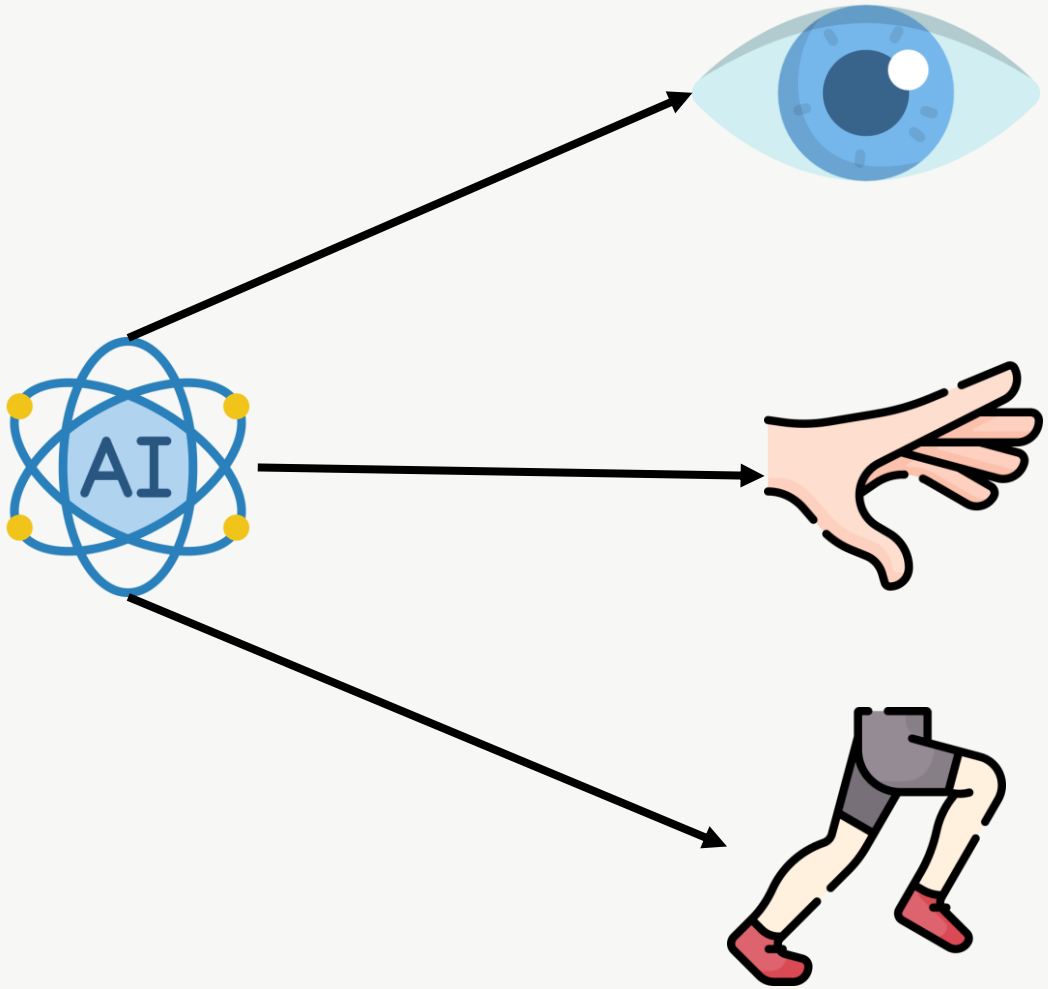
香港中文大學（深圳）
The Chinese University of Hong Kong, Shenzhen

CSC4801
AI-assisted Software Engineering

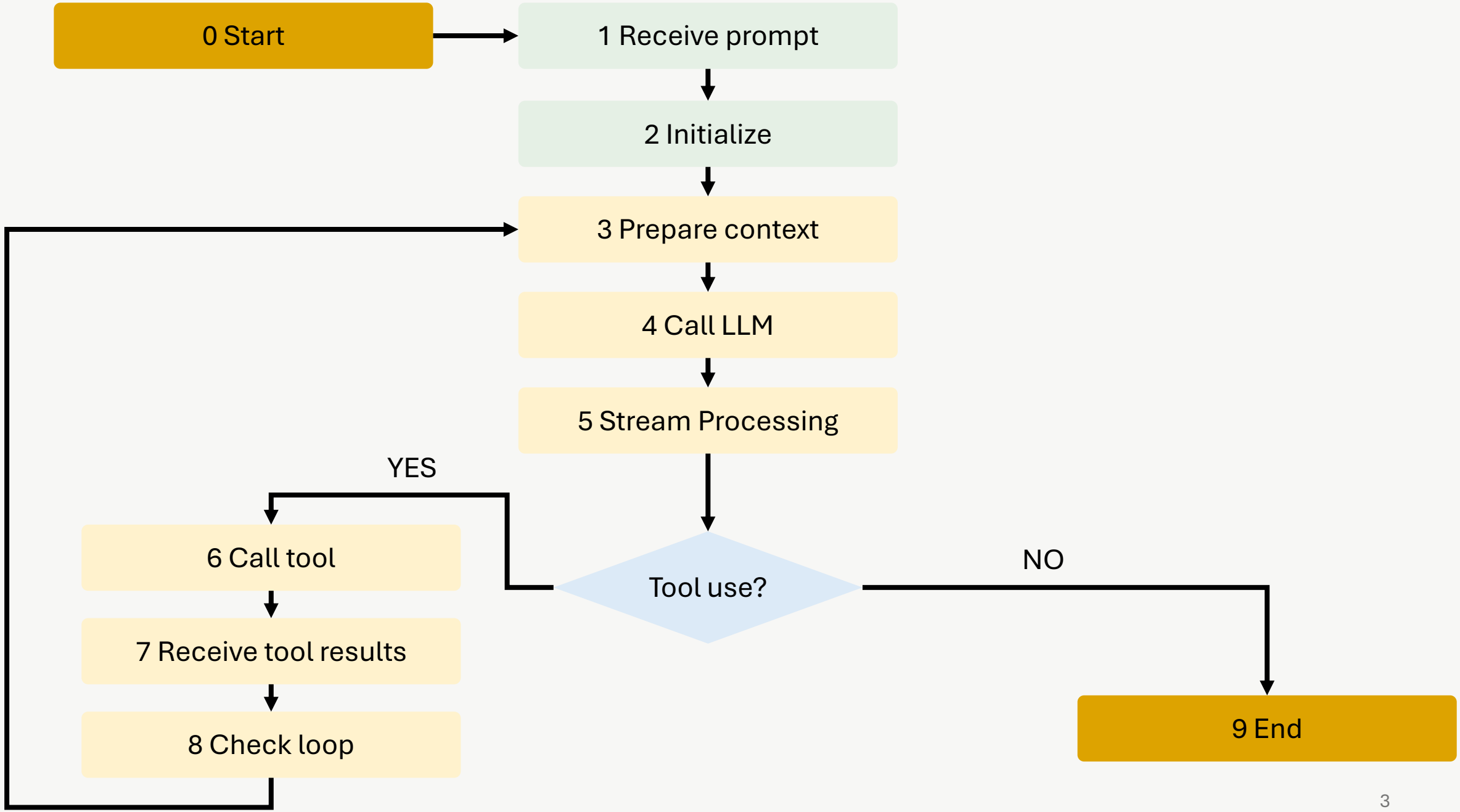
Lecture 03: Advanced Agents

Instructor: Jinsheng Ba | 2026 Fall

Last Lecture



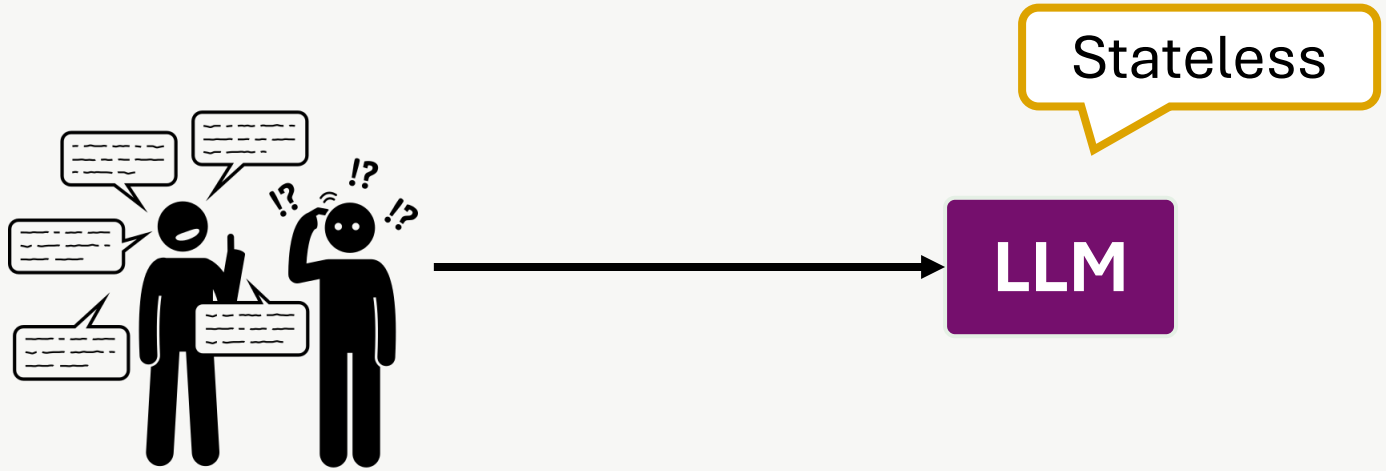
Let’s make agents stronger!


0 Start
1 Receive prompt
2 Initialize
3 Prepare context
4 Call LLM
5 Stream Processing
Tool use?
YES
NO
6 Call tool
7 Receive tool results
8 Check loop
9 End

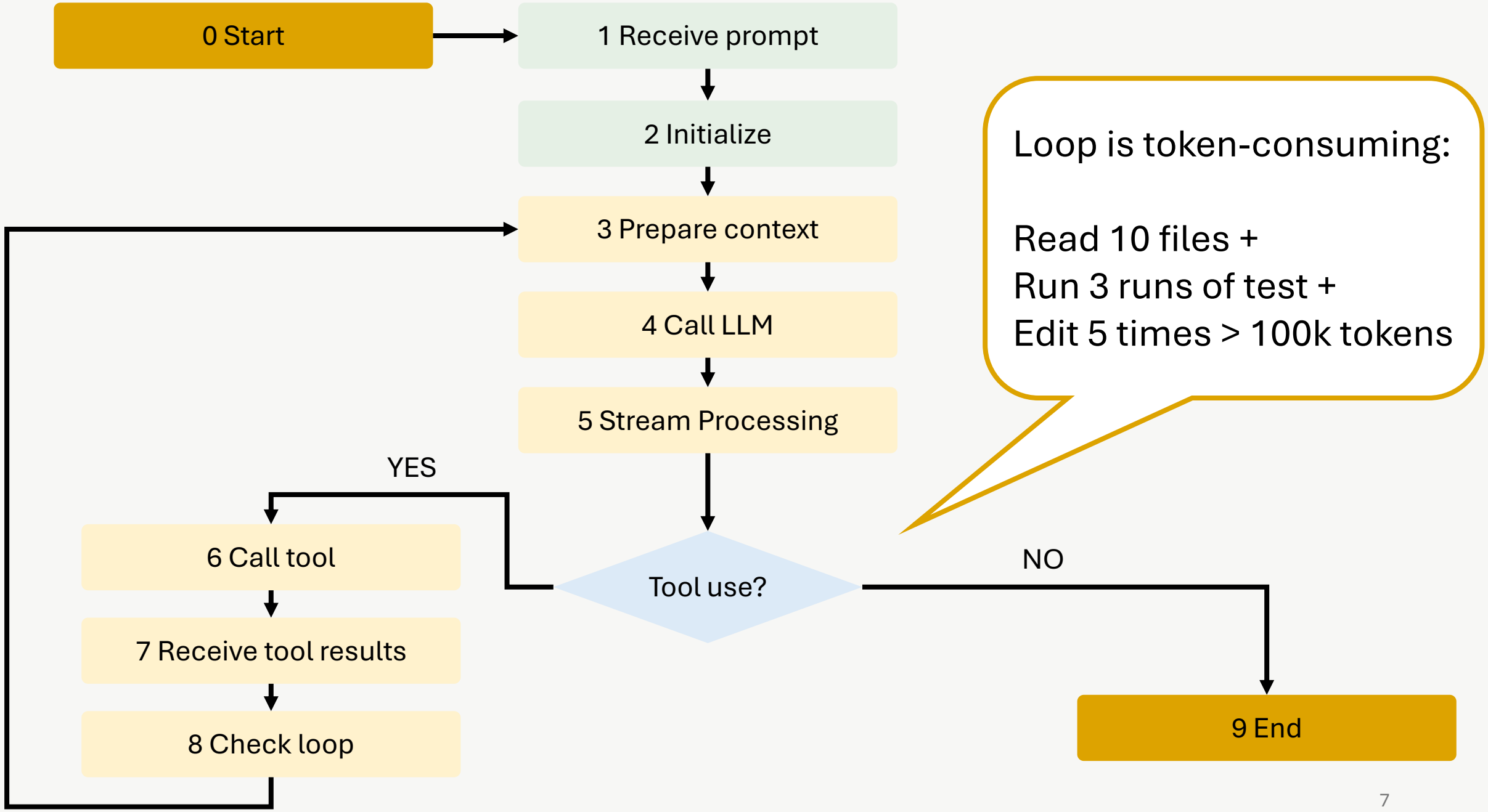
Several Problems:

