Whitepaper

Logical Data Fabric

A TECHNICAL WHITEPAPER

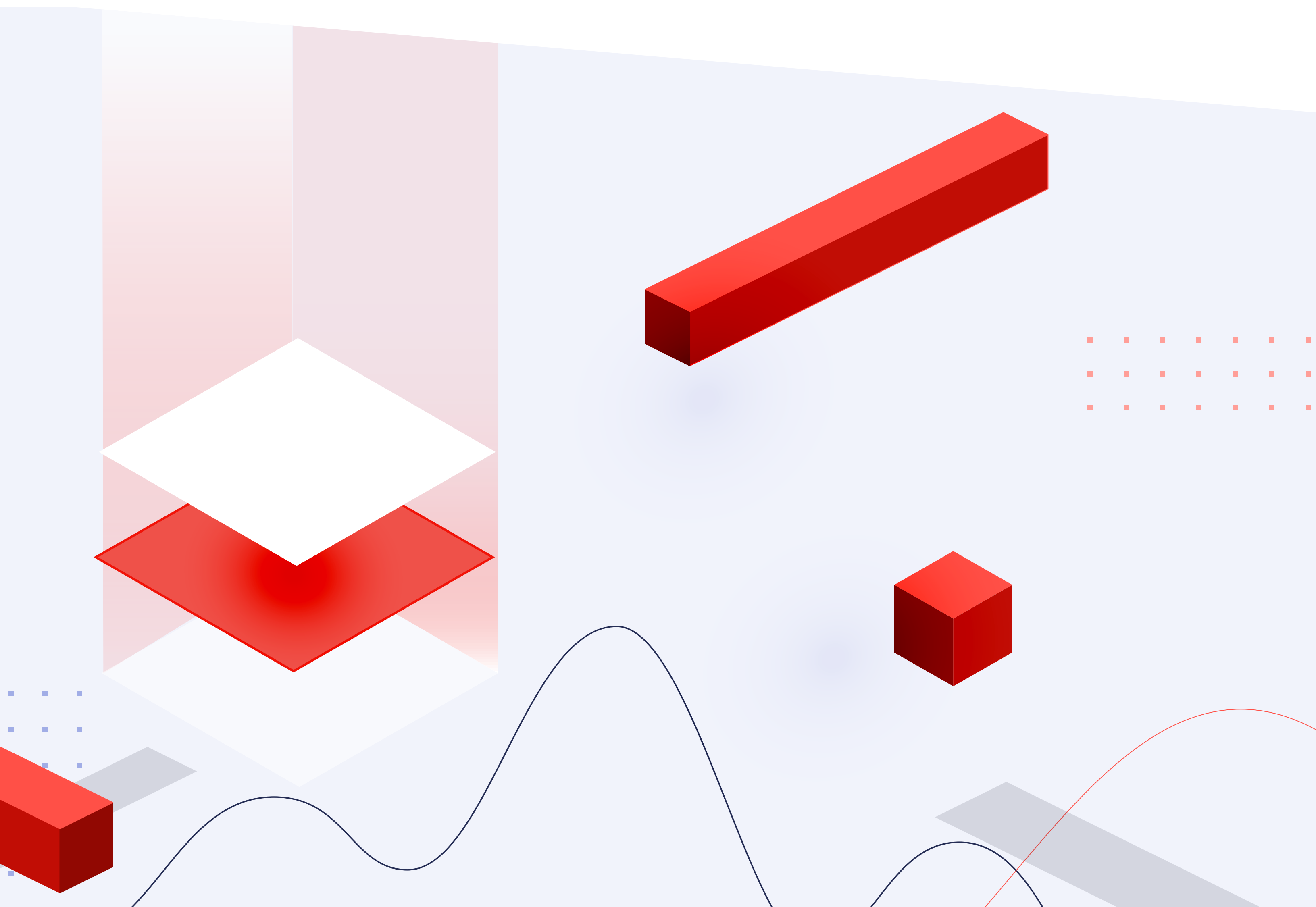


Table of Contents

Abstract

In this whitepaper we will define “data fabric,” look at different implementation strategies, and highlight the value of a logical approach. The paper will delve more deeply into the technical architecture of a logical data fabric and the key components to build a successful one, including the role played by artificial intelligence (AI) and data virtualization.

Introduction

The concept of a data fabric is relatively new, and its definition is broad. To understand where the concept comes from, it makes sense to step back and analyze the evolution of data management practices over the past decade. The era dominated by the enterprise data warehouse (EDW) is passing. New big data initiatives have fueled the rise of emerging disciplines like Machine Learning (ML) and data science for business decision making. Citizen integrators and power users were brought onto the business intelligence stage for self service, as IT departments struggled to manage more systems with lower budgets. Cloud and software-as-a-service (SaaS) solutions addressed some of these issues, but this also increased data distribution and siloes, creating new challenges. In summary:

- New methods in advanced analytics and machine learning practices gave rise to increasingly complex data requirements.

- The evolution of different specialized tools addressing different data management needs impeded organizations in establishing a “single version of the truth.” These new tools include EDWs, data marts, relational databases (RDBMS), data lakes, noSQL systems, internal and external REST APIs, real-time data feeds including social media feeds, and many more.

- Multiple personas now require access to the data: BI analysts, citizen integrators, data scientists, data stewards, IT, and data security professionals, each with different skills and requirements.

- Transitions to the cloud (or to multiple cloud platforms), which creates hybrid ecosystems in which data becomes physically fragmented. IT needs flexibility to adapt to new architectures while supporting the business with minimal interruption.

- Organizations must demonstrate higher standards in compliance and governance to fulfill specific legal frameworks (GDPR, CCPA) and external threats.

- Securing and governing a hybrid ecosystem can be complex and error prone.

In this era of data management turmoil, the idea of a data fabric was born to set some guidelines for future evolution. Data fabric outlines an architecture based on these core ideas:

- A common access layer for all data sources and all consumers, which hides the complexity of the deployment and provides a single logical system for consumption.
- Availability of multiple data integration strategies to be used seamlessly, depending on the use case, for both analytical and operational scenarios.
- Additional semantics to make data elements (and the relationships and connections between them) easier to consume, operate, and manipulate.
- Broader governance, documentation, and security features across the board, geared toward providing stronger trust and confidence in the data.
- Automation, leveraging active metadata and AI, to make it significantly easier to develop, operate, and use such a system.

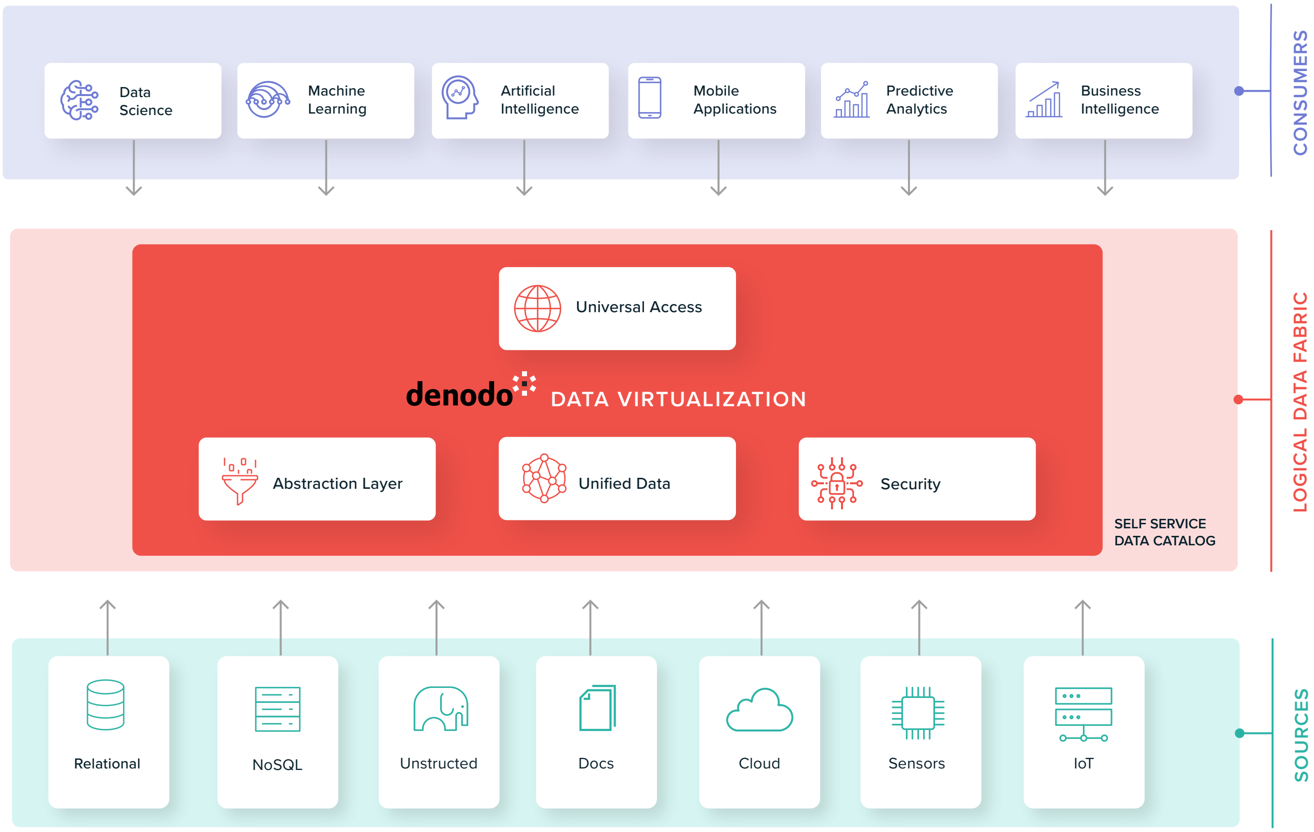


Figure 1: High Level Diagram of a Logical Data Fabric

Leading industry analyst Gartner defines data fabric as *"an architecture pattern that informs and automates the design, integration, and deployment of data objects regardless of deployment platforms and architectural approaches. It utilizes continuous analytics and AI/ML over all metadata assets to provide actionable insights and recommendations on data management and integration design and deployment patterns. This results in faster, informed and, in some cases, completely automated data access and sharing."*

The final goal of data fabric is, therefore, to enable a more agile, seamless, and, in many cases, automated approach to data access and data integration. It should provide enough complexity to enable advanced analytics while at the same time offering a friendly facade with which business users can interact. A mature data fabric should be able to support both analytical and operational scenarios.

From a more tactical perspective, Gartner says *"The core of the matter is being able to consolidate many diverse data sources in an efficient manner by allowing trusted data to be delivered from all relevant data sources to all relevant data consumers through one common layer."* (Demystifying the Data Fabric, by Jacob Orup Lund, September 2020).

The next question is how to translate those concepts into a realistic data architecture. We see six key components that are necessary to build a functional data fabric:

1. **A Data Virtualization Engine:** abstracts data and allows for the decoupling of applications/data usage from data sources to provide a common access layer.
   - A key component of this engine is an intelligent query optimizer that helps reduce processing costs and optimizes speed.
2. **An Augmented Data Catalog:** facilitates data exploration and discovery and improves collaboration and data governance.
3. **Active Metadata:** enables auditing and historical analysis and serves as the foundation for AI processes.
4. **A Semantic Layer with Extended Metadata**: enriches traditional technical information with business terms, tags, status, or documentation, to fuel improvements in self-service, security, and governance across all data assets.
5. **AI-Based Recommendations:** useful throughout the platform to learn from usage and simplify the entire lifecycle of the data management practice, including development, operations, performance tuning, etc.
6. **DataOps and Multi-Cloud Provisioning:** reduces management and operational cost and enables the system to be cloud-vendor agnostic.

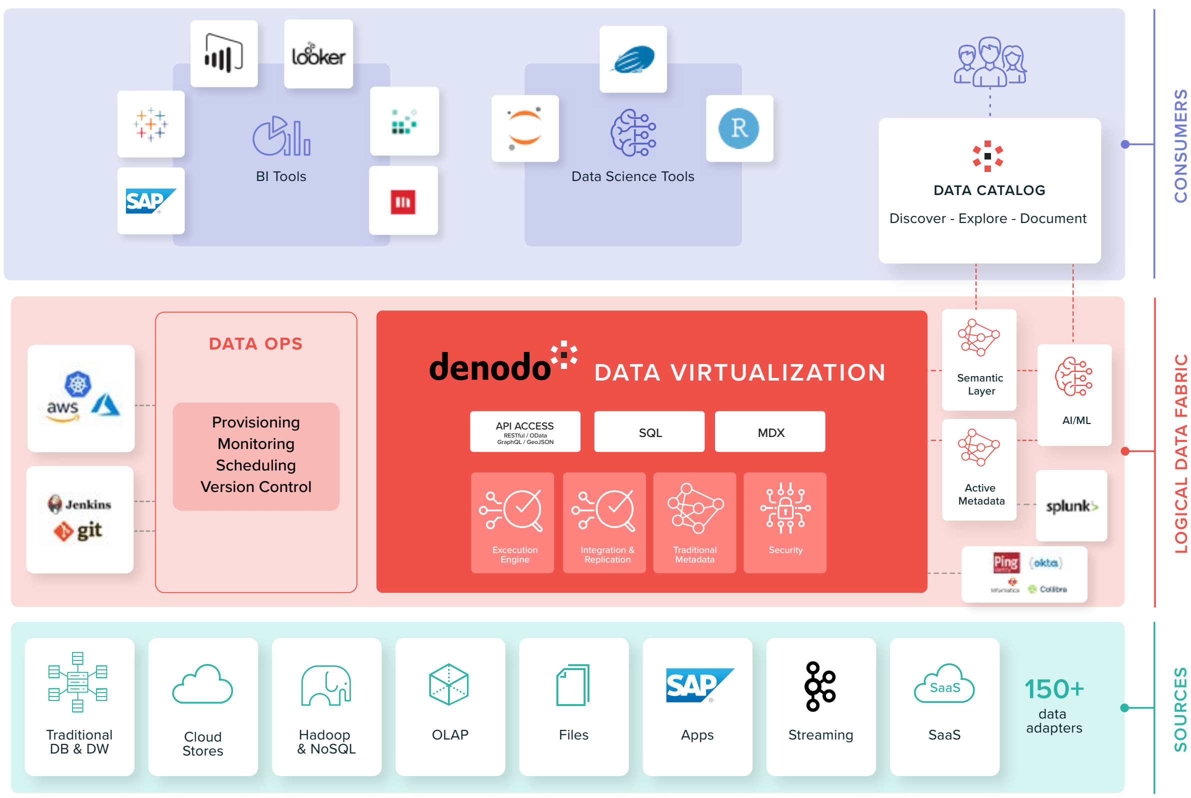


Figure 2: Functional diagram of a Logical Data Fabric

The core data virtualization layer abstracts the underlying data sources, centralizes access, data integration, and security. Its capabilities are extended with active metadata, an AI engine, and a data catalog, and other features, to create the foundation for a data fabric strategy. In addition, as depicted in the diagram, it also provides connection points with other common elements in the ecosystem like version control, governance, log management, and many others.

Additional information on data fabric can be found in industry analyst papers, such Gartner's Demystifying the Data Fabric (September 2020, by Jacob Orup Lund) and Forrester's Enterprise Data Fabric Wave. Also see Gartner's blog post entitled "Data Fabric Architecture is Key to Modernizing Data Management and Integration" this Forbes post entitled "Big Data Meets Data Fabric and Multi-Cloud."

In the following sections, we'll move into a more in-depth analysis of the components and capabilities required to enable a data fabric.

Where in an Organization is a Data Fabric Relevant?

A data fabric is relevant wherever an organization accesses or processes data. Any and every user within the organization that needs to report on data, process data, or just view data will be touching the data fabric.

These traditionally include business analysts, data scientists, front-line employees, back-office functions such as IT. and even more so today, data governance professionals such as chief information security officers or chief information officers.

Last but not least, IT professionals like DBAs, architects, and data integration specialists will all help operate and evolve the data fabric.

What Are the Different Approaches to a Data Fabric implementation?

