

OBSERVABILITY
— IN THE —
AI-NATIVE AGE

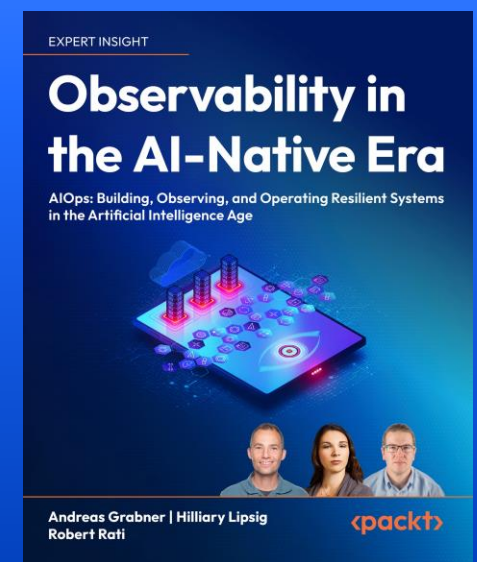
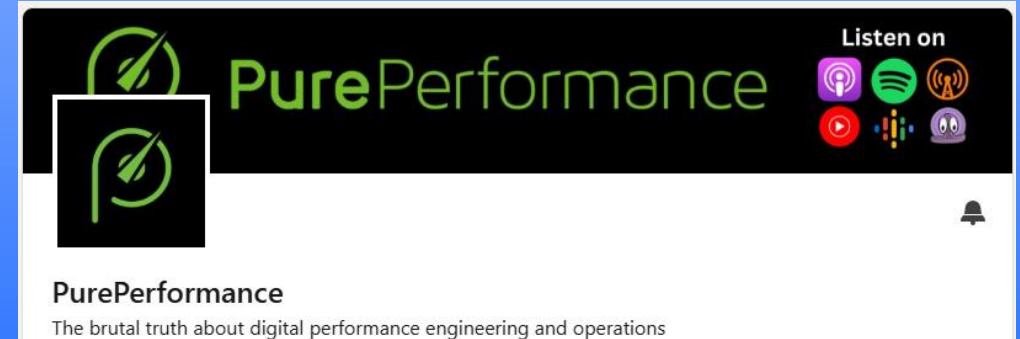
**HELPING THE HUMAN AND THE AI
THROUGH OBSERVABILITY**

Who am I?



Andi (Andreas) Grabner

- CNCF Ambassador
- Fellow DevRel @ Dynatrace
- PurePerformance Podcaster
- Occasional Book Author
- Salsa Dancer



When you hear “Observability and AI”

What is the first Use Case you think of?



1

**Monitor and
Optimize the
AI/LLMs we host**

2

**Monitor usage of
CoPilot, Claude,
Cursor, Codex**

3

**Integrate
Observability into
IDE, e.g: via MCP**

4

**A different
use case - or –
no thoughts!**

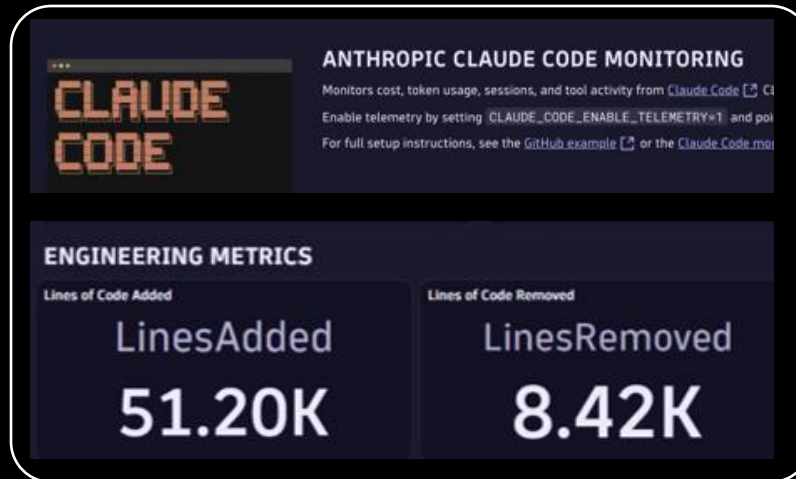
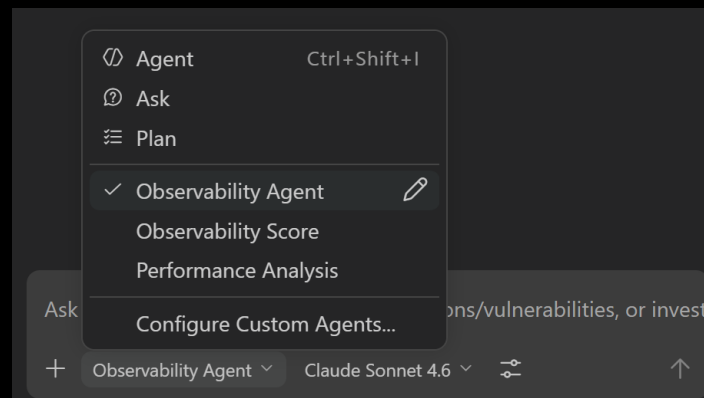
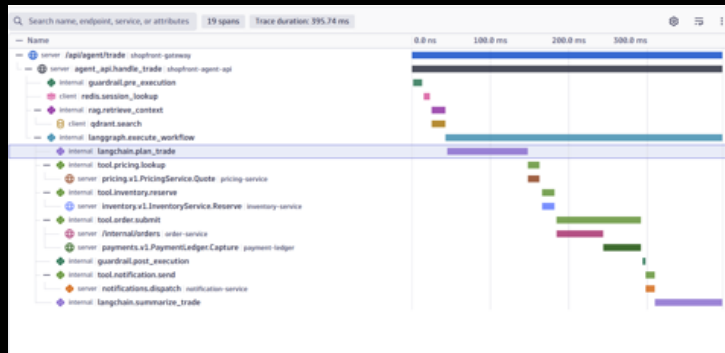
Based on what I see, I've prepared this!

3 Things I hope you take away today + maybe a bonus point 😊

The Basics
Observability Standards

The AI Delivery Lifecycle
Observability for and with AI

Cost and Impact
Optimize usage and alternatives



“What’s different in observing an app, an AI or a workflow?”

