



The Knowledge Layer

The foundation enterprise AI needs to work.



Contents

01 Introduction

02 Overview

Challenges Addressed	4
Our Solution	5
Unified Knowledge Graph	5
Ontology	5
Aluna: Our AI Assistant	7
Smart Dashboards	7
Built-In Explainability	8
Reusable Logic Available	8

03 Architecture & Components

Data Sources	10
Data Virtualization Layer	10
Knowledge Graph Layer	11
Action Layer	11
Orchestration Layer	12
Use Case Layer	13

04 Extensibility

14

05 Deployment Options

15

06 Differentiators

16

07 Summary

17



Introduction

d.AP transforms fragmented enterprise knowledge – facts, relationships, rules, context – into a unified, machine-readable knowledge layer that any AI system can query as a service. It bridges the gap between your data infrastructure and the AI applications that need to understand your business.

It enables:

- A machine-readable representation of your enterprise knowledge
- A shared semantic foundation across all systems and AI applications
- A queryable knowledge service for agents, copilots, and any AI system

It is doing all of this to enable faster and smarter data-driven decisions through a unified understanding of your business landscape.





Overview

Challenges Addressed

In many organizations, data is fragmented across disconnected systems (CRM, ERP, support tools, analytics platforms etc.) making it difficult to get answers depended on information from those siloed systems. Teams struggle to align on metrics, answer important questions quickly or reuse past insights.

Key challenges the platform solves:

- **Siloed Data Sources:**
Business data lives in dozens of applications. Without a common structure, linking various classes of data across systems becomes a manual effort.
- **AI without Grounding:**
AI systems hallucinate or deliver inconsistent results because enterprise knowledge exists nowhere in a machine-readable form.
- **Knowledge locked in people's heads:**
Critical business rules, exceptions, and context live in the minds of experienced employees - invisible to any AI system.
- **Siloed AI agents:**
Every new AI use case rebuilds knowledge from scratch. The SAP agent understands SAP, the Salesforce Agent understands CRM. None can speak to each other with a shared understanding.
- **Proprietary Knowledge Lock-in:**
Vendor-specific semantic layers encode business knowledge in proprietary formats, turning your knowledge into a platform dependency.



Our Solution

To overcome the limitations of fragmented data landscapes and missing contextual information, our platform combines a knowledge modeling workbench, a semantic knowledge graph, and a knowledge API that any AI system can query, completed by optional built-in interfaces for direct exploration. Together, these components bridge technical and business worlds, allowing users & AI across the organization to access, understand, and act on data.

Unified Knowledge Graph

At the core of the solution is an ontology-grounded knowledge graph that integrates data from various source systems (CRM, ERP, analytics tools, etc.) using a federated architecture. This graph models your domain using a federated ontology that defines business concepts along with their relationships. It ensures that data is both structurally consistent and meaningfully connected.

By organizing data around ontologies, the graph abstracts away the complexity of source systems and enables users & AI to explore their business through a conceptual lens not database tables.

Ontology

Ontologies define the shared vocabulary of your business on what entities exist (e.g., “Customer”, “Subscription”), how they relate (“has contract”, “placed order”), and what constraints apply (“a contract must have an end date”).

- **Alignment with business logic:**
Ontologies encode how your organization talks, thinks and acts.
- **Consistency across teams:**
Every user, tool, or system refers to the same definitions.
- **Reusability of knowledge:**
Once defined, semantic rules and relationships can be reused across use cases and departments.
- **Readable both for humans and machines:**
Overcome the separation of documentation for humans and machines, making knowledge directly actionable for digital processes

The platform comes with a set of common upper ontologies reusing well-established definitions as well as supporting ontologies that introduce shared concepts. Users can then define and evolve domain or use-case specific ontologies easily, while leveraging a solid foundation. Domain-specific extensions are supported without breaking the core structure, ensuring long-term flexibility.

By building on open standards like RDF/OWL, we ensure your knowledge model remains interoperable, reusable, and fully portable. This prevents vendor lock-in and guarantees that your modeled business knowledge remains a valuable, independent company asset.



A well-structured ontology transforms isolated data into machine-readable knowledge, creating the trusted foundation for any AI to reason upon.



Aluna - Your AI Agent for Enterprise Knowledge

Aluna is the core AI agent of the platform, built for deep ontology comprehension. It serves two key roles: It acts as the grounding service for other AI systems (e.g., copilots, agents) via open protocols like MCP/A2A, providing them with reliable context. It also allows users to interact directly with the knowledge graph using natural language:

- *"Which customers are at risk of churn this quarter?"*
- *"What's the month-over-month growth for subscriptions in EMEA?"*
- *"List all high-priority tickets for enterprise clients."*

The AI Agent translates these questions into formal graph queries, executes them in real time, and returns results in a variety of forms: tables, charts, entity graphs or textual summaries.

