
expoQA®26
MADRID 26th, 27th & 28th May
expoqa.eu

Building Trust in AI Agents

Solving the Communication & Oracle Problems in Agentic Development

expoQA®

Szilard Szell, DevOps & AI Transformation Lead

27 May, 2026

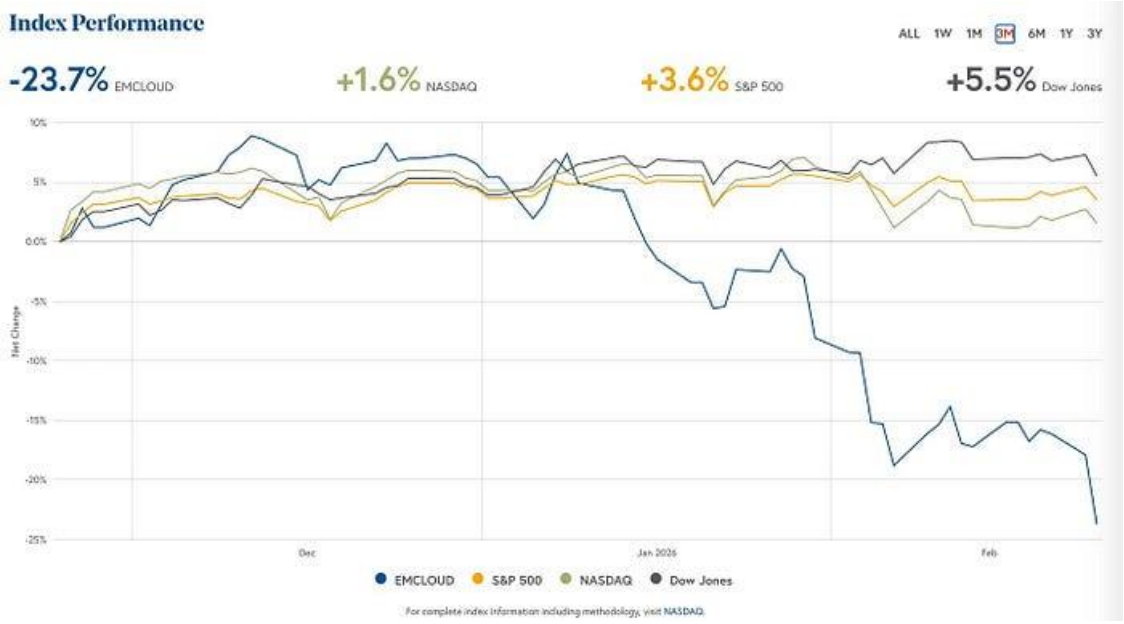
There will be greater change in the next 5 years than in the last 40 years.

Martin Woodward

Vice President Developer Relations, GitHub,

2024

For the first time ever, with this AI wave, people are questioning the terminal value of SaaS.



Lucas Swisher,
Coatue Management,
January 30th 2026

https://medium.com/@vcnewsfr/10-quotes-from-february-2026-that-show-where-tech-is-headed-54454e44573b

Szilárd
Széll

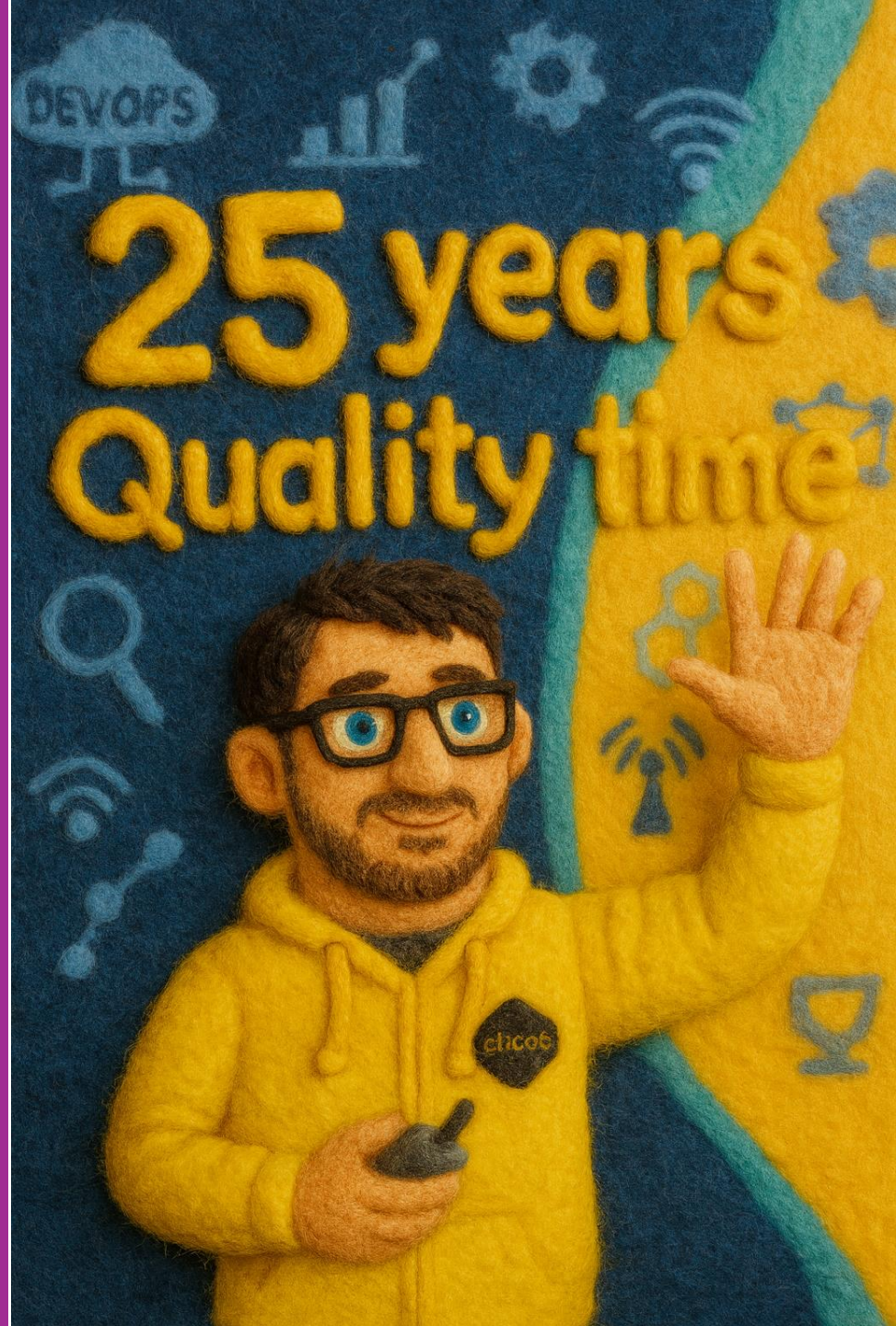
Tester of the Year 2024 in Finland by Tieturi

- ✓ DevOps and AI transformation lead at Eficode
- ✓ Principal Consultant, Test and Quality Coach
- ✓ Agile coach and SAFe SPC, trainer

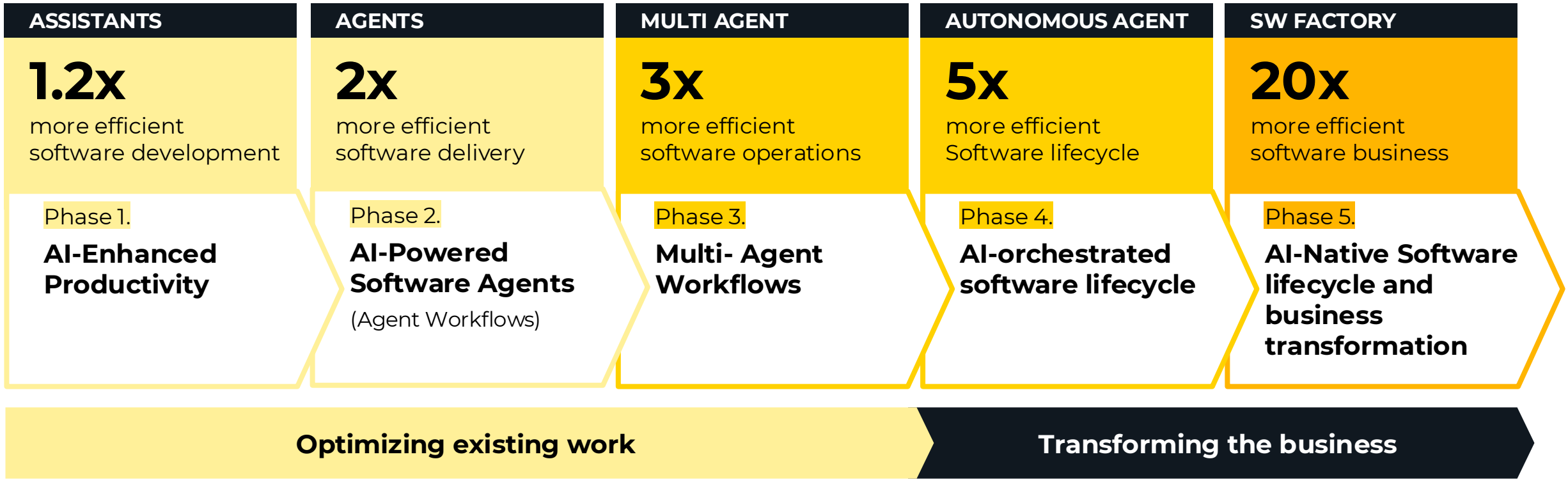
- ✓ Volunteer in ISTQB
- ✓ Public speaker
- ✓ FTMG organiser

Collectibles:

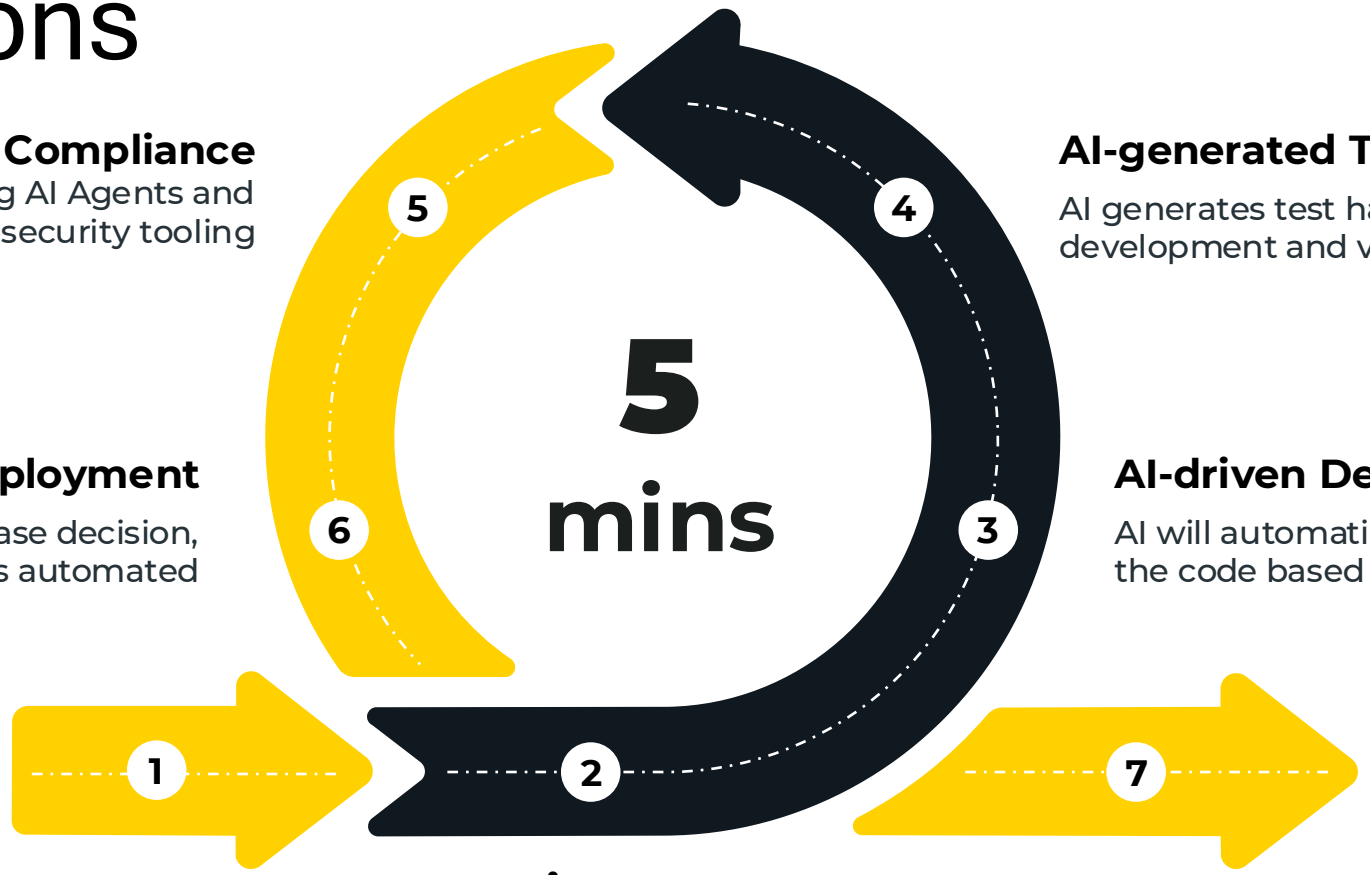
- 25 years of experience in QA and DevOps, mainly Telco
- 14 years of experience as change agent
- SAFe SPC, Certified Scrum Master, DevOps DASA
- ISTQB CTEL-ITP-Full, CTAL-TM, CTFL-AT, CTFL, IREB CPRE
- ITIL4 Foundation
- Lean Six Sigma Green Belt
- Lean Service Creation - Facilitator
- XRAY Certified Expert
- NVIDIA Generative AI - Sales AI Adviser



Your AI Adoption Journey



AI has unlimited potential for modern organizations



Market Discovery
Defines strategic themes and focus areas for the product

Requirements
Generated together with AI, based on both strategy and user behavior

AI-driven Development
AI will automatically generate most of the code based on requirements

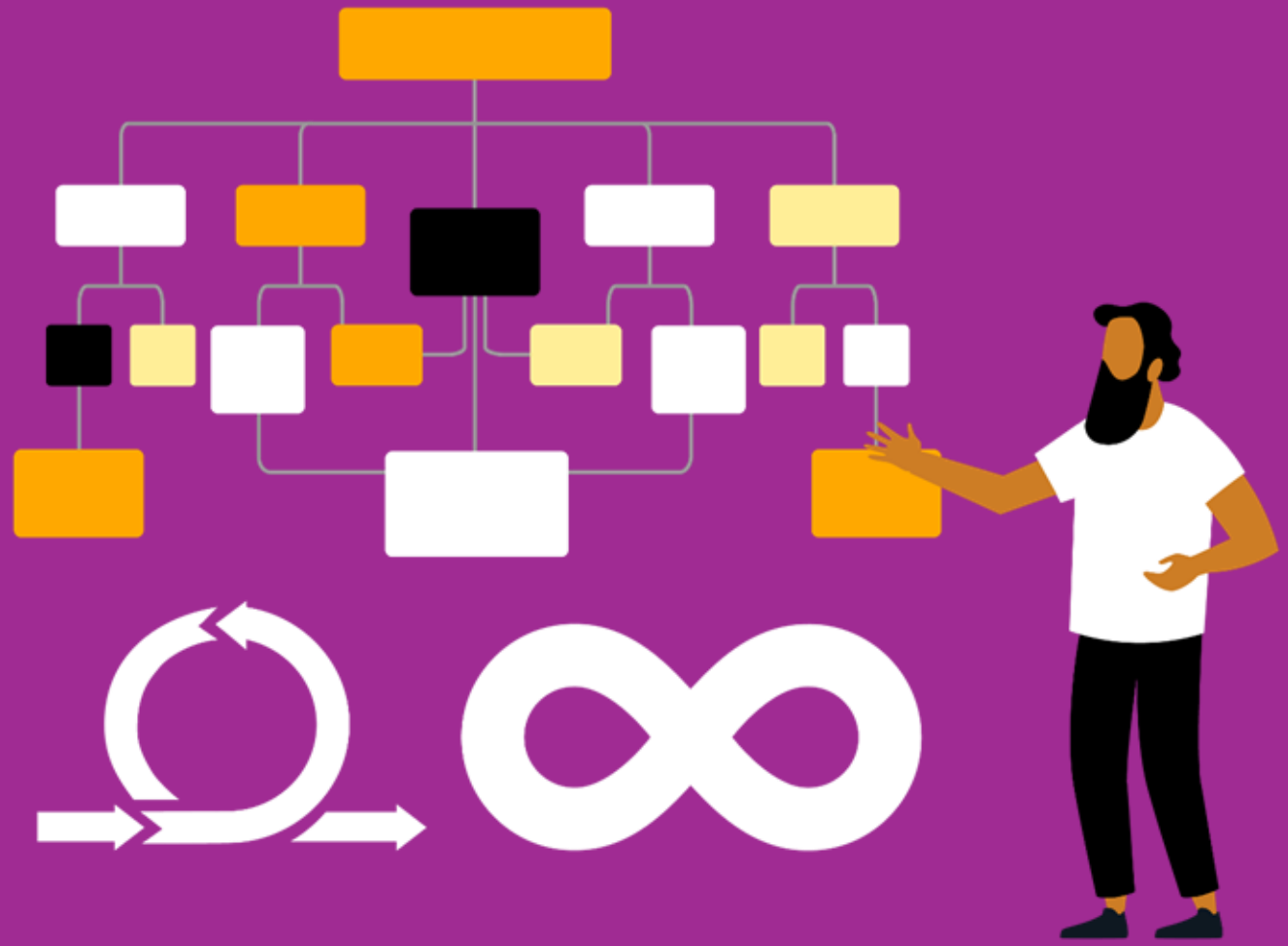
AI-generated Test Automation
AI generates test harness guiding the development and verifying the requirements

Security and Compliance
Automated using AI Agents and modern security tooling

Continuous Deployment
Humans make the release decision, everything else is automated

Continuous Improvement
Product is adjusted based on the user behavior and observability

INTEGRATING AI INTO THE SDLC

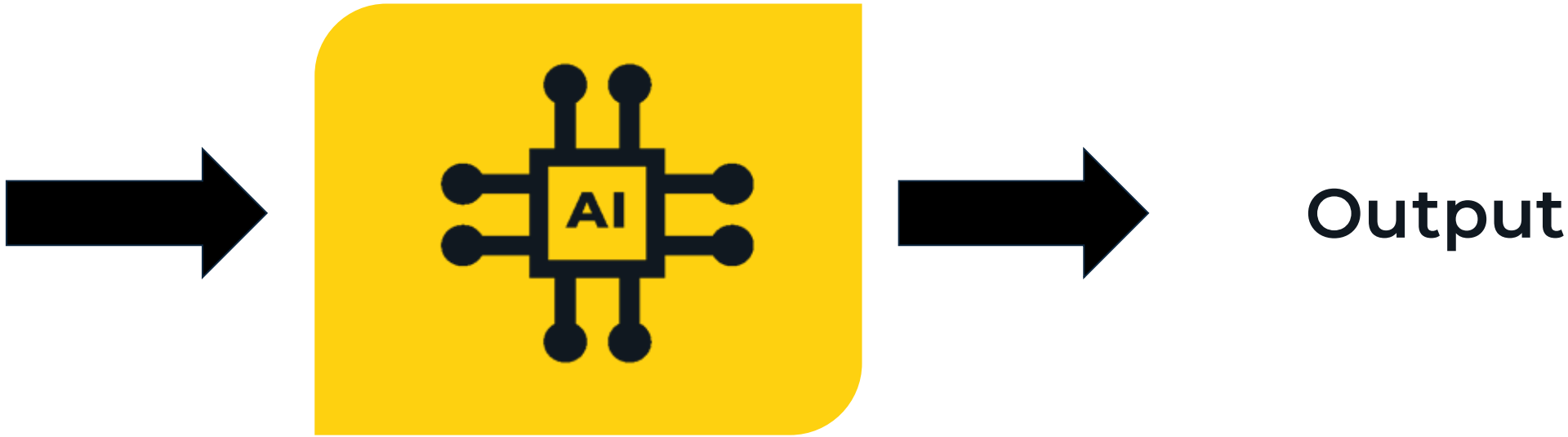


How I (over) simplify the AI world?