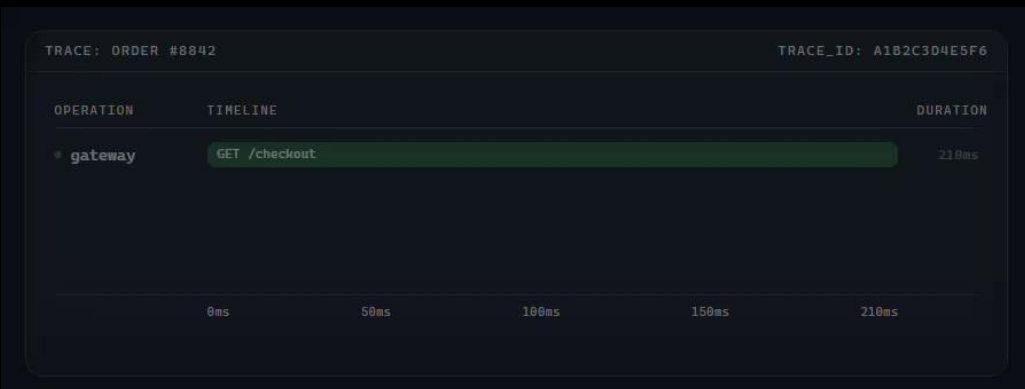
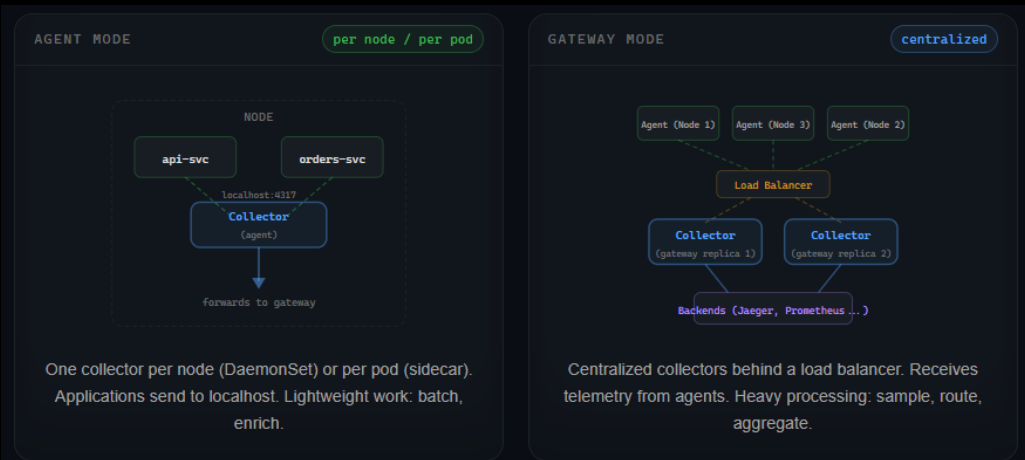
“How can my AI become better with close observability loops?”

“How to maximize the outcome while staying within budget?”

THE OBSERVABILITY
— AND AI PRIMER: —
BASICS YOU
NEED TO KNOW!

New to OTEL? 12 Koans of OpenTelemetry!

A playful way to understand observability: <https://otel.mreider.com/>



CHECKOUT.JS

```
function checkout(order) {
  // ...

  let total = calculateTotal(order)
  chargePayment(order.user, total)
  updateInventory(order.items)

  return { status: "confirmed" }
}
```

TELEMETRY OUTPUT

nothing - the code is silent

CHOOSE LINES TO ADD

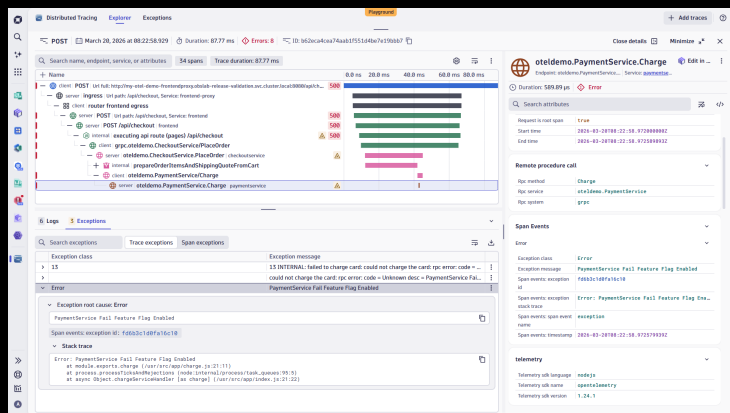
- `let span = tracer.startSpan("checkout")`
- `console.log("done")`
- `span.setAttribute("order.id", order.id)`
- `meter.record("order.total", total)`
- `span.end()`

3 Types of Traced Systems: App, Agentic, GH Workflow

Its all Spans: but they tell a different story and enable different use cases

App Requests

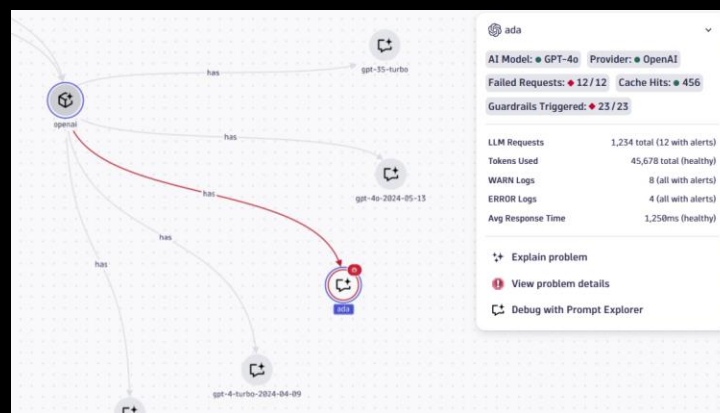
User clicks on a link ...



“Aah – the root CAUSE of my HTTP 500 is THIS exception!”

Agentic Workflow

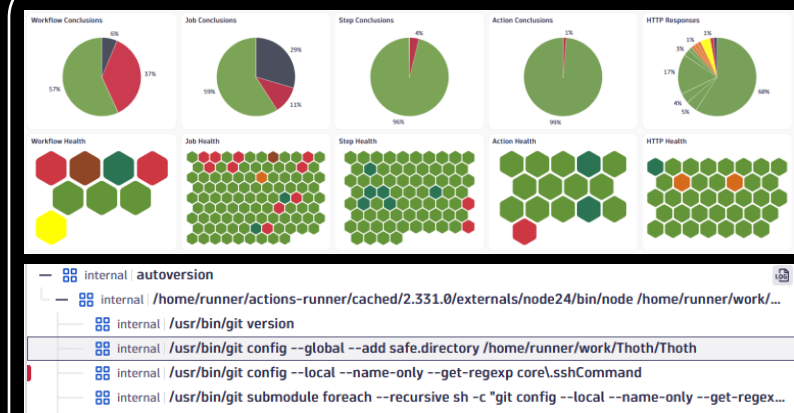
Agent takes complex tasks



“Aah – THAT workflow called the SAME tool 10 times!”

GitHub Workflow

Pull Request triggers CI/CD



“Aah – THAT GitHub action connects to the WRONG k8s!”

Type 1: Analyzing Root Cause of an App Request

What caused a problem? Exception, Database, CPU, Memory ...

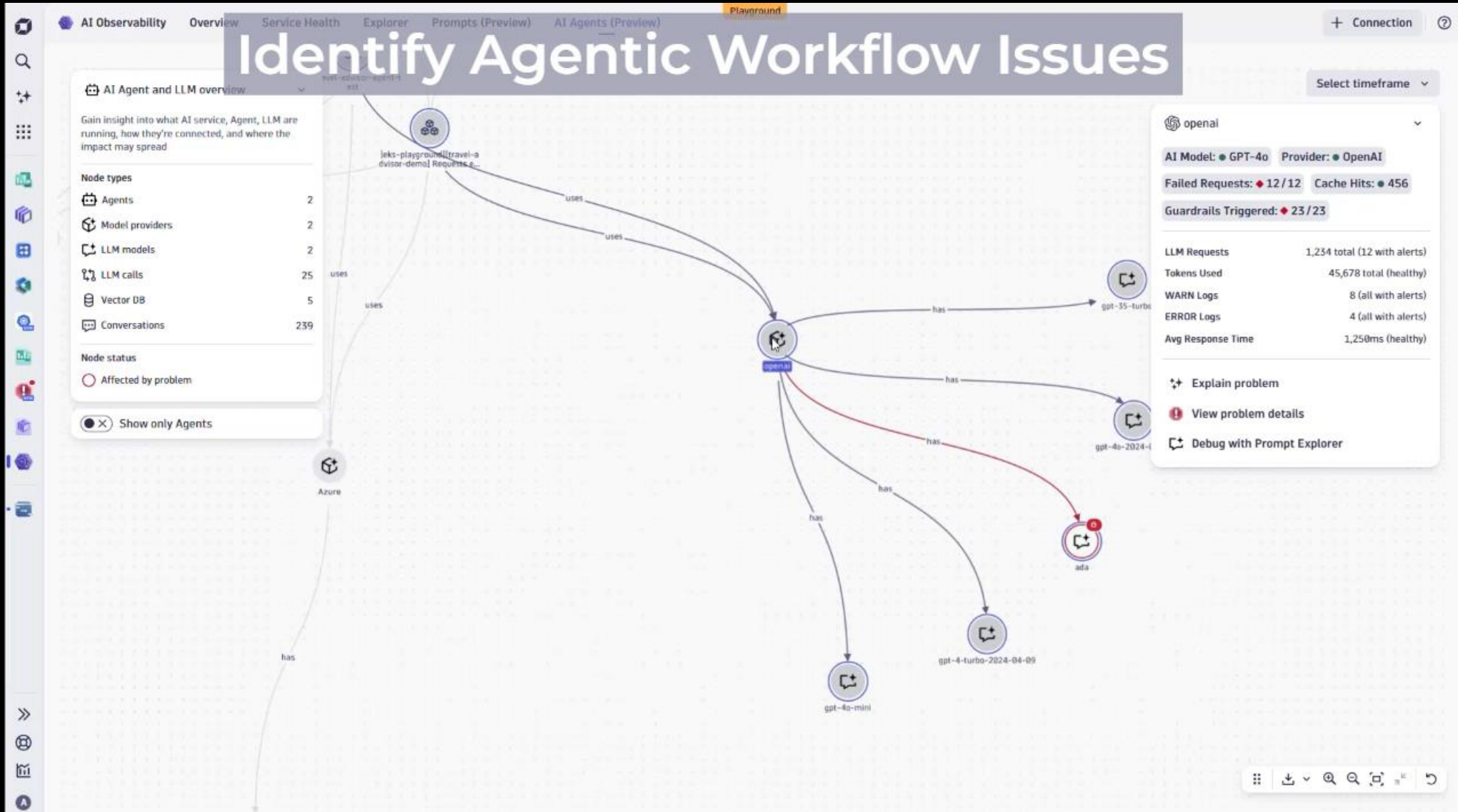
Identify Exceptions that cause an Error

Exceptions 10 records

Exception class	Exception message	Contribution breakdown	Count	Failed	Failure rate
PHP Deprecated			370	0	0%
langgraph.errors.ParentCommand			330	330	100%
14			270	270	100%
13	13 INTERNAL: failed to charge card: could not charge the card: rpc error: code = Unk...		250	250	100%
Error			250	250	100%
*connect.Error	not_found: Flag not found		210	210	100%
error			30	30	100%
org.dynatrace.microblog.exceptions.NotLoggedInException	null		20	10	50%
grpc_channel_MultiThreadedRendezvous	<_MultiThreadedRendezvous of RPC that terminated with: status = StatusCode.DEA...		10	10	100%
org.dynatrace.profileservice.exceptions.BioNotFoundException	Bio for user with id '2732' not found		10	0	0%

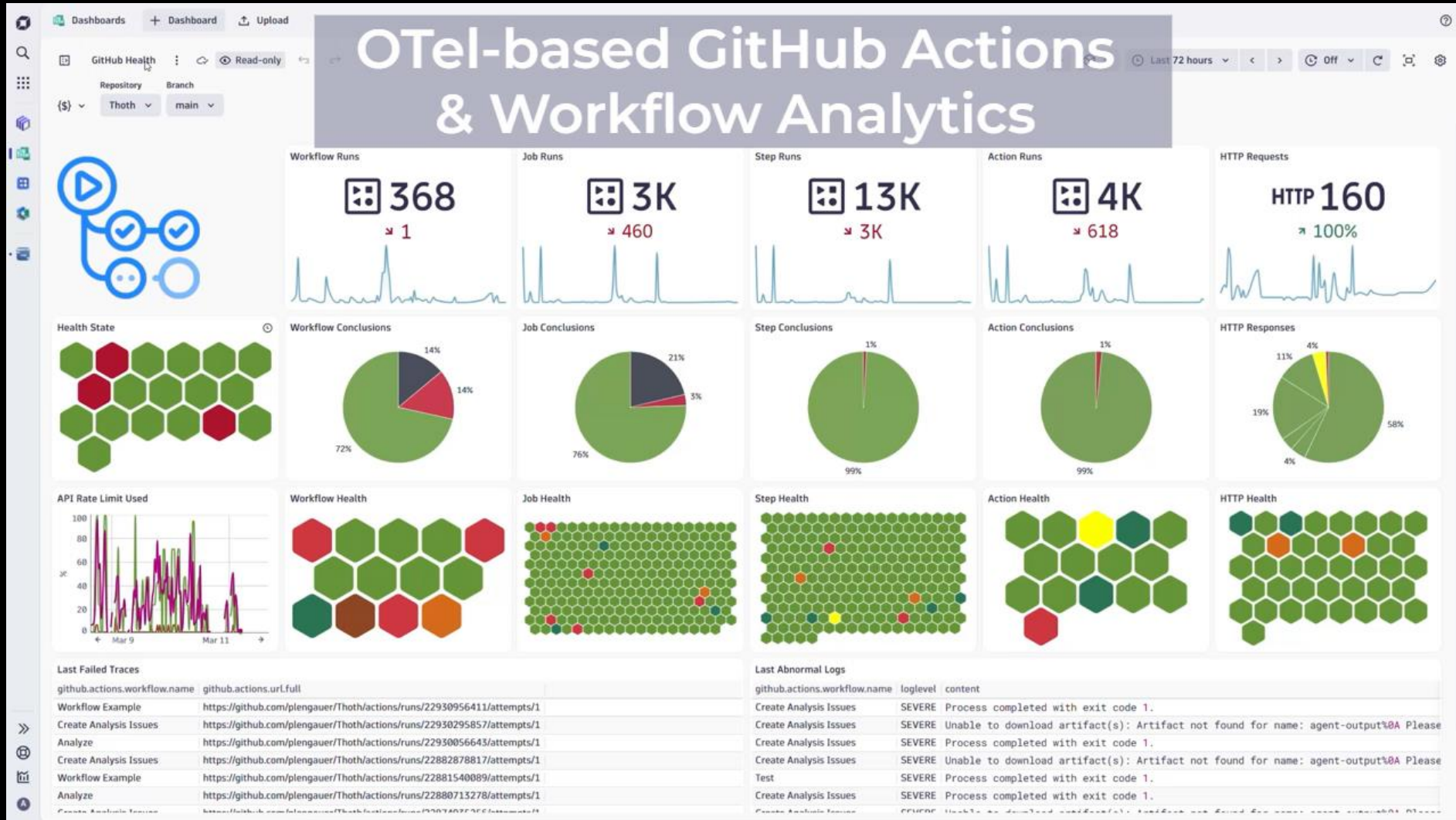
Type 2: Understand an Agentic Workflow

What tools are executed, which models used ...?



