Hedra White Paper

Deploying a Private AI Data Analyst in Production

How customers use Hedra to deliver fast, safe, and consistent analytics with AI

Executive summary

As organizations look to adopt AI for analytics, many face a fundamental constraint:
they want the speed and flexibility of AI-driven insights, but cannot risk exposing sensitive business data or bypassing existing governance and metric definitions.

This white paper describes how **Hedra**, a provider of private AI agents for analytics, enables customers to deploy a **production-grade AI Data Analyst** on top of their existing data stack.

Using Hedra's multi-agent architecture, customers can ask complex analytical questions in natural language—directly in Slack—and receive:

- Metric-aligned answers grounded in their existing dbt definitions
- Automatically generated and executed SQL in their data warehouse
- Human-readable summaries with clear scope and caveats
- Branded HTML and PDF reports, ready to share

All while keeping data fully within the customer's environment and under their control.

This paper explains the business context, system design, architecture, and lessons learned from running a private AI data analyst in daily production use.

1. Business context

The customer operates a digital subscription and content business where timely, accurate analytics are critical for decisions across finance, product, marketing, and operations.

Over recent years, the customer invested heavily in a modern analytics stack, including:

- A centralized cloud data warehouse (e.g. BigQuery)
- dbt as the transformation and semantic layer
- Automated ELT pipelines
- Event-level data from analytics tools, subscription systems, and payment providers

As a result, the primary challenge shifted from *data availability* to *data accessibility*.

Common questions—such as revenue growth over time, subscriber trends, or application engagement—still required:

- Translating business questions into metric names and SQL
- Repeated ad-hoc requests to the data team
- Manual exports, one-off queries, or custom dashboards

The customer needed a way to make analytics:

- Faster and more self-serve
- Consistent with existing metric definitions
- Available where teams already work
- Safe from a privacy and compliance perspective

2. Why customers choose Hedra

Customers evaluating AI-driven analytics often reject solutions that:

- Require sending data or prompts to third-party LLM providers
- Bypass their existing semantic layer
- Produce answers without auditability or traceability

Hedra is designed specifically for these environments.

Customers choose Hedra because it is:

- **Private by design** — data never leaves the customer's environment
- **Semantic-layer first** — all logic maps to dbt models and metrics
- **Production-oriented** — focused on reliability, governance, and trust

Key requirements Hedra addresses:

- Full control over data access and execution
- Clear lineage from question → SQL → result
- Consistent metrics across teams
- Outputs that can be reviewed, shared, and reused

---

3. Design principles

Hedra deployments follow a set of non-negotiable design principles.

3.1 Privacy and access control

- AI agents are not permitted to access or expose personally identifiable information (PII).

- Requests involving PII are detected and refused with clear explanations.
- Database permissions and role-based access controls are enforced end-to-end.

3.2 Semantic-layer first

Rather than generating ad-hoc SQL over raw schemas, Hedra's agents are required to:

- Use existing dbt models and metric definitions
- Inherit filters, joins, and business logic
- Avoid redefining or "hallucinating" KPIs

3.3 Multi-agent specialization

Hedra uses multiple specialized agents instead of a single monolithic assistant:

- One agent understands the semantic layer
- One generates and executes SQL
- One produces reports and narratives

This separation improves reliability, auditability, and debuggability.

3.4 Guardrails and tooling

Each agent is restricted to a defined set of tools (e.g. reading repository files, executing warehouse queries, rendering reports). This prevents unintended behavior and makes failures easier to diagnose.

3.5 Human-readable outputs

Every answer includes:

- The numbers
- Scope, timeframe, and filters

- Plain-language interpretation
- Data freshness and limitations

---

4. Architecture overview

At a high level, Hedra deploys a **multi-agent AI Data Analyst** on top of the customer's existing analytics stack.

4.1 Semantic layer integration

Hedra reads the customer's dbt project, including:

- Dimensional models
- Fact tables
- Metric definitions in YAML

This allows the system to understand:

- Where each KPI is defined
- How it is calculated
- Which dimensions and time grains are valid

4.2 Multi-agent system

A typical deployment includes:

- **Coordinator agent**
  Interprets the user's intent and orchestrates other agents.
- **Context agent**
  Inspects dbt models and metrics to identify the correct sources of truth.
- **Query agent**
  Generates warehouse-specific SQL, executes it, handles errors, and returns structured

results.

- **Report agent**
  Produces HTML and PDF reports using predefined templates and delivers them via Slack.

Optional agents can generate charts or retrieve external benchmarks when required.

4.3 Slack as the interface

Slack acts as the primary interface for business users.

A typical flow:

1. A natural-language question is asked in Slack
2. Scope and intent are clarified automatically
3. SQL is generated and executed in the warehouse
4. A structured answer and shareable report are returned

What previously required multiple interactions across teams is consolidated into a single conversational flow.

---

5. Example workflows

5.1 Multi-year growth analysis

A stakeholder asked for growth in revenue, subscribers, customers, and engagement over multiple years, including YoY comparisons.

Hedra's agents:

- Identified the relevant models and metrics
- Computed baseline and latest full periods
- Calculated total growth, YoY change, and CAGR

- Delivered a one-page PDF report directly in Slack

All calculations respected the customer's existing metric definitions.

5.2 App activation and OKRs

For a mobile application, Hedra analyzed event-level data to:

- Establish activation and engagement baselines
- Derive conversion and stickiness metrics
- Propose concrete OKRs for activation and monetization

This allowed product teams to move directly to prioritization and decision-making.

---

6. Impact and lessons learned

Customers report clear qualitative benefits:

- Faster answers to complex analytical questions
- Consistent metrics across teams
- Reduced dependency on specialists for routine analysis
- Improved data literacy among non-technical stakeholders

Key lessons:

- Guardrails and data freshness handling are essential
- Report templates dramatically improve output quality
- Multi-agent systems are easier to operate in production than monolithic assistants

---

7. Governance and confidence

Hedra emphasizes transparency and trust:

- Generated SQL is retained for review
- All outputs map back to governed metrics
- Errors are surfaced and handled explicitly

Each response includes a **confidence score** (1–100) based on execution quality, metric clarity, and the degree of assumptions involved. This helps teams decide when additional validation is required.

Conclusion

The real promise of AI in analytics is not simply "chatting with data," but connecting natural language to a governed semantic layer—and turning the result into reliable, reusable outputs.

Hedra enables customers to deploy private AI data analysts that:

- Work with existing data stacks
- Respect governance and privacy constraints
- Deliver insights directly where teams work
- Scale analytics without scaling headcount

Rather than replacing analysts, Hedra acts as a force multiplier—helping organizations move faster while maintaining trust in their data.