ORACLE

Beyond the Context Window: Building Memory-Aware AI Agents with MCP and Oracle AI Database

Developer Coaching Webinar

Srinidhi Sathyamurthy

October 01, 2026

Speaker



Srinidhi Sathyamurthy

AI Developer Advocate, Oracle

Who Am I?

- AI Developer Advocate at Oracle.
- Computer Science background with a Master’s in AI.
- Create Oracle Developers content across blogs, notebooks, videos, and hands-on demos.
- Work across Oracle AI Agent Memory, MCP workflows, vector search, and agentic application patterns.
- Explore how MCP and Agent Memory connect with coding agents such as Codex, ChatGPT, Cursor, and Antigravity.

The Developer Problem

Developer Problem

- Agents need continuity across sessions, but uncontrolled memory can preserve stale context or expose the wrong data.
- Developers need a repeatable workflow for what gets saved, who can retrieve it, and how memory reaches the agent.
- The session focuses on a practical pattern: extract, approve, secure, connect, and recall memory.

What Is Agent Memory

- Agent memory is durable context that an application can store and retrieve across sessions or tasks.
- Useful records include preferences, project facts, decisions, constraints, follow-ups, and summaries.
- Good memory is typed, scoped, searchable, and safe to inject into a later prompt.

Why Agent Memory Matters

- Memory reduces repeated setup across sessions.
- Memory gives agents a stable way to reuse preferences, decisions, constraints, and follow-ups.
- Governed memory keeps recall useful without creating an uncontrolled shadow data store.

Memory Beyond Chat History

- Chat history records what was said.
- Agent memory records what the application chooses to make reusable.
- A human or policy step can decide which extracted candidates become durable memory.
- That approval step is the difference between useful recall and uncontrolled context buildup.

What Is MCP?



- Model Context Protocol gives agents a standard way to call external tools and read controlled context.
- For developers, MCP turns application capabilities into explicit tool contracts.
- The important design point is control: the agent calls approved tools through a defined interface.

How MCP Fits



- MCP sits between the agent client and the application or database capabilities.
- In this session, MCP exposes memory tools such as save_memory, search_memory, and list_memories.
- The workflow stays inspectable because memory writes and reads pass through controlled tools.