Type 3: GitHub Workflow Insights

Insights into your Workflows, Actions, Bash, Agents ...



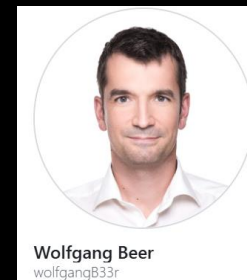
Philipp Lengauer
plengauer



<https://dt-url.net/devrel-pl-github-thoth>

Simulate your own OpenTelemetry

<https://github.com/wolfgangB33r/otel-demo-service>



Wolfgang Beer
wolfgangB33r



```
PS C:\GitHub\grabnerandi\otel-demo-service> python .\app.py
```

OTEL Demo Service Control

ai-agent-application

Purpose: simulate a LangGraph and LangChain RAG agent running an autonomous shopfront.
Topology: gateway, agent API, graph runtime, retrieval, tools, and commerce backends share one trace.
Signal shape: spans include GenAI model, token, cache, tool, and guardrail observability fields.
Failure modes: vector search slowness, LLM rate limits, tool failures, cache staleness, and checkout delays.
Best for: validating AI agent observability, RAG flow analysis, and autonomous commerce operations.

Path: scenarios/ai-agent-application.py
Status: ● Running
PID: 26296

Scheduled Problems

40 3 * * * → slow_vector_search for 60 min	Remove
40 3 * * * → llm_rate_limit for 60 min	Remove
40 3 * * * → tool_failures for 60 min	Remove
40 3 * * * → guardrail_blocks for 60 min	Remove
40 3 * * * → stale_inventory_cache for 60 min	Remove
40 3 * * * → checkout_latency for 60 min	Remove

slow_vector_search Cron (e.g. 0 0 * * 1) 60 Add

Examples: Monday 00:00 = 0 0 * * 1, Tuesday 13:00 = 0 13 * * 2

Request Rate: 10 req/min

Start Stop

Distributed Tracing Explorer

Search spans: 1000 spans

Start time	Duration	Gen ai system	Gen ai request	Gen ai respons...	Gen ai usage input tok...	Service	Gen ai prompt	Gen ai completi...	Gen ai completi...	Llm request type	Trace id
Mar 18, 14:21:59.010	102.66 ms	openai	gpt-4.1-mini-2...	gpt-4.1-mini-2...	2,395	langchain-orch...					a97633476682...
Mar 18, 15:40:42.856	210.71 ms	openai	gpt-4.1-mini-2...	gpt-4.1-mini-2...	2,394	langchain-orch...					6ead171f6491...
Mar 18, 14:17:55.816	94.53 ms	openai	gpt-4.1-mini-2...	gpt-4.1-mini-2...	2,391	langchain-orch...					2df9003df99d5...
Mar 18, 15:55:32.767	159.27 ms	openai	gpt-4.1-mini-2...	gpt-4.1-mini-2...	2,391	langchain-orch...					d3bb0b401dde...
Mar 18, 14:21:07.691	189.99 ms	openai	gpt-4.1-mini-2...	gpt-4.1-mini-2...	2,389	langchain-orch...					675f612c0f254...
Mar 18, 14:21:46.254	169.56 ms	openai	gpt-4.1-mini-2...	gpt-4.1-mini-2...	2,386	langchain-orch...					f058b4226c6e...
Mar 18, 14:10:38.000	170.36 ms	openai	gpt-4.1-mini-2...	gpt-4.1-mini-2...	2,384	langchain-orch...					62806f481f77...

/api/agent/trade

March 18, 2026 at 14:21:58.965 Duration: 395.74 ms id: a97633476682f2e7f9f77e054bb54f4c

Search name, endpoint, service, or attributes 19 spans Trace duration: 395.74 ms

langchain.plan_trade

Service: langchain-orchestrator
Duration: 102.66 ms

Search attributes

gen ai

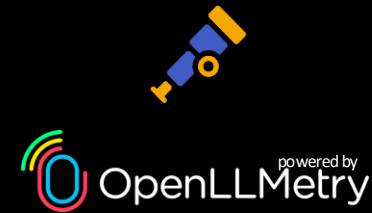
Gen ai agent name	shopfront-autonomy-agent
Gen ai cache hit	false
Gen ai operation name	chat
Gen ai prompt template	shopfront_trade_planner_v3
Gen ai provider name	openai
Gen ai rag documents retrieved	4
Gen ai request max tokens	768
Gen ai request model	gpt-4.1-mini-2026-02-01
Gen ai request temperature	0.17
Gen ai response finish reason	stop
Gen ai response model	gpt-4.1-mini-2026-02-01
Gen ai system	openai
Gen ai usage input tokens	2395
Gen ai usage output tokens	241
Gen ai usage total tokens	2636


telemetry

Telemetry sdk language	python
Telemetry sdk name	opentelemetry
Telemetry sdk version	1.40.0

Instrument your own LLM Challenge!

<https://community.open-ecosystem.com/>





☰  **OPEN ECOSYSTEM** Topics Challenges About


🏠 Challenges > 🗺️ The AI Observatory > tags > Latest Hot


🏷️ Topic

📌 About the The AI Observatory category
adventure-03

Welcome to the third adventure in the Open Ecosystem Challenge series!
Your mission: investigate a mysterious bandwidth anomaly at a remote research station by instrumenting its AI system. This is a hands-on journey thro... read more 

🟡 Instrument & Debug a RAG Pipeline: Adventure 03 | Intermediate is Live!
opentelemetry, ai, prometheus, openllmetry, jaeger 

🟢 Instrument Your First LLM: Adventure 03 | Beginner is Live!
opentelemetry, ai, openllmetry, jaeger, the-ai-observability 

🔴 Reduce Telemetry Noise: Adventure 03 | Expert is Live!
opentelemetry, ai, openllmetry, jaeger 

🔴 Reduce Telemetry Noise: Adventure 03 | Expert is Live! ✎

🏠 Challenges 🗺️ The AI Observatory opentelemetry, ai, openllmetry, jaeger



KatharinaSick 📌

18d

In Adventure 03: The AI Observatory, you made it to RaviHyral aboard the **Perihelion** (ART). But now Outpost Verada's monitoring team is complaining — ART's traces are a mess. Non-standard span names, missing token usage, and Jaeger drowning in noise from every single request.

Your mission? **Fix the instrumentation, record errors properly, and filter out the noise.**

🔴 Expert: The Noise Filter

In this level, you will take ART's observability from "technically working" to closer to production-ready.

🧠 What You'll Learn:

- **GenAI Semantic Conventions:** How to follow the OpenTelemetry GenAI spec for LLM spans, including token usage attributes.
- **Tail Sampling:** How to configure the OpenTelemetry Collector to keep only meaningful traces (errors and slow requests in this case) and drop the rest.

🔧 The Tech Stack:

- [Python](#) & [Ollama](#) (Local LLM)
- [LangChain](#) & [Qdrant](#) (Vector Database)
- [OpenTelemetry](#) & [OpenLLMetry](#)
- [Jaeger](#)
- [Kubernetes](#) (pre-provisioned)



THE

AI DELIVERY
LIFECYCLE

OBSERVABILITY
FOR AND WITH AI

Observability in the AI-Delivery Lifecycle

Provide code suggestions based on insights, change it, validate it ...