Communication problem

Oracle problem

Input

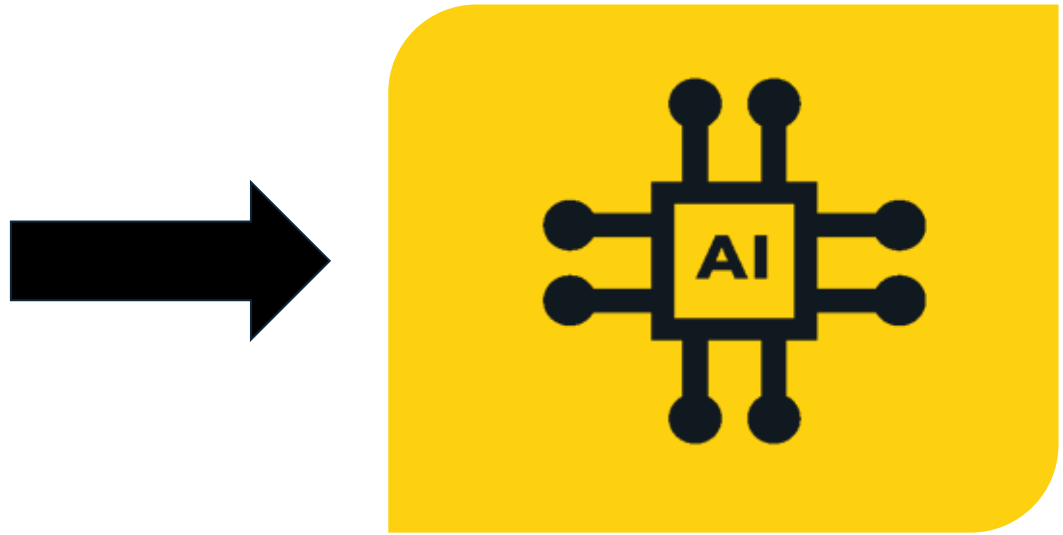


Output

The key to design Agentic development

Communication **AI design** **Oracle**

Intent,
Task,
Context

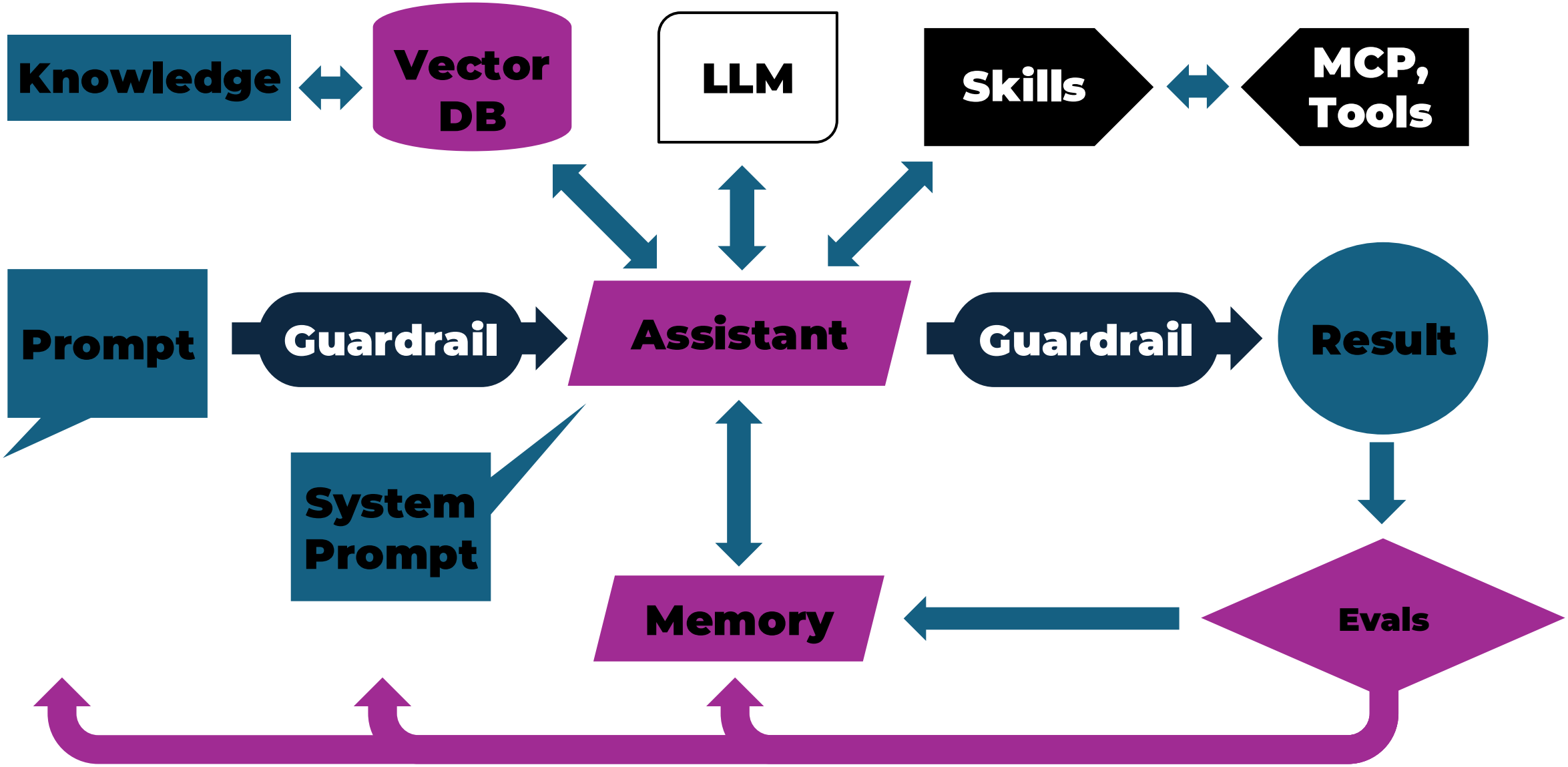


Agents, Tools

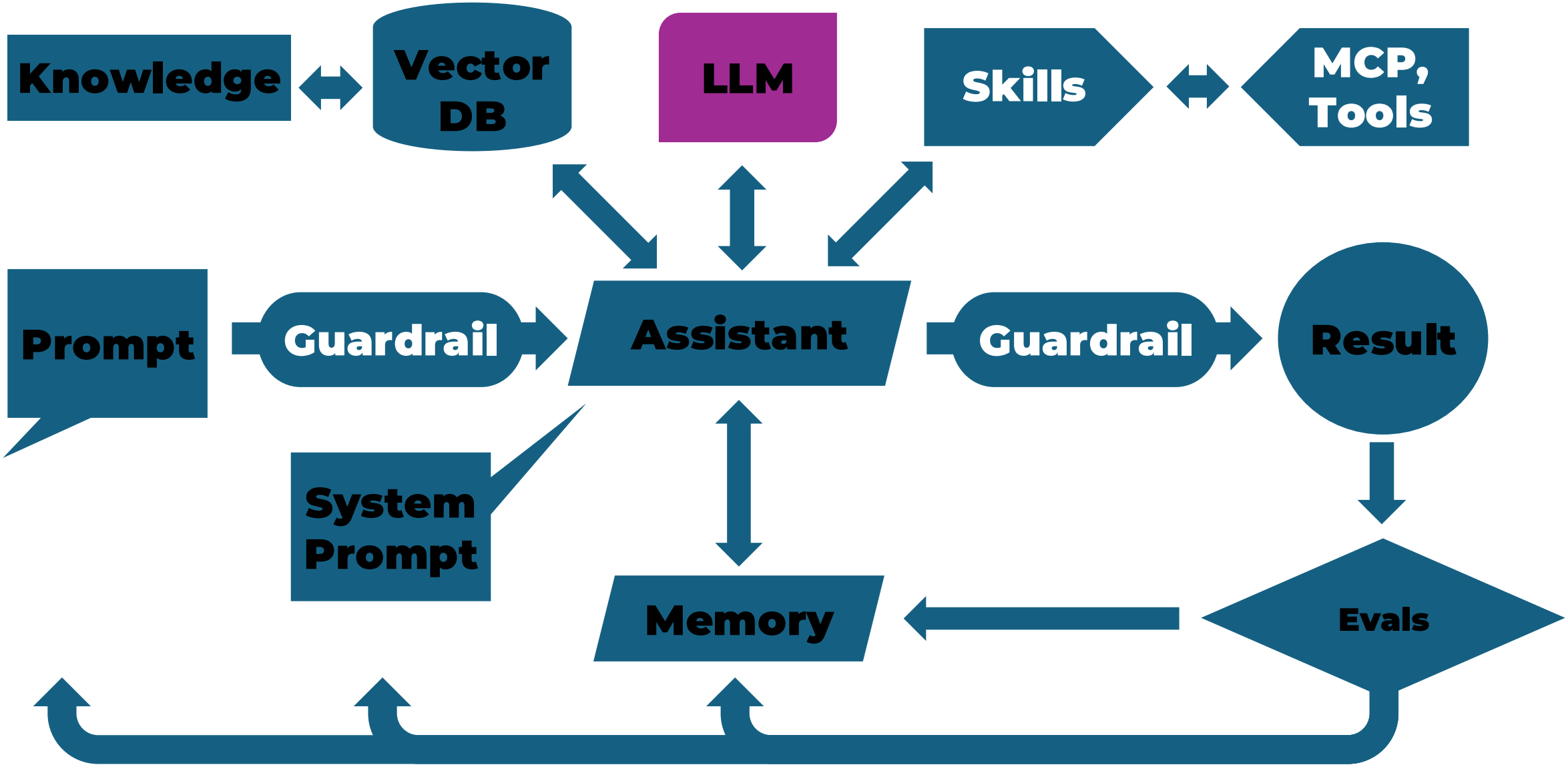
Evaluations

THE BUILDING BLOCK OF AI ASSISTANTS AND AGENTS

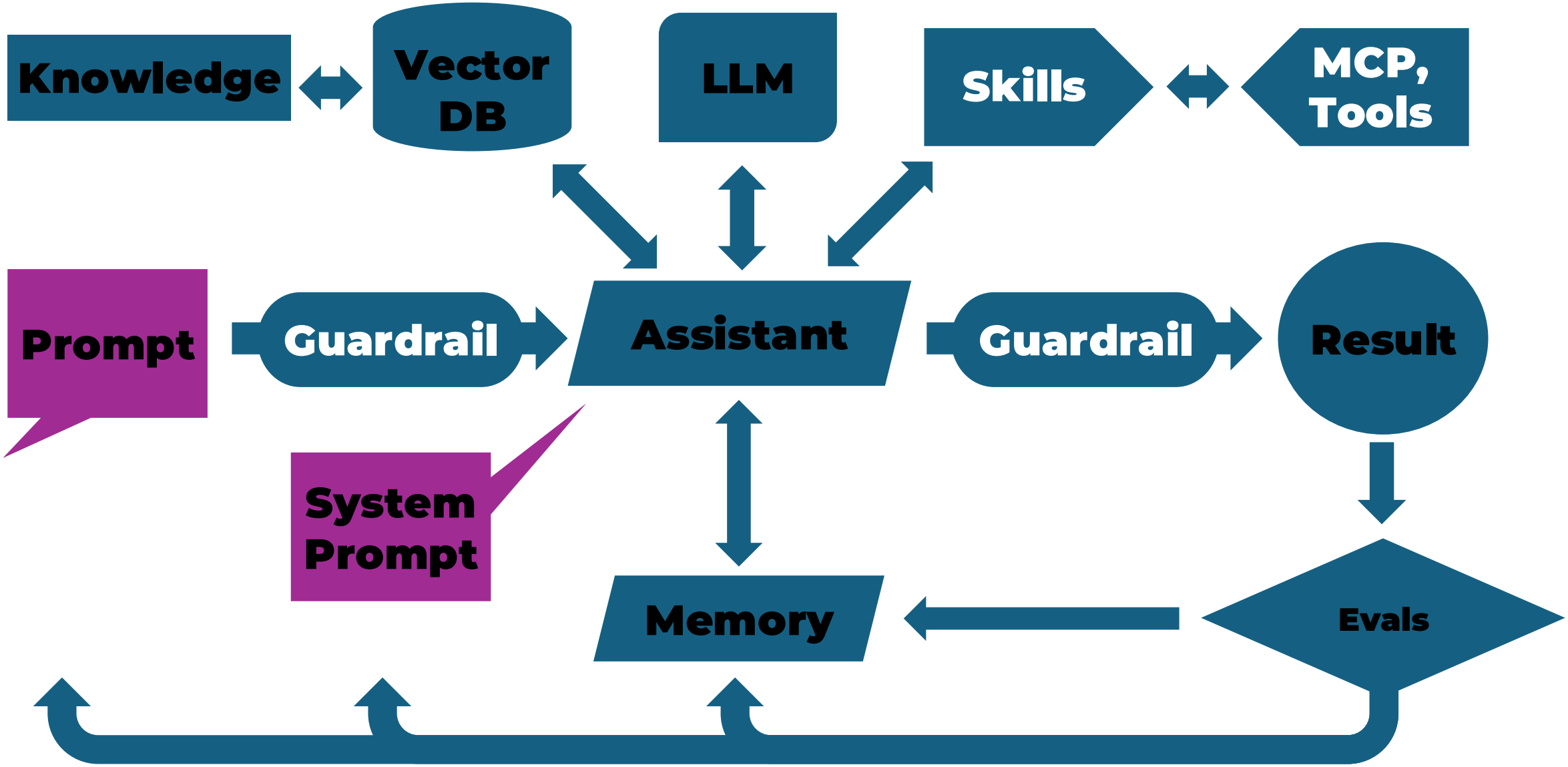
Build trust into AI assistants



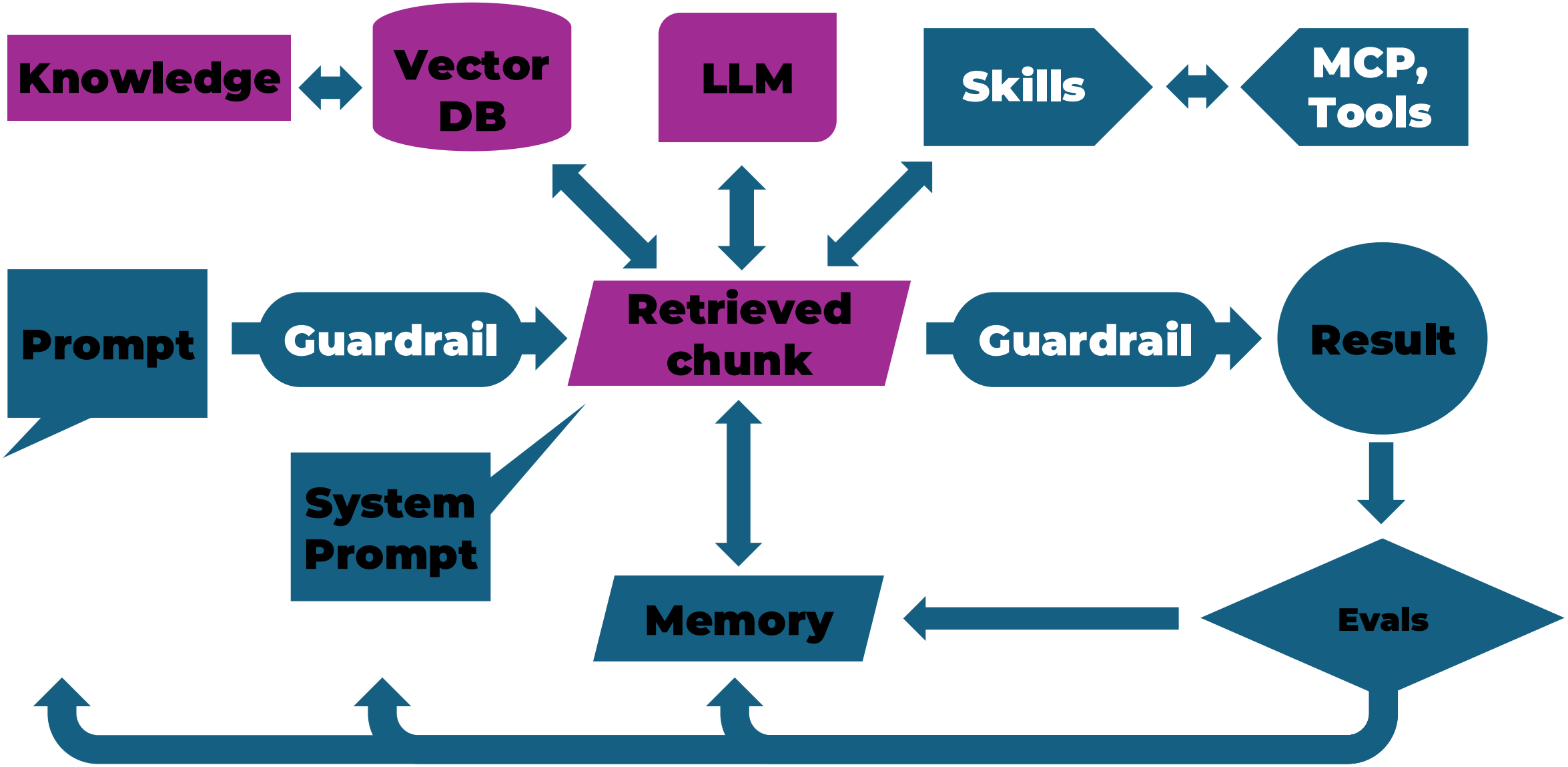
Select the right LLM for your needs



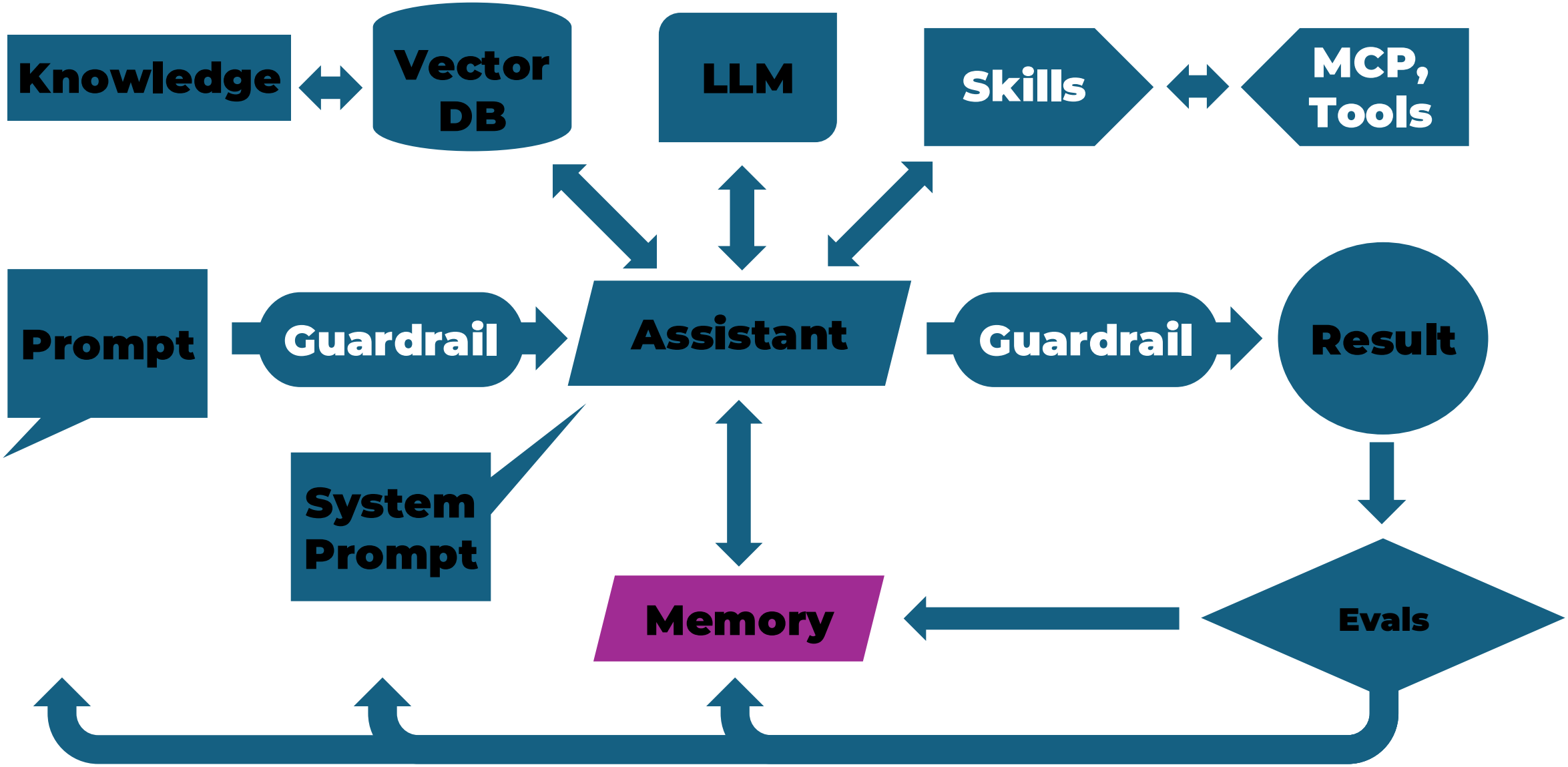
Communicate the intention



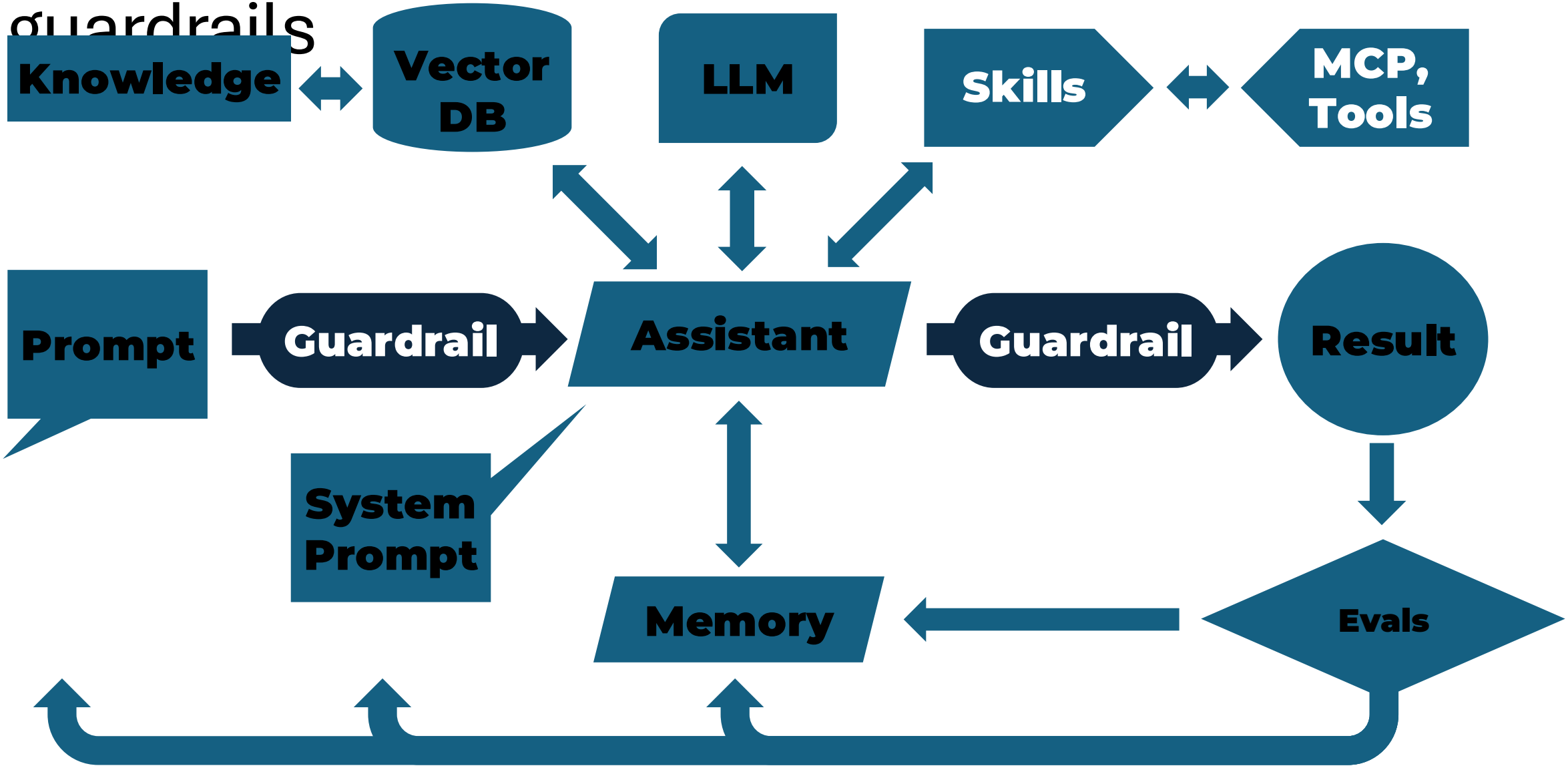
