

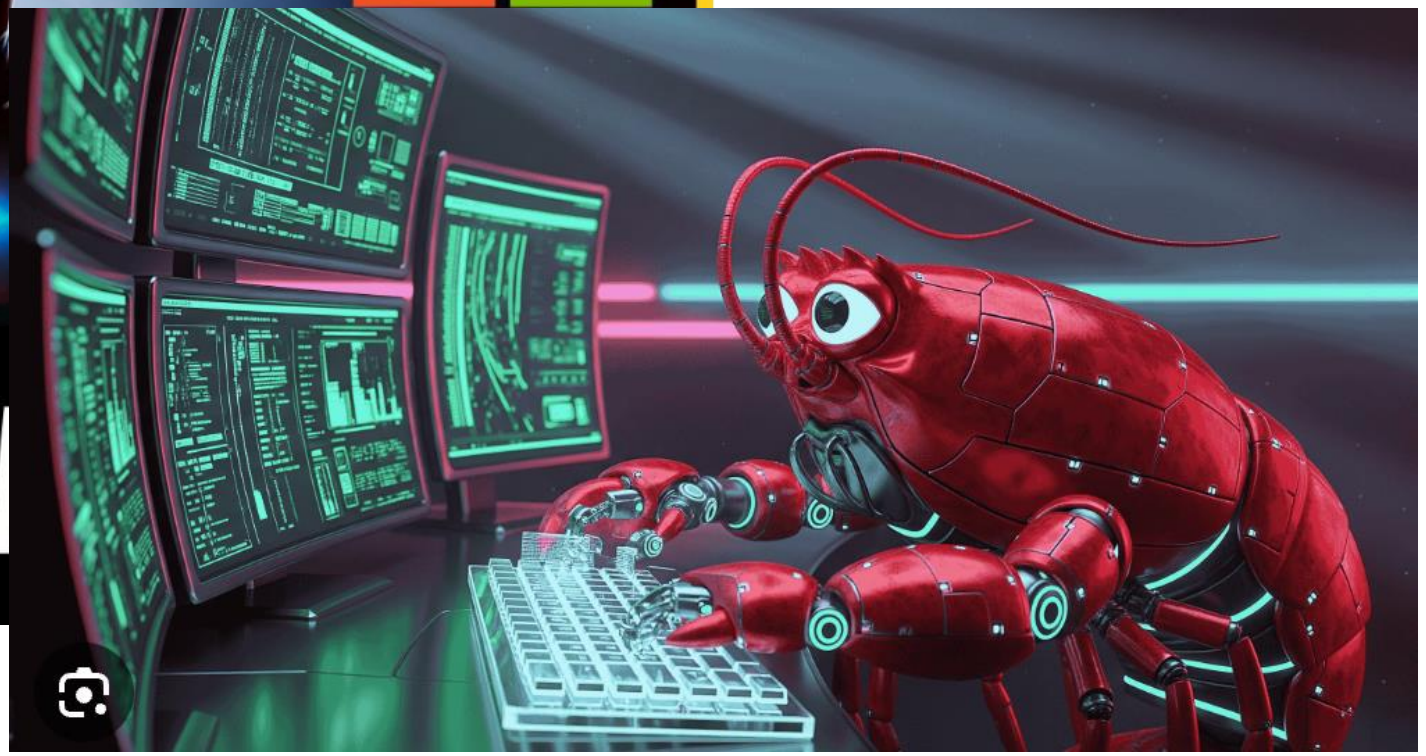
CS 329X: Human Centered LLMs

Building Full-Stack Human-Centered AI Systems

Yijia Shao

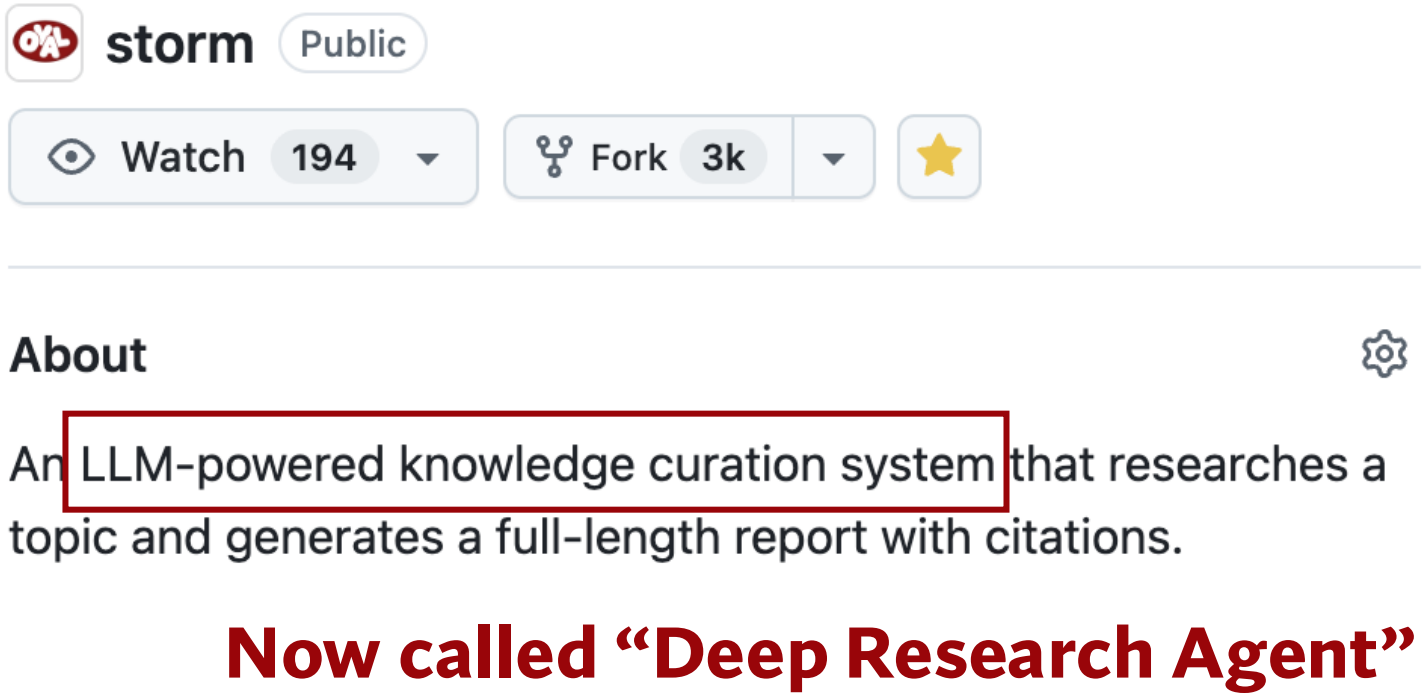
Agents are very trendy these days





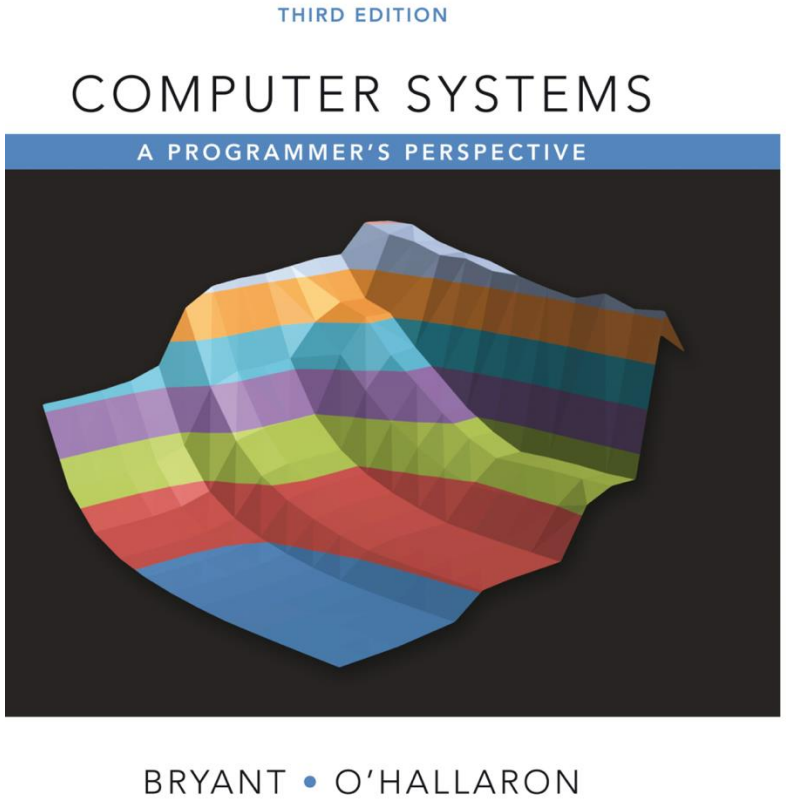
Agents are very hot these days, but…

- Back in 2023, we use the term LLM-empowered system because people worry “agent” is not accurate.



Now called “Deep Research Agent”

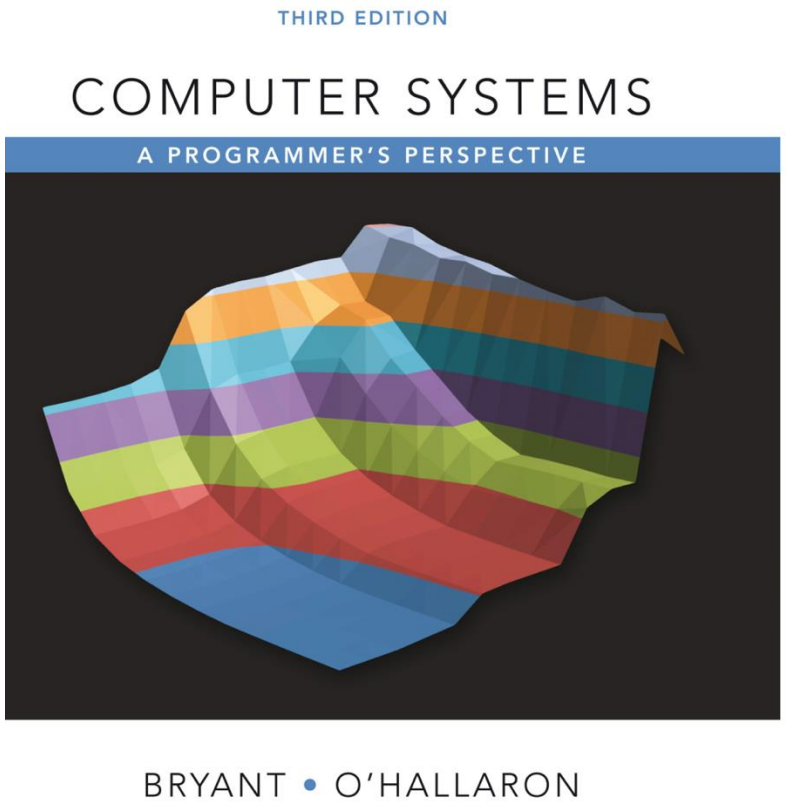
Systems Thinking

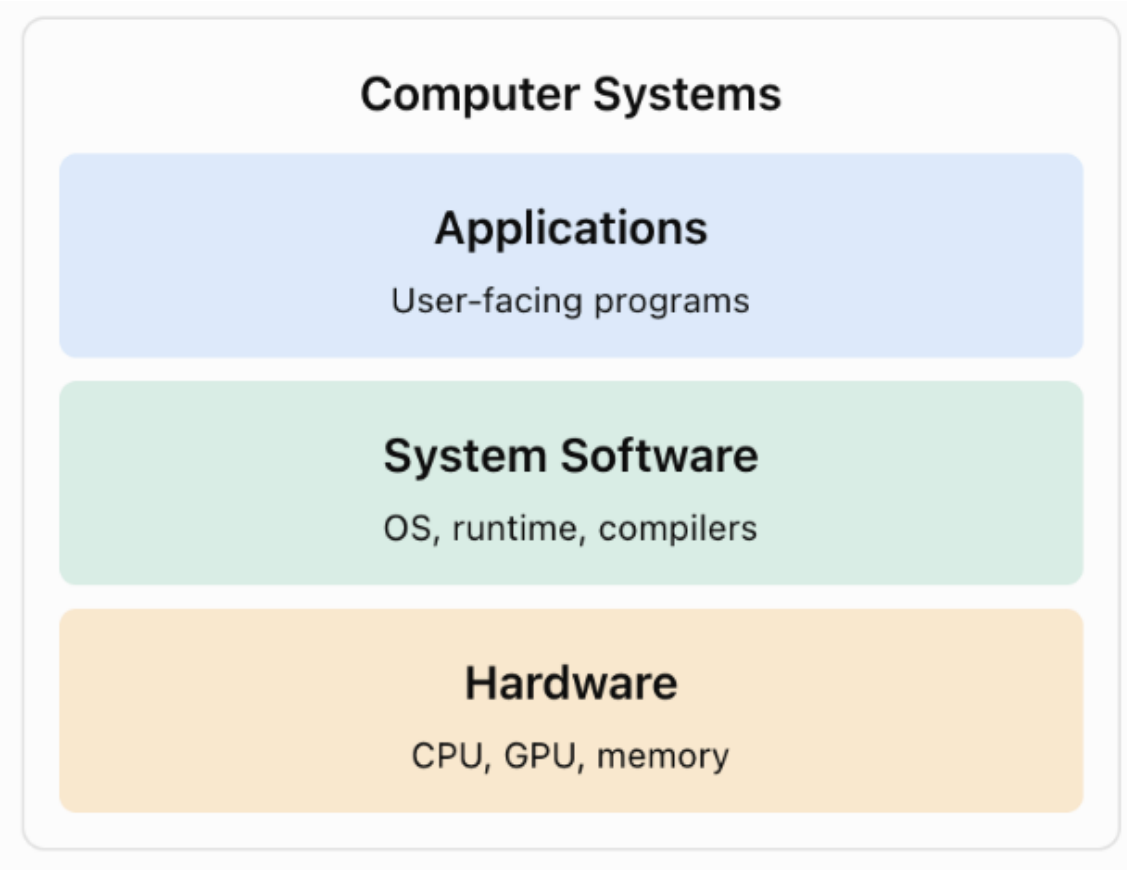


"Digital computers are themselves more complex than most things people build."

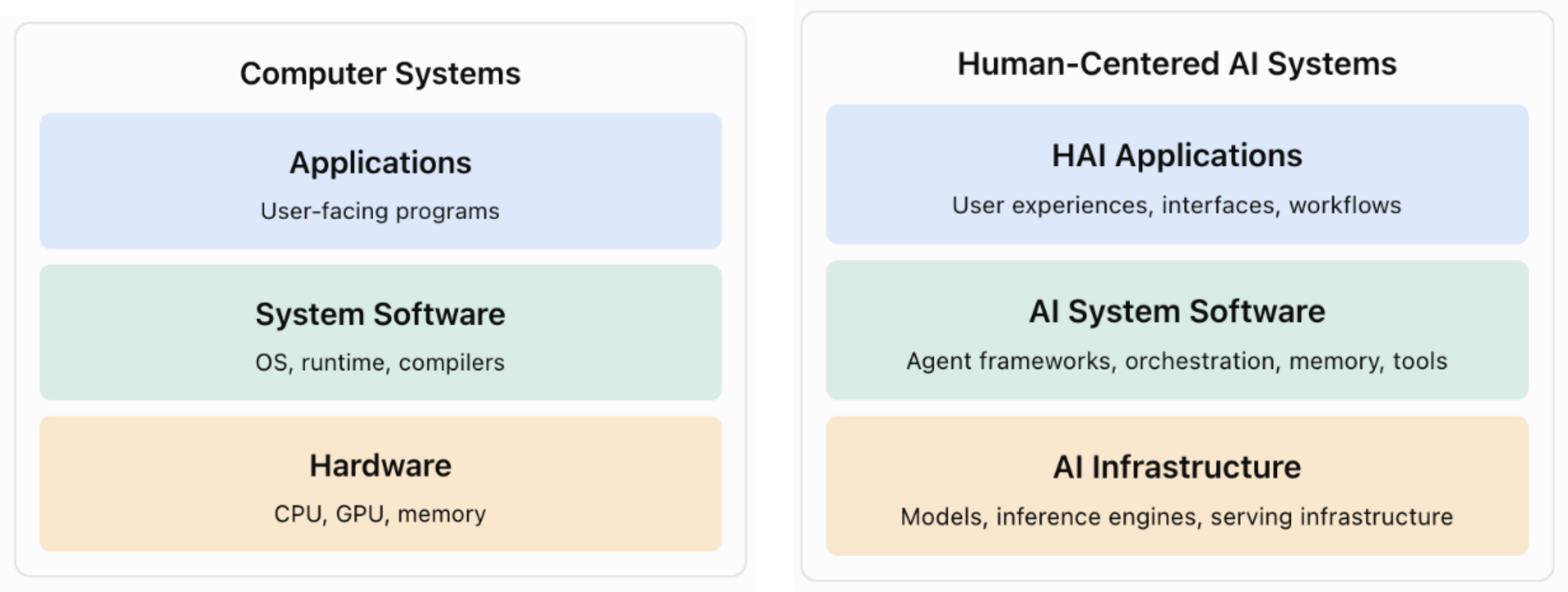
--No Silver Bullet: Essence and Accidents of Software Engineering, Fred Brooks, 1986

Systems Thinking





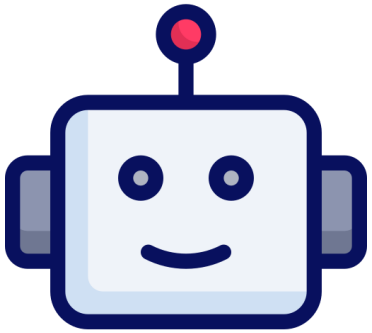
Full-Stack Human-Centered AI Systems

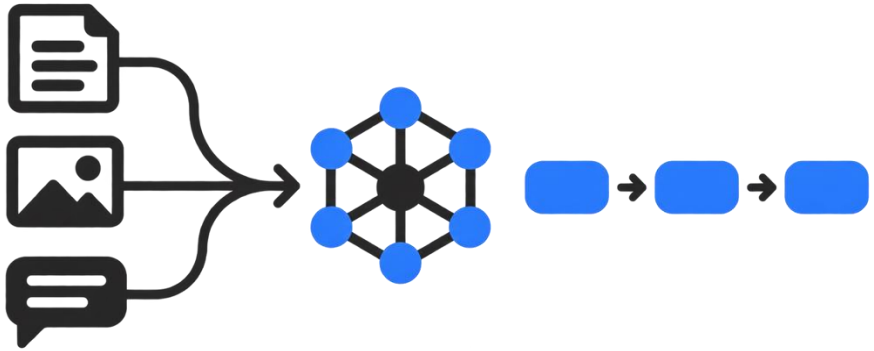


This lecture

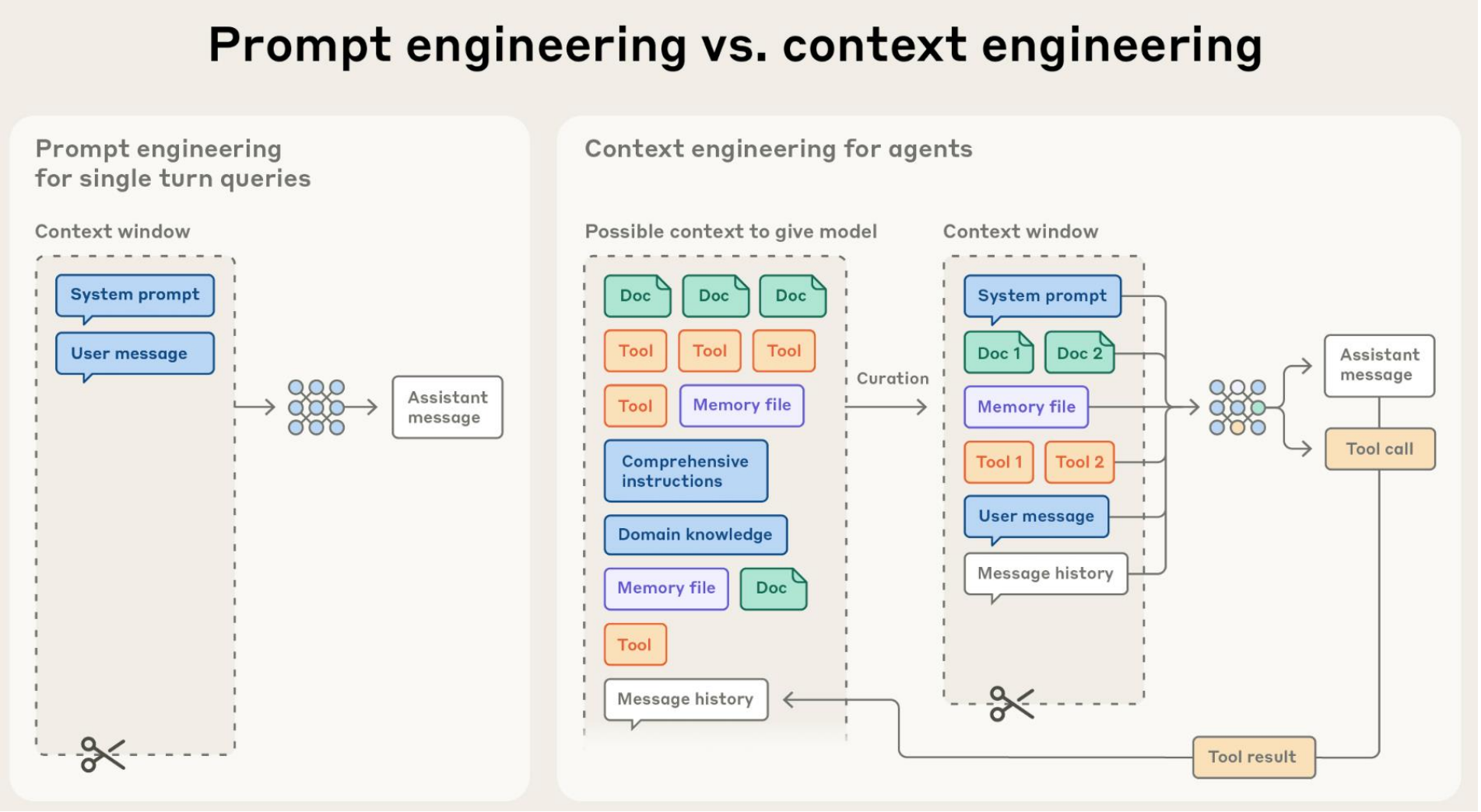
Unpack a few principles behind building full-stack HAI Systems

Lesson 1: Intelligence is Token + Context



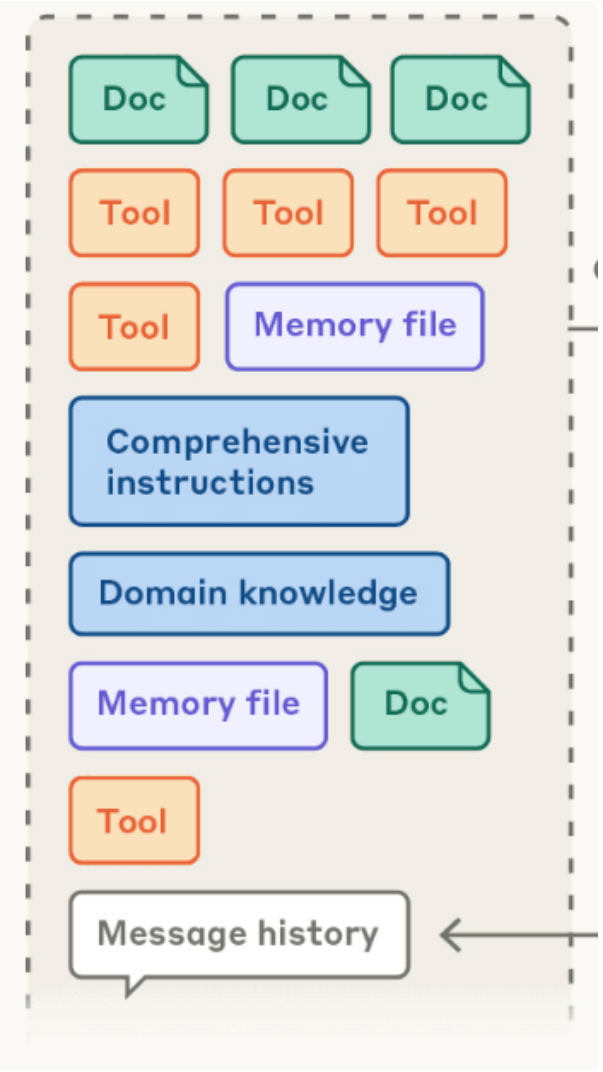


What Does Context Consist of?



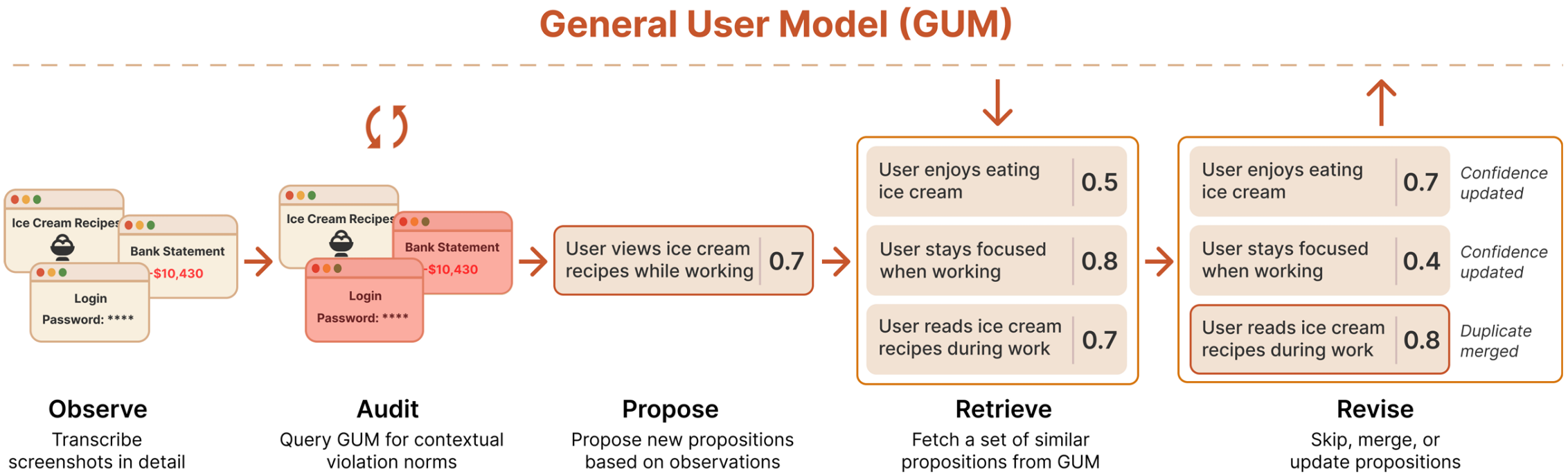
https://www.anthropic.com/engineering/effective-context-engineering-for-ai-agents

How To Acquire Contexts From Users?



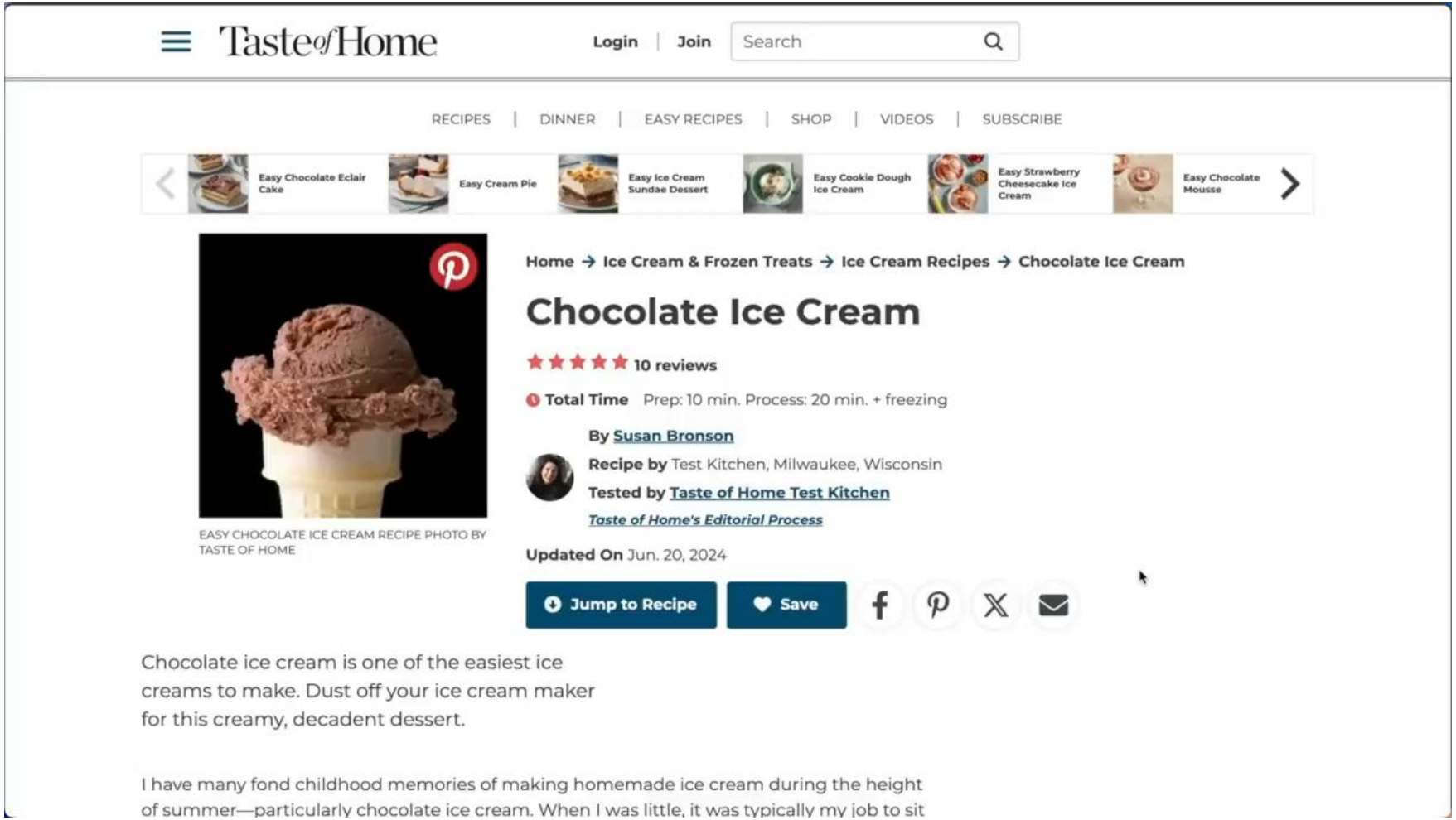
- **Two directions:**
  - Observe
  - Elicit (aka, ask user questions)

Acquire Context Through Observation



Creating General User Models from Computer Use, Shaikh et al., UIST 2025

Acquire Context Through Observation



Creating General User Models from Computer Use, Shaikh et al., UIST 2025

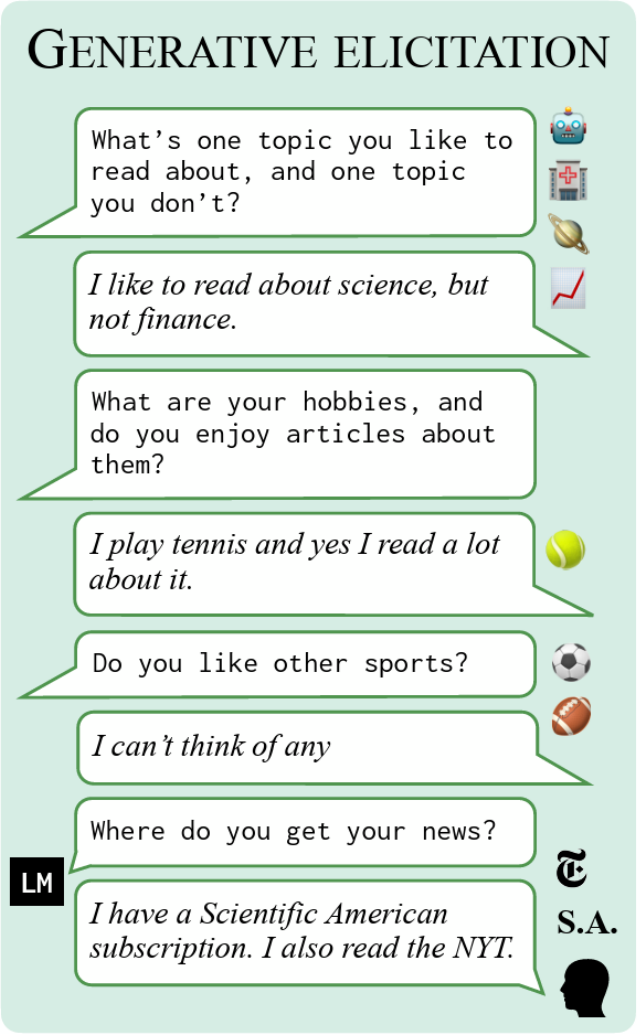
Acquire Context Through Elicitation

- Objective:

  α·human–predictor alignment −

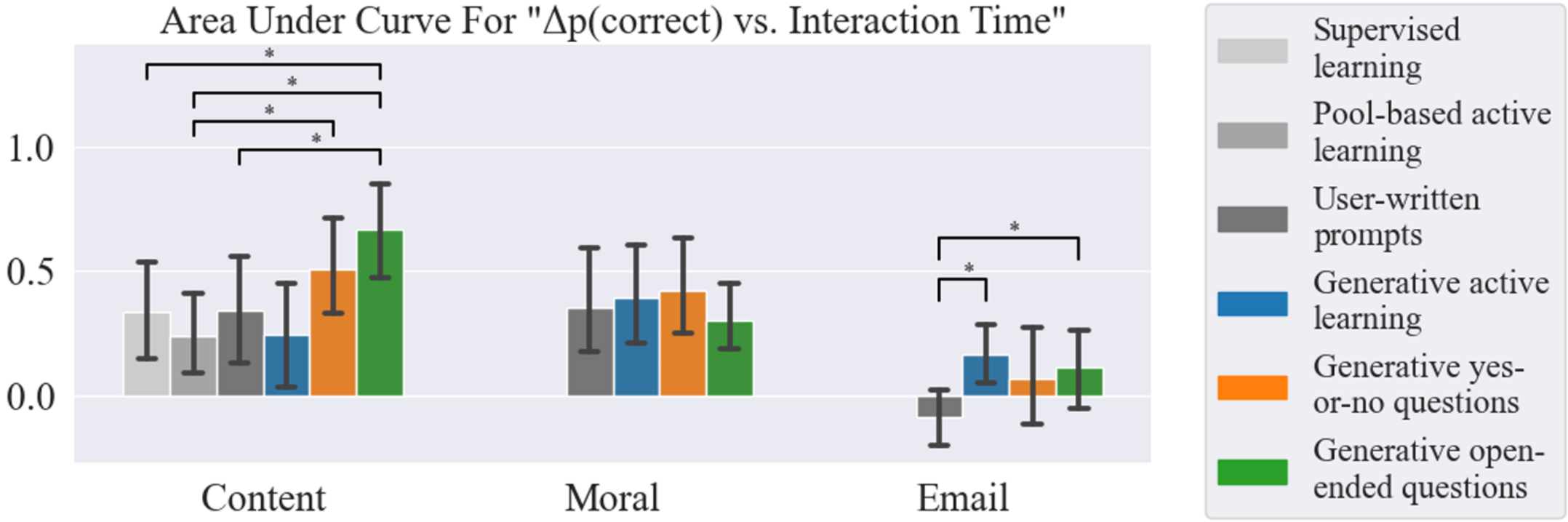
  β · specification cost

- Method: Prompting an LM to ask the user questions while conditioning on the history of previous questions and answers.

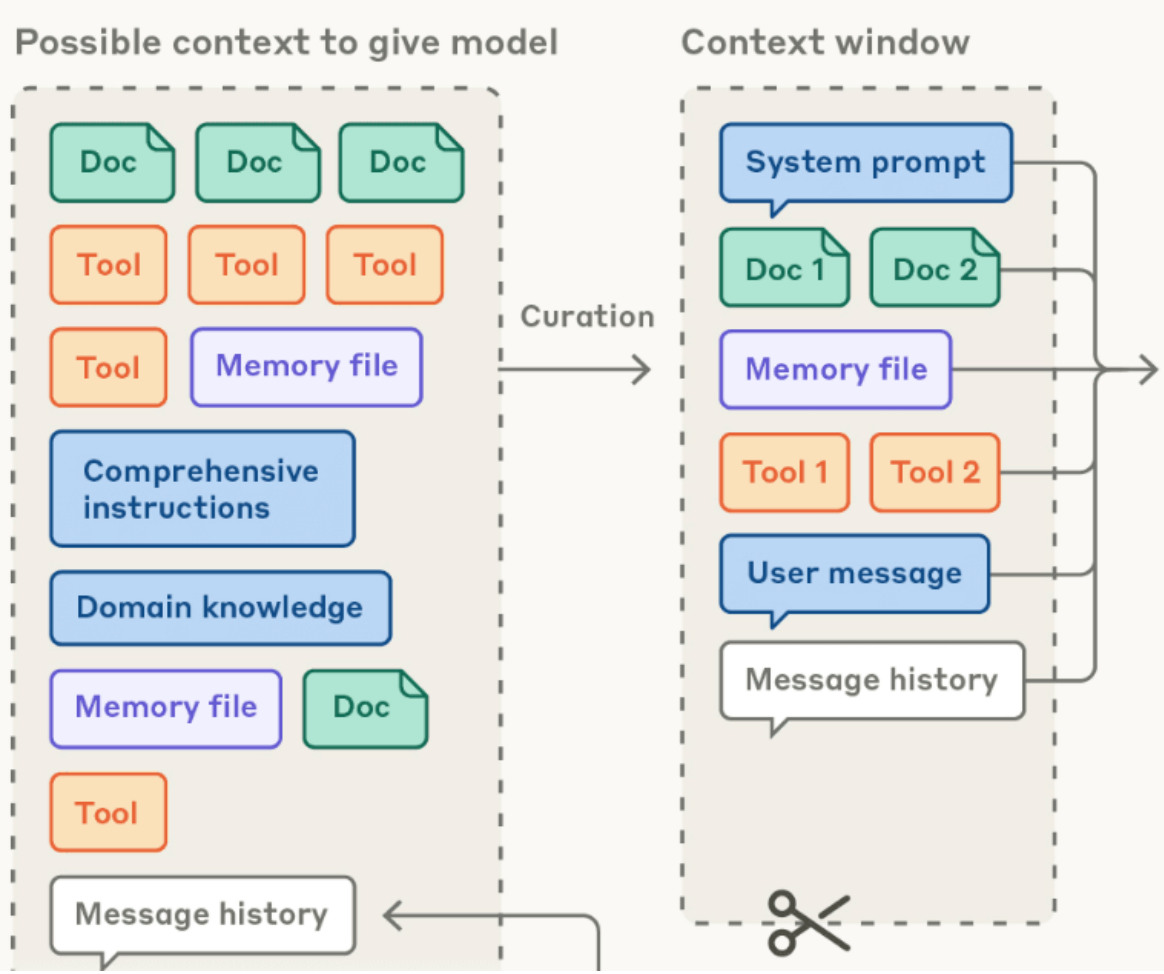


Eliciting Human Preferences With Language Models, Li et al., ICLR 2025

Acquire Context Through Elicitation



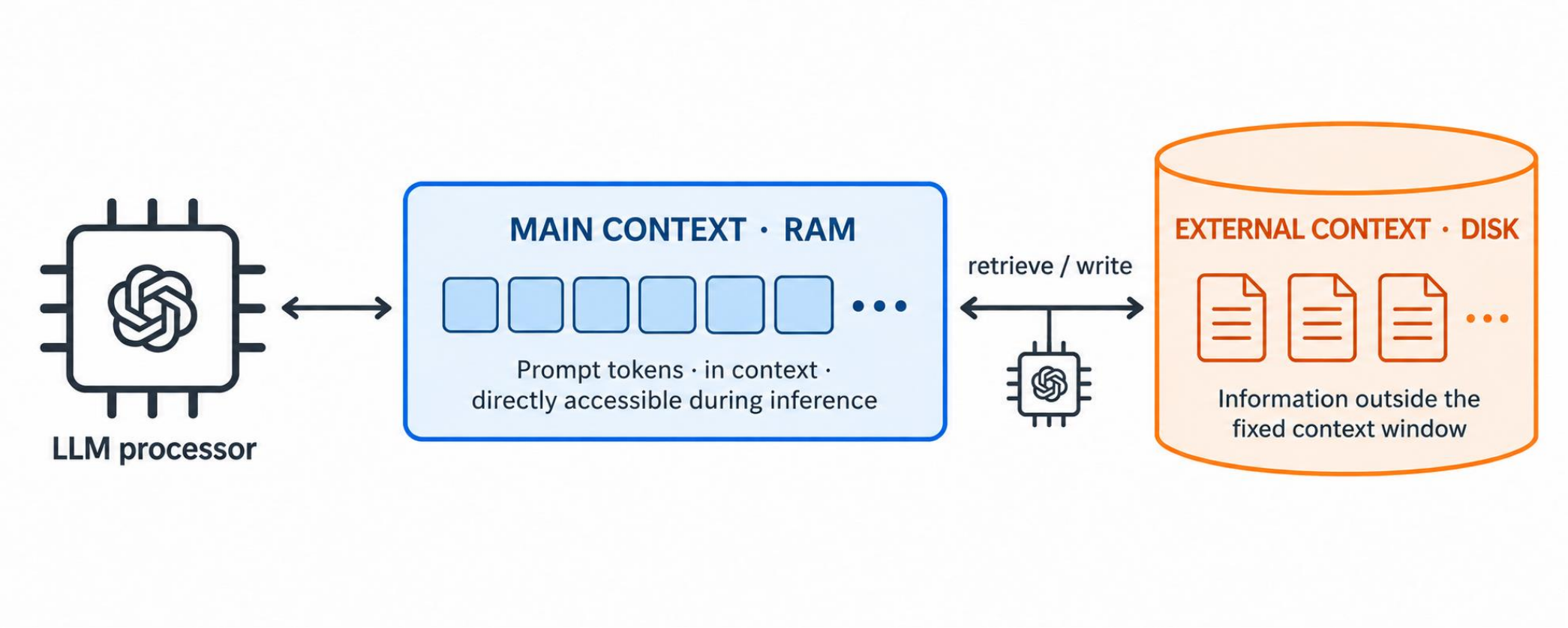
How to Manage the Working Context?



Bitter lesson:

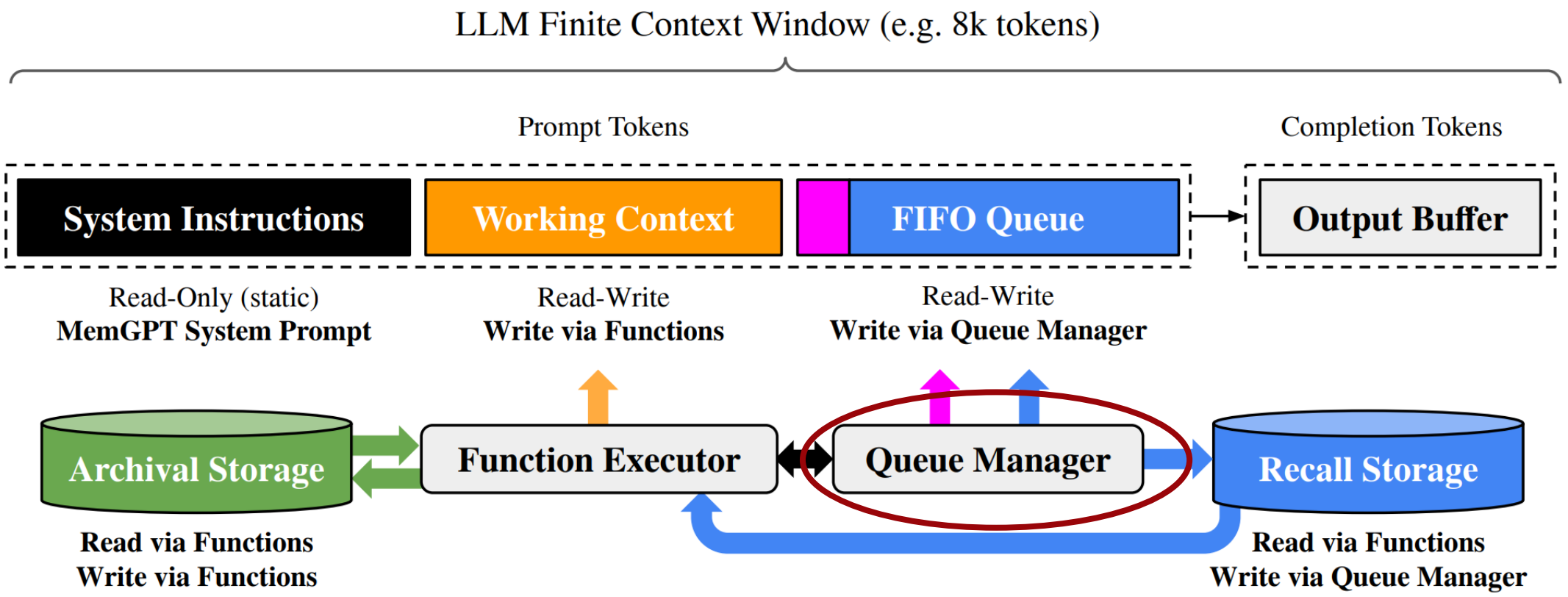
"Artificial intelligence research historically fails when it relies on injecting human domain knowledge and heuristics into algorithms."

Use Model to Manage Its Own Context!



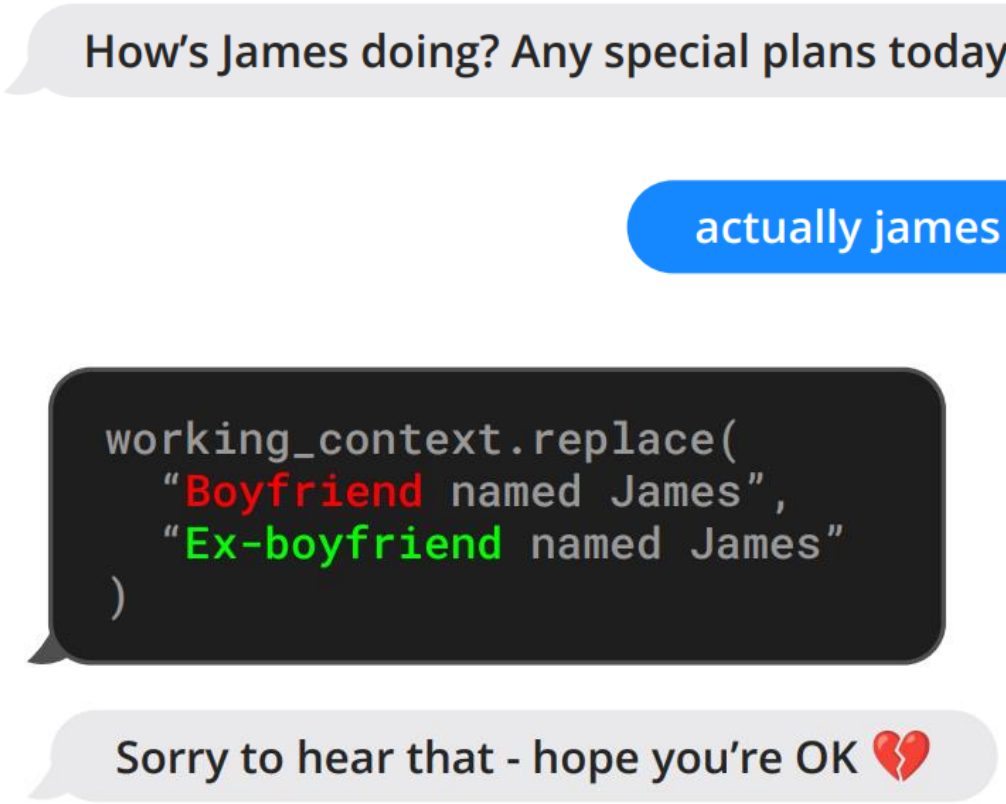
MemGPT: Towards LLMs as Operating Systems, Packer et al., 2023

Use Model to Manage Its Own Context



MemGPT: Towards LLMs as Operating Systems, Packer et al., 2023

Use Model to Manage Its Own Context



When we ask the agent a specific question about a topic discussed in a prior conversation

Model	Accuracy ⇑
GPT-3.5 Turbo	38.7%
+ MemGPT	66.9%
GPT-4	32.1%
+ MemGPT	92.5%
GPT-4 Turbo	35.3%
+ **MemGPT**	**93.4%**

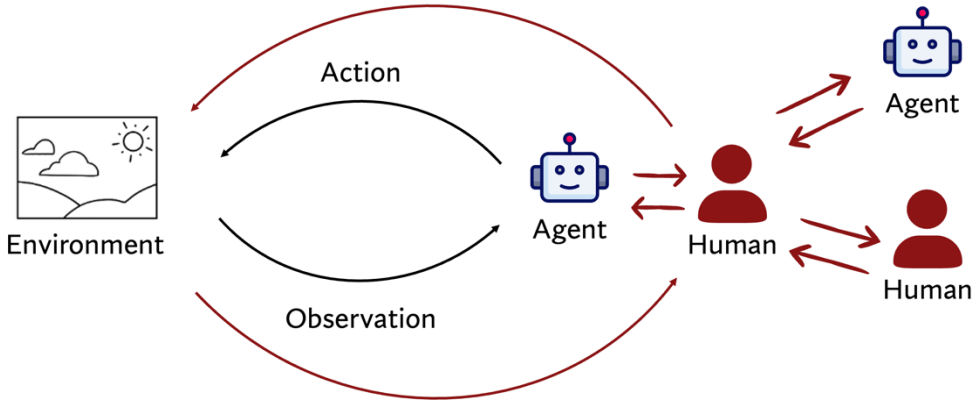
MemGPT: Towards LLMs as Operating Systems, Packer et al., 2023

This lecture

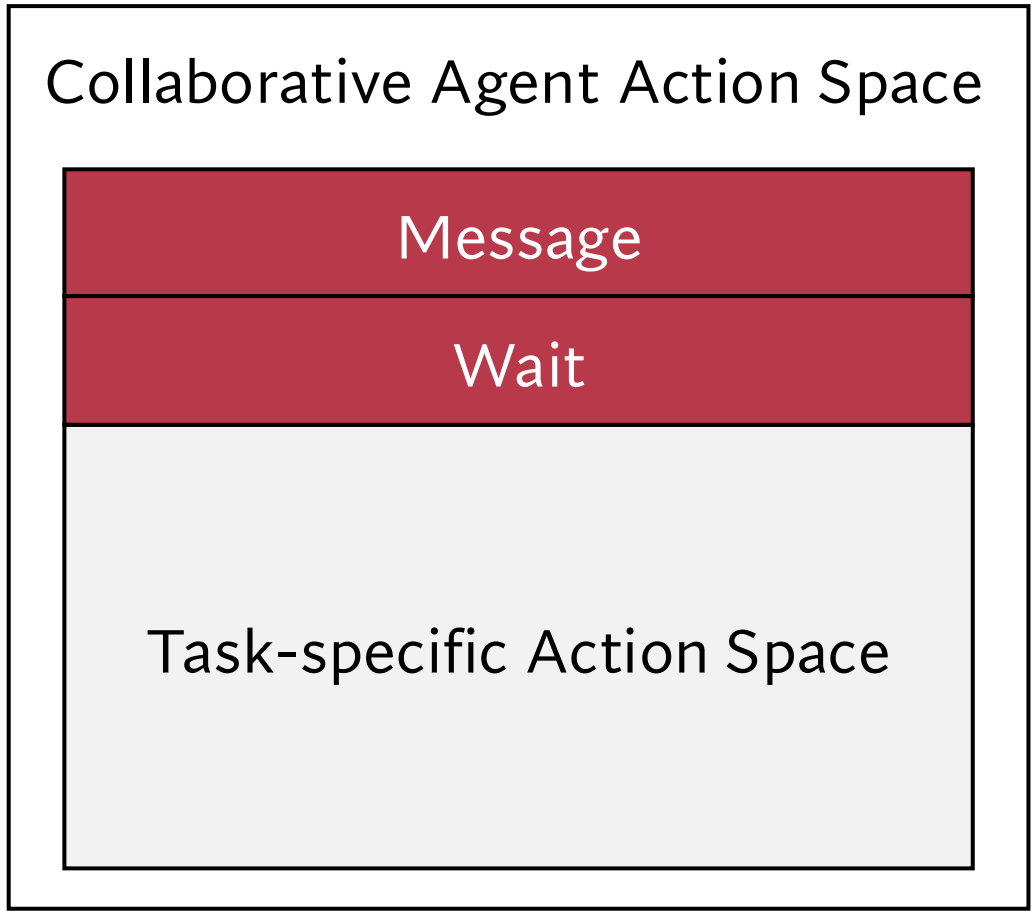
Unpack a few principles behind building full-stack HAI Systems

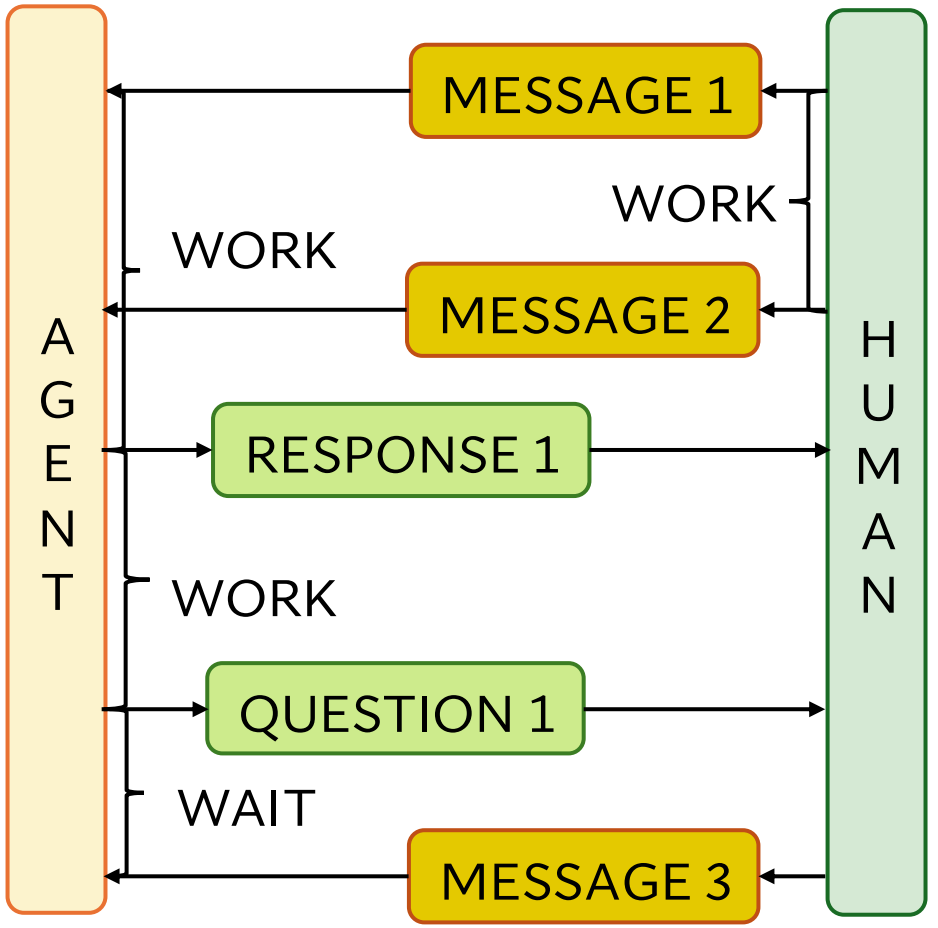
Lesson 1: Intelligence is Token + Context

Lesson 2: Humans and Systems Interact Through a Few Primitives



Recall: Two Foundamental Primitives

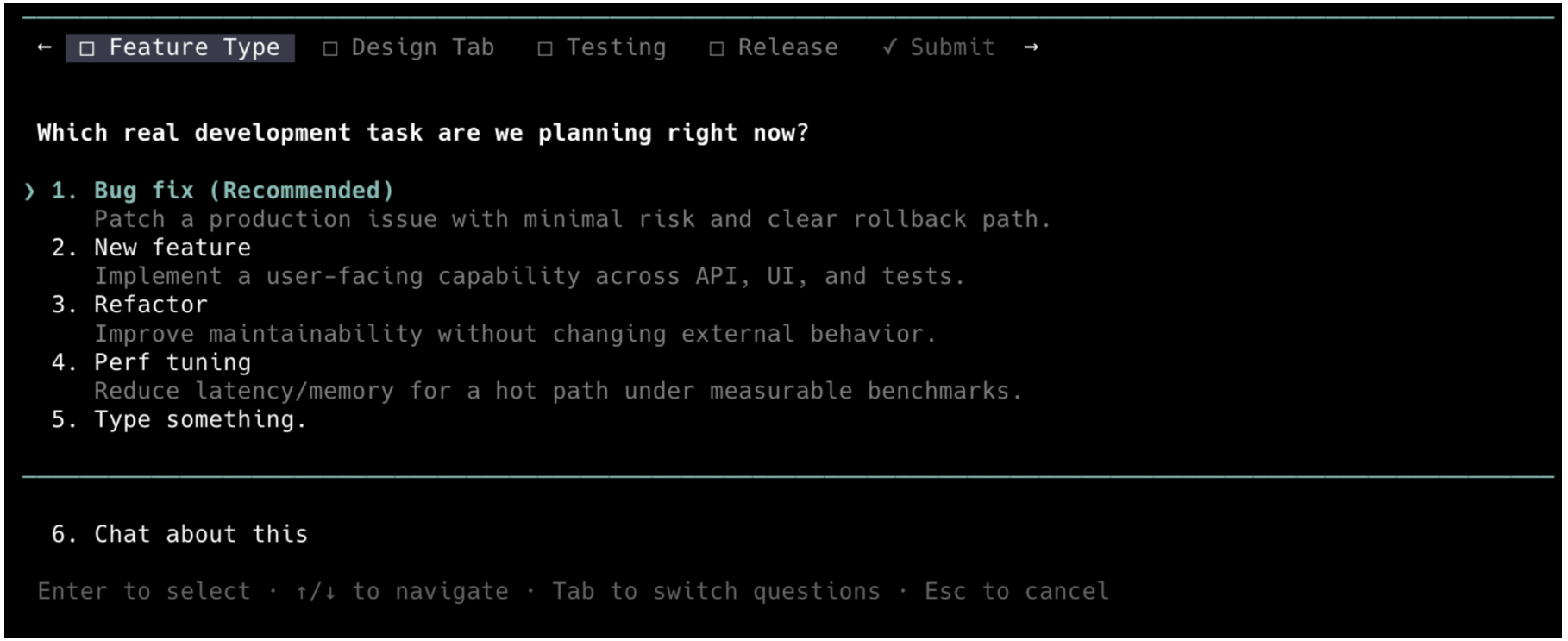




Collaborative Gym: A Framework for Enabling and Evaluating Human-Agent Collaboration, Shao et al., ICLR 2026

Add “Message” Primitive to Make an Agent Collaborative

- In Pi harness, “ask-user-question” is one of the most downloaded plugins. People needs the agent to be collaborative!



DOWNLOADS

272.5K/mo · 27.6K/wk

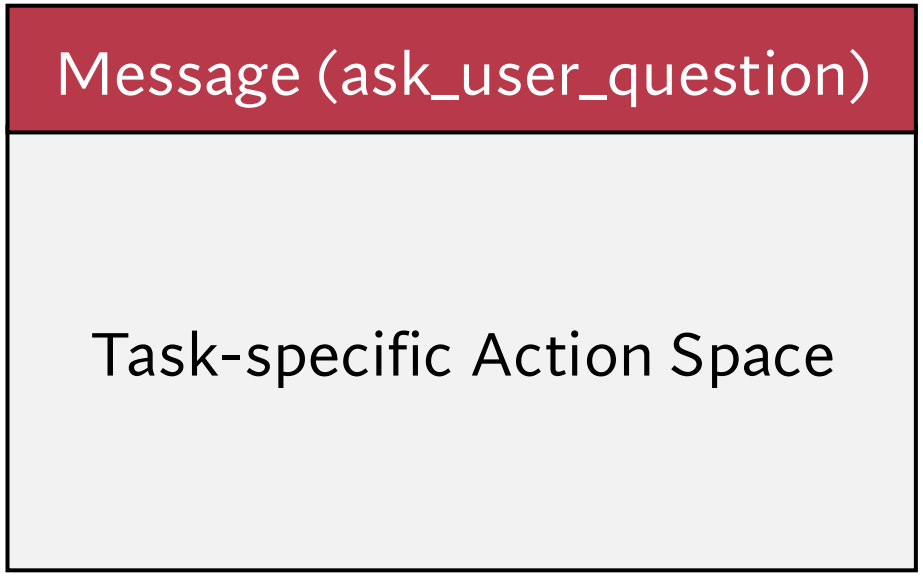
AUTHOR

juicesharp

https://pi.dev/packages/@juicesharp/rpiv-ask-user-question

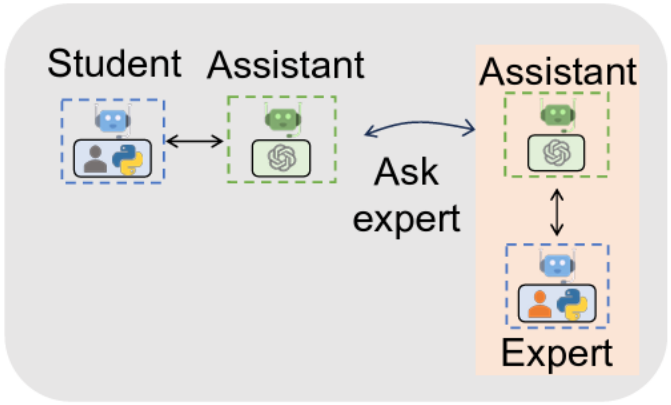
Add “Message” Primitive to Make an Agent Collaborative

```
ask_user_question({
  questions: [
    {
      question: string,      // full question text, ends with "?"
      header: string,        // chip label, max 16 chars
      options: [
        {
          label: string,        // 1–5 words, max 60 chars
          description: string, // choice meaning or trade-off
          preview?: string,     // markdown shown next to options
        },
        // ... 2–4 options total
      ],
      multiSelect?: boolean,  // default false
    },
    // ... 1–4 questions total
  ]
});
```

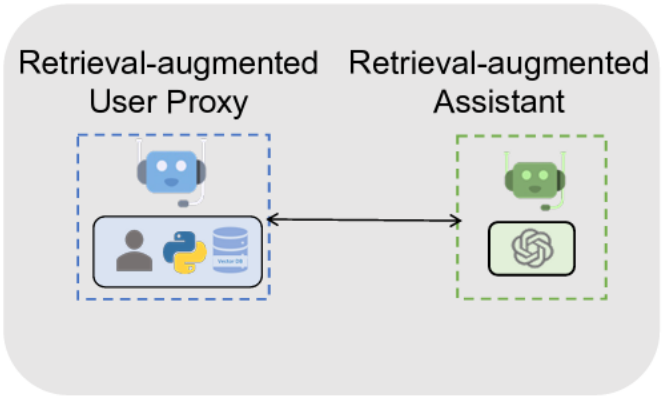


https://pi.dev/packages/@juicesharp/rpiv-ask-user-question

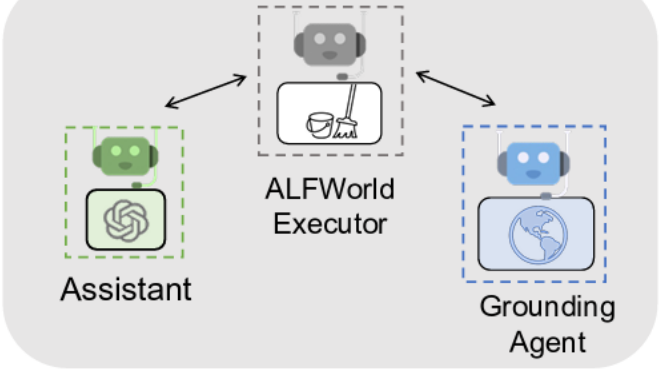
You Can Orchestrate Human-Agent Team With "Message"



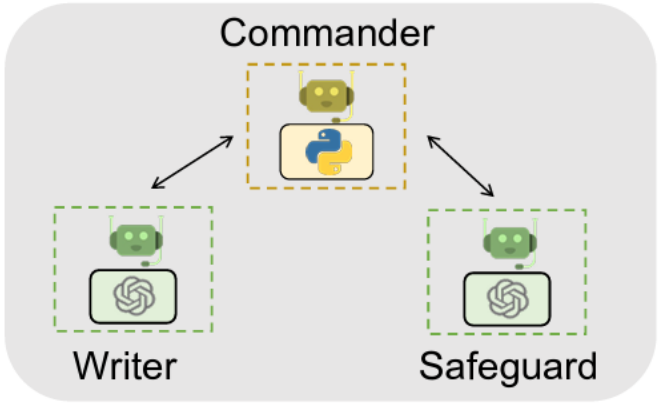
A1. Math Problem Solving



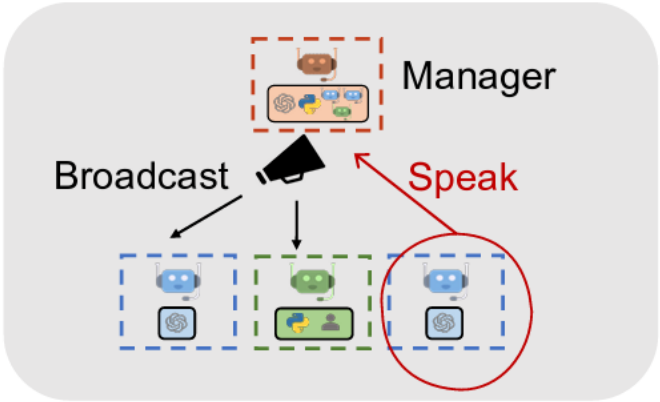
A2. Retrieval-augmented Chat



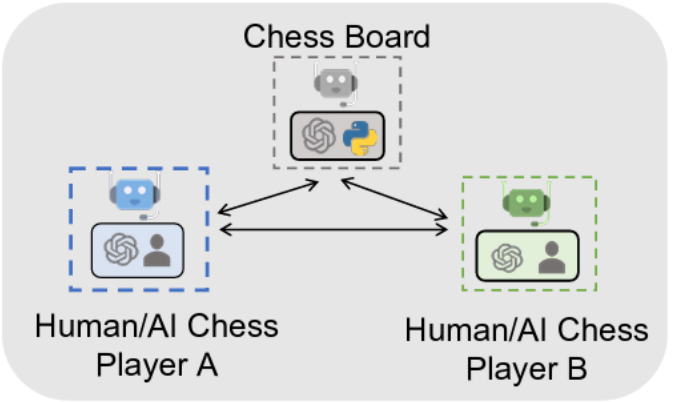
A3. ALF Chat



A4. Multi-agent Coding



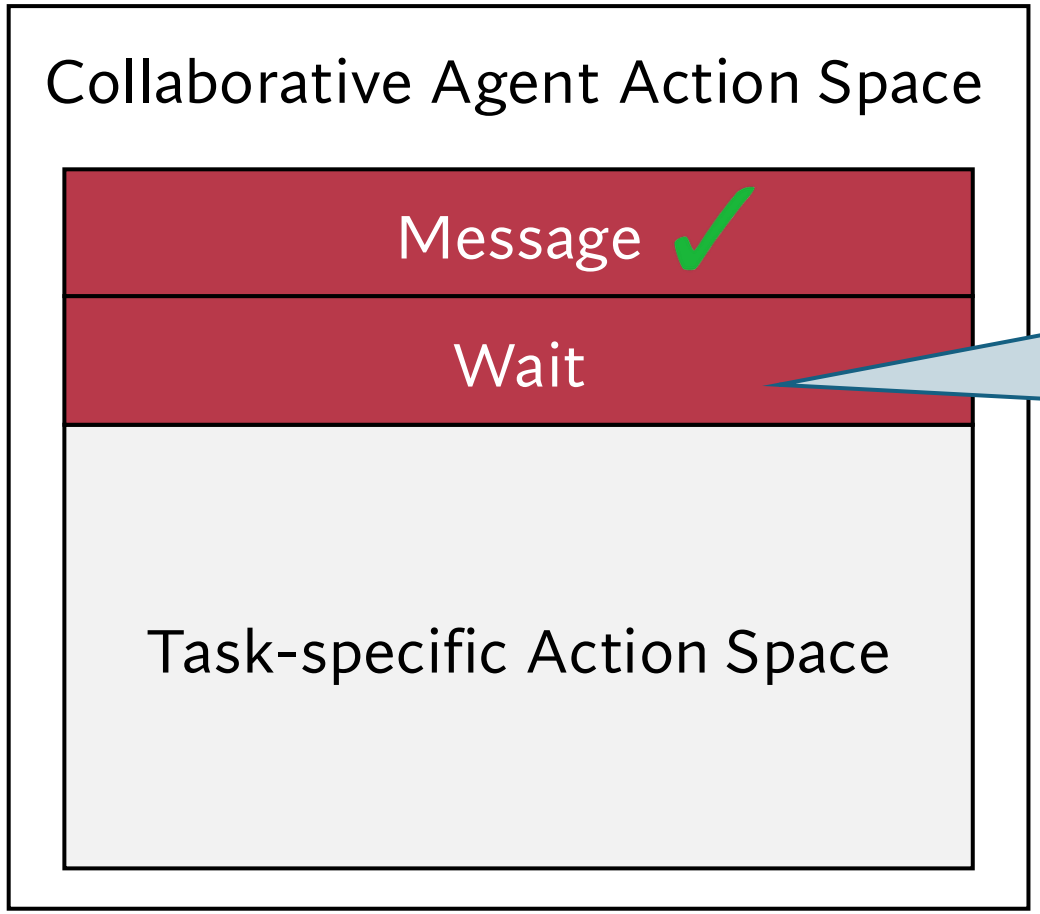
A5. Dynamic Group Chat



A6. Conversational Chess

AutoGen: Enabling Next-Gen LLM Applications via Multi-Agent Conversation, Wu et al., COLM 2024

Two Fundamental Primitives



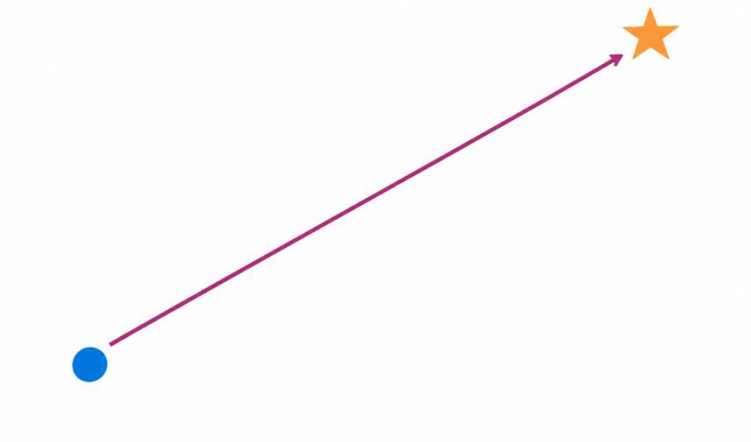
Why does the agent need this?

Alternative view:
Agent shall focus on completing the tasks and human can decide when to use it.

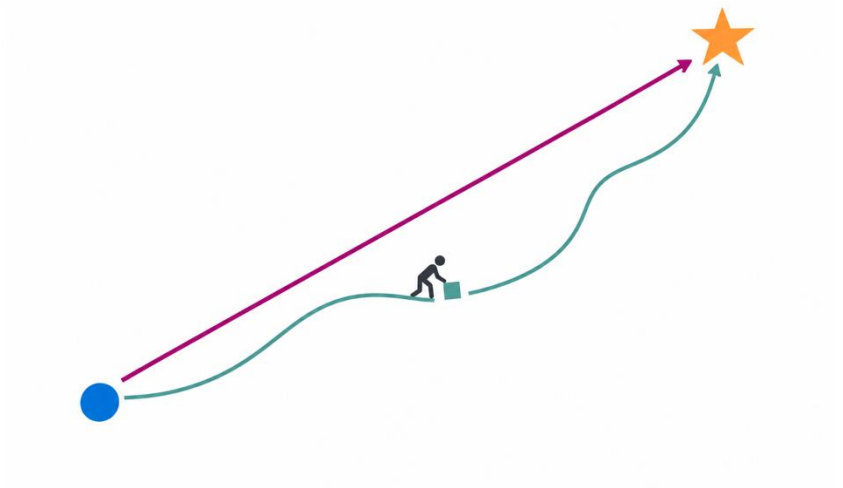


The Problem of Over-indexing on Task Completion

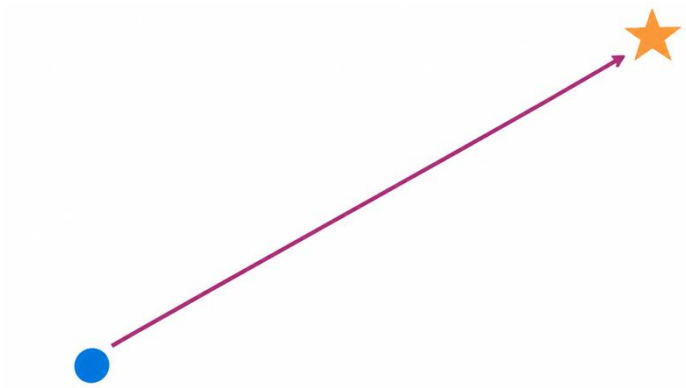
“Is this a good agent trajectory?”

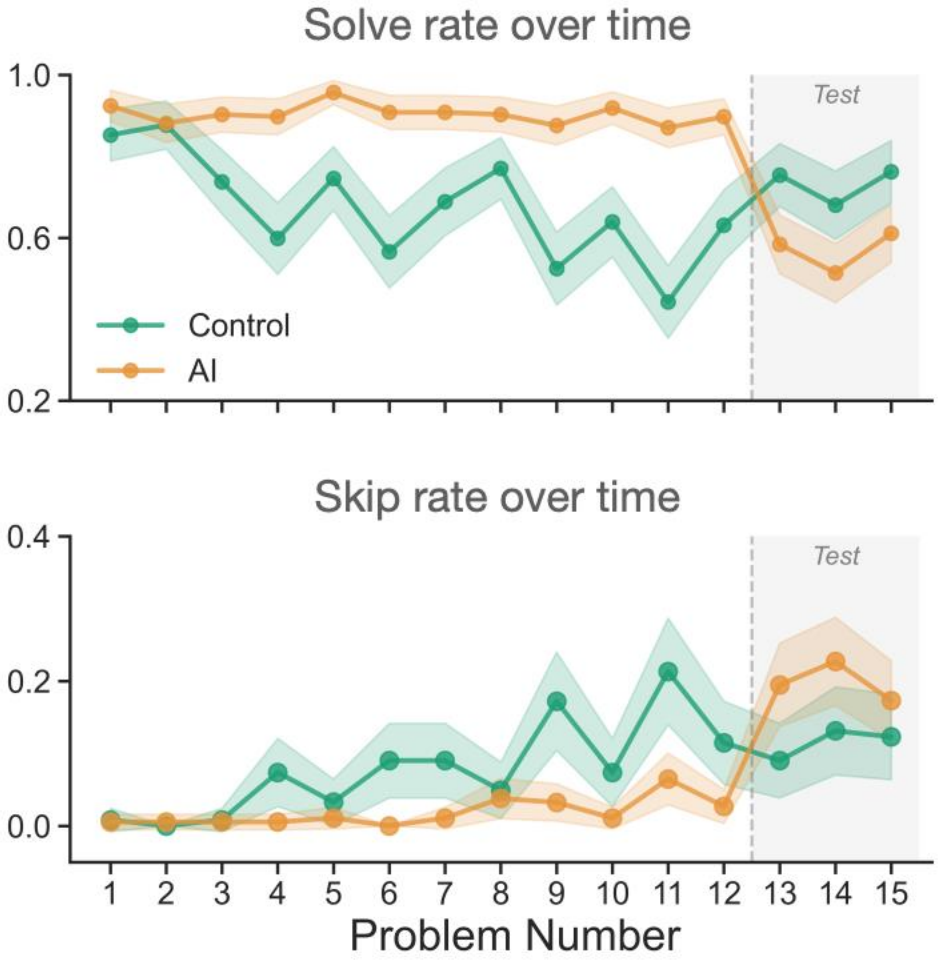


“Which trajectory is better?”



The Problem of Over-indexing on Task Completion: AI Impairs Unassisted Performance & Persistence





AI Assistance Reduces Persistence and Hurts Independent Performance, Liu et al., COLM 2026

General Public Worry About Deskilling



https://www.anthropic.com/features/81k-interviews

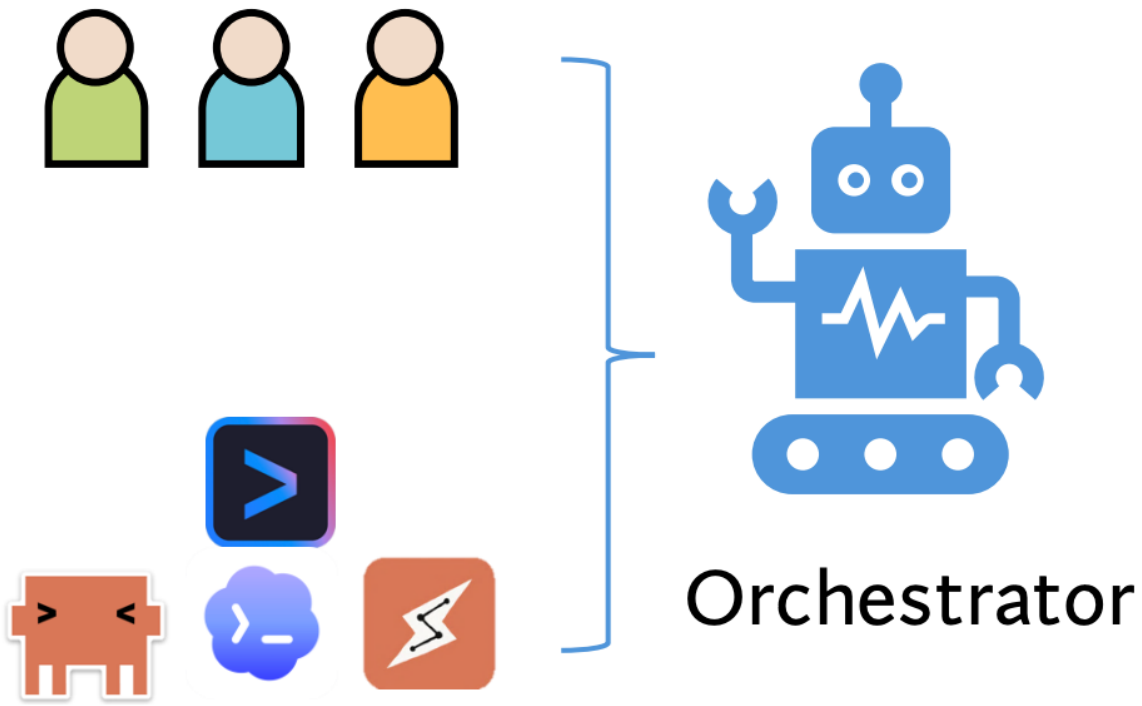
General Public Worry About Deskilling





https://www.anthropic.com/features/81k-interviews

When Should AI Defer to Humans? An Open Challenge



- **Recall from last lecture:** Our CollabOrchestra addresses this at the task level

- How do we account for human expertise and long-term goals?
- How do we keep humans engaged and motivated to contribute?
- Open research frontier...

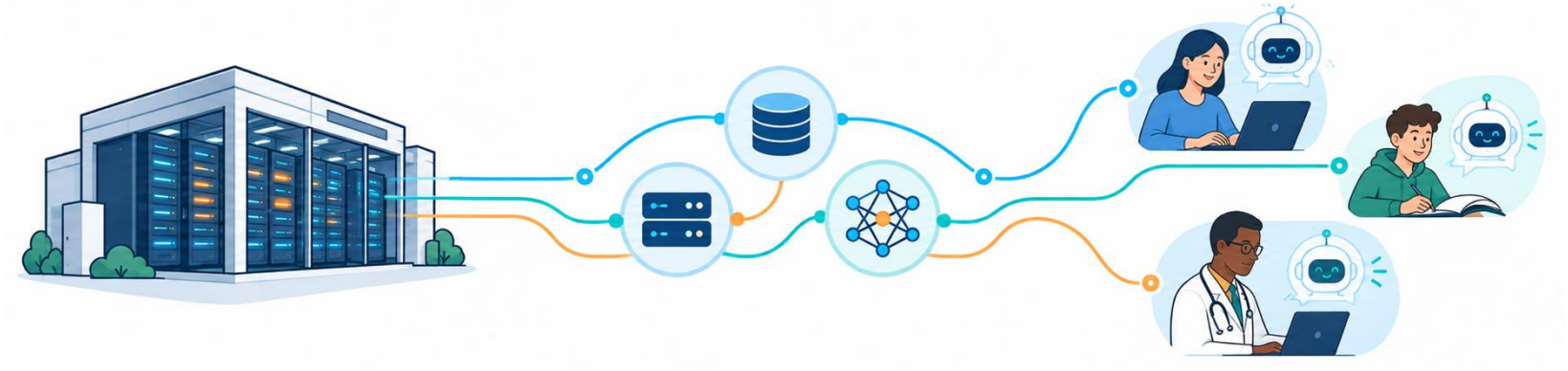
This lecture

Unpack a few principles behind building full-stack HAI Systems

Lesson 1: Intelligence is Token + Context

Lesson 2: Humans and Systems Interact Through a Few Primitives

Lesson 3: Infrastructure Matters



Common Metrics for LLM Infrastructure

- **TTFT (time to first token)** → perceived responsiveness
- **End-to-end latency** → smooth human–AI turn-taking
- **Throughput** → support multiple users and multi-agent systems
- **KV-cache efficiency** → responsive, continuous long interactions
- **GPU memory, utilization, and energy use**

  → determine which LLM sizes you can practically serve

Co-design Infrastructure and Applications

Familiar benefits ✓

Speed up
Less waiting

Reduce cost
More users, lower cost

What co-design makes possible

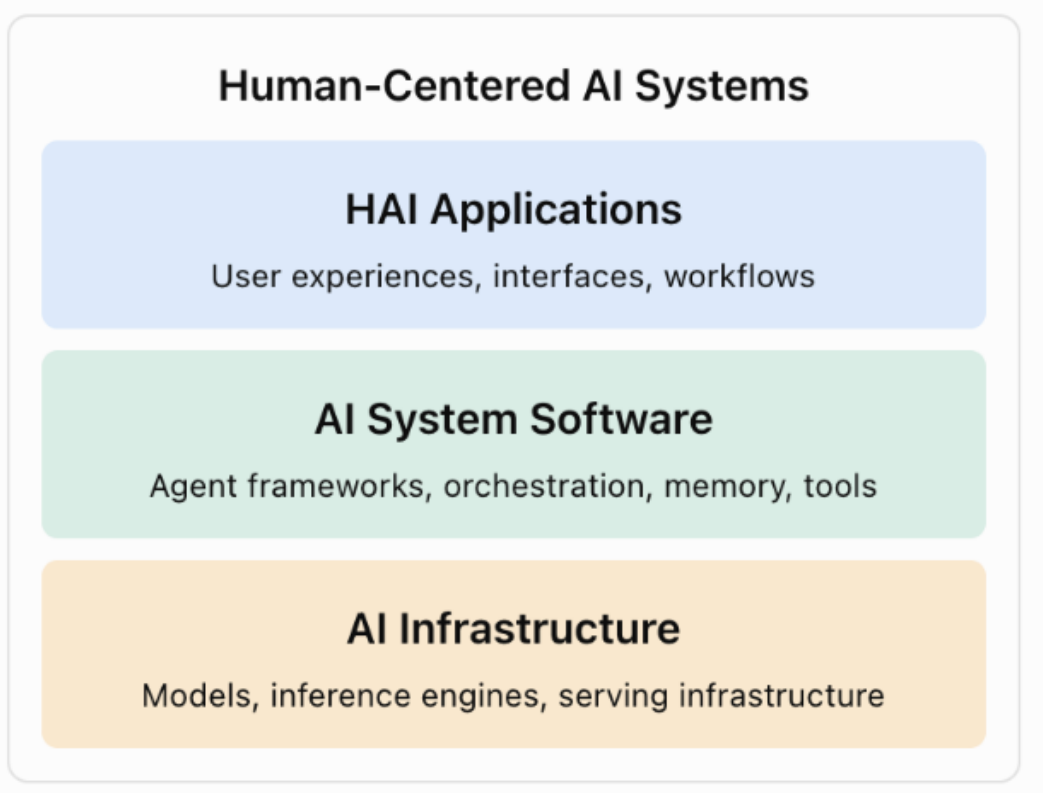
New HAI systems

Real-time, streaming,
ambient interaction

Trust & privacy

On-device inference,
local personalization

Full-Stack Human-Centered AI Systems



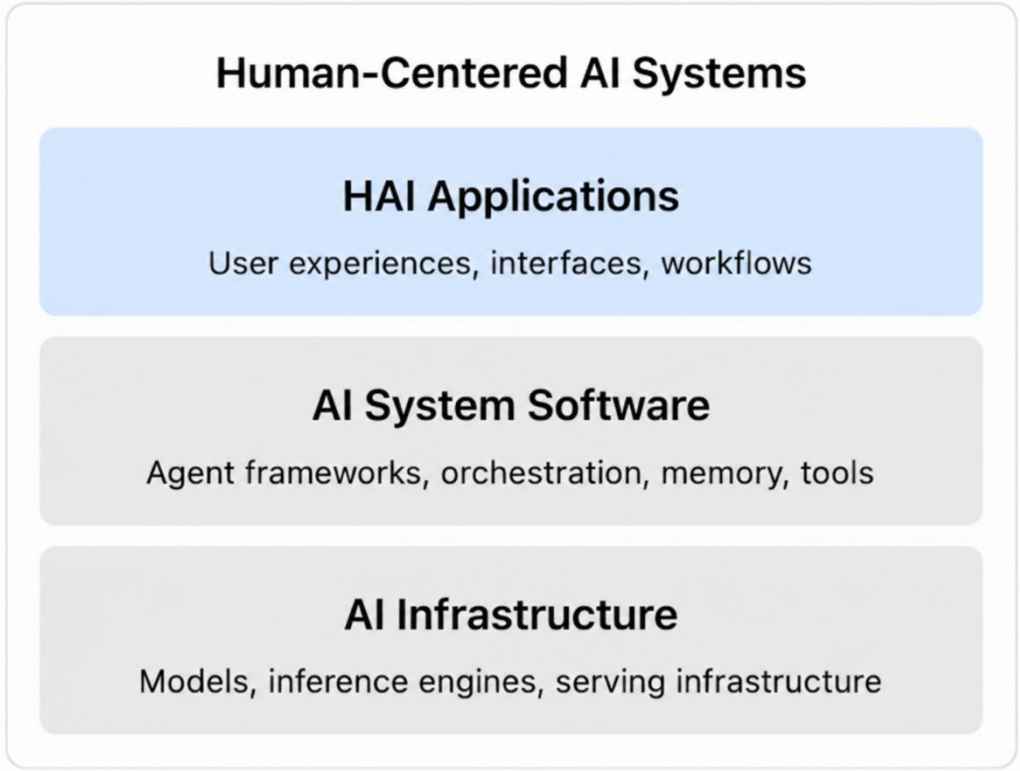
Intelligence is Token + Context

Humans and Systems Interact Through a Few Primitives

Infrastructure Matters

Let’s piece things together!

Case Study 1: Real-time Voice Agent

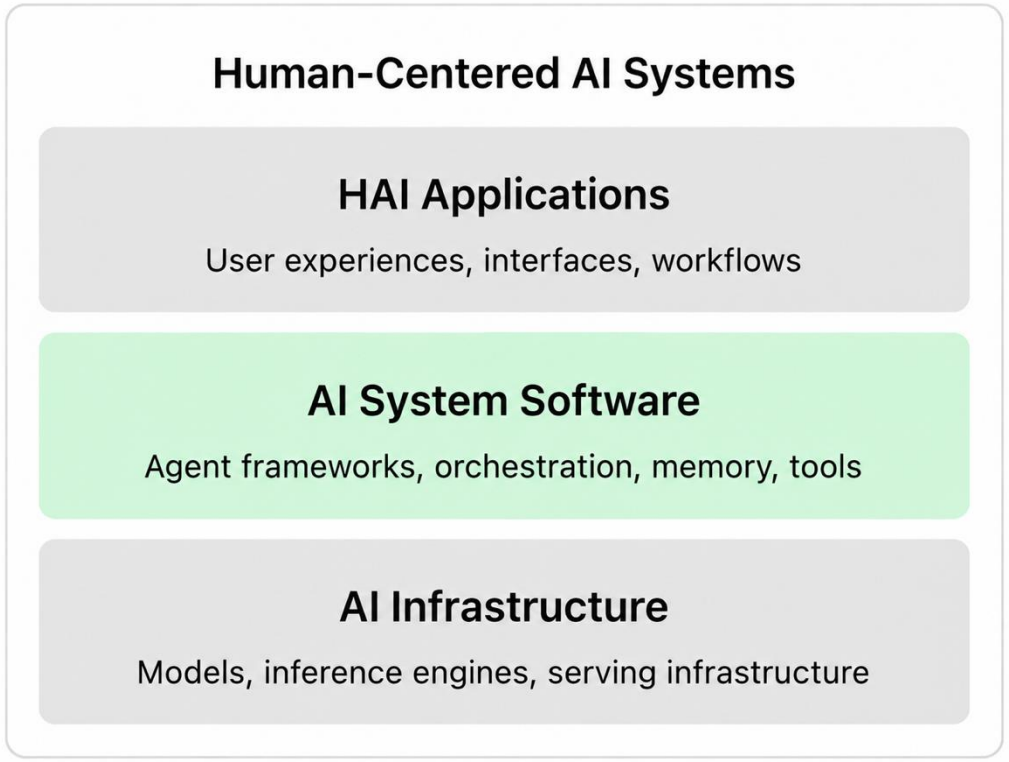


Voice can be **faster for rough capture** and can be essential for **accessibility**.

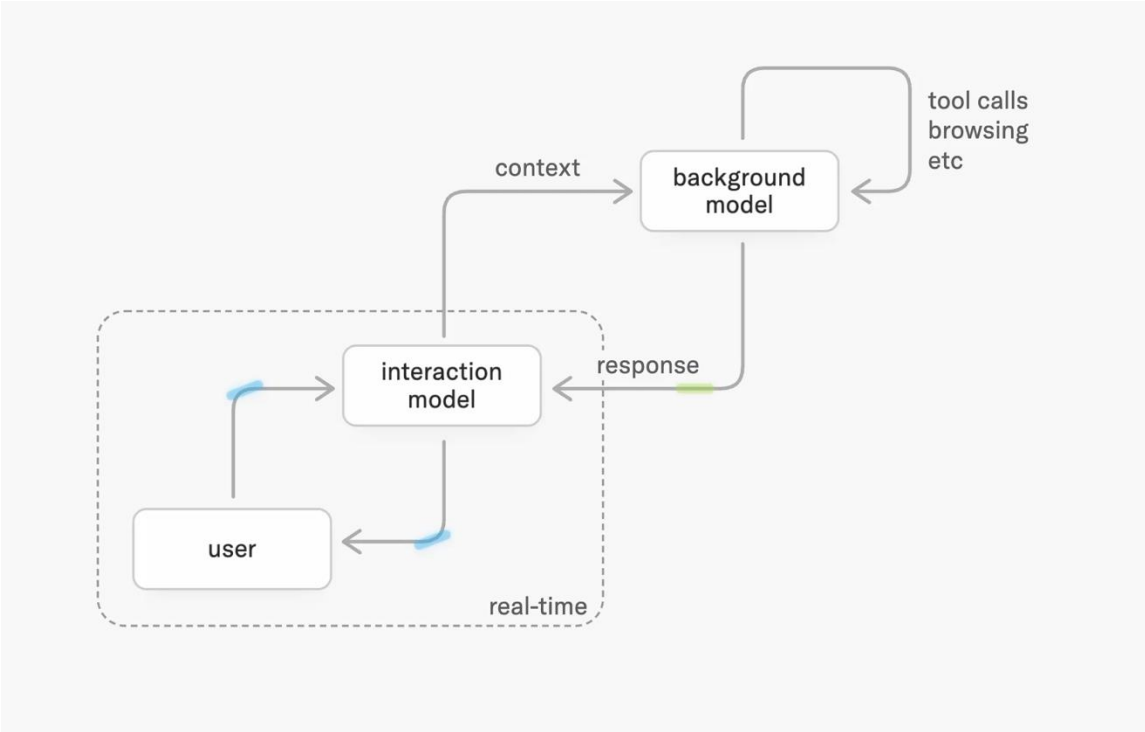


https://keytopia.org/blog/typing-vs-voice-input-speed

Case Study 1: Real-time Voice Agent

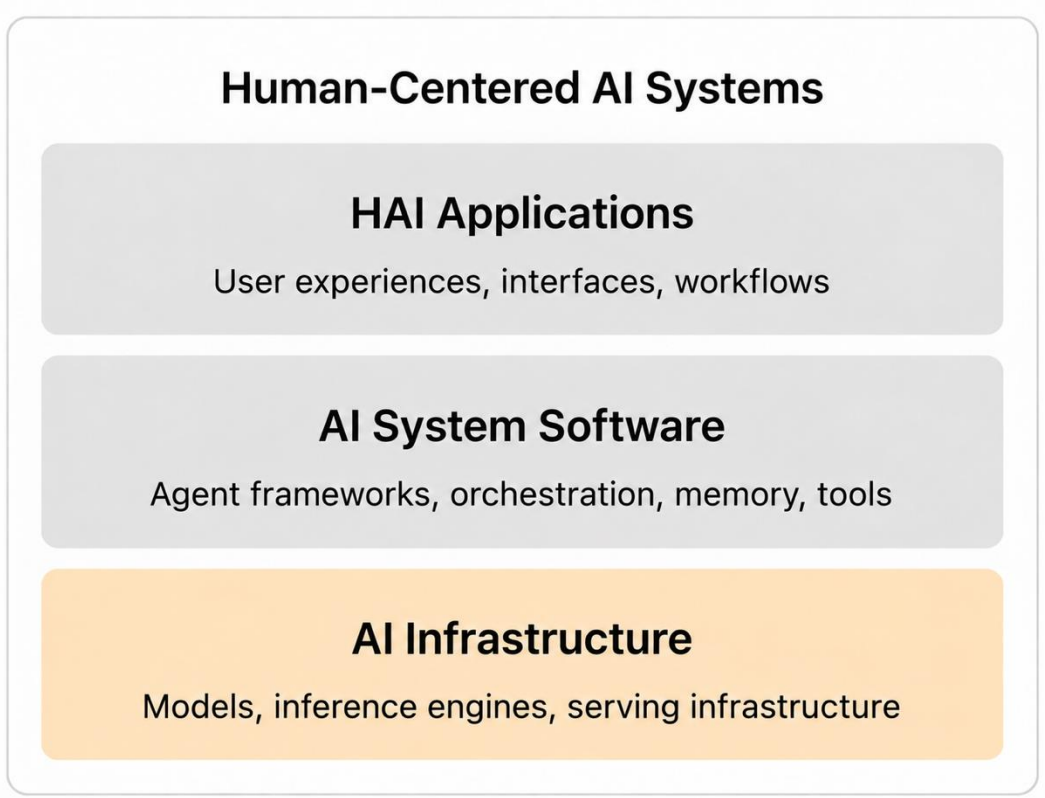


The system is is in constant **two-way exchange** with the user—perceiving and responding at the same time.

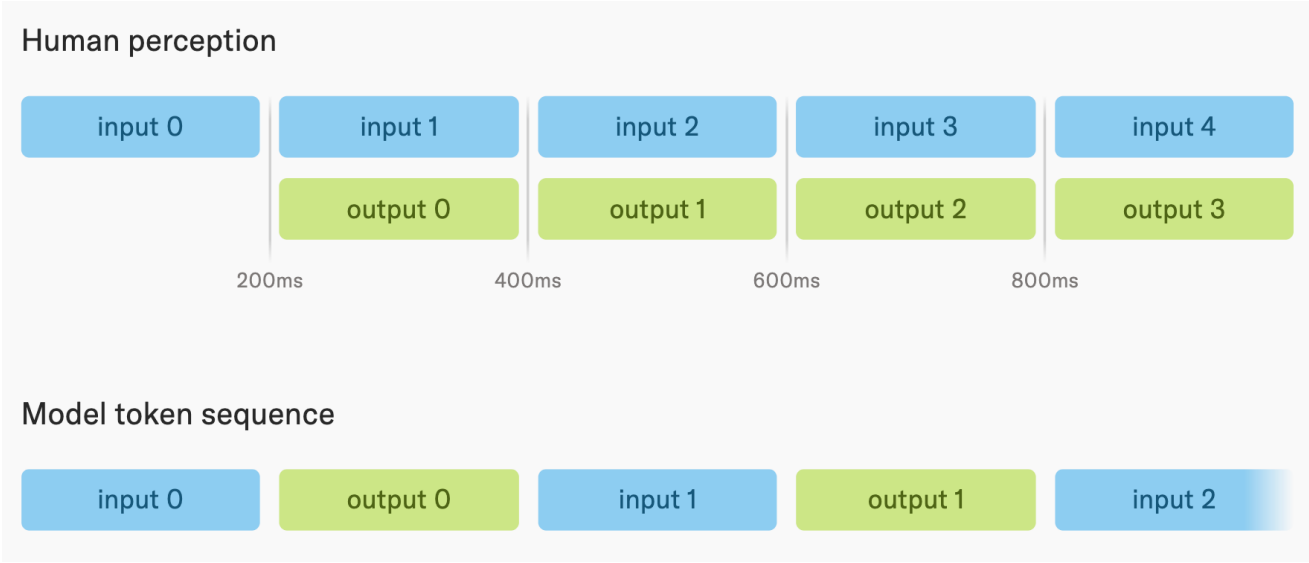


https://thinkingmachines.ai/blog/interaction-models/

Case Study 1: Real-time Voice Agent



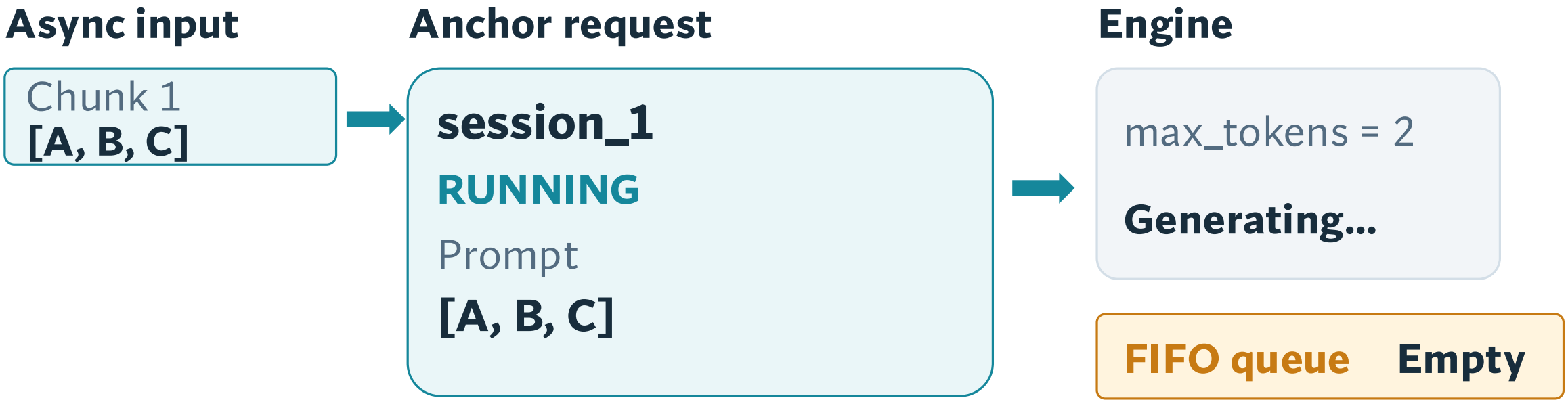
Rather than consuming a complete user-turn and generating a complete response, **both input and output tokens are treated as streams**.



