



How to Close the Data Virtualization Gap:

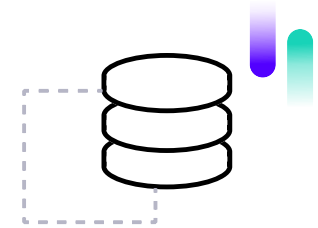
Augmenting AWS Athena to Optimize Performance & Cost

By David Krakov

Data virtualization has become synonymous with reducing data duplication and lowering the cost of both data-ops and devops. It allows administrators to maintain access control and provide audits of data access. By harnessing the power of high-performance query engines, combined with infinitely scalable cloud data lakes, it's apparent why data virtualization is becoming the dominant pattern to give users a way to democratize and analyze all of their data. As with all big data technologies, it's critical to understand why everyone is adopting data virtualization as well as the advantages and common pitfalls.

Data Virtualization

Brings Access and Control to Data Lakes at Lower Cost



Data Lakes have emerged as the standard for big data management with data virtualization becoming the predominant approach for versatile large scale analytics. Data virtualization is achieved by layering a query engine on top of a data lake. Pioneered by early open source data technologies such as Hive on Hadoop, modern day data virtualization rivals the most powerful data warehouse technologies with one critical advantage. Rather than moving data from a data lake out to separate data warehouses for analysis, data virtualization gives users the ultimate flexibility to access a single view of all their data.


Users are embracing data virtualization because they get direct access to the data they need. With less ETL overhead, there's no waiting around for the data teams to load a separate data warehouse, change access control settings, and correct the inevitable mistakes in data transformation. Because query engines rely on a shared catalog, users can better collaborate and easily find the data they need. For data lake administrations, data virtualization means less operational overhead.

Instead of building, running, and maintaining complex ETL pipelines to

move data around to different data warehouses, then synchronizing controls and managing multiple audit trails, data virtualization gives administrators control over all user access to data in one place.

As data virtualization gains popularity, administrators are running into new challenges. The best solutions for virtualization are query engines such as Presto, which is highly versatile and available in hosted native implementations such as AWS Athena. AWS Athena is a devops light implementation of Presto that integrates natively with the rest of the AWS ecosystem, offering significantly reduced operational overhead. Yet at scale, this simplistic approach doesn't give administrators the tools they need to properly and efficiently tune performance or manage costs and resources with an ever-growing user base.

This whitepaper explores the benefits of query engines on top of data lakes compared with a hybrid data lake / data warehouse approach. These benefits are not without their challenges and we discuss solutions, including those developed by Varada, that allow organizations to standardize on a data virtualization architecture.



Presto is the Leading Data Virtualization Query Engine

One of the early query engines developed to support high-performance data virtualization is Presto. A distributed query engine with support for a wide range of data lake platforms, Presto gives data teams the ultimate versatility. Presto delivers the core benefits of data virtualization, with no data duplication, giving administrators centralized access controls, and a shared catalog to make collaboration easier. Among query engines, Presto stands apart from other solutions because of its broad support, deep extensibility, and powerful standard ANSI SQL language.

Any ANSI SQL

Presto supports all of the standard SQL and NoSQL data stores and can read data directly from S3 and HDFS. Users don't need to wait for a data engineer to build an ETL process that transforms raw data into a data warehouse. Presto lets users analyze data as soon as it's made available in the data lake. Using Presto, combined with a shared catalog, means faster and greater analyst collaboration and more organizational value out of data analytics. Presto even supports queries that cross data sources, so raw data can be enriched with data stored in a structured SQL database entirely on the fly.

Built-in Cost-Based Optimizer

Under the covers, Presto processes queries in memory without relying on slow intermediate disk-based storage. The in-memory pipelined distributed execution engine ensures highly efficient query processing. Integrated with the in-memory pipeline processing is a cost based optimizer. Like a data warehouse, the Presto query optimizer evaluates different execution plans for each query and chooses the best one based on data statistics and available resources, reducing overall processing time. The latest version of Presto includes dynamic filtering, which accelerates join performance by reducing the amount of data that queries need to process when joining tables by as much as x10.

Varada's engineers [added an elegant design](#) for dynamic filtering after realizing that the cost based optimizer allowed for an implementation of a broadcast join, pushing the inner join values to filter the larger joined table scan phase.