File Edit Selection View Go ... Untitled (Workspace)

CHAT

Analyze my security and performance hotspots based on my observability data

Add Context...

Can you ide

Local Agent Claude Sonnet 4.6

main* 0 0

Ln 18, Col 78 Spaces: 4 UTF-8 CRLF Agent

MCP or CLIs

- Source Guides
- ...
- Logs Metrics Spans
- ...
- CVEs Exploits
- ...
- PRs Deploys Flags
- ...

Observability in the AI-Delivery Lifecycle

Provide code suggestions based on insights, change it, validate it ...

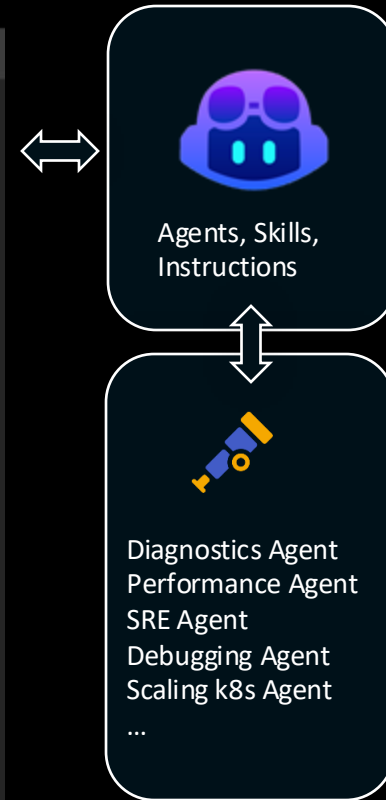
The screenshot shows an IDE window titled "opentelemetry-demo" with a chat interface. A large text overlay reads "Create your own agents based on learnings ...". The chat content includes:

- Overview of logs and vulnerabilities in opentelemetry-demo.
- Query results for "otelcol 'connection refused' logs": 11.92 GB, 216.2M, 0.
- Subtotal: ~176 GB.
- Session Totals table:

Metric	Value
Total data scanned	~614.6 GB
Budget consumed	61.5% of 1,000 GB
Remaining budget	~385 GB
Successful queries	15
Failed queries (parse errors / wrong field)	8

- Cost Observations:
 - ~58% of data scanned (~358 GB) came from just 3 queries that scanned the full logs index without a tight filter — the two failed `service.name` attempts and the `loglevel` summary on `currency.service`.
 - The correct pattern — filtering by `k8s.deployment.name` with `matchesPhrase()` — is much more targeted and should be the default for all future queries against logs in this environment.
 - The span queries are relatively cheap (~6 GB each) since the spans index is smaller than logs.

At the bottom, a chat input field shows "Based on" and "Agent Claude Sonnet 4.6".



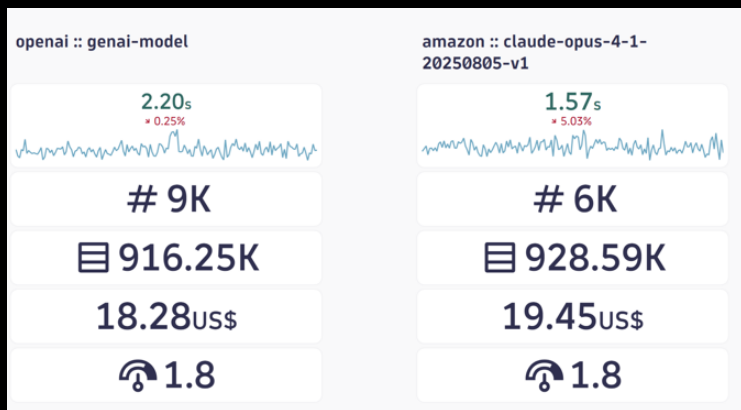
UNDERSTANDING
— EFFICIENCY, —
USAGE, GUARDRAILS,
AND COSTS OF AGENTS

3 Lenses on AI Usage, Cost and Guardrails

Models, MCPs, Agents

Models

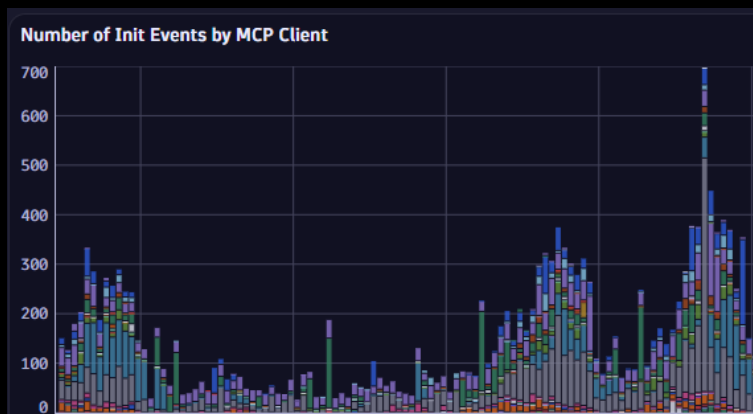
A/B Testing for Use Cases



“Which model is best for every use case?”

MCPs

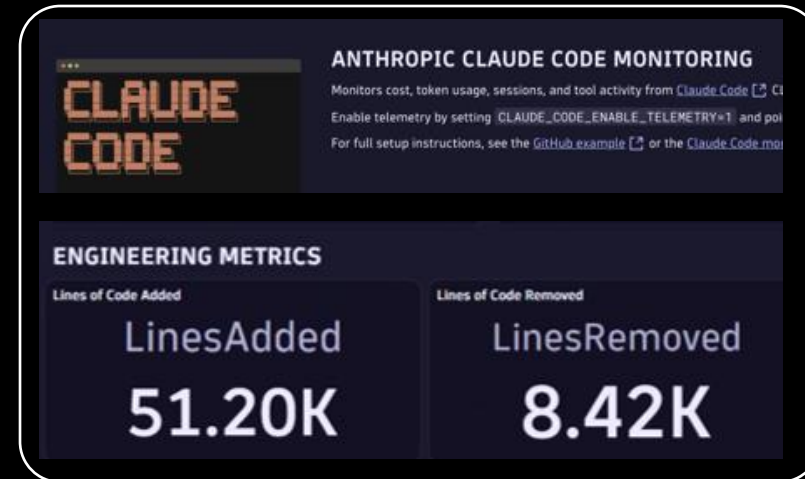
Adoption, Performance



“Which tools are used? Which tools aren’t? And Why?”

