# LingoAI Observability Platform – 8‑Slide Executive Deck

**Duration:** 40–45 minutes (≈5 min/slide + Q&A) **Audience:** Engineering leadership, architects, SRE/ops, product owners

## **Slide 1  |  Title & Agenda**

**Title:** *LingoAI Observability Platform*  
**Subtitle:** Illuminating Every Step of the Multi‑Agent Journey

**Agenda**

1. Context & Imperative
2. Problems We’re Solving
3. Architectural Blueprint
4. Runtime Interaction Sequence
5. Log Schema & Data Flow
6. Metrics & Insights
7. Roadmap
8. Q&A

**Speaker Notes (≈4 min)**

“Welcome! In the next 40 minutes we’ll unpack why observability is mission‑critical for LingoAI, the design principles behind our platform, how each component interacts, and where we’re headed next.”

## **Slide 2  |  Context & Imperative**

### Why Now?

* Explosion of **multi‑agent workflows** → exponential events
* **Enterprise SLA** expectation: sub‑second answers, 99.9 % uptime
* Regulatory focus on **explainability** & **audit trails**

### The Cost of Blind Spots

* Hours lost chasing intermittent errors
* Token overruns unnoticed until month‑end bill
* Hard to prove accuracy or compliance without traceability

**Speaker Notes (≈6 min)**

“As agents proliferate, blind‑spot costs grow non‑linearly. A single latent API or hallucinating LLM can ripple across the customer journey. Observability turns unknowns into actionable data.”

## **Slide 3  |  Goals & Value Proposition**

| **Goal** | **Outcome** |
| --- | --- |
| **Unified Trace Schema** | One record format for every span across agents, tools, LLMs |
| **Real‑Time Streaming** | Ops teams see issues within < 3 s of occurrence |
| **Human + LLM Feedback Loop** | Continuous quality scoring & rapid tuning |
| **Full‑Stack Correlation** | AI events mapped to node/DB/infra metrics in Dynatrace |
| **Self‑Service Analytics** | Product & data teams explore without engineering tickets |

**Speaker Notes (≈4 min)**

“The platform isn’t just for SRE. Product can measure feature‑level latency, data scientists can study misuse, and compliance gains an immutable audit trail.”

## **Slide 4  |  Architectural Blueprint**

%%{init: {"theme":"base","flowchart":{"nodeSpacing":35,"rankSpacing":30}}}%%  
flowchart TD  
 subgraph AG[Agent Layer]  
 P["Primary Assistant"]:::agent --> S["Secondary Agents"]:::agent  
 S --> T["Tool / API"]:::tool  
 end  
 T --> L["Log Ingestor"]:::ing  
 L -->|Validate & Enrich| M["MongoDB  
(Structured Spans)"]:::db  
 M --> CS["Change Stream"]:::cs  
 CS --> API["Obs API  
(SSE/WS)"]:::api  
 API --> UI["React Dashboard"]:::ui  
 L --> DT["Dynatrace  
(APM + Logs)"]:::dt  
  
 classDef agent fill:#D5E8D4,stroke:#4AAE4F,stroke-width:2px;  
 classDef tool fill:#FFF2CC,stroke:#C8A000,stroke-width:2px;  
 classDef ing fill:#F8CECC,stroke:#C0504D,stroke-width:2px;  
 classDef db fill:#DAE8FC,stroke:#6C8EBF,stroke-width:2px;  
 classDef cs fill:#E1D5E7,stroke:#9673A6,stroke-width:2px;  
 classDef api fill:#EFEFEF,stroke:#666,stroke-width:2px;  
 classDef ui fill:#FFFFFF,stroke:#000,stroke-dasharray:5 3;  
 classDef dt fill:#D9EAD3,stroke:#6AA84F,stroke-width:2px;

**Speaker Notes (≈7 min)**

“Color‑coding helps: green boxes are agents, yellow is external tools, red the ingest firewall, blue for our durable store, purple for real‑time bus, grey API, white UI, and olive shows Dynatrace integration. Key principle: **write once, stream everywhere**.”

