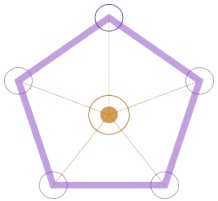


polycog

Reasoning Infrastructure for Trustworthy Agents

Shiwali Mohan | **Nate Derbinsky**

@ Soar Workshop 2026

AI agents are the next paradigm shift.
However agents today are not trusted in high-stakes use-cases.

Reliability = making a correct decision. Agents today are built using LLMs which hallucinate, correctness is not guaranteed.

Controllability = ability to observe, explain, and guide what an agent does. Enterprises do not have the ability to understand or change LLM decisions.

Enterprise trust in fully autonomous AI agents fell from 43% to 27% in a year, driven by reliability, transparency and understanding concerns - McKinsey 2025 Report[1]

Less than half of leaders trust AI agents to make autonomous decisions, and only 8% are comfortable giving them full autonomy – CIO.com 2025 Survey[2]

'You cannot automate something that you don't trust' — today's LLM-based agents still trigger reliability concerns." - Gartner VP[3]

1. Source: McKinsey 2025 Report - Seizing the Agentic AI Advantage
2. Source: CIO.com 2024 Survey Results
3. Source: Gartner

Polycog

- Building **trustworthy AI systems** by…
- **bridging** complementary strengths of (Soar) cognitive architecture and LLMs to **augment** the agentic AI stack for …
- supporting **intuitive** system authoring and **reliable** execution.

Current Agentic AI stack is inadequate

Agents
Replit, Cursor, Harvey, Glean…
Orchestration
Players: LangChain, CrewAI, AG2
Technology: Decision Graphs, Foundation Model
Inference
Players: OpenAI, Anthropic, Google
Technology: Deep Neural Networks
Cloud
Amazon AWS, Google Cloud, MS Azure
Hardware
Players: Nvidia, Intel, AMD
Technology: GPU, CPU, TPU

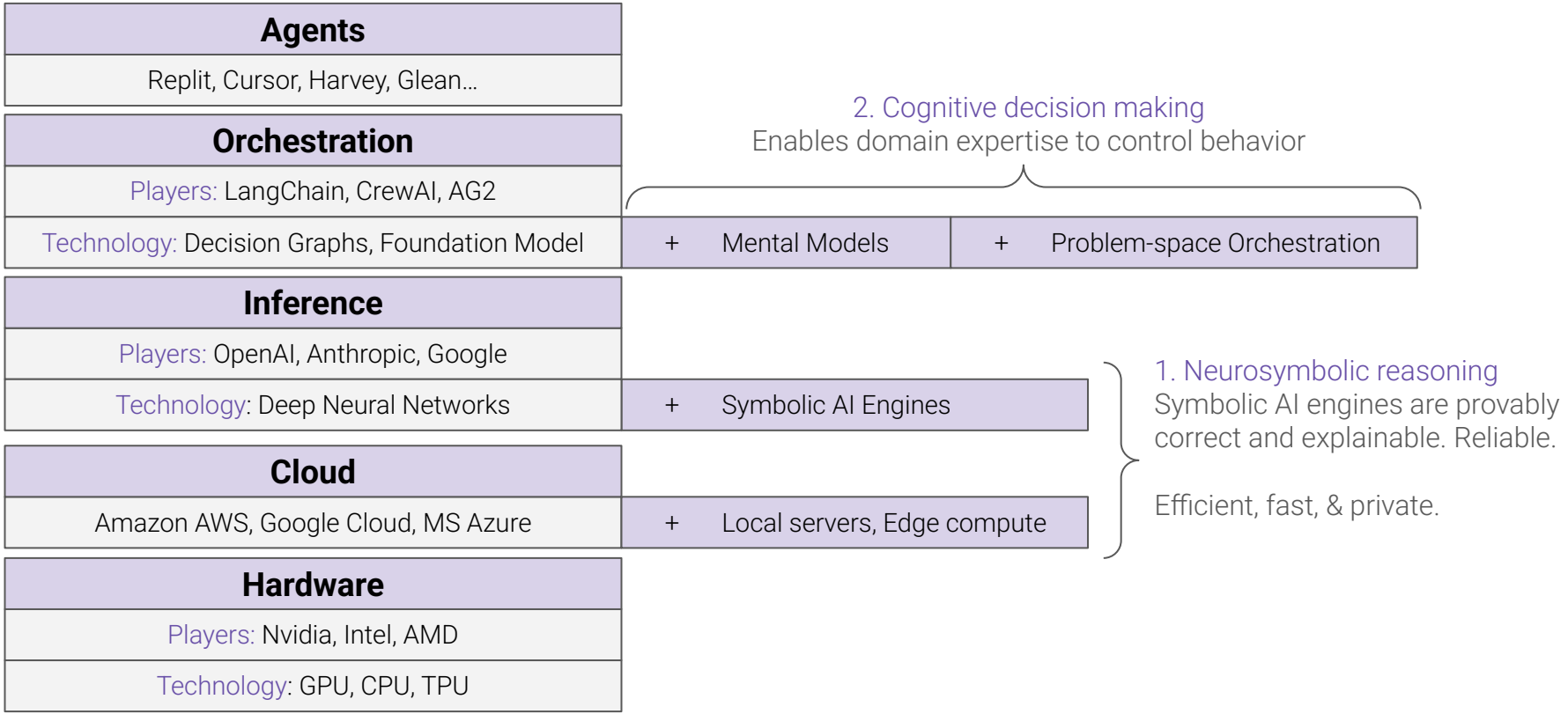
Agents are unreliable because they exclusively depend upon LLMs that hallucinate:

