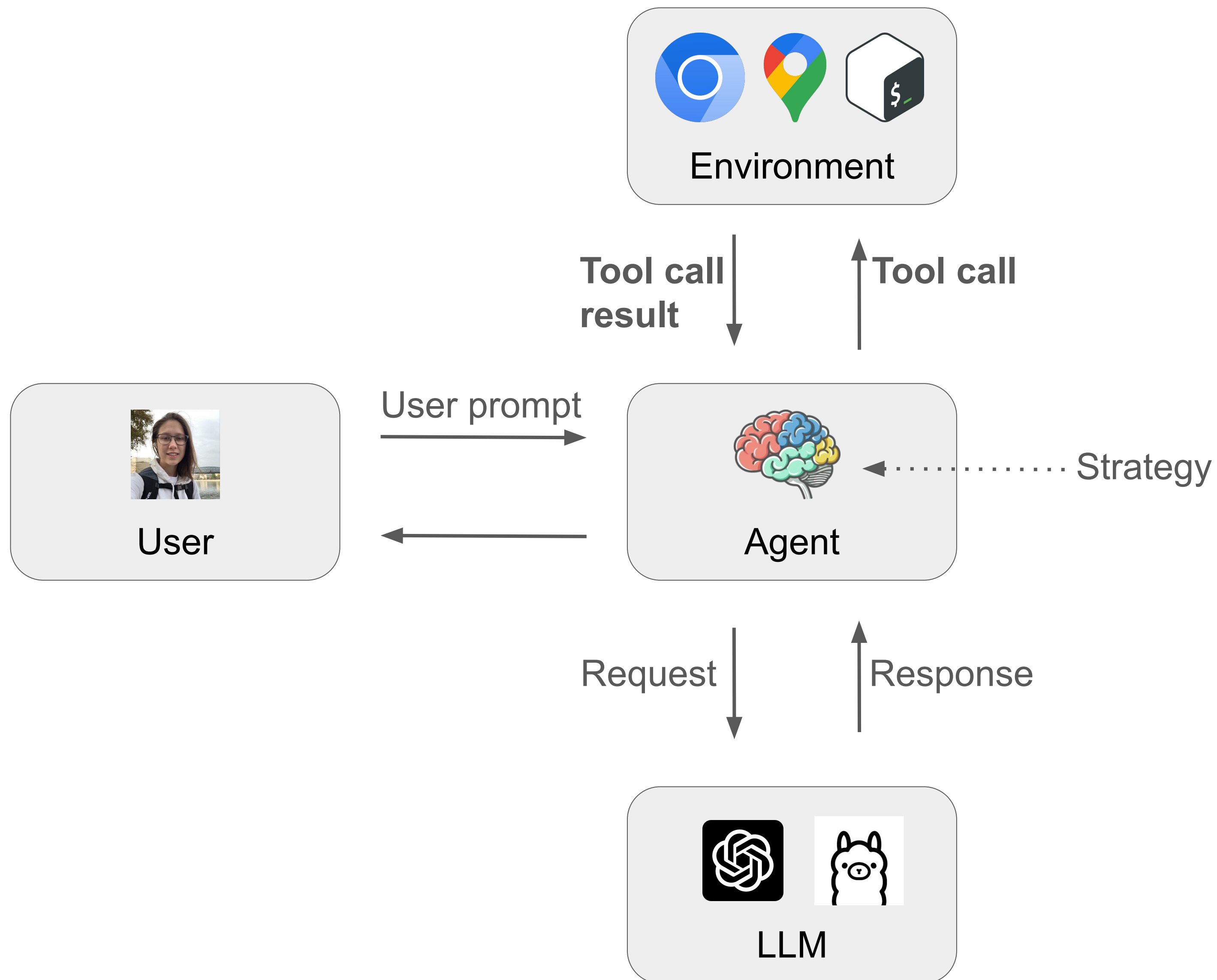
What is an Agent in Koog?



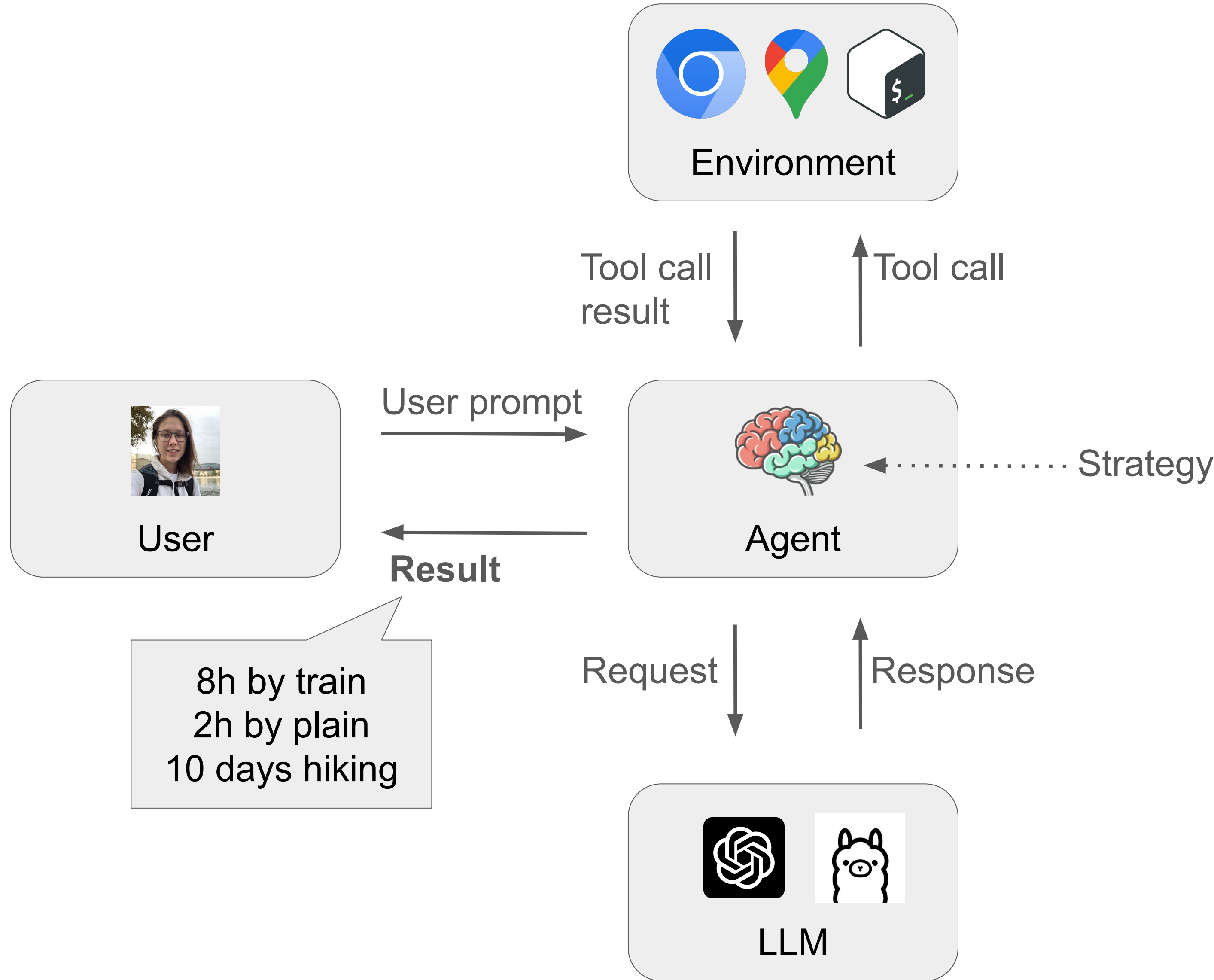
What is an Agent in Koog?