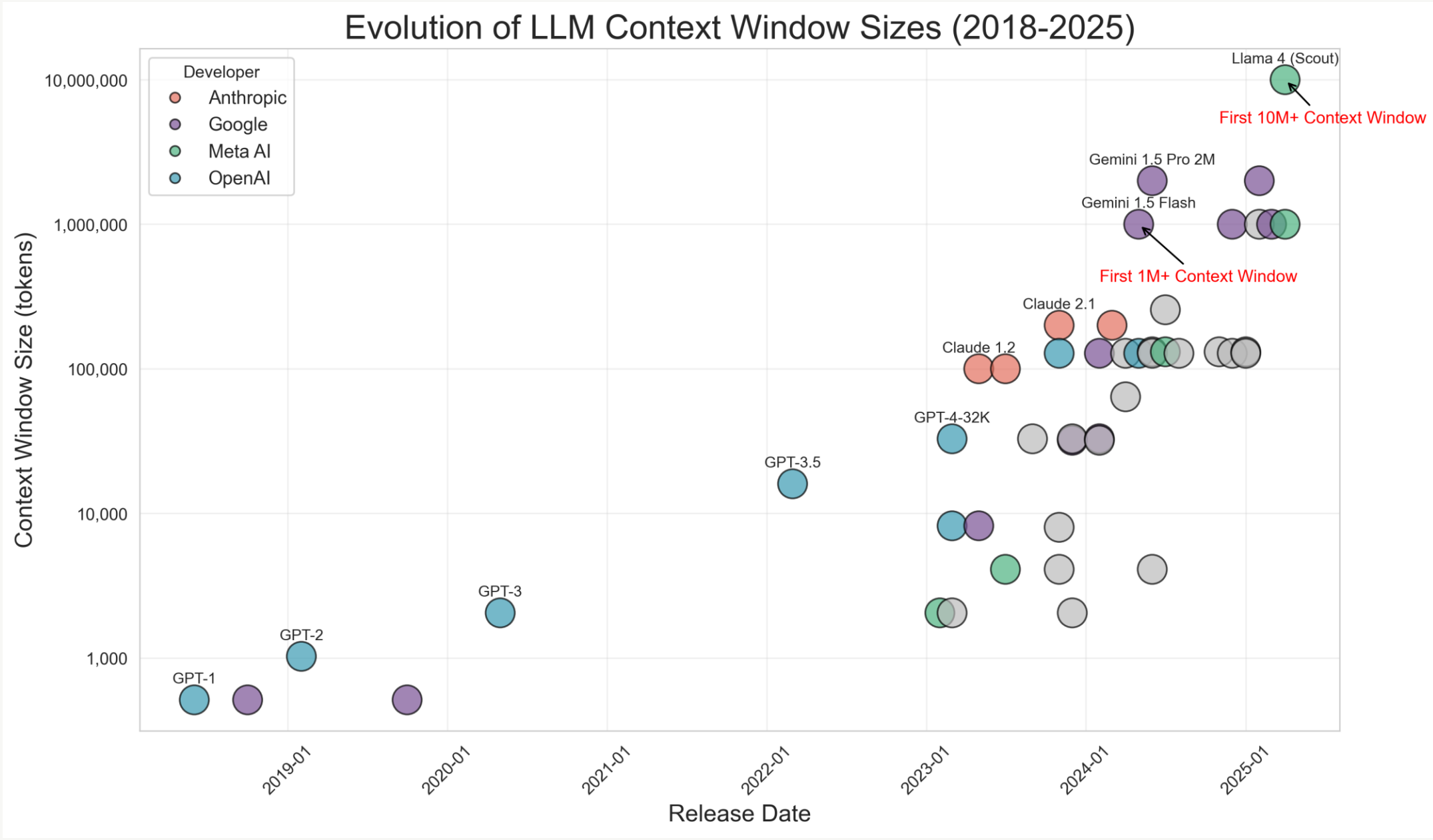
1 Context Management

Problem



All historical messages are included in **each** LLM API call.


0 Start
1 Receive prompt
2 Initialize
3 Prepare context
4 Call LLM
5 Stream Processing
Tool use?
YES
6 Call tool
7 Receive tool results
8 Check loop
NO
9 End
Loop is token-consuming:
Read 10 files +
Run 3 runs of test +
Edit 5 times > 100k tokens



*** https://www.meibel.ai/post/understanding-the-impact-of-increasing-llm-context-windows**

Key Statistics

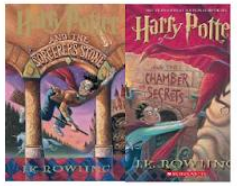
- As of September 15, 2026, the largest LLM context window tracked by BenchLM is 10M tokens, offered by Pokee AI's Pokee-Isaac 28B.
- 24% of the 386 AI models with published context windows tracked by BenchLM offer 1M tokens or more as of September 15, 2026.
- The median context window across all AI models tracked by BenchLM is 256K tokens as of September 15, 2026.
- A 1M-token context window — offered by 93 models tracked by BenchLM as of September 15, 2026 — fits the full text of War & Peace (roughly 750,000 tokens) with room left for an average novel.

https://benchlm.ai/stats/context-windows

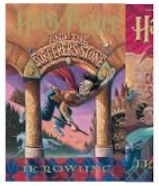
Google's Gemini 1.5 can (almost) fit the entire Harry Potter + Lord of the Ring series in its 2 million context window

Gemini 1.5 2M (June 2024)

Claude 2.1 (July 2023)



GPT-4 Turbo (March 2023)



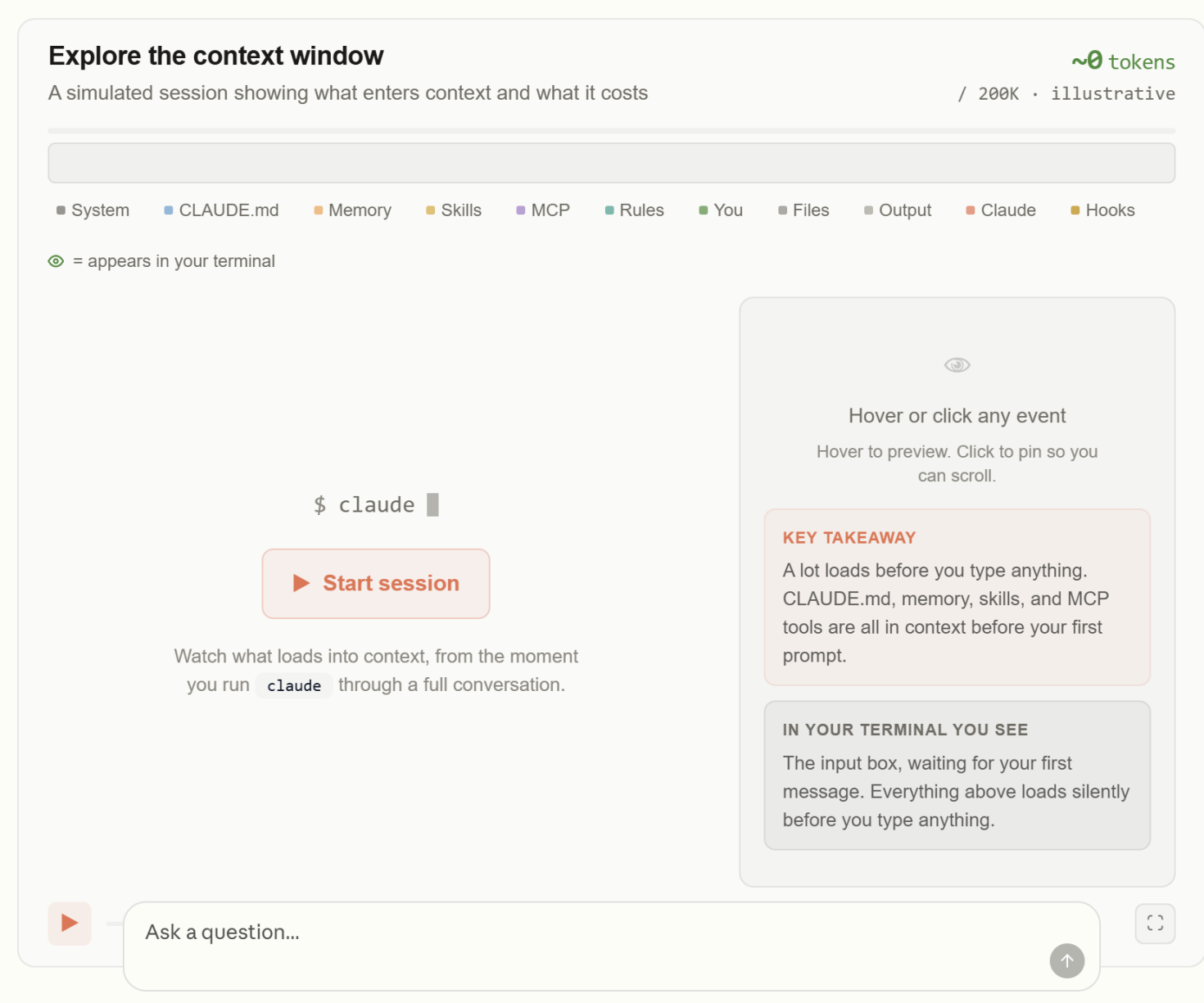
GPT-3.5 Turbo (March 2022)

Source of image: https://www.artfish.ai/p/long-context-llms

Too long context may fail

https://www.dbreunig.com/2025/06/22/how-contexts-fail-and-how-to-fix-them.html

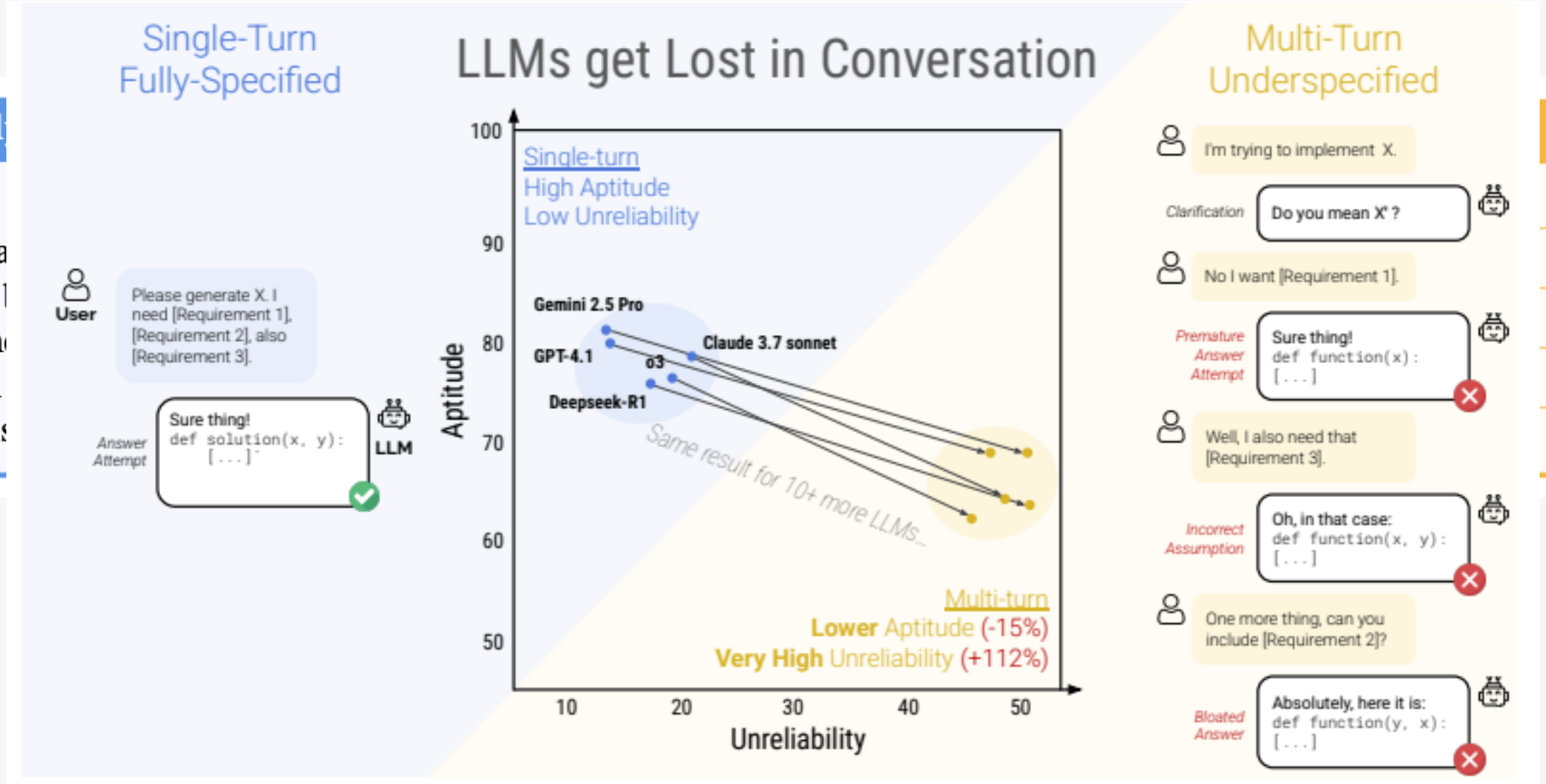
Visualization



- https://code.claude.com/docs/en/context-window

Lost in the Conversation?

https://arxiv.org/pdf/2505.06120v1



Lost in the Conversation?

