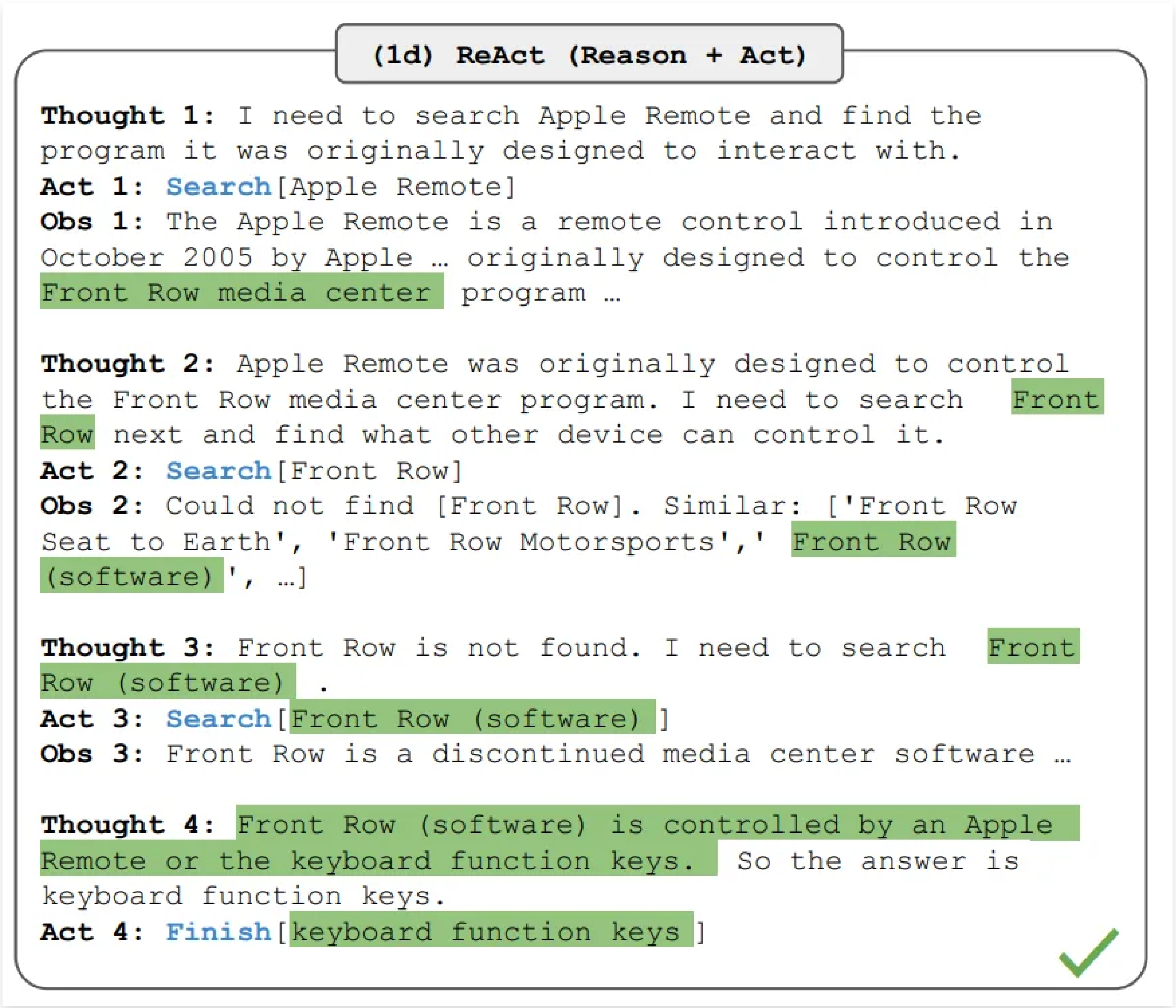
Agent

reAct

1. 思考–行动–观察



(1d) ReAct (Reason + Act)

Thought 1: I need to search Apple Remote and find the program it was originally designed to interact with.
Act 1: Search[Apple Remote]
Obs 1: The Apple Remote is a remote control introduced in October 2005 by Apple … originally designed to control the Front Row media center program …

Thought 2: Apple Remote was originally designed to control the Front Row media center program. I need to search Front Row next and find what other device can control it.
Act 2: Search[Front Row]
Obs 2: Could not find [Front Row]. Similar: ['Front Row Seat to Earth', 'Front Row Motorsports',' Front Row (software) ', …]

Thought 3: Front Row is not found. I need to search Front Row (software) .
Act 3: Search[Front Row (software)]
Obs 3: Front Row is a discontinued media center software …

Thought 4: Front Row (software) is controlled by an Apple Remote or the keyboard function keys. So the answer is keyboard function keys.
Act 4: Finish[keyboard function keys]

✓

2.langchain 实现 reAct 使用的 prompt

`libs/langchain/langchain/agents/mrkl/prompt.py`

```python
PREFIX = """Answer the following questions as best you can. You have access to the following tools:"""  # noqa: E501
FORMAT_INSTRUCTIONS = """Use the following format:

Question: the input question you must answer
Thought: you should always think about what to do
Action: the action to take, should be one of [{tool_names}]
Action Input: the input to the action
Observation: the result of the action
... (this Thought/Action/Action Input/Observation can repeat N times)
Thought: I now know the final answer
Final Answer: the final answer to the original input question"""
SUFFIX = """Begin!

Question: {input}
Thought:{agent_scratchpad}"""
```