Data fabric is a popular topic in the current data landscape, and thus multiple vendors have approached it from different angles.

A key tenet of the data fabric is the flexibility in data integration methods to enable data management in a distributed ecosystem. That is, depending on the nature of your use case and its requirements, any user (or the fabric itself, in an AI-based decision) should be able to use the best integration strategy and the data system that is the most suitable for that workload. This includes the use of extract, transform, and load (ETL) processes, extract, load, and transform (ELT) processes, real-time federation, and a combination of data lakes, enterprise data warehouses (EDWs), operational systems, etc.

Many vendors, especially in the data lake and ETL space, ignore these premises. Their approach to data fabric focuses on physical replication into a monolithic central repository, and automation around those pipelines.

However, these approaches come with severe disadvantages:

- **Solution Lock-in:** This reduces the flexibility of the fabric, forcing it to rely on a single system with which the customer is locked in. This also prevents the fabric from evolving, in that changes to storage and execution are impossible.
- **Slow data provisioning:** This imposes replication as a requirement for consumption of new assets, which leads to reduced agility and longer time-to-value
- **Single solution for all workloads**: This forces all workloads into a single system, which may not be the best fit for each use case. As a simple example, an operational scenario (e.g. an API to retrieve customer details) will not have the desired performance when the backend is in a data warehouse or a data lake. It also prevents reusing analytic systems (e.g. domain-specific data marts) that may be already available and almost ready for consumption
- **High maintenance cost:** These monolithic approaches come at a high operational cost, as replication pipelines need to be constantly executed, maintained, and evolved.

Of course, replication is often necessary, and definitely a part of what a data fabric needs to provide. But to solely rely on it as the means of integration fails to address some of the key challenges that the fabric is trying to solve.

A data fabric must embrace the ideas of **distributed** data and **logical** access. What does that mean?

- **Distributed** implies that the modern data ecosystem is composed of multiple elements. There is no one-size-fits-all system in data management. A modern data ecosystem requires data warehouses, data lakes, operational stores, noSQL sources, real-time feeds, and more. In addition, hybrid and multi-cloud environments are becoming the norm, increasing the distribution of data.
- **Logical** means that access to data is done through a logical abstraction layer. This hides the complexity of the backend and provides a single access point for consumption, security, and governance. The logical layer must also enable multiple integration strategies. Its metadata should enable direct access to the sources, but also real-time federation, selective materialization of specific datasets (e.g. caching, aggregate aware tables), extract, load, and transform (ELT) curation in a data lake, and full dataset replication.

It is easy to see how distributed architectures address the problem of different workloads (one size never fits all) and how logical architectures address the vendor lock-in problem thanks to the abstraction capabilities of logical access. Embracing the distributed nature of the ecosystem, and being able to immediately use any data without initial replication, accelerates data provisioning. Additional capabilities like semantic modeling, advanced metadata management, and many others, help reduce operational costs.

In a distributed world, a logical layer is the foundation for an efficient data integration strategy. This is when data fabric becomes a *logical* data fabric. Data virtualization is the key technology for enabling logical data fabric, as it provides real-time access to disparate data sources. However, only advanced data virtualization vendors provide the necessary infrastructure to fulfill the ideals of logical data fabric. In the following sections, we will dig deeper into the different areas to really understand how logical data fabric works.

What Are the Core Components of a Logical Data Fabric?

The diagram above illustrates the main components of a logical data fabric. However, architects need to understand the specific requirements of each individual implementation and how the components of the logical data fabric work together to meet those requirements. In the next section, we will cover those components in detail.

DATA VIRTUALIZATION: THE CONNECT, COMBINE, AND PUBLISH LAYER

Data virtualization is at the heart of any logical data fabric, as it is the middle layer that abstracts sources from consumption. It is a layer that connects, combines, and publishes data.

A data virtualization platform should be agnostic to the location and structure of the source data. The connect part accesses information from the various repositories and decouples any differences in the underlying communication protocol, query language, and formats from the upper layers. The user can generate “base views” over data sources, which represent a normalized schema that is available to upper layers in a tabular structure. Specific adapters for each kind of data source simplify this process, so that users of the data virtualization tools do not need to know these details.

These abstracted objects can be combined, creating virtual data models in a business-friendly format. Joins, aggregations, and transformations in structure and data allow the virtual layer to match the expectation of data consumers. Graphical wizards, and advanced metadata management options (like source change detection, impact analysis for propagation of changes, version control, etc.) enable the efficient use of the tools.

The final key component to any data virtualization solution is the data publishing component. The data virtualization layer exposes a single point of access for consuming applications. Data is exposed using a variety of standard protocols like SQL via JDBC, ODBC, and ADO.NET, as well as web services APIs in different formats: RESTful, OData, GraphQL, and GeoJSON.

Data access can be secured, normally in conjunction with an external Active Directory instance or an identity provider.

EXECUTION AND DATA INTEGRATION: EXECUTION ENGINE AND OPTIMIZER

A very important element of the data virtualization layer is the execution engine and its optimizer, which is in charge of creating the execution plans that retrieve data in the most optimal manner. It works similarly to a relational database's engine: it analyzes the incoming SQL and the internal metadata and comes up with a query execution plan that produces the results. However, it has a huge difference: the virtual layer only contains metadata. Data comes from the original sources (or managed copies like the cache). This requires different logic and algorithms that blend RDBMs techniques with those of data integration.