What sets it apart:

- **Semantic understanding:**
It understands business concepts from the ontology, not just keywords.
- **Explainable answers:**
Every result includes context like data source, logic applied, filters used.
- **Refinement by conversation:**
Follow-up questions let users filter, group, or drill into results interactively.

Reusable Logic

Insights delivered by Aluna can be instantly turned into dashboard widgets. Unlike traditional BI dashboards, these are a direct reflection of the underlying knowledge graph and are therefore:

- **Linked to the logic behind them:**
So, you know exactly what each KPI represents and where the data comes from.
- **Context-aware:**
They inherit the semantic structure of the graph, enabling deeper interaction and navigation.
- **Reusable and composable:**
Any widget can be reused across views, copied, adjusted, or shared without redoing the underlying work.

Everything updates dynamically, providing a transparent, real-time view into the knowledge graph. This allows you to visually validate the exact logic and context that Aluna provides to other AI agents.



The Agent Endpoint: Knowledge as a Service

This is the platform's core delivery mechanism. The entire trusted knowledge layer is exposed as a service, allowing any system to consume curated knowledge in the appropriate format:

- AI Agents & Copilots connect via open protocols like MCP and A2A to receive the semantic context and grounding needed for reliable, hallucination-free performance.
- Applications & Services access knowledge through standard REST APIs, ensuring consistent business logic across your software landscape.
- Analytics & BI Platforms query aggregated Data Products (e.g., as Iceberg tables), ensuring analyses are always based on the same trusted definitions.

This guarantees that the same, centrally-managed business knowledge is reused everywhere, ensuring consistency across all systems—from AI agents to BI reports.

Built-In Explainability

A core design principle is transparency. Every answer Aluna provides—whether to a user or another AI agent—is fully explainable.

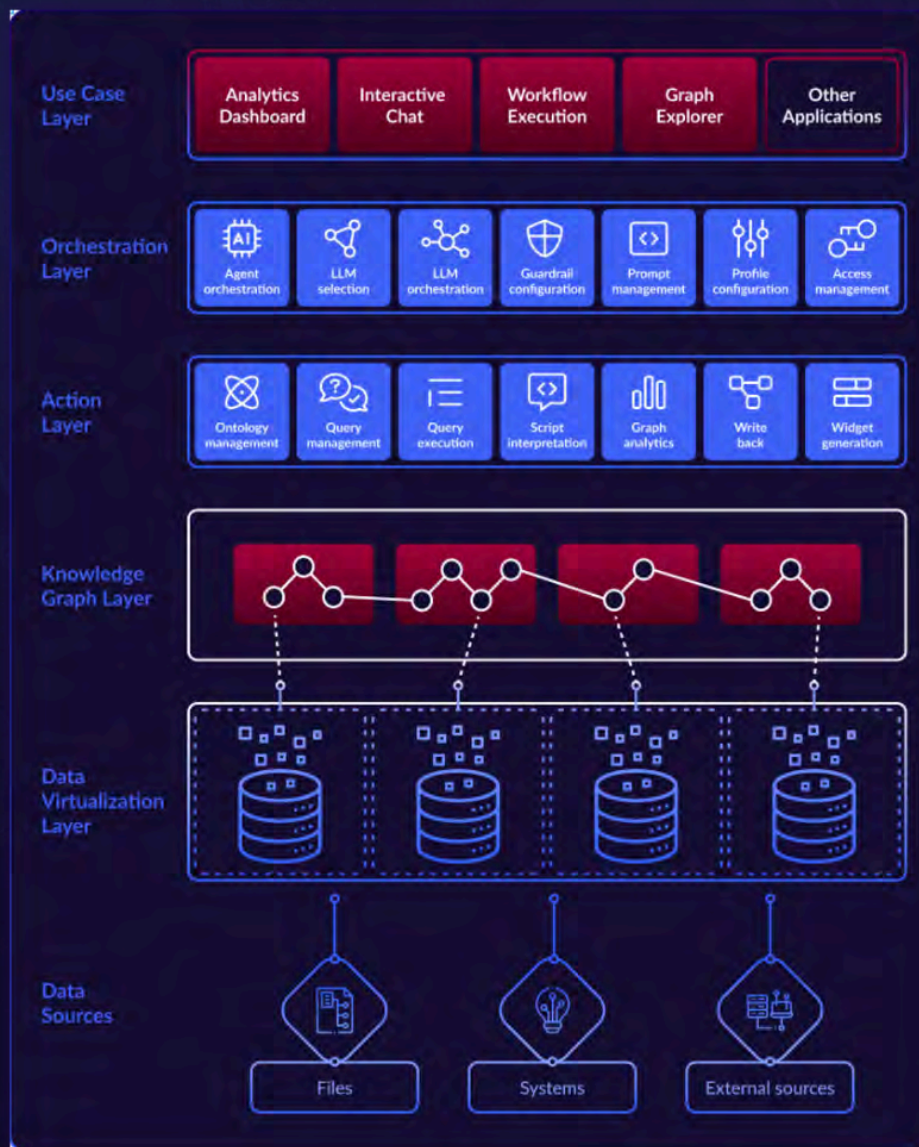
Users can inspect:

- Where did this data come from?
- What filters, joins, or aggregations were applied?
- How did Aluna arrive at this conclusion?

This builds trust for auditing and compliance, and enables both humans and AI agents to operate on a shared, unambiguous understanding of the business.

Architecture & Components

The platform architecture is modular, layered, and designed for extensibility, enabling seamless integration, semantic reasoning and natural language interaction over distributed data environments.





03 Architecture & Components

Data Sources

At the foundation are your existing data assets: files, systems, and external sources. These remain in place and can be linked into the graph in flexible ways via the Data Virtualization Layer.

Data Virtualization Layer

Federators establish the pipeline to your data sources and can be configured flexibly to meet the exact requirements of each source to be integrated. We never make investments you made obsolete - d.AP extends on the existing infrastructure investments, taking you from where your organizations stands in the journey of becoming data-driven and AI-enabled.

Three core capabilities:

- **Source connectivity:** Built-in connectors for all common standard technologies (REST, JDBC, SOAP, Iceberg, JSON/CSV flat files).
- **Transformation logic:** Python jobs can be included in the configuration to enable data cleaning or transformation steps.
- **Data mapping:** Source data is mapped to graph elements using RML, the open-source standard description language for this purpose.

Federators and the pipelines they define can be tailored precisely to each source, providing a specific portion of the graph. Depending on the use case, they can be further configured for:

- **ETL to Zero-ETL operation:** From caching graph segments in the triplestore (ETL) to real-time, on-demand mapping during query execution (Zero-ETL).
- **Update frequency:** Adjustable refresh intervals for cached data.
- **Caching strategy:** Support for update or upsert mechanisms for cached data.

This flexible federation approach allows handling and provisioning of virtually any data volume. Each federator can be optimized for its scenario-balancing performance, storage, and compute resources to achieve the desired operational efficiency.



03 Architecture & Components

Knowledge Graph Layer

All virtualized data is modeled and contextualized in the platform's core the **Knowledge Graph**. Entities, relationships, and business logic are defined through ontologies, ensuring that data remains interoperable, consistent, and comprehensible for both machines and humans.

The Knowledge Graph consists of two key components:

- **The Ontology**, which defines the conceptual model - classes, relationships, and rules.
- **The Fact Graph**, which contains the actual data mapped to the ontology via the various federators.

Multiple ontologies can coexist within the system. In most implementations, they do - allowing combinations of base concepts, domain-specific models, use case ontologies, and even publicly available ontologies. All ontologies follow open standards (OWL2) and are provided as Turtle files, which can be easily loaded into the platform. For version control, Git-based processes are recommended.

By supporting multiple ontologies and the federator-based pipeline concept, the platform enables the realization of federated graphs: a scalable, modular approach to unifying knowledge across diverse data sources.

Action Layer

The Action Layer is where the Knowledge Graph becomes operational: this is where federation is executed, ontologies are managed, and their interaction with individual federators is coordinated. As a result, the graph becomes traversable and fully queryable through SPARQL. Queries are executed, merged, and optimized here - either on demand by users or dynamically by Aluna, our AI agent.

A key component of this layer is the Query Library, where predefined queries can be stored, managed, and reused. These queries can be created manually by users or supported by the AI assistant. Each query can be configured in respect to its real-time needs. Queries with higher loads and less changing data can thereby only be run in efficient cycles and queries with real-time needs can be run in the moment they are referenced.

Queries serve as the foundation for dashboard widgets, external applications, AI agents, or aggregated Data Products - all accessible via the API Gateway.

Together, these capabilities make the Action Layer the bridge between knowledge and action - transforming semantic insights into interactive, operational intelligence.



Orchestration Layer

The Orchestration Layer governs the interaction between the Knowledge Graph and the use cases. It coordinates system-wide logic, access, and integration ensuring that insights and actions flow seamlessly across components.