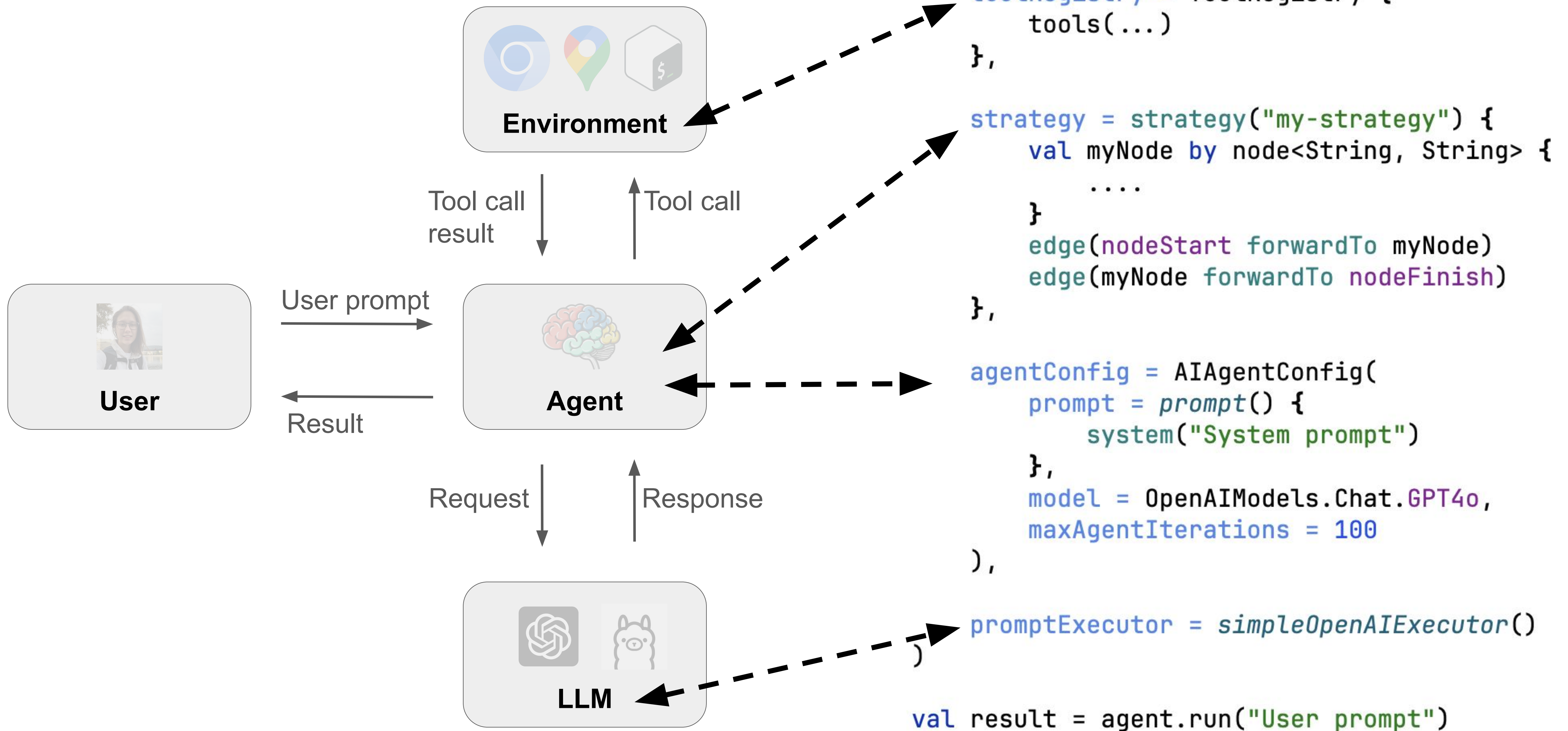
What is an Agent in Koog?



What is an Agent in Koog?



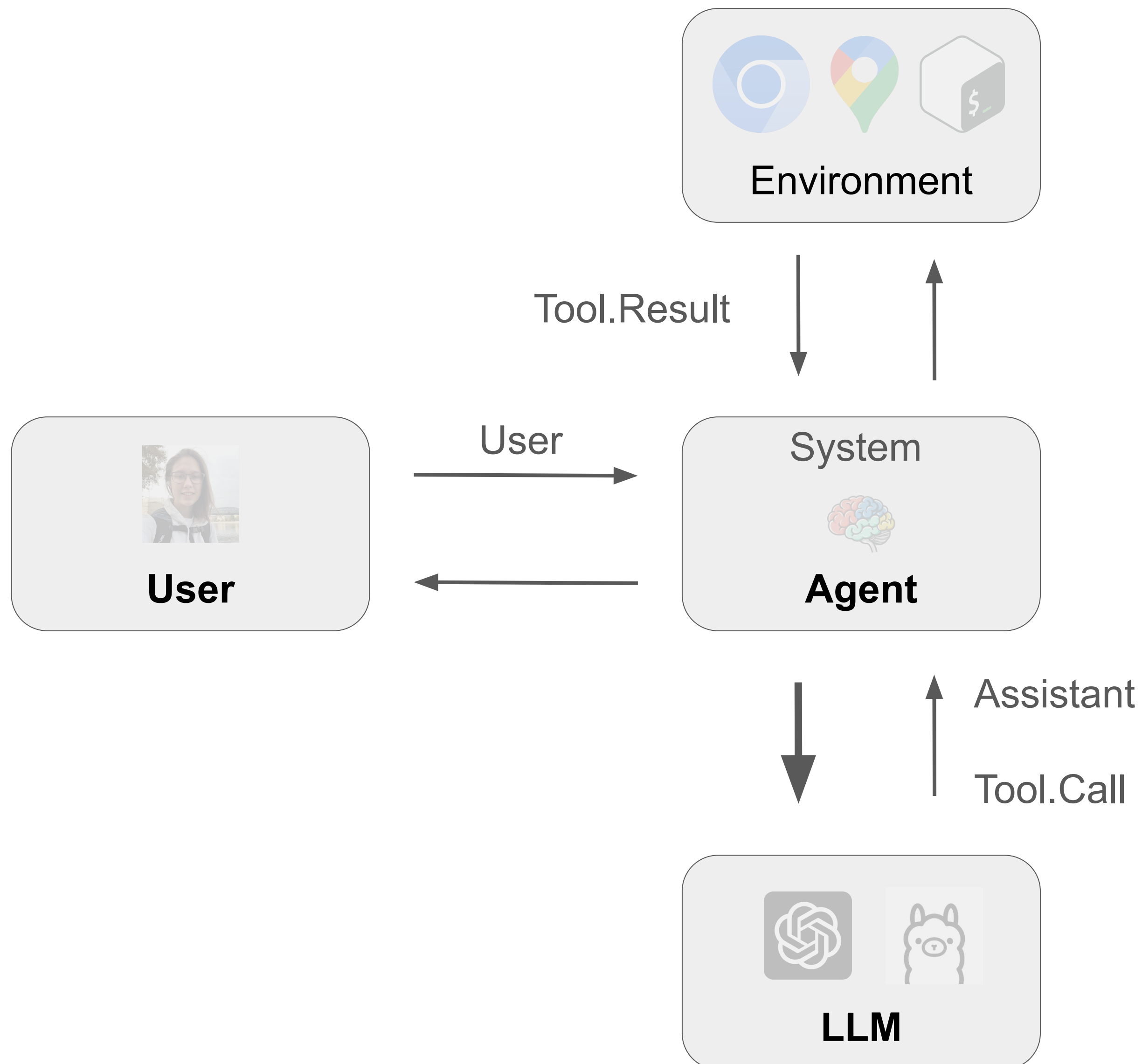
What is an Agent in Koog?



git clone <https://github.com/tiginamaria/koog-workshop>
git checkout intro

export GRAZIE_TOKEN=(get from <https://platform.jetbrains.ai/>)

Prompt



Message types:

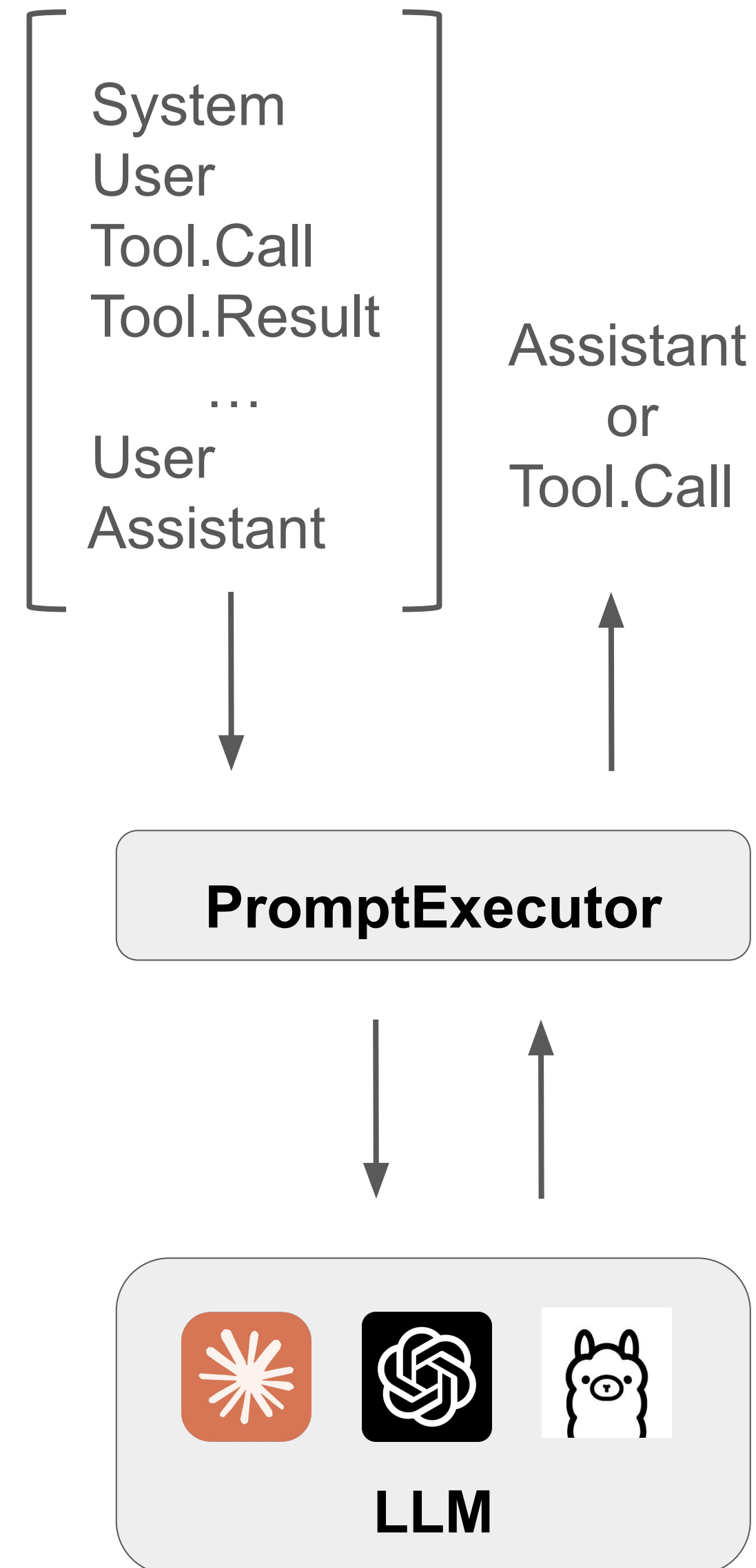
- System – general instructions for agent
- User – user request or additional feedback
- Assistant – text LLM response
- Tool.Call – tool LLM response
- Tool.Result – tool execution result (should follow Tool.Call)

Good to know:

- The whole message history goes to the LLM as context
- You can attach files/images to the user messages

 01-Prompt.kt

PromptExecutor

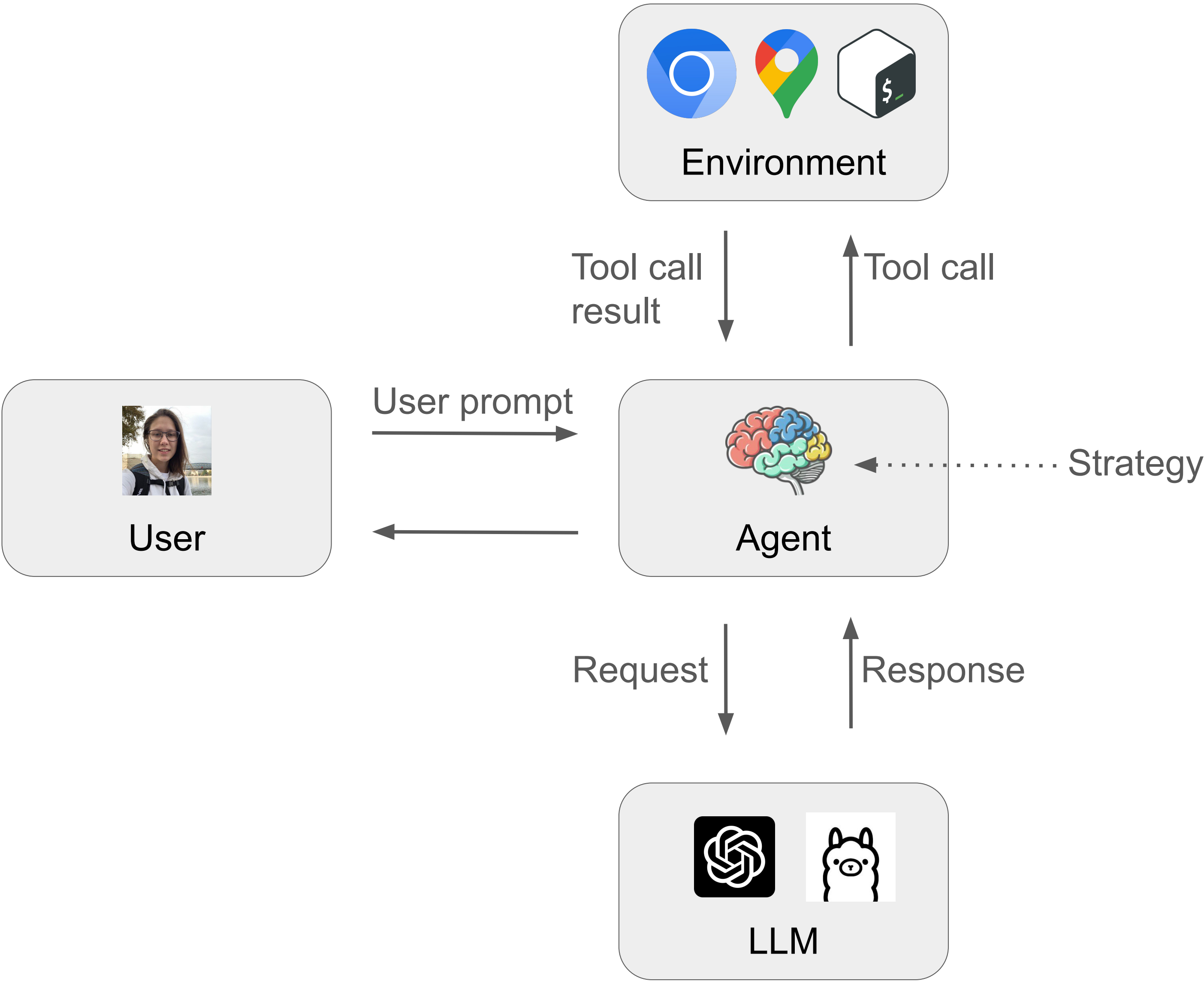


- Sends the list of messages to the LLM, return message (or list of messages) generated by the LLM
- Wraps the LLM clients (providers)
- For every client the messages content and format can differ, executor is made to unify them

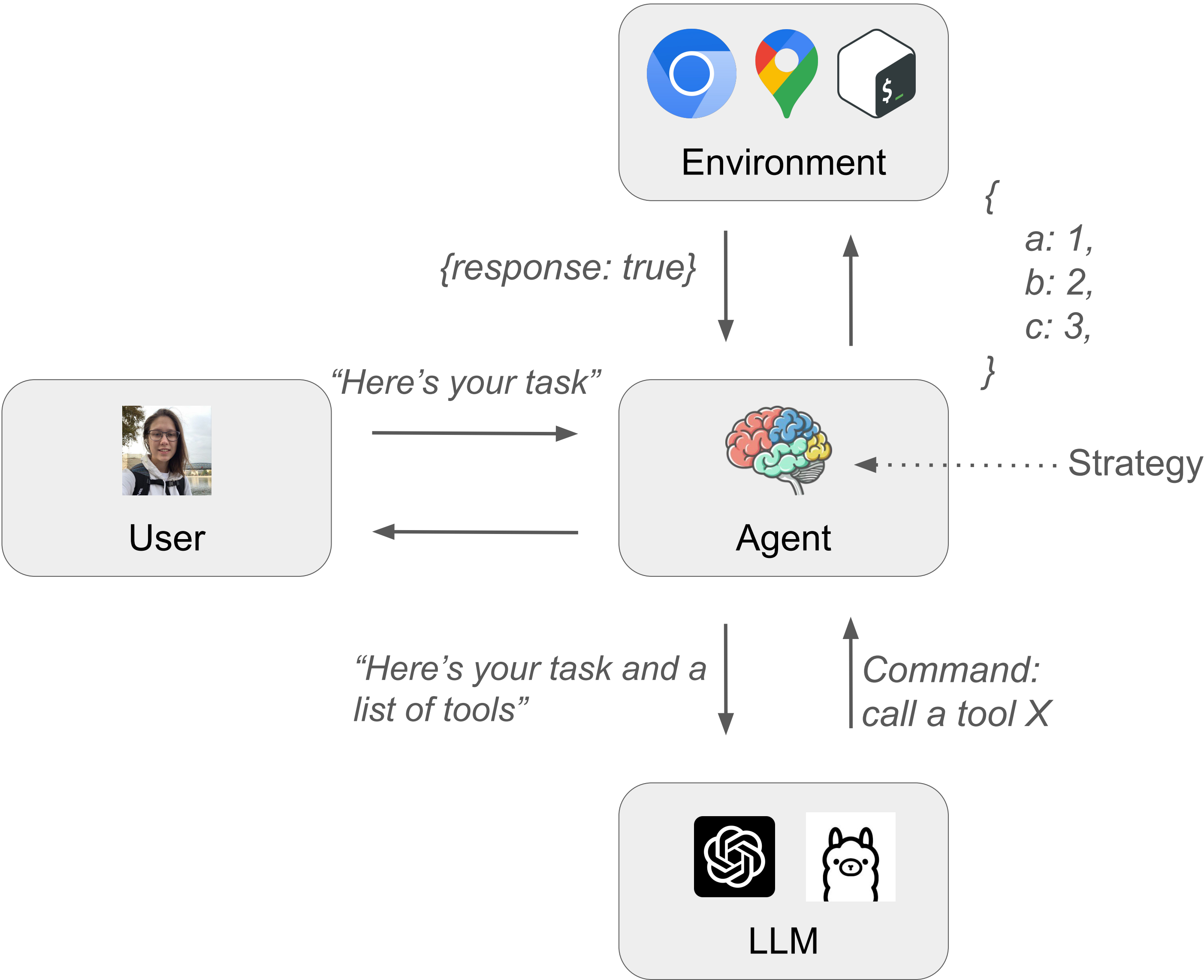
 02-PromptExecutor.kt

 03-StructuredOutput.kt

Tools

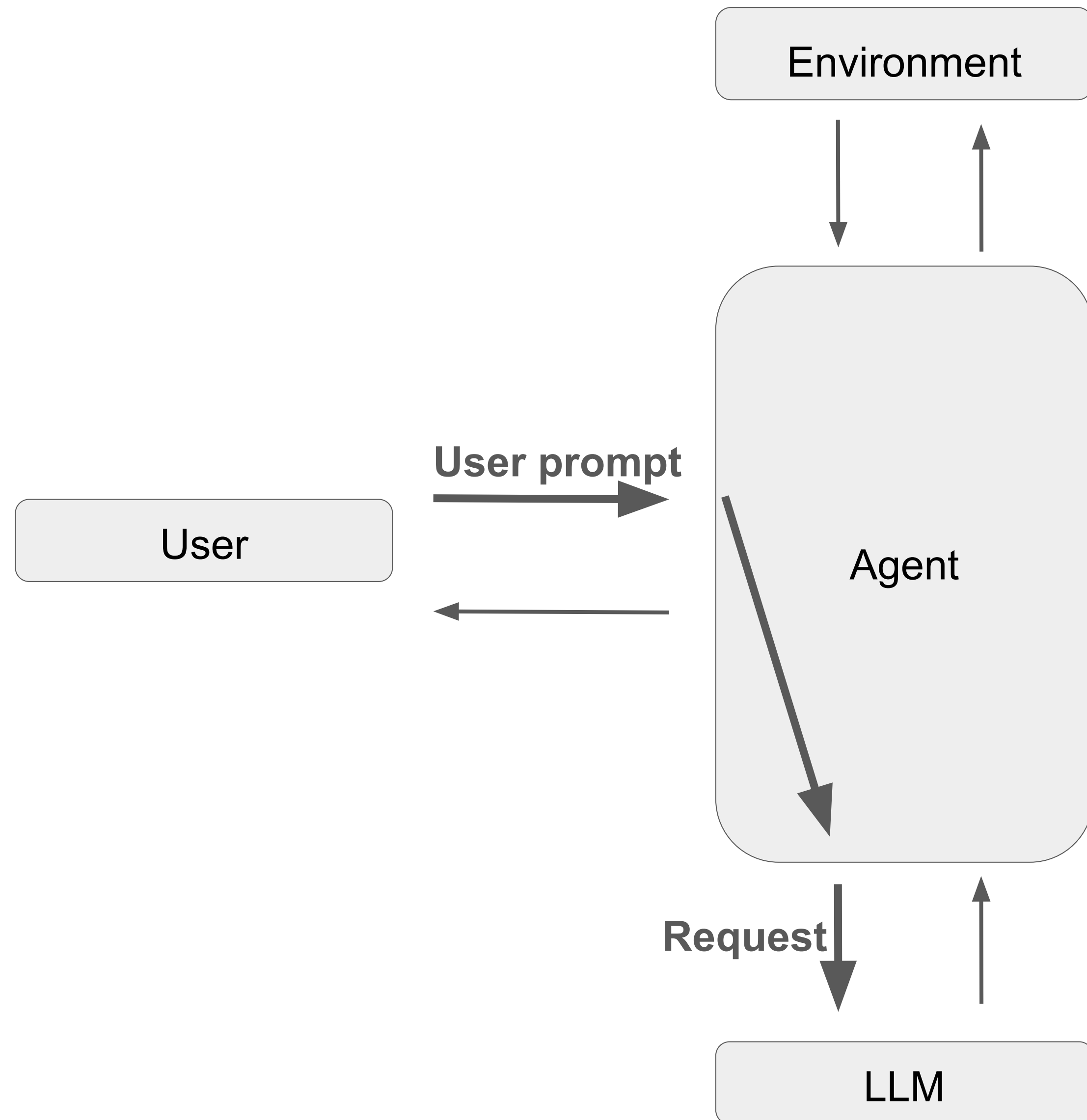


Tools



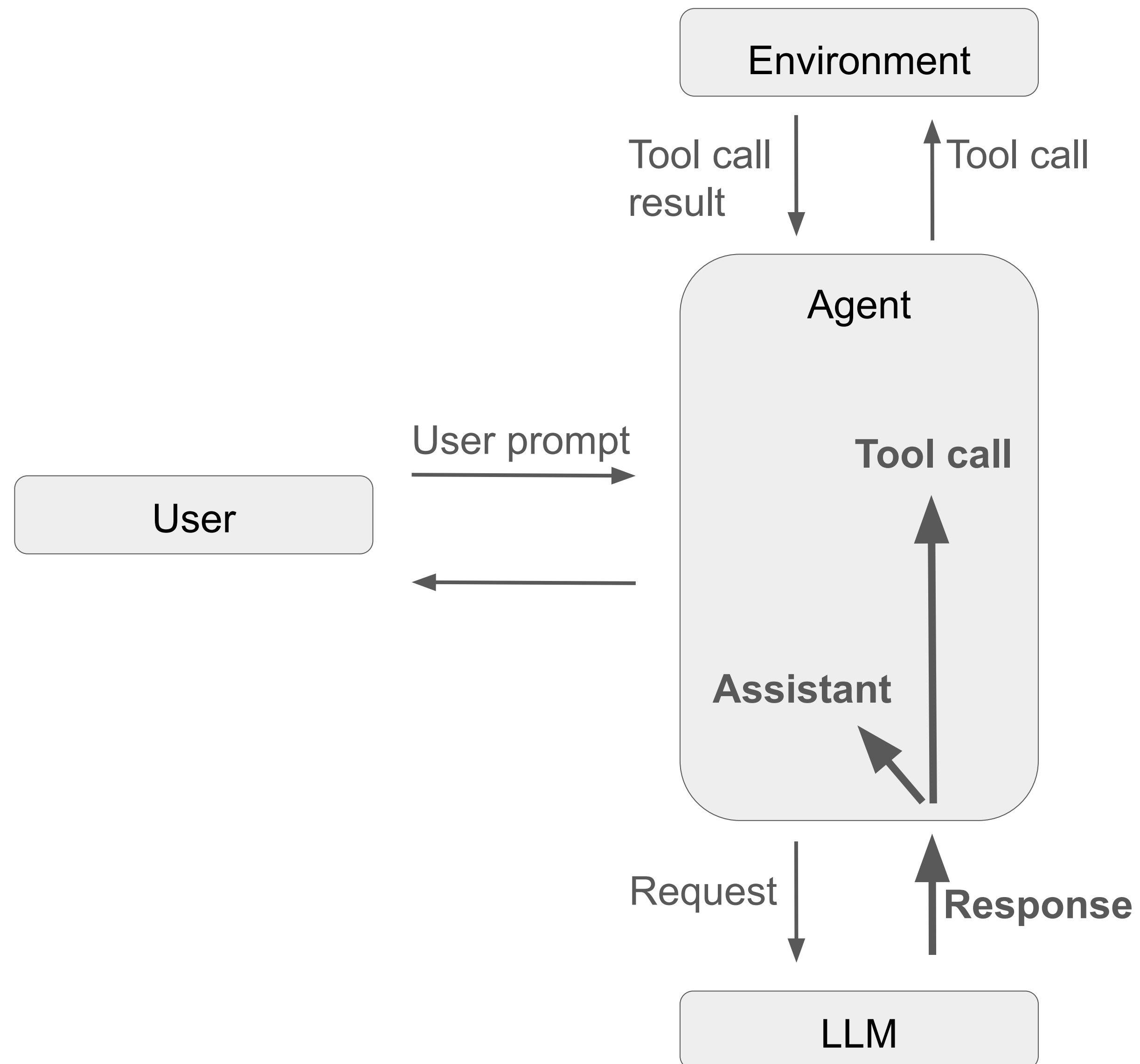
-  04-Tool.kt
-  05-ToolSet.kt
-  06-ToolRegistry.kt
-  07-McpToolRegistry.kt