When data comes from a single source, the data virtualization layer performs a dialect translation but leaves most of the job to the underlying source. It acts almost like an API manager, but for SQL. The overhead it adds is minimal.

However, data virtualization comes into its own when the requested data is located in more than one source. For a real-time execution, data virtualization needs to choose among a variety of techniques for operations like joins or aggregations (in-memory merges, hash joins, nested loops, on-the-fly data movement to temp tables, etc.) and query rewriting techniques (branch pruning, partial aggregation splits, etc.). The data virtualization engine's cost-based optimizer plays a big role in this case, as it estimates partial volumes and uses them to weigh different options.

Besides real-time access to data sources, a good data virtualization engine must also support selective materialization techniques, such as the use of caching or aggregate-aware summaries to enhance performance. The Denodo Platform also enables full replication and is able to run ETL or ELT jobs when the extra copy is justified.

A full discussion of optimization and performance is outside the scope of this paper, but more information on these topics can be found in the Denodo website and the following articles in the Data Virtualization blog:

- Achieving lighting fast performance in the logical data warehouse
- Smart query acceleration increases performance of the logical data fabric.

AUGMENTED DATA CATALOG

A key aspect of any self-service strategy is the ability for business users to find what data sets are available in the data delivery layer and work out which ones are relevant to them.
The data catalog is aimed at providing this ability to all users through intuitive user interfaces, often available as a web portal or marketplace.

Some of the important functions of the data catalog include the ability to:

- Provide a user-friendly representation of the datasets and documentation, as well as the data lineage and relationship with other datasets.
- Leverage advanced search capabilities with a variety of filters.
- Support AI-powered features that analyze user activity to provide personalized recommendations.
- Classify datasets using business categories and/or tags to facilitate browse and exploration by business users.
- Preview datasets and obtain data profiling information.
- Collaborate, for example to endorse datasets, add comments or warnings about them. This will help data stewards and administrators to further contextualize dataset usage and better understand how other users experience the data.
- Use alternative exploration methods that help find the right information more quickly: popularity ranks, frequently used data, new content.

These features are often the backbone of a self-service strategy, where the data catalog is the component that bridges the gap between the complex lingo of IT professionals and the business language of end users.

Denodo includes a comprehensive Data Catalog as part of its platform, which surfaces the metadata of all data models across the enterprise with a focus on the business community. In addition to the features described above, it also includes some unique options:

- Perform Google-like free text searches without having to know the underlying schema to all the views available in the server, which have been previously indexed.
- Access to activity usage: "who," "when," and "how" a dataset is used including information such as expected execution time, most popular queries on the dataset, or the users and applications that use the dataset most frequently.
- Enable business users to create their own queries, and execute them from the catalog using a drag-and-drop interface, without needing to know SQL.
- Export results in various formats like CSV, TDE (Tableau Data Extract), or Excel.
- Save queries as favorites and export them as new views in the Denodo Server (given the right privileges), allowing reusability for other tools or users.

ACTIVE METADATA MANAGEMENT

In a logical architecture, access to any data source is channeled through a single layer, normally enabled by a data virtualization layer, as described above. This approach gives that logical layer a privileged position to capture data access activity, such as who accessed data, when they accessed it, how it was accessed, from what tool it was accessed, and how long it took.

This usage history, once it is properly processed and summarized, becomes a natural extension to the traditional metadata of a data set (like column names and data types). Since it represents the activity around a particular data asset, it's known as "active metadata."

Active metadata plays a critical role in multiple tasks, like security auditing ("Who accessed this restricted dataset between dates Y and Z?"), governance and stewardship ("Which are the least used datasets, that may be candidates for deprecation?") and to set the right expectations for end users in a self-service portal ("What's the average execution time for this dataset?"). In platforms like the Denodo Platform, this information is managed and integrated in other components, like the data catalog, to make the data more accessible to end users.

However, the most interesting aspect of active metadata is that it is also used as the main input for AI algorithms that learn from usage. This active metadata management is the foundation of AI-based automation, which is described more in detail in the next section.

AI-BASED RECOMMENDATIONS

A truly effective logical data fabric is intelligent and provides automatic recommendations tailored to the specific usage and workloads it serves. It must be able to analyze past activities to predict the future in order to simplify and reduce the cost of using and operating a logical data fabric. This is at the forefront of research and development and some advanced vendors are currently incorporating AI-based recommendations into data management solutions.

The Denodo Platform already incorporates several AI-based recommendation features, with more coming in future updates. Some of the features already available are:

- Performance recommendations for Smart Query Acceleration: Using a combination of active metadata and data distribution statistics, the Denodo Platform's AI engine can recommend intermediate datasets that can be precomputed to boost the performance of analytical queries. In other words, the Denodo Platform can automatically recommend the selective materialization of certain data that would make the overall performance of the logical data fabric improve significantly without any action by the end user.
    - This is a great example of the power of a logical data fabric. A combination of active metadata, multiple data integration techniques, and AI to automate performance tuning in a way that is totally transparent to data consumers.

- Data discovery recommendations: similar to what e-commerce or video streaming platforms do, a data catalog can analyze user profiles and access behavior patterns to recommend datasets of potential interest to each user. This capability is integrated in the data catalog.

Current research areas include data discovery, data modeling, query self-tuning, and identification of sensitive data, all of which will be included in upcoming versions of the Denodo Platform.

A SEMANTIC LAYER WITH EXTENDED METADATA

A universal semantic layer has long been understood to be the key to democratizing data. Creating this universal semantic layer is essential to enable business users to align their understandings of the underlying data. Semantic metadata extends traditional metadata (column names, data types, etc.) with additional meaning. For example:

- Dependencies between elements, which enables users to explore data lineage and enables developers to perform change impact analysis
- Relationships between related datasets, even across data sources, to simplify exploration and querying
- Descriptions and other documentation elements that enable a better understanding of what's what
- Status messages, deprecation notices, or warnings, which enable communication between IT and end users
- Updates from owners, stewards, approvers, and other governing stakeholders who have performed specific tasks on that data asset
- Tags and business terms that enable the definition of a standardized data dictionary
- Identifiers for sensitive data elements
- Technical definitions of data metrics (e.g. profit, benefit, margin, etc.), which enable the centralized, platform-agnostic definition of enterprise-wide measurements

These semantic metadata elements need to be shared and used across all other components. For example, the data catalog should display status and documentation. But it should also include tags as well as business terms that can be used to define access control policies. We can see again the power of a logical data fabric in these scenarios, in which a smart combination of capabilities by the different components enables secure, well governed, self-service capabilities.

DATAOPS AND MULTI-CLOUD PROVISIONING

Once a logical data fabric is built, it needs to be successfully operated. This involves the management of new developments (version control, deployment management), monitoring and auditing, scheduling recurrent jobs (including notifications, error handling, retries, and other management aspects of batch execution), and many more. Although these features are often overlooked, they play an intrinsic role in a mature logical data fabric and are key to its success. Platforms like the Denodo Platform include components to specifically address those tasks. Additionally, it also offers integration points in the form of REST APIs to interact with other key elements of the architecture, like git, Jenkins, Splunk, and many others.

Finally, a logical data fabric must be agnostic to the location of the deployment, including any cloud provider, and be able to manage provisioning at minimal cost taking advantage of the automated provisioning options that those cloud platforms provide. In many large corporations, architectures go beyond one cloud, and logical data fabrics often span across different cloud providers as well as different geographies. We can see from the below diagram a real-life example of the application of a multi-cloud, multi-location logical data fabric.

In this diagram we have an enterprise cloud across geographic borders with some of the data held within the European Union. Due to GDPR compliance rules, access to some of the data in the EMEA zone needs to be restricted and only accessible from within Europe. By implementing a multi-cloud logical data fabric with data virtualization, with nodes in each cloud, we can achieve this, by making sure the data governance rules within EMEA and the sensitive data is all held within the EMEA availability zone.

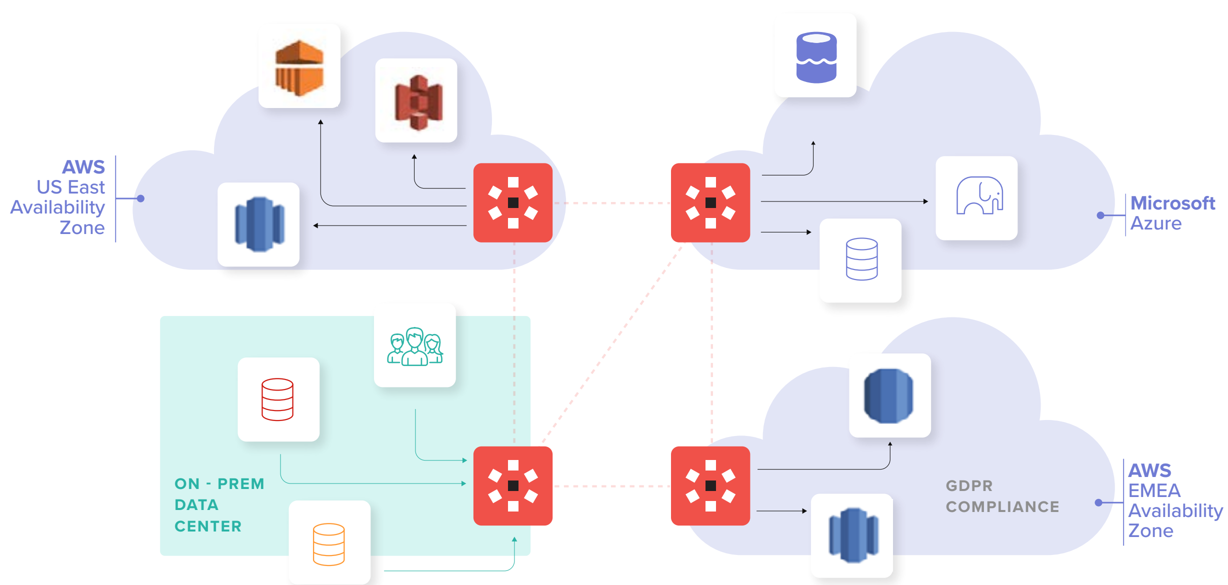


Figure 3: Multi-Cloud Provisioning

Multi-cloud data integration also brings the additional benefit of enabling the solution to optimize query processing within the local cloud, and using techniques like caching and Smart Query Acceleration to reduce egress costs. A final benefit of implementing a logical data fabric across a multi-cloud environment means that the architecture becomes cloud agnostic, so it can take advantage of individual cloud applications and data stores. It is not unheard of for an enterprise to shop around for cloud data storage and to regularly migrate to the most cost-effective offering. A logical data fabric enables IT to make these decisions with little to no impact on the business.

Key Benefits of a Logical Data Fabric

A logical data fabric approach to data management offers numerous benefits, but some of the most salient ones are:

1. **Better data discovery and self-service:** By combining an integrated data catalog, which offers a simple-yet-powerful tool to explore data across systems, with the advanced use of semantics, which brings meaning and context to data, logical data fabric makes self-service a more streamlined, trusted activity.

