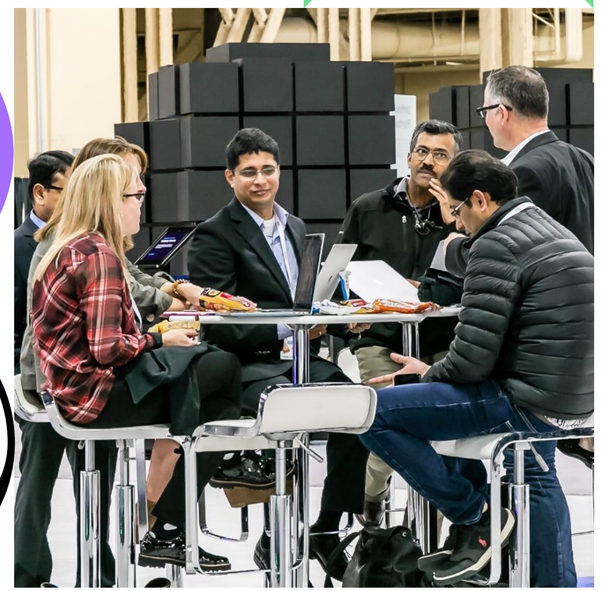
Tridex Db2 LUW

December 11th, 2024

An introduction to the data lakehouse architecture and IBM watsonx.data



Kelly Schlamb
kschlamb@ca.ibm.com
Executive IT Specialist, Data & AI, IBM

Notices and disclaimers

This document is distributed “as is” without any warranty, either express or implied. In no event shall IBM be liable for any damage arising from the use of this information, including but not limited to, loss of data, business interruption, loss of profit or loss of opportunity.

Customer examples are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual performance, cost, savings or other results in other operating environments may vary.

Workshops, sessions and associated materials may have been prepared by independent session speakers, and do not necessarily reflect the views of IBM.

Not all offerings are available in every country in which IBM operates.

Any statements regarding IBM’s future direction, intent or product plans are subject to change or withdrawal without notice.

IBM, the IBM logo, and ibm.com are trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at “Copyright and trademark information” at: www.ibm.com/legal/copytrade.shtml.

Certain comments made in this presentation may be characterized as forward looking under the Private Securities Litigation Reform Act of 1995.

Forward-looking statements are based on the company’s current assumptions regarding future business and financial performance. Those statements by their nature address matters that are uncertain to different degrees and involve a number of factors that could cause actual results to differ materially. Additional information concerning these factors is contained in the Company’s filings with the SEC.
Copies are available from the SEC, from the IBM website, or from IBM Investor Relations.

Any forward-looking statement made during this presentation speaks only as of the date on which it is made. The company assumes no obligation to update or revise any forward-looking statements except as required by law; these charts and the associated remarks and comments are integrally related and are intended to be presented and understood together.

The **Enterprise Data Warehouse**

- Highly performant data management platform
- Data from multiple sources organized into a centralized, highly-structured relational database
- Strongly governed
- Primarily supports data analytics and business intelligence applications
- Data stored in proprietary formats on fast, expensive block-based storage devices

What do you like about your **Enterprise Data Warehouse**?

- ✔ Trusted, reliable data – high-quality, consistent
- ✔ Provides a great data foundation for your business-critical BI and analytics workloads
- ✔ Scalable
- ✔ Fast!

(yeah, EDWs are great!)

But what don't you like about your **Enterprise Data Warehouse**?

- ✗ Can only store structured data
- ✗ Data locked in
- ✗ Maybe not scalable enough
- ✗ Not ideal for ML/AI projects
- ✗ It's costly! (infrastructure, processes, effort)

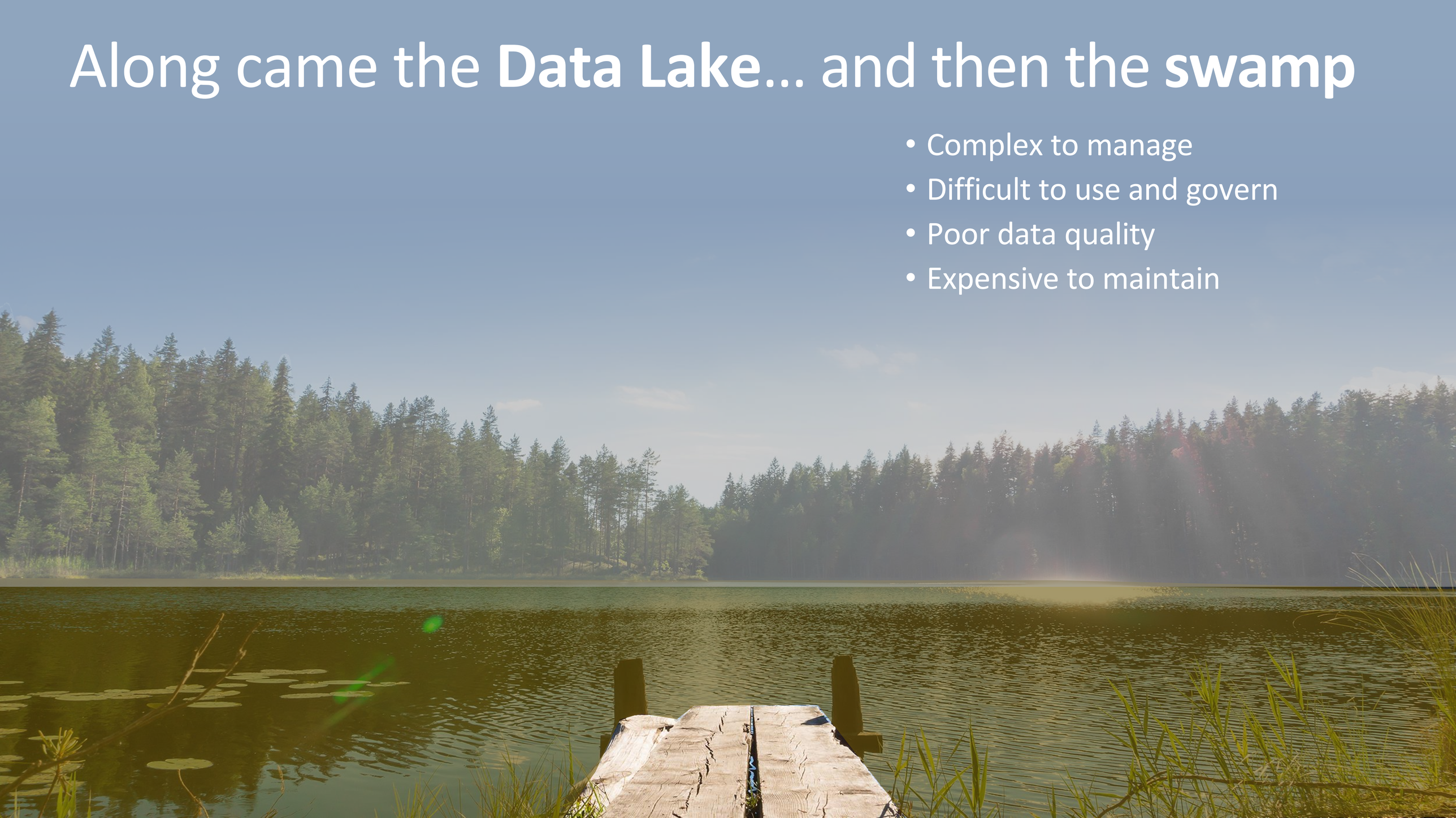
So, organizations have explored alternatives,
or additional data stores to augment their EDW…

Along came the **Data Lake**

- Low-cost, scalable to petabytes of raw data
- Stores structured, semi-structured, and unstructured data
- Commonly associated with Apache Hadoop
- Traditionally has used HDFS, but object storage increasingly more common
- Designed for data discovery and data science/ML use cases

Along came the **Data Lake**… and then the **swamp**

- Complex to manage
- Difficult to use and govern
- Poor data quality
- Expensive to maintain



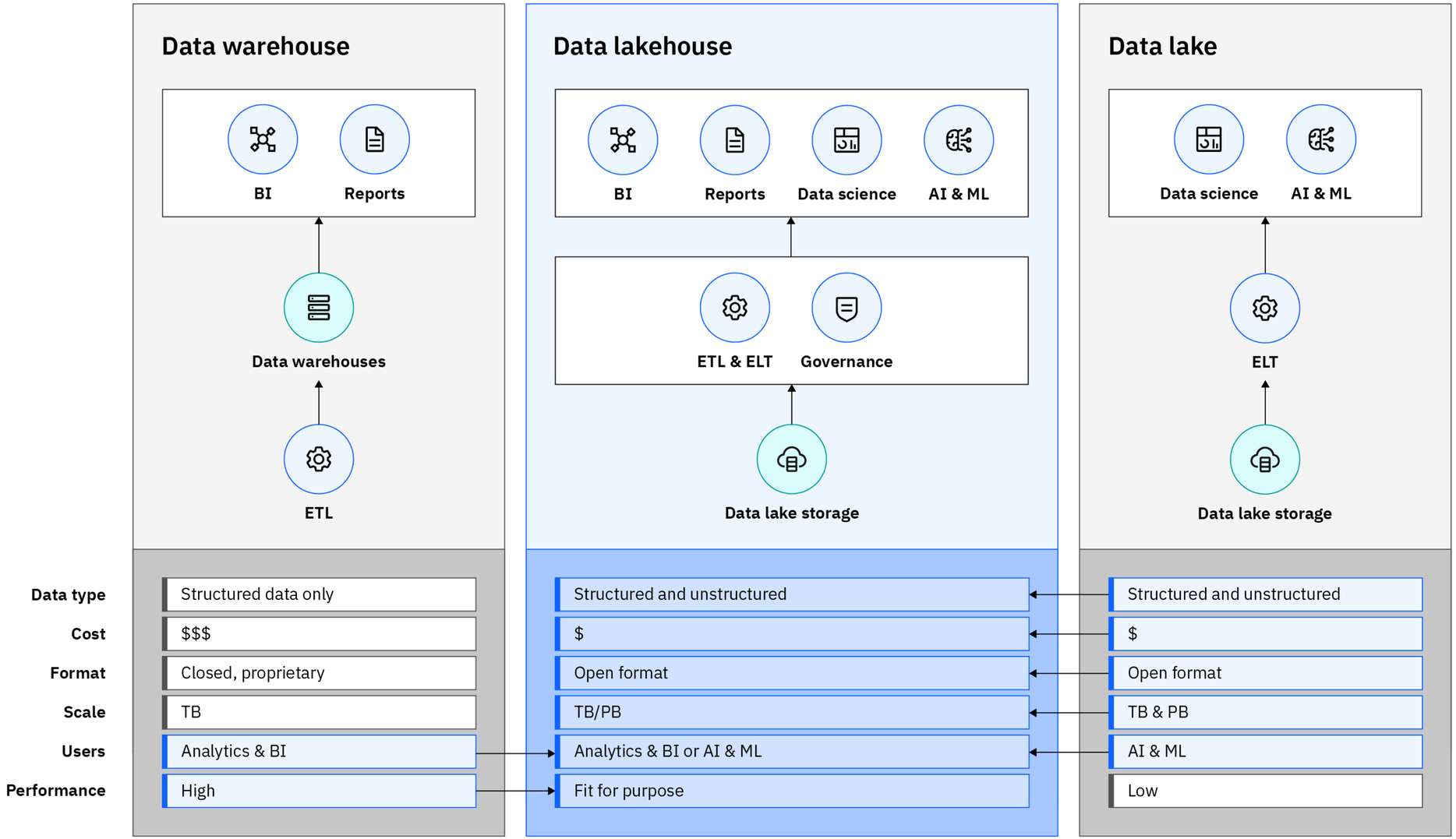
Introducing… the **Data Lakehouse**

- Brings together the best attributes of data warehouses and data lakes
- Utilizes low-cost object storage
- Exploits open data and table formats
- Flexibility to support both data analytics, BI and ML/AI workloads
- Highly scalable
- Fit for purpose query engines (ideally)

74% of surveyed organizations have adopted a lakehouse architecture, with most of the rest expected to do so in the next three years.

MIT Technology Review (Oct 2023)

Lakehouses are a new class of data store that combines the best of data warehouses and data lakes



First generation lakehouses are still limited by their ability to address cost and complexity challenges:

- Single query engines set up to support limited workloads ... typically just BI or ML
- Typically deployed on cloud only with no support for multi-/hybrid-cloud deployments
- Minimal governance and metadata capabilities to deploy across the entire ecosystem

The platform for AI and data

watsonx

Scale and accelerate the impact of AI with trusted data.

watsonx.ai

Train, validate, tune and deploy AI models

A next generation enterprise studio for AI builders to train, validate, tune, and deploy both traditional machine learning and new generative AI capabilities powered by foundation models. It enables you to build AI applications in a fraction of the time with a fraction of the data.

watsonx.data

Scale AI workloads, for all your data, anywhere

A hybrid, open data lakehouse to power AI and analytics with all your data, anywhere – supported by querying, governance, and open data formats to access and share data.

watsonx.governance

Accelerate responsible, transparent and explainable AI workflows

End-to-end toolkit for AI governance across the entire model lifecycle to accelerate responsible, transparent, and explainable AI workflows

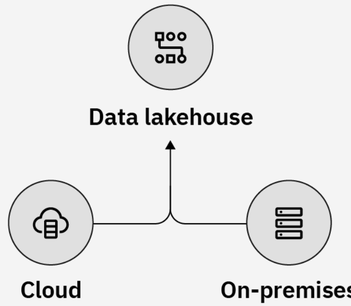
watsonx.data

Scale AI workloads, for all your data, anywhere

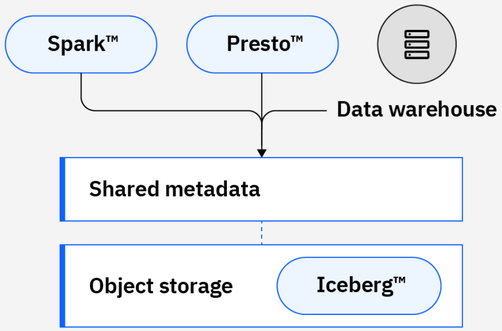
A hybrid, open data lakehouse to power AI and analytics with all your data, anywhere – supported by querying, governance, and open data formats to access and share data.

Seamlessly deploy across any cloud or on-premises environment in minutes with workload portability through Red Hat® OpenShift®.

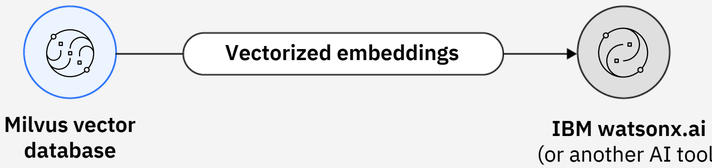
Access all your data through a single point of entry across all clouds and on-premises environments.



Reduce the cost of your data warehouse by up to 50%* through workload optimization across multiple query engines and storage tiers.



Unify, curate, and prepare data for AI

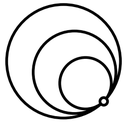


*When comparing published 2023 list prices normalized for VPC hours of IBM watsonx.data to several major cloud data warehouse vendors. Savings may vary depending on configurations, workloads and vendors.

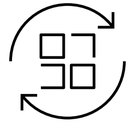
The IBM approach to a data lakehouse architecture combines the best of IBM with the best of open source

Best-in-class cost and performance optimizations for compute and storage

Built-in integrations with IBM data repositories and data fabric

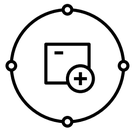


Deep expertise and capabilities in data and storage



Open and vendor-agnostic across architectural tiers

Enables hybrid, multicloud deployments with the Red Hat OpenShift platform



The best of open source

presto

Apache Spark

AVRO

HIVE

ICEBERG

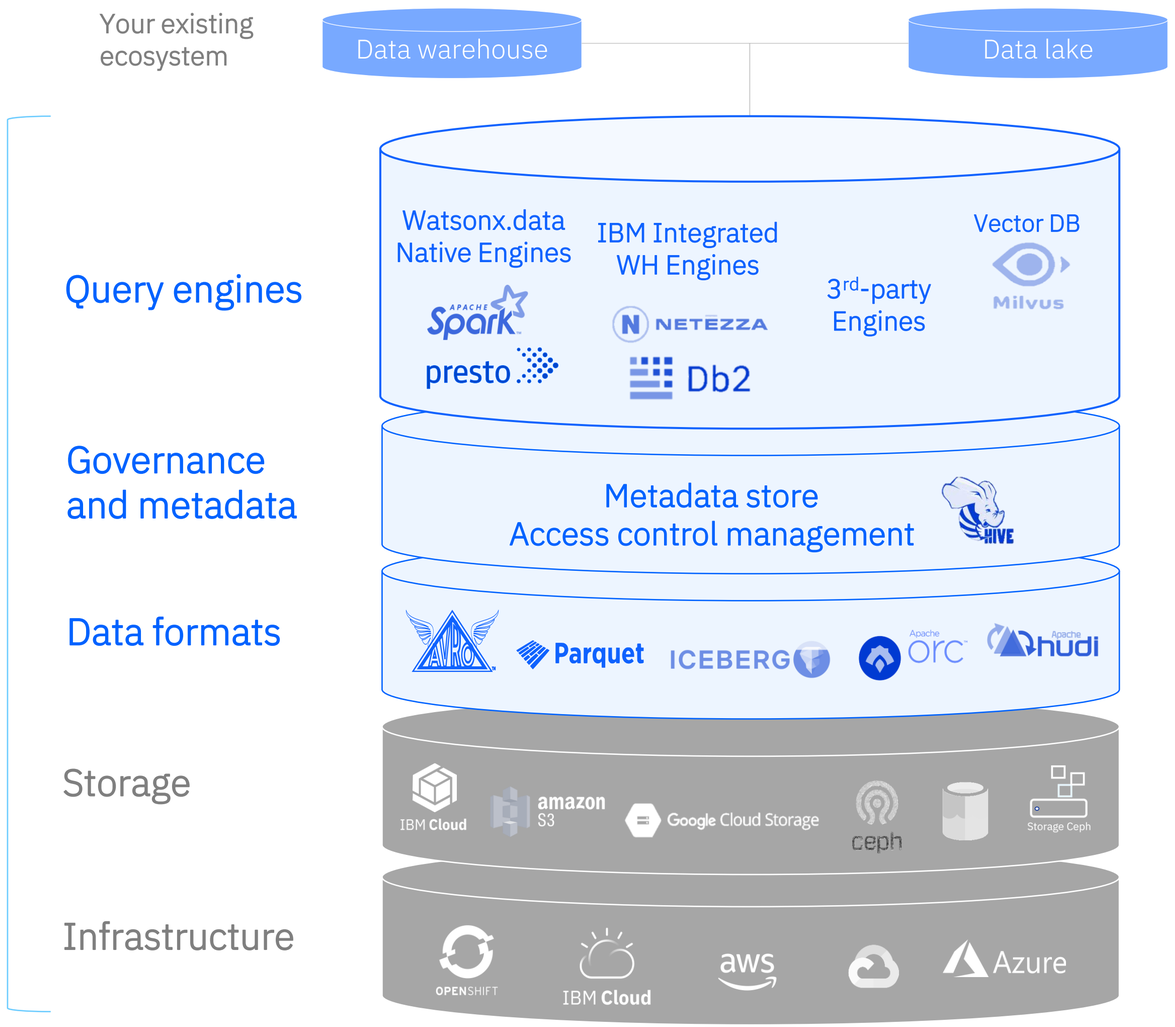
Parquet

Apache orc

IBM **watsonx.data** – the next generation data lakehouse

Completely open.
No lock-in!

Built on a foundation of industry-embraced open-source technologies.



watsonx.data

Multiple engines including Presto and Spark that provide **fast, reliable, and efficient processing of big data** at scale

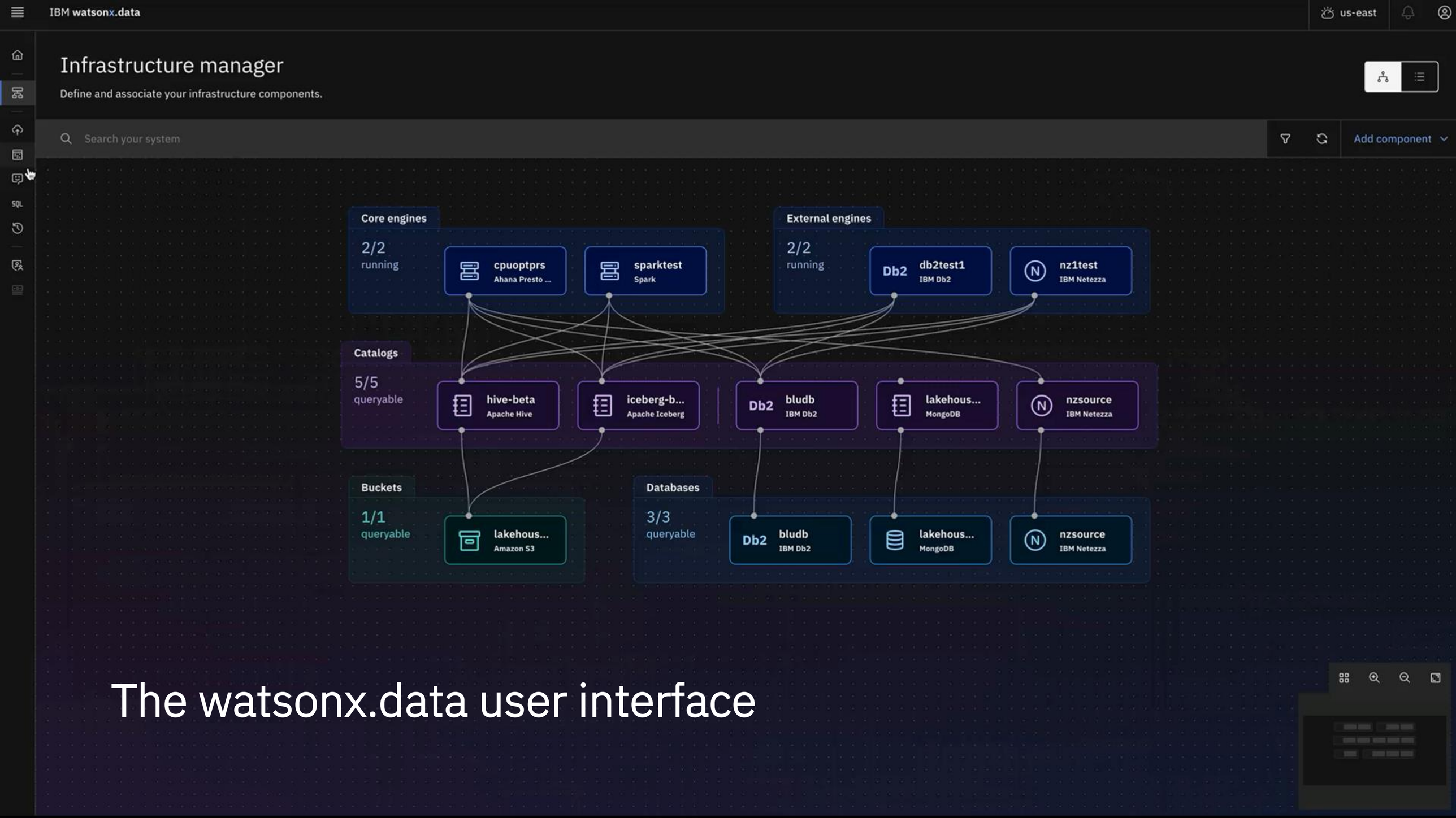
Milvus for **semantic searching and RAG** uses cases

Built-in governance that is compatible with existing solutions such as watsonx.governance and IBM Knowledge Catalog

Vendor agnostic open formats for analytic data sets, allowing different engines to access and share the same data, at the same time

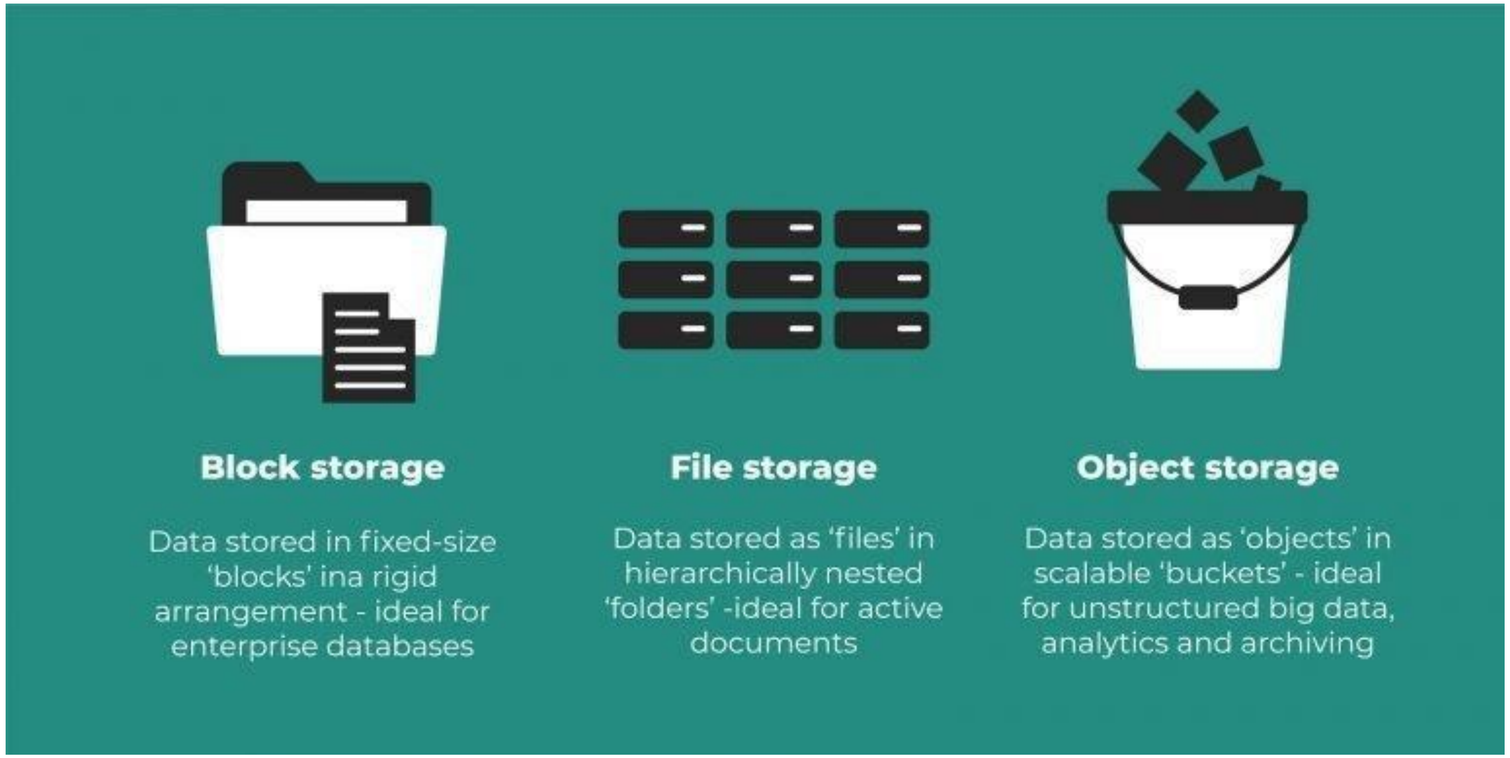
Cost effective, simple object storage available across hybrid-cloud and multicloud environments

Hybrid-cloud deployments and workload portability across hyperscalers and on-prem with Red Hat OpenShift



The watsonx.data user interface

What is object storage?



Object storage:

- Low cost
- Near unlimited scalability
- Extreme durability & reliability (99.999999999%)
- High throughput
- High latency (but can be compensated for)
- Basic units are *objects*, which are organized in *buckets*

- Most notable provider for object storage is Amazon S3 (Simple Storage Service)
- Other vendors offer S3-compatible object storage

amazon S3
IBM Spectrum Fusion
Storage Ceph
DELL EMC ISILON
Cloud Object Storage
NetApp

Common open data file formats

Computer systems and applications store data in files

Data can be stored in binary or text format

File formats can be open or closed (proprietary/lock-in)

Open formats (Parquet, ORC, and Avro) are commonly used in data lakes and lakehouses

CSV

- Human-readable text
- Each row corresponds to a single data record
- Each record consists of one or more fields, delimited by commas

{ JSON }

- Human-readable text
- Open file and data interchange format
- Consists of attribute-value pairs and arrays
- JSON = JavaScript Object Notation

Parquet

- Open-source
- Binary columnar storage
- Designed for efficient data storage and fast retrieval
- Highly compressible
- Self-describing

Apache orc

- Open-source
- Binary columnar storage
- Designed and optimized for Hive data
- Self-describing
- Similar in concept to Parquet

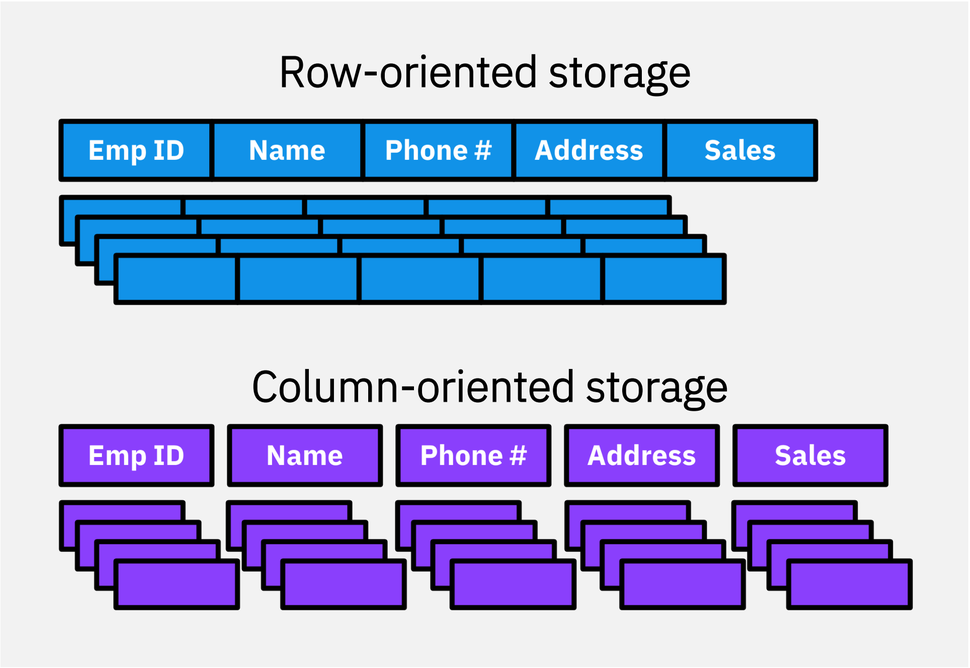
AVRO

- Open-source
- Row-oriented data format and serialization framework
- Robust support for schema evolution
- Mix of text/binary

Apache Parquet

Parquet is designed to support fast data processing for complex data

- Open-source
- **Columnar storage**
- Highly compressible with configurable compression options and extendable encoding schemas by data type
- Self-describing: schema and structure metadata is included
- Schema evolution with support for automatic schema merging

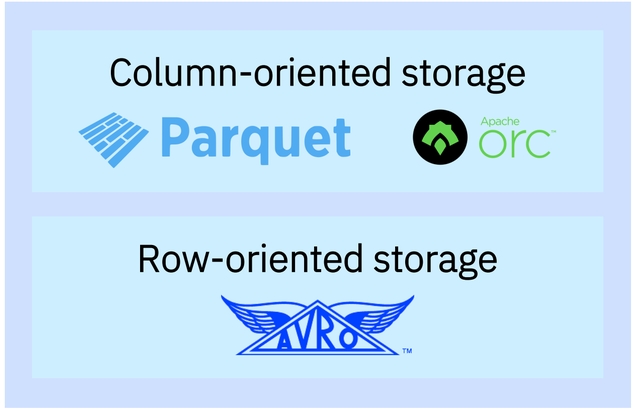


Why do these things matter in a lakehouse?

- Performance of queries directly impacted by size and amount of file(s) being read
- Ability to read/write data to an open format from multiple runtime engines enables collaboration
- Size of data stored, amount of data scanned, and amount of data transported affect the charges incurred in using a lakehouse (depending on the pricing model)

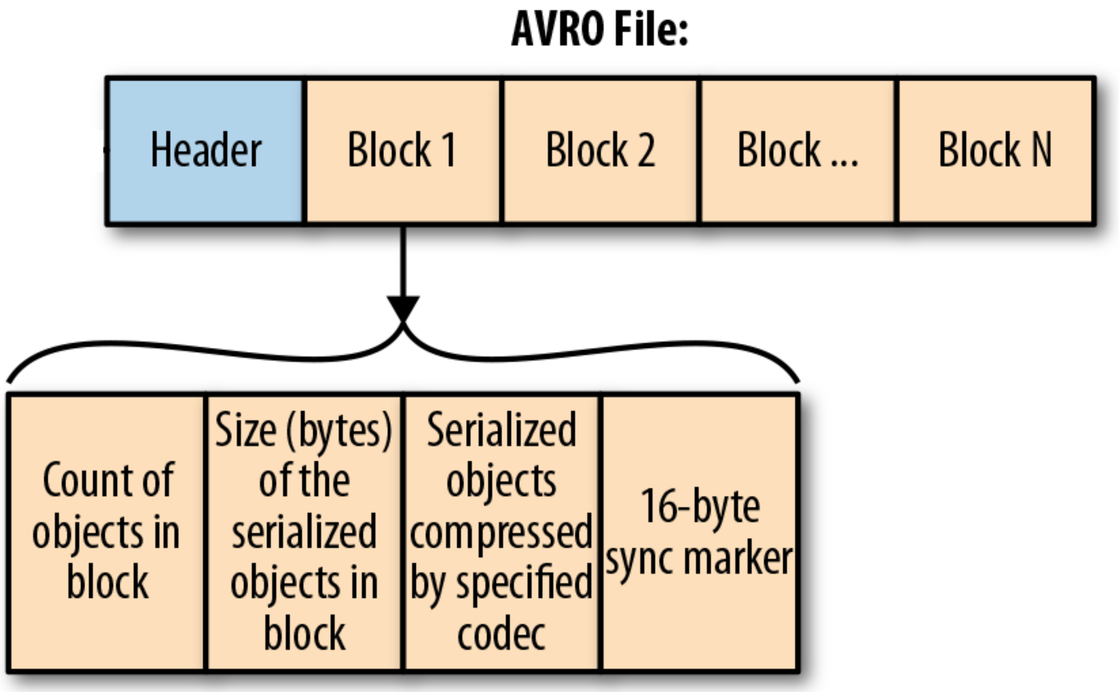
Apache ORC

- Open-source, **columnar storage** format
  - Similar in concept to Parquet, but different design
  - Parquet considered to be more widely used than ORC
- Highly compressible, with multiple compression options
  - Considered to have higher compression rates than Parquet
- Self-describing and type-aware
- Support for schema evolution
- Built-in indexes to enable skipping of data not relevant to a query
- Excellent performance for read-heavy workloads
  - ORC generally better for workloads involving frequent updates or appends
  - Parquet generally better for write-once, read-many analytics



Apache Avro

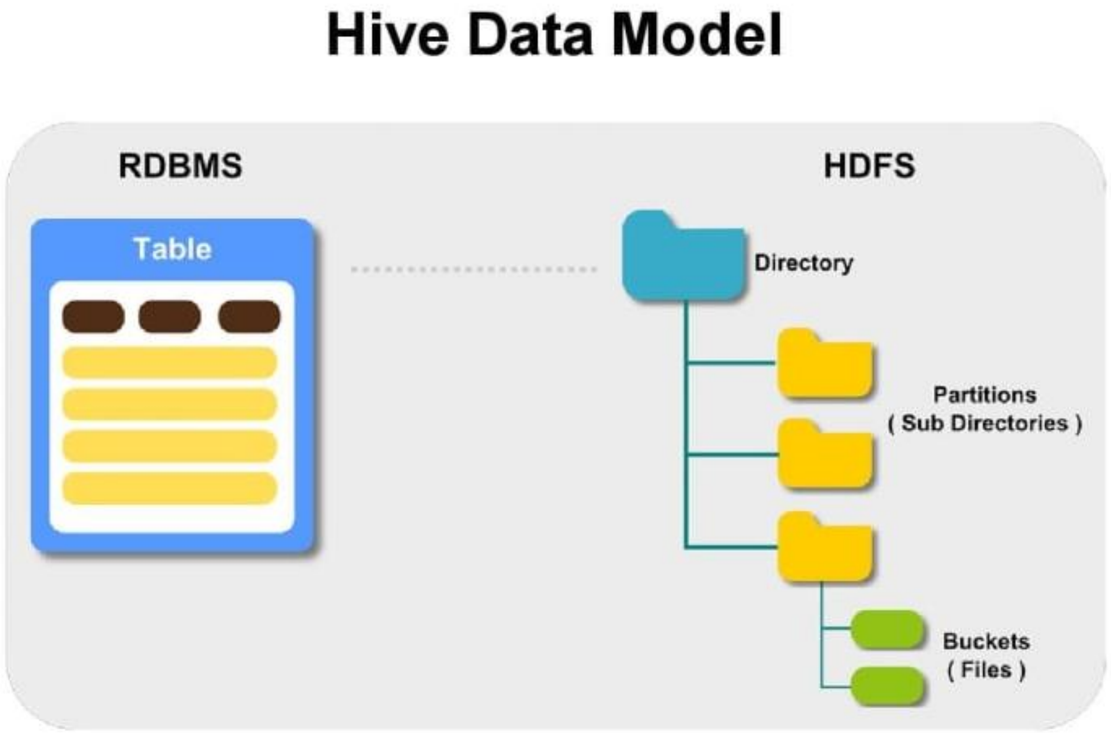
- Open-source, **row-based** storage and serialization format
  - Can be used for file storage or message passing
- Beneficial for write-intensive workloads
- Format contains a mix of text and binary
  - Data definition: Text-based JSON
  - Data blocks: Binary
- Robust support for schema evolution
  - Handles missing/added/changed fields
- Language-neutral data serialization
  - APIs included for Java, Python, Ruby, C, C++, and more



Source: https://www.oreilly.com/library/view/operationalizing-the-data/9781492049517/ch04.html

What are Hive tables?

- **Apache Hive** was introduced in 2010 to provide a data warehouse-like structure on top of **Hadoop**
- Supports the distributed analysis of large datasets in Hadoop's **HDFS**, as well as S3-compatible object storage
- SQL-like **HiveQL** queries are converted to **MapReduce** jobs
- **"Schema on read"** enforces structure at query time
- Tables are just **"data files in directories"** – supporting plain text, ORC, RCFile, Parquet, and other formats
- **Metadata store** (HMS) component tracks metadata such as schema and location
- No concurrency control, inefficient updates/deletes, and schema changes require rewriting entire dataset



Source: https://dev.to/aws-builders/introduction-to-hivea-sql-layer-above-hadoop-kk1

Table management and formats

Sits “above” the data file layer

Organizes and manages table metadata and data

Typically supports multiple underlying disk file formats (Parquet, Avro, ORC, etc.)

May offer transactional concurrency, I/U/D, indexing, time-based queries, and other capabilities

- Open-source
- Designed for large, petabyte (PB)-scale tables
- ACID-compliant transaction support
- Capabilities not traditionally available with other table formats, including schema evolution, partition evolution, and table version rollback – all without re-writing data
- Advanced data filtering
- Time-travel queries let you see data at points in the past

- Open-source, but Databricks is primary contributor and user, and controls all commits to the project – so “closed”
- Foundation for storing data in the Databricks Lakehouse Platform
- Extends Parquet data files with a file-based transaction log for ACID transactions and scalable metadata handling
- Capabilities include indexing, data skipping, compression, caching, and time-travel queries
- Designed to handle batch as well as streaming data