Core functions include:

- **AI assistant integration and orchestration:** Embeds and manages the agentic system that powers Aluna, our AI agent. Its configuration ensures that ontologies are purposefully leveraged, providing structured navigation for generating accurate and context-aware queries. The models used are hosted in Europe and represent a modular technology component that is continuously refined and, when needed, replaced with optimized versions.
- **User interface management:** Controls the orchestration and presentation of the platform's user interfaces.
- **Access management & security:** Centralized management of user roles, authentication, and permissions to ensure secure and compliant access to all layers.
 - Enterprise Identity & Access Management (IAM): Native support for Single Sign-On (SSO), role-based access control (RBA), and integration with corporate identity providers (SAML).
 - Fine-grained permissions ensure secure, compliant access across user groups and data domains.
 - Each user group can be granted access to defined parts of the graph even on class-level.
- **Consumption by external systems / applications:** the information contained in the graph is accessible to other systems and applications by two basic patterns:
 - API Gateway
 - MCP / A2A – enabling external AI agents and copilots to connect directly to Aluna for semantic grounding and context-aware decision support
 - REST / JSON – for queries with low cardinality, e.g. search requests
 - Query-results as Data Products provided in Iceberg – for big analytical queries



03 Architecture & Components

Use Case Layer

The Use Case Layer connects the platform's intelligence to real-world applications. Any system can consume knowledge from the platform – from AI agents and copilots that need semantic grounding, to BI tools, custom applications, and services that rely on trusted, consistent business logic.

Users can also interact directly with the platform's built-in interfaces:

- **Graph Explorer:**
Navigate your business data in real time, explore entities, relationships, and patterns through an interactive semantic graph interface.
- **Aluna:**
Our AI agent for direct interaction – ask questions in natural language, explore the knowledge graph, and generate reusable queries for widgets or the query library.
- **Dashboard with Widgets:**
Visualize key metrics, insights, and query results in configurable dashboards tailored to each use case.

Through this layer, the platform delivers actionable knowledge – enabling both humans and AI systems to access, explore, and operationalize the full value of your enterprise's connected knowledge.



Extensibility

The platform is designed to grow with your business needs. Whether you're adding new systems, refining your data model, or embedding AI capabilities, everything is modular, standards-based, and developer-friendly.

Plug in any Data Source via Federation

Seamlessly integrate structured, semi-structured, and real-time data from CRMs, ERPs, databases, and APIs. Federators allow in-place querying and schema mapping to your ontology.

Extend Ontologies with Domain-Specific Knowledge

Use open standards like OWL to model your business logic and evolve it over time. Add custom entities, relationships or taxonomies, without disrupting the core semantic graph.

Connect to AI, Search and LLM Systems

Connect external AI agents, copilots, or agent-based systems. Through Aluna, the knowledge graph serves as a reliable, explainable grounding layer for any AI system in your enterprise.

Standardized interfaces such as Model Context Protocol (MCP) and Agent-to-Agent (A2A) communication enabled AP to interoperate within hierarchical agent topologies or distributed agent swarms, without coupling the platform's core logic to any external LLM.

Whether you're scaling across departments or embedding intelligence into operations, our open architecture makes it easy to adapt and extend.



Deployment Options

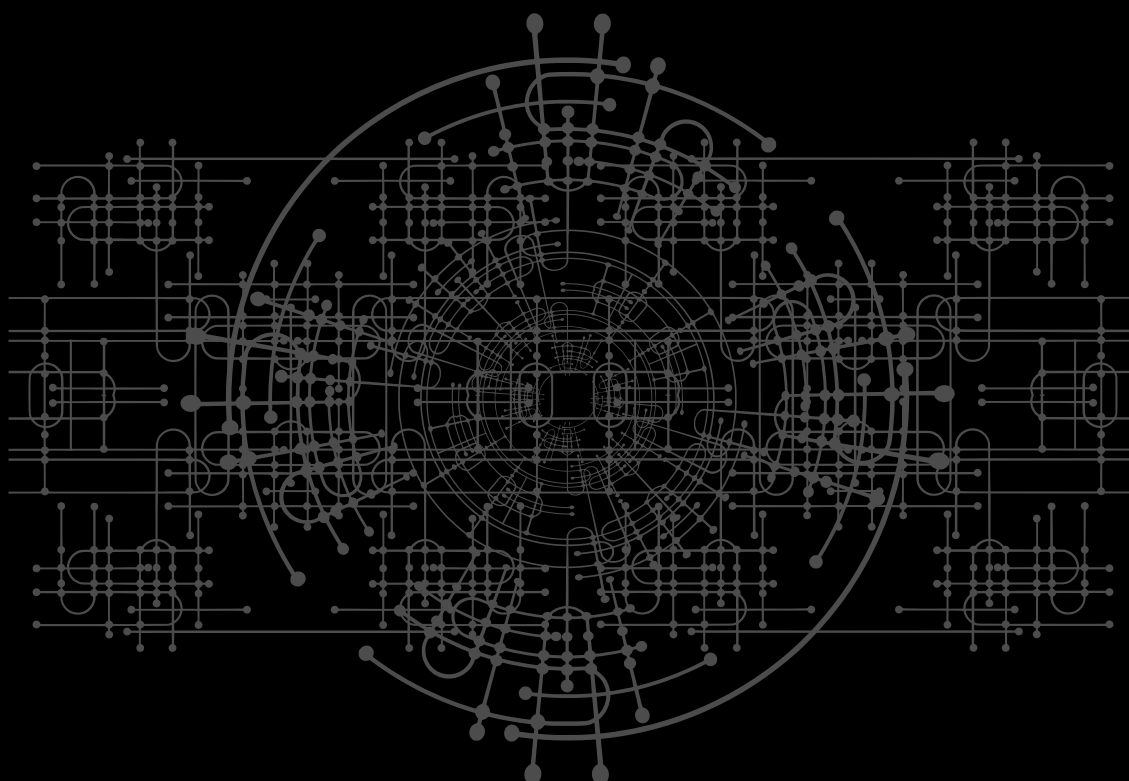
The platform offers flexible deployment models to meet your organization's security, compliance, and infrastructure needs.

Software as a Service (SaaS) in a dedicated Virtual Private Cloud

Run the platform as a fully managed service in secure, enterprise-grade environments hosted in the EU with end-to-end encryption and high availability by default.

Platform as a Service (PaaS) in your own Cloud account

For regulated industries or sensitive data environments, the platform can be deployed in an own managed Cloud account. This ensures full control over infrastructure.





Differentiators

Explainable AI, Not Just RAG

Our platform provides a live, explainable grounding layer. Aluna and the Ontology Workbench anchor every AI-generated answer in your semantic model, ensuring provenance and trust instead of just retrieving text passages.

Operational Ontology Workbench

We provide more than just a database with a modeling UI. d.AP includes an integrated workbench with governance, versioning, and executable business logic. This allows you to define business concepts once and deploy them as reusable, governed "Knowledge Products" across all systems.

Federation & True Portability

Our architecture is built on open standards (RDF/OWL) and federated access patterns. Crucially, we guarantee clear export and migration paths for your ontologies. Your business logic remains portable and independent of any single LLM, cloud, or database vendor, preventing lock-in.

Aluna: The Operational Grounding Agent

Aluna acts as a persistent, versioned grounding service for all consumers, from AI agents to dashboards. With an explicit, MCP/A2A-compatible API, it delivers provenance-first answers, moving beyond the simple "chat with your data".

Governed Multi-Consumption Patterns

d.AP serves the same trusted semantic source to AI agents, BI tools, data products, and custom applications. Centralized governance with quality gates prevents the divergence and inconsistency common in multi-tool environments, ensuring everyone operates on the same logic.

Designed for Autonomy

Business users can explore data, ask questions, and build dashboards using natural language, with no need for SQL, SPARQL, or data engineering support. Every chart and KPI is fully explainable and linked to its underlying logic, so you never wonder where a number came from.



Summary

d.AP: Your Knowledge Layer for Enterprise AI

d.AP transforms your fragmented business logic and data into a unified, living knowledge layer. We provide a single source of truth that grounds both your AI systems and your human experts in verifiable, consistent, and explainable knowledge.

The platform helps you:

- **Ground your AI and Agents:**
Deliver reliable, contextual answers by grounding your AI, copilots, and agents in a layer of trusted, explainable knowledge.
- **Create a Single Source of Truth:**
Unify business logic and eliminate silos by creating one queryable source of truth for both human decision-makers and automated systems.
- **Operationalize Business Knowledge:**
Transform your business logic into a governed, reusable asset that evolves with your organization and is directly operational in any workflow.
- **Ensure Strategic Freedom:**
Future-proof your enterprise with an open, portable knowledge architecture that prevents vendor lock-in.

The result is enterprise-wide semantic consistency, enabling true autonomy and decision velocity for every team and every system.



The Knowledge Layer

The foundation enterprise AI needs to work.

digetiers.

Lautenschlagerstr. 16
70173 Stuttgart
Germany

info@digetiers.com
+49 170 321 5984
www.digetiers-dap.com