singleRunStrategy



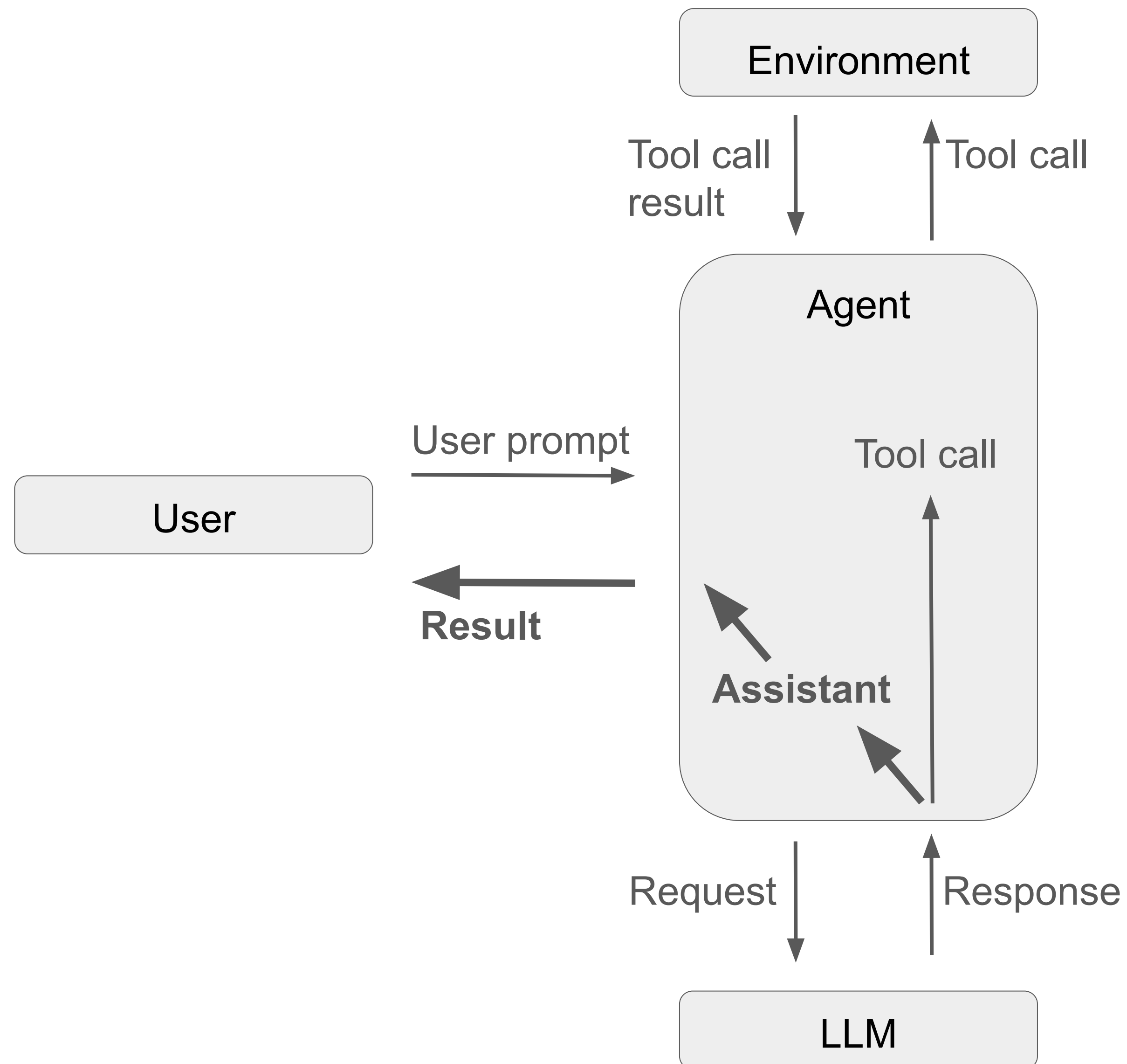
1. Gets user message and sends to the LLM

singleRunStrategy



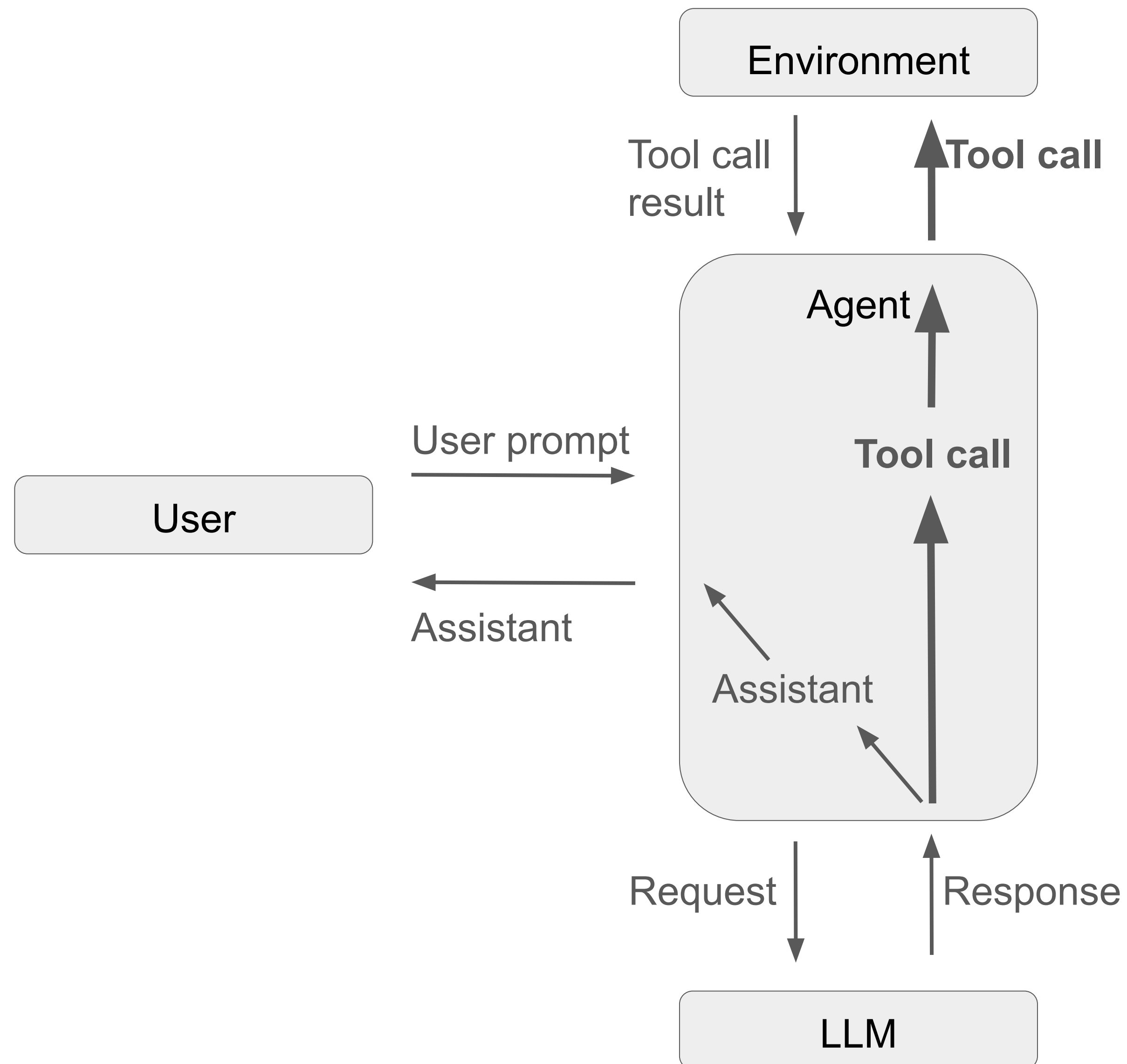
1. Gets user message and sends to the LLM
2. Gets the response (can be Assistant or Tool.Call message)

singleRunStrategy



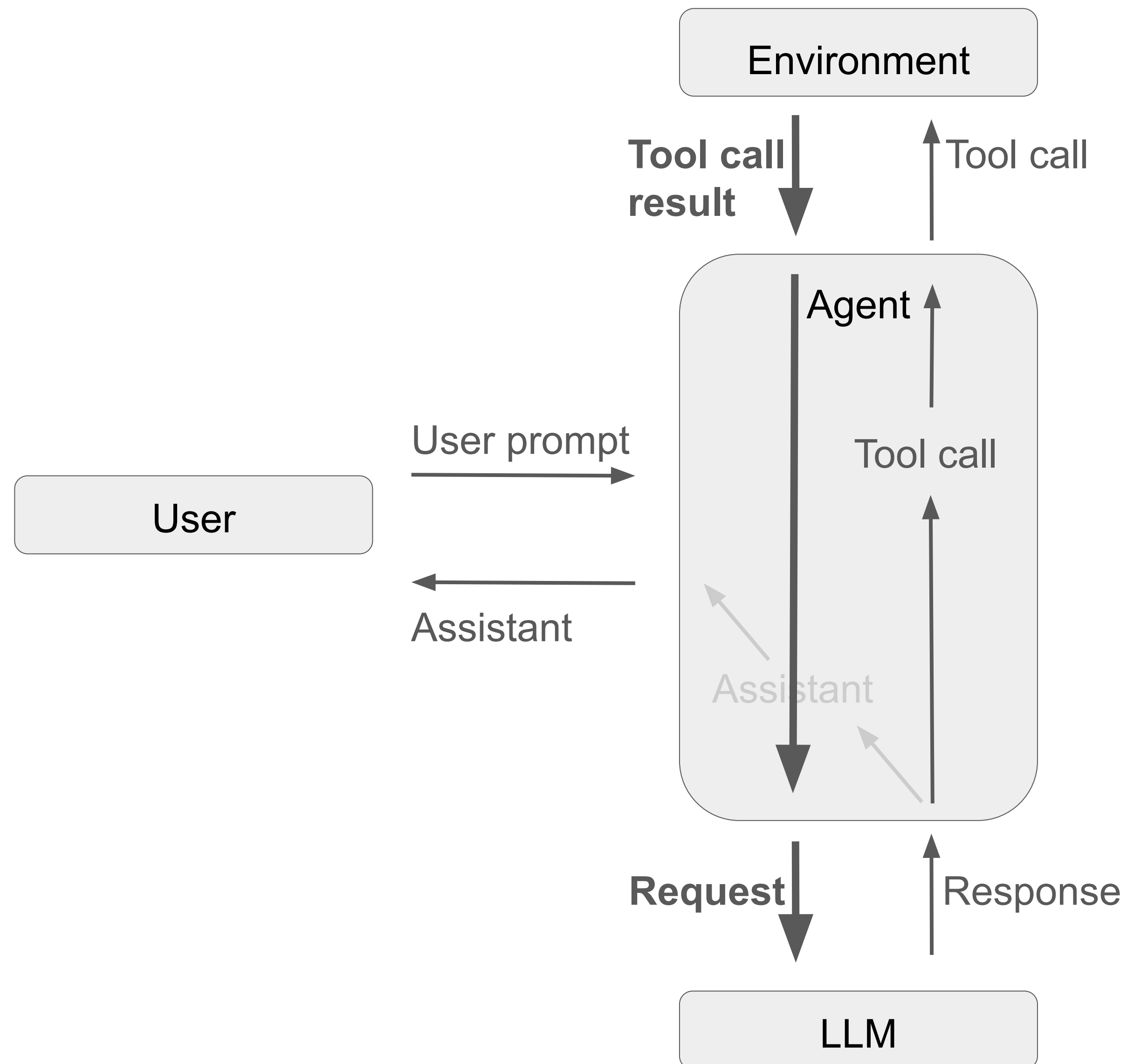
1. Gets user message and sends to the LLM
2. Gets the response (can be Assistant or Tool.Call message)
3. If Assistant returns the content to user

singleRunStrategy



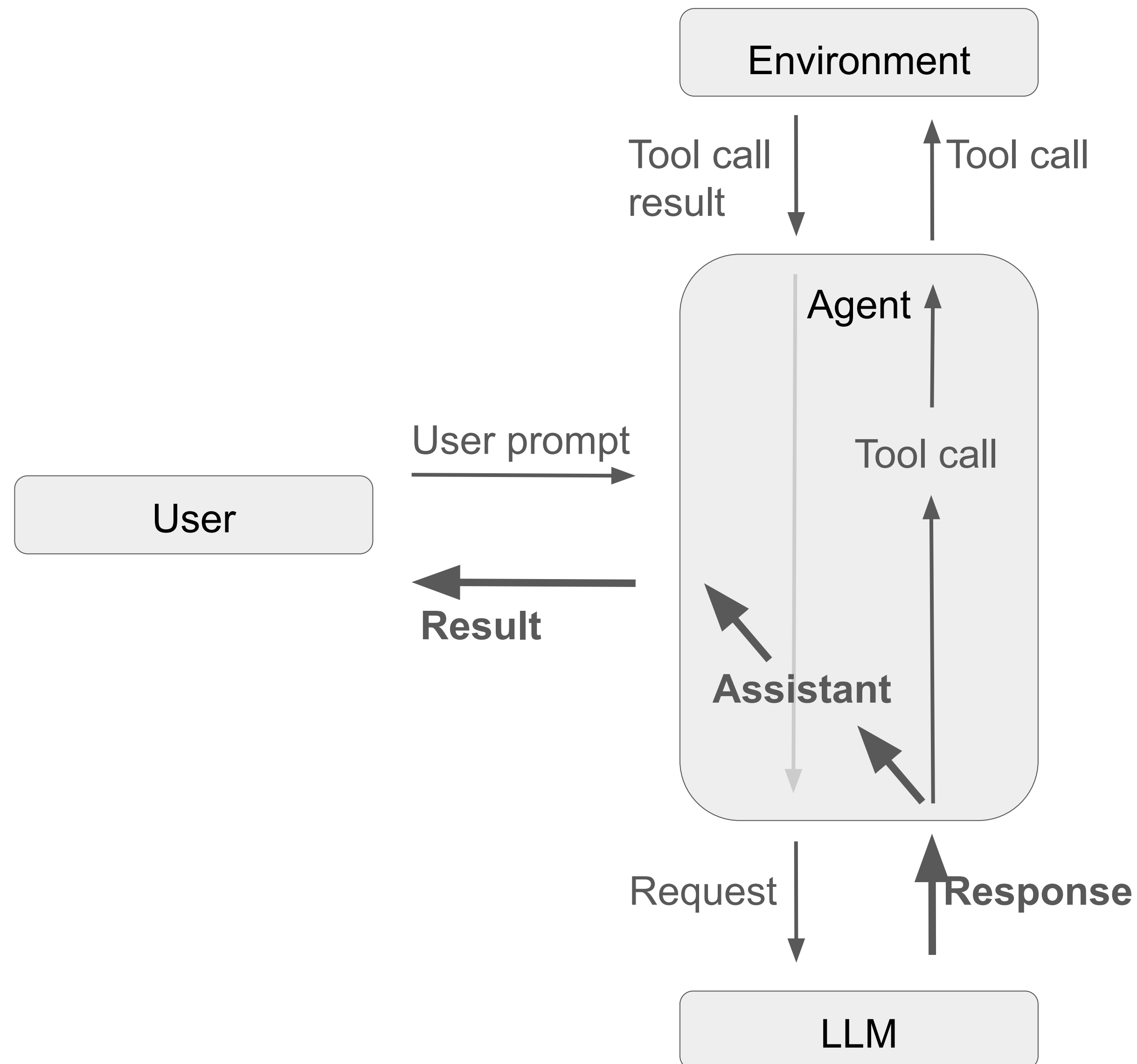
1. Gets user message and sends to the LLM
2. Gets the response (can be Assistant or Tool.Call message)
3. If Assistant returns the content to user
4. If Tool.Call executes the tool

