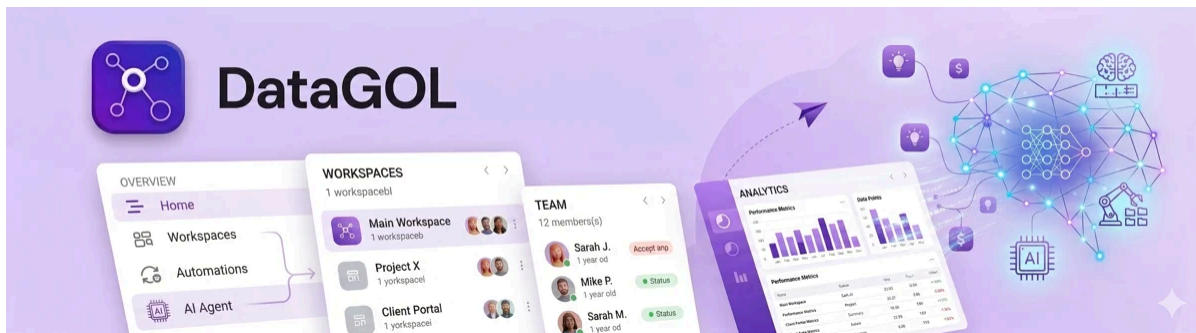




Path to becoming an AI-Native company: The 10 pillars of AI infrastructure

Around this time last year, the question was: *"What's the ROI of AI?"*

This year, it's: *"How fast can we become AI-native?"*

Here's the uncomfortable truth: most enterprise AI today is a confident liar. It sounds authoritative. It responds instantly. It's operating with incomplete data, no organizational memory, and no real understanding of your business.

And yet — with SaaS valuations dropping and boards demanding speed, the pressure to become AI-native is relentless.

Leadership pushes: Ship AI. Be AI-Native. Move fast. And so, AI gets shipped.

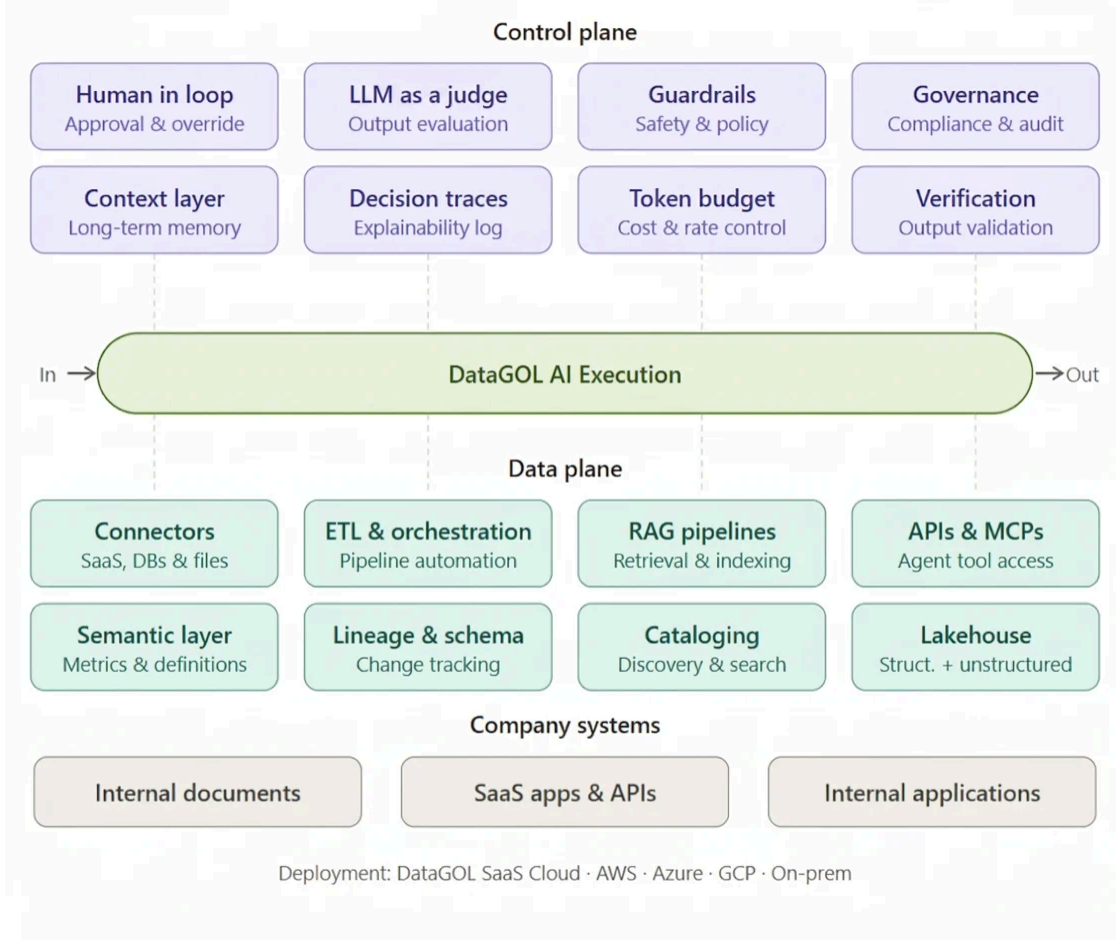
What gets shipped is patchwork: disconnected systems, fragile pipelines, and AI agents that users neither use nor trust.

