The Global Understanding Environment

Enabling Proactive Industrial Devices with Mobile Agents and the Semantic Web

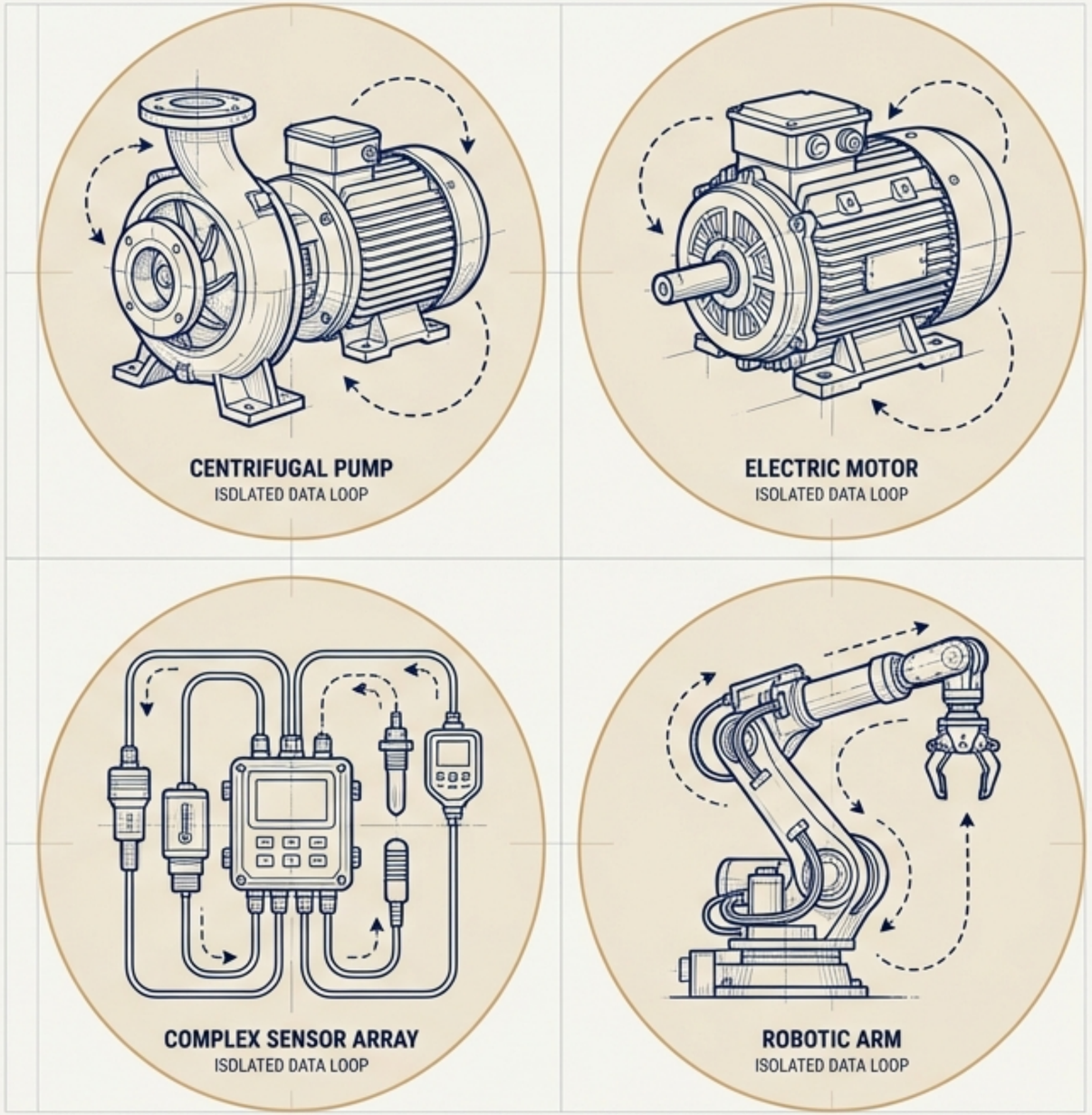
MOBILE AGENTS | SEMANTIC WEB | WEB SERVICES | FIELD DEVICES | CONDITION MONITORING | E-MAINTENANCE

Our Devices are Smart, But They Don't Cooperate

The industrial landscape is filled with smart devices, but they exist in isolated ecosystems.

This heterogeneity leads to critical problems:

- **Data Silos:** Valuable information on device performance and maintenance history is trapped within proprietary systems.
- **Reactive Maintenance:** Without a holistic view, maintenance is often reactive, performed only after a failure, rather than proactively predicted.
- **Integration Bottlenecks:** Connecting diverse systems requires custom, brittle solutions, stifling on-the-fly collaboration.

