MCP in Action

Connecting AI to Enterprise Systems

Otavio R. Piske

About me

- Software Engineer - IBM
- Apache Camel
  - Committer
  - PMC
- ASF Member
- Wanaku Lead Developer

Agenda

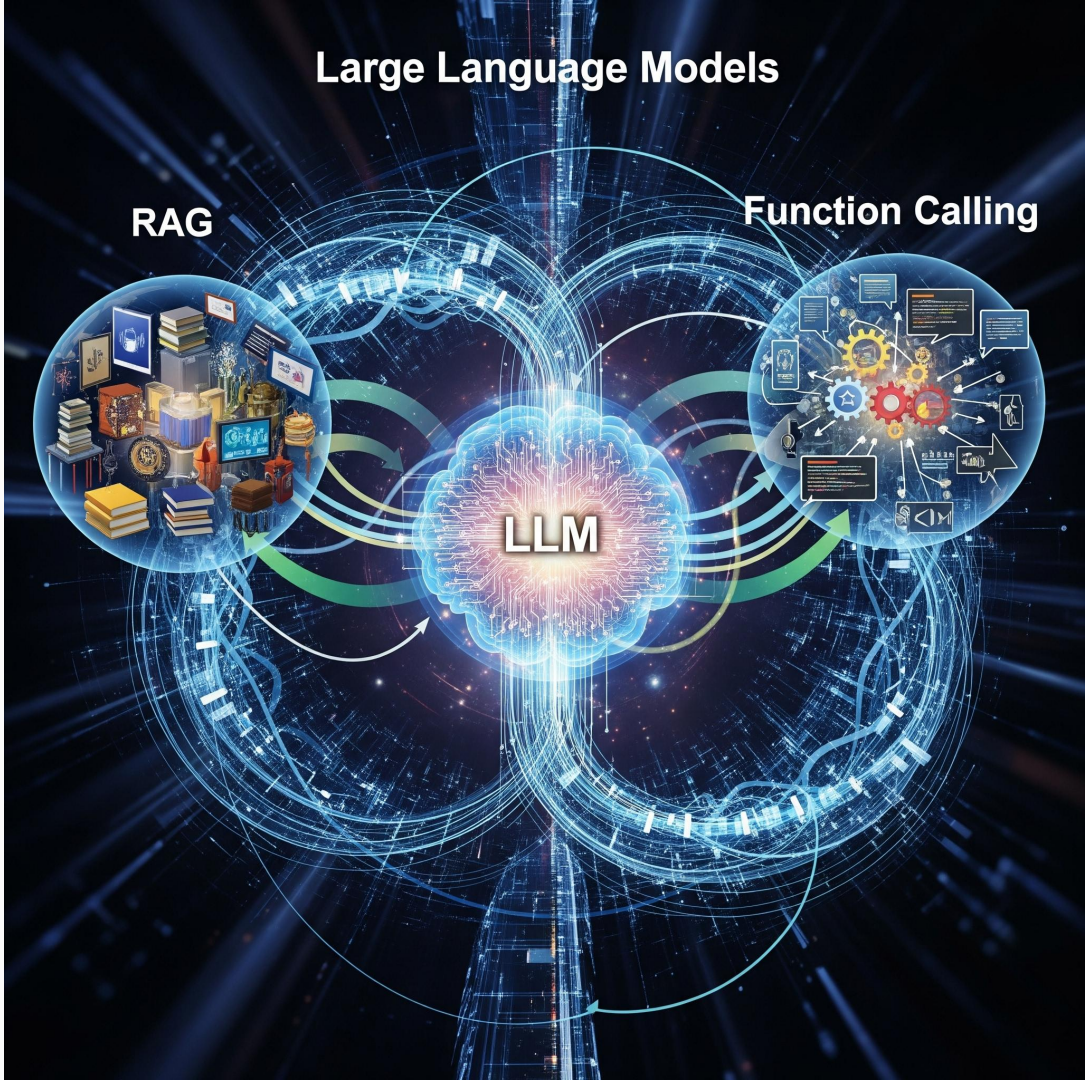
- Introduction to AI
- Integration Challenges
- Understanding the Model Context Protocol (MCP)
- Introducing Wanaku: An Open-Source MCP Router
- Demo
- Roadmap and Plans
- Q&A