The result? **Low adoption. Low confidence. Limited impact.**

In this edition, we break down what it actually takes to make AI work in the real world and the 10 pillars of AI infrastructure needed to get there.

Before we jump in the 10 pillars, let's discuss a few challenges in implementing an AI system

- Is your data truly AI-ready?
- Is context being continuously enriched? Are agents actually learning and improving over time?
- Will the vector database and RAG pipelines scale efficiently: handling growing data volumes, multimodal inputs, and optimized retrieval (latency, relevance, dimensionality)?
- Are the agents operating in a secure, governed runtime—or are we exposing ourselves to data leakage and compliance risks?
- Do we have the right guardrails and evaluation frameworks to ensure grounded responses and minimize hallucinations?



Below, we break down what we believe are the foundational pillars of a production-grade AI infrastructure.

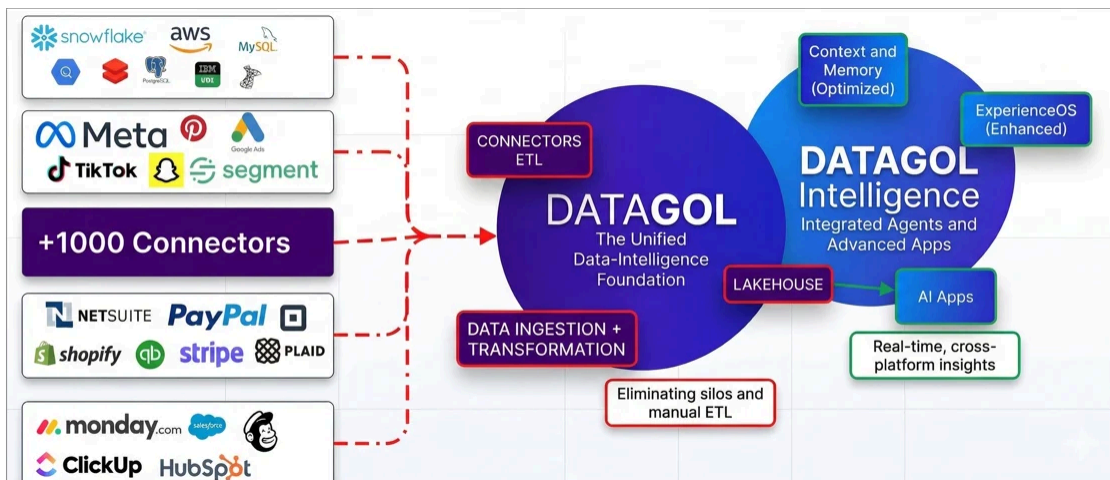
1. Unified and agentic data foundations

AI systems need connections to strong underlying data systems - whether as an API, MCP, Semantic layer using an ETL pipeline orchestration.

It must support:

- Structured and unstructured data
- Analytical and agent-facing access patterns
- APIs, tables, files, documents, events, and services
- Durable storage with governed access

The key outcome is grounded responses with full context available to AI agents.



[Learn more about DataGOL](#)

2. Native context system

AI systems do not just need data access. They need meaning. They need to know:

- What a metric means, which definition the business trusts
- How entities relate to each other: ontologies
- Which dataset is authoritative and how to resolve conflict

This requires native support for:

- Ontology, semantic models
- Knowledge graphs
- Lineage and metadata

In the old world, this was an assembly system with a **system of records**. In the AI stack - this is both **a system of record and a system of actions** to create a system of agents. Any action taken e.g. a chart is built or a decision is made is additional context. Such knowledge compounds the context and makes the agents smarter over time.

3. Traceability

Trust and Verification are key with any AI system, it is important to trace back, what decisions were made and why

- What data was used, what context was retrieved
- What model version ran, what tools were called
- What intermediate reasoning steps occurred
- What action was taken? What policy allowed it?

Without decision provenance, **intelligent systems** cannot be debugged, governed, or safely deployed. Such information must be natively logged in [AgentOS](#) - in the AI native data foundations.

4. Agent memory frameworks

Organizations need agents that can persist and reuse:

- User preferences
- Organizational memory
- Agent memory
- Outcomes from previous executions

Agent memory is not just chat history. It is the evolving working memory of an intelligent operating system inside a business. Without memory, agents remain glorified request-response interfaces. With memory, they begin to function as durable systems and become self improving.

[Read more about DataGOL's agents in action](#)

5. Beyond Basic RAG: Advanced indexing and retrieval infrastructure

Production AI systems need retrieval architectures that can combine:

- Vector search
- Graph relationships
- Structured query
- Entity-aware joins
- Freshness and recency handling
- Business-context prioritization
- Complex layout and content awareness
- Different data modality, cross references

So business users can make confident decisions with answers that are accurate, relevant, and **grounded in the full context of the data**.

6. Evaluation infrastructure for model evolution

Models evolve, and what worked yesterday won't work tomorrow. In agentic systems, CI/CD is powered by Evals, without them, shipping is just a recipe for failure.

So infrastructure must support:

- Agent evals
- Workflow evals
- Regression testing
- Model comparison
- Backward compatibility checks
- Benchmark suites tied to business tasks
- Monitoring for degradation after upgrades

Evaluation is not a nice-to-have. It is the release infrastructure for intelligent systems. So business leaders can trust that every AI update **improves outcomes** without disrupting existing workflows.

7. Scalable MCP and tool interface infrastructure

As agents become useful, they need access to more tools, systems, and services. That means organizations need a scalable way to expose:

- APIs
- Internal services
- Databases
- SaaS actions
- Workflows
- Knowledge sources

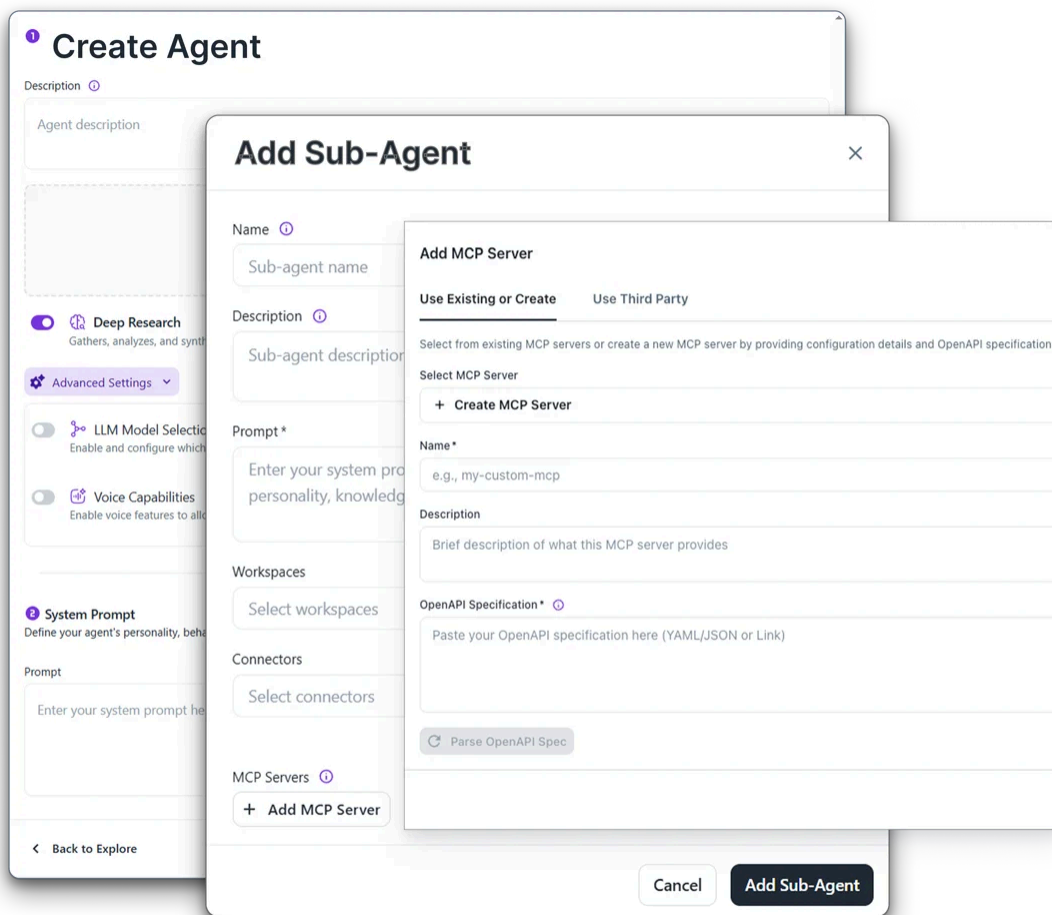
MCPs and related tool interface layers are a major part of this future.

