Tool Binding, Tool Calling, and Tool Execution in LangChain

Overview

In modern LLM frameworks such as **LangChain**, tools act as external functions or APIs that extend the capabilities of language models. These tools allow models to perform tasks such as:

- Fetching real-time data (e.g., weather, exchange rates)
- Performing calculations or data transformations
- Accessing structured databases or APIs

This process occurs through three main phases:

1. **Tool Binding**
2. **Tool Calling**
3. **Tool Execution**

1. Tool Binding

Definition

Tool Binding is the stage where tools are **registered or exposed to the LLM**. This tells the model:

- What tools are available
- What each tool does (via a description)
- What inputs are required (via a schema)

Example of Tool Binding

```
bind_tools([get_conversion_factor, convert_currency])
```

This does not execute the tool—it only defines it.

2. Tool Calling

Definition

During reasoning, the LLM may decide that a tool is necessary to answer the user's query. At this point, it produces a structured "call" specifying:

- The tool name
- The input arguments (in JSON format)

Example of Tool Calling

```
 {
"tool":  "multiply",
"args":  { "a":  8, "b":  7 }
}
```

The LLM itself does not run the function; it only recommends what to run.

3. Tool Execution

Definition

This is where the suggested tool is actually executed by the framework (or developer). The system takes the tool name and arguments, calls the real Python function, and returns the result back to the LLM.

Example

LLM Suggests: Call `multiply(a=8, b=7)`.
Execution: Python runs the function.
Result: 56

After execution, the LLM integrates the result into its reasoning and provides a final human-readable answer.

4. Example: Currency Conversion Flow

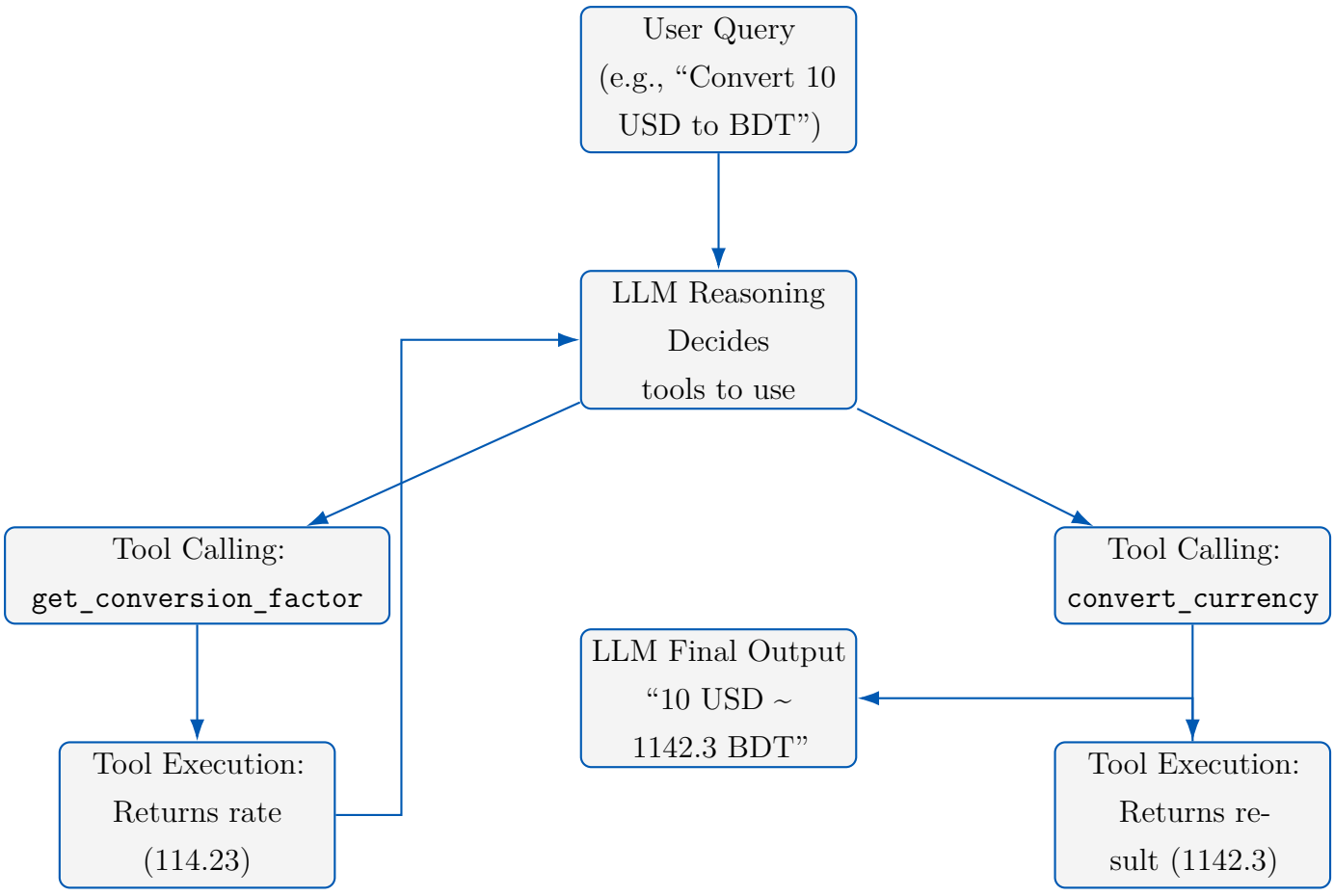
User Query

"Convert 10 USD to BDT."

Process

1. The LLM realizes it needs the exchange rate. → Calls `get_conversion_factor("USD", "BDT")`.
2. Tool Execution returns the rate: `114.23`.
3. LLM then calls `convert_currency(10, 114.23)`.
4. Execution returns: `1142.3 BDT`.
5. LLM responds: "10 USD is approximately 1142.3 BDT."

5. Visual Representation



6. Summary

Tool Lifecycle Summary

Stage	Description
Tool Binding	Registering tools and their input schemas.
Tool Calling	LLM chooses which tool to use with what arguments.
Tool Execution	Framework runs the tool and returns results.

7. Key Takeaways

- Tool Binding makes tools discoverable to the LLM.
- Tool Calling is the LLM's reasoning step—deciding which tool fits the context.
- Tool Execution connects language reasoning with computational power.
- This pipeline allows LLMs to act as intelligent agents that can interact with the external world.