Powerful Query Engine

All of this processing is performed using a vectorized columnar query engine. Even as Presto is able to read any type of data, all of the in-memory processing is optimized around a columnar architecture, ideal for analytic queries. Combined with data sources that are stored in columnar optimized formats, Presto can optimize query execution by reading only the fields that are required for any individual query. When exposing these advanced query processing capabilities through standard ANSI SQL and JDBC connectors, it's obvious why Presto has become the leading query engine for data virtualization.



The Top Choice for Presto Based Data Virtualization

Amazon Web Services Athena is a critical building block in any modern data lake architecture. As an AWS native implementation of Presto, Athena offers both the Presto query engine's versatility and tight integration with other AWS services. Athena lets data engineering teams create direct access to their AWS data sources such as S3, feed data into other AWS services, as well as share and control ad-hoc access to AWS data. Users can build reporting dashboards against original datasets via Athena's standard Presto endpoint rather than complex and brittle ETL pipelines. Athena also gives administrators centralized control over ad-hoc access to data.

Zero Data-Ops vs. Control

As the leading hosted Presto based query engine for data virtualization, Athena offers a wide range of capabilities. It runs directly on an AWS-based data lake, with no data duplication, data movement, and data consistency issues. As a result, the operational overhead of Athena is incredibly low for basic data virtualization use cases. What drives users to augment Athena, is lack of control over performance and cost. Introducing a data warehouse instantly voids all of these benefits.

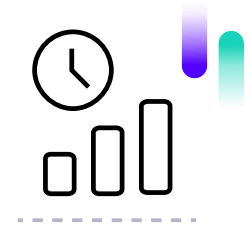
Suddenly, data engineering teams need to deal with data migration, consistency, multiple permission models, and users struggling with finding data across multiple data catalogs.

As Athena administrators know, Athena works exceptionally well out of the box until users run into performance issues. Even though core Presto has powerful tools for optimization, a zero devops solution such as Athena doesn't include any tooling for analyzing performance issues. As a result, when an Athena deployment gains adoption in an organization, users run into roadblocks trying to productionalize the system. For example, some queries incur expensive and time consuming scans, which means users can't reliably power real time dashboards. Users also run into issues getting predictable query times when issuing complicated joins.

Athena administrators find they need domain expertise to understand the data that users are querying and to optimize users' workflow. Administrators struggle to help users because there is no good way to analyze what Athena is doing under the covers. Since data sets and use cases change quickly, any hard earned gains through manual optimization goes out the window. As a result, there are practical limits on how broadly users can adopt Athena for access to data in the data lake. While the benefits of using Athena still outweigh the limitations, these issues are what ultimately lead organizations to abandon the pure data lake strategy and adopt a hybrid approach, duplicating data into a data warehouse. Instead of being the unifying solution promised by data virtualization, Athena and the data lake ends up being relegated to yet another data silo.

Augmenting Athena:

Presto with Price and Performance Optimized Presto



Presto's beauty is that it offers an alternative to throwing in the towel on a 100% data lake solution. Because of Presto's power, it's possible to augment an Athena-based data virtualization architecture with different optimized query engines that run on the same data sets, use the same shared catalog, and abide by the same access controls. By combining Athena with additional Presto query engines, data teams can preserve the core benefits of their data lake: data doesn't move, they maintain centralized access and audits, and provide better collaboration among users through a shared catalog. At the same time, Presto allows teams to preserve the core benefits of Athena with reduced devops and data ops, no need to have deep domain expertise or familiarity with user workflows, and seamless scaling up and down based on demand.