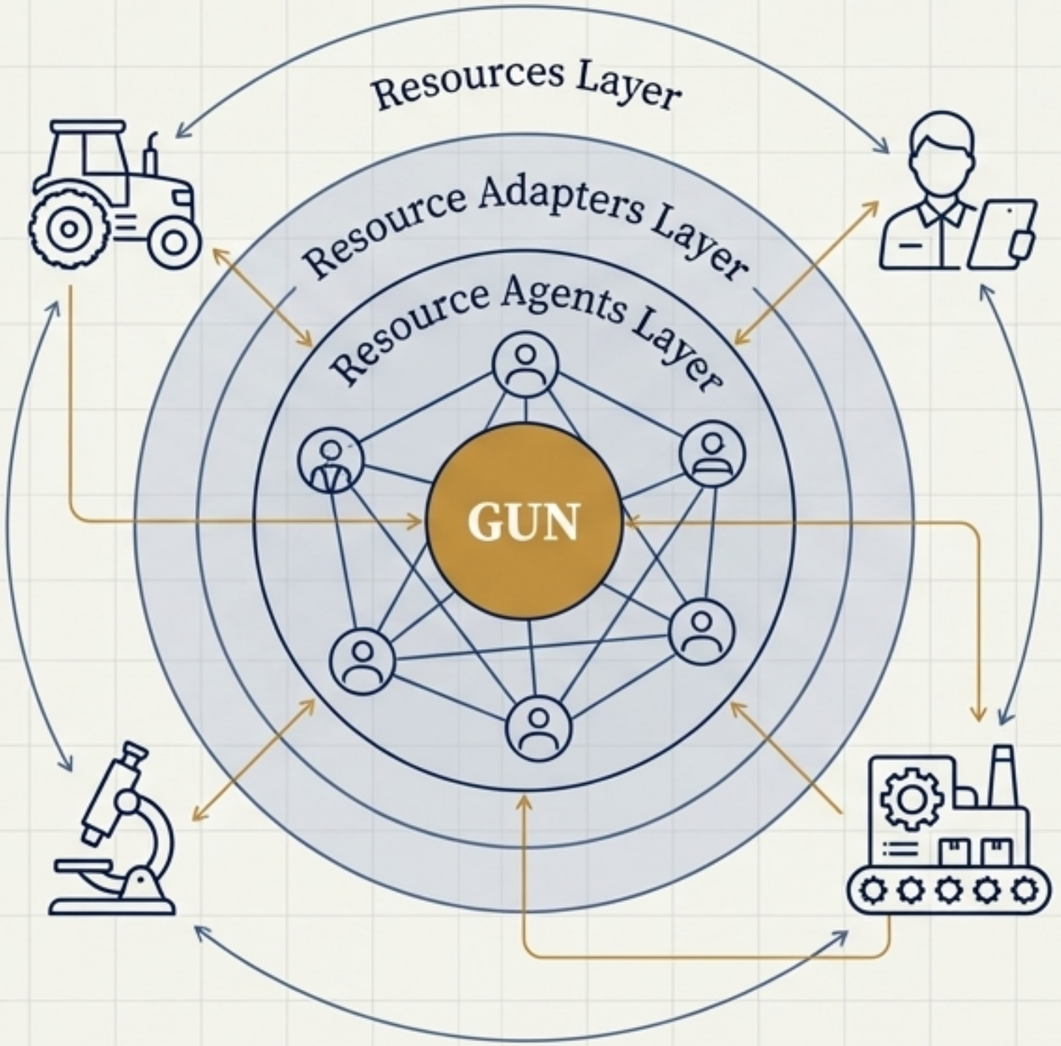


The Vision: A Proactive “Welfare” Ecosystem for Industrial Resources

We propose the Global Understanding Environment (GUN): a web-based platform that provides a global system for automated “care” over industrial resources.

Within GUN, resources are proactive and cooperative. They can:

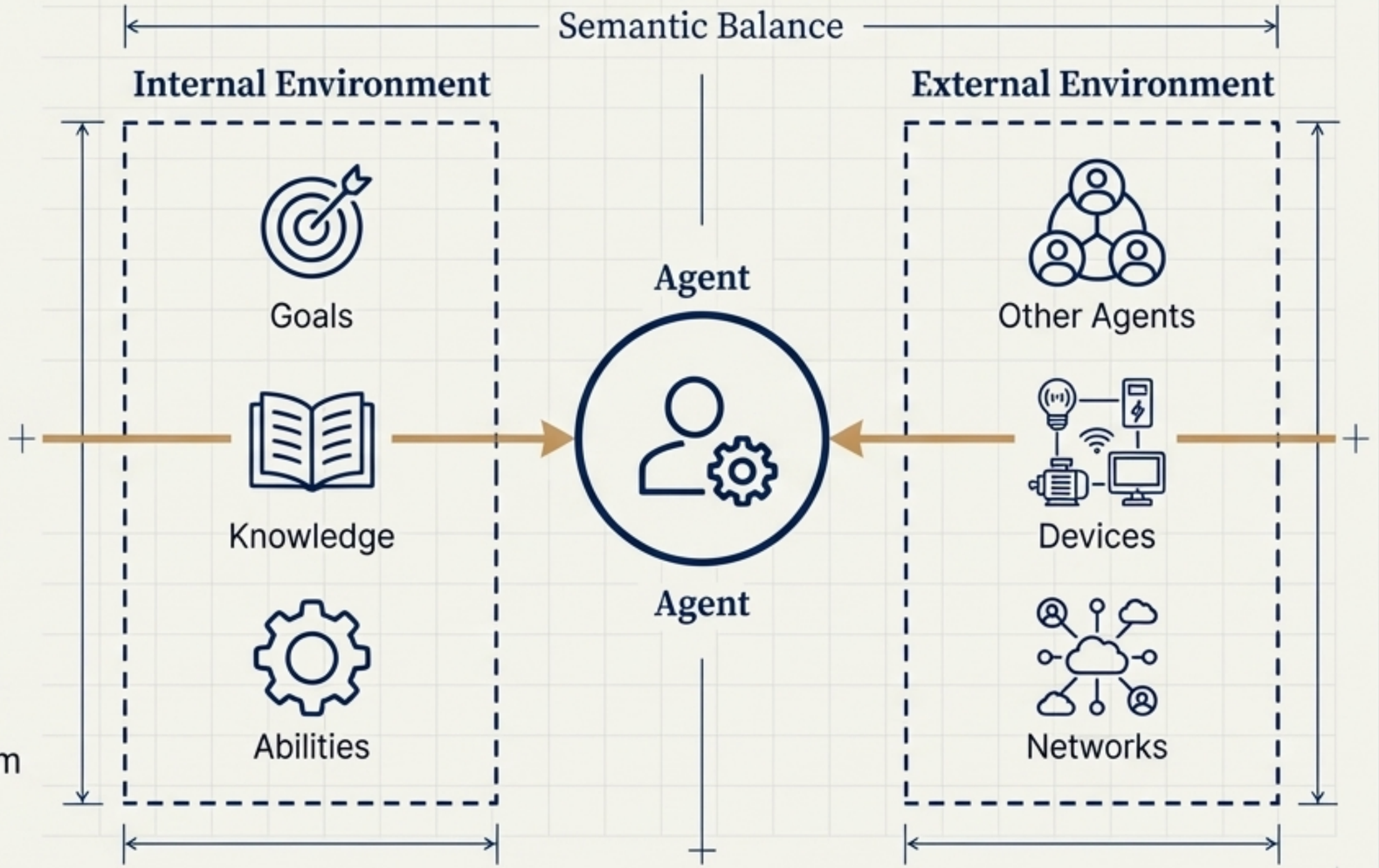
- Automatically monitor their own state and context.
- Discover and request remote diagnostics from decision-making services.
- Independently order necessary maintenance from service providers.