Agents

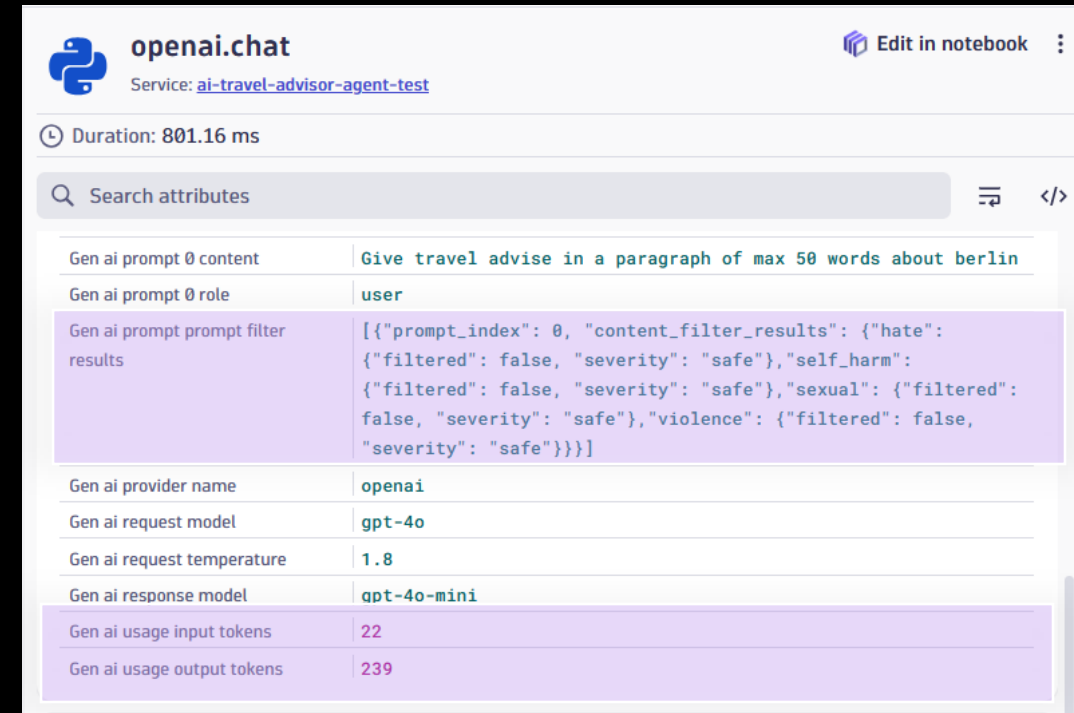
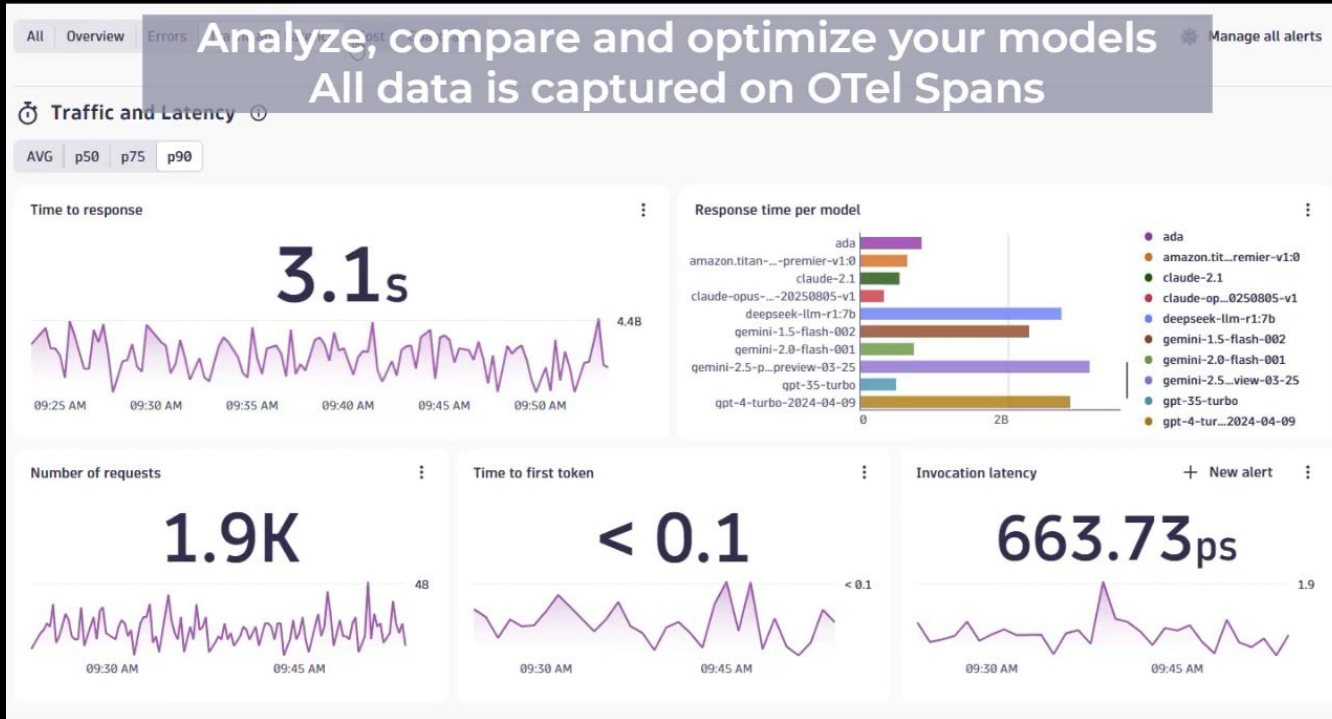
Impact vs Cost



“Who is using this? And what’s the ROI?”

Analyzing different Models – A/B Testing

All data is on the spans or comes in through metrics



MCP Usage Observability

Understand how often your MCP is used by which agent, which tools ...

[CA] [Dynatrace MCP Server] MCP Client Tracking
Read-only
Save as new
Track your MCP Server Usage & Errors
Off

Track your MCP Server Usage & Errors

Who is using them? Popular tools? Errors?


Tracking MCP Client Usage (Demo Env)

Sessions

countDistinct(dt_rum.session.id)

21K

Number of Init Events by MCP Client

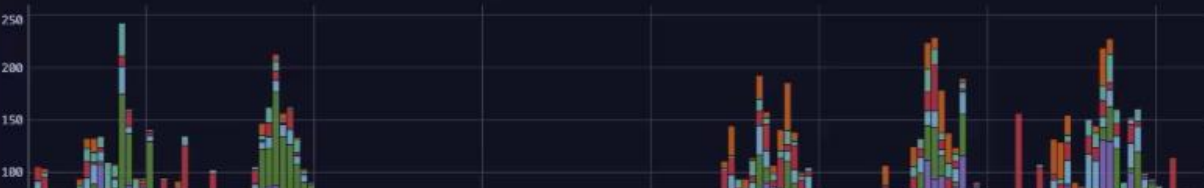


- @llbrech...
- Cline
- JetBrains...
- JetBrains...
- Q DEV CL1
- Visual St...
- Windsurf
- augment...
- claude-co...
- codex-shell
- continue...
- dcx-moni...
- dt-local...
- dynatrace...
- @n8n/n8...
- Cursor
- JetBrains...
- JetBrains...
- Roo Code
- Visual St...
- antigravit...
- claude-ai
- codex-mc...
- continue...
- cursor-vs...
- devwarp...
- dt-script
- fetchLogs...

Top 10 used MCP Clients

client_name	initEvents	versions
claude-code	4,780	2.1.74, 2.1.72, 2.1.70, 2.1.63, 2.1.69, 2.1.73, 2.1.5, 2.1.45, 2.1.71, 2.1.12, and 24 more
mcp	4,390	0.1.0
codex-mcp-client	2,856	0.114.0, 0.112.0-alpha.3, 0.115.0-alpha.4, 0.108.0-alpha.12, 0.112.0, 0.113.0, 0.111.0, 0.104.0, 0.108.0-alpha.8, 0.107.0, and 3 more
github-copilot-developer	1,630	1.0.0
q-chat-plugin	1,533	1.0.0
cursor-vscode	1,237	1.0.0
Visual Studio Code	947	1.111.0, 1.110.1, 1.109.2, 1.110.0, 1.106.1, 1.109.5, 1.108.2, 1.109.4, 1.107.1, 1.108.1, and 9 more
opencode	849	1.2.24, 1.2.22, 1.2.20, 1.2.10, 1.2.17, 1.2.23, 1.2.6, 1.2.19, 1.2.21, 1.2.16, and 2 more
kiro	658	0.0.0
claude-ai	604	0.1.0

Number of Init Events by MCP Client (IDEs only)



- JetBrains-AI/copilot-intellij
- JetBrains-IC/copilot-intellij
- JetBrains-IU/copilot-intellij
- JetBrains-PV/copilot-intellij
- Roo Code
- Visual Studio Code
- Visual Studio Code - Insiders
- claude-code
- codex-mcp-client
- codex-shell

8°C Cloudy

Search

8:55 AM 3/12/2026

MCP Client / Coding Agent Insights:

Coding Agents exposing OTEL Metrics, Logs or provide an API to pull metrics



```
# 1. Enable telemetry
export CLAUDE_CODE_ENABLE_TELEMETRY=1

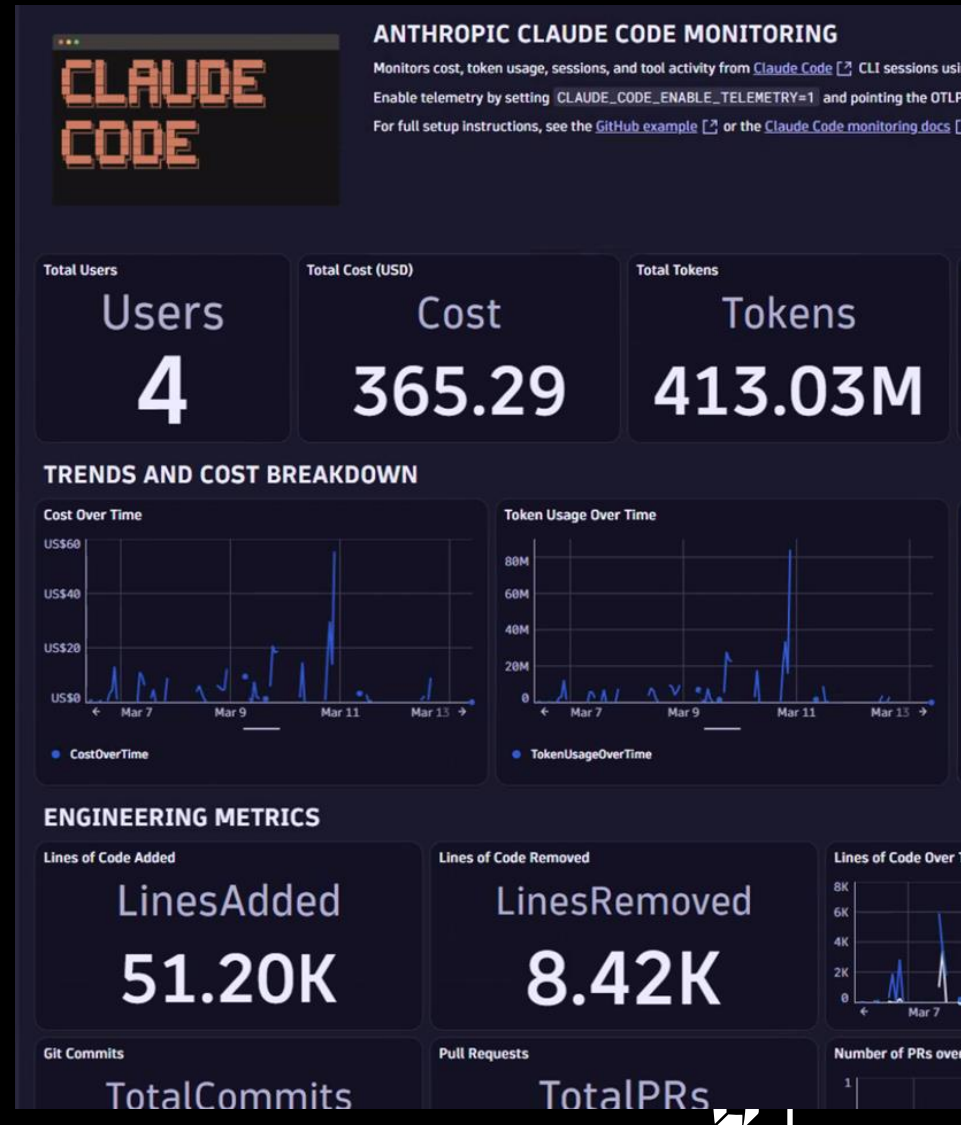
# 2. Choose OTLP exporters (Dynatrace ingests metrics and logs)
export OTEL_METRICS_EXPORTER=otlp
export OTEL_LOGS_EXPORTER=otlp

# 3. Use HTTP/protobuf – the protocol Dynatrace expects
export OTEL_EXPORTER_OTLP_PROTOCOL=http/protobuf

# 4. Point to your OTLP Endpoint
export OTEL_EXPORTER_OTLP_ENDPOINT=https://...

# 5. Add Authentication Header
export OTEL_EXPORTER_OTLP_HEADERS="Authorization=xxx"

# 6. Specify metric export cumulative/delta
export OTEL_EXPORTER_OTLP_METRICS_TEMPORALITY_PREFERENCE=delta
```



Adoption and Impact Analytics at Scale

See trends, tool usage, identify hotspots, ...



```
GET /orgs/{org}/copilot/metrics

{"date": "2024-06-24",
 "total_active_users": 24,
 "total_engaged_users": 20,
 "copilot_ide_code_completions": {
  "total_engaged_users": 20,
  "languages": [
    {
      "name": "python",
      "total_engaged_users": 10
    },
    {
      "name": "ruby",
      "total_engaged_users": 10
    }
  ]
},
 "editors": [
  {
    "name": "vscode",
    "total_engaged_users": 13,
    "models": [
      {
        "name": "default",
        "is_custom_model": false,
        "custom_model_training_date": null,
        "total_engaged_users": 13,
        "languages": [
          {
            "name": "python",
            "total_engaged_users": 6,
```

A large, glowing red lightning bolt strikes down from the top, illuminating the text below. The bolt is jagged and has a bright, fiery core with a darker red, smoky outer glow.

DO WE NEED
— AGENTS —
FOR EVERYTHING?

Detecting bad patterns in observability data is simple!

Duplicated Spans, excessive logs, unexpected dependencies ...

```

1 fetch spans
2 | summarize cnt=count(), by:{service.name, span.name, dt.openpipe}
3 | filter cnt > 1
4 | sort cnt desc
    
```

1,000 records Executed at: 2/5/2026, 09:20:55, Timeframe: 09:15:44 - 09:20:44, Scanned bytes: 20 GB

Duplicated Spans

```

Log Pattern Examples
↓ C... Status Pattern
4,400 ERROR [Error] File: /URLPATH [OTLP TRACE GRPC Exporter] Export() failed with status_code: UNAVAILABLE
error_message: "data refused due to high memory"

2,300 ERROR [Error] File: /URLPATH [OTLP LOG
    
```

Excessive Logs

Automate through a Daily Standup Reminder

Automate and Push the results back to your Pull Request!



PlatformBot APP 11:22 AM

Daily Observability Insights

Your team is currently owning 23 services running across 3 environments.

⚠️ **Your Observability Score is 4.5 (out of 10).**

Here are some things you can improve today!

- ⚠️ New NullPointerException in the payment service
- 🔍 Duplicated Spans in the /api/checkout endpoint
- 🗄️ 10% of logs have no log level
- 🦖 2 SQL Statements that can be optimized

Please [click here](#) to see more details 11:22 AM

Merged V95 v95 into main

```

argoapp/app.yml
@@ -10,7 +10,7 @@ metadata:
10 10     proj_name: "simplenodeservice"
11 11     stage: "preprod"
12 12     owner: "team01"
13 -     version: "94.0.2"
13 +     version: "95.0.2"

manifests/rollout.yml
@@ -24,14 +24,14 @@ spec:
24 24     dt.owner: "team01"
25 25     app.kubernetes.io/name: userinterface
26 26     app.kubernetes.io/part-of: "financial.portal"
27 -     app.kubernetes.io/version: "2.0.2"
27 +     app.kubernetes.io/version: "3.0.2"
28 28     dynatrace-release-stage: "preprod"
29 29     backstage.io/component: "backend.service"
30 30     priority: gold
31 31     spec:
    
```

Administrator merged 5 minutes ago

Administrator mentioned in commit 990b3d71 5 minutes ago

Observability Agent @root · 2 minutes ago
 Dynatrace Site Reliability Guardian evaluation completed for 'backend-service'. Guardian 'Observability Score' completed with a status: FAIL

Overall score	92.0.2	93.0.2	94.0.2	95.0.2
container logs	🟢	🟢	🟢	🟢
pod availability	🟢	🟢	🟢	🟢
service availability	🟢	🟢	🟢	🟢
service performance	🔴	🔴	🔴	🔴
service throughput	🟡	🟡	🟡	🟡
synthetic availability	🟢	🟢	🟢	🟢
version	92.0.2	93.0.2	94.0.2	95.0.2

null	1,190	Since symfony/http-foundation 5.1: Passing a non
null	556	Since symfony/http-foundation 5.1: Retrieving a n
paymentservice	391	PaymentService Fail Feature Flag Enabled
frontend	184	14 UNAVAILABLE: data refused due to high memo



Demo: Observability in your PR-Flow

We don't need to prompt an agent to ask for feedback

The screenshot displays the Backstage application interface. A large blue banner at the top reads "Open Pull Request to update App Version". The interface is divided into several sections:

- Header:** Shows the component name "simplenodeservice-team01-preprod" and the owner "Team Andi".
- Navigation:** A sidebar on the left contains "Home", "APIs", "Docs", and "Create...".
- About Section:**
 - Buttons for "VIEW SOURCE" and "VIEW TECHDOCS".
 - DESCRIPTION:** "Template for the simplenodeservice owned and operated by team01 in environment preprod".
 - OWNER:** Team Andi
 - SYSTEM:** No System
 - TYPE:** website
 - LIFECYCLE:** preprod
 - TAGS:** No Tags
- Relations Section:** A diagram showing "Team Andi" as the ownerOf/ownedBy of "simplenodeservice-team01-preprod".
- Links Section:** Includes "GitLab Repo", "View in Dynatrace", "Browse Application", "Dynatrace community", and "View in ArgoCD".
- Kubernetes Deployments Table:**

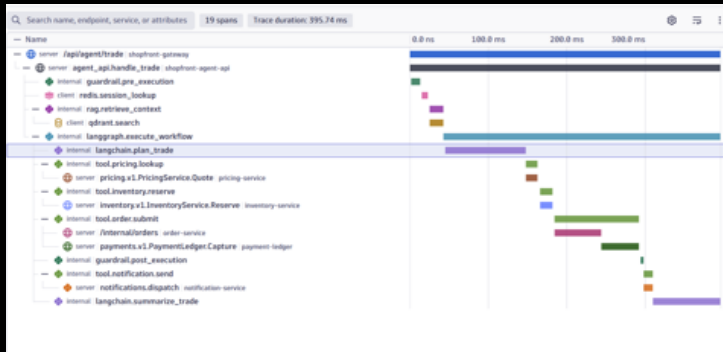
WORKLOAD	CLUSTER	NAMESPACE	PROBLEMS	VULNERABILITIES	LOGS	ENVIRONMENT
simplenodeservice-team01-7694474557	hot-day-platform-engineering	simplenodeservice-team01-preprod	0	0	Show logs	hd34192

At the bottom right, there is a small icon and the number "33".

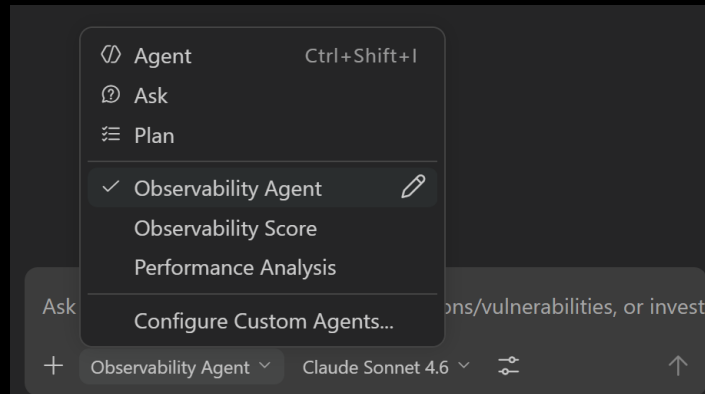
I am done now – so ...

3+1 Things I hope you took away from this today

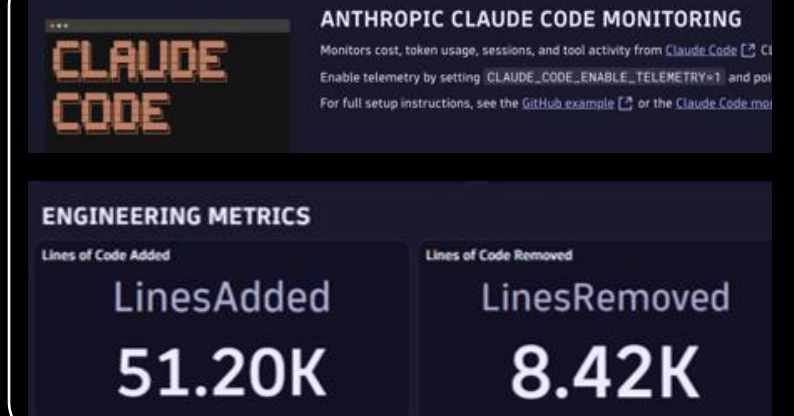
The Basics Observability Standards



The AI Delivery Lifecycle Observability for and with AI



Cost and Impact Optimize usage and alternatives



“Observability beyond apps: AI, Workflows, CI/CD ...”

“Integrate Observability Loops into your AI Development”

“Observability helps you optimize usage and impact!”

Remember

“Some problems can be solved with simple automation!”

Thank You!



Andi (Andreas) Grabner

- CNCF Ambassador
- Fellow DevRel @ Dynatrace
- PurePerformance Podcaster
- Occasional Book Author
- Salsa Dancer