## **Slide 5  |  Runtime Interaction Sequence**

%%{init:{"theme":"base","sequence":{"actorBkg":"#F0F0F0","actorBorder":"#555"}}}%%  
sequenceDiagram  
 autonumber  
 participant U as User  
 participant PA as Primary Assistant  
 participant SA as Secondary Agent  
 participant SV as External Service  
 participant LI as Log Ingestor  
 participant DB as MongoDB  
 participant AP as Obs API  
 participant UI as Dashboard  
  
 U->>PA: Submit Query  
 PA->>LI: agent-span  
 PA->>SA: Delegate Task  
 SA->>LI: llm-span  
 SA->>SV: API Call  
 SV-->>LI: tool-span / error  
 SV-->>SA: Result  
 SA-->>PA: Collated Answer  
 LI->>DB: Persist Spans  
 DB-->>AP: Change Stream Event  
 AP->>UI: Push SSE/WS  
 UI-->>U: Live Update

**Speaker Notes (≈6 min)**

“Green arrows represent normal spans; red signals an error. Notice the constant side‑channel to the Dashboard—operators gain second‑level awareness without polling.”

## **Slide 6  |  Log Schema & Data Flow**

{  
 "trace\_id": "123e4567-e89b-12d3-a456-426655440000",  
 "span\_id": "span‑uuid",  
 "parent\_span\_id": null,  
 "event\_type": "tool",  
 "agent\_type": "plan\_agent",  
 "status": "error",  
 "exception": {  
 "type": "TimeoutError",  
 "message": "Benefits API timed out after 5 s"  
 },  
 "metrics": {  
 "prompt\_tokens": 0,  
 "completion\_tokens": 0,  
 "duration\_ms": 5123,  
 "cost\_usd": 0  
 },  
 "feedback": {  
 "llm\_score": 0.40,  
 "user\_rating": 1,  
 "user\_comment": "Missing plan details"  
 },  
 "timestamp": "2025‑06‑20T14:32:08.123Z"  
}

**Why It Matters**

* Immutable audit: every key/value traceable
* Joins unnecessary—query by any field
* Compatible with OpenTelemetry → Dynatrace ingestion

**Speaker Notes (≈5 min)**

“This schema is our lingua franca. Everything—agent, tool, feedback—becomes queryable. And because it mirrors OTLP, we can flip a switch to export into any APM.”

## **Slide 7  |  Metrics & Insights**

**Live Widgets (Sample):**

* **Requests/min:** ▷ 135
* **Avg Latency:** ▷ 620 ms
* **Token Usage:** ▷ 12 K / hr
* **Error Rate:** ▷ 1.8 %
* **Avg User Rating:** ▷ 4.3 / 5

**Alert Examples:**

1. Error Rate ≥ 3 % over 5 min → Slack #ops
2. Avg Latency ≥ 1 s for Plan Agent → PagerDuty Tier 2

**Speaker Notes (≈4 min)**

“Metrics surface the big picture; alerts zero in on action. By coupling logs and Dynatrace metrics, we can answer ‘is this latency spike code or Kubernetes?’ in seconds.”

## **Slide 8  |  Roadmap & Call to Action**

1. **Q3 2025** – Migrate to OTLP Collector; add per‑tenant RBAC dashboards
2. **Q4 2025** – Anomaly detection via WhyLabs; SLA alert integration
3. **Q1 2026** – Automated remediation (scale agents, retry jitter)
4. **Get Involved:**
   * Developers: instrument agents with new schema helpers
   * Ops: define alert thresholds
   * Product: propose feedback heuristics

**Speaker Notes (≈4 min)**

“Success hinges on cross‑team adoption. Whether you write code, triage tickets, or own product KPIs, there’s a hook for you to contribute.”

## **Slide 9  |  Q&A**

**Thank You!**  
Contact: [genoj.mathew@lingoai.com](mailto:genoj.mathew@lingoai.com)

**Speaker Notes (3–5 min)**

“Happy to dig deeper on any component.”