Add local knowledge through RAG



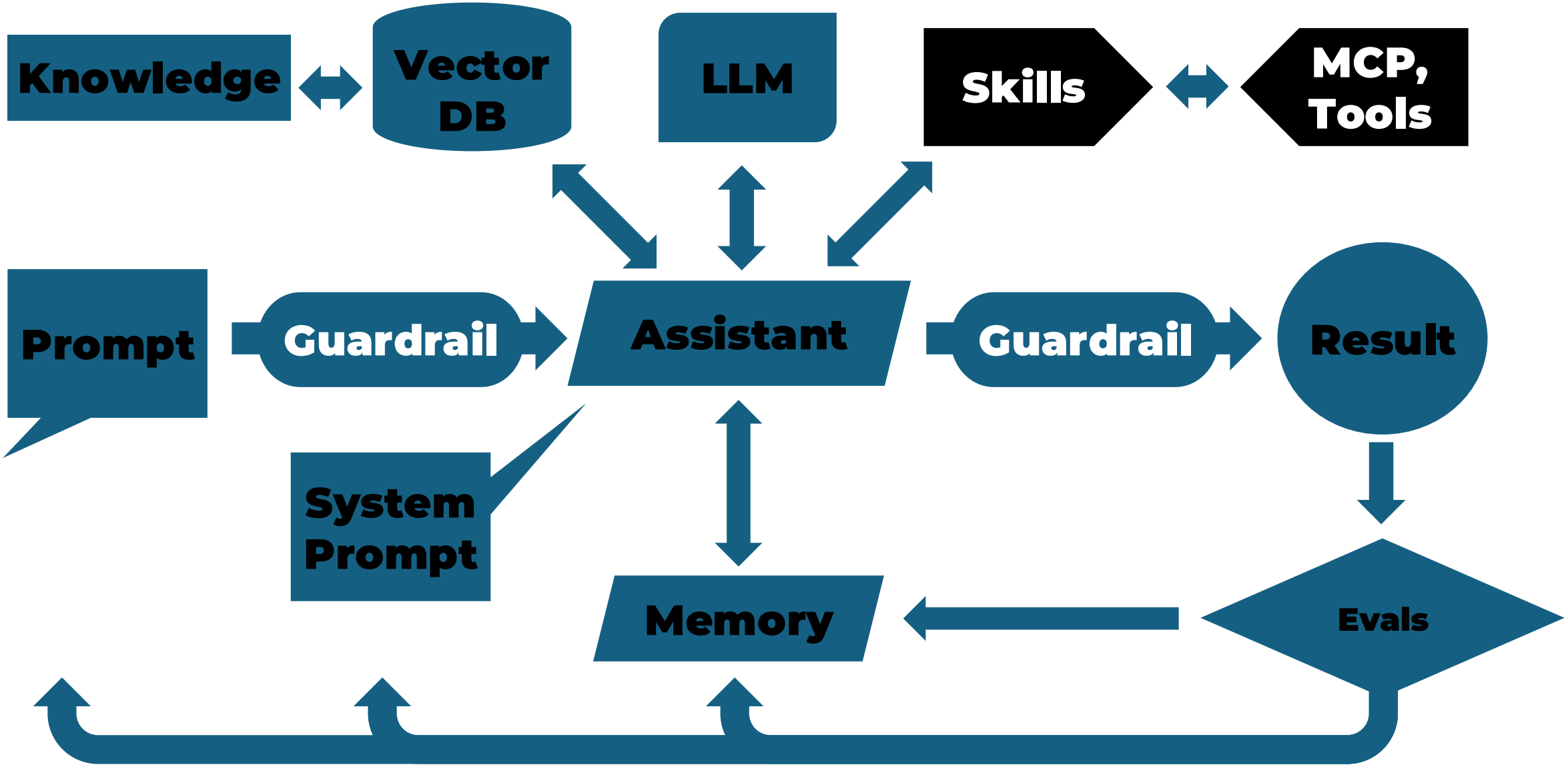
Remember context using memory



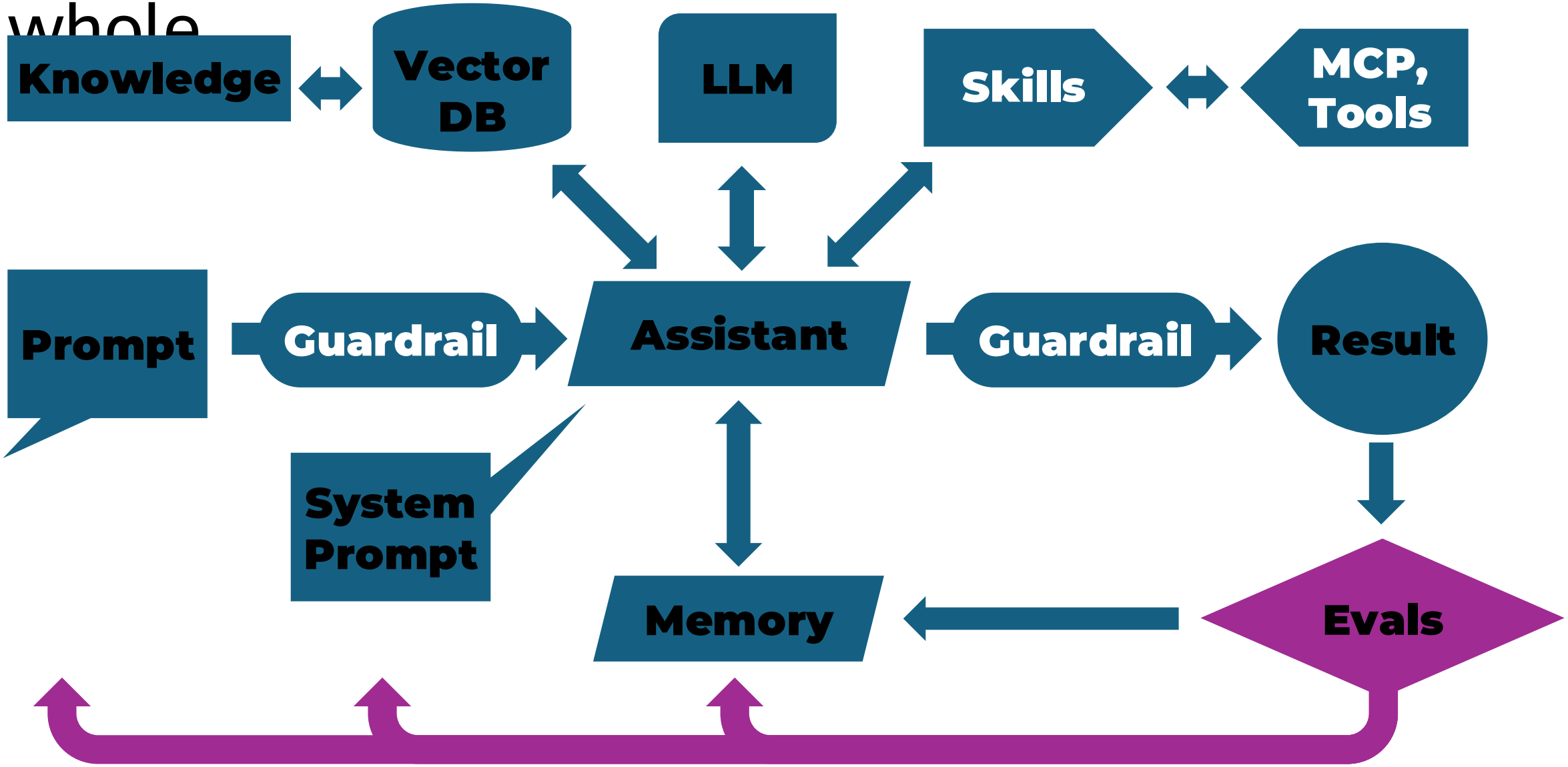
Build responsibility and ethics through guardrails



Add capabilities through tools



Constantly evaluate each component and the whole



THE BUILDING BLOCK OF CODING AGENTS

Chat vs. Agent

Plain Chat / Copilot Assistant

User Prompt → LLM reply → done.
The model suggests; the human executes.
No feedback loop, no tools, no file access.

Coding Agent

The LLM gets a **goal,** is given **tools** and runs in a **loop**. It reads files, writes code, runs tests, observes errors — and repeats.

Key shift: The model is no longer an “oracle” you query — it is an **autonomous actor** that perceives an environment, makes decisions, and changes state.

The ReAct loop: Reason → Act → Observe

Single Loop Iteration

STEP 1

Receive

Orchestrator appends latest message (user or tool result) to conversation

→

STEP 2

Think

LLM receives full context. Generates reasoning. Decides: text, tool

→

STEP 3

Act

Orchestrator checks stop_reason. If `tool_use` → executes

→

STEP 4

Observe

Tool output appended as `role:tool` message. Loop restarts with

Interruption

Orchestrators inject human-in-the-loop pauses for high-risk actions: delete, deploy, external API calls.

Guard Rails

Max turns, max tokens, max cost — enforced by the orchestrator to avoid loop forever.

Complex tasks may take 20–80 iterations.

The anatomy of a coding agent

Orchestration (The Manager)

Single, Hierarchical, Peer-to-Peer Architecture

Memory (The Context)

Short-term, Vector/RAG, Episodic

Reasoning Layer (The Brain)

Intent parsing, task decomposition, self-reflection

Action Layers (The Hands)

Skills, Tools, APIs, MCPs, Browser, Code Execution

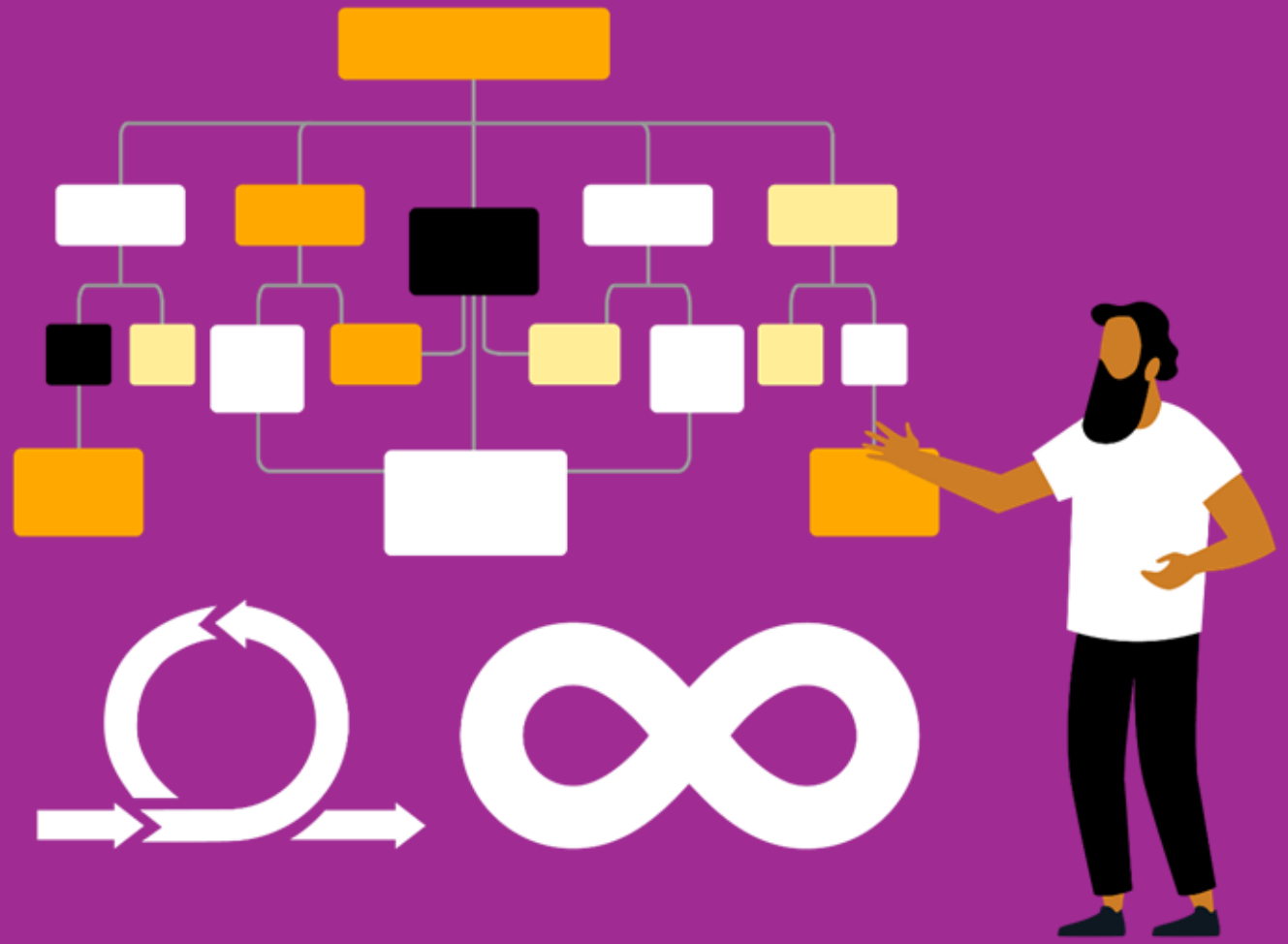
Mandatory Hooks: PreToolUse, PostToolUse, Stop

What goes into the context?