https://thinkingmachines.ai/blog/interaction-models/

Support Streaming Input in vllm

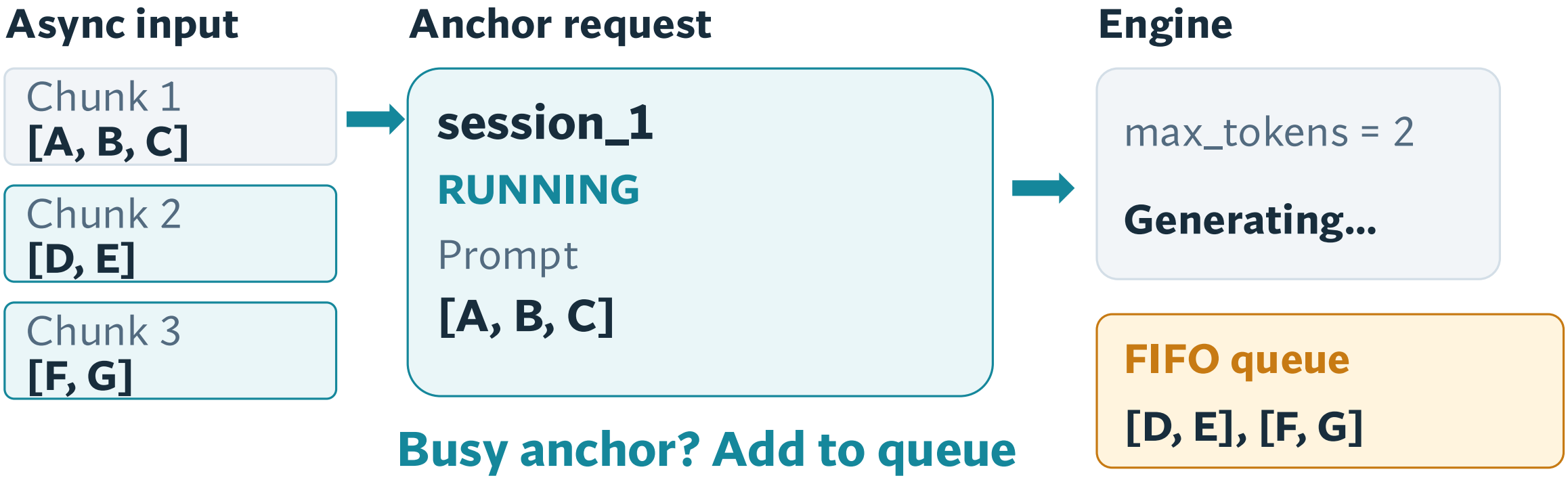
Sticky session: keep one request alive as new audio chunks arrive



https://vllm.ai/blog/2026-01-31-streaming-realtime

Support Streaming Input in vllm

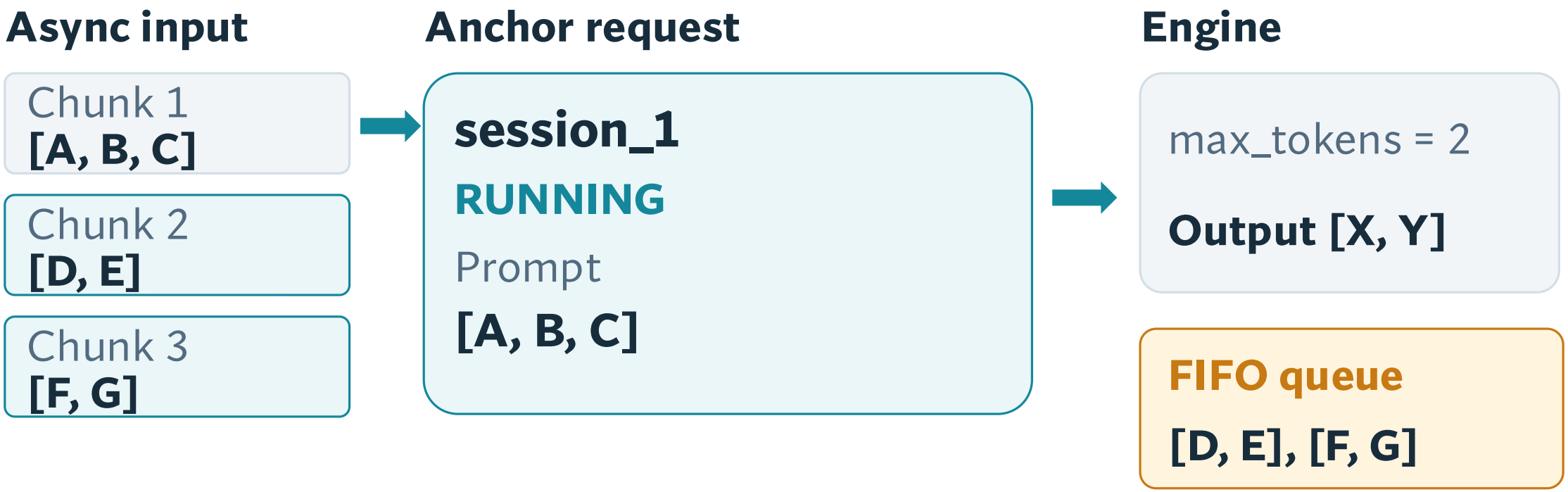
Sticky session: keep one request alive as new audio chunks arrive



https://vllm.ai/blog/2026-01-31-streaming-realtime

Support Streaming Input in vllm

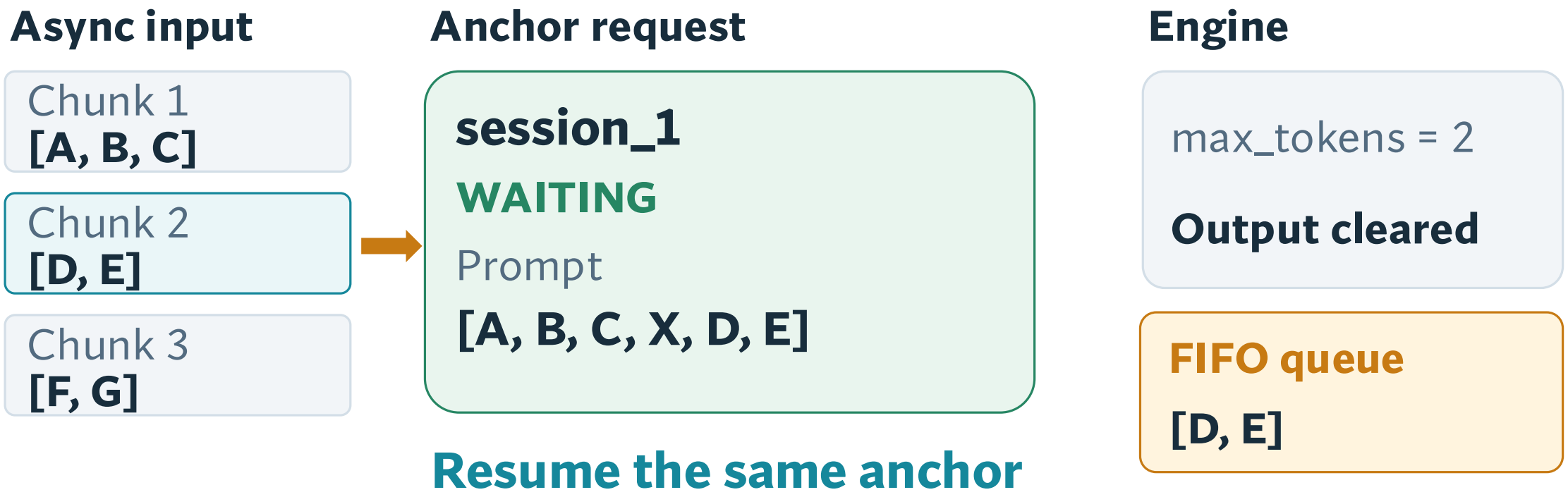
Sticky session: keep one request alive as new audio chunks arrive



https://vllm.ai/blog/2026-01-31-streaming-realtime

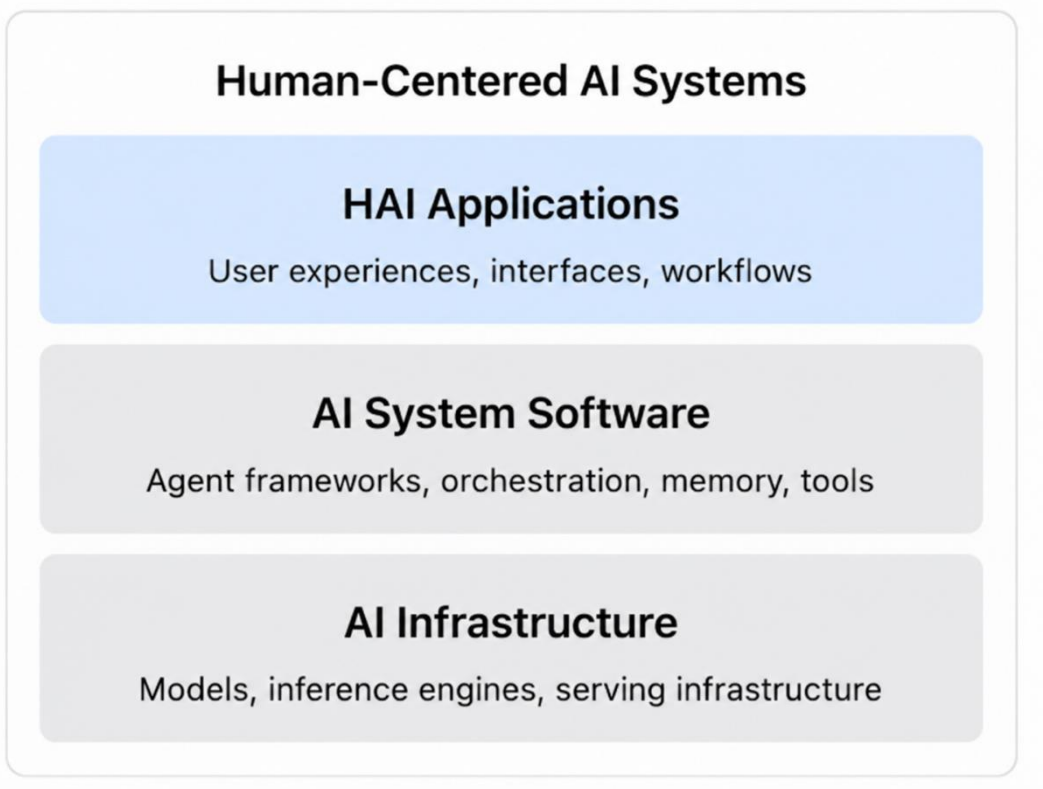
Support Streaming Input in vllm

Sticky session: keep one request alive as new audio chunks arrive



https://vllm.ai/blog/2026-01-31-streaming-realtime

Case Study 2: Personalized Proactive Agent

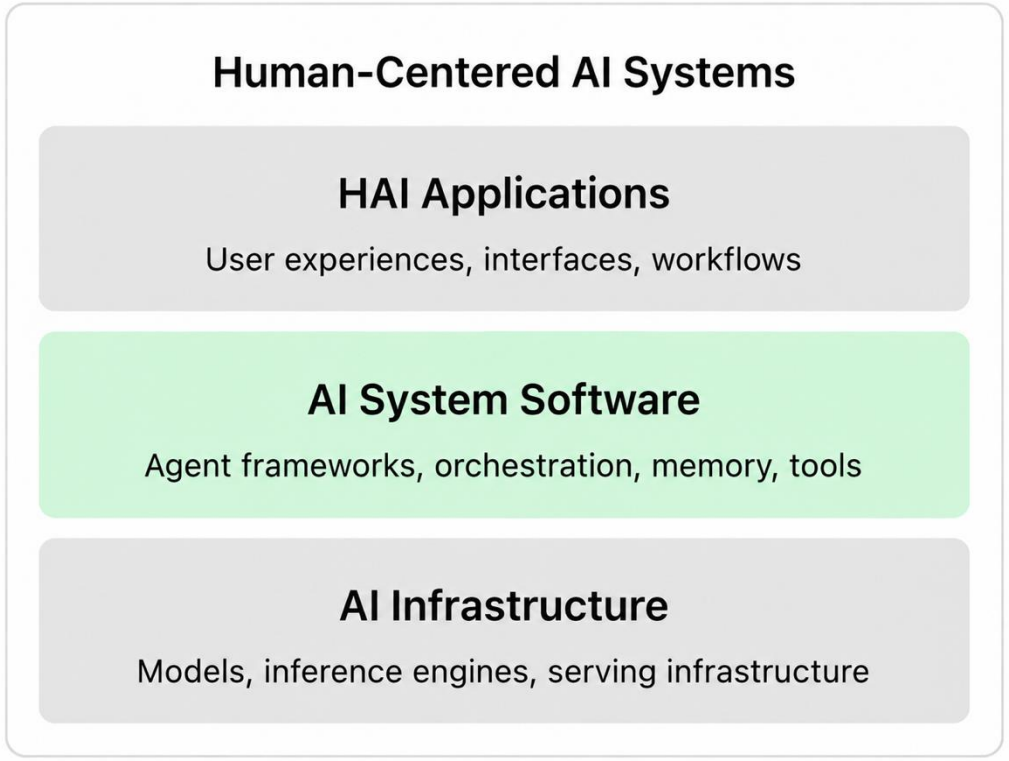




Proactive support across context

https://github.com/collaborative-agents/coco

Case Study 2: Personalized Proactive Agent



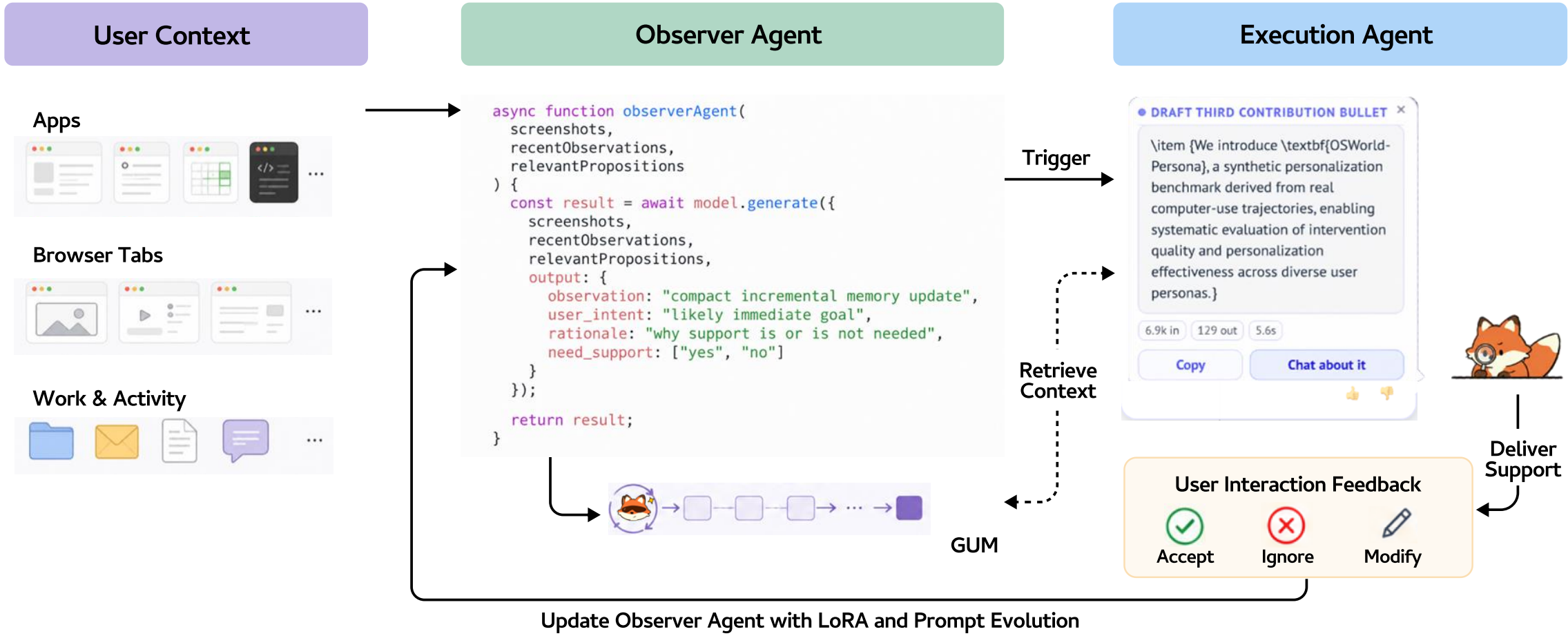
Proactive support is not new (Horvitz, 1999; Shaikh et al., 2025)

Core challenge:

- Give proactive support at the right time.
- This involves highly contextualized preferences.