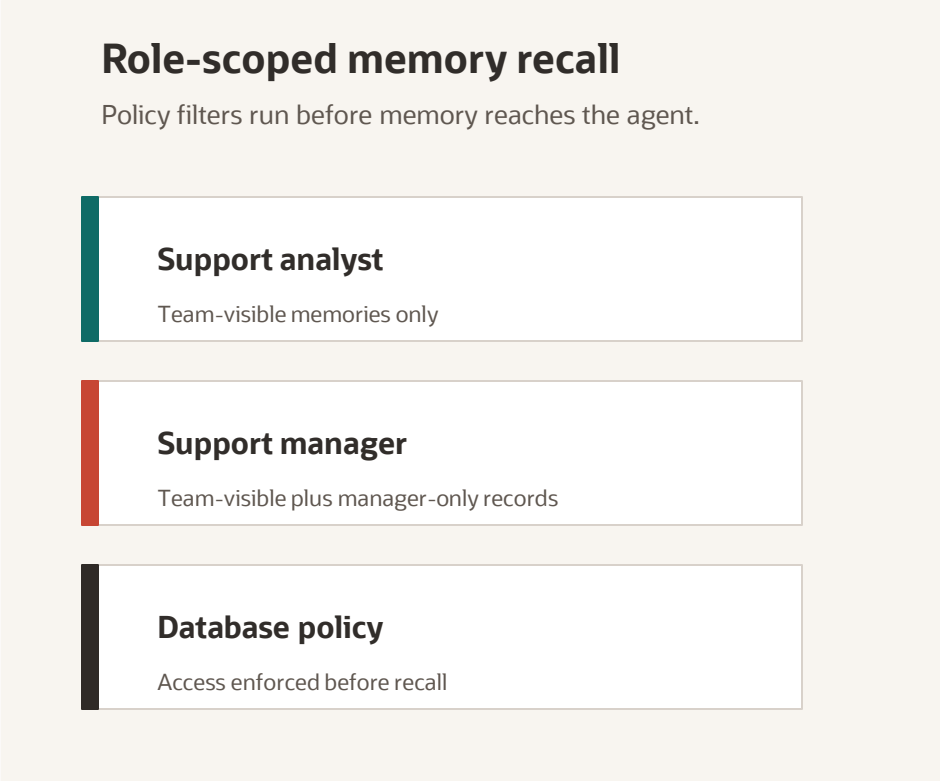
Where Oracle AI Agent Memory Fits

- Oracle AI Agent Memory provides the persistent memory layer backed by Oracle AI Database.
- The application decides what gets stored and which user, agent, thread, or role can retrieve it.
- The local demo rehearses the workflow, then maps the same pattern to Oracle AI Agent Memory.
- Current version: 26.8.

Deep Data Security

Enterprise Control

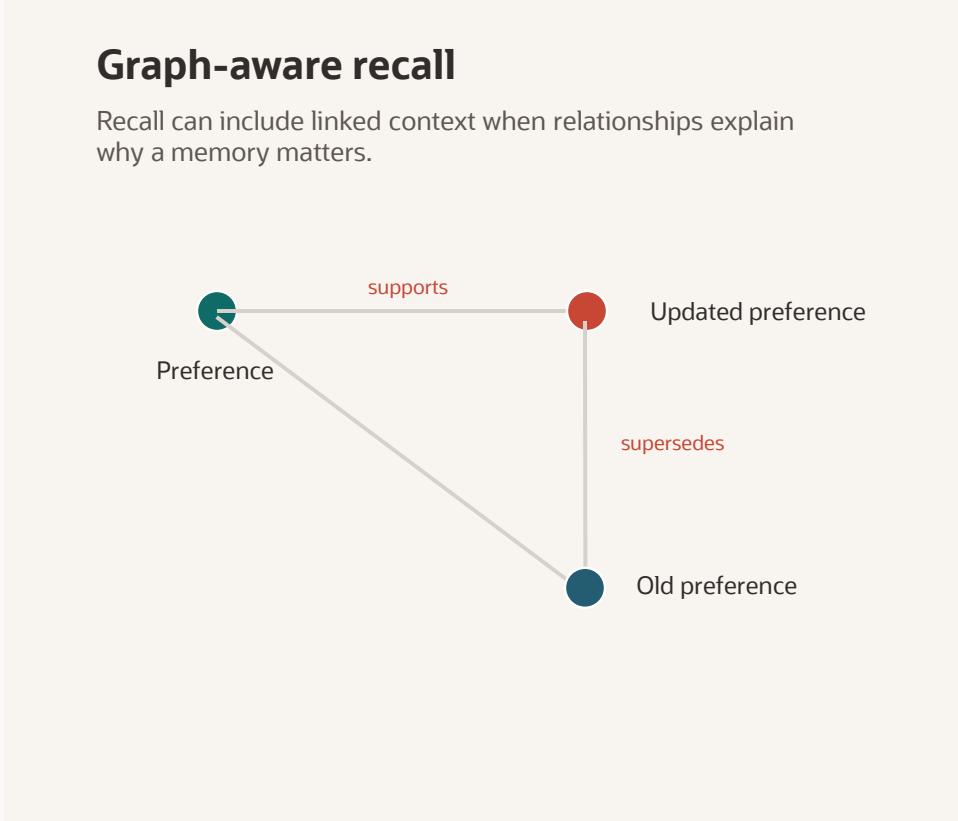
- Enterprise agents should not see every memory just because the data exists.
- Deep Data Security keeps access decisions close to the database layer.
- A support analyst can receive team-visible guidance, while manager-only escalation details stay hidden.
- This matters for industries such as banking, healthcare, and finance where roles and data boundaries are strict.



Graph-Aware Retrieval

Enterprise Recall

- Memories are rarely isolated facts. A new preference can supersede an older one.
- A decision can support a constraint, and a follow-up can refine a project fact.
- Graph-aware retrieval uses these relationships to return the direct match plus linked context.
- The demo shows links such as supersedes so the audience can see why related memories appear.



What Developers Will Learn

Demo Outcome

- Separate source notes, candidate memories, approved records, graph links, and recall results.
- Use MCP as a controlled layer for memory operations.
- Apply role-based access before memory reaches the agent prompt.
- Use graph-aware retrieval to explain memory evolution, not just keyword matches.

XYZ Support Copilot

Demo Context

Company

XYZ Support

Fake SaaS support team

Source

Notion

Working notes exported as Markdown

Goal

Agent memory

Governed recall for support answers

- XYZ Support is a fake SaaS company with Starter, Growth, and Enterprise customers.
- The support team keeps working notes in Notion: preferences, policy facts, decisions, constraints, and follow-ups.
- The demo exports those notes as Markdown, extracts memory candidates, approves useful records, and recalls them in a new support session.

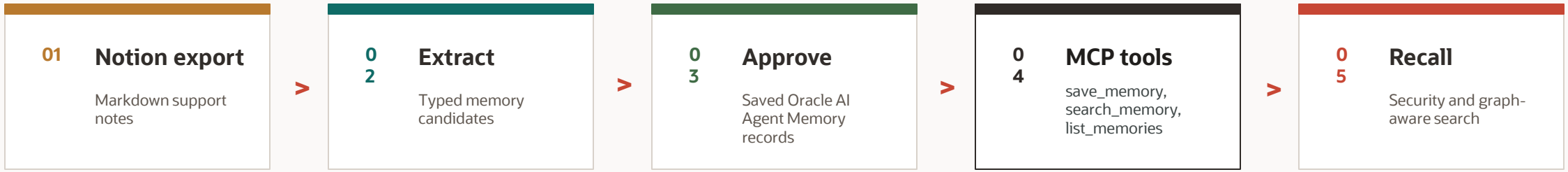
Why this example works

It is familiar enough for the audience to follow, but still shows access control, memory evolution, and recall behavior.

Demo Architecture

One controlled path from note to recall

The agent uses approved tools and scoped memory instead of reading every note directly.

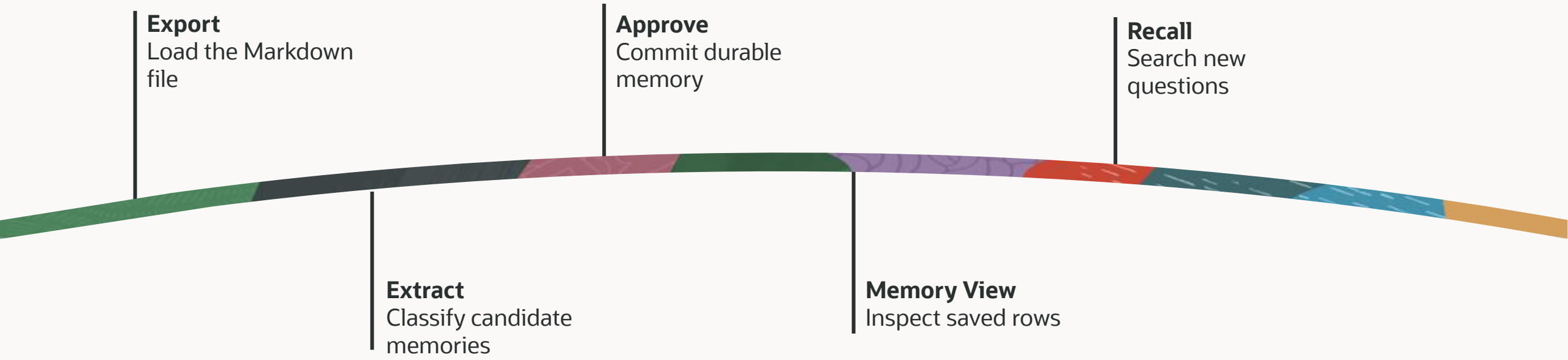


Control point	Every memory write and read passes through an explicit tool contract, so the workflow stays inspectable.