1. They are non-deterministic machines that do not conduct any principled reasoning.
2. They match patterns, but do not actually understand rules and cannot perform deduction, causal reasoning, rule adherence or planning.

Agents are hard to control because the decision logic in LLMs is represented as network weights, which are hard to understand and update.

And, they are expensive and expose private data.

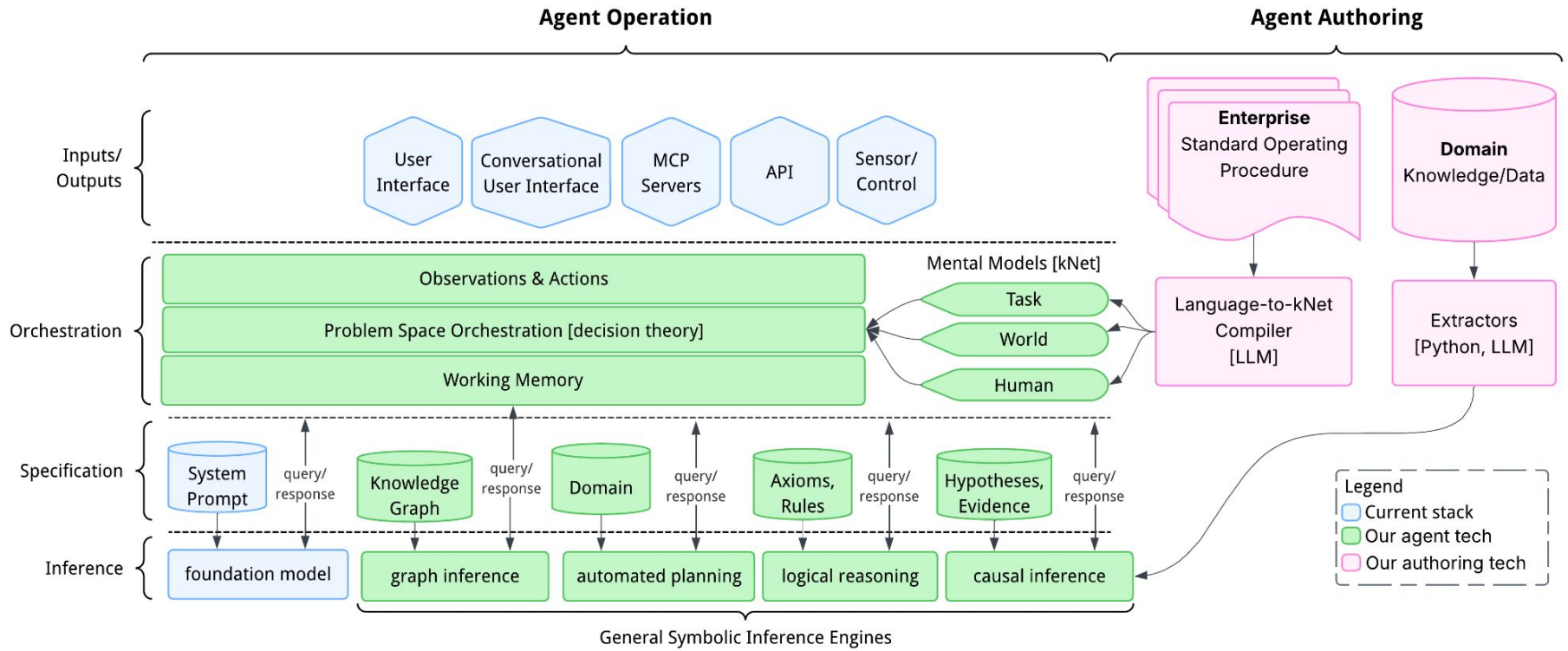
We are extending the Agentic AI stack with full AI arsenal to enable reliable and controllable agents

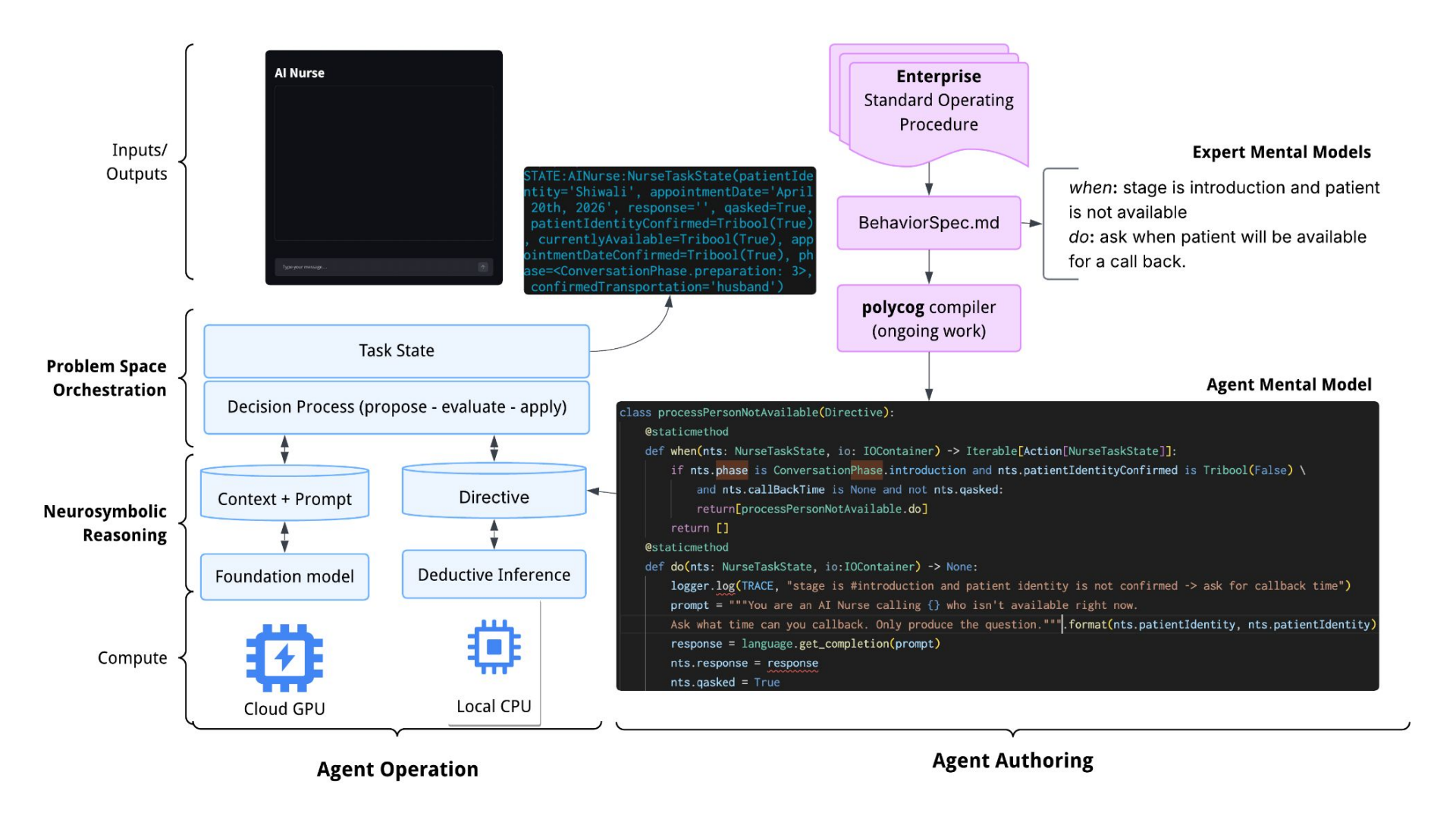


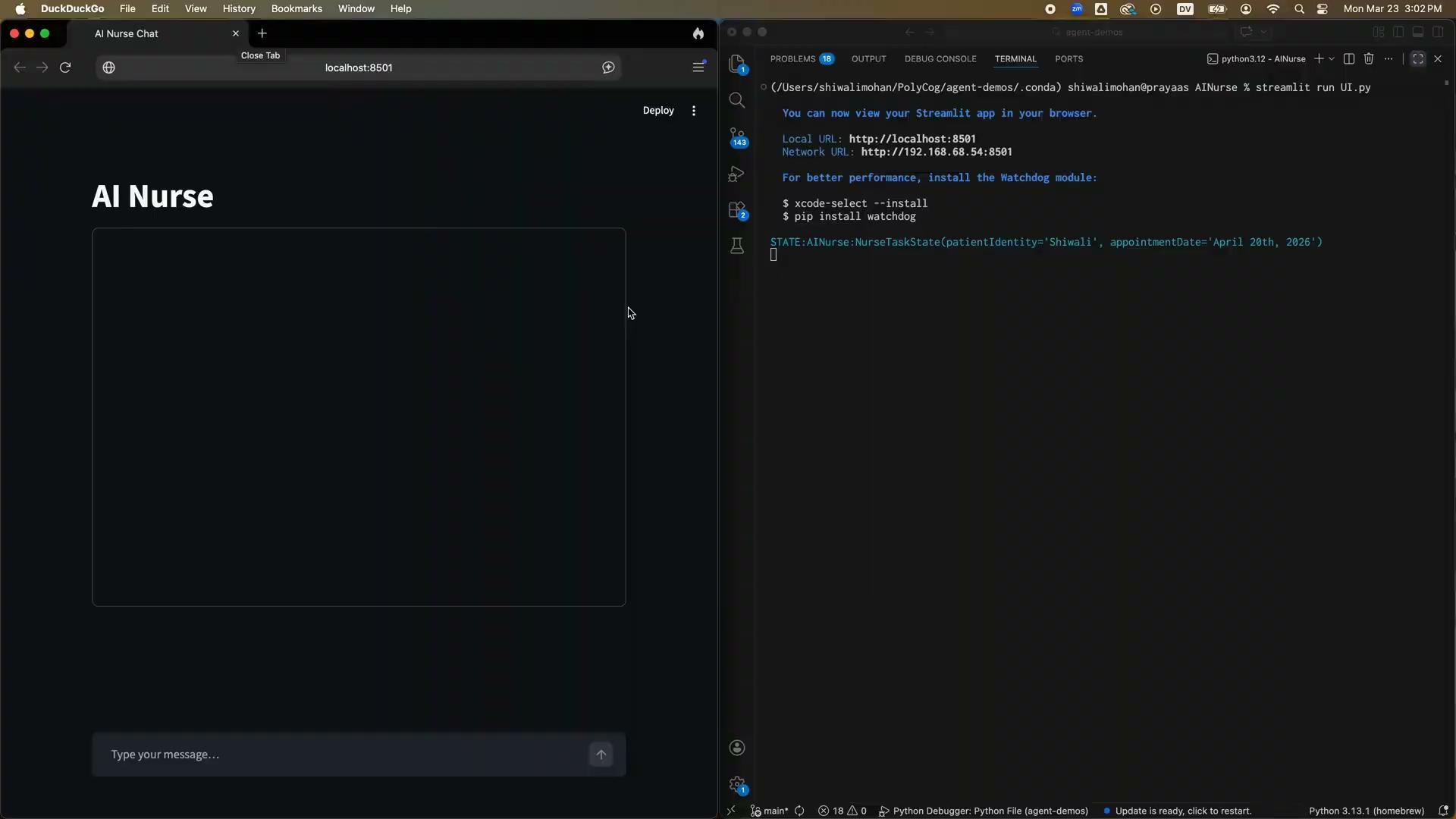
Approach: reliable, controllable, and flexible

	Prompt + RAG	Workflow: LangGraph	Neurosymbolic: PolyCog
Author Define agent behavior	**Easy** Written as a detailed prompt. Easy to write from an SOP.	Complicated Behaviors are nodes in a graph with conditional routing logic. Every scenario must be anticipated upfront.	**Straightforward** Written as independent behavioral directives from the SOP; each is isolated and doesn't interfere with each other.
Reason Analyzing contextual information to produce results.	Unreliable LLMs are prone hallucination & drift; may apply irrelevant knowledge		**Reliable** Applies LLMs (neuro) + symbolic reasoning for provably correct reasoning. Explainable.
Decision Determine what to do next.	**Flexible**, Uncontrollable Determined by LLM machinery, hard to observe and supervise.	Rigid, **Controllable** Driven by routing logic. Predictable on happy paths, but agent fails when a conversation goes off-script.	**Flexible**, **Controllable** Maintains state explicitly. Evaluates all directives, activating those that contextually apply. Multiple concerns are handled simultaneously without routing logic.
Update Change agent behavior.	Risky Easy to edit the prompt, but a small change can unintentionally cause unrelated behaviors.	Expensive, Risky Requires a developer to modify nodes and the routing logic, with the risk of breaking the logic in long SOPs.	**Straightforward** Each directive can be added/edited individually, without needing to rewrite full logic.

Polycog architecture




Inputs/
Outputs
AI Nurse
Type your message...
STATE:AINurse:NurseTaskState(patientIde
ntity='Shiwali', appointmentDate='April
20th, 2026', response='', qasked=True,
patientIdentityConfirmed=Tribool(True)
, currentlyAvailable=Tribool(True), app
ointmentDateConfirmed=Tribool(True), ph
ase=<ConversationPhase.preparation: 3>,
confirmedTransportation='husband')
Enterprise
Standard Operating
Procedure
BehaviorSpec.md
Expert Mental Models
when: stage is introduction and patient
is not available
do: ask when patient will be available
for a call back.
polycog compiler
(ongoing work)
Problem Space
Orchestration
Task State
Decision Process (propose - evaluate - apply)
Agent Mental Model
class processPersonNotAvailable(Directive):
@staticmethod
def when(nts: NurseTaskState, io: IOContainer) -> Iterable[Action[NurseTaskState]]:
if nts.phase is ConversationPhase.introduction and nts.patientIdentityConfirmed is Tribool(False) \
and nts.callBackTime is None and not nts.qasked:
return[processPersonNotAvailable.do]
return []
@staticmethod
def do(nts: NurseTaskState, io:IOContainer) -> None:
logger.log(TRACE, "stage is #introduction and patient identity is not confirmed -> ask for callback time")
prompt = """You are an AI Nurse calling {} who isn't available right now.
Ask what time can you callback. Only produce the question.""".format(nts.patientIdentity, nts.patientIdentity)
response = language.get_completion(prompt)
nts.response = response
nts.qasked = True
Neurosymbolic
Reasoning
Context + Prompt
Directive
Foundation model
Deductive Inference
Compute
Cloud GPU
Local CPU
Agent Operation
Agent Authoring


DuckDuckGo File Edit View History Bookmarks Window Help
Mon Mar 23 3:02 PM
AI Nurse Chat
Close Tab
localhost:8501
Deploy
AI Nurse
Type your message...
agent-demos
PROBLEMS 18 OUTPUT DEBUG CONSOLE TERMINAL PORTS
python3.12 - AINurse
(/Users/shiwalimohan/PolyCog/agent-demos/.conda) shiwalimohan@prayaas AINurse % streamlit run UI.py
You can now view your Streamlit app in your browser.
Local URL: http://localhost:8501
Network URL: http://192.168.68.54:8501
For better performance, install the Watchdog module:
$ xcode-select --install
$ pip install watchdog
STATE:AINurse:NurseTaskState(patientIdentity='Shiwali', appointmentDate='April 20th, 2026')
main*
18 0
Python Debugger: Python File (agent-demos)
Update is ready, click to restart.
Python 3.13.1 (homebrew)

Polycog - Looking Forward!

🏅 Increasing momentum behind neuro-symbolic approaches to building reliable agents

🏅 Polycog is accruing a community of advisors & design partners

🏅 Resources @ **https://polycog.ai**

🪨 Uphill battle for hearts & minds (not to mention funding!) within a violently dynamic AI ecosystem

🪨 Still stealthily building demos, library, tools, and documentation

🪨 Interest in early adoption?