The Agent: The Heart of Proactivity is ‘Semantic Balance’

An Intelligent Agent is an entity driven to maintain a continuous balance between its internal environment (goals, knowledge, abilities) and its **external environment. This core drive makes an agent:**

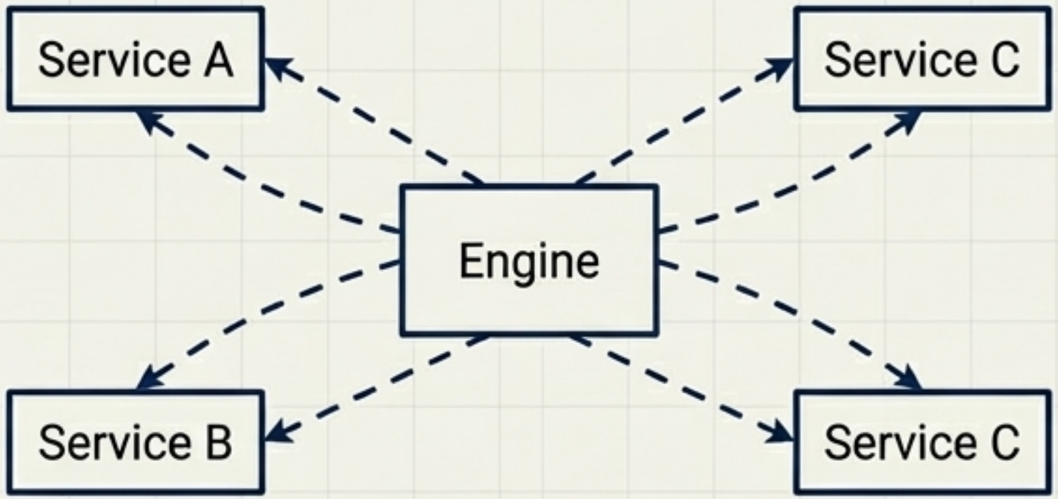
- **Goal-Oriented:** Its primary goal is to maintain balance.
- **Adaptive:** It can change its internal state.
- **Creative:** It can change its external environment.
- **Mobile:** It can move to a new environment where balance exists.
- **Social:** It can communicate and form communities to achieve balance.



Agent Mobility: Moving the Intelligence to the Data

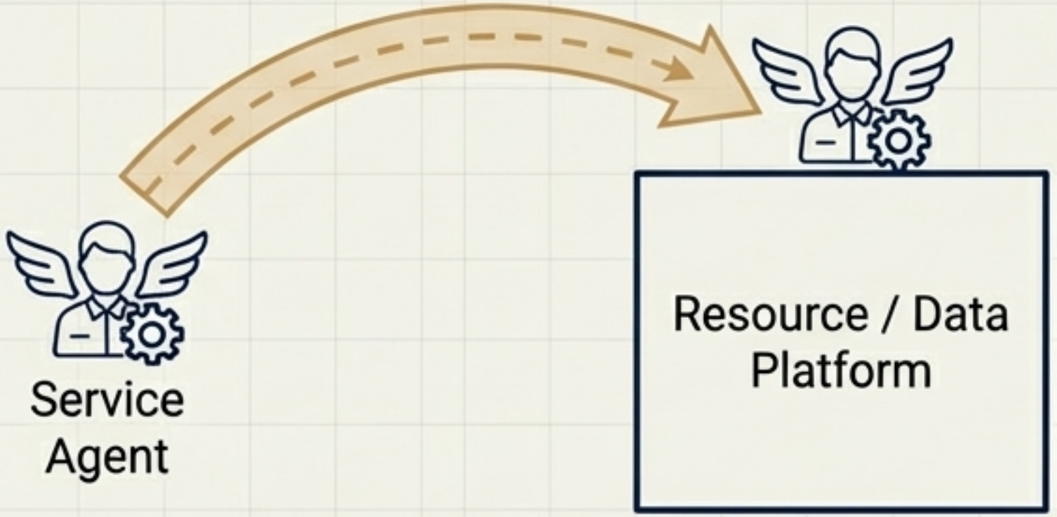
GUN is built on a 'Mobile Components' architecture. Instead of a central engine coordinating distributed services remotely, the services themselves—encapsulated as mobile, self-interested agents—travel to the resources.

Traditional: Mobile Engine



A central engine coordinates activities remotely, leading to network latency.

GUN: Mobile Components



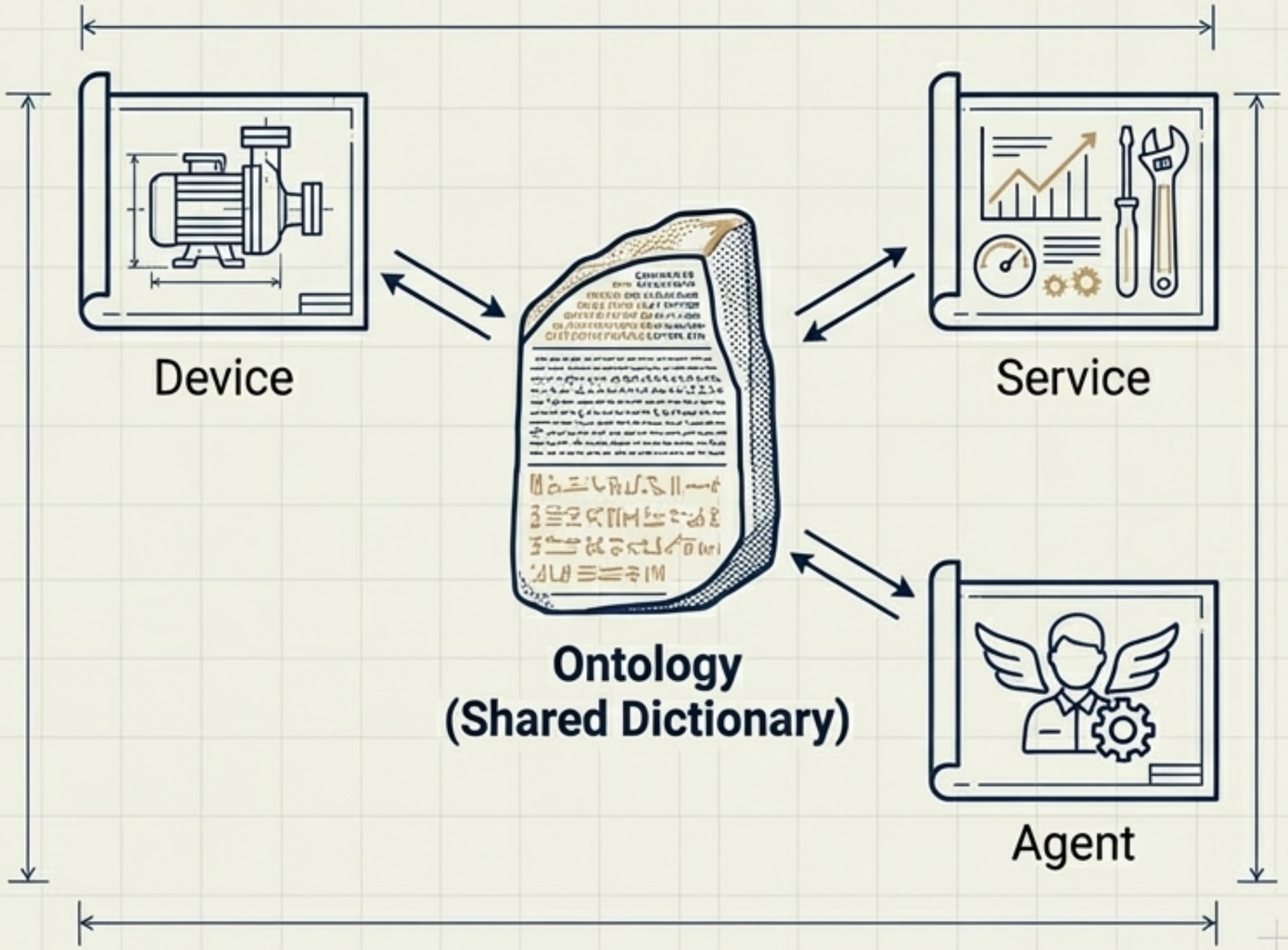
The agent (the service component) moves to the data source to perform its task locally, reducing latency and enabling autonomous operation.

The Semantic Web: A Universal Language for Machines

For agents and devices to cooperate, they need a shared understanding of information. The Semantic Web provides this through ontologies.

