What are AI agents?

Foundations, Architectures, and Implications

Wed, Jan 14, 2025

Reading: Agents Survey based on Wang et al. (2024) & Jungwei et al. (2025)

University of Colorado Boulder

Danny Dig

What do you most want to get out of today's class?

Course Syllabus

Participate in Class Discussions [10%] – individual

Paper Critiques [20%] – individual

Paper presentations [20%] – individual

Project [50%] – team

Family


adidas

Occupation:
Faculty in Software Engineering

Change is the heart of software development

Programming is program transformation

Q1: Analyze what software changes occur in practice?

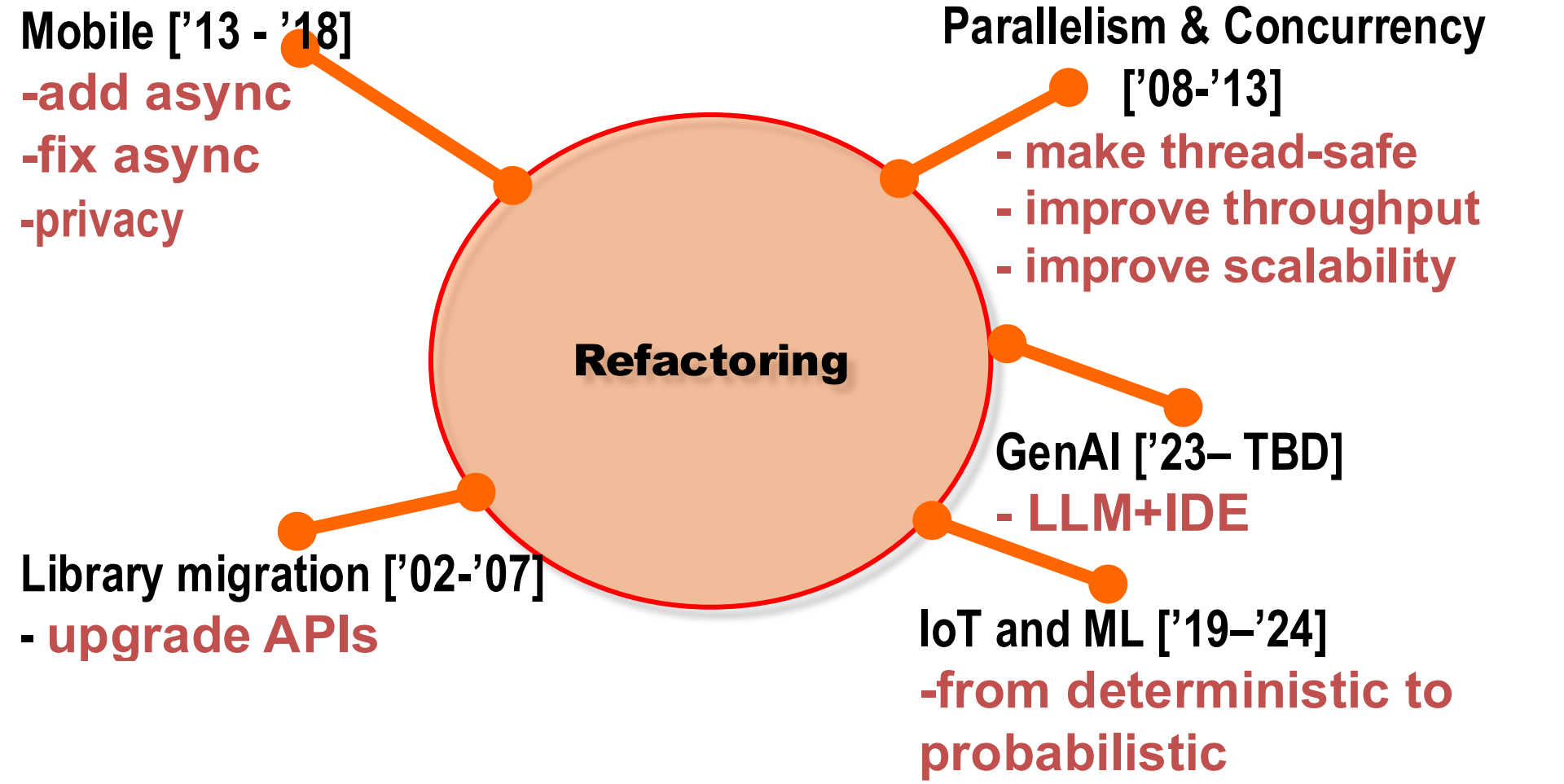
Q2: How can we automate them?

Q3: Can we represent programs as transformations? Archive, retrieve, and visualize them?

Q4: Can we infer higher-level transformations?

