

Enterprise Agentic Framework

A unified approach to building intelligent agents for enterprise
applications

Core agent

From basic to autonomous agent architectures



Building blocks

Each type builds upon previous capabilities



Intelligence spectrum

Framework supports various agent intelligence levels



Decision making

Enhanced functionality for complex problem solving



Core capabilities

Fundamental features shared across agent types



Learning autonomy

Advanced agents capable of independent operation

Overview

Framework agnostic development with unified grammar

A consistent annotation and programming model that works across multiple agent frameworks.

 Ensures enterprise

 Automates orchestration

 Works across multiple

 Abstracts framework



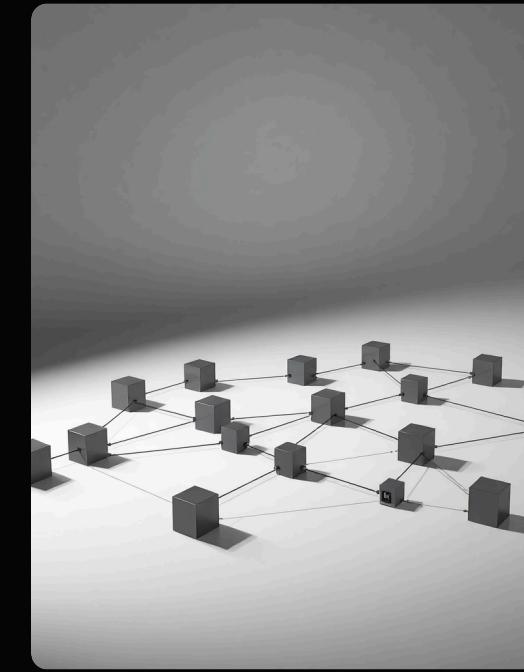
Hexagonal architecture with modular components

The framework uses a flexible hexagonal structure with sidecars that separates core agent logic from tools and services. This design enables clean integration points and maintains separation of concerns for enterprise-grade implementations.

Through this architectural approach, organizations can achieve greater modularity and extensibility while ensuring that core business logic remains isolated from external dependencies and implementation details.

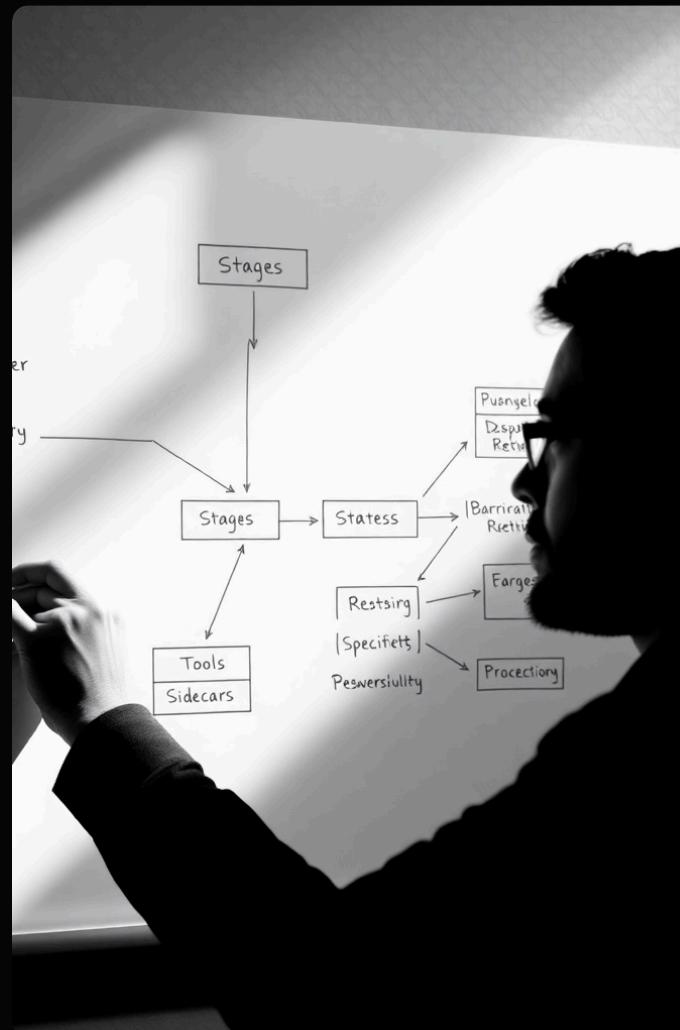
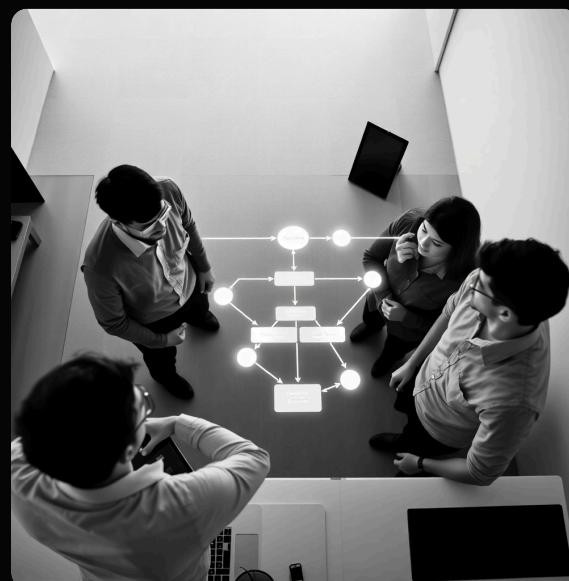


Cloud native infrastructure for agent deployment



Built for containerization with individual tasks as containers, enabling scalable computing with CPU/GPU choice per task.