There are a few core capabilities to look for in a Presto query engine that can complement an Athena based solution. These optimizations are transparent to the end user and result in better performance and lower operational costs. Under the covers, an optimized Presto query

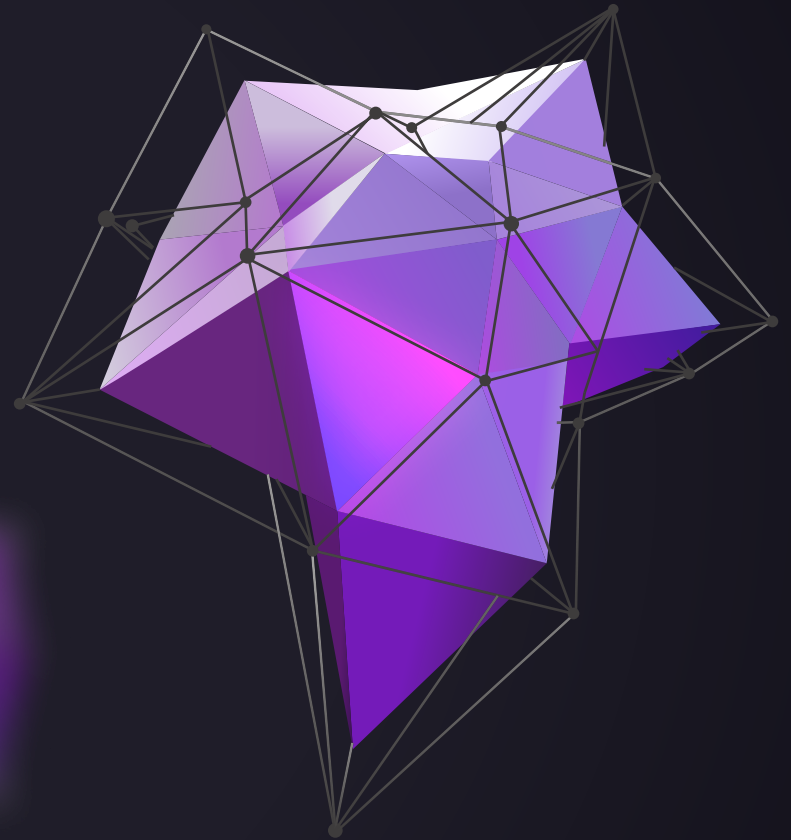
engine includes: dynamic adaptive indexing, to automatically accelerate low performing queries; an unsupervised machine learning system that uses cost based optimization for resource allocation, so that large complex queries can have performance guarantees; and cost reporting and hints so users and administrators can provide business requirements to influence the automated systems.

Even though Presto doesn't offer these improvements out of the box and they're not available in Athena, they can be adapted in open source Presto or added via other Presto based query engines. The downside of using a modified open source approach is that it comes with high maintenance costs: installing, configuring, and updating the query engine becomes an operational burden. Most companies that run open source Presto have dedicated teams managing the Presto codebase for their deployment. Hiring such highly skilled staff is expensive, making the "free" open source solution more costly than an off the shelf solution.



Out-of-the-Box & Self-Optimizing Presto to Accelerate AWS Athena with Zero Data-Ops

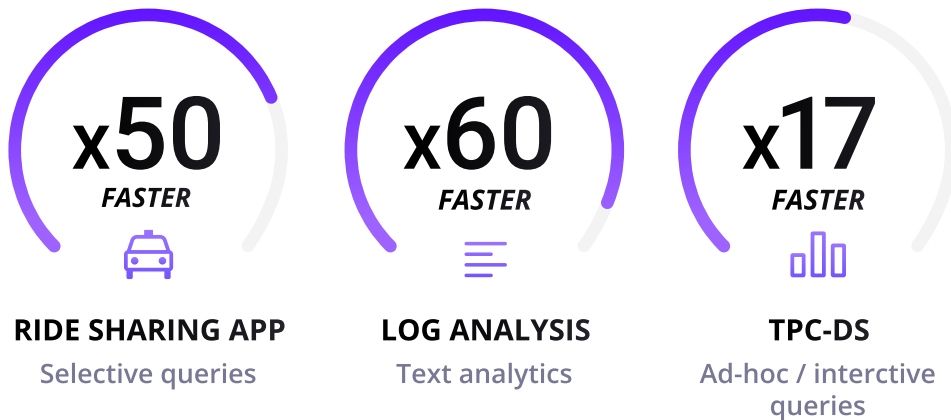
Varada has introduced a Presto based query engine that gives data lake administrators the power to optimize their Athena based data virtualization architecture. Varada lets you run a full scale production grade data virtualization solution without needing to resort to an add-on data warehouse or hand optimize every query. Best of all, Varada runs directly in your VPC, with an easy initial deployment through **AWS Marketplace**. Users can access everything in the data lake via Varada through the shared catalog using AWS Glue or the Hive metastore. Administrators simply need to make Varada available to users via a standard Presto endpoint.