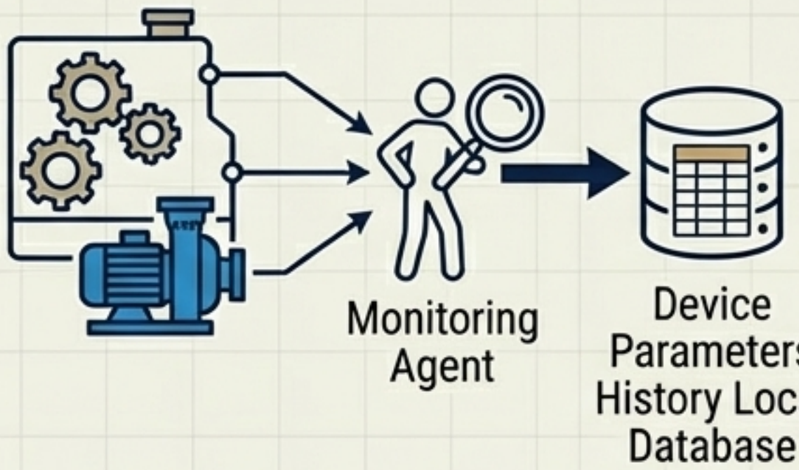
What is an ontology? It's a formal "shared dictionary" that defines concepts and the relationships between them in a specific domain (e.g., device maintenance).

Why is it critical? Ontologies provide a machine-understandable language (like DAML-S/OWL-S) that allows diverse components from different manufacturers to achieve true interoperability, automating service discovery, composition, and use.



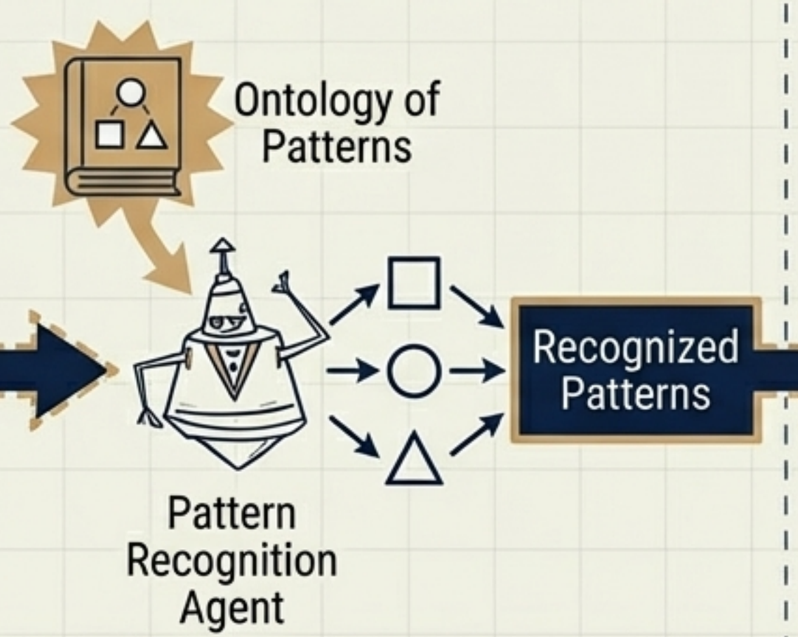
In Action: An Agent-Based Diagnostic Workflow

Step 1: Monitor & Collect



A “Monitoring Agent” tracks device parameters from multiple sensors, creating a local history.

Step 2: Recognize Symptoms



A “Pattern Recognition Agent” analyzes this data, using an “Ontology of Patterns” to identify specific, recognized “symptoms” of malfunction.