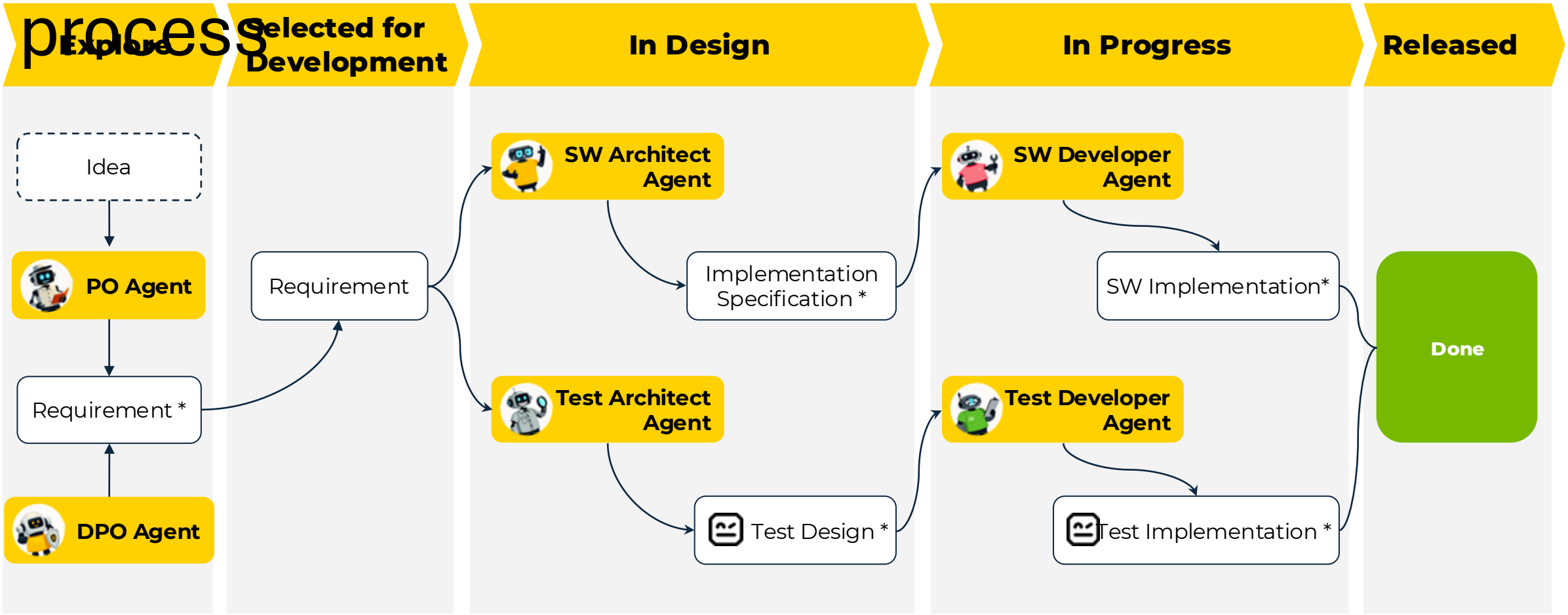
System prompt	**~2k–8k tokens**
Conversation history	**grows unbounded**
Tool definitions	**~1k–4k tokens**
File contents (RAG)	dynamic
Tool results	**accumulates / loop**

Context overflow strategies:
Truncation, Summarization, Sliding window, External vector memory

TRUSTWORTHY AI AGENT WORKFLOWS IN THE SDLC

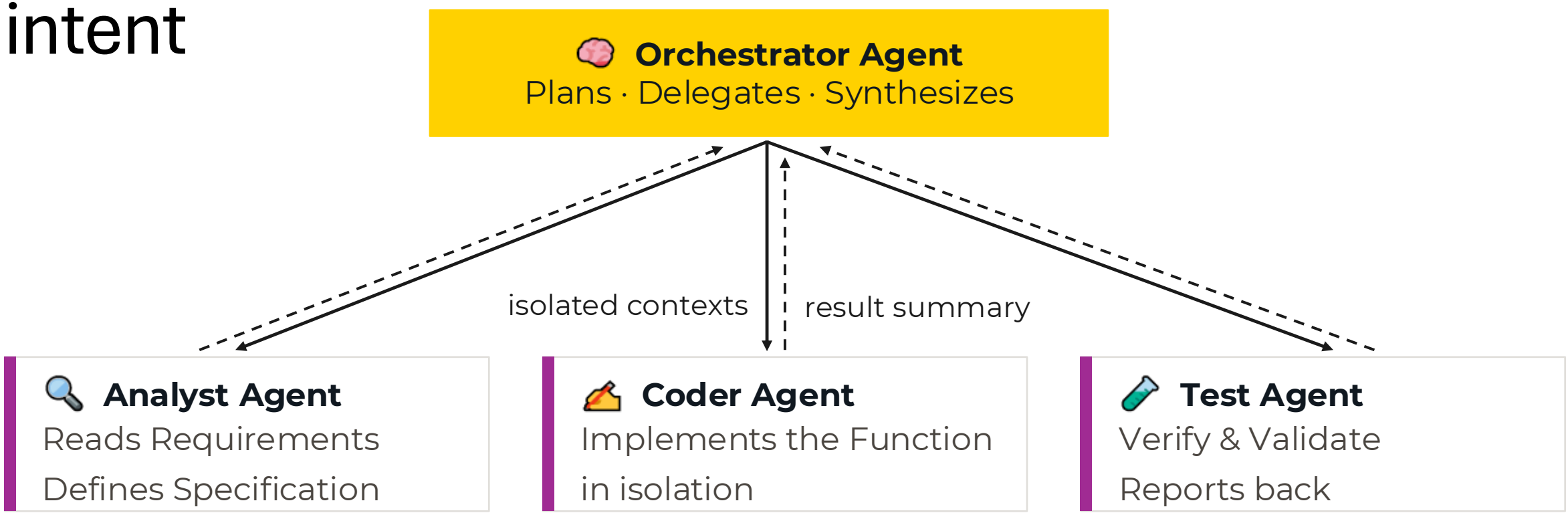


Workflow of single agents following your process



* **Human-in-the-loop: Reviewed and approved by a Human**

Autonomous agent swarm follows your intent

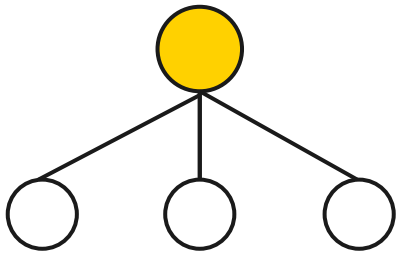


Communication models
Sequential · Parallel · Hierarchical
Shared memory store

Quality implications
Testing of the generated code
Evaluation of the agentic system

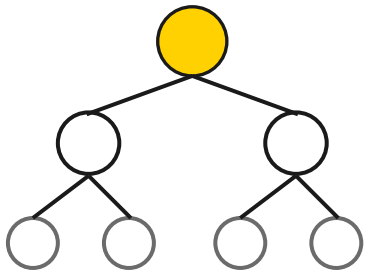
Choose the right multi-agent topology for your goal

Centralized Supervisor



One coordinator routes tasks and aggregates results.

Multi-level Hierarchical

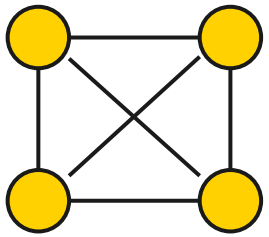


Managers direct specialists.

Sequential Workflow

Each agent refines the previous output.

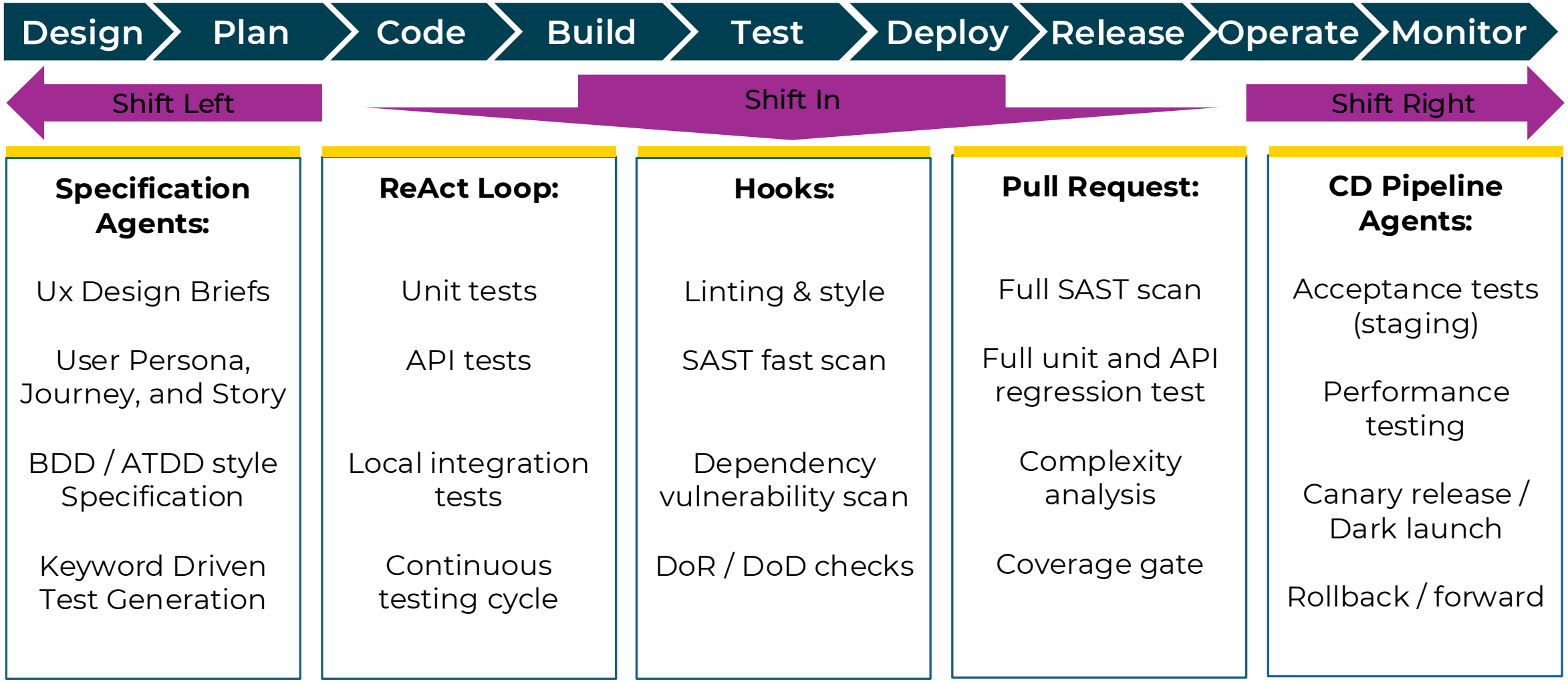
Decentralized Swarm / Mesh



Peer agents hand off via shared state.