From LLMs to Functions

LLMs have significantly evolved over time

RAG improved LLM context understanding

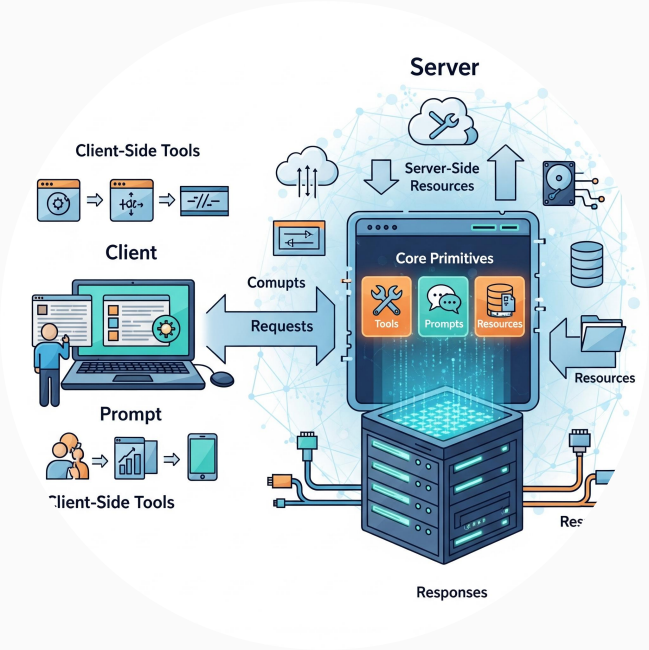
Function calling enhanced AI computational abilities



Introducing MCP

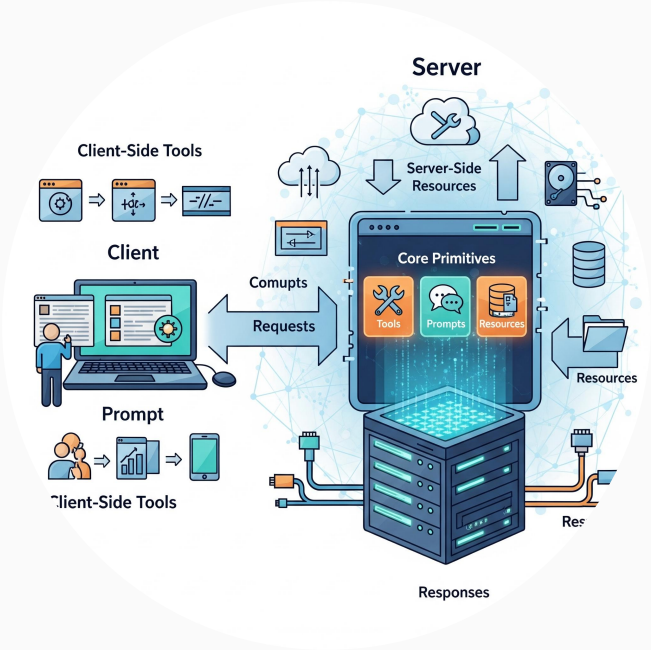
- MCP is an open standard
- It links AI systems to business tools and databases
- MCP overcomes AI models' relevance limits
- Visualizing MCP as a bridge clarifies its function

MCP Main Concepts



- MCP uses a client-server architecture.
- Client manages protocol communication and security.
- Server exposes resources and tools via MCP.
- Resources and tools are "core primitives"

MCP Core Primitives

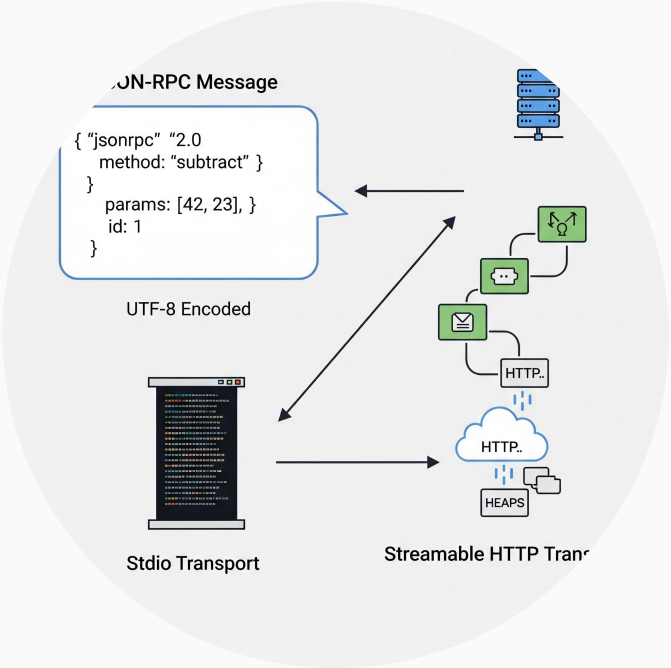


- Tools
  - Executable functions, computational
- Resources
  - Passive and read-only (files, logs, etc)
- Prompts
  - Pre-built instruction templates
- Sampling
  - Server-side requests for completions
- Elicitation
  - Server-side request for additional input

MCP Transport Mechanisms

01 — MCP uses JSON-RPC to encode messages reliably.