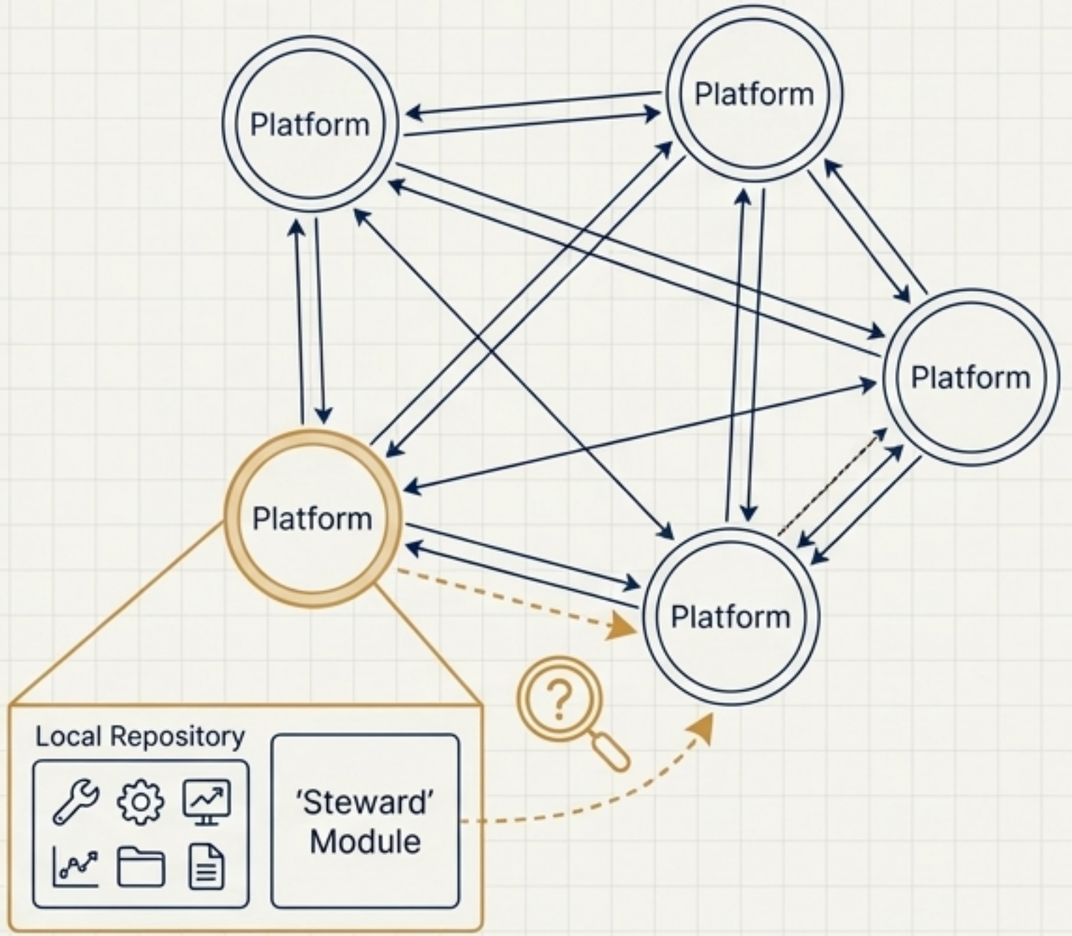
Step 3: Diagnose Disease

A “Field Agent” aggregates these recognized symptoms. It then consults an “Ontology of Diseases” to determine the final diagnosis.

The Network Layer: OntoServ.Net

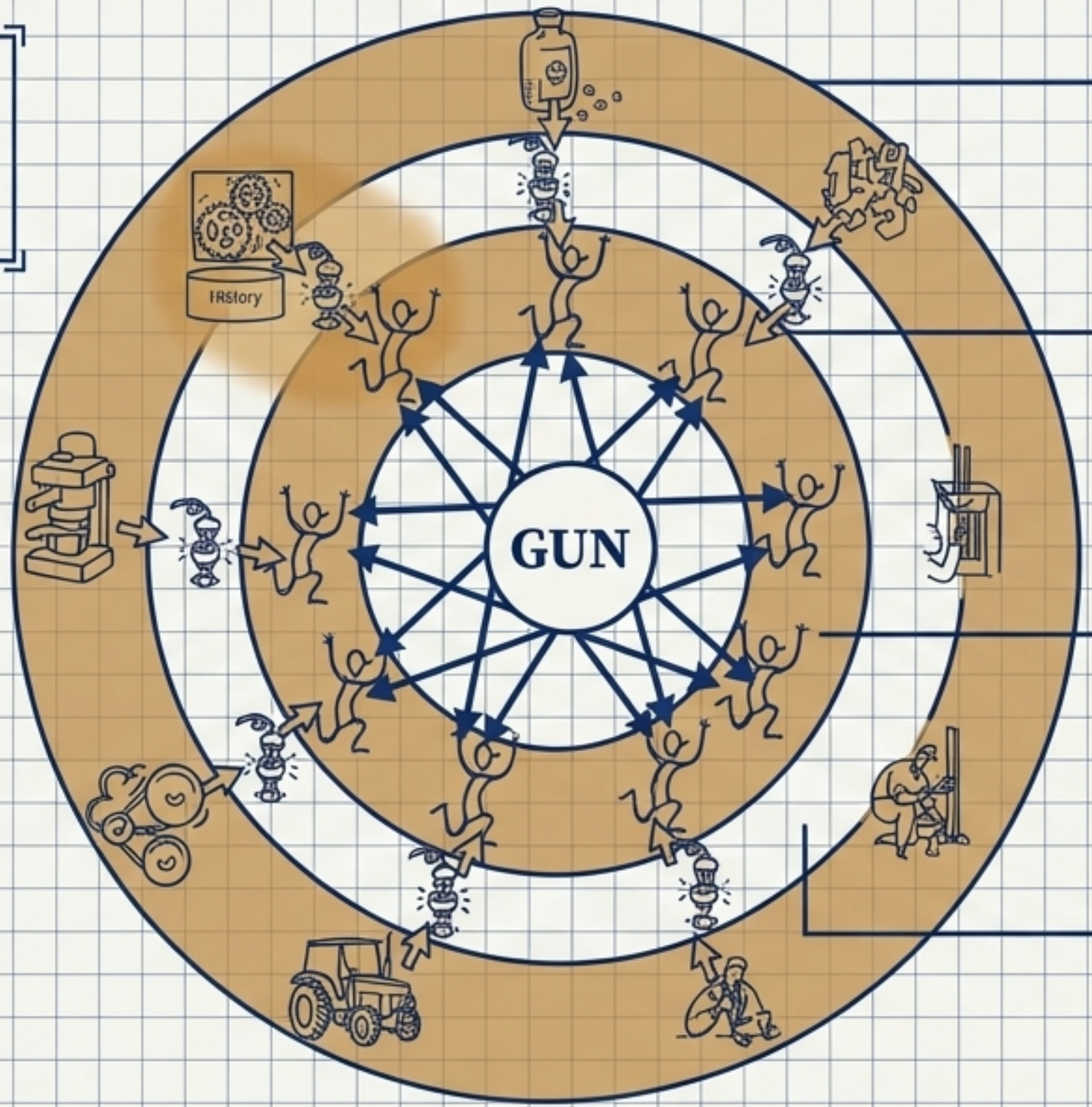
OntoServ.Net is the implementation concept for a large-scale, automated industrial maintenance environment. It is a peer-to-peer network where:

- **Services are Standardized**: Maintenance and diagnostic services are described using a common ontology.
- **Discovery is Semantic**: A ‘Platform Steward’ module on each node uses a semantic search engine to find the right service.
- **Knowledge is Shared**: Locally improved maintenance experience can be shared across the network.



Deconstructing the GUN Blueprint

A ‘GUN Resource’ is more than a device; it is a composite entity fully integrated into the environment.



Resources Layer:

The physical or digital assets (devices, experts, software).

Resource Adapters Layer:

The hardware and software bridge connecting the resource to GUN and annotating its data with semantics.

Resource Agents Layer:

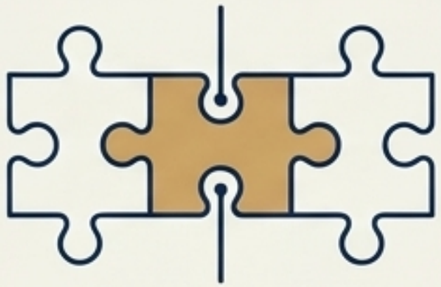
The proactive software agents assigned to each resource, responsible for monitoring and decision-making.

Resource Platform:

The computing environment where the agents live and execute.

An Implementation Roadmap: Three Core Frameworks

The realization of GUN is structured around three integrated research and development frameworks, each addressing a fundamental requirement:

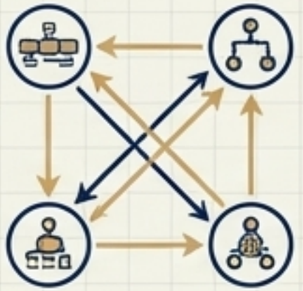


GAF: General Adaptation Framework
(for Interoperability)

Provides the tools and ‘Semantic Adapters’ to connect diverse resources to GUN and describe their state (RscDF).

GPF: General Proactivity Framework
(for Automation)

Enables resources to be proactive by formally describing their goals and individual behaviors (RgbDF).



GNF: General Networking Framework
(for Integration)

Defines how the behaviors of individual proactive resources are integrated into larger, coordinated business processes (RpiDF).

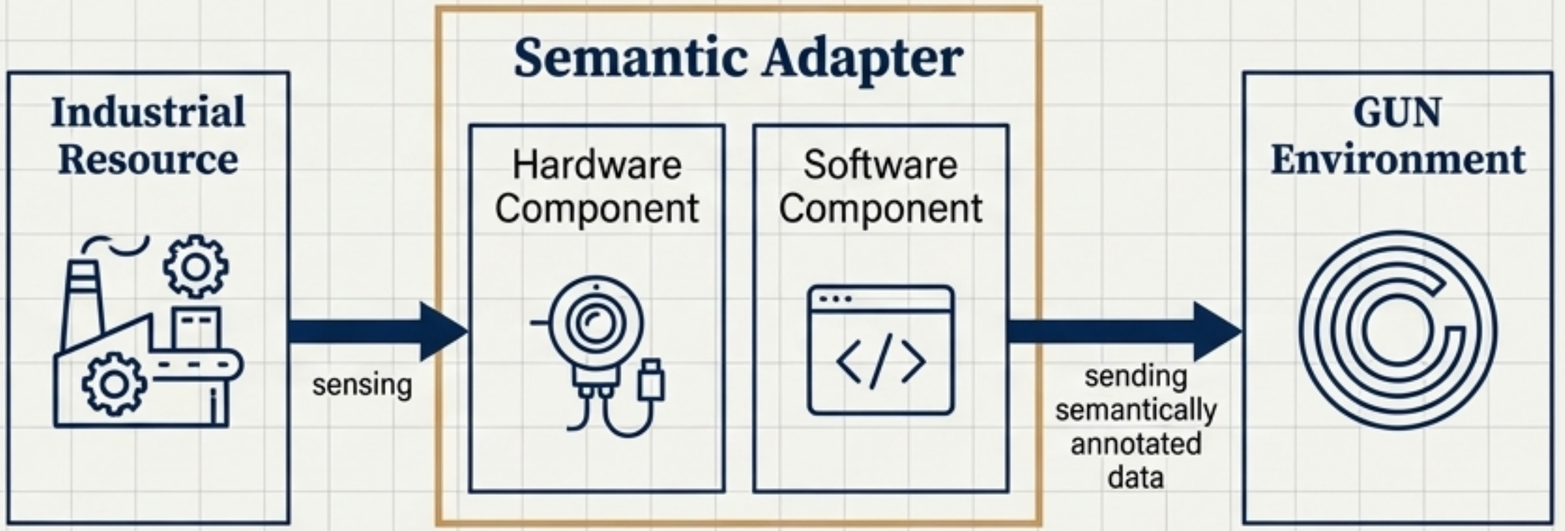
Challenge 1: The Adapter Problem

How do we semantically connect the physical world?

The greatest challenge is bridging extremely heterogeneous real-world resources to the GUN environment. This requires more than just software.

The goal is to create a set of standardized, reusable components (hardware sensors, software annotators) and a methodology for automatically designing a "Semantic Adapter" for any given resource based on its description.

This is the focus of the General Adaptation Framework (GAF).