Trade-off in one line — centralized topologies are easier to reason about; decentralized topologies are more resilient and parallel.

Add your QE practice to the right feedback level



SUMMARY

Elements of designing trustworthy AI Agents

Communication

What the AI Agent shall do
System prompt, Input Prompt, Reasoning, Orchestration

Remembering Context
Context Engineering, Instructions.md, Memory

Guardrails
Avoid dangerous input to the LLM or to sanitize its results

Foundation Model

Oracle

Foundation Model
Model Cards, Fine-tuning, Evals leaderboards, Benchmarks

Access to knowledge
RAG, MCP, Skills, Tools, A2A

Feedback Loops
LLM-as-Judge, Human-in-the-loop, Red Team, DevOps pipeline

AI Orchestration Platform

What’s really stopping AI adoption?

It’s not the tech—it’s the fears

Job Displacement Fear

Will AI replace developers?

Empower, not replace. AI automates routine tasks so teams can focus on high-value work.

Ownership & Control

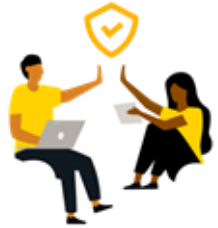
Who owns AI in software workflows?

Define clear AI roles across Dev, Ops, and Security, so AI becomes an asset, not a risk. Governance is key

Security & Compliance

Can we trust AI in development?

Build AI governance into DevOps, ensuring traceability, compliance, and secure automation.



Quality

Will AI produce high quality?

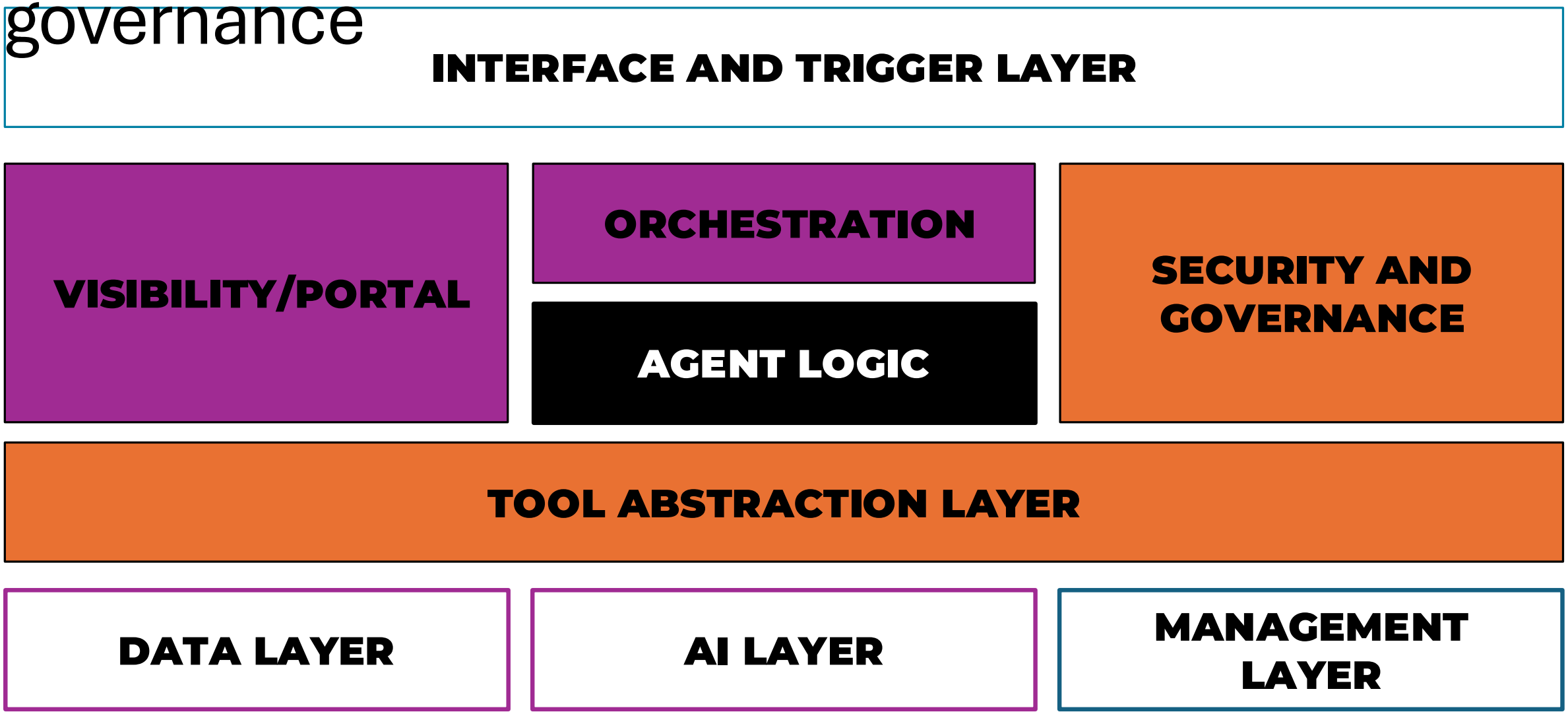
Understand how AI works, reuse your current processes, and rely on your CI/CD pipeline. **Shift In!**

Thank you

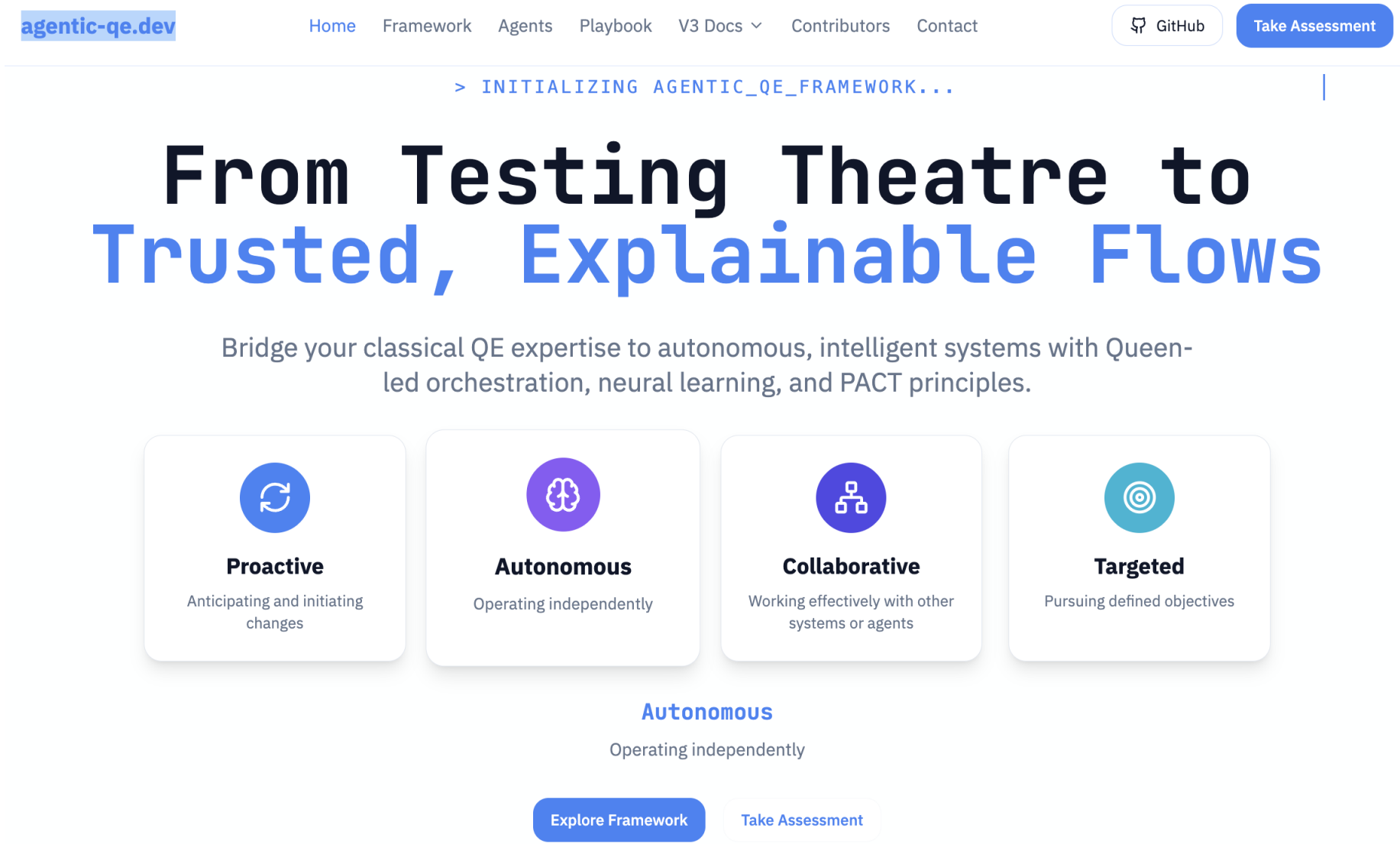
https://www.linkedin.com/in/szellszilard/

https://www.eficode.com/szilard-szell

AI agent Orchestration Platform adds governance



Add feedback with Agentic Quality



expoQA®26

MADRID 26th, 27th & 28th May

Thank you for attending

expoqa.eu