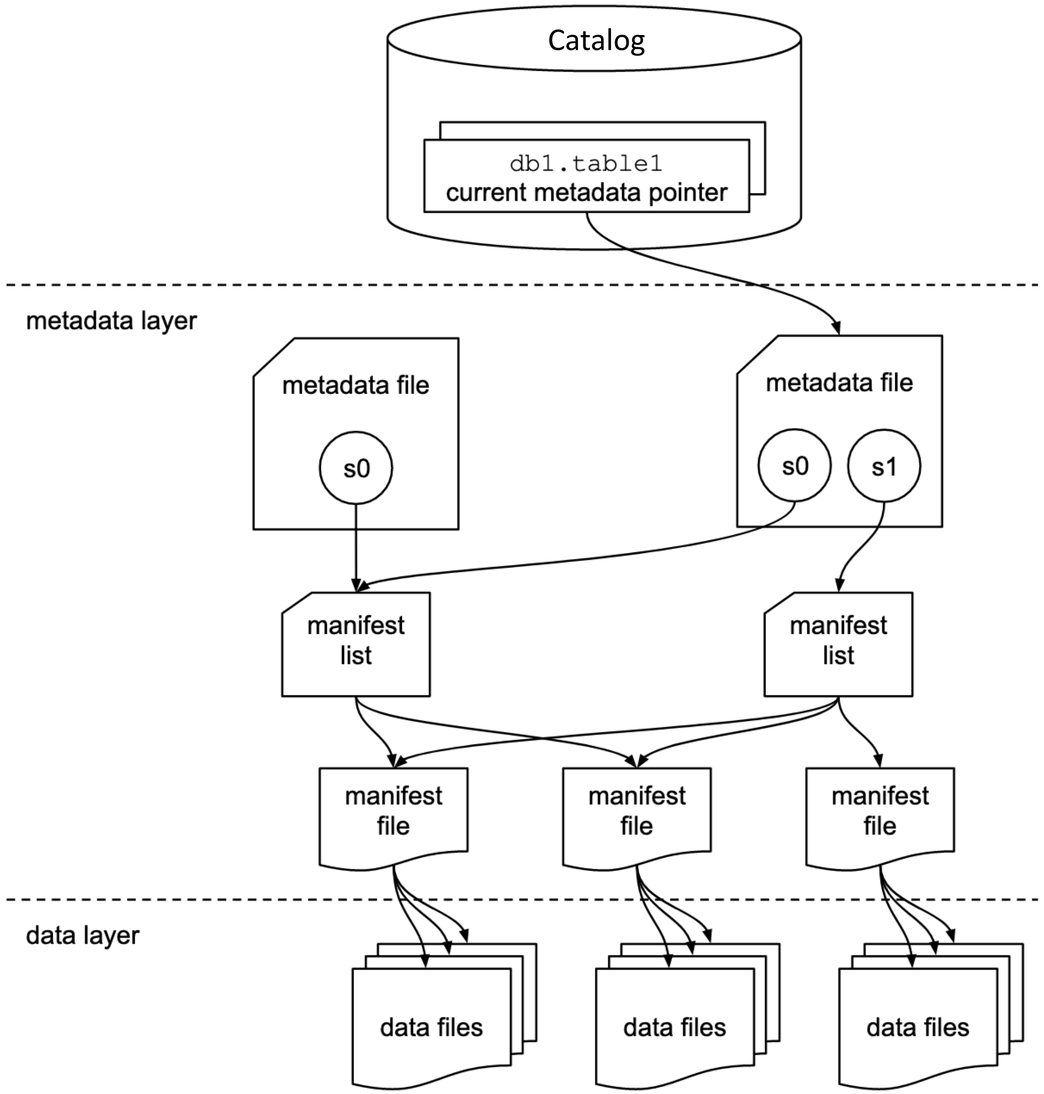
- Open-source
- Manages the storage of large datasets on HDFS and cloud object storage
- Includes support for tables, ACID transactions, upserts/ deletes, advanced indexes, streaming ingestion services, concurrency, data clustering, and asynchronous compaction
- Multiple query options: snapshot, incremental, and read-optimized

Apache Iceberg open data table format

Open-source data table format that helps simplify data processing on large dataset stored in data lakes

People love it because it has:

- SQL access — Build the data lake and perform most operations without learning a new language
- Data Consistency — ACID compliance (not just append data operations to tables)
- Schema Evolution — Add/remove columns without distributing underlying table structure
- Data Versioning — Time travel support that lets you analyze data changes between update and deletes
- Cross Platform Support — Supports variety of storage systems and query engines (Spark, Presto, Hive, +++)



Source: https://iceberg.apache.org/spec/

What is a metastore?

- Manages metadata for the tables in the lakehouse, including:
  - Schema information (column names, types)
  - Location and type of data files
- Similar in principle to the system catalogs of a relational database
- Shared metastore ensures query engines see schema and data consistently
- May be a built-in component of a larger integration/governance solution

HMS used by watsonx.data

- Hive metastore (HMS) is a component of Hive, but can run standalone
- Open-source
- Manage tables on HDFS and cloud object storage
- Pervasive use in industry

AWS Glue Data Catalog

- Component of AWS Glue integration service
- Inventories data assets of AWS data sources
- Includes location, schema, and runtime metrics

Microsoft Purview Data Catalog

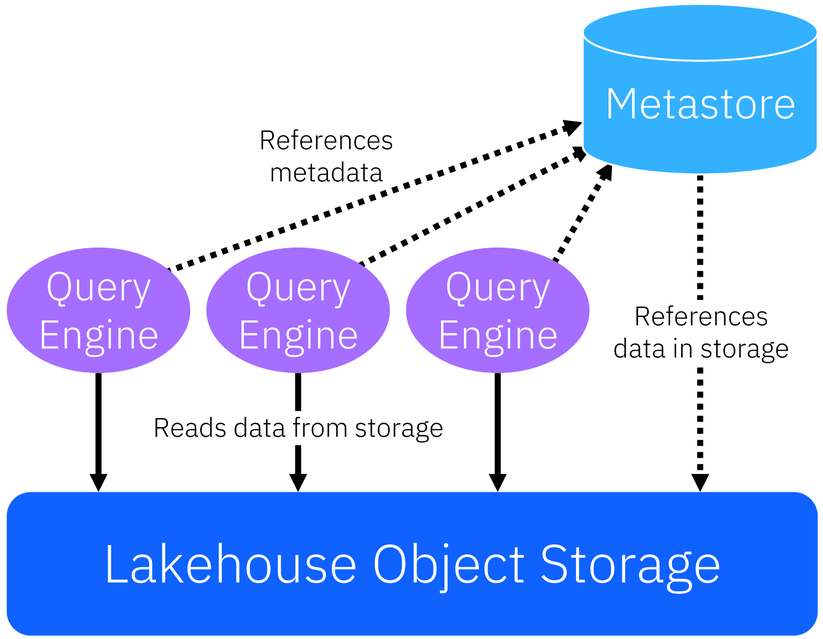
- Component of Microsoft Purview data governance solution
- Helps manage on-premises, multicloud, and SaaS data
- Offers discovery, classification, and lineage

Databricks Unity Catalog

- Provides centralized access control, auditing, lineage, and data discovery across a Databricks lakehouse
- Contains data and AI assets including files, tables, machine learning models, and dashboards

Hive Metastore (HMS)

- Open-source **Apache Hive** was built to provide an SQL-like query interface for data stored in Hadoop
- **Hive Metastore (HMS)** is a component of Hive that stores metadata for tables, including schema and location
- HMS can be deployed standalone, without the rest of Hive (often needed for lakehouses, like watsonx.data)
- Query engines use the metadata in HMS to optimize query execution plans
- The metadata is stored in a traditional relational database (PostgreSQL in the case of watsonx.data)
- In watsonx.data, IBM Knowledge Catalog integrates with HMS to provide policy-based access and governance



presto

Make sense of all your data, any size, anywhere

Get the insights you need with Presto, a fast and flexible open-source SQL query engine

Scalable architecture

- Designed for analytic queries
- Uses open source query engines

Pluggable Connectors

- Allows access to external data sources without moving data
- Wide variety of connector for cloud and on- premises data sources

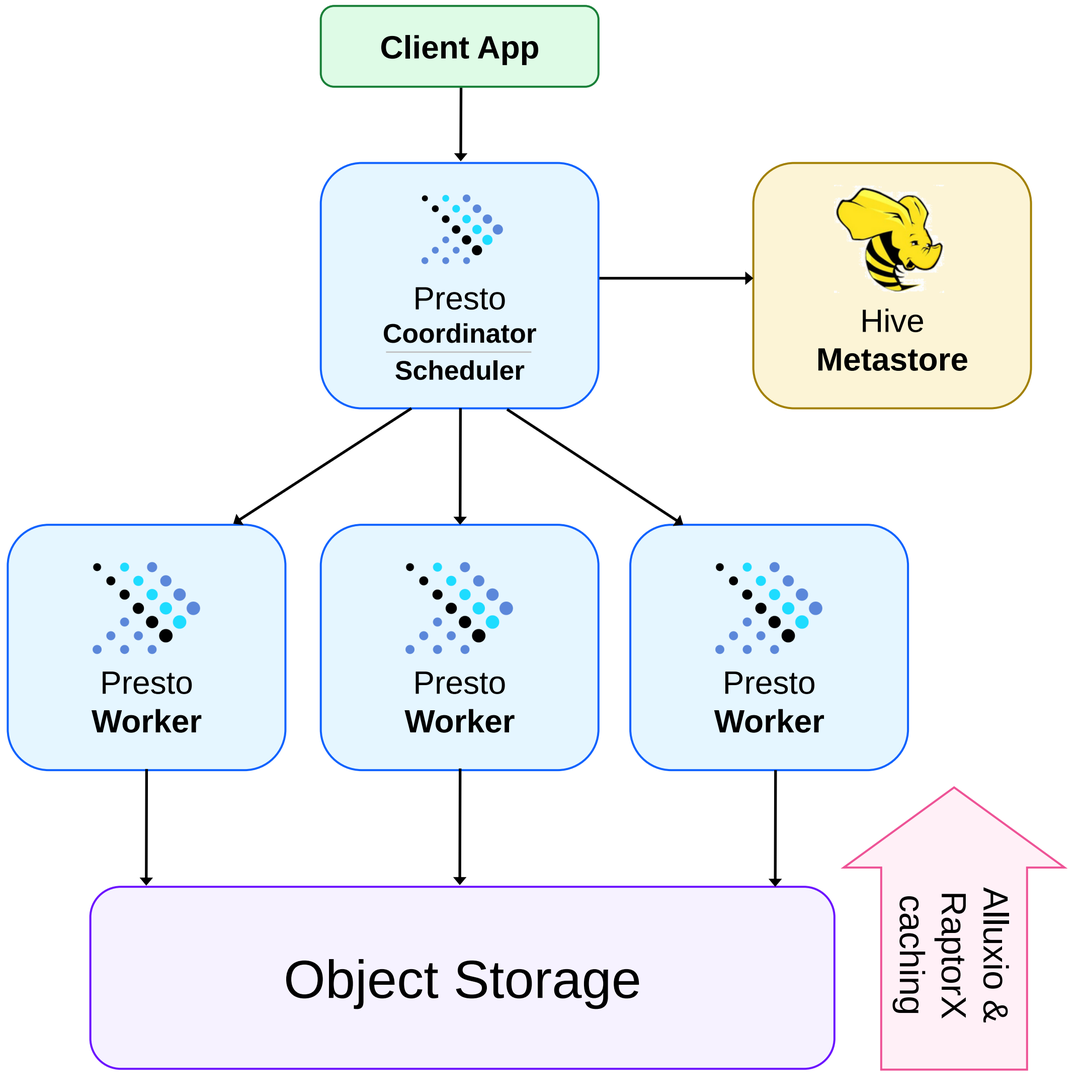
Performance

- MPP architecture for processing large data sets
- Can scale worker nodes as needed

Presto architecture

The structure of Presto is similar to that of classical MPP database management systems.

- **Client:** Issues user query and receives final result.
- **Coordinator:** Parses statement, plans query execution, and manages worker nodes. Gets results from workers and returns final result to client.
- **Workers (Java/C++):** Execute tasks and process data.
- **Connectors:** Integrate Presto with external data sources like object stores, relational databases, or Hive.
- **Caching:** Accelerated query execution through metadata and data caching (provided by Alluxio and RaptorX).