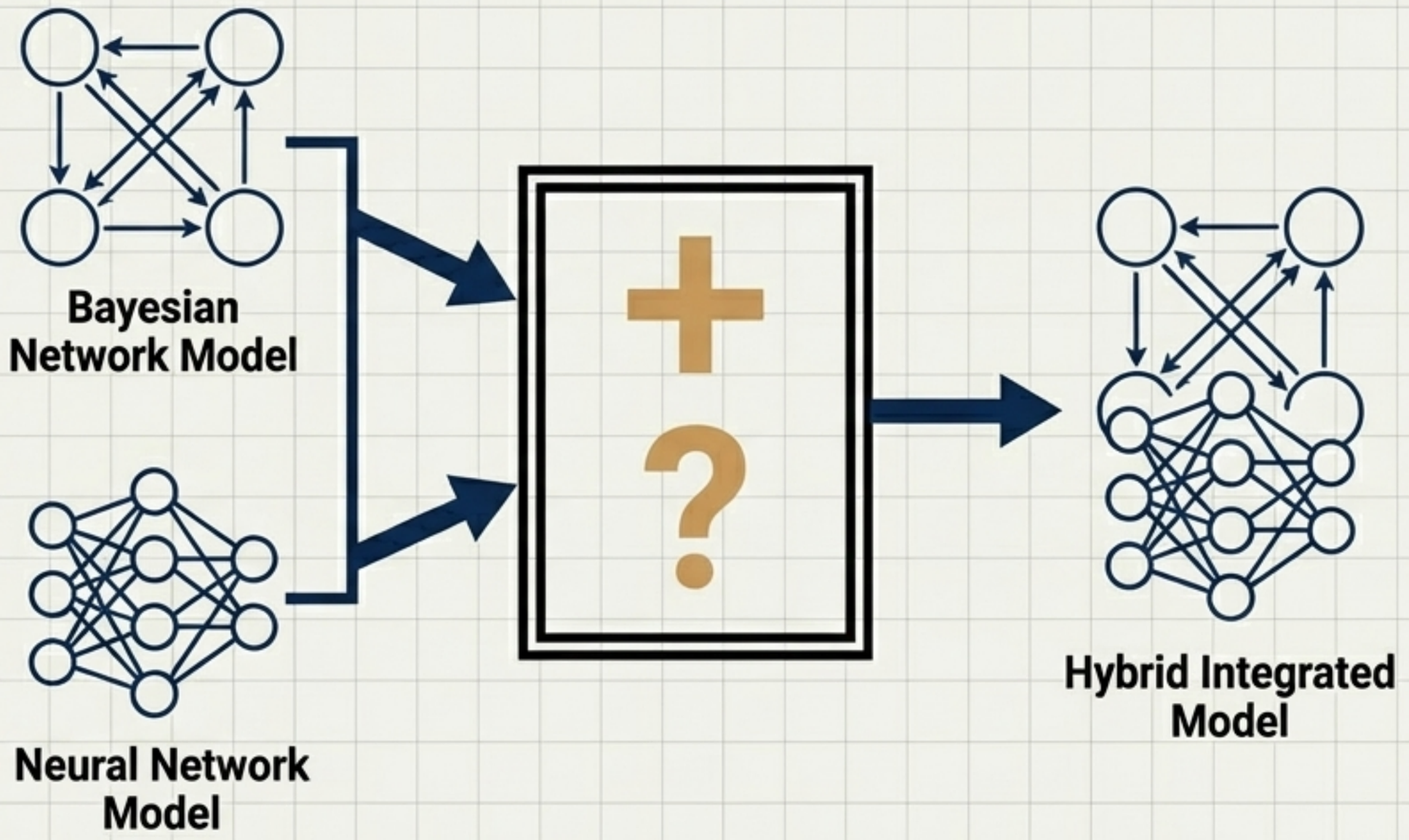


Challenge 2: Portable Intelligence

How can diagnostic models be exchanged and combined?

Diagnostic services use various machine learning models (e.g., Bayesian Networks, Neural Networks). A key challenge is enabling these models and their reasoning engines to be shared, reused, and even composed.

- **The Problem:** A Bayesian model learned by one service is stored in a proprietary format and is unintelligible to another.
- **The Goal:** Develop a standard semantic markup and ontology for describing the models themselves. This would allow two different models to be integrated, or even a hybrid model to be created and executed.



Challenge 3: The Currency of Trust

In a decentralized network, who do you trust?

The challenge of “Learning Distributed Trust” involves combining three key strategies. A complete trust evaluation is a smart combination of all three.

1. Personal Experience

2. Certification

3. Peer-to-Peer Reputation

Challenge 4: The Human as a Web Service

What if human expertise was just another node on the network?

The GUN framework adapts all categories of industrial resources, including human experts. This leads to a paradigm-shifting possibility: a smart device could be a service requestor, and a human expert could be the service provider.

The challenge is to create a system that can automatically discover, invoke, integrate, and compose online human resources reliably within an automated workflow, just like any other software service.

Towards a Collaborative and Proactive Industrial Future

Today’s smart industrial devices are powerful but isolated. The Global Understanding Environment offers a new paradigm: a proactive, cooperative ecosystem built on mobile intelligent agents and the common language of the Semantic Web. By tackling the grand challenges of adaptation, model portability, distributed trust, and human-in-the-loop integration, we can build a truly intelligent industrial web that moves from reactive repair to predictive care.