Historical messages need to be concise.

RULER: What's the Real Context Size of Your Long-Context Language Models?

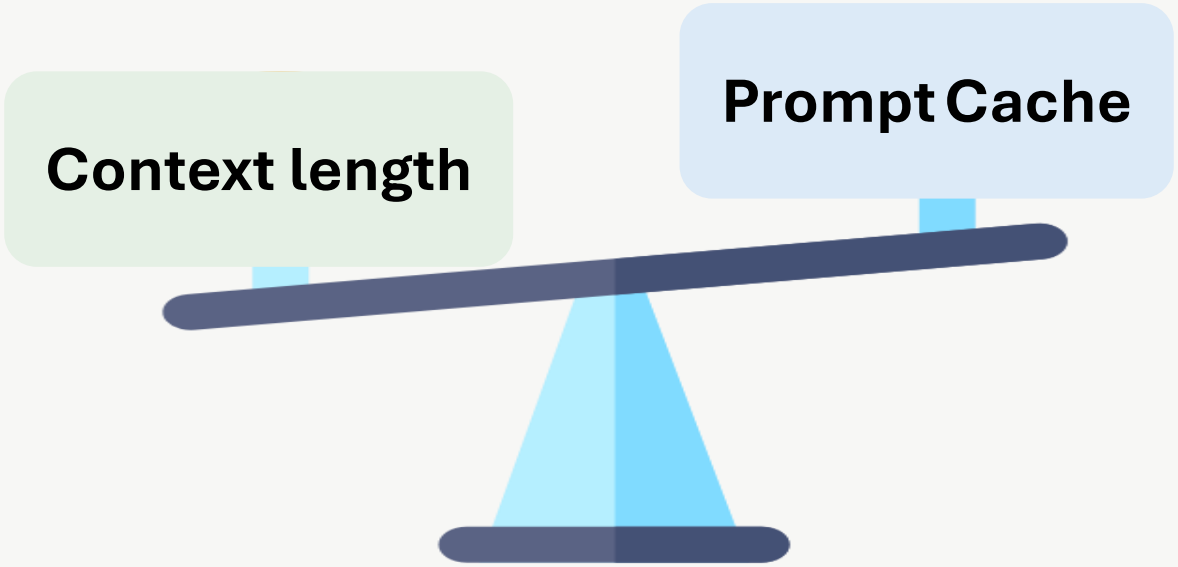
Cheng-Ping Hsieh*, Simeng Sun*, Samuel Kriman, Shantanu Acharya
Dima Rekesh, Fei Jia, Yang Zhang, Boris Ginsburg

NVIDIA
{chsieh,simengs}@nvidia.com

long-context LMs with 13 representative tasks in RULER. Despite achieving nearly perfect accuracy in the vanilla NIAH test, almost all models exhibit large performance drops as the context length increases. While these models all claim context sizes of 32K tokens or greater, only half of them

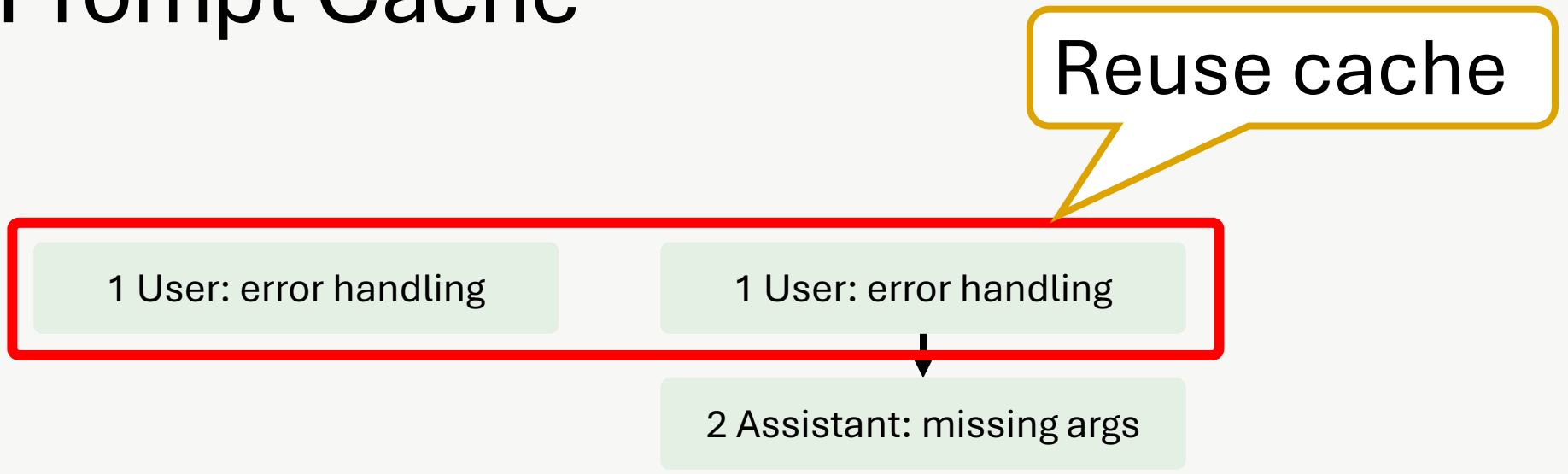
https://arxiv.org/pdf/2404.06654

Design Principle



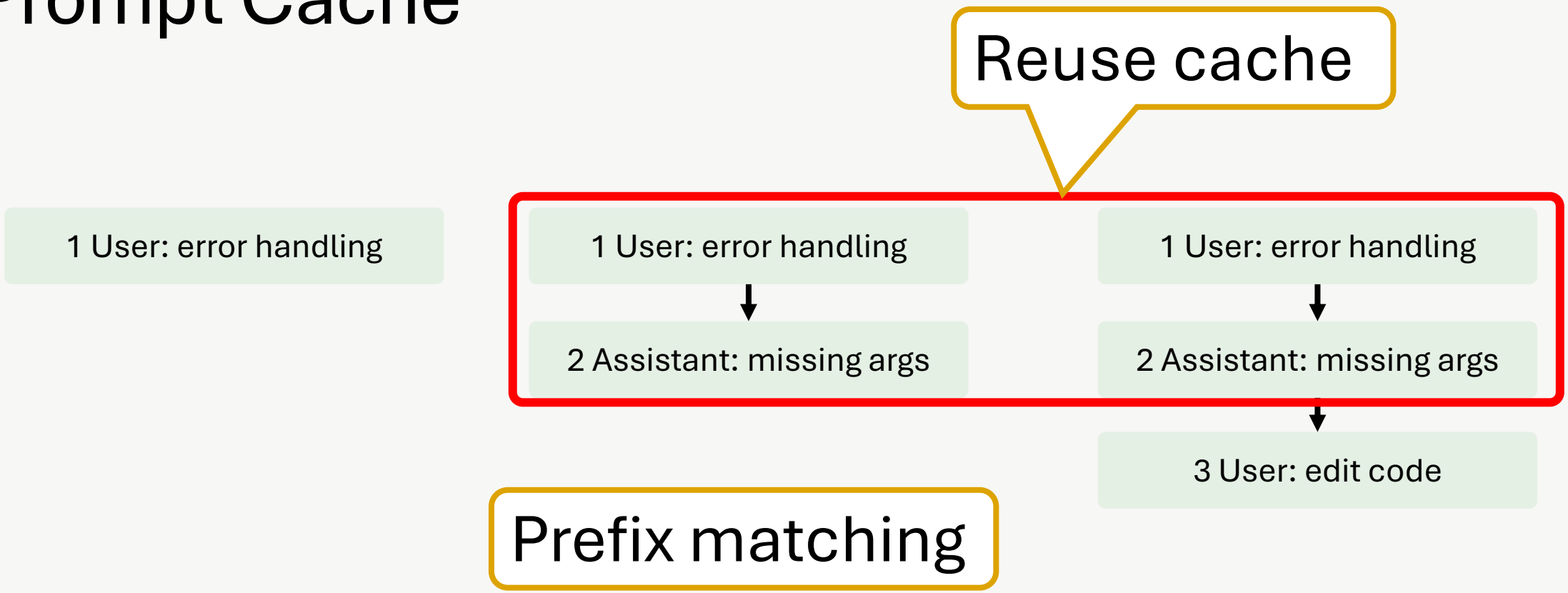
How to reduce context length with minimal cache miss?

Prompt Cache



PROMPT CACHE: MODULAR ATTENTION REUSEFOR LOW-LATENCY INFERENCE
https://arxiv.org/pdf/2311.04934

Prompt Cache



PROMPT CACHE: MODULAR ATTENTION REUSEFOR LOW-LATENCY INFERENCE
https://arxiv.org/pdf/2311.04934

Prompt Cache

PRICING	1M INPUT TOKENS (CACHE HIT)	$0.0028
	1M INPUT TOKENS (CACHE MISS)	$0.14
	1M OUTPUT TOKENS	$0.28

50X cheaper!

Also lower latency

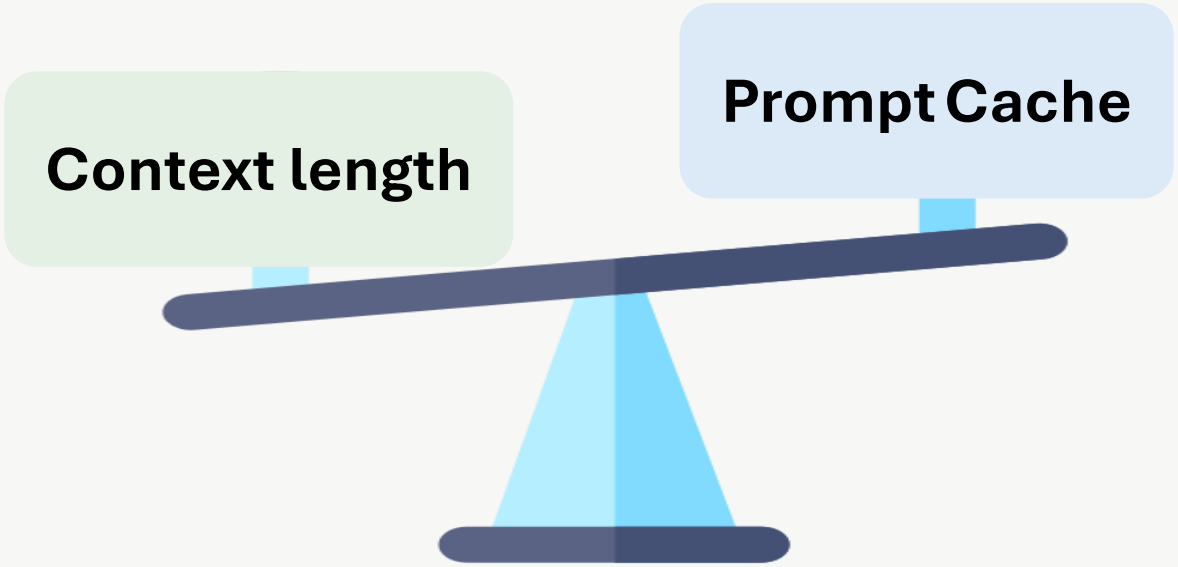
DeepSeek V4 Flash API price:
https://api-docs.deepseek.com/quick_start/pricing

System Prompt Before User Prompt

- System prompts are typically fixed
- Increase prompt cache hit possibility

```
{
    ...
    "system": "You are a helpful coding assistant.",
    "type": "user",
    "message": {
        "role": "user",
        "content": "Add error handling in the file @main.py"
    },
    ...
}
```

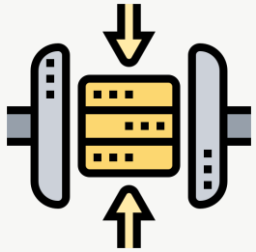
Design Principle



How to reduce context length with minimal cache miss?