Function call

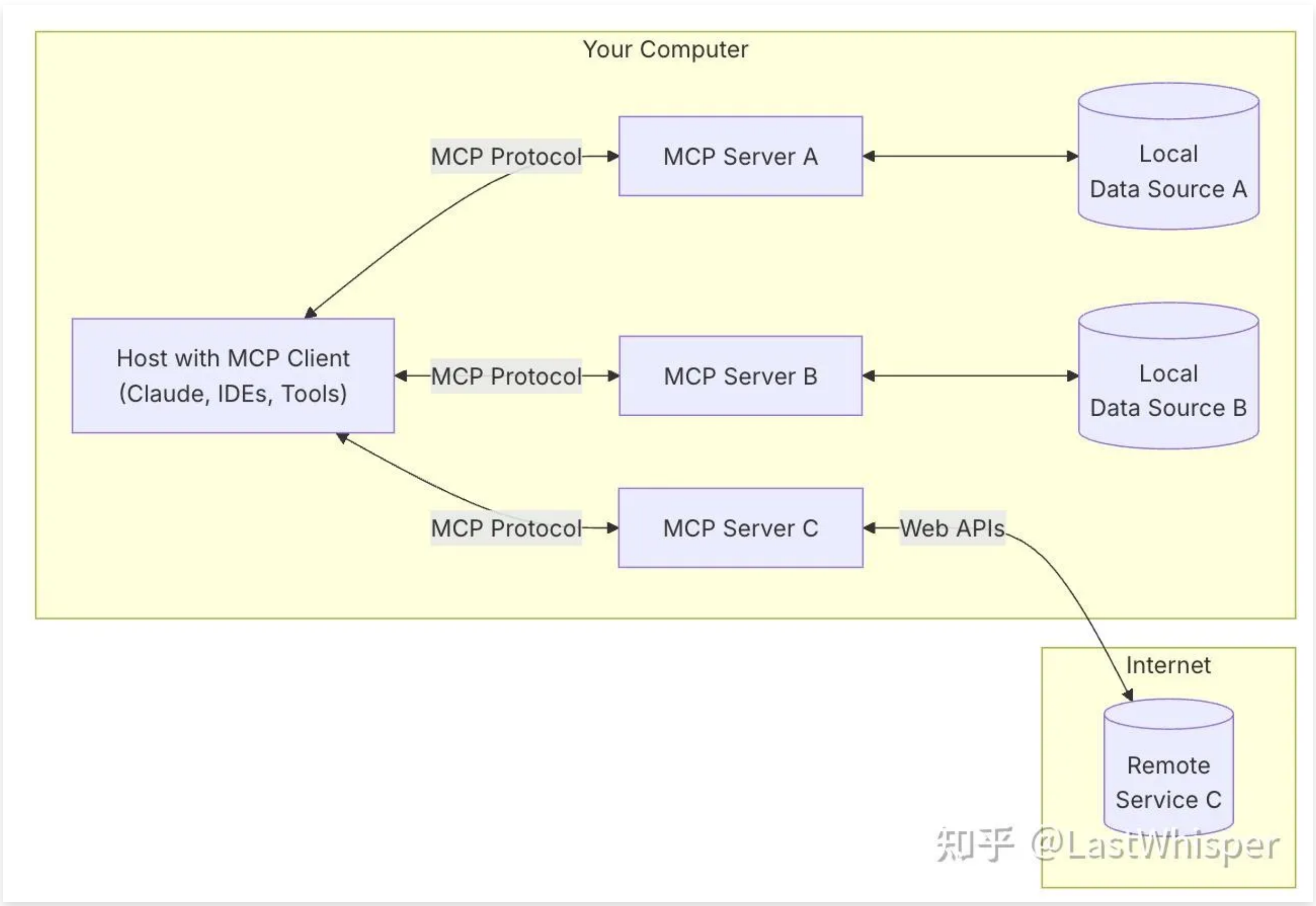
1. 用户提出问题。
2. 系统接收到问题，并检查是否有可用的函数可以调用。
3. 如果有，系统会生成一个工具调用请求（ToolCall，包括要调用的函数以及提取的参数），并发送给应用程序。
4. 应用程序执行请求的函数，并返回结果。
5. 系统将函数的响应（ToolCallResponse）发送回 LLM 模型。
6. LLM 模型使用这个响应来生成最终的用户响应。

训练：将每条数据样本组合成模型可以理解的格式。通常是将“用户输入”和“可用函数描述”拼接起来作为模型的输入 (Prompt)，将“期望的输出”（无论是 JSON 函数调用还是文本回答）作为目标输出 (Completion/Target)。使用标准的 SFT 方法（全参数微调或 PEFT 如 LoRA）在准备好的数据集上训练模型。

示例：https://huggingface.co/datasets/glaiveai/glaive-function-calling-v2

MCP

function call 的问题：标准不同，例如 openai 与 Google



由 server 进行函数调用，client 与 server 交互

框架

langchain

AutoGen