Provides a clear programming model with stages and tools.



Streamlined development workflow

Six step process from defining agent grammar to deployment using Helm charts for comprehensive agent implementation.



Write application logic

Develop the core functionality that powers your agent's capabilities and responses.



Configure service sidecars

Set up supporting services that run alongside your main application for extended functionality.



Define agent grammar

Establish the foundational rules and structure for your agent's communication and functionality.



Test implementation

Validate the agent's performance and functionality against requirements before deployment.



Deploy with Helm

Use Helm charts to efficiently deploy your complete agent implementation to production environments.



Set up persistence

Implement mechanisms for storing and retrieving data across agent sessions and interactions.

Integration

Enterprise integration options for services

Multiple communication protocols and platform support for enterprise deployment.



Event sourcing captures all changes as domain events.



gRPC protocol supports efficient binary communication between services.



Kubernetes operators simplify deployment and management in cloud environments.



Agent Streams provide MQTT messaging for IoT connectivity.



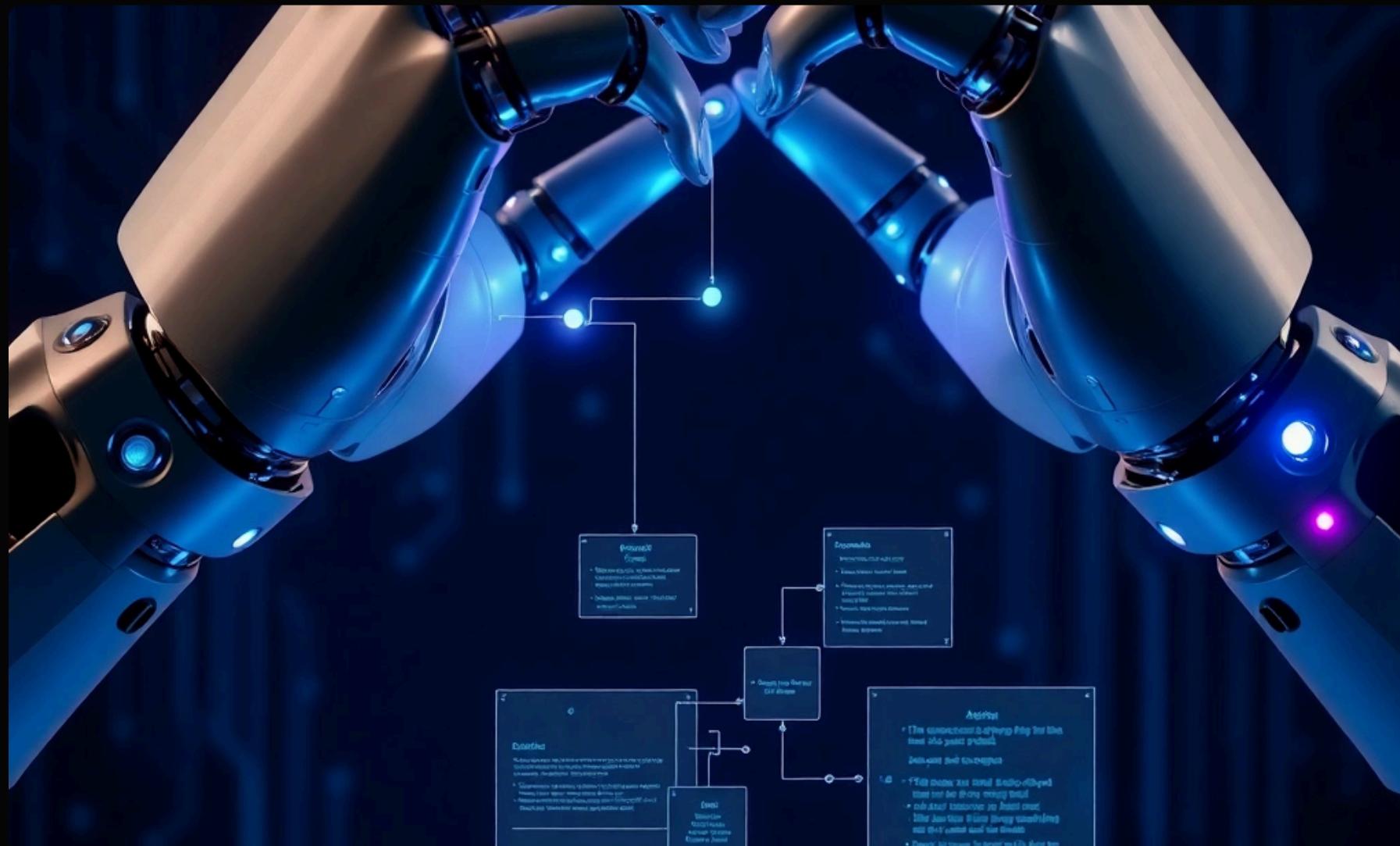
Database abstraction facilitates persistence and state management capabilities.



REST APIs enable standard HTTP-based service communication.

Multi agent collaboration capabilities

Enables intelligent collaboration through service port communication via hexagonal architecture.



Technical and contextual agent types



Distributed decision-making for complex problems



Supports context-aware agent interactions

Business advantages

Offers consistency through uniform grammar across all frameworks



Independent containerization

Scalable architecture through task isolation



Consistent grammar

Uniform structure across all frameworks



Reduced complexity

Increased productivity in development workflows



Technology agnostic

Flexible design for diverse implementation needs



Cloud-native support

Portability with Kubernetes infrastructure

Demo and implementation roadmap

Live demonstration showing agent creation, communication and task orchestration.