5 Strategies

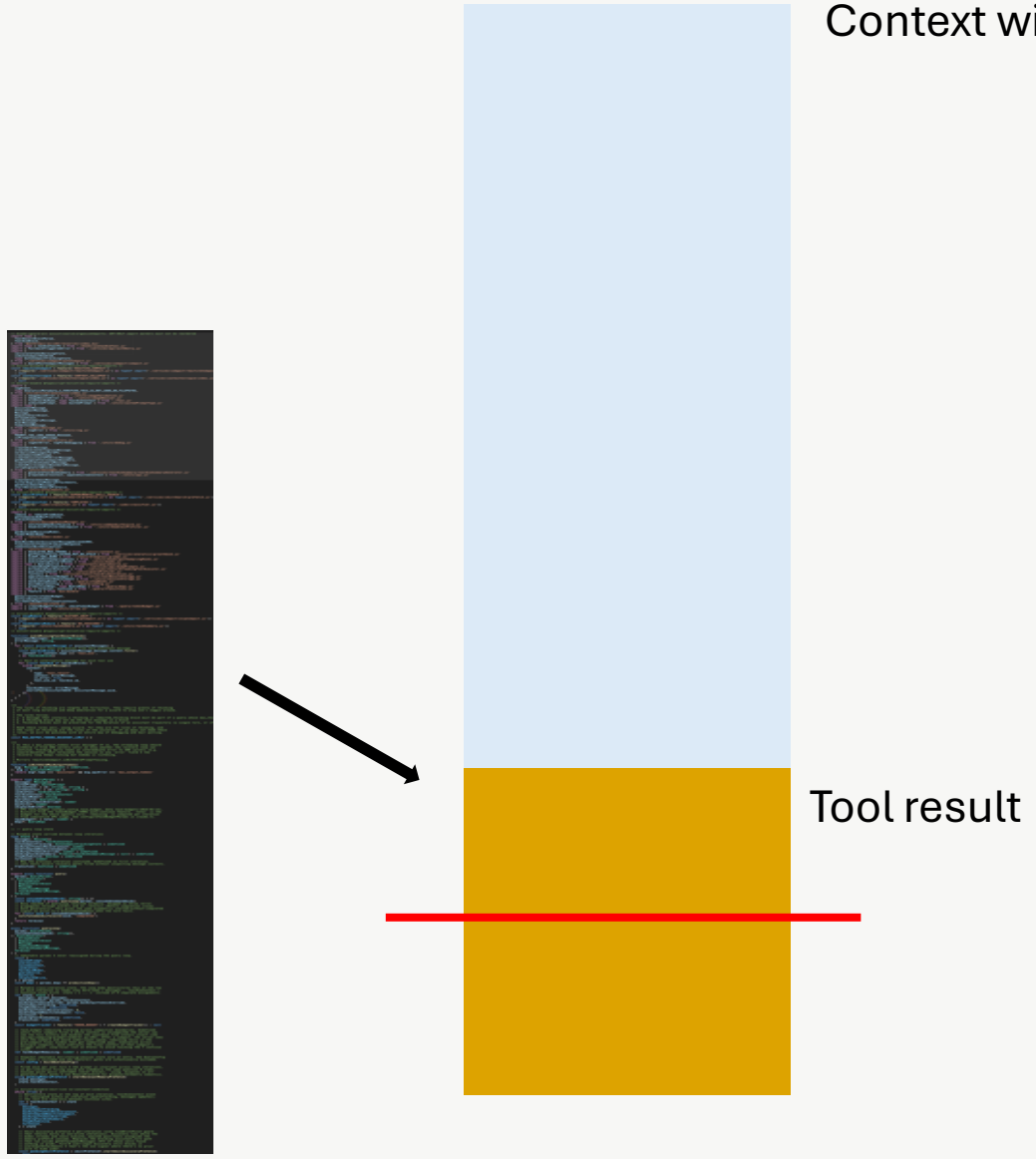
Cut



Compress

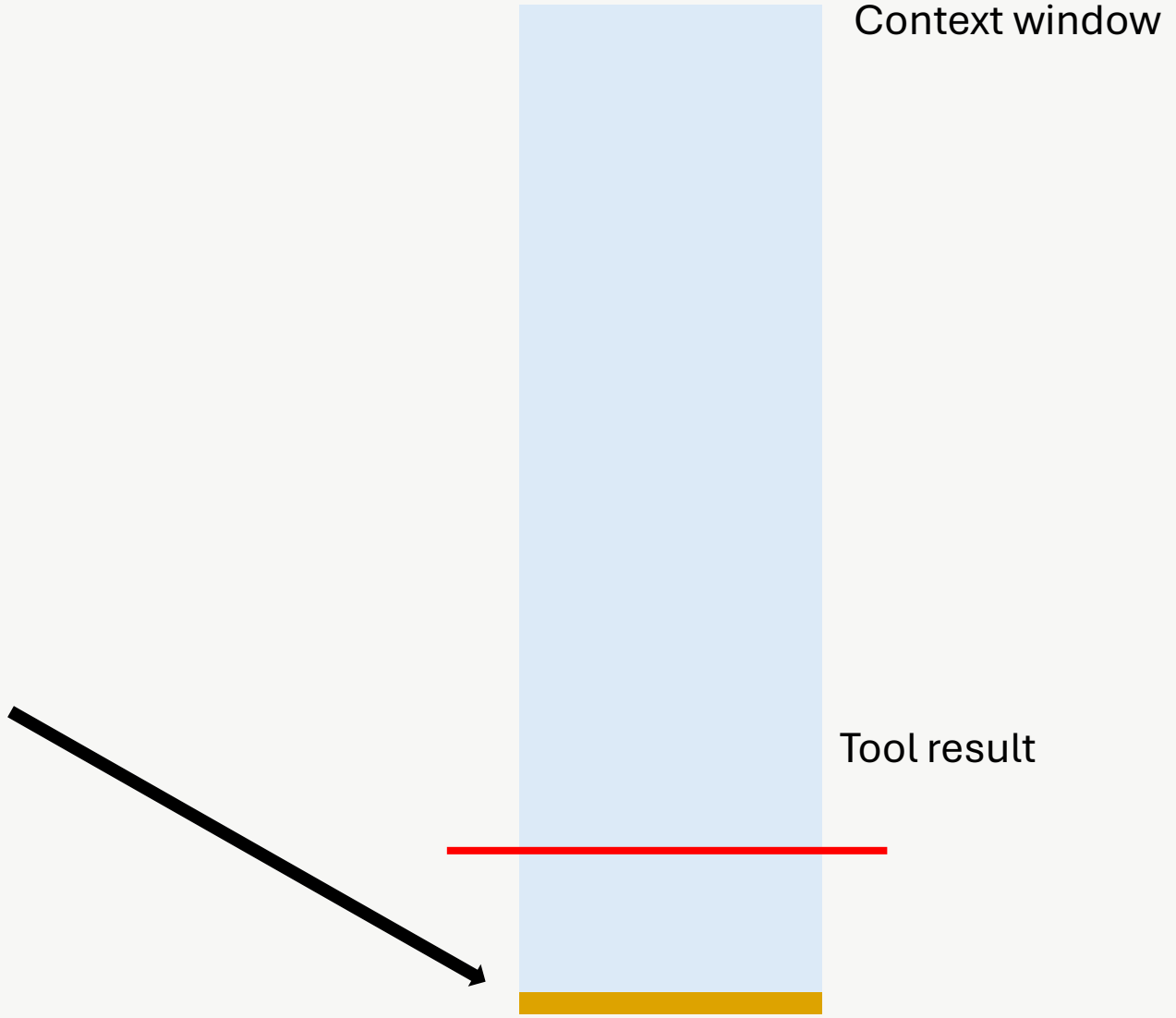
1: Tool Result Budget

- Set length threshold for tool result
- Write into temp files
  - /tmp/XXX
- Keep preview
  - “Saved to /tmp/xxx, 80k in total”



1: Tool Result Budget

- Set length threshold for tool result
- Write into temp files
  - /tmp/XXX
- Keep preview
  - “Saved to /tmp/xxx, 80k in total”
- Read again if needed



2 Tool Count Budget

- Set count threshold for tool results
- Remove old tool call results

Read, Bash, Grep, Glob

Read tools:
Keep 5 recent results

Can redo

Edit, Write

Write tools:
Keep all results

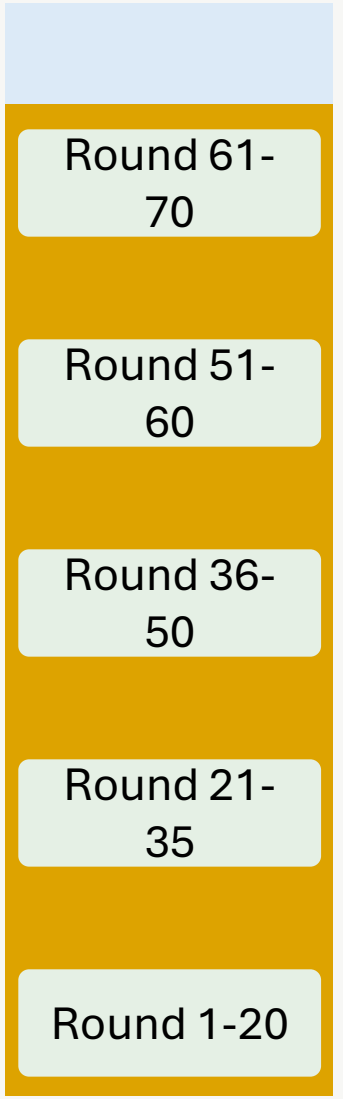
Can NOT redo

Replace Old Read Tool Results

[Old tool result content cleared]

3: Conversation Round Budget

- A round = An assistant message + all tool call and results
- You may explore wrong directions in old rounds
- Remove old rounds



Context window

Current Conversation

3: Conversation Round Budget

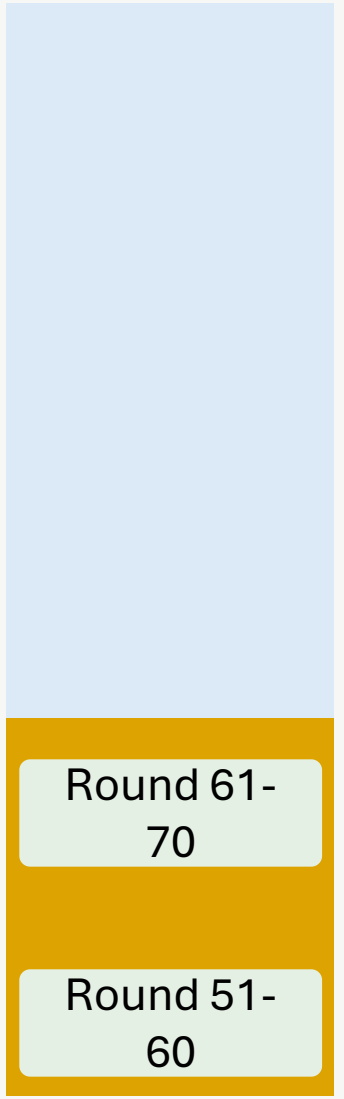
- A round = An assistant message + all tool call and results
- You may explore wrong directions in old rounds
- Remove old rounds

Context window

Current Conversation

3: Conversation Round Budget

- A round = An assistant message + all tool call and results
- You may explore wrong directions in old rounds
- Remove old rounds