Varada vs. Athena: Performance Stats

We ran a set of 19 queries on AWS Athena and compared them to a Varada cluster. We used different use cases to illustrate the performance uplift data teams can expect over a wide range of workloads.

With smart and dynamic acceleration technology,
Varada outperforms AWS Athena on all fronts!



Deliver superb cost control and predictability



AWS Athena
Pay per data
scanned



Varada
Auto-scaling w/deep visibility,
admins can cap budget

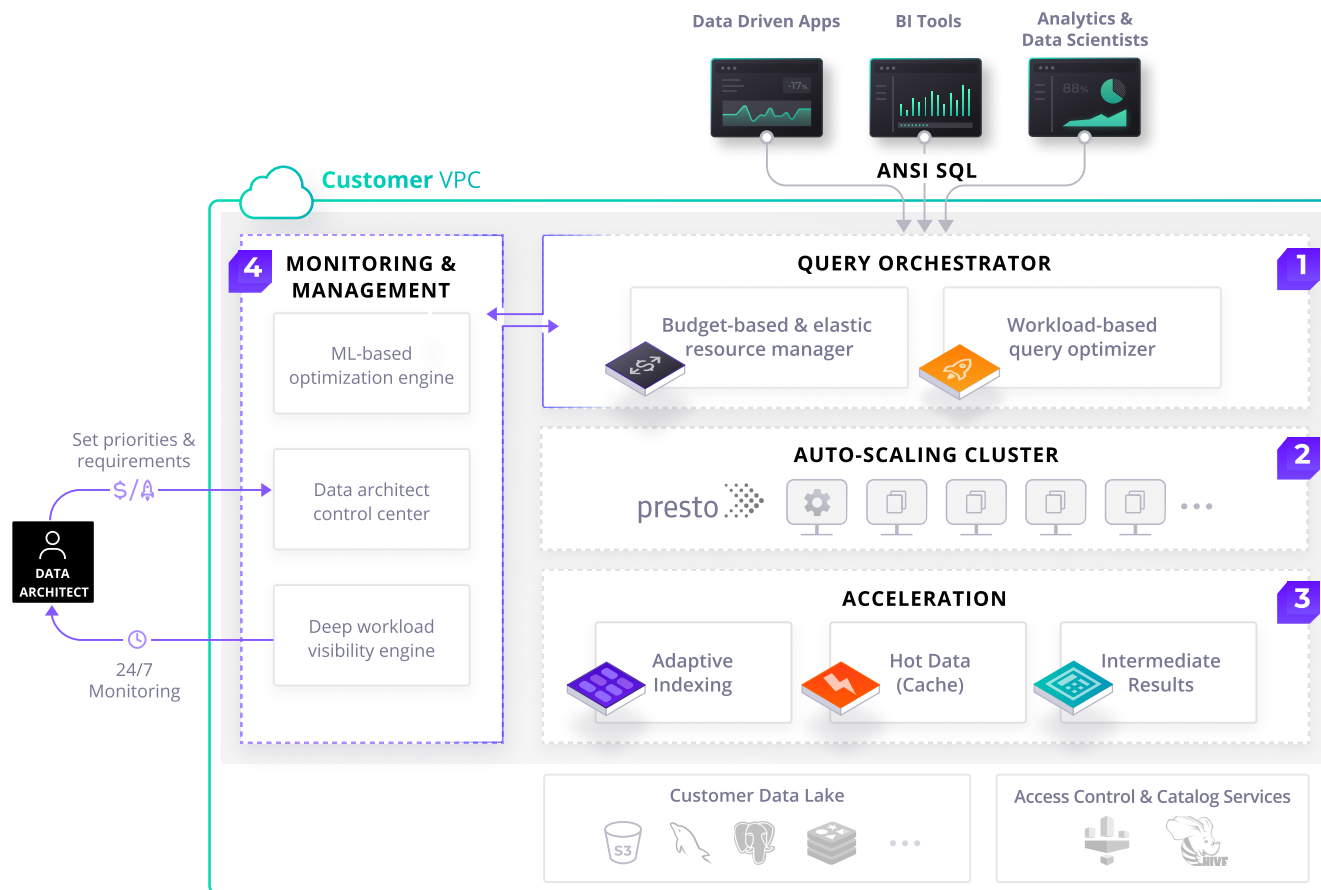
On average, Varada delivered x30 faster response time and as much as x70, at a minimal compute footprint. On full scan queries, Varada ran slower than Athena due to the small cluster. To view the detailed benchmarking report [click here](#).