Powered by

presto

Adobe

Digital advertising platform

Over 2000 daily reports and 100s of pipelines on a 7 PB data lake with over 400 billion records

Bolt

Ride-hailing, micromobility rentals, and food delivery in Europe and Africa

Up to 100,000 daily queries (over 1.5 million queries per month) with over 2000 active internal users on 2 PB data lake

Meta

Social media

30,000 queries per day with 1000 daily active users on a 300 PB data lake

Uber

Ride-hailing, food delivery

Over 100 million queries per day with 7000 weekly active users on a 50 PB data lake

ByteDance

Internet technology

Over 2 million queries per day for business intelligence and one-off use cases

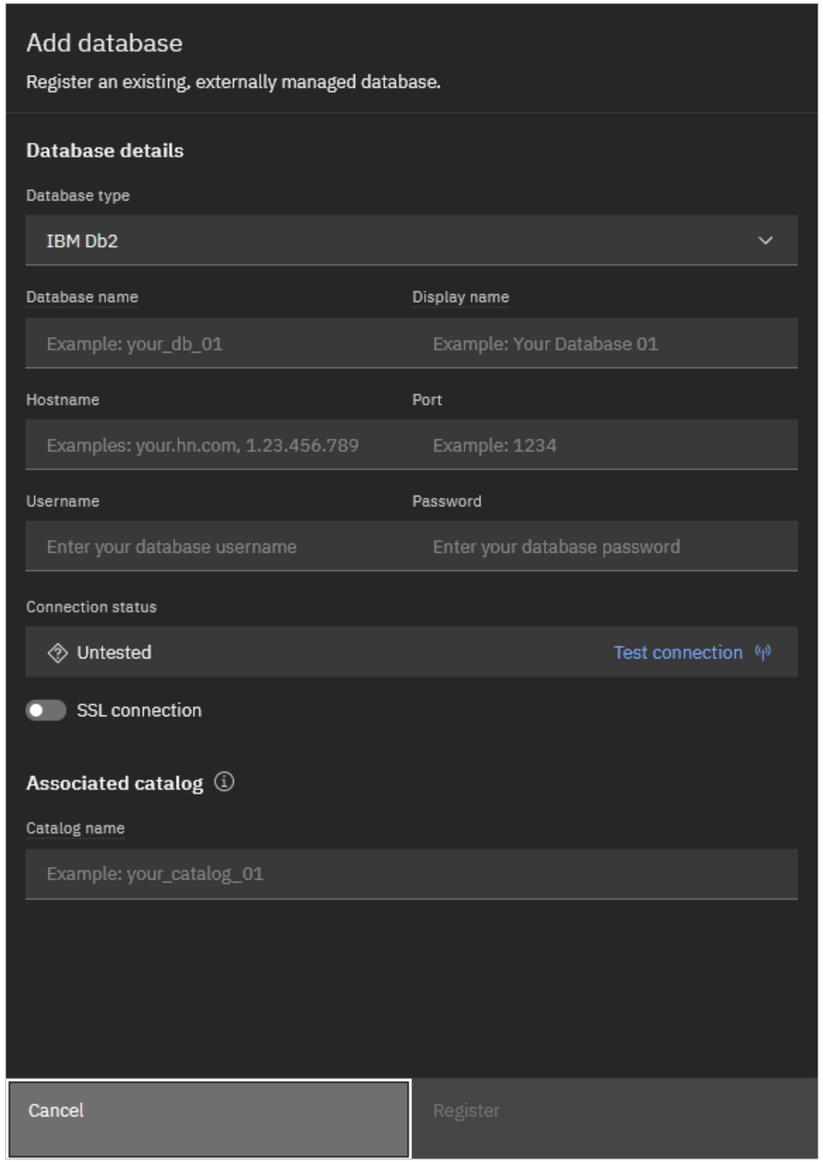
twilio

Communications API technology

Over 2700 active internal users running 1 million queries scanning 40 PB of data per month

Presto connectors in watsonx.data for federated data access

- IBM Db2
- IBM Netezza
- IBM Data Virtualization Manager for z/OS
- IBM Informix
- Apache Druid
- Apache Kafka
- Apache Pinot
- Amazon Redshift
- BigQuery
- Cassandra
- ClickHouse
- Elasticsearch
- MongoDB
- MySQL
- Oracle
- PostgreSQL
- Prometheus
- Redis
- SAP HANA
- SingleStore
- Snowflake
- SQL Server
- Teradata
- *... with more to come*



watsonx.data

Use Cases

Data warehouse optimization

Optimize workloads from your data warehouse by choosing the right engine for the right workload, at the right cost. Replace ETL jobs and reduce costs of your data warehouse by up to 50% through workload optimization.

Data lake modernization

Augment Hadoop data lakes using watsonx.data and access better performance, security, and governance, without migration or ETL

Mainframe data for AI

Unleash the power of mainframe data for AI and analytics in watsonx.data with integration to IBM Data Gate for watsonx and Data Virtualization Manager for z/OS. Readily virtualize or replicate data to Iceberg for analytics and AI.

Datastore for Generative AI

Unify, curate, and prepare data efficiently for AI models and applications. Integrated vectorized embedding capabilities enable RAG use cases at scale across large sets of your trusted, governed data.

Generative AI powered data insights

Leverage Gen-AI infused in watsonx.data to find and understand data and unlock new data insights through semantic search — no SQL required. Unleash cryptic structured data using auto-generated semantic metadata in natural language for easy self-service access to data.

Investments in an open and trusted data foundation will accelerate and scale your AI initiatives

Databases

Store, unify, and share data with open formats and native integrations

Database
IBM Db2
IBM Informix

Data Warehouse
IBM Db2 Warehouse
IBM Netezza

Data Lakehouse
IBM watsonx.data

3rd Party Databases
SingleStore, C, mongoDB., PostgreSQL, DATASTAX, snowflake, aws, Any many more

Data Fabric

Simplify data access across the enterprise and hybrid cloud environments with a data fabric architecture

Data Governance
IBM Knowledge Catalog
Manta Data Lineage

Data Integration
IBM DataStage
IBM Data Replication
IBM Data Virtualization

Data as a Product
Delivering data products

Data Observability
IBM Databand

Master Data Management
IBM Master Data Management

IBM Cloud Pak for Data

Data
Delivering quality data to consumers and users

AI
Enabling data teams to be more productive with automated insights

watsonx™

Scale and accelerate the impact of AI with trusted data

AI and Data Platform

watsonx.ai
watsonx.data
watsonx.governance

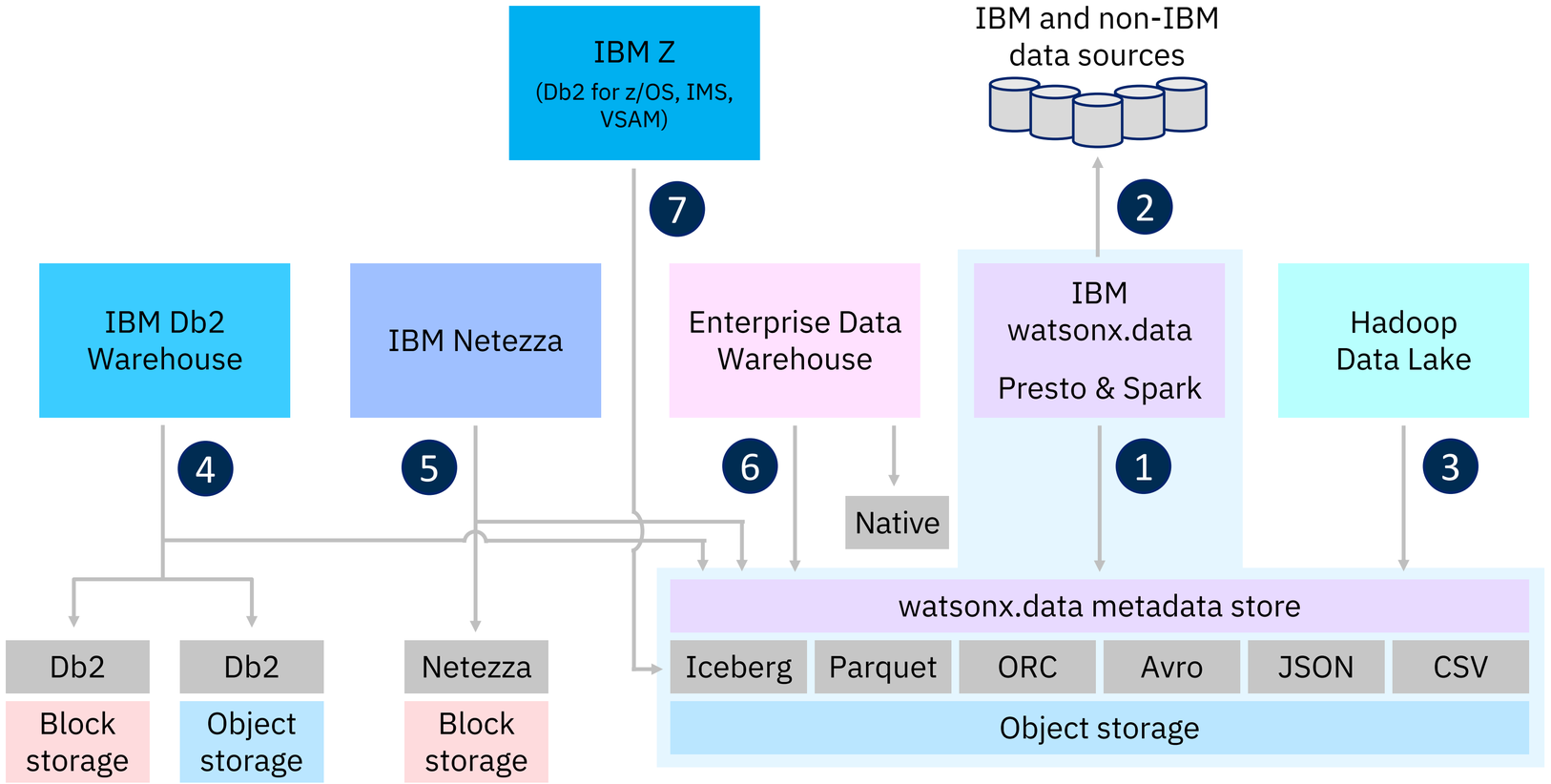
AI Assistants

watsonx Orchestrate
watsonx Assistant
watsonx Code Assistant

Automated data lineage

Gain deeper visibility into your data and its journey from source to end-use for regulatory compliance and AI use cases with Manta, an IBM company.