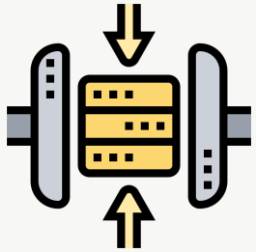


Context window

Current Conversation

5 Strategies

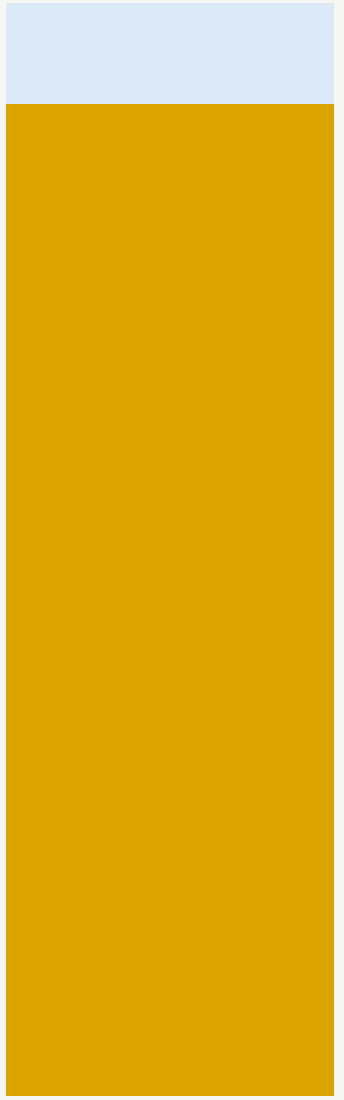
Cut



Compress

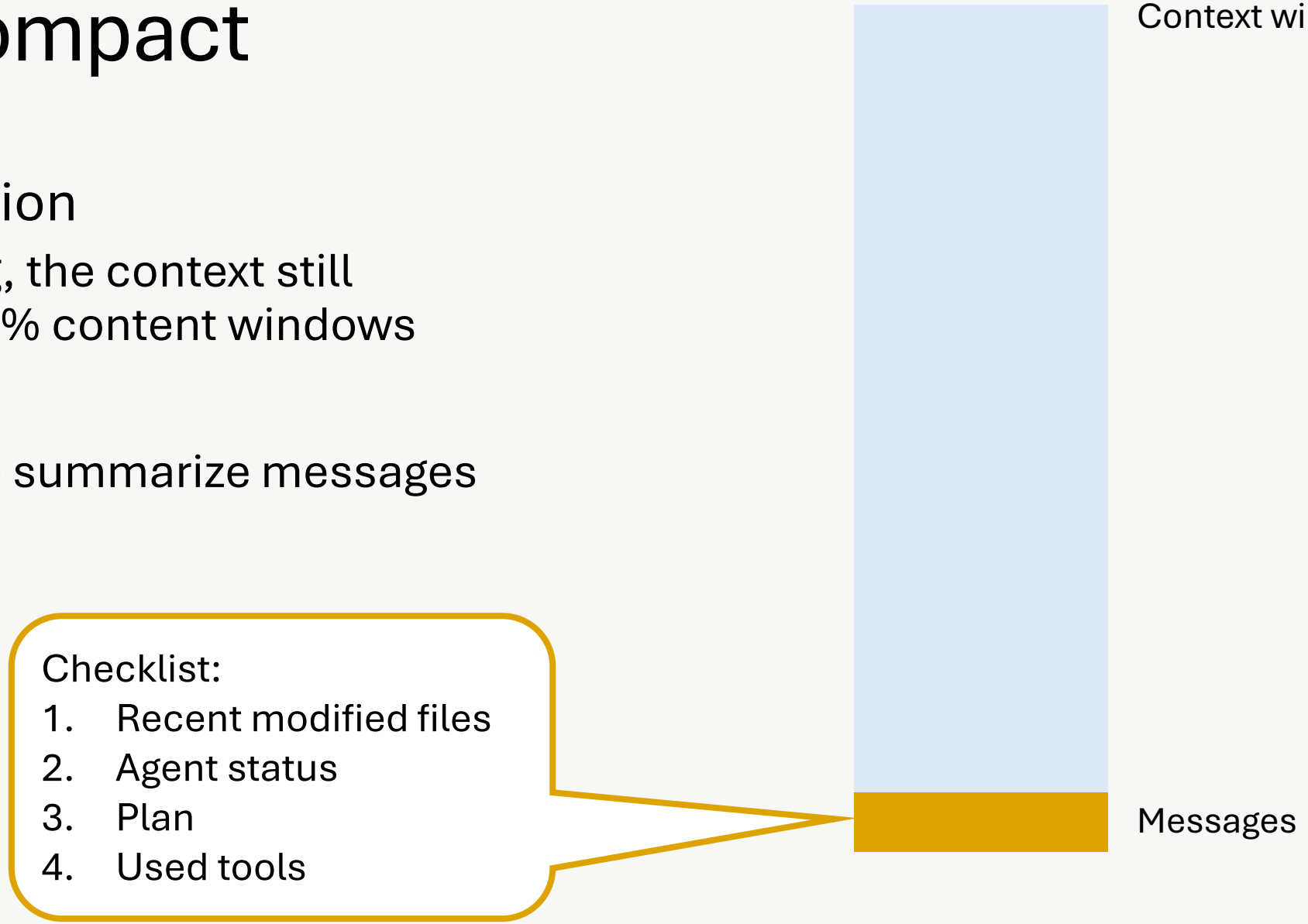
4: Auto Compact

- Trigger condition
  - After cutting, the context still occupies 83% content windows
- Action
  - Call LLMs to summarize messages

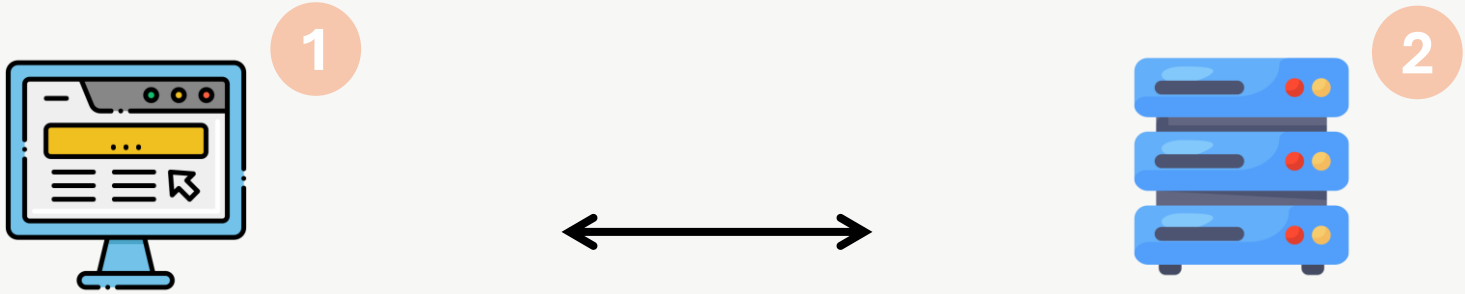


4: Auto Compact

- Trigger condition
  - After cutting, the context still occupies 83% content windows
- Action
  - Call LLMs to summarize messages



4: Auto Compact



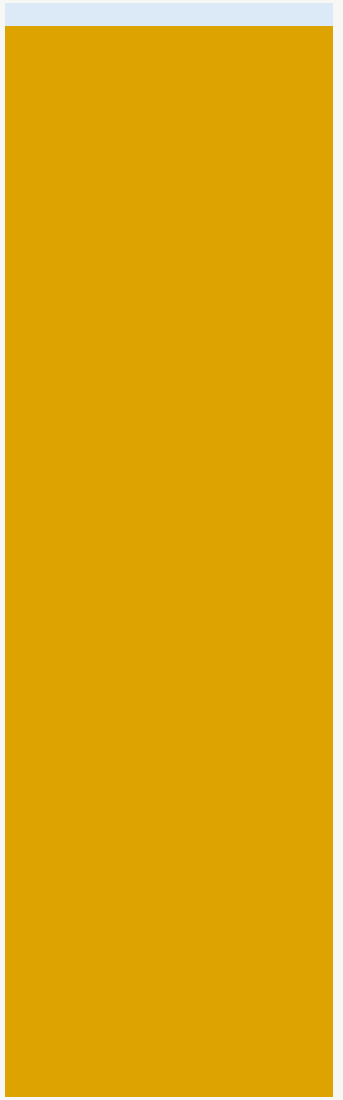
Agent [Client]

LLM [Server]

Compression happens at both agent and LLMs.

5: Context Collapse

- After all strategies, the window is still almost full?
- Save everything into disk



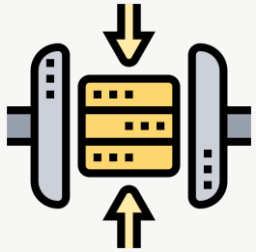
5: Context Collapse

- After all strategies, the window is still almost full?
- Save everything into disk
- Restart the session

Content Management

- Solid and practical engineering methods in industry

Cut



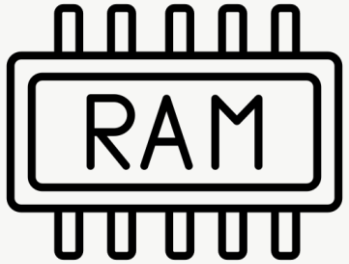
Compress

Content Management

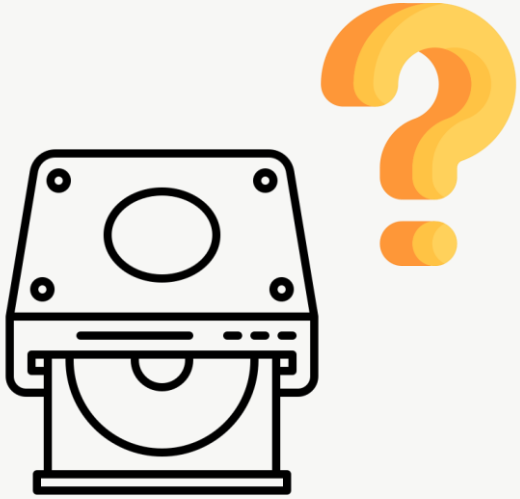
How would you design a content manager?

Short-term VS Long-term Memory

- Context management is short-term memory.

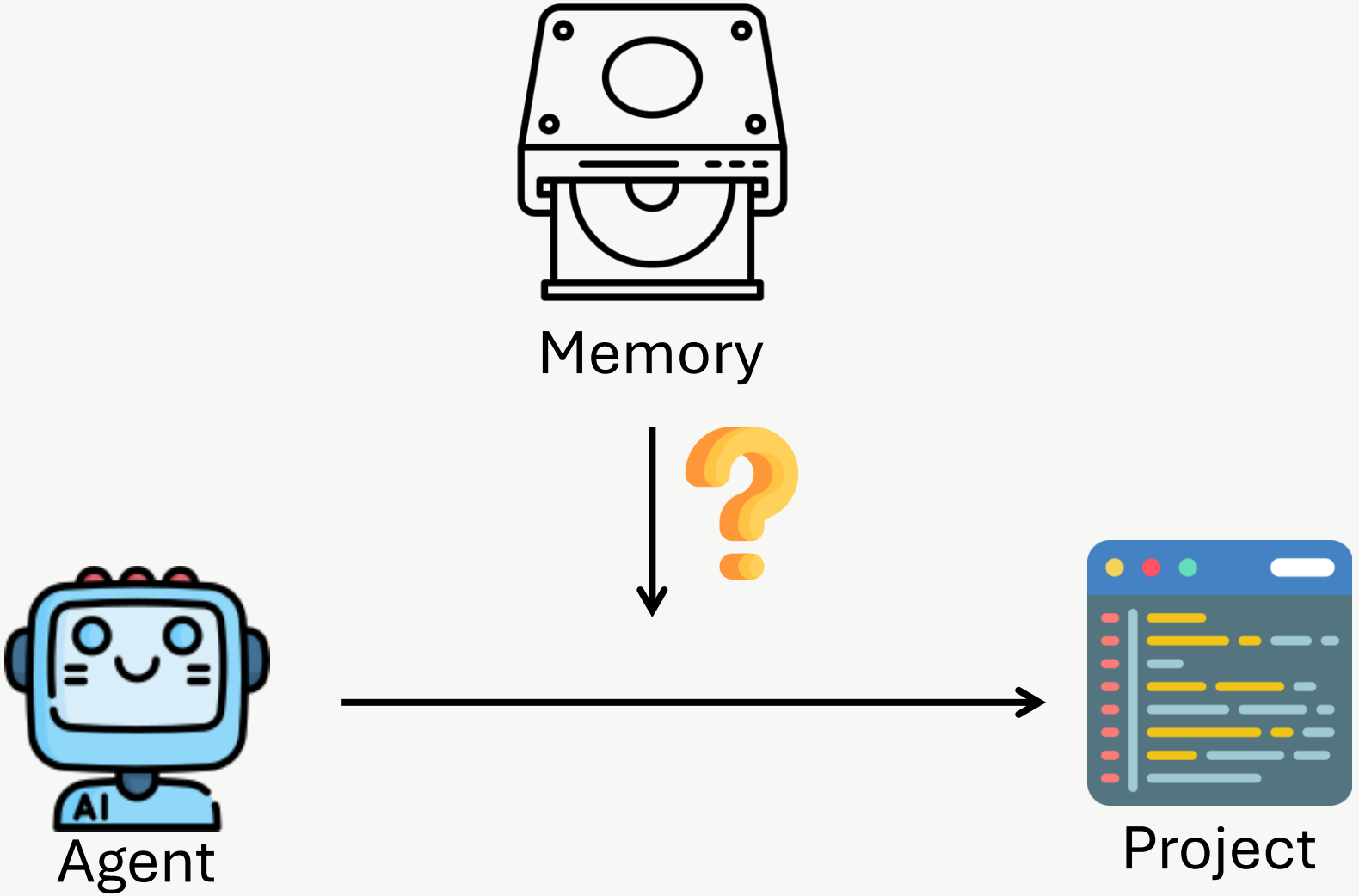


short-term memory



Long-term memory

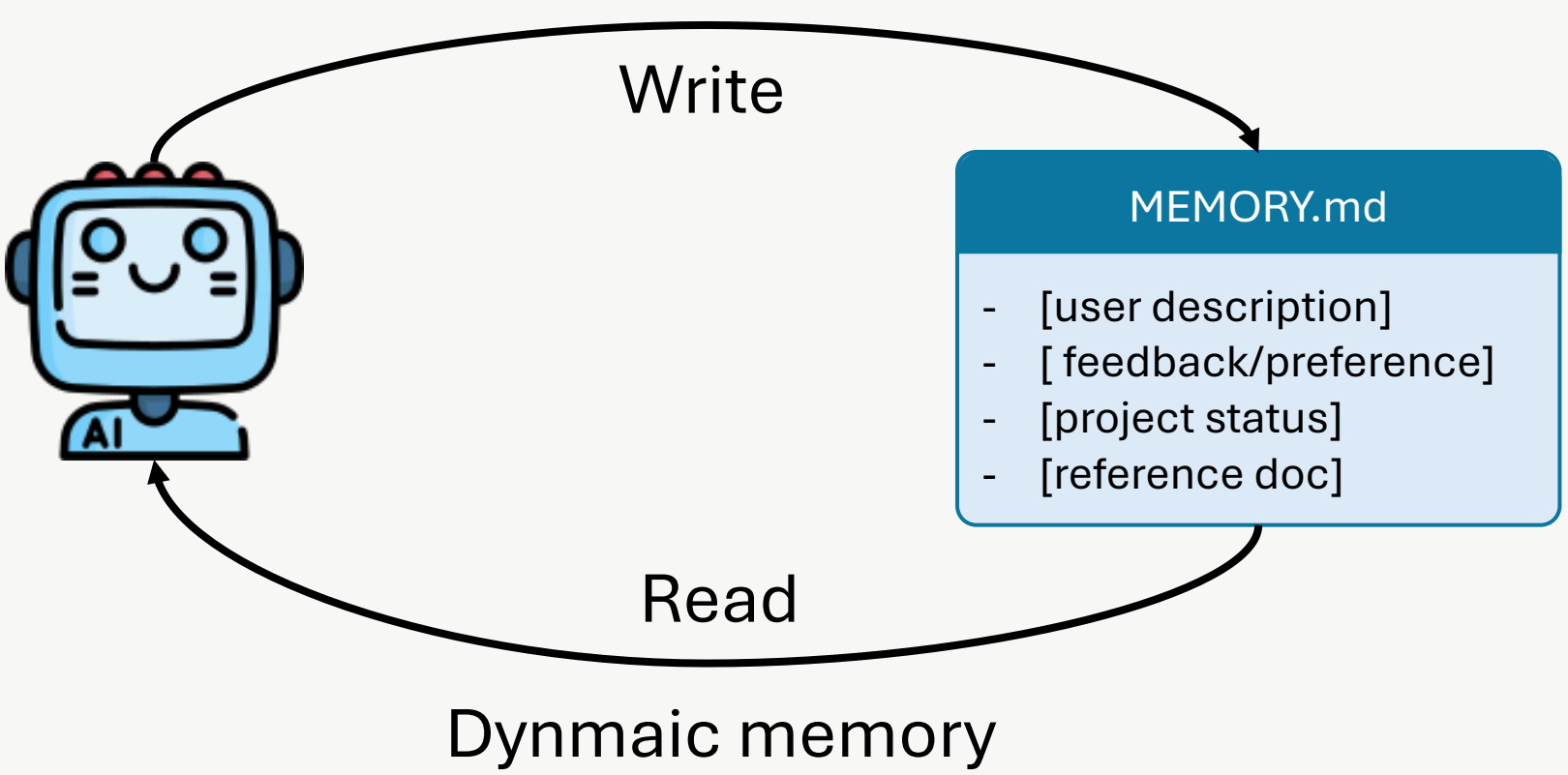
Problem

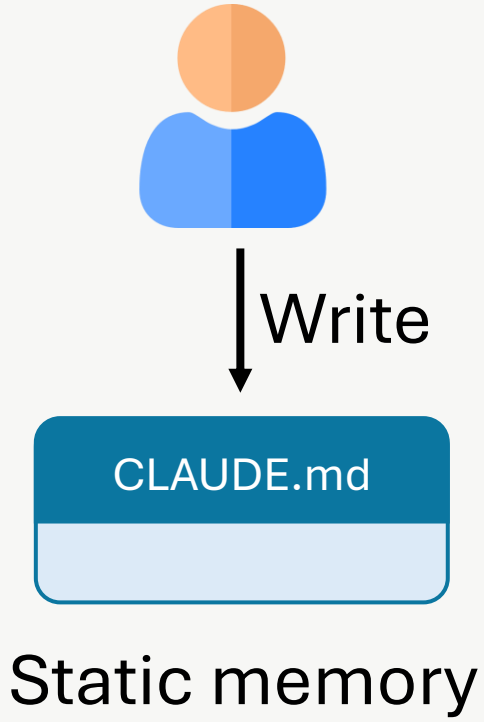


2 Retrieve Augmented Generation

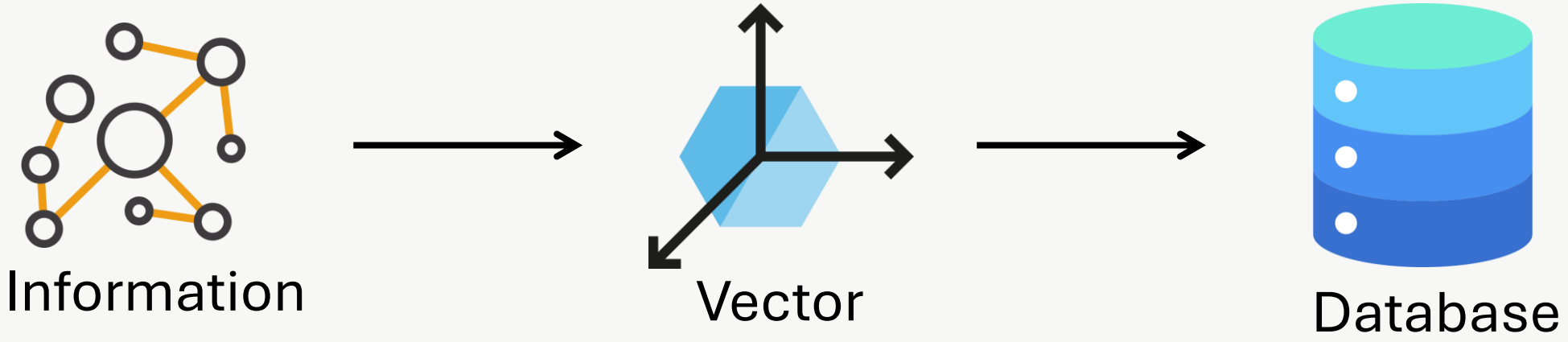
We Already Have it

- We already tools to read and write files for long-term memory.
- But let's talk about some advanced technology.





Vector Database



Example

ID	Text chunk	Embedding Vector (Simplified)
1	For boolean checking functions, use the prefix "has_" followed by the object and property.	[0.12, 0.87, 0.34, 0.66]
2	Each file has most 30 functions.	[0.72, 0.21, 0.91, 0.15]
3	Folder depth at most 5 layers.	[0.31, 0.45, 0.88, 0.20]

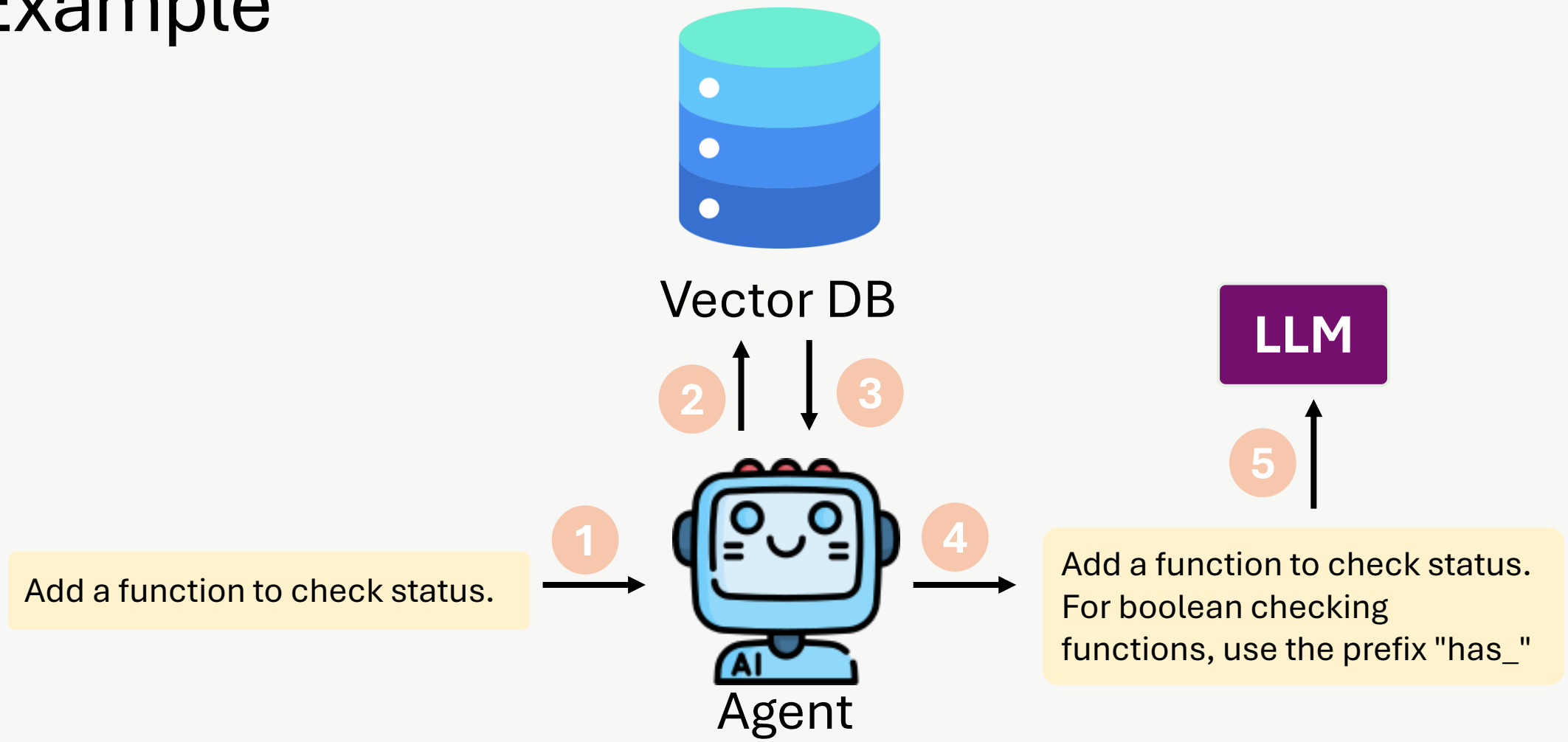
Example

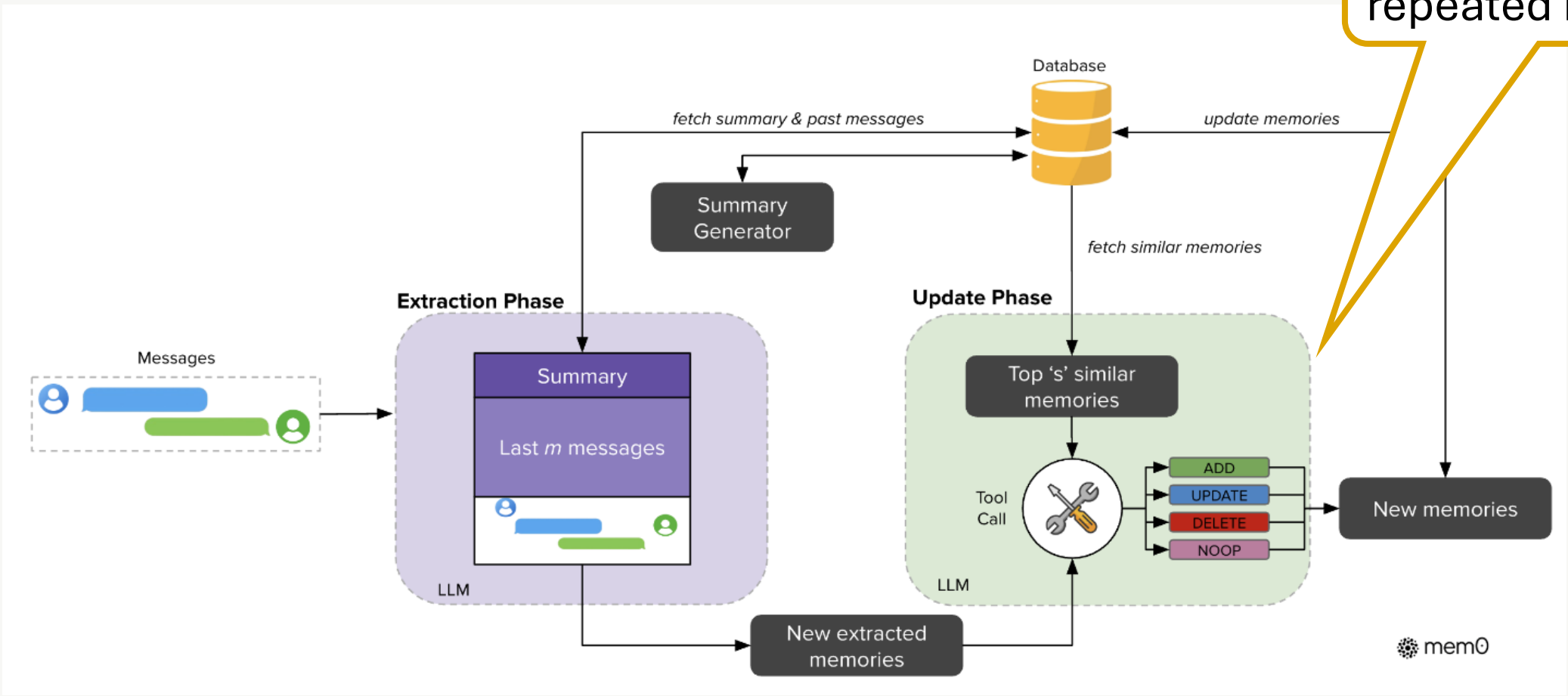
Query: Add a function to check status.
Vector: [0.15, 0.82, 0.38, 0.61]

