MONTE CARLO

The Big Book of Data Observability

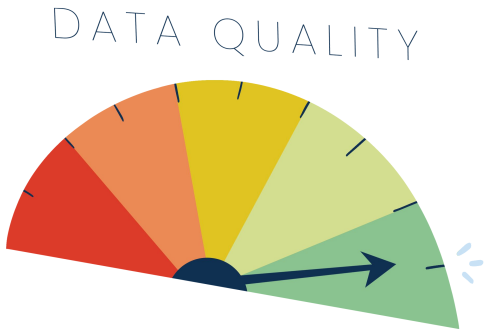
Introduction: The Rise of Data Downtime

More data means more insight into your business. But more data also introduces a heightened risk of bad data breaking your otherwise perfectly good pipelines. When that happens, your business suffers from the consequences of what we call **data downtime**—moments when your data is missing, stale, erroneous, or otherwise compromised.

And the consequences of data downtime can be dire, including wasted time, ill-informed decision-making, missed revenue, and loss of customer trust. Research suggests that companies spend up to 40 percent of their time tackling data downtime, and 1 in 5 companies have lost customers due to data quality issues.

Fortunately, over the last few years, a new approach to data reliability and trust has emerged to address it: **data observability**.

In this ebook, we'll go over the challenges that poor data quality presents, what data observability is and how it can help, best practices for improving data reliability, and how to implement observability tooling within your data tech stack.



Challenge: Complex Pipelines & Constant Change

Modern data environments are growing in complexity and evolving constantly. With the widespread adoption of cloud services like Snowflake and Databricks, the data tech stack now involves more pipelines, more tables, and more points of failure than ever before.

The old data quality adage of "*garbage in, garbage out*" still holds true, but is no longer sufficient. Data issues can occur at any stage of the pipeline—not just ingestion—as data flows between storage and processing to transformation and modeling to BI tools. Teams can be blindsided by unexpected schema changes or broken APIs that lead to inaccurate or stale data, and these issues can go undetected for days—until a panicked Slack message arrives just before an important meeting, asking why a report is all wrong.

The **constant change** happening within modern data platforms only adds more unpredictability. The data that flows through your pipelines is dynamic, especially if you're managing unstructured data in a lake or streaming data in real time. And as your tech stack evolves, keeping documentation and data catalogs up-to-date becomes nearly impossible.

Without clear visibility into upstream data pipelines or field-level anomalies, troubleshooting data issues in a complex, dynamic environment becomes increasingly burdensome.

CASE STUDY: THE FARMER'S DOG

As fresh dog food company The Farmer's Dog built out a complex data tech stack that encompassed AWS, several Postgres databases, Google Cloud Platform, BigQuery, ETL tools, Segment, Kustomer, and Looker, data challenges increased in tandem. "If you own data pipelines, you're extremely familiar with this problem," said Rick Saporta, Head of Data Strategy and Insights. "And it's not a problem most people look forward to tackling."

LEARN MORE: How The Farmer's Dog Builds Data Reliability

Challenge: High-Growth Teams & Lack of Communication

Along with tech stacks, modern data teams are growing in size and complexity as well. This means more people are working with your data, making changes that may impact data quality.

For example, today's data teams often include data engineers, data scientists, data analysts, and analytics engineers. And as those teams mature, they may adopt decentralized organizational models that put the ownership of analytics functions within specific departments. This can lead to confusion on where responsibilities lie, inconsistent standards, and a lack of perspective on how data impacts the entire business. The potential for complication only increases for advanced teams that are adopting the **data mesh framework**, which distributes ownership of data and federates data governance altogether.

Decentralized teams can be incredibly effective—if communication is clean and orderly. But when data team members lack an understanding of how their work fits into the bigger puzzle, seemingly minor decisions can cause outsize impacts. Making simple schema changes or adding new data sources without proper communication can lead to broken pipelines, compromised tables, and more instances of data downtime.

Daniel Rimon
Head of Data Engineering

RESIDENT

CASE STUDY: RESIDENT

At direct-to-consumer mattress brand Resident, data quality issues prevented stakeholders from accessing the reliable insights they needed to make decisions. This led to difficult conversations among the decentralized data team. "It negatively impacted relationships between business units," said Daniel Rimon, Head of Data Engineering. "For example, if you're in data engineering, it can really strain relationships with a BI or analytics team."

LEARN MORE: How Resident Reduced Data Issues by 90%

Challenge: Testing Gaps & Reactive Teams

Traditionally, data teams have used testing to detect and prevent potential data quality issues. But today, companies ingest so much data and maintain such complex pipelines that **testing alone cannot solve for data reliability**.

Schema tests or manual thresholds can identify specific, familiar problems so that when new data or new code goes against your predictions, you can find out why and resolve the problem. But achieving good testing coverage in a complex data system is a big challenge, and testing improvements often get deprioritized against more timely requests.