But once again, exposing tools is not enough.

The infrastructure must handle:

- Discovery
- Permissions
- Versioning
- Execution controls
- Latency
- Auditing
- Tenancy
- Policy enforcement

We found in our implementations that tool use becomes another brittle integration surface and **context bloat** if this is not thought through.



[Read more on DataGOL's MCP](#)

8. Long-running agent execution environment and compute

AI Agents in production require

- Multi-step research
- Background investigation
- Periodic monitoring
- Queue-based operations
- Asynchronous approvals
- Complex orchestration
- Waiting on humans
- Waiting on external systems
- Revisiting tasks over hours or days

Most agent demos ignore this reality. AI infrastructure must support agents that **operate like workers**, not just chat sessions.

9. Guardrails and governed action

The more useful an AI system becomes, the more risk it creates. So useful AI infrastructure must include:

- Action guardrails
- Human approval checkpoints
- Role-based access
- Policy-aware execution
- Sensitive data controls
- Audit logs
- Compliance workflows
- Exception handling

Agents API keys vs Users API keys must be different. Small language models can be used to protect the larger model from token abuse.

10. A shared platform for analytics, agents, and applications

This may be the most overlooked requirement. Most companies still build three separate systems:

- One for analytics
- One for AI agents
- One for customer-facing AI products

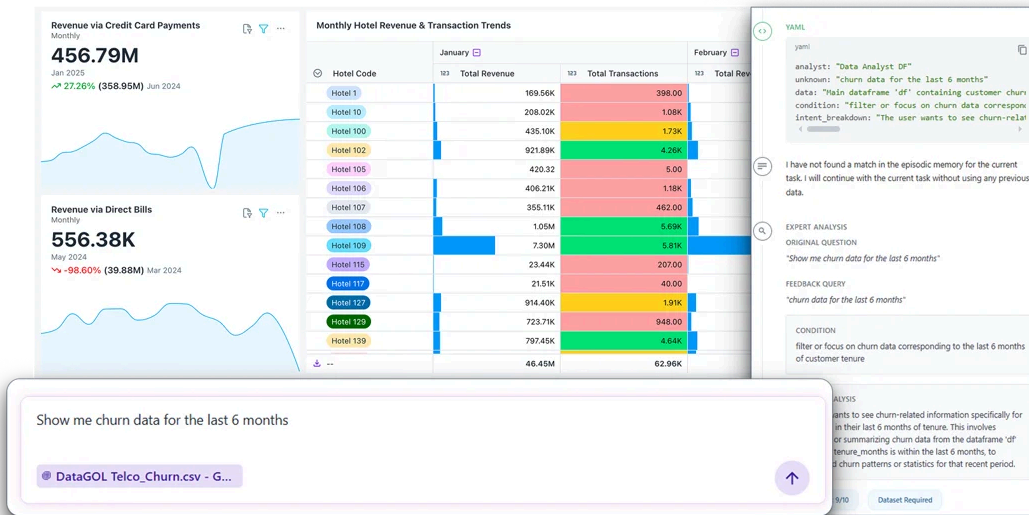
That creates duplication everywhere:

- Duplicate data movement
- Duplicate security models
- Duplicate retrieval systems
- Duplicate observability
- Duplicate governance
- Duplicate engineering effort

The [next architecture](#) should not force companies to rebuild the same foundation three times.

One infrastructure layer should support:

- Human analysis
- Internal AI operations
- External AI products.



DataGOL AI infrastructure

[DataGOL](#) unified AI infrastructure layer collapses:

- The data plane
- The context plane
- The execution plane

Into one coherent system. A platform where:

- Data is connected
- Meaning is structured
- Retrieval is native
- Agents are governed
- Memory is durable
- Evals are continuous
- Actions are traceable
- **Internal and external applications are built on the same foundation**

That is the architecture intelligent organizations will require. **DataGOL is the AI-Native infrastructure that unifies this.** The same infrastructure that powers AI inside the company should also power AI products outside it.

Read more on Medium

Get a free trial now! Try [DataGOL](#) now!

Check out [DataGOL's Documentation](#)



sales@datagol.ai

© DataGOL 2026 All Rights Reserved



DataGOL, 103 Carnegie Center, Princeton, New Jersey 08540, United States

[Unsubscribe](#) [Manage preferences](#)