



Data QA Nightmares: From Broken Dashboards to Observability Maturity

Our dashboards looked great—until they started lying.

After years of chasing mysterious reporting errors and "phantom" bugs, our team learned how to tame data chaos with a new approach.

 by Chaitanya Reddy



Problem Statement



Shiny Dashboards, Shaky Truth

Beautiful visualizations masked serious data quality issues. Users made decisions based on incorrect data.



Hidden Data Issues

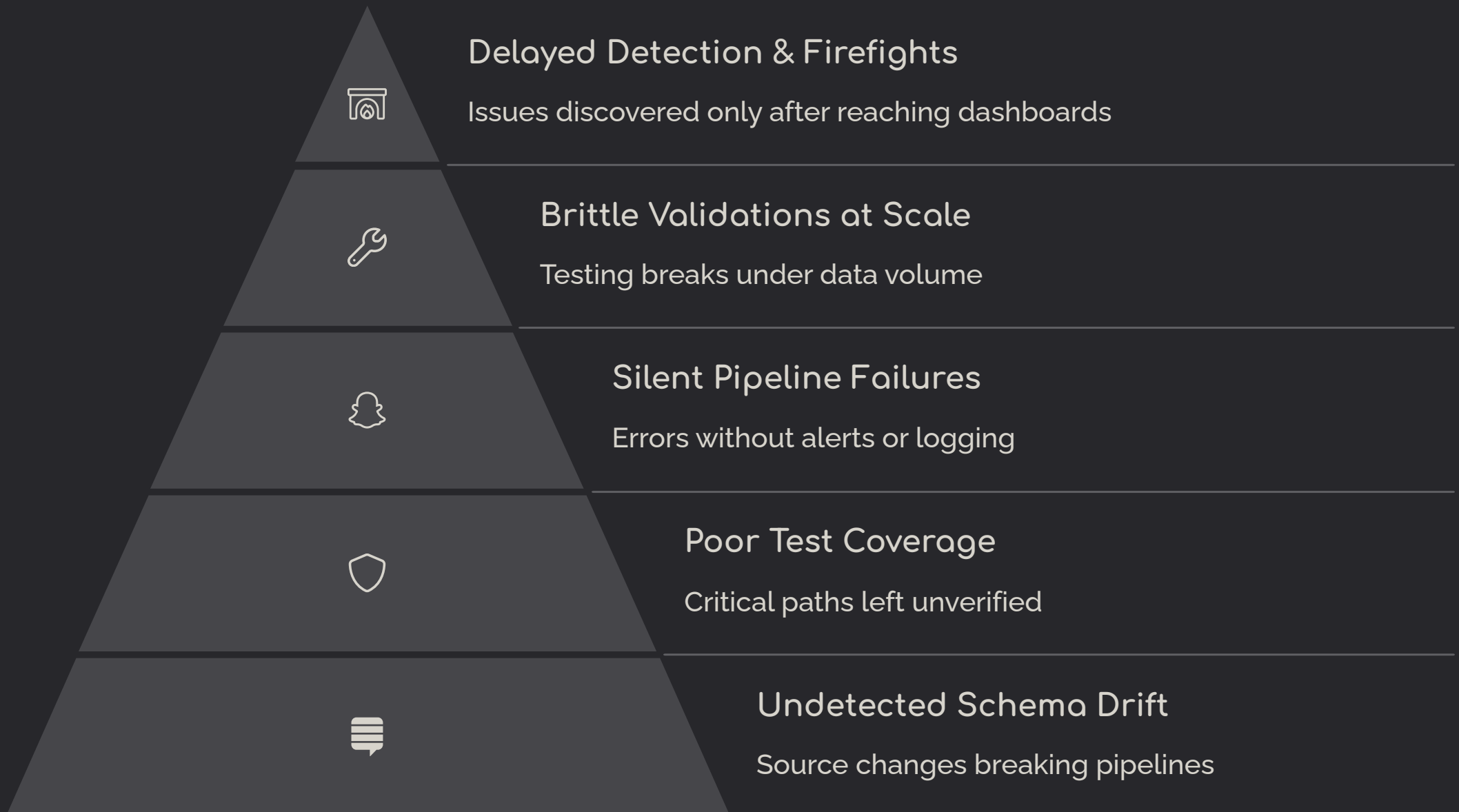
Problems lurked beneath the surface. Schema drift and pipeline failures went undetected for weeks.

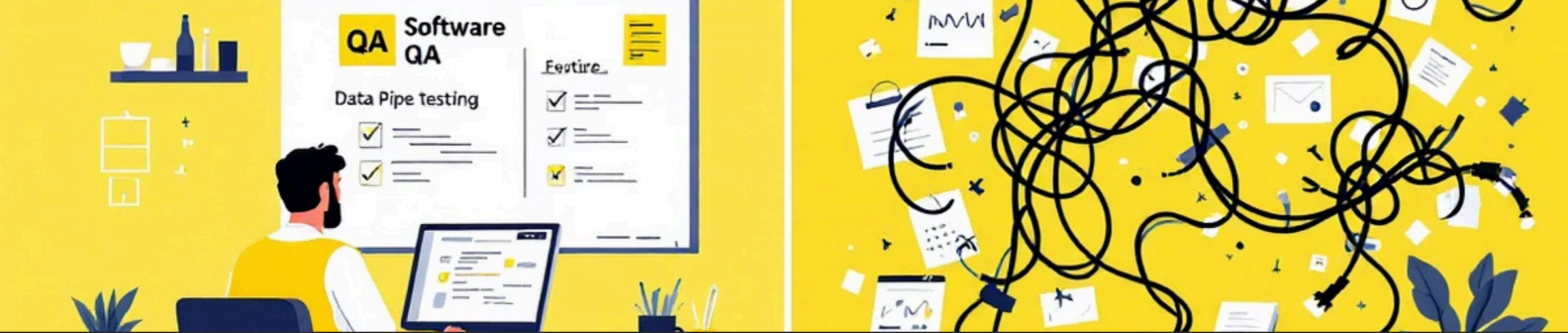


Eroding Trust

Stakeholders lost confidence in our data. Each error diminished credibility and raised questions.