Demo Walkthrough

- Load the Markdown export and explain where it came from.
- Extract typed memory candidates and explain confidence scores.
- Approve useful memories and inspect raw rows.
- Open Graph View to show memory clusters and links.
- Run recall as support analyst and support manager, then compare graph-aware retrieval off and on.

Demo steps


Export
Load the Markdown file
Extract
Classify candidate memories
Approve
Commit durable memory
Memory View
Inspect saved rows
Recall
Search new questions

Connect with me

LinkedIn QR code

Srinidhi Sathyamurthy

ORACLE
Data Deep Dive
at AI World

WEDNESDAY, OCTOBER 28 · 9:00–11:00AM PT · LAS VEGAS

Codex + Oracle AI Database Hackathon

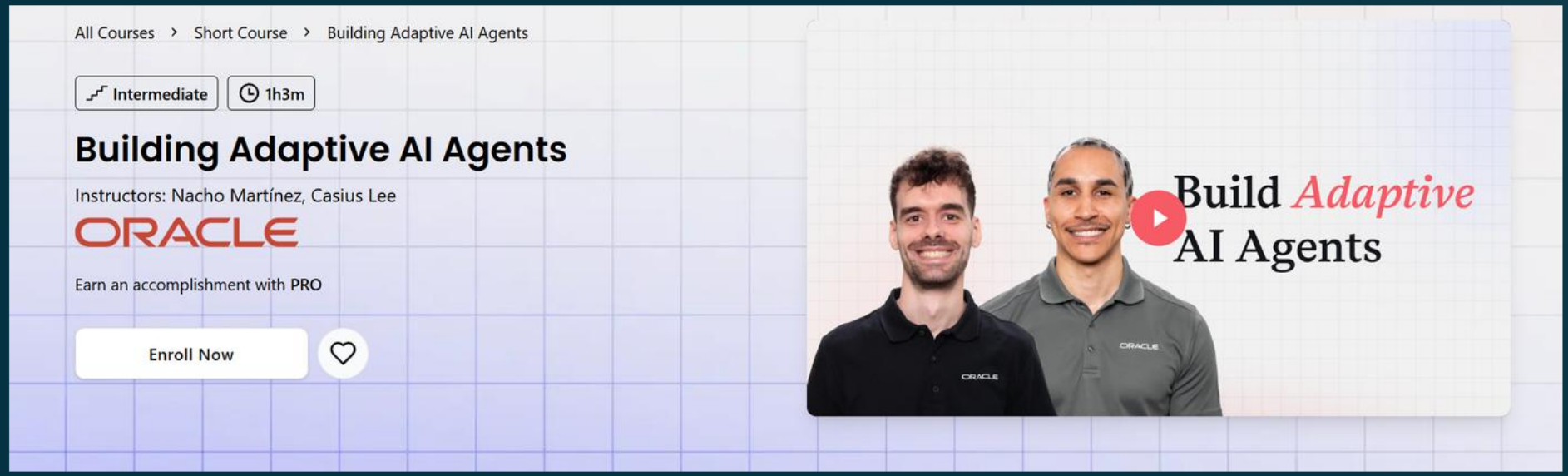
Bring an idea. Leave with a working prototype.

Register separately from AI World

Session HAC1394 in the AI World session catalog

Build agents that improve with experience

DEEPLEARNING.AI + ORACLE



SCAN TO ENROL

deeplearning.ai

Turn agent traces into reusable skills and better retrieval.

1h 3m | 7 video lessons | 3 code examples

Thank you




ORACLE