singleRunStrategy



1. Gets user message and sends to the LLM
2. Gets the response (can be Assistant or Tool.Call message)
3. If Assistant returns the content to user
4. If Tool.Call executes the tool
5. Gets the Tool.Result and sends back to the LLM

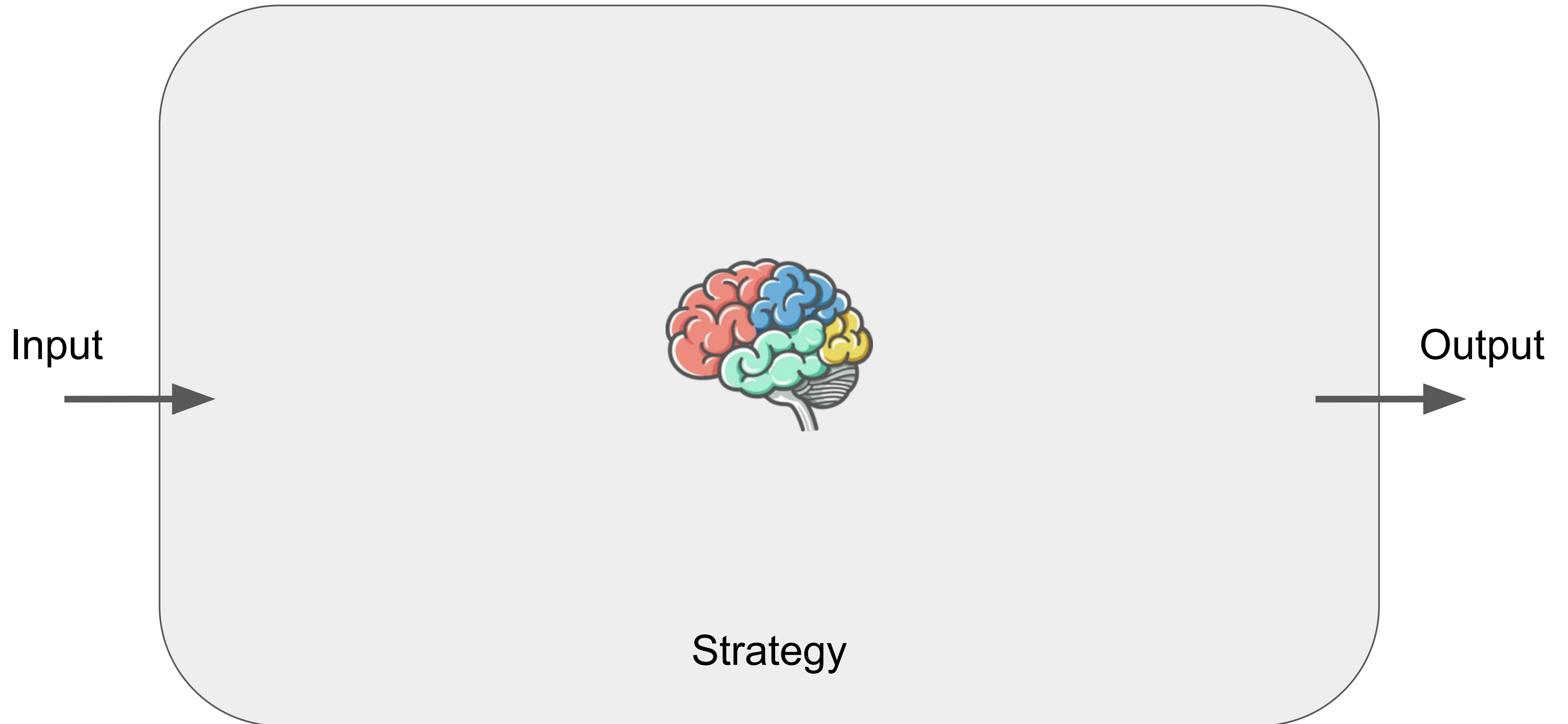
singleRunStrategy



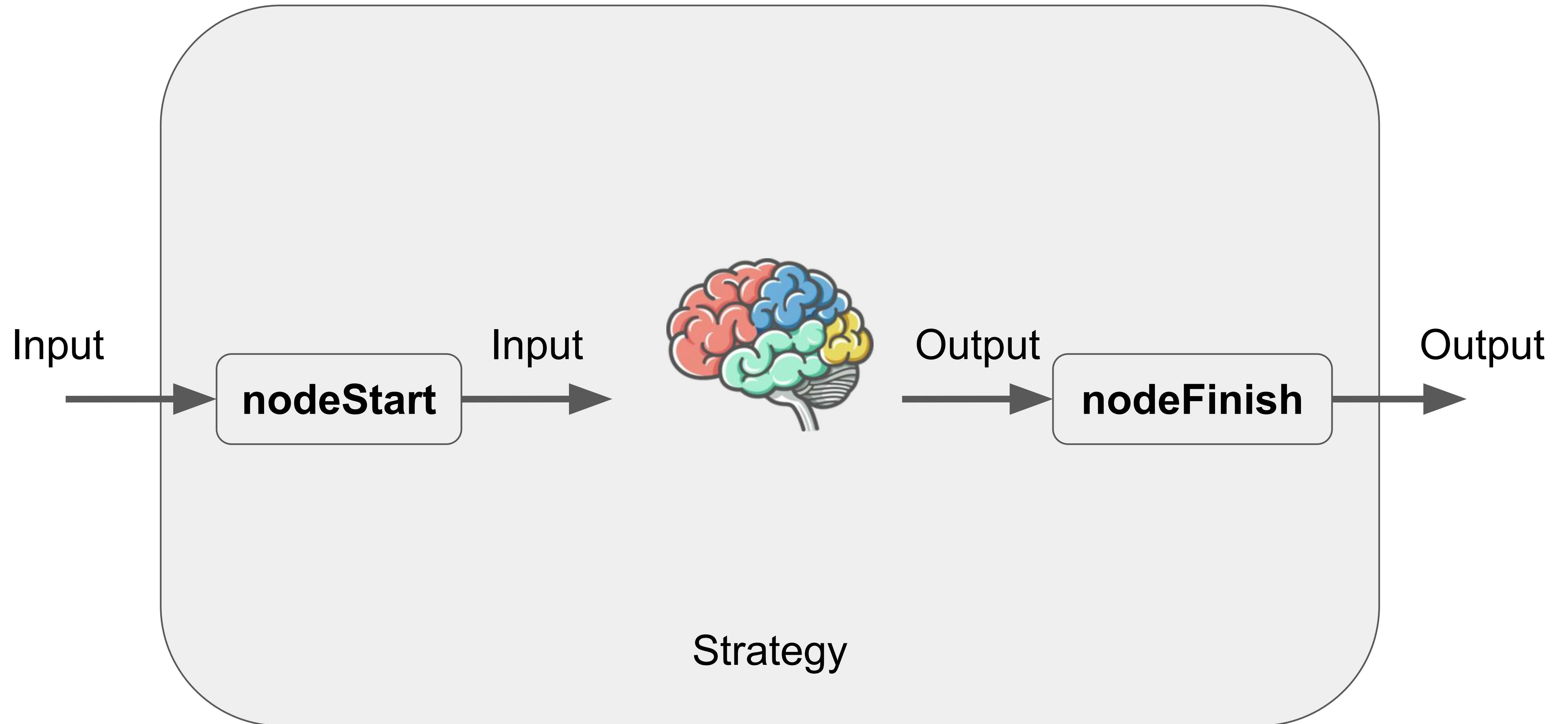
1. Gets user message and sends to the LLM
2. Gets the response (can be Assistant or Tool.Call message)
3. If Assistant returns the content to user
4. If Tool.Call executes the tool
5. Gets the Tool.Result and sends back to the LLM
6. Repeats starting from 2.

