REACTOR

Your Permanent Record:

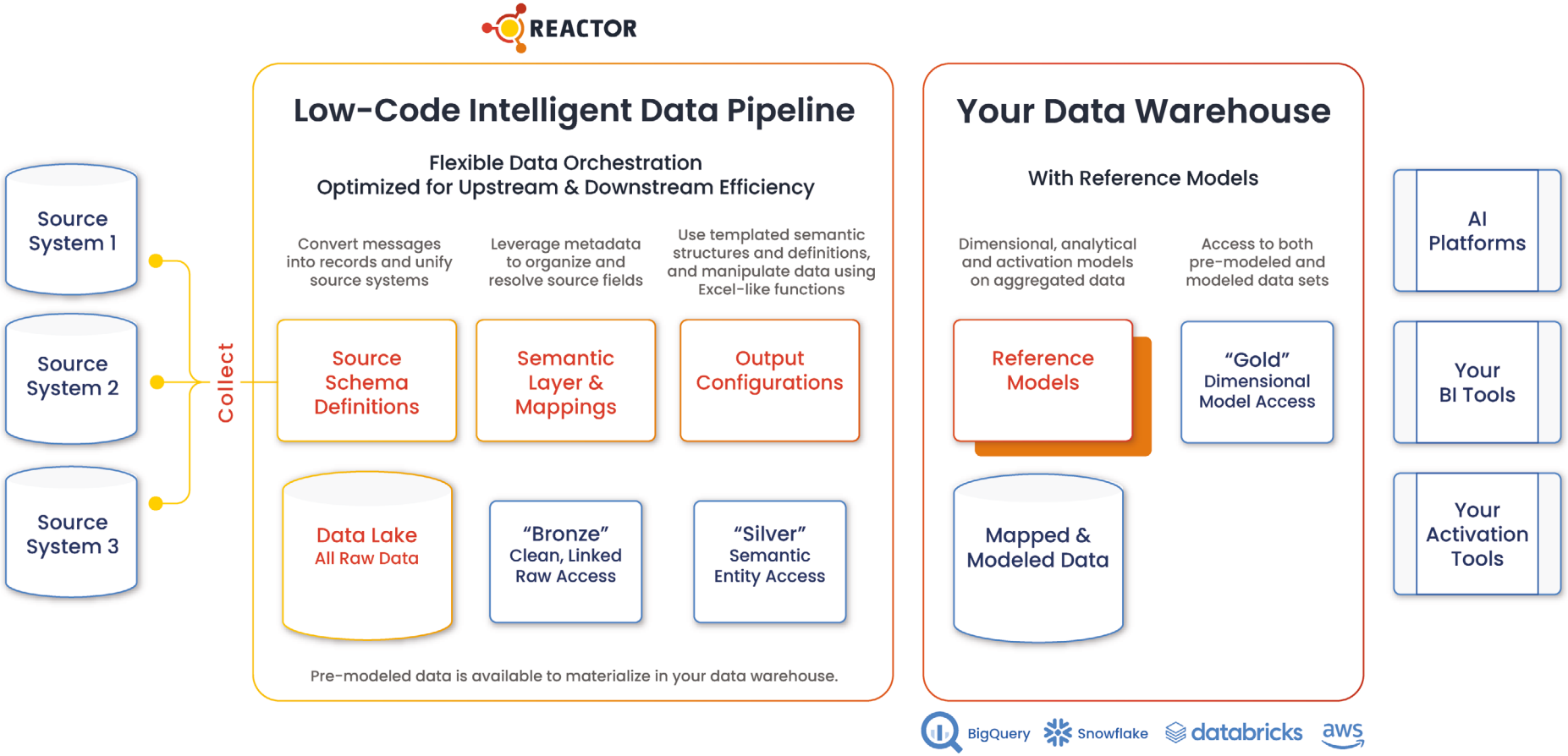
Immutably Log Raw Data to Deliver Major Benefits for your AI and Data Teams

Jared Stiff, CTO, Co-Founder
Rachel Workman, VP Value Engineering
Eric Best, CEO, Co-Founder

Introduction: The cloud changes everything

Cloud data warehouses can transform the way you run your business, revealing the drivers and detractors of profitable growth. But cloud data warehouses can also become expensive dumping grounds for unusable data.

A useful and cost effective data infrastructure requires more than just a data warehouse filled with raw data, dependent upon brute-force data engineering to map and model data into useful business output.



In arguably the most iconic scene from **Bladerunner**, replicant Roy Batty describes his personal memories as **"lost in time, like tears in rain."** Until immortality is invented, we'll have to settle for solving the same problem in data enablement.

Actionable data lost to time. How are we still talking about this? With incredible advances in data storage and processing, cloud-native solutions supporting streaming ingestion, and convergence of the data warehouse and data lake into the big-data-analytics-ready lakehouse, how can this possibly still be a challenge?

One way to look at it is that we are going against nature itself. Our very brains rarely try to store information for which they haven't already identified some specific purpose, and when they do, our ability to use information in a meaningful way degrades, so perhaps we're asking to eat our cake and have it too?

For actionable data not to be lost to time, we need reasonable ways to ingest and store data as it is generated, regardless of whether or not we've defined a specific purpose for it yet, and we need to store it in such a way that:

- Allows for fault-tolerant ingestion
- Stores all the data (full history)
- Leaves the data highly representative of its source
- Keeps the data immutable and auditable
- Makes the data accessible in the context of when it was ingested

Technology advances are making all of the above possible, yet going against nature doesn't come cheap or easy. There are a few ways to go about getting this done, but one way is rising to the top as the most cost-efficient, flexible, and forward-thinking solution.

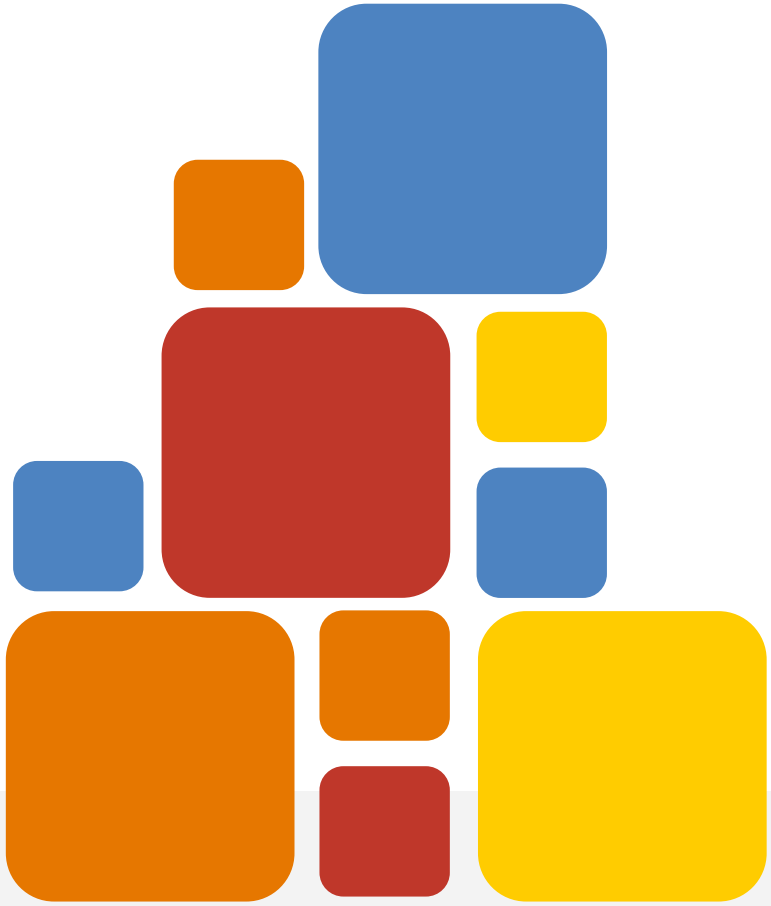
Challenges Posed by ETL

The various macro building blocks for this solution are not new. A data warehouse, with its structured, business-consumable data marts, is excellent for supporting business intelligence and reporting needs. A data lake, with its ability to ingest and store vast amounts of raw data, is cost-effective for storing data ingested from a source that hasn't yet been earmarked for a business purpose as well as for storing data in raw form for experts to access for AI/ML and big data analytics. And those two constructs, as noted earlier, have already in many cases merged into a single solution: the data lakehouse. Most cloud-native data warehouses support big data, and most cloud-native data lakes support business intelligence.

Yet we are still left with two key challenges:

Storing data in a way that makes it useable now or in the future is difficult and costly

Storing data only at rest restricts future flexibility to support real-time use cases



The first challenge goes back to our need to defy nature itself. **To make data usable for business consumers, it needs to be structured**, which is why the data warehouse with its business-specific data marts persists in its key business role. To structure data, we must make decisions about purpose. Data without purpose doesn't have a place in structure.

Prior to ELT becoming a viable alternative to ETL, the decision of what data had purpose occurred when retrieving data from the source, as only data with purpose was retrieved, transformed, and stored. However, in many cases, ETL-based data pipelines were built with at best little thought to future reusability and at worst ad hoc design to satisfy isolated functional needs. Notably fragile in many cases due to cascading effects of batch processing, it was not uncommon for some data pipelines within one organization to be wholly or partly redundant, apply varying transformations for derived values, or have pipes that had been abandoned without any official decommission as business needs continued to evolve. Data not retrieved from source was lost in time as sources changed or purged data of a certain age.

The rise of the data lake and its ability to accept data without structure permitted us to switch around the order from ETL to ELT and load directly into the data lake prior to any transformation. Data at rest in the data lake could then be transformed at will as business needs arose. As all available data could be pulled from the source, in theory, no data was lost. Unfortunately, as often happens, theory translated into practice gets messy, and the explosion of data landing in the data lake in raw form left many organizations with data that was not able to be audited for compliance and also not able to be accessed in the context of its source and origination time when a future need for it arose. This left many organizations with what came to be known as virtually unusable data swamps.

To make that data usable, many organizations evolved by implementing the aforementioned lakehouse and their various modeling overlays, such as Delta Lake or Data Vault, applying light but necessary structure, at least enough to ensure auditable compliance requirements (e.g., ACID) and an ability to access data in the context of its source and origination time when a future need for it arose. Additional modeling, however, even when applied to raw data, comes with overhead and cost. Expert resources are needed to carefully construct these complex models, and the larger teams, including business subject matter experts, need to be trained in complex concepts since even data ingested into a data vault still requires additional modeling to make it business consumable, and centralized data teams cannot be expected to understand all domain data to the point needed to make it usable.

So, with a data vault (or similarly) modeled data lake (house), we are at least at a point where we don't lose any data to time, though with a tradeoff of greater cost. We still have our second challenge, however, and that is data stored at rest in a lakehouse can't enable real-time use cases.

Is there a way to resolve both the greater cost and inability to flex to real-time usage by taking one more step forward in modern data architecture?

Dump the Monolithic Architectures, Adopt ETL

That answer, of course, is yes. Just as continually changing business needs required us to find a way to make sure all data lake data was auditable and usable no matter when accessed, the continually increasing need for real-time data demands we move away from monolithic architectures that don't support real-time use cases.

Wait! Move away? Many organizations haven't even been able to establish lakehouses yet, let alone move on to the next thing. The good news is that, in this case, moving away from a monolithic architecture is additive. The data warehouse/data lake/data lakehouse is still very relevant because data at rest is needed to enable reporting, BI, AI/ML, and big data analytics. However, making an architectural change to add the ability to also process data in motion (not just use streaming for ingestion), can:

- Reduce overall storage costs
- Reduce/eliminate the need for modeling overlays for raw data
- Enable real-time use cases
- Maintain high data fidelity and audit-ability
- Enable full data replay

How does this happen?

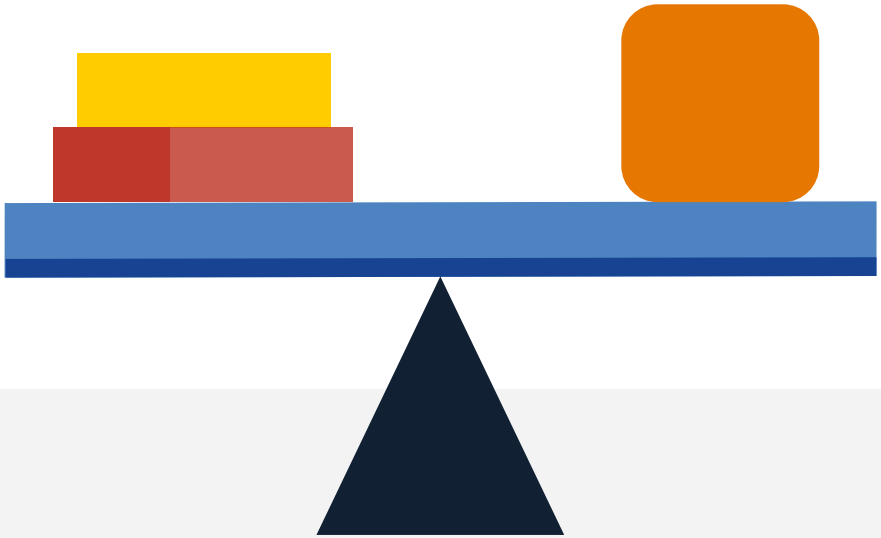
Well, adding a technology like Apache Kafka as a core component to your modern data architecture gives you more flexibility in how you "land" data. Data no longer has to be landed in totality into raw data zones with modeling overlays applied in order to store "all the data" in a usable and governable format. Kafka logs are natively immutable and shippable. All logged data persists with high fidelity in cost-efficient, replayable storage. Because processing data in motion is its core capability, only data with an identified purpose needs to be placed in semi-structured/governed storage for further processing, reducing overall complexity along with the storage cost. Having the data accessed and processed in motion opens up the new capability to interact with real-time systems and use streaming ETL, even while not restricting existing interactions with batch-based systems and storage.