Real Production Pain Points





Why Traditional QA Failed

Pipeline != Software?

We treated data pipelines differently than software. This fundamental mistake led to inconsistent quality standards.

Incomplete Testing

Test coverage focused on transformation logic. We missed data quality, completeness, and consistency checks.

Not Built for Scale

Manual checks worked for gigabytes. They collapsed under terabytes and petabytes of diverse data sources.

Siloed QA Processes

Testing happened in isolation. No one owned end-to-end quality across the entire data lifecycle.

Pivot to Observability

Lightbulb Moment

We realized traditional testing wasn't enough. We needed continuous visibility into our data systems.

Just as physical pipelines have gauges and sensors, data pipelines need monitoring at every stage.

What is Observability?

- Ability to measure internal state
- Real-time data quality metrics
- Complete visibility into pipelines
- Proactive issue detection

New Approach

- Treating data as code
- Embedded quality checks
- Continuous monitoring
- Automated alerting

Key Principles of Observability in Data QA

Instrument Everything

Add logs, metrics, and traces throughout data pipelines.

Nothing moves without being measured.

Fail Fast, Fail Safe

Catch issues early in the pipeline.

Prevent bad data from propagating.

Data Lineage

Track data flow and impact analysis.

Understand dependencies and ripple effects.

Data Quality Checks as Code

Version-controlled, automated tests that evolve with your data.

Repeatable, consistent validation.

Continuous Monitoring

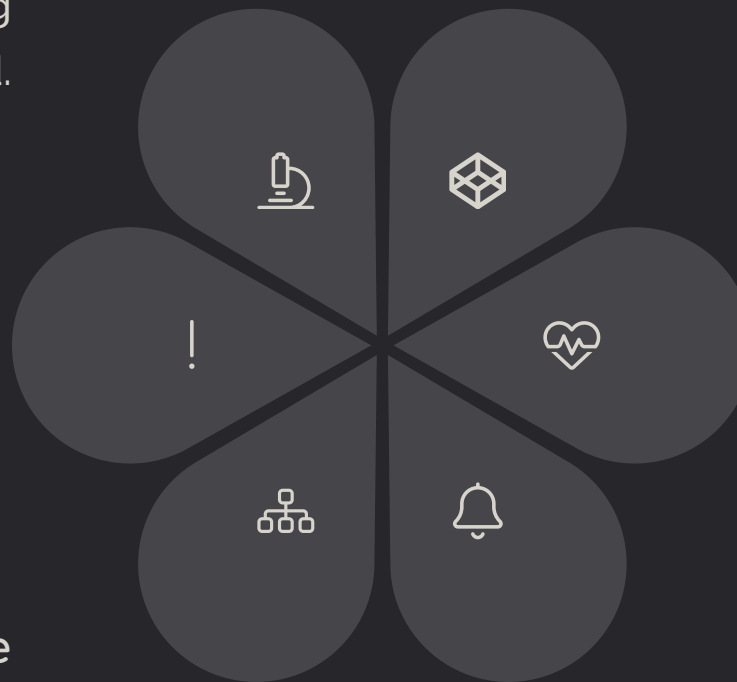
Real-time health metrics for all data assets.

Dashboards showing pipeline vitals.

Alerting & Response

Immediate notifications for anomalies.

Defined incident management process.





Tools and Techniques Used



Great Expectations (GX)

Data validation framework for implementing quality checks. Defines "expectations" for how data should look.



dbt (Data Build Tool)

Transformation framework with built-in testing. Allows embedding tests directly in models.



OpenTelemetry (OTel)

Standard for collecting metrics, traces, and logs. Provides visibility across distributed systems.



Prometheus/Grafana

Monitoring stack for visualization. Real-time dashboards showing pipeline health.

Architecture of the New QA Process

Data Sources & Ingestion

Source validation checks. Schema conformance testing. Volume and frequency monitoring.

Staging & Validation

Data quality rules execution. Anomaly detection. Reference data validation.

Transformation & Lineage

Business rule verification. Transformation validation. Lineage tracking for impact analysis.

Monitoring & Alerting

Centralized observability. Real-time alerts. Historical performance tracking.

BI Layer & Dashboards

Final validation checks. Confidence scoring. End-user feedback collection.

Lessons Learned and Best Practices



Trust But Verify

Even reliable systems need validation. Never assume data quality without verification.



Observability is a Team Sport

Involve engineers, analysts, and business users. Quality requires cross-functional collaboration.



Start with Your Pain Points

Focus on most critical issues first. Address areas causing business impact.



Automate and Standardize

Manual checks don't scale. Build repeatable processes and standardized tests.



Treat Data Incidents Like Software Incidents

Apply rigorous incident management. Document root causes and preventive measures.

Call to Action



Assess Your Data QA Today

Evaluate current observability gaps



Embed Observability

Start instrumenting critical pipelines



Leverage Available Tools

Use open-source solutions to begin



Make It a Team Effort

Build a data quality culture

Don't settle for unknowns in your data. Embrace observability to transform from data nightmares to reliable insights.