 08-SimpleAgent.kt

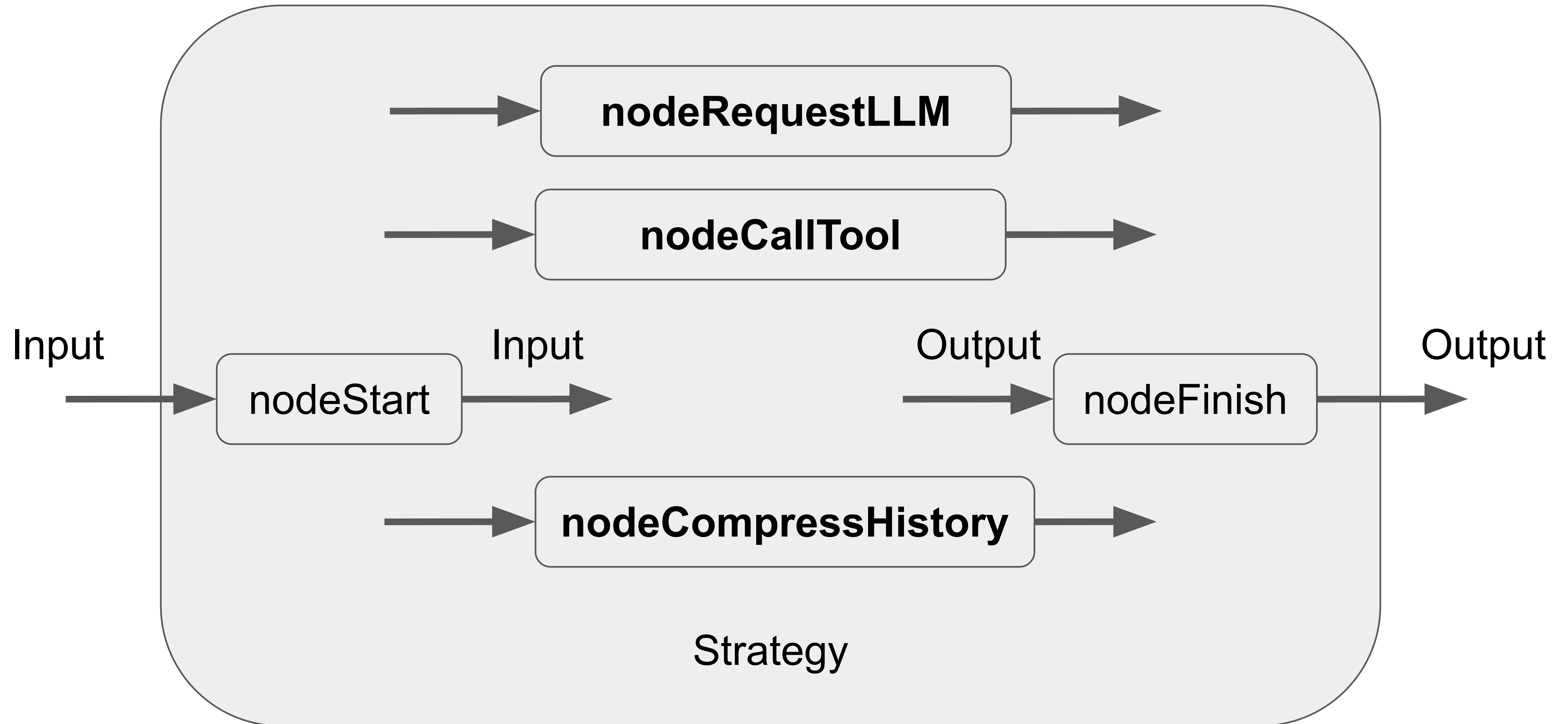
Strategy, Node, Subgraph



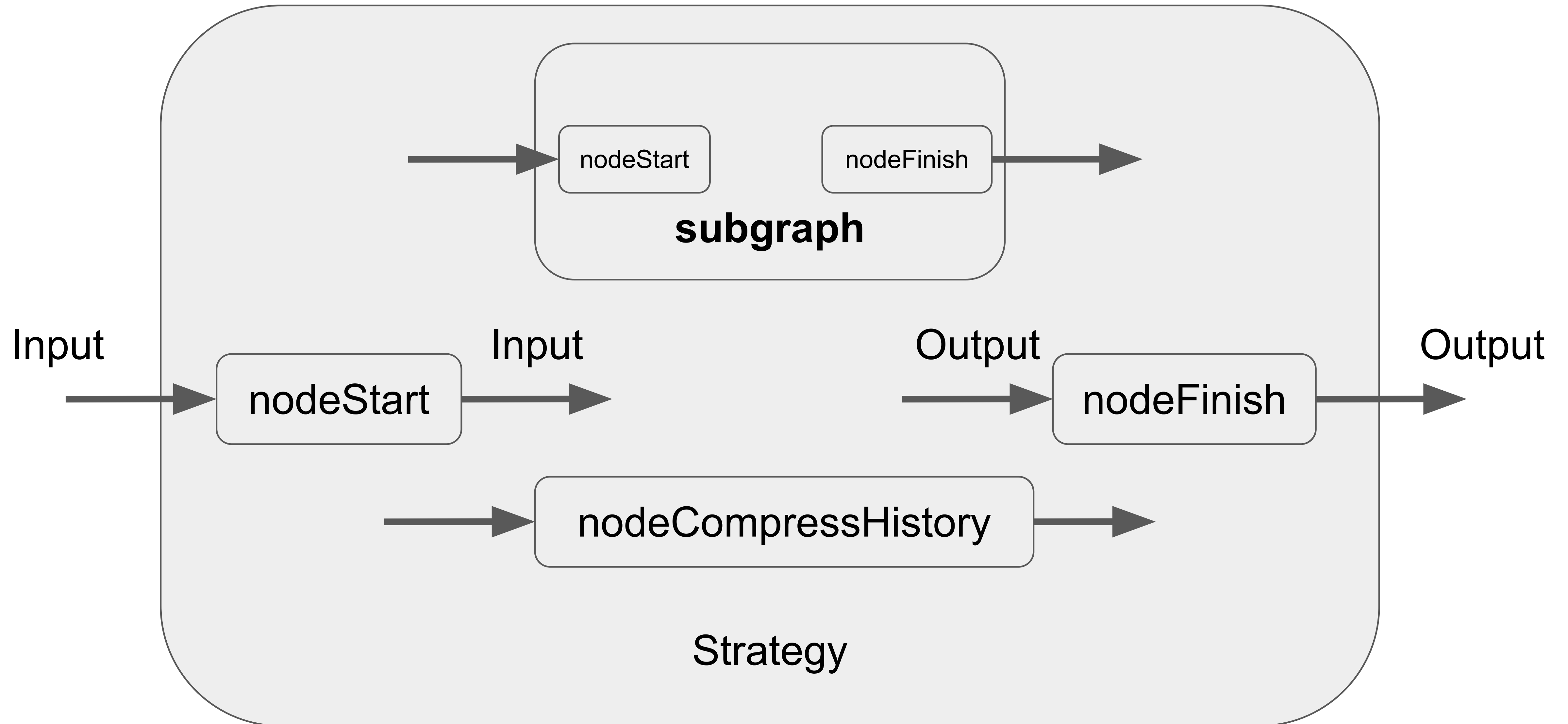
Strategy, Node, Subgraph



Strategy, Node, Subgraph



Strategy, Node, Subgraph



-  09-StrategyWithBasicNodes.kt
-  10-StrategyWithTools.kt
-  11-StrategyWithSubgraph.kt
-  12-StrategyWithStructuredOutput.kt

Thanks for attending!