IBM watsonx.data and your data ecosystem



1. Watsonx.data's native **Presto** and **Spark** engines work with open data and table formats.
2. Presto's connectors allow for **federated data access** to many different data sources, without having to move or copy data.
3. Sync metadata with watsonx.data. Convert legacy file storage structures to Iceberg (over time).
4. **Natively store Db2 WH data in open data formats.** Offload/promote between Db2 WH and watsonx.data.
5. Access lakehouse data **natively through Netezza**.
6. For DWs that "speak" Iceberg, offload data/workloads to lakehouse.
7. With Data Gate for watsonx, **replicate mainframe transactional data to Iceberg**, where it can be used for analytics and AI workloads.

Data Gate for watsonx (different than Data Gate for Db2 for z/OS)

Use cases

- For analytics and ML in watsonx.data
- Combination with Spark streaming for streaming queries or writing to further data stores

Minimal operational impact on Db2/z

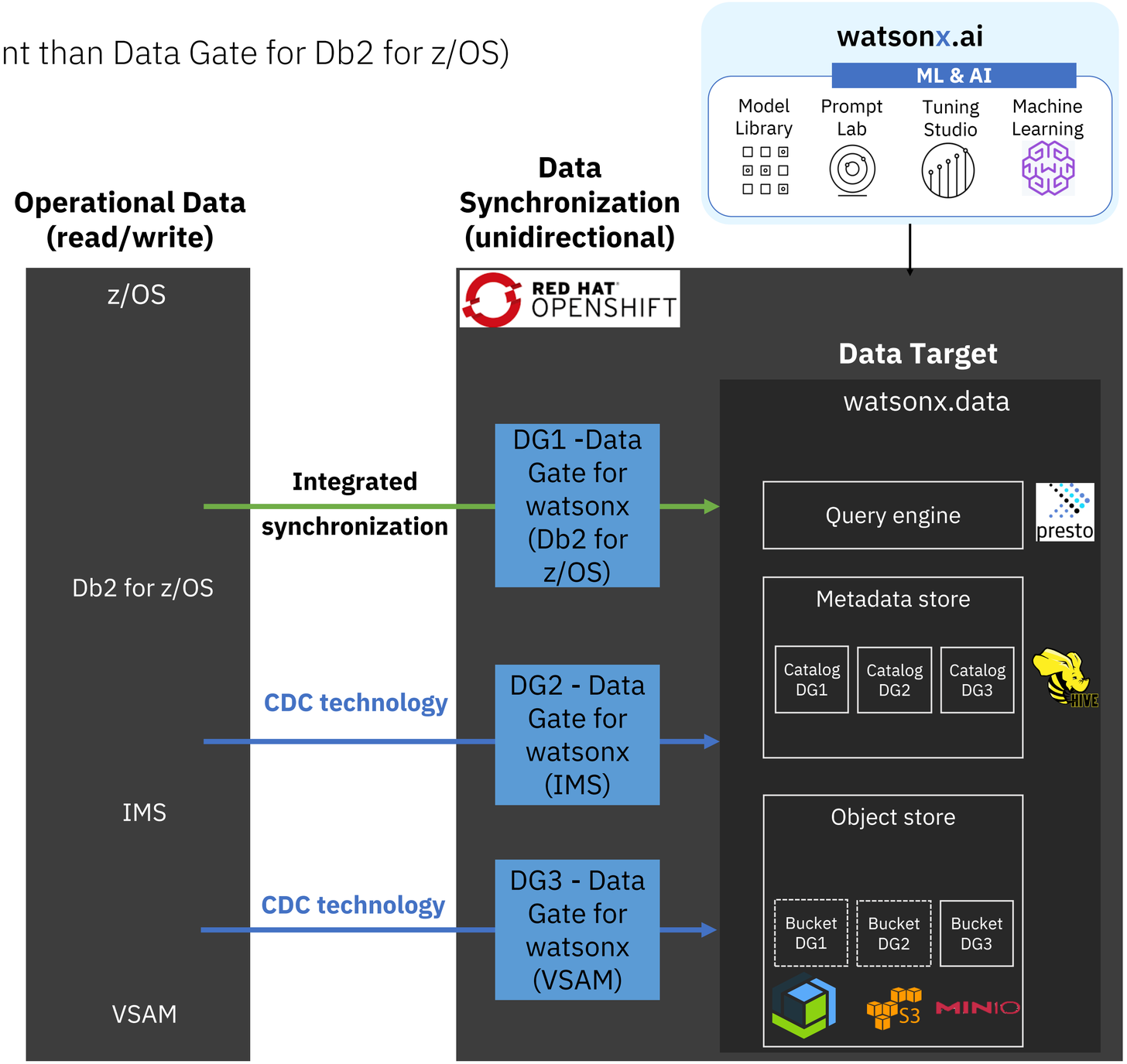
- Working on cached copy
- zIIP-eligible integrated synchronization workload

High performance requirements

- Low-latency data synchronization protocol
- High performance ingest into Iceberg

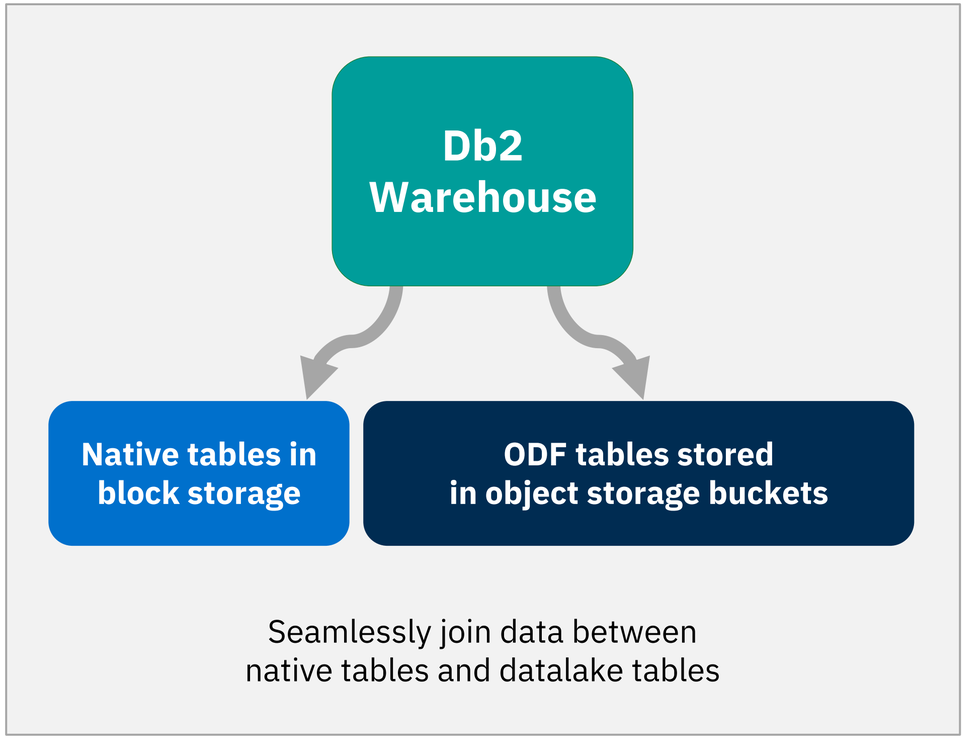
With…

- End-to-end encryption
- Hides CDC complexities
- Containerized, OpenShift-based installation (for non z/OS parts) – runs where watsonx.data runs



Db2 Warehouse DATALAKE tables (doesn't require watsonx.data)

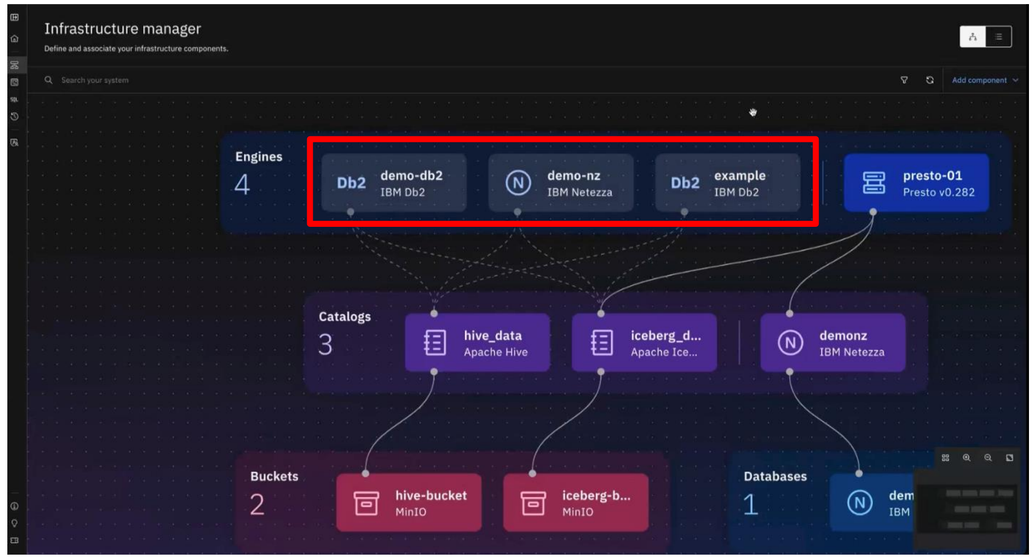
- Work with Db2 data in open data & table formats (e.g. Parquet, Iceberg) hosted on low-cost object storage
- Optimize resources by segmenting workloads across the warehouse and other datalake/lakehouse engines
- Seamlessly combine warehouse data with enterprise lakehouse data
- Export Db2 warehouse data to object storage (e.g. CTAS), while retaining the ability to query that data
- Use a datalake engine (e.g. Spark) to cleanse and transform data; then bring that curated data into Db2

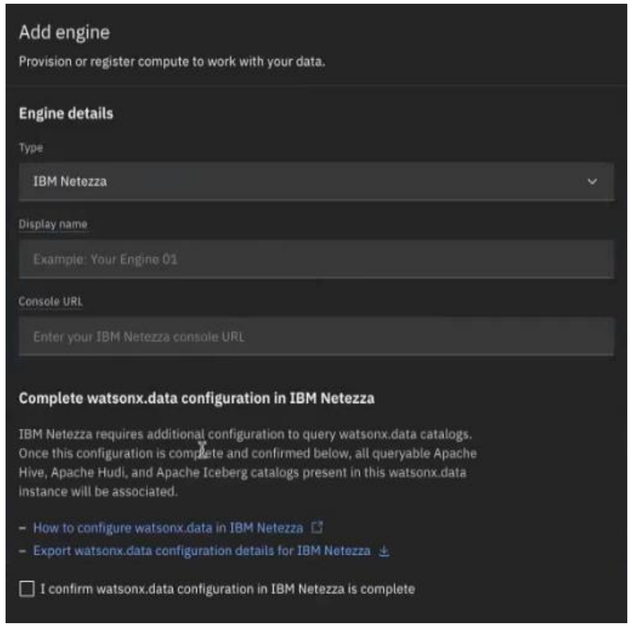


Db2 Warehouse and Netezza integration with watsonx.data

- Functionality in Db2 WH and Netezza (NPSaaS):
  - Ability to work with open data format tables in object storage (e.g. Db2's DATALAKE tables)
  - Integration with watsonx.data's metastore (w/ syncing of metadata for tables in object storage)

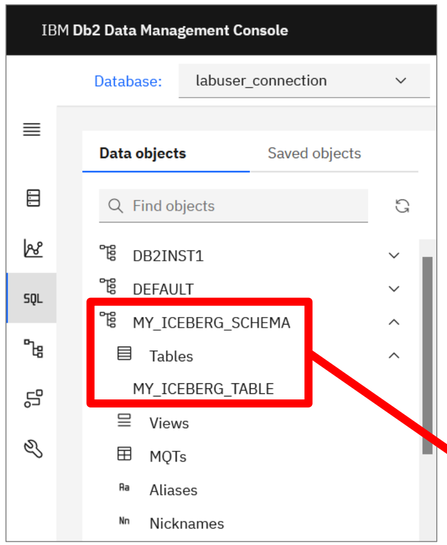
- Db2 WH and Netezza can be registered as "External Engines" in the watsonx.data console



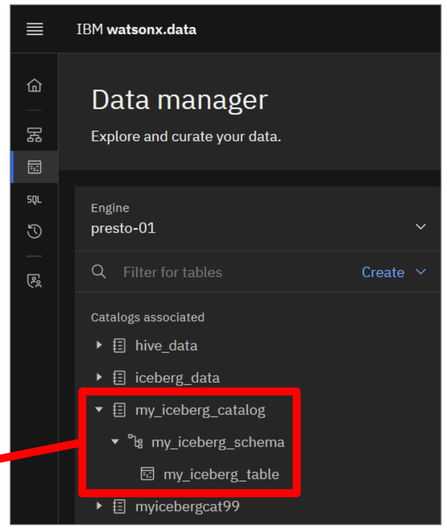


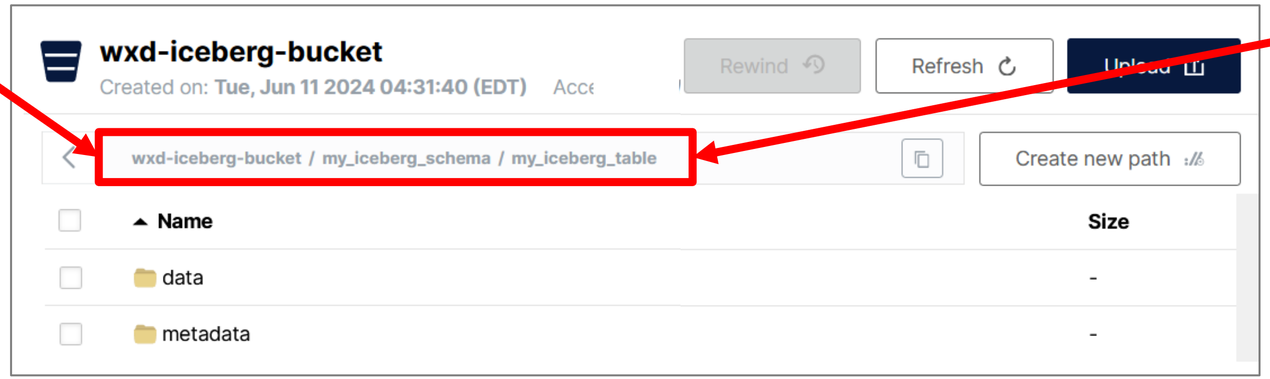
Db2 Warehouse (lakehouse tables)

watsonx.data

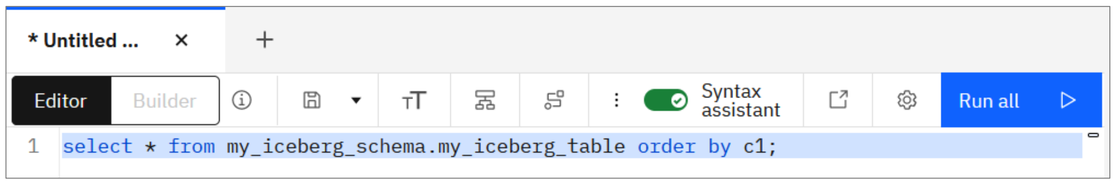


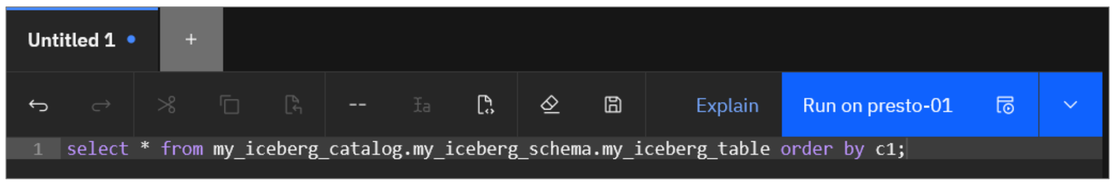
and also...



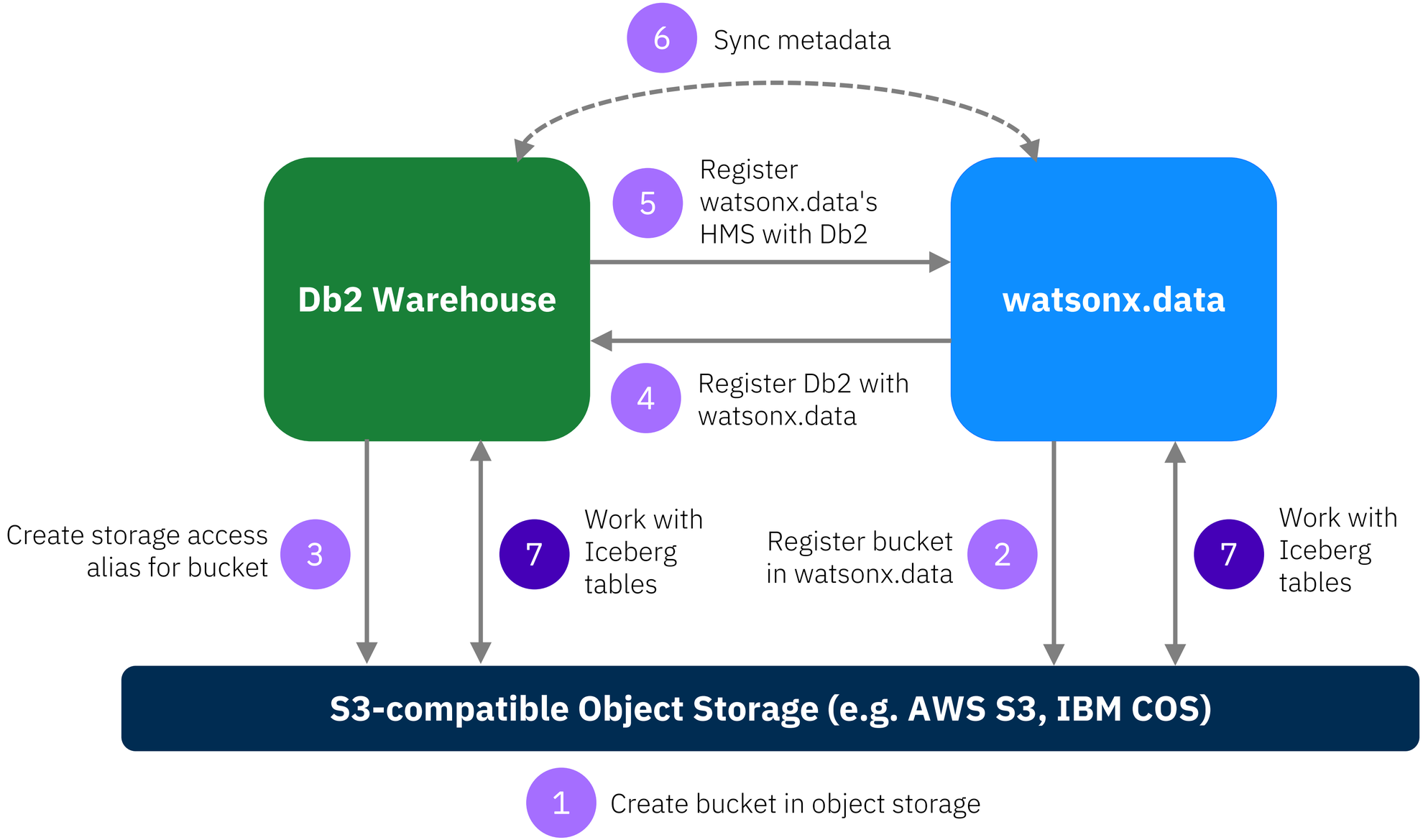


Concurrent access
to the same data





Configuring watsonx.data and Db2 Warehouse



watsonx.data is helping companies scale their AI workloads

intel

“We look forward to partnering with IBM to optimize the watsonx.data stack and contributing to the open-source community.”

Das Kamhout
VP and Senior Principal Engineer
Intel

capital bank

"We’re excited to see how watsonx can help us drive predictive analytics, identify fraud, and optimize our marketing.”

Bahaa’ Awartany
Chief Data Officer
Capital Bank of Jordan

CLOUDERA

“Customers will benefit from a truly open and interoperable hybrid data platform that fuels the adoption of AI.”

Paul Codding
EVP of Product Management
Cloudera

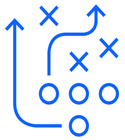
NucleusTeq

“We believe watsonx.data will help enterprises lower storage costs, optimize compute, and ensure seamless data management.”

Ashish Baghel
CEO and Founder
NucleusTeq

Three ways to get started with watsonx.data today

IBM’s investment in partnering with clients



Free trial

Experience watsonx.data and test out core capabilities with the free "Lite" plan.

[Try the free Lite plan]

Client briefing

Discussion and custom demonstration of IBM’s generative AI watsonx point-of-view and capabilities. Understand how watsonx.data can be leveraged in any businesses AI strategy.

2-4 hours

Pilot program

Watsonx pilot developed with IBM AI engineers. Prove watsonx.data value for the selected use case(s) with a plan for adoption.

1-4 weeks

For those staying for the hands-on lab:

- If you don't already have an **IBM ID**, please sign up for one:

https://ibm.biz/ibm-id-signup