Visual Studio NetBeans IDE eclipse Android Studio

Work in Your Strength Zone but Reinvent Yourself



Principles for changing between different programming models

My dream: Practical Impact on SW Development

Automating

-ship with official

- hundreds of accepted patches

- first open-source refactoring

Refactoring

Inferring

- used at 

- dozen labs

founded Workshop on Refactoring Tools, HotSwUp, Dagstuhl S.

Understanding

- shaped APIs in Java and .NET official concurency libraries

-learnparallelism.net
150,000+ visitors

Testing

Recreation


TRIPLE ZERO

On Aug 5, 2015 …

From personal success to significance

From a ladder climber to a ladder holder

Motivation

Dominicana se transforma

Why this survey on Autonomous Agents by Wang et al.?

Synthesizes dozens of agent systems

Proposes a **unified agent architecture**

Covers:
- Construction
- Applications
- Evaluation
- Challenges

What was the most surprising finding from these two surveys?

Before reading this paper, what did you call an “agent”?

Which systems you've used would not qualify as agents after this reading?

What is an autonomous agent?

Classic definition

"A system situated within and part of an environment
that senses and acts over time in pursuit of its own agenda."
— Franklin & Graesser (1997)

Key properties
- Situated
- Acts over time
- Goal-directed

Which of these properties are *rare* in today's LLM tools?

Why LLMs Changed the Agent Landscape

What LLMs brought
- Broad world knowledge
- Language-based reasoning
- Natural interfaces

Contrast
- Traditional agents: narrow, trained
- LLM agents: broad, prompted

Automated Software Engineering (ASE) conference before & after & now

Knowledge is power

Does world knowledge help agents reason — or bias them?

When might *less* knowledge be better?

Unified Agent Architecture

Profiling
Memory
Planning
Action

Most modern LLM-based agents fit this structure.

Which module do current agent systems overemphasize?

Which module is most under-designed?

Profiling Module

What profiling does
- Defines role and identity
- Constrains behavior
- Often prompt-based

Examples
“You are a senior software engineer…”
Persona-driven agents

Questions
Is a role just a prompt — or a constraint?
Should an agent’s role ever change at runtime?

Memory Module

What memory is (and isn't)