2. **More flexibility:** IT has a variety of methods and systems at their fingertips with which to manage data. They can incorporate new datasets in a few clicks and secure them, and they can choose among a variety of integration techniques (virtualized federation, full replication to another system, ELT, etc.) from a single, integrated Design Studio.

3. **Improved query performance in highly distributed data environments:** Advanced techniques like Smart Query Acceleration and advanced caching options, with an architecture designed to scale, ensure good performance in demanding scenarios.

4. **Automation across the board:** Activity metadata combined with AI-based recommendations simplify usage and operation of the platform.

5. **Centralized security and governance:** Logical data fabric provides a global access layer with which to enforce security and governance across the organization, regardless of the capabilities of each individual source.

Case Study: DNB

DNB is Norway's largest financial services group and one of the largest in the Nordic region by market capitalization. DNB Group offers a full range of financial services, including loans, savings, advisory services, and insurance and pension products for retail and corporate customers.

INITIATIVE

The deployment of a Denodo-Platform-based logical data fabric at DNB came as part of the definition of DNB's new Insights Platform for Analytics (IPA). IPA is a self-managed analytics ecosystem, fully deployed in AWS. It is focused on enabling and empowering users to bring data and code together for creating data insights and valuable data products.

One of the foundational ideas of IPA was to be able to support data and analytics everywhere for everyone in a distributed, decentralized IT Landscape.

IPA is a broad platform, built with multiple best-of-breed components. Among others, these included a data lake, a data science toolkit, visualization tools, and a logical data fabric powered by Denodo.

20 PLATFORM ENGINEERS	**248** DATA SCIENTISTS & ANALYSTS
40+ ML PIPELINES & APIS	**20+** WEB APPLICATIONS
9000+ VIEWS FROM **40+** DATA SOURCES	**25** COMMON LIBRARIES

It reaches a broad community of engineers, data scientists, and analysts, who use a variety of tools for data consumption and analysis. It spans across more than 40 data sources, including:

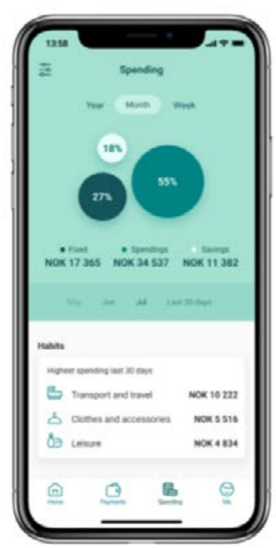
- A data lake in AWS with transactional data, storing more than 7 billion transactions
- Multiple on-premises data warehouses, built on multiple technologies like Oracle and Teradata
- External APIs (e.g. Google APIs)

Most internal business units are on-boarded to the platform: Personal Banking, Corporate Banking, Wealth Management, Markets, and others. In total, around 250 data scientists and analysts have access to the Data Science Lab and DataMarketplace through the logical data fabric.

USE CASES

The Denodo logical data fabric, and globally the IPA framework that the logical data fabric enables, host a broad variety of use cases.

MOBILE BANKING



DNB's first major use case enabled by the logical data fabric was DNB's new mobile bank application, a serverless application built in AWS. Currently, the application has nearly 1 million active users. But the interesting part here is that about 820,000 of these users are leveraging this as their personal finance management tool. Therefore, it has been a critical tool to improve customer care and retention.

A very popular function of this application is the automatic analysis of the transaction history to draw statistics and recommendations. For example, it can find your subscriptions to online services, like Spotify or Netflix. It identifies regular costs, in addition to mortgage and other recurrent payments. With that information, it automatically provides recommendations, such as how much a customer would have left to spend on the next salary. It can also break down transactions in different categories, such as travel, groceries, and so on. In summary, it provides custom finance advice.

ML AND ADVANCED ANALYTICS

IPA was built for machine learning and advanced analytics. DNB has over 250 active data scientists and analysts that use the Data Science Lab and Data Marketplace. They have built real-time models that are served via Denodo APIs for applications, like individual pricing, and batch models for other applications like churn prediction, leads, and next-best offers.

RISK ANALYSIS

IPA also supports some of the more traditional compliance and IT use cases. Risk analysis is a good example, and IPA has interesting risk models in place, such as e-fraud, cybercrime, and anti-money laundering.

GDPR

DNB supplies its transparency report for GDPR reporting from the logical data fabric, as DNB needs to integrate a lot of different systems. The Denodo Platform provides the right federation capabilities for this task.

BI REPORTING

From a traditional reporting perspective, DNB serves a large user base of MicroStrategy dashboards, also channeled to IPA though the logical data fabric enabled by the Denodo Platform.

CRITICAL CAPABILITIES

When evaluating its requirements for a logical data delivery layer, DNB focused on the following:

- Data Access Layer: Uniform SQL access to any underlying data source. Acts like a RDBMS with various access methods: SQL (JDBC/ODBC/ADO.net), APIs (REST/SOAP/OData), etc.
- Data Security Capabilities: Authentication (SSO) and authorization (fine-grained data security) across any underlying data source (files, RDBMS, HTTP etc.).
- Real-time Operational and Analytical Capabilities: Supports both operational queries and analytical queries to serve applications and operational BI, and data science use-cases.
- Query Performance Capabilities: Supports data federation, data caching, and platform scaling, as well as the ability to push down processing to various compute engines.
- Auditing and Monitoring Capabilities: Logs all administrative actions and data actions for full control.
- Top-Down and Bottom-Up Capabilities: Supports both top-down "contract first" model driven design and bottom-up data driven design. (Start with API/models or start by exploring data and metadata).
- Productivity Capabilities: Supports development with open API's that fit into DNB's CICD pipeline and consumers with decent web connections and SQL-based self-service features to make data accessible, intuitive, and informative.
- Enterprise Capabilities: Ticks off the most important boxes when it comes to high availability, disaster recovery, clustering, scaling, etc.