02 — STDIO for local usage.



03 — Streamable HTTP enables remote usage.

04 — Custom transports may be supported.

95% of AI Pilots Fail

- Most AI pilot projects encounter failure
- Custom solutions often stall due to complexity
- Integration issues hinder AI adoption
- Lack of fit with workflows is a key reason

AI Integration Hurdles

01 AI integration with enterprise systems is a major hurdle

02 Each new data source requires specific integration efforts

03 Securely connecting AI agents to existing business logic is critical

04 Legal and regulatory concerns may also impact integration

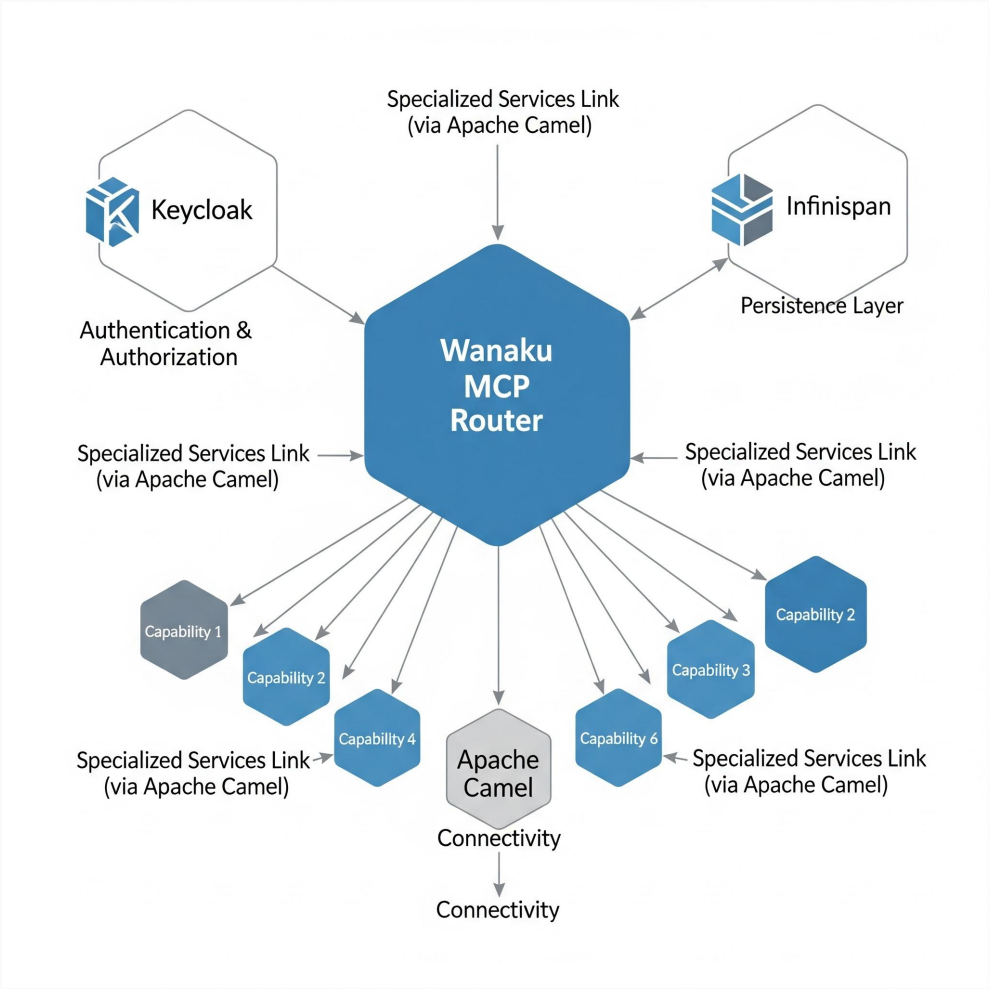
Introducing Wanaku MCP Router

- Wanaku is an open-source MCP router
  - Meet Wanaku at https://www.wanaku.ai/
- Developed by our team
- It provides unified access for AI agents
- Extensible

Wanaku: The Integration Hub

- Wanaku connects AI agents with external resources
- It manages and governs access between agents
- Wanaku relies on specialized connectivity services

Wanaku Architecture



- Wanaku's architecture comprises key components
  - Auth
    - Keycloak
  - Connectivity
    - Apache Camel
  - Persistence
    - Infinispan
- Powered by Quarkus
- Collection of microservices