Further, for those organizations that haven't yet made the move to implement lakehouses or other raw-data modeling, making a move to this type of architecture could alleviate the need for this heavy lift while remaining overall cheaper and more in line with evolving architectural concepts like domain-driven design and data mesh. For those who are even earlier in their journeys, taking this direction can avoid the pain experienced by other organizations but that resulted in the key learnings that led us to this point.

Make it Immutable

Organizations can avoid losing important data to history and entropy by leveraging modern cloud infrastructure and the right enterprise architectures.

Until humans are directly linked to, or even part of, the cloud – perhaps by 2049? – we'll have to settle for the business benefits that come with modern data architecture.



Future Proof your Data Stack with Immutable Logging

You can avoid losing important data to history and entropy by leveraging modern cloud infrastructure and the right enterprise architectures. Make sure your designs incorporate raw event logs and data lakes to get needed data closer to pipelines for faster, more flexible processing and analysis.

Find out more about all **nine characteristics of a Future-Proof Cloud Data infrastructure** in our comprehensive exclusive ebook.

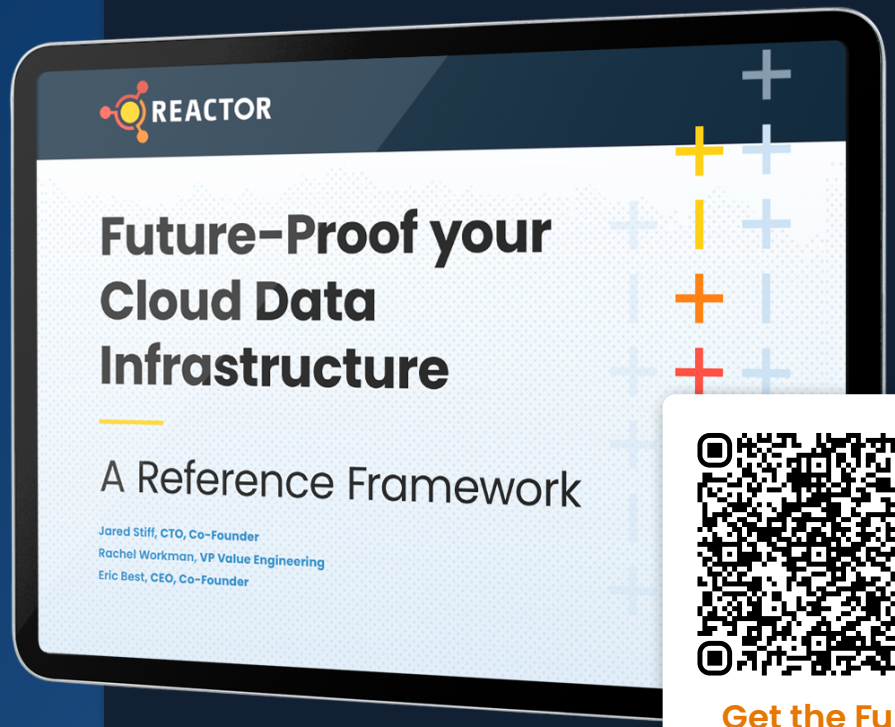
Contact Reactor to learn more and get started today!

REACTOR

Reactor provides the fastest, most efficient path to useful, business-ready data for generative AI, analytics and activation. Built for retailers of any size or complexity, Reactor transforms your unique data infrastructure into an easy-to-use, no-code environment that's accessible to everyone — no engineering degree required. Reactor onboards and ingests data from business critical systems and applications, landing clean, well-defined data modeled directly in your data warehouse.

REACTOR

Put Your Data to Work.



Get the Full E-Book

www.reactordata.com
1-888-417-6863
Grow@reactordata.com