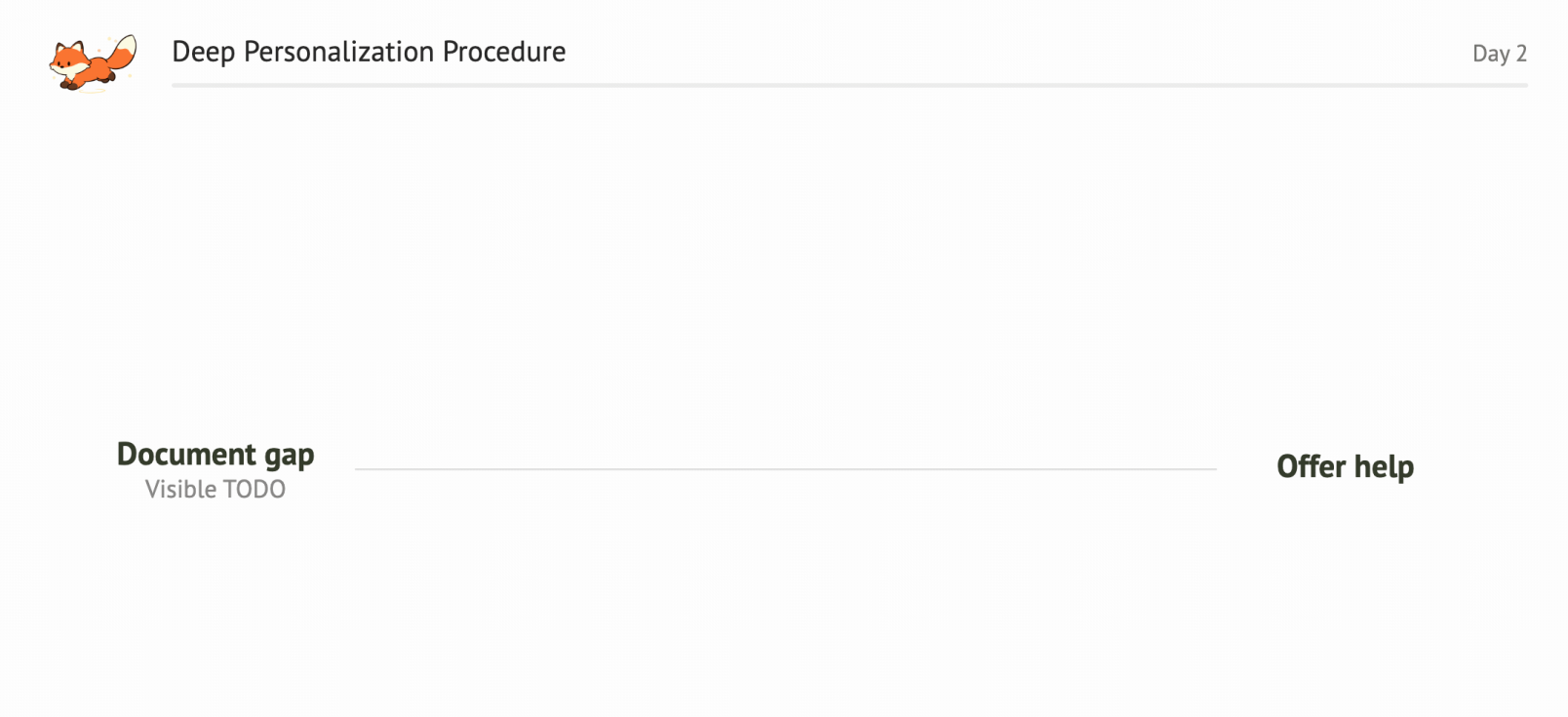
https://github.com/collaborative-agents/coco

Deep Personalization



https://github.com/collaborative-agents/coco

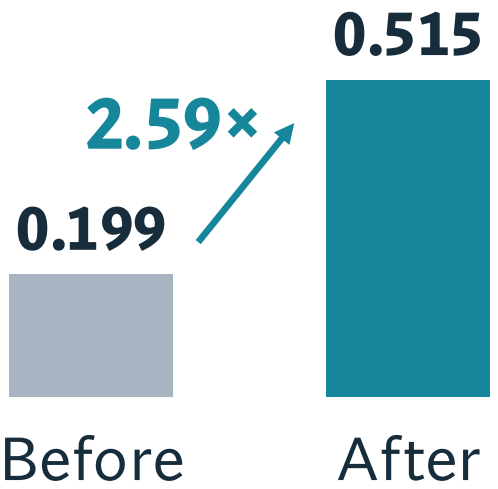
Deep Personalization Improves Proactive Support Timing Appropriateness



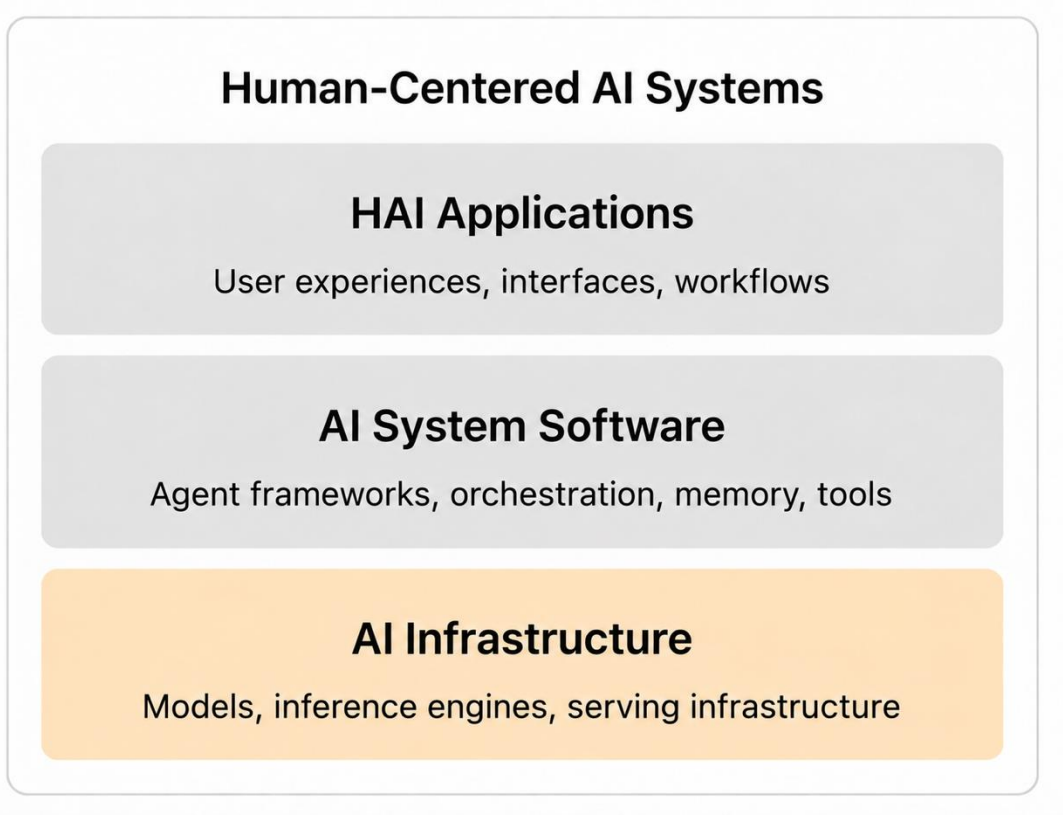
1-week user study

N = 10

Macro-averaged F1 proxy



Case Study 2: Personalized Proactive Agent



Personalized AI has a built-in tension.

- The more context an agent sees and retains, the better it personalizes;
- But the more sensitive information it exposes.

How to solve this?

https://github.com/collaborative-agents/coco

Heterogeneous Form of Compute

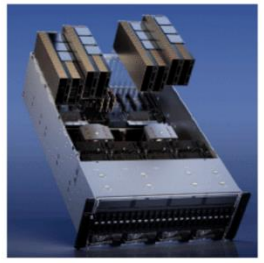
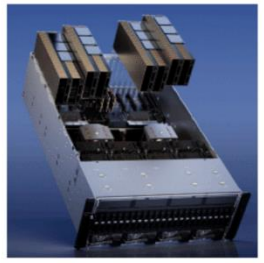
	Cloud AI	Mobile AI		Tiny AI
				
	Nvidia H100[1]	Apple M2 Ultra[2]	Qualcomm S8Gen2[3]	STM32F746NG[4]
Memory	80GB	64-192 GB	8-24 GB	320kB
Storage	~TB/PB	~GB/TB	~GB	1MB
Computing power	1,979 TOPS	31.6 TOPS	36 TOPS	462 MOPS

Table credits to “TinyML and Efficient AI Computing” course from MIT (https://efficientml.ai/)

Heterogeneous Form of Compute

	Cloud AI		Mobile AI		Tiny AI
					
	Nvidia H100[1]		Apple M2 Ultra[2]	Qualcomm S8Gen2[3]	STM32F746NG[4]
Memory	80GB		64-192 GB	8-24 GB	320kB
Storage	~TB/PB		~GB/TB		1MB
Computing power	1,979 TOPS		31.6 TOPS		462 M(



Llama 3 70B (quantized)

Qwen 7B

Stable Diffusion

Classification model

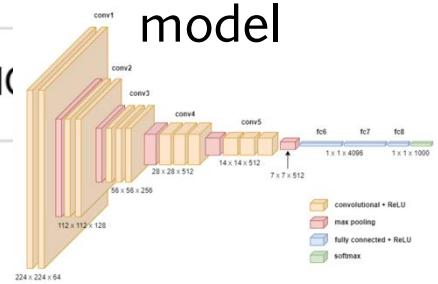
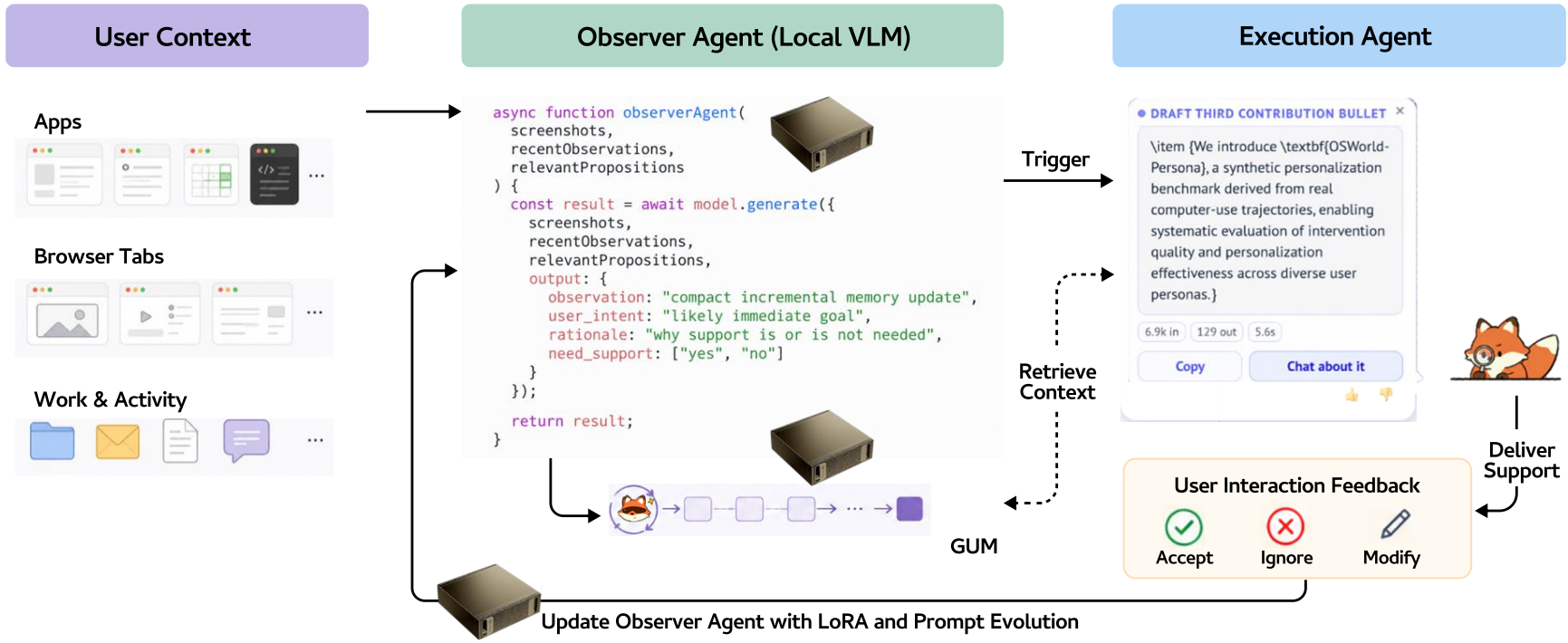


Table credits to “TinyML and Efficient AI Computing” course from MIT (https://efficientml.ai/)

Leverage Local Compute For Deep Personalization

Qwen3.5-35B-A3B-FP8 on DGX Spark: TTFT 2.8s, latency 7.5s
Meets Coco's real-time interaction requirement ✓

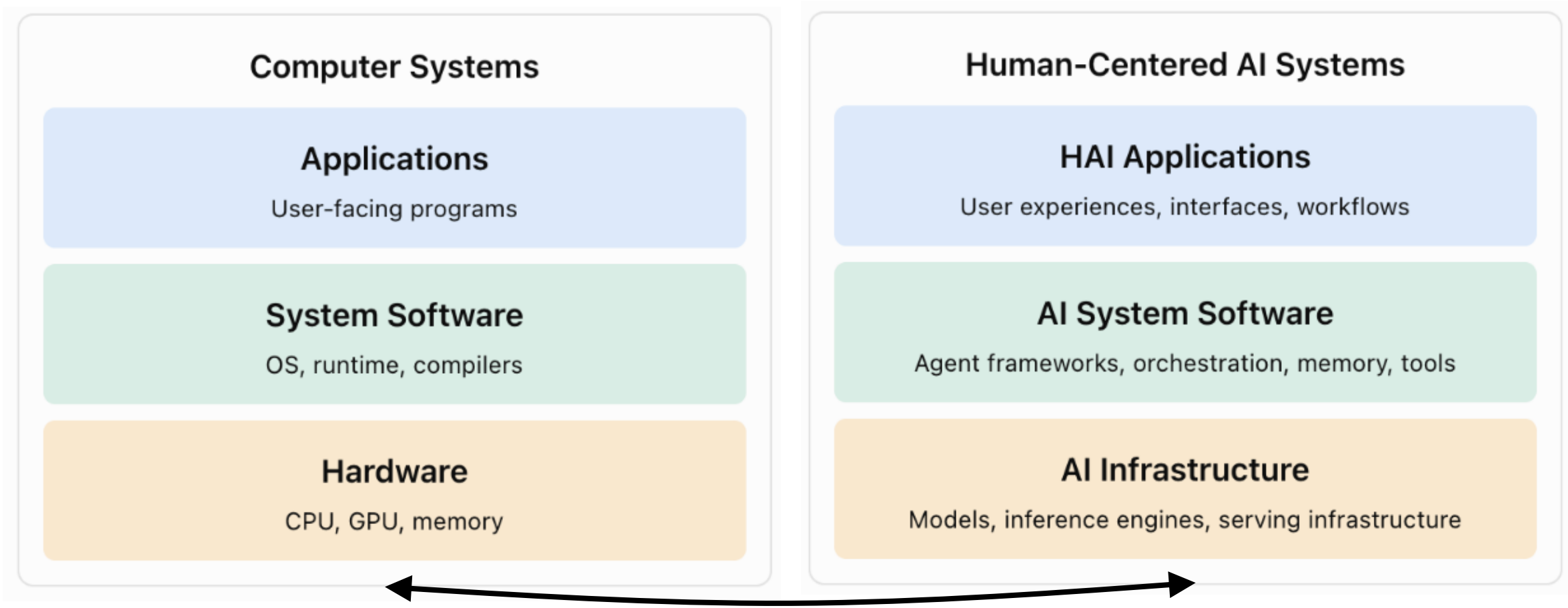






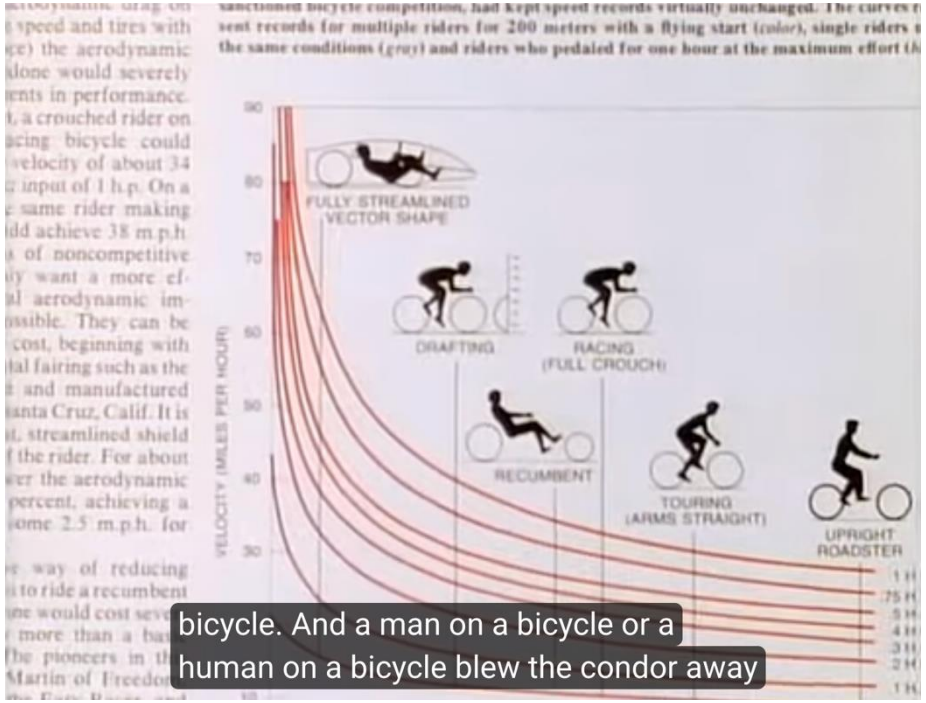
https://github.com/collaborative-agents/coco

Full-Stack Human-Centered AI Systems



Share Some Design Principles; Both Need System Thinking









Computers and HAI Systems Also Share Similar Grand Mission

Summary

- **Intelligence is Token + Context**
  - Use model to manage its own context
  - Combine observation and elicitation to acquire context
- **Human and Systems Interact Through a Few Primitives**
  - Message, Wait
- **Infrastructure Matters**

Next Class: New Frontiers in Generative Interaction