Top-1 similarity: 0.96

ID	Text chunk	Embedding Vector (Simplified)
1	For boolean checking functions, use the prefix "has_" followed by the object and property.	[0.12, 0.87, 0.34, 0.66]
2	Each file has most 30 functions.	[0.72, 0.21, 0.91, 0.15]
3	Folder depth at most 5 layers.	[0.31, 0.45, 0.88, 0.20]

Example

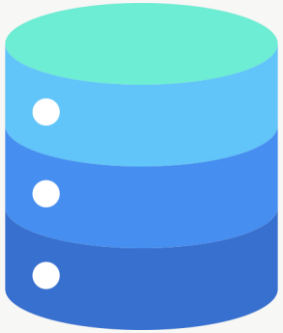

Vector DB
2
3
LLM
5
Add a function to check status.
1
AI
4
Add a function to check status. For boolean checking functions, use the prefix "has_"
Agent

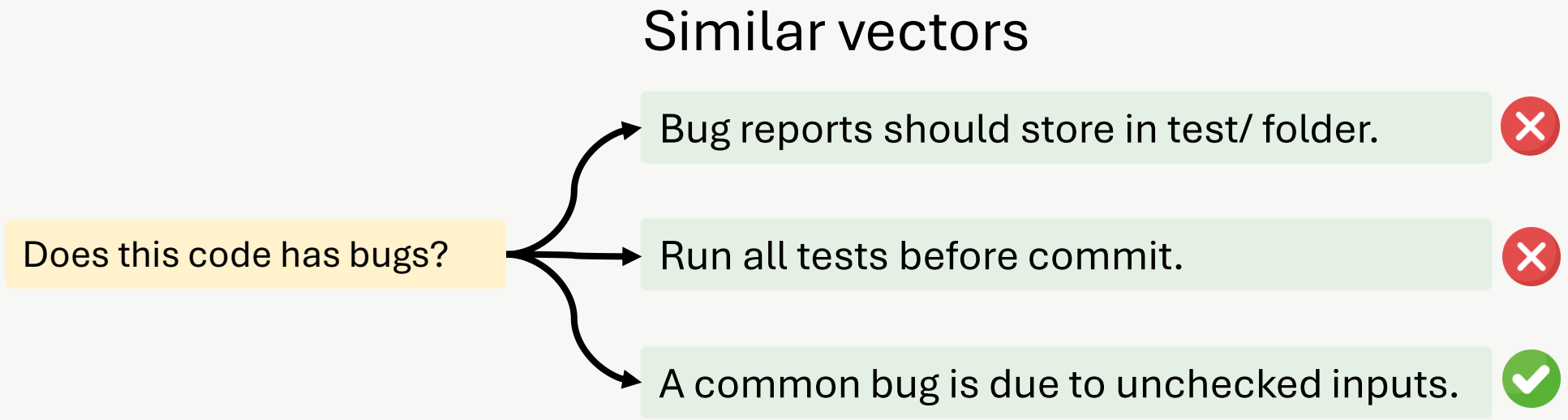


Mem0
https://arxiv.org/pdf/2504.19413

Challenges

- 1. Similarity != Relevance.





Challenges

- 1. Similarity != Relevance.
- 2. Maintain difficulty.

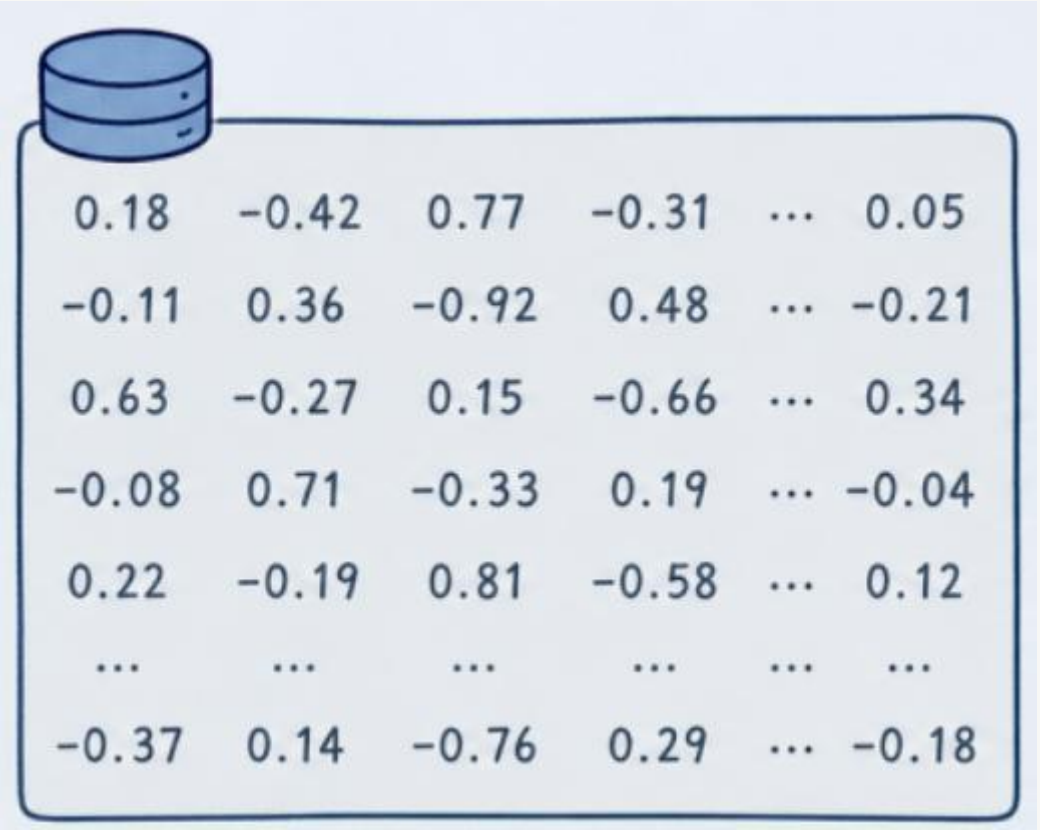
Pinecone

Milvus

Multiple vendors, embedding configuration, chunk size, index update, conflict merge...

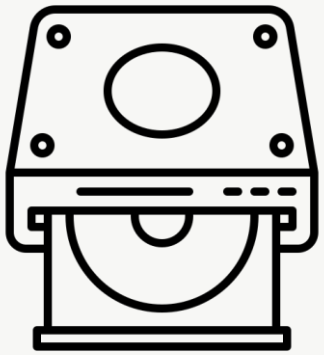
Challenges

- 1. Similarity != Relevance.
- 2. Maintain difficulty.
- 3. Unreadable.



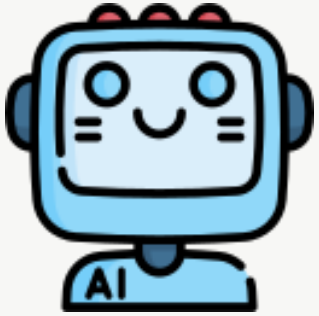
Practice Time

Beyond Memory



Memory

Security Policy



Agent

Project

MEMORY.md

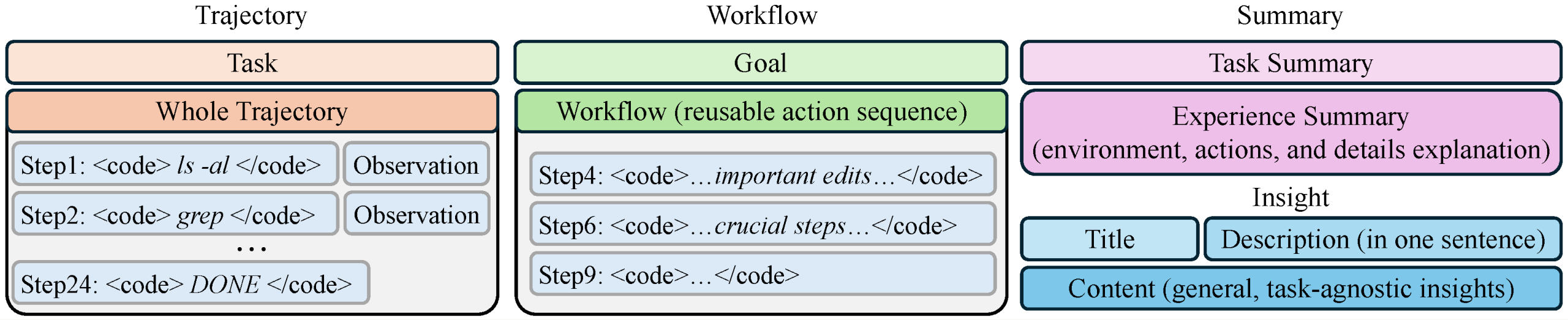
- [user description]
- [feedback/preference]
- [project status]
- [reference doc]

Vector Database

No absolute answer yet

What is a Good Memory Design?

Structural Info May Be Helpful



https://memorytransfer.github.io/

0

Raw conversation

Three sessions of LOCOMO conv-30, twelve days apart — the unstructured input the agent receives.

LOCOMO conv-30 — verbatim excerpts

```
SESSION 1  ·  20 January, 2023
Gina: Hey Jon! Good to see you. What's up? Anything new?
Jon:  Lost my job as a banker yesterday, so I'm gonna take
      a shot at starting my own business.
Gina: Sorry about your job Jon! Unfortunately, I also lost
      my job at Door Dash this month.
Jon:  I'm starting a dance studio 'cause I'm passionate
      about dancing and it'd be great to share it.

SESSION 2  ·  29 January, 2023
Jon:  On the hunt for the ideal spot for my dance studio...
      I even found a place with great natural light!

SESSION 3  ·  1 February, 2023
Gina: I emailed some wholesalers and one said yes today!
      Now I can expand my clothing store.
```

https://01.me/research/user-as-code/

1 Phase 1 · Memorize

An LLM extracts every fact as a flat string and **appends** it — never overwriting. Relative dates are resolved against the session timestamp: **“lost my job yesterday” → 2023-01-19.** (Mem0 keeps the literal “yesterday” and answers the date wrong.)

phase-1 output — facts.py (indexed in ChromaDB)

```
facts = [   # append-only · 1,419 facts extracted from conv-30's 19 sessions
    "[session_1, 4:04 pm on 20 January, 2023] Jon lost his job on January 19, 2023.",
    "[session_1, 4:04 pm on 20 January, 2023] Jon's former job was as a banker.",
    "[session_1, 4:04 pm on 20 January, 2023] Gina lost her job in January 2023.",
    "[session_1, 4:04 pm on 20 January, 2023] Jon is starting a dance studio.",
    "[session_2, 2:32 pm on 29 January, 2023] The studio location Jon found is in a downtown area.",
    "[session_2, 2:32 pm on 29 January, 2023] Jon believes that good flooring is crucial for a dance studio.",
    # ... 1,413 more facts across all 19 sessions ...
]
```

https://01.me/research/user-as-code/

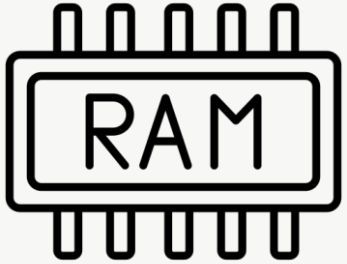
2 Phase 2 · Structure

An LLM regenerates the **entire** typed Python from the full fact list — dates become `date()` objects, repeated entities become typed lists, leftovers go to `notes`. This **jon** object is what the next pipeline queries.