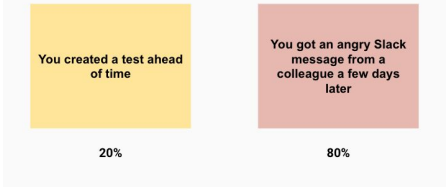
And even robust testing can't detect unknown unknowns. These are data issues that can't be easily predicted, such as silent schema changes, failed jobs, or broken integrations. Often, these incidents fly under the radar until a downstream data consumer notices wonky numbers or missing data—and sends the data engineering team into fire-drill mode.

When testing falls short, teams have to get reactive. Our research shows that teams spend up to 40 percent of their time firefighting or responding to urgent requests to fix data quality—hours that could be spent innovating, building new products, or otherwise adding value to the business.

CASE STUDY: hotjar

Product experience insights company Hotjar used to rely on dbt for testing and transforming their data before it entered their business intelligence layer. But this left a sizable gap in alerting around pipeline delays.

LEARN MORE: How Hotjar Reduced Data Infrastructure Costs by 3x



Challenge: More Stakeholders & Sloppy Workarounds

Data democratization is a good thing—increasing direct, self-serve access to data for your business partners helps speed up decision-making and instill a culture of data within your organization.

But, without careful planning and consistent **data governance**, democratizing access to data can lead to unforeseen issues.

When you have more stakeholders with varying degrees of technical expertise handling and relying on data, and they will be constantly tapping you on the shoulder when something goes wrong. And if bottlenecks occur, you may find more fragile Excel sheets or one-off workarounds springing up—or, in extreme cases, bypassing you entirely.

CASE STUDY: Clearcover

Insurance provider Clearcover had a one-man data engineering team responsible for overseeing a complex data stack—and bottlenecks inevitably started to happen. As stakeholders grew frustrated by delays, many found workarounds by simply querying the source data directly.

“All this investment that I was making in this data stack was for naught,” said Braun Reyes, Senior Manager of Data Engineering.

LEARN MORE: How Clearcover Increased Quality Coverage by 70%

Challenge: Losing Organizational Trust in Data

Every year, companies are investing more money in data and touting their status as data-driven organizations. In 2021, 99% of Fortune 1000 companies reported investing in data and AI.

But when data downtime occurs, trust in data erodes. Business leaders spend more time debating about data versus acting on it, and the old habits of relying on gut instinct or intuition can start to creep back in.

This makes sense: if business leaders can't rely on their reports to be accurate, or lose revenue due to decisions made because of bad data, their skepticism is warranted. And even if data teams address these problems and implement better tooling and processes to protect data health, the damage is already done and trust is difficult to rebuild.

CASE STUDY: Blinkist

Ebook subscription service Blinkist relies on paid performance marketing to fuel customer acquisition. And when the pandemic hit in 2020, C-level executives and campaign managers grew increasingly dependent on real-time insights to drive marketing strategy, budget spend, and ROI.

At the same time, the data engineering team was struggling with data downtime. Dashboards weren't updated in a timely manner, and pipelines were breaking.

"Every Monday, we had executive calls," said Gopi Krishnamurthy, Director of Engineering. "And almost every Monday, I was trying to answer why we are not able to scale, what were the issues, how many problems we face in terms of tracking data…trying to explain the severity of the problem and trying to boost confidence with executive stakeholders."

Gopi's team was spending 50% of their working hours firefighting data drills while trying to rebuild data trust within the organization.

LEARN MORE: How Blinkist Prevents Broken Pipelines at Scale

The Benefits of Data Observability

All of these data quality challenges—complex pipelines, poor communication, testing gaps, demanding stakeholders, and loss of trust—are significant. But organizations aren't doomed to data downtime. The new approach of **data observability** is helping real teams solve real problems every day.

Data observability is an organization's ability to fully understand the health of the data in their systems. Its goal is to reduce the frequency and impact of data downtime.

Drawing on the best practices of DevOps and Site Reliability Engineering, data observability centers on measuring and addressing across key pillars that describe high-quality data.

The five pillars of data observability are:

- **Freshness**: how up-to-date is your data?
- **Distribution**: does your data fall within an accepted range?
- **Volume**: is your data complete?
- **Schema**: has the structure of your data changed?
- **Lineage**: what are the upstream and downstream impacts of data downtime?

DATA OBSERVABILITY PILLARS

Freshness | Distribution | Volume | Schema | Lineage

OUTCOMES: The Impact of Data Observability

Data observability helped real teams solve real problems:

- The Farmer's Dog **improved reliability** across complex pipelines
- Resident **reduced data issues** by **90%**
- Hotjar **addressed testing gaps** and got proactive about data quality
- Clearcover **increased quality coverage** for ELT by **70%**
- Blinkist **rebuilt data trust** and saved **120 hours/week**

Data Observability: How It Works

Data observability reduces data downtime by layering **monitoring and alerting** for the five pillars across your entire data stack, from ingestion to transformation to analysis. Observability tooling also provides **end-to-end data lineage**, or mapping of upstream and downstream dependencies.

- **Monitoring and alerting reduces time-to-detection**
  Data observability tooling provides automated monitoring and intelligent alerting when data issues are detected. This enables your team to respond swiftly, reducing time-to-detection and turning panicked Slack messages you receive from stakeholders into proactive updates sent by your data team.

- **Data lineage speeds up time-to-resolution and enables proactive communication**
  With the lineage provided by data observability tooling, teams can visualize every possible dependency when data downtime occurs, from upstream ingestors to downstream reports—even to the field- and table-level. This allows your team to conduct swift root cause analysis, as well as communicate any potential impacts to end data consumers. Bottom line: data downtime is resolved more quickly with less negative impact on the business.

CASE STUDY: choozle

Digital advertising software provider Choozle surfaces campaign performance data from multiple ad platforms to small and medium-sized businesses—so Choozle adopted data observability tooling to ensure their complex pipelines provide data that is accurate and reliable.

“Without a tool like this, we might have monitoring coverage on final resulting tables, but that can hide a lot of issues,” said Adam Woods, Chief Technology Officer. “Now, we’re at a level where we don’t have to compromise. We can have alerting on all of 3,500 tables. And we see two to three real incidents every week of varying severity. Those issues are resolved in an hour, whereas before, it might take a full day.”

LEARN MORE: How Choozle Reduced Data Downtime by 88%

Data Observability: How It Helps Your Business

Improving the ROI of Data Quality
Bad data carries a real cost—so reducing data downtime means tangible savings. Think about the labor costs involved in data downtime: when data breaks, engineers have to spend valuable hours troubleshooting and fixing what went wrong. At most organizations we talk to, data engineers can spend up to 40 percent of their time responding to data downtime. (And these in-demand positions tend to be well-paid, so those hours aren't cheap.)

Learn how to calculate the full cost of data downtime with our ROI framework.

Enabling Better Data Discovery
The automated table- and field-level lineage provided by data observability helps surface information and draw connections between data assets. This can be used to help teams move from static data catalogs to modern data discovery. Data discovery replaces the need for a data catalog by providing a domain-specific, dynamic understanding of your data based on how it's being ingested, stored, aggregated, and used by a set of specific consumers.

Quantify Data Team Performance
Data observability tools make it simple for teams to set and measure data SLAs. These attributes of data reliability, as agreed-upon across teams and stakeholders, typically include metrics like time-to-detection, time-to-resolution, and the number of data incidents for a particular asset. Tracking these numbers helps manage stakeholder expectations and enables data leaders to quantify and demonstrate the value their teams are providing to their organization and customers.

(TTD hours + TTR hours)
*
downtime hourly cost
=
cost of data downtime

CASE STUDY: RED | VENTURES

Red Digital, an advertising agency under the Red Ventures umbrella, implemented data SLAs to understand how their data systems were performing. In the process of creating SLAs, the data team gained a more holistic perspective of how business teams were using data and interacting with key assets—and the rest of the company could see real numbers about how the data team was performing. "Having a record also evolved the evaluations of our data team from 'I feel the team is/isn't doing well' to something more evidence-based," said Brandan Beidel, Senior Data Scientist.

LEARN MORE: One SLA at a Time: Our Data Quality Journey at Red Digital

Data Observability: How to Get Started

When you introduce any new element into your data platform, it pays to adhere to best practices—and observability is no exception. As you adopt observability tooling, mind the following principles:

Organize The Team And Get Buy-In
Before adding an observability layer to your data platform, it's critical to get all the teams that might interact with the tooling on board. Employees in every division across the organization should understand how observability will ultimately provide value to *them*. That's the initial job of the data team: to explain and showcase that value, and to establish a method of measuring success even as the company scales.

Build A Comprehensive Data Reliability Workflow
The modern data reliability layer is made up of four distinct layers: testing, continuous integration (CI) / continuous delivery (CD), data observability, and data discovery. Each represents a different step in your company's data quality journey, so don't neglect any of them. Learn more about building a reliability stack here.

Treat Your Data Platform Like A Product
Data platforms, including observability layers, should be treated as a product—not a means to an end. A few tips include aligning your observability goals with the goals of the business, incorporate observability into your data platform roadmap, and set data SLAs to measure the effectiveness of your observability tooling. Learn more about treating your data platform like a product here.

RESOURCE: The Data Observability Checklist

To achieve the goal of reducing data downtime, your data observability layer should include, at a minimum:

- End-to-end visibility across ingestion/transformation/analysis
- Seamless connections to your data stack
- Anomaly detection based on historical data and patterns
- Allow team members to create custom thresholds
- Proactive alerting
- Intelligent alert routing based on dataset owners
- Rich context for root cause analysis, triage, & troubleshooting, down to the field level
- Comprehensive query logs
- Dynamic data catalog creation
- Automated data lineage creation
- Monitoring data at rest

LEARN MORE: The Ultimate Data Observability Checklist

Data Observability: Build or Buy?

Most data stacks combine custom-built, SaaS, and open-source solutions. Data observability is still an emerging technology, but the usual question of "**build or buy?**" applies. Here's what to consider:

Understand the total cost and time-to-value
You can manually check dashboards, regularly query Snowflake, or even build a custom SQL integrity checker—but all of these activities are time-consuming. Estimates show that the resources required to build a custom data observability solution could add up to half a million dollars, and take up to 24 weeks to implement.

Factor in opportunity cost
Unless you have the rare problem of employing too many data engineers, you'll have to redirect team members from solving customer problems and improving your product to manually building data tests to account for possible edge cases.

Adam Woods, Chief Technology Officer, Choozle

"I understand the instinct to turn to open-source, but I actually have a lower cost of ownership with a tool like Monte Carlo because the management burden is so low and the ecosystem works so well together. After one phone call with the Monte Carlo team, we were connected to our data warehouse, and we had data observability a week later. We are able to reinvest the time developers and database analysts would have spent worrying about updates and infrastructure into building exceptional customer experiences."

Data Observability: The Cost of Custom Builds

Build

Resources

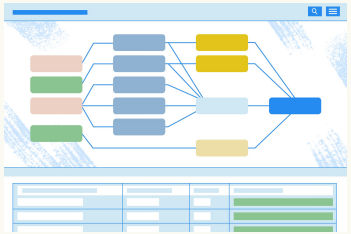
- 3 data engineers to scope & build over 9 months
- .5 data analyst to scope over 3 months
- .25 data engineers to maintain once live

Cost

$506,125

Time to Value

24 weeks



4 Key Considerations:

1. Understanding the expected time-to-value for data observability solution
2. Factoring in the opportunity cost of building a data observability solution
3. Taking a proactive approach to solving data quality problems
4. Scoping our data observability solutions

Resource	Monthly cost	Number of resources engaged	Number of months engaged	Total cost
Data Engineer *(build)*	$17,500*	3	9	$472,500
Data Analyst	$14,000**	.5	3	$21,000
Data Engineer *(maintain)*	$17,500*	.25	3	$13,125
				$506,125

*Data Engineer fully loaded annual cost: $210,000
~$140,000 total pay + additional 50% in benefits + taxes = $210,000
Source: https://www.glassdoor.com/Salaries/san-francisco-data-engineer-salary-SRCH_IL.0,13_IM759_KO14,27.htm

**Data Analyst fully loaded annual cost: $168,000
***$112,000 total pay + additional 50% in benefits + taxes = $168,000*
Source: https://www.glassdoor.com/Salaries/san-francisco-data-analytics-salary-SRCH_IL.0,13_IM759_KO14,28.htm

Data Observability: Evaluating Third-Party Solutions

Investing in an out-of-the-box data observability solution is worth careful consideration. Take time to compare your options.

SOLUTION CRITERIA

Data observability solutions should provide:

- **Unified platform**: All stakeholders are able to collaborate in a single, self-serve platform
- **Rapid, ML-based detection**: Automation enables data lineage, incident prioritization, and intelligent alerts
- **End-to-end visibility**: Rather than a point solution, observability should be adaptive to evolving needs as data flows through lakes, warehouses, and BI tools.
- **Security-first architecture**: Compliance, security, and residency requirements should be supported, and data exposure should be minimized.

COMPANY CRITERIA

Don’t just evaluate the software—consider the organization behind the product. Look for:

- **Category leader**: In an emerging category, you want to bet on the leader—because others will get acquired or stumble on their way
- **Enterprise-readiness**: The business should be able to support the scale and complexity of enterprise customers, including support, training, and customer success
- **Pace of innovation**: Teams that are developing and adding value to their platform, including integrations, will continue to deliver results over time

Data Observability: Start Trusting Your Data

Data observability helps teams build trust in data by eliminating data downtime and increasing data reliability. And with the Monte Carlo platform, your team could start reaping the benefits of observability immediately.

Looking for more insights? Dive into our recommended resources below:

- The Data Downtime Blog
- Data Observability Case Studies
- The Modern Data Leader's Playbook

Learn more about how Monte Carlo can help eliminate data downtime at your company.

Request a Demo