Varada's Innovative Data Virtualization Architecture

In order to achieve these dramatic performance gains while maintaining a zero devops and dataops footprint, Varada uses several unique technologies that address the gaps which often lead people to abandon a purely Athena based architecture. Varada's dynamic and adaptive indexing technology automatically accelerates relevant queries without any overhead to query processing or any background data maintenance. Varada's indexing works transparently for users and indexes are managed automatically by Varada's proprietary cost based optimizer extensions. Varada is able to identify which queries to accelerate and which indexes to maintain.

The Varada Data Platform

is Composed of Four Major Components



1 Query Orchestrator

Manages queries and cluster resources according to budgets and workload priorities, and which elastically grows and shrinks the cluster compute and data caches based on the load.

2 Distributed Query Engine (Presto-based)

Includes a Coordinator node that optimizes and distributes queries, and Workers that execute queries using massively parallel processing.

3 Acceleration Engine

Optimizes queries on the fly using adaptive indexing, data materialization, and intermediate result calculation based on workload insights.

4 Workload Monitoring & Learning Engine

Uses machine learning to detect repeating patterns and hotspots in queries and adaptively choose dynamic acceleration. This engine exposes information to data teams, providing full visibility, the ability to explore workloads, and prioritize according to business needs.

Varada includes out-of-the-box native support for all community supported Presto connectors to access a wide array of data sources. The Varada query engine also expands upon the open source Presto query engine by adding enterprise grade support for high availability in the Coordinator and Workers, so both can withstand node failures. Varada's cost-based optimizer extends the basic optimizer with knowledge of how and when to accelerate queries with adaptive indexes. Varada Workers are able to auto-scale based on dynamic workload and administrator configuration.

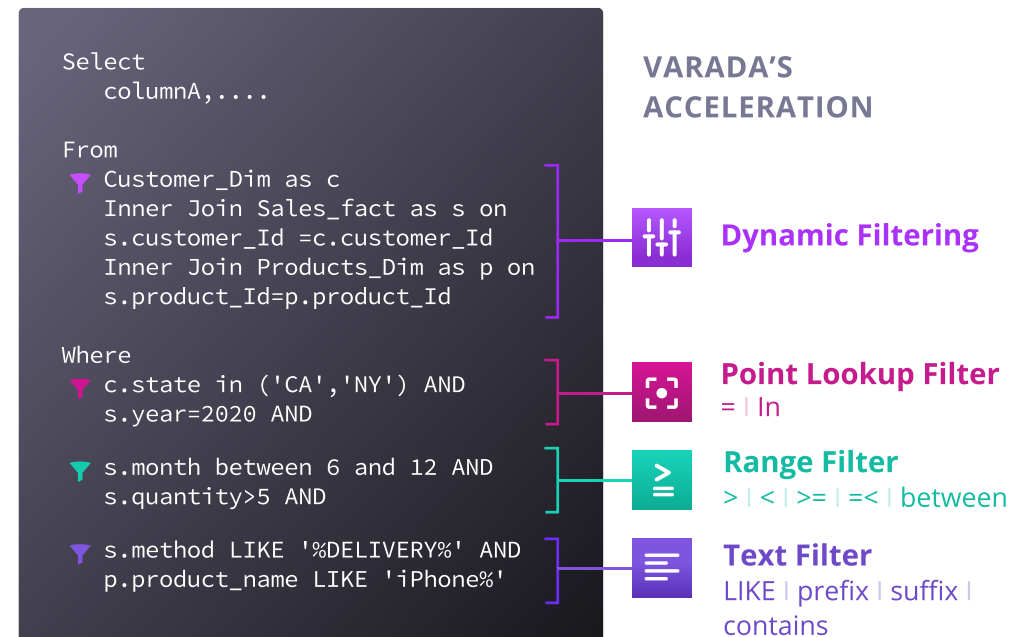
Adaptive Indexing

Varada's unique indexing efficiently indexes data directly from the data lake across any column so that every query is optimized automatically. Varada indexes adapt to changes in data over time, taking advantage of Presto's vectorized columnar processing by splitting columns into small chunks, called nanoblocks™. Based on the data type, structure, and distribution of data in each nanoblock, Varada automatically creates an optimal index. To ensure fast performance for every query and each nanoblock, Varada automatically selects from a set of indexing algorithms and indexing parameters that adapt and evolve as data changes to ensure best fit index any data nanoblock.

At query time when running through the Varada endpoint, users see transparent performance benefits when filtering, joining and aggregating data. Varada transparently applies indexes to any

SQL WHERE clause, on any column, within a SQL statement. Indexes are used for point lookups, range queries and string matching of data in nanoblocks. Varada automatically detects and uses indexes to accelerate JOINS using the index of the key column. Varada indexes can be used for dimensional JOINS combining a fact table with a filtered dimension table, for self-joins of fact tables based on time or any other dimension as an ID, and for joins between indexed data and federated data sources. SQL aggregations and grouping is accelerated using nanoblock indexes as well.

This example highlights the different techniques Varada leverages to optimize and accelerate queries:



Resource Aware Intelligent Cost Based Optimizer

The built in Presto cost based optimizer does an excellent job selecting efficient execution plans based on the table, filter, and projection statistics. The optimizer creates optimal plans in benchmarks and performs admirably in real world tests. Varada takes the CBO to the next level, by automatically analyzing and introducing indexes for filtering, joins and aggregates, continuously reanalyzing query performance on the fly, and balancing resources across the entire system.

Varada uses machine learning to decide when and what to optimize. With the benefit of lightweight indexing, Varada is able to use intelligent and elastic resource allocation, and leveraging intermediate results. The resulting cost model is exposed to administrators and users who can then prioritize specific user queries.

Highly Available with Streamlined DevOps

Varada's cluster is deployed as a private managed service, within your VPC environment. This deployment model ensures that data remains within your secured VPC. Varada integrates with standard AWS data access and governance solutions. Varada includes a control center that enables easy and ongoing monitoring of Varada clusters. Just as with the Varada cluster, the control center is deployed in your VPC and uses cross-account permissions to provision and set up instances in your VPC.

Seamlessly Integrated in Your AWS Environment

Varada uses EC2 instances as well as auxiliary services to spin up clusters in the VPC. Using AWS services such as SQS, to ensure data freshness, and RDS for highly available cluster management, Varada easily integrates with your existing data lake architecture. Varada's cloud-native architecture is built on ephemeral nodes with local NVMe SSDs, which are used as caches for indexes, materialized data, and intermediate results. Locally attached NVMe SSDs or SSDs on Kubernetes connected shared storage provides a high performance storage layer for indexing and data, shared across the cluster. Varada also supports a fully decoupled deployment mode when a separated shared NVMe storage is available.

When needed, Varada fetches operational datasets from the data lake, indexes it in-line with processing and stores indexes, data, intermediate results and any combination on SSDs. With Varada's architecture, there is no data skew, common with other distributed systems. All operational data on SSDs is accessible with consistent and predictable latency. Memory, and compute resources for queries are automatically managed by Varada automatically and does not require any data modeling or configuration. Varada eliminates all data locality considerations in materialized data, typical of shared-nothing architectures. Since compute resources are evenly balanced between workers, there are no inherent data related bottlenecks. Additional compute nodes can be added via auto-scaling.

Realize True Data Virtualization

Whether you're considering more cost effective architectures for a cloud data lake or have already gotten started with Presto and Athena, you'll find a lot of success with an AWS Athena based solution paired with Varada. Athena brings the reality of a no-devops query engine to AWS based data lakes, enabling true data virtualization without costly data duplication and brittle data movement. For large-scale use cases, Varada allows data architects to seamlessly accelerate and optimize workloads to meet specific performance and cost requirements with zero data-ops and effective resource utilization.

VARADA

THE NEW STANDARD FOR DATA VIRTUALIZATION