Wanaku Namespaces Explained

01 Namespaces organize and isolate different resources effectively

02 They are a key feature for managing complexity in Wanaku

03 They provide logical separation for various tools and data

04 Namespaces ensure clarity and prevent potential conflicts

MCP Forwards Explained

01 The MCP Bridge feature facilitates seamless cross-server communication

02 Enables a Wanaku server to forward requests to other servers

03 Enhances connectivity options

Wanaku Security Features

01 —— Authentication and authorization: multiple OAuth standards support

02 —— Dynamic Client Registration

03 —— Wanaku integrates Keycloak for robust identity management

Other Wanaku Features

- Toolsets enable diverse functionality
- OpenAPI importer streamlines integration

Wanaku Capabilities Explained

- Wanaku is a blank MCP server
- Capabilities define what Wanaku can do
- They specify actions and functions
- They allow for diverse integrations

Camel Integration Capability

01 Apache Camel bridges systems and connects to AI

02 This capability links AI agents to diverse enterprise systems

03 Secure interactions between AI and business data are enabled

04 Camel offers over 300 components for extensive connectivity

Camel Integration Capability

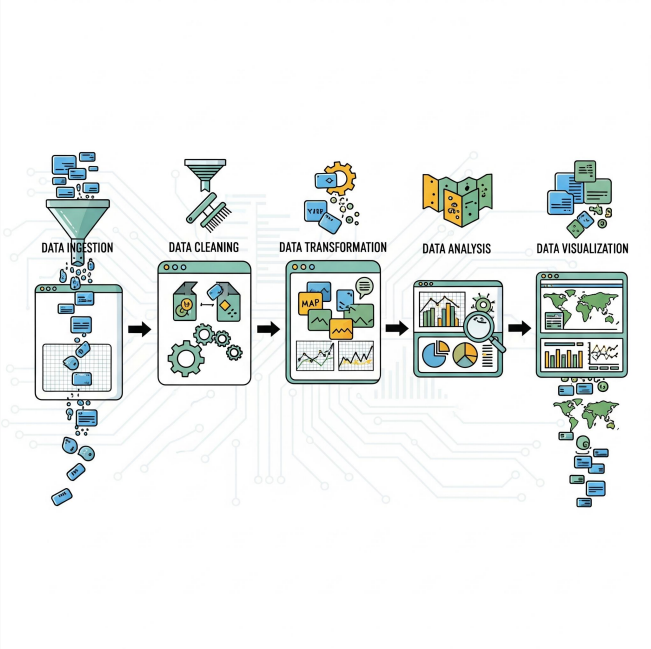
01 —— Apache Camel routes are exposed as MCP Tools or MCP resources

02 —— Rules define how Wanaku exposes them

03 —— Use visual editors to design and view the routes

Anatomy of a Camel Route

- A pipeline is a set of data processing elements.
- Data flows sequentially through each connected stage.
- Each stage performs a specific transformation or task.
- Pipelines organize complex processes efficiently.



Anatomy of a Camel Route: the code

```
- route:
    id: get-employee-reviews
    description: Retrieve employee performance reviews
    from:
      uri: direct:employee-reviews
      steps:
        - setHeader:
            constant: GET
            name: CamelHttpMethod
        - toD:
            uri:
http://employee-backend-service:8081/employee/${header.EMPLOYEE_ID}/r
eviews
            parameters:
              bridgeEndpoint: true
        - log:
            message: "Retrieved employee reviews for ID:
${header.EMPLOYEE_ID}"
        - convertBodyTo:
            id: convertBodyTo-2339
            type: String
```

What Are Rules?

- Rules are explicit instructions or guidelines for action.
- Provide necessary structure for exposing routes
- Determine the type of core primitive to use for the route

Anatomy of a Wanaku Camel Rule: the code

```
mcp:
  tools:
    - get-employee-information:
        route:
          id: "get-employee-information"
        description: "Fetches core profile data for a
specific employee (eg. name, id, level and days in
level)."
        properties:
          - name: employeeId
            type: int
            description: The employee ID to retrieve
information for
            required: true
            mapping:
              type: header
              name: EMPLOYEE_ID
```

Wanaku Demo: Camel Integration

01 Short demonstration of Wanaku in action

02 Highlighting the Camel integration capability

03 Wanaku demo will showcase the interface

Roadmap

- Operator
- Increase support for core MCP primitives
- Fine-grained access control
- Observability
- A2A

Building Wanaku: Our Experience

- Over attention in STDIO
- Lack of maturity on the ecosystem

Closing Thoughts

01 Consider integration challenges to ensure AI pilot success

02 Tools like Apache Camel can solve integration issues

03 Your feedback can help shape Wanaku's direction

04 Meet Wanaku at https://wanaku.ai

Feedback

01 —— Rate the talk

02 —— Share the feedback

03 —— Help us improve and deliver better content for you

Questions?

01 —— Open the floor for questions

02 —— Engage in discussion and clarify

03 —— We are ready for your questions