Enterprise application workflows with an Agentic AI Swarm on MCP

Introduction

With evolving enterprise landscape, organizations must process and analyze transaction data, invoices, and regulatory requirements with speed, accuracy, and auditability. Traditional batch pipelines and monolithic applications struggle to keep pace with evolving tax laws, diverse data formats, and the need for real-time insights. To address these challenges, our solution leverages a swarm-based Agentic AI architecture powered by the **Model Context Protocol (MCP)** framework.

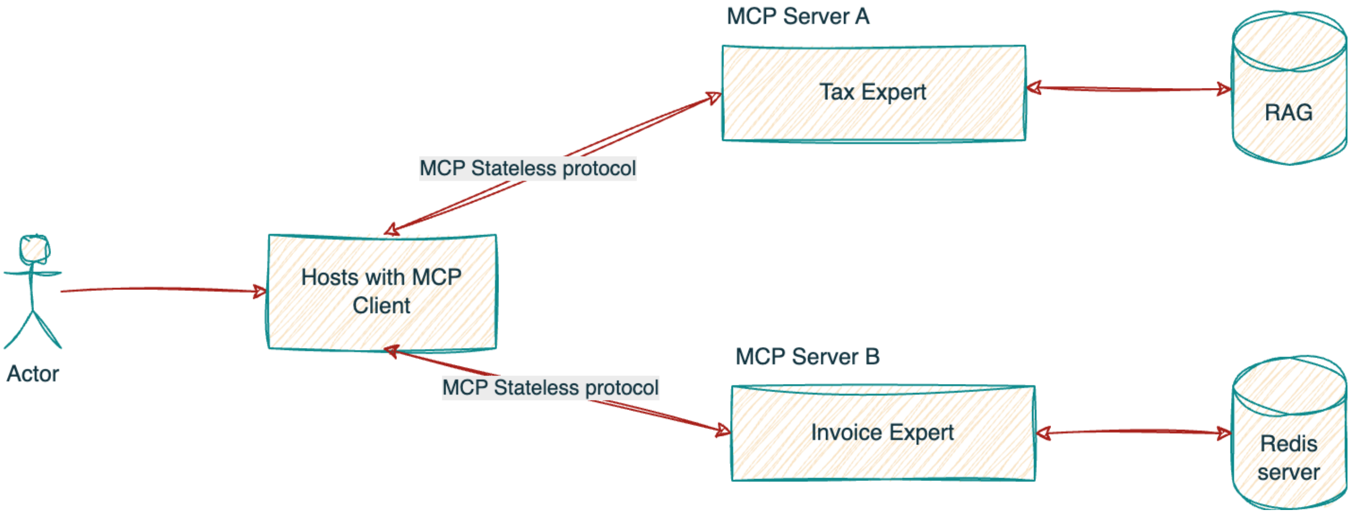
MCP is an open protocol that standardises how applications provide context to LLMs. This is a standardised way to connect AI models to different data sources and tools. LLMs frequently need to integrate with data and tools, and MCP provides:

- A growing list of pre-built integrations that LLM can directly plug into
- The flexibility to switch between LLM providers and vendors
- Best practices for securing data within customer infrastructure

For more details about MCP, please visit MCP Introduction.

In this blog, we will see a real world example, where more than 1 agent help to answer a query. Two specialized AI "experts" operate in concert and support customer tax queries.

Usecase: Agentic AI Swarm on MCP



There are 2 AI experts, one for tax queries and information and the other one offers Invoice related tools like GetDetails, GetMetadata and more.

- Tax Expert
  - Capability: Uses Oracle Cloud Infrastructure’s Retrieval-Augmented Generation (OCI RAG) agent with country specific taxation data, sitting atop a base LLM.
  - Function: Dynamically queries up-to-date tax codes, regulations, and precedent cases to compute tax liabilities, generate compliance checklists, and surface optimal filing strategies.
  - Integration: Exposes a simple tool interface backed by MCP Server hosted on OCI Data Science as HTTP endpoint, so that other agents can request tax calculations or regulatory excerpts on demand.
- Invoice Expert
  - Capability: Interfaces with a Redis-backed MCP Server tool, that maintains the organization’s invoice ledger and metadata.
  - Function: Retrieves, aggregates, and normalizes invoice data—such as vendor totals, due dates, and line-item details—enabling precise downstream analysis.
  - Integration: Publishes its outputs via MCP, making them immediately available to any other agent in the swarm.

Behind the scenes, OCI Cohere Manage Service orchestrates the “react” (ReAct) decision cycles—routing requests, managing state transitions, and ensuring each expert invokes the right tool at the right time.

Why Adopt MCP for Agentic AI?

1. **Modularity & Scalability**
   - Each expert lives in its own microservice, with clearly defined tool interfaces. Teams can independently update taxcode logic or invoice schema without impacting the other agent.
   - New experts (e.g., a “Credit Risk Expert” or “Audit Trail Expert”) can be onboarded rapidly by simply registering another MCP tool.
2. **Real-Time Collaboration**

    - The MCP bus glues disparate services into a cohesive workflow. An analyst’s “What’s my tax liability on unpaid June invoices?” query triggers both the Invoice Expert (to fetch data) and the Tax Expert (to compute liability) in a seamless handoff—delivering answers in seconds rather than hours.

3. **Maintainability & Governance**
    - Tool calls and data exchanges are logged and auditable at the MCP layer, supporting compliance and traceability requirements.
    - Role-based access controls can be enforced at each microservice boundary, ensuring sensitive financial data remains protected.
4. **Cost Efficiency**
    - By decoupling compute workloads, each service can autoscale independently on OCI, optimizing resource utilization and controlling costs.
    - Teams only pay for the exact mix of container or functionrun time they consume, rather than over provisioning a monolith.
5. **Rapid Innovation**
    - With MCP’s plug-and-play model, business units can pilot new AI capabilities—such as invoice anomaly detection or predictive cash-flow forecasting—by adding lightweight experts, without touching the core pipeline.

Enabling MCP Server Streamable HTTP with Model Deployment

At OCI, Model Deployments are a managed resource in Data Science service used to deploy machine learning models and extend to host MCP Servers as HTTP endpoints. Hosting in OCI Data Science enables AI applications and agents to securely access tools and data.

In the context of Model Deployment, MCP can be hosted as a middleware layer in the model serving pipeline. Clients can invoke HTTP inference endpoints wrapped in a MCP client. This client would negotiate session identifiers, propagate shared metadata, and optionally fetch historical context from a server. On the model server side, MCP handlers would parse and apply context during pre-processing, before forwarding the prompt to the LLM or fine-tuned model. Post-processing can also leverage context (e.g., user preferences) for formatting or filtering outputs.

With MCP embedded into Model Deployment, OCI can:

- Enable **stateful, dynamic interactions** with models.
- Improve **prompt relevance** and **output coherence**.
- Support **enterprise use cases** such as context-aware summarisation, contextual code generation, and session-based reasoning.

Conclusion

Adopting the MCP-based swarm architecture empowers finance and compliance teams to move from rigid, batch-oriented processes to a dynamic, real-time AI ecosystem. By decomposing domain logic into reusable expert agents—seamlessly coordinated by MCP and empowered by OCI’s RAG, Redis, and Cohere services—organizations unlock faster time-to-insight, stronger governance, and ongoing flexibility to evolve their AI capabilities as business needs change.

Hosting MCP servers

Checkout more samples of OCI Data Science