VALUE GENERATED BY DENODO'S LOGICAL DATA FABRIC

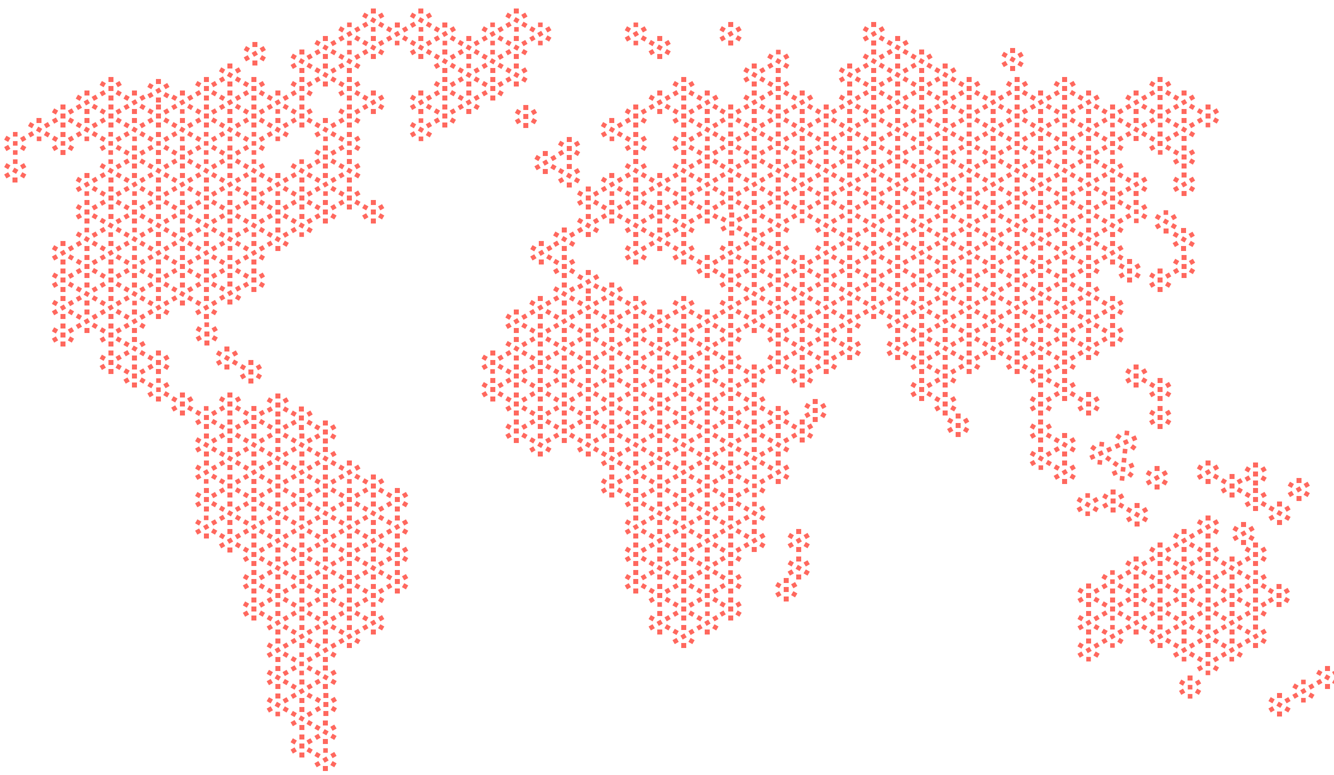
The adoption of IPA with the Denodo logical data fabric has been central in creating a state-of-the-art platform to support modern analytics and data science use cases at DNB. It has provided greater agility and flexibility to the technical teams, and it has brought cutting edge capabilities not just to internal departments but also to the bank's customers, with initiatives like Mobile Banking.

For more information, visit https://www.denodo.com/en/webinar/dnb-case-study-bringing-data-and-code-together-creating-insights-and-valuable-data-products

Conclusion

Logical data fabric is a forward-thinking approach to data management, combining the power of multiple components to create a powerful distributed and logical architecture that aims to address the big challenges of the enterprise data landscape today.

A virtual layer, enabled by a platform like Denodo, offers an enterprise-ready foundation for logical data fabric. In addition to traditional data virtualization capabilities like the federation engine, the Denodo Platform provides a thorough set of components that enable multiple integration strategies, advanced performance techniques, and AI-based recommendations. At the same time, it offers multiple UIs tailored for different personas (a design studio, a data catalog, a solution manager, etc.) and mature API-based integration points to successfully integrate data across any organization.



Denodo is the leader in data virtualization providing agile, high performance data integration, data abstraction, and real-time data services across the broadest range of enterprise, cloud, big data, and unstructured data sources at half the cost of traditional approaches. Denodo's customers across every major industry have gained significant business agility and ROI.

Visit www.denodo.com | Email info@denodo.com | Discover community.denodo.com