- Not just chat history

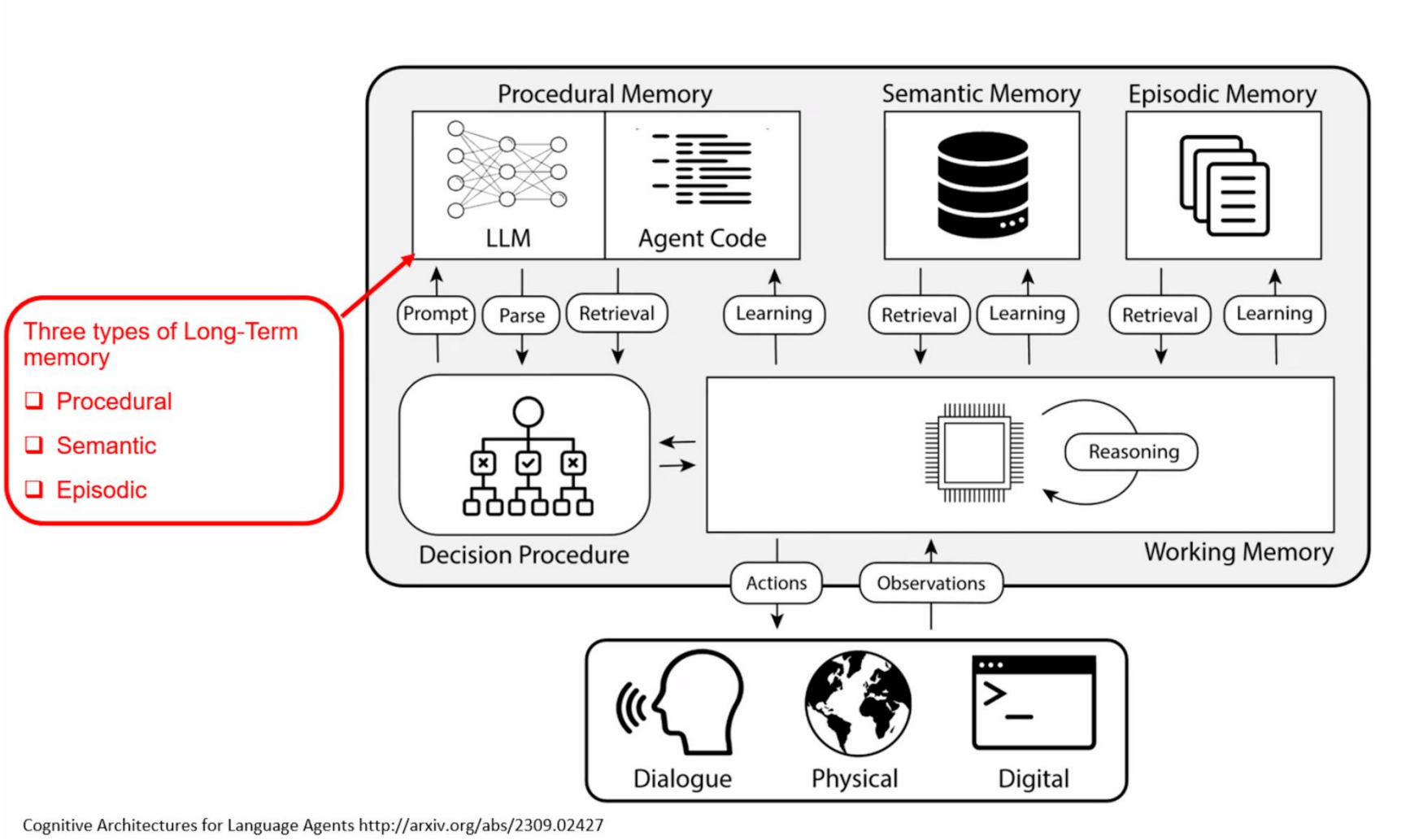
Operations: Read / write Reflection

Structures: Unified, Hybrid (short + long term)

Questions

- What happens if memory is wrong but consistent?
- Should agents ever forget?

Different types of agentic memory



Cognitive Architectures for Language Agents http://arxiv.org/abs/2309.02427

Planning Module

Planning capabilities
- Task decomposition
- Single-path vs multi-path reasoning
- Feedback-driven vs static plans

Examples: Chain-of-Thought, ReAct, Tree-of-Thought

Questions
When does planning stop being helpful?
Can too much planning reduce reliability?

Action Module

Actions make agents risky
- Tool invocation
- Environment interaction
- Side effects are real

SE examples: Editing code, Running tests, Triggering pipelines

Questions
Should agents ever act without reversibility?
What's the most dangerous action an agent could take in SE?

Capability Acquisition & Self-driven Evolution

How agents gain capability: Fine-tuning, Prompting & mechanism engineering

Beyond static agents
Reflection
Learning from trajectories
Multi-agent learning

Promise: Improvement over time

Reality: Hard to evaluate, Hard to trust

Questions
Does an agent need to learn to be useful?
Would you trust an agent that changes its own behavior?

Application Landscape

Where agents are applied
- Social science
- Natural science
- Engineering

Observation: Many are simulations, not deployed systems

Questions
Which applications feel most convincing?
What could go wrong: trust an AI therapist, counselor?
Where does *software engineering* fit?

Evaluation Strategies

How agents are evaluated
- Objective metrics
- Human / subjective evaluation

Challenges
Long-horizon behavior
Non-determinism
No gold standards

The problem with benchmarks

Questions
What would “unit testing” mean for an agent?
How do you know an agent is reliable?

Key Challenges

Hallucinations
Knowledge boundary leakage
Efficiency
Cost

Your Questions

1> What is an agent, really?

- Are LLM-based agents a genuinely new research direction, or a repackaging of prompt-chained LLMs with tools and loops?
- Is “agent” a useful unit of analysis for research, or merely an implementation pattern?
- How should we define autonomy in the era of frozen foundation models?
- Is the distinction between tools, models, and agents conceptually meaningful?
- What does “capability acquisition” mean if model parameters are not changing?

Your Questions – part 2

2> Evaluation, Assessment, and Research Standards

- What should count as the gold standard for evaluating agents: task success, robustness, cost, human satisfaction, or something else?
- Should hybrid evaluation (automated + human judgment) be the default for agent assessment?
- Should survey papers be evaluated differently from technical research contributions?
- How do hallucinations and other LLM weaknesses uniquely affect agent evaluation?
- Which research assumptions break down in real-world or enterprise deployments?

Your Questions – part 3

3> Practical Use, Adoption, and Real-World Impact

- What are the most compelling real-world use cases for LLM-based agents today?
- Which agents mentioned in the paper are actually deployed and used, beyond demos?
- Who is using these agents, and how accessible are they to broader audiences?
- Are social or simulation-based agents (e.g., social AI or educational agents) meaningfully adopted?

Your Questions – part 4

4> Memory, Feedback Loops, and Planning Tradeoffs

- Does long-term memory introduce more failure modes than benefits?
- Under what conditions is planning-with-feedback superior to static planning?
- When do feedback loops meaningfully improve performance versus just increasing latency and cost?
- What are the tradeoffs between fine-tuning models and adding context or memory?
- How do techniques like ToT and feedback loops change agent reliability?

Your Questions – part 5

5>Human Feedback, Oversight, and Autonomy

- How much human feedback is necessary for stable and trustworthy agent behavior?
- When should human intervention be mandatory versus optional?
- Does reducing human involvement meaningfully increase autonomy, or just reduce safety margins?

Breakout Activity: Agent Design

When you build an agent this semester:

- Which module would you design first?
- Where would you limit autonomy?