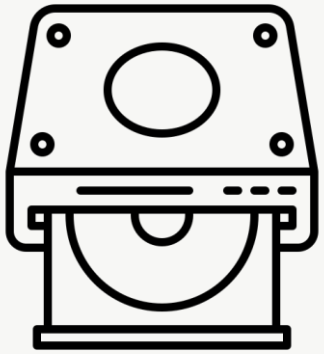
phase-2 output — state.py (the user, as code)

```python
@dataclass
class Person:
    name: str
    passions: list[str] = field(default_factory=list)
    skills: list[str] = field(default_factory=list)
    jobs: list["JobHistory"] = field(default_factory=list)
    notes: list[str] = field(default_factory=list)

jon = Person(
    name="Jon",
    passions=["Dancing (Contemporary, Hip-hop)", "Sharing dance with others"],
    skills=["Choreography", "Contemporary dance", "Hip-hop"],
)
jon.jobs.append(JobHistory(title="Banker", date_ended=date(2023, 1, 19),
                           notes=["Lost job on Jan 19, 2023"]))

jon_studio = Business(
    name="Jon's Dance Studio", owner="Jon", industry="Dance Studio",
    status="Establishing / Searching for location",
    location_description="Downtown area (potential), ideally by the water",
    features=["Marley flooring (planned for grip/durability)", "Natural light"],
)
```

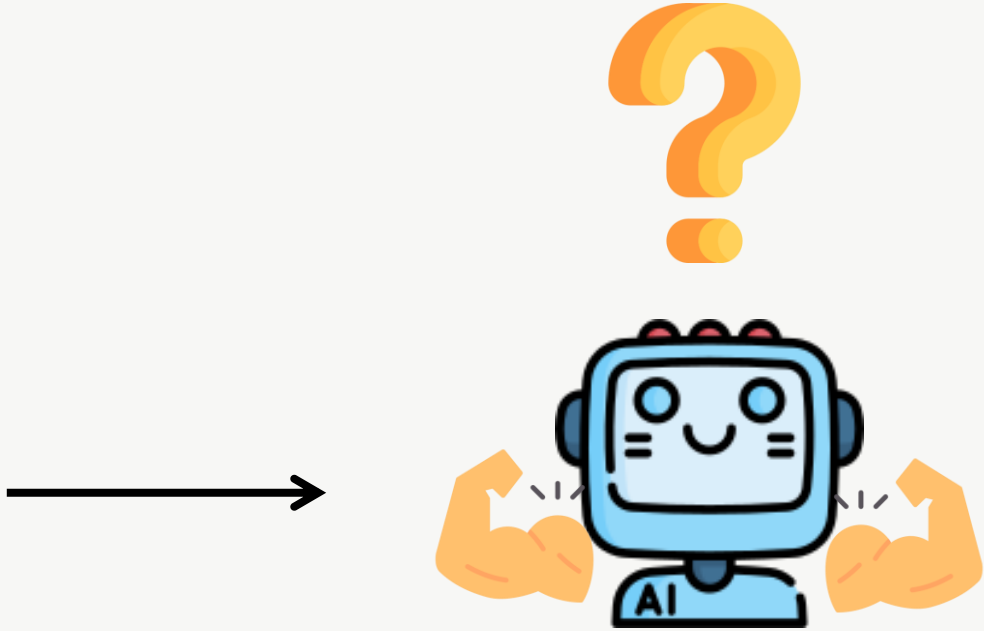
https://01.me/research/user-as-code/



short-term memory



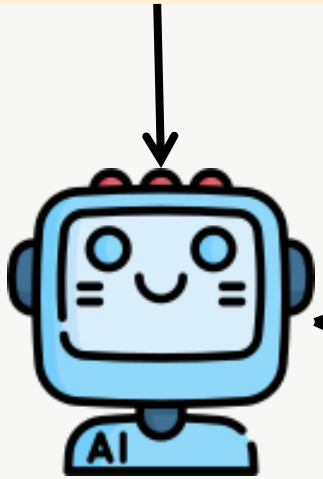
Long-term memory



Agent

Problem: Context Pollution

Add a function to calculate the discount price for premium users. Generate code + tests.




```python
def discount_price(price, is_premium):
    return price * 0.8
```

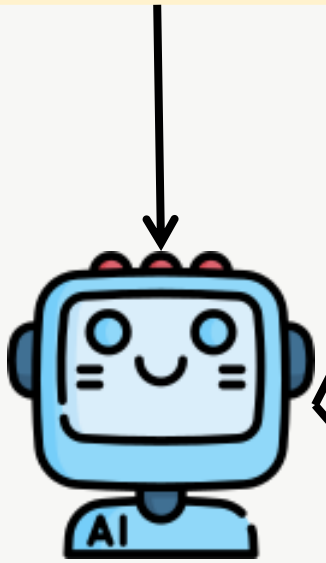
```python
def test_discount():
    assert discount_price(100, True) == 80
```

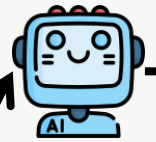
Passed testing with incomplete test cases.

3 Subagents

Problem: Context Pollution

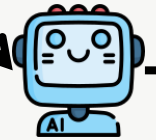
Add a function to calculate the discount price for premium users. Generate code + tests.






```python
def discount_price(price, is_premium):
    return price * 0.8
```

Generate the code to calculate discount price.

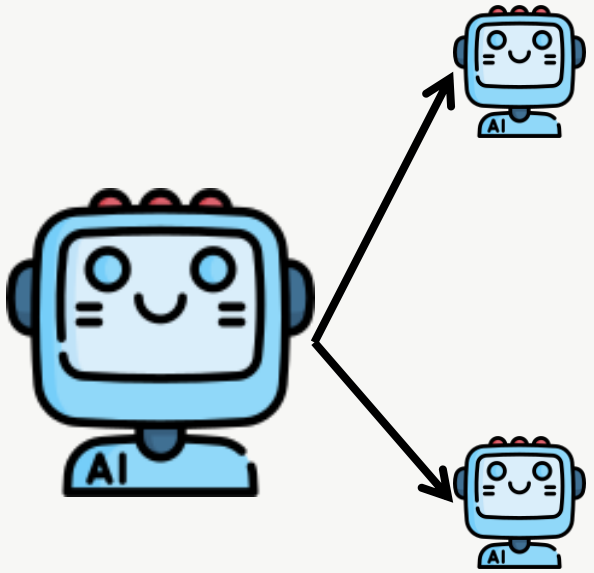



```python
def test_discount():
    assert discount_price(100, True)==80
    assert discount_price(100, False)==100
```

Generate the test for discount calculation.

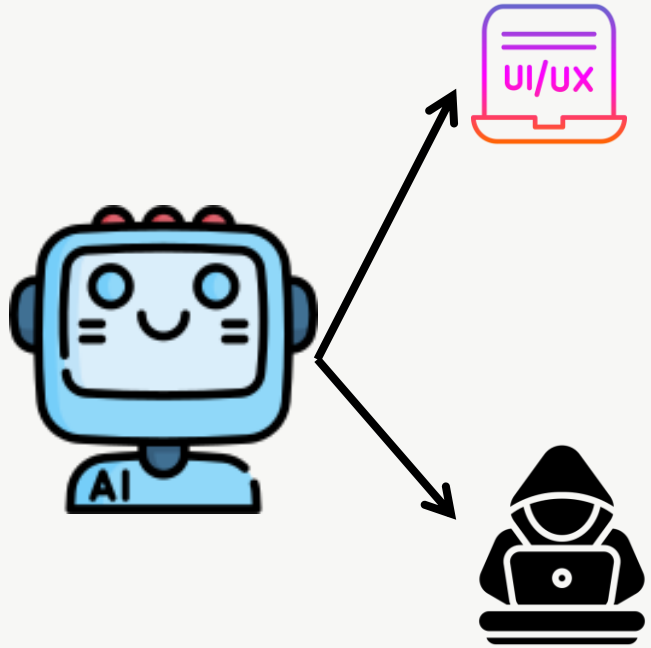
Three Advantages of Subagents

- Context isolation
  - Isolated context window, tools, and so on.



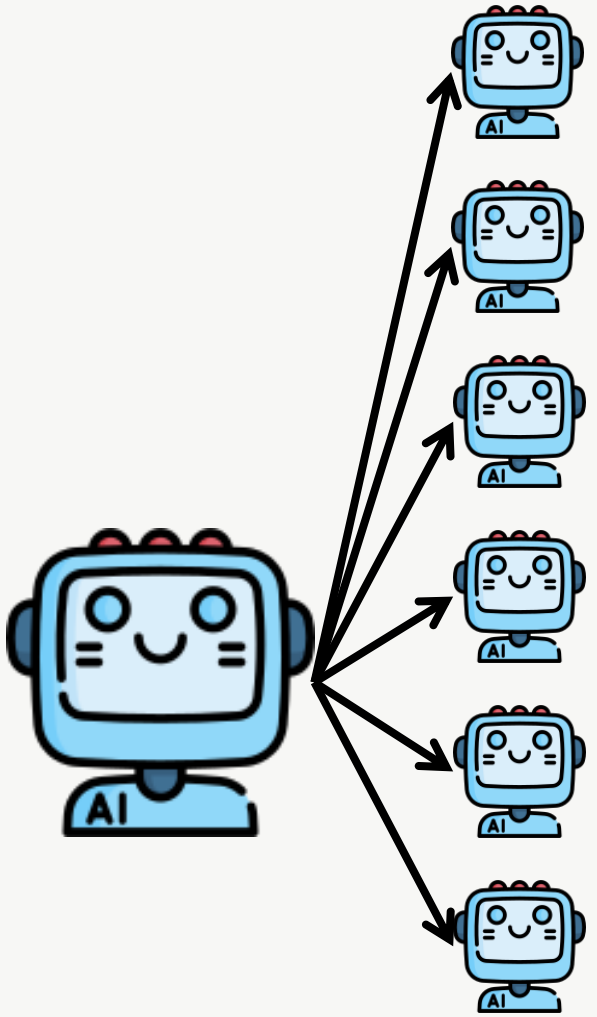
Three Advantages of Subagents

- Context isolation
  - Isolated context window, tools, and so on.
- Specialization
  - UX designer expert
  - Security expert



Three Advantages of Subagents

- Context isolation
  - Isolated context window, tools, and so on.
- Specialization
  - UX designer expert
  - Security expert
- Parallelization

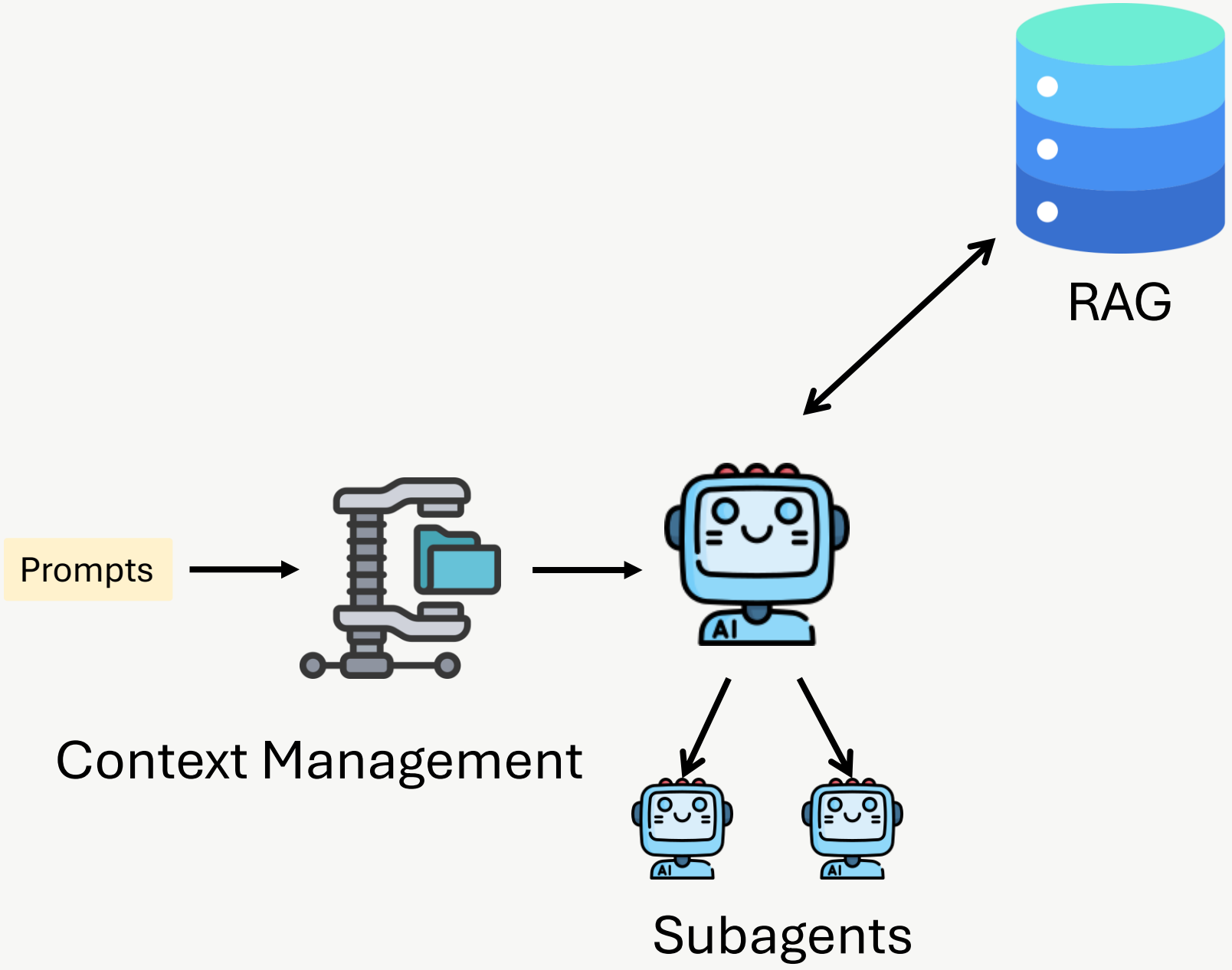


Predefined Subagent

```
---
name: code-reviewer
description: Reviews code for quality and best practices
tools: Read, Glob, Grep
model: sonnet
---

You are a code reviewer. When invoked, analyze the code and provide
specific, actionable feedback on quality, security, and best practices.
```

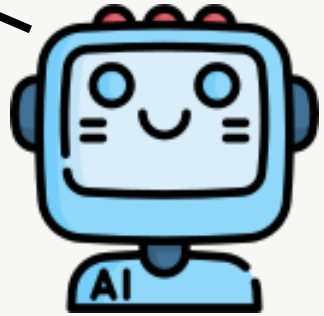
Use markdown files.

https://code.claude.com/docs/en/sub-agents


RAG
Prompts
AI
Context Management
AI
AI
Subagents

Next Problem in Next Lecture

Context is well managed.



Predefined tools are restricted.

香港中文大學（深圳）
The Chinese University of Hong Kong, Shenzhen

Questions

Next class: MCP & Skills