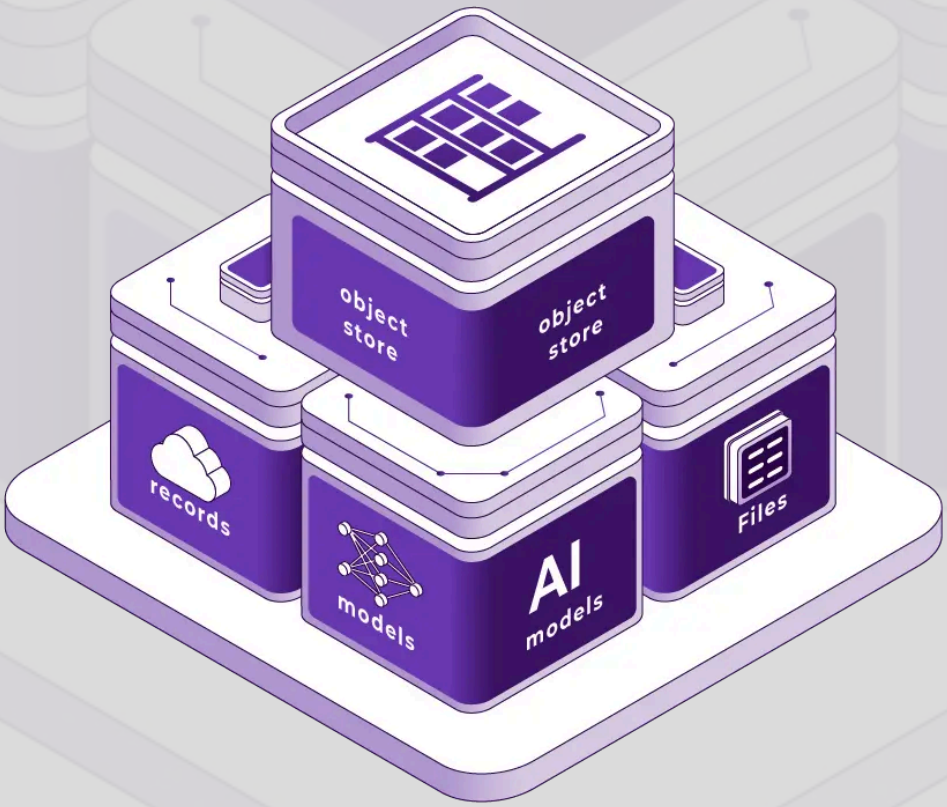
ReductStore

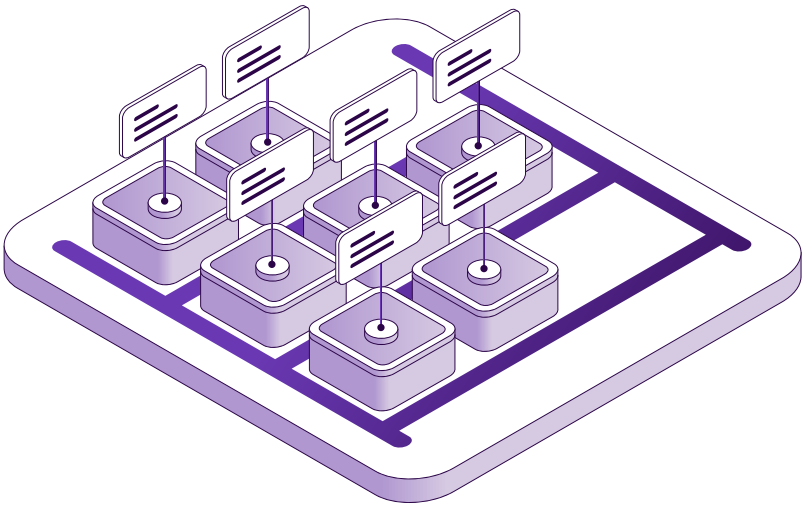
High Performance Storage and Streaming Solution for Data Acquisition Systems



Abstract

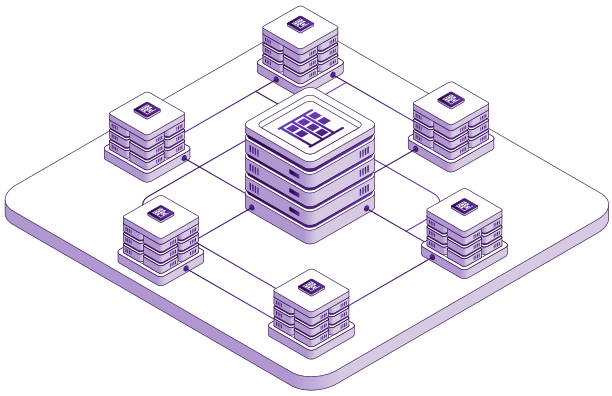
This white paper explores various data storage solutions, focusing on their suitability for different applications, particularly in the context of time-series data in robotics and the Industrial Internet of Things (IIoT) for Data Acquisition (DAQ) systems. Traditional time series databases like InfluxDB, OpenTSDB, and TimescaleDB are highlighted for their strengths in handling structured data with high write throughput and complex queries. InfluxDB, for instance, is noted for its analytics and real-time monitoring capabilities, while OpenTSDB excels in long-term storage and analysis of massive datasets thanks to flexible querying capabilities. This paper explores solutions that support unstructured data for robotics, IIoT, and more broadly, edge computing and artificial intelligence (AI) applications. A major challenge in such networks is to effectively manage the growing volume of data from many sources and different forms of time series data (multimodal data) to meet the performance requirements of these applications. Time series data management is crucial for optimizing operations and has emerged as an important area of academic and industrial research.

The main objective of this work is to evaluate and challenge the status quo of current data management methods used in the robotics and IIoT domains, with a particular focus on unstructured time-series data.



Introduction:

Every day, enormous volumes of raw data are being produced due to the continuing expansion of many applications, including robotics, industrial monitoring, surveillance systems, and others.



No matter what kind of intelligence is used - AI, robotics - people are incredibly excited about "smart" things in their cars, homes, or even appliances. Data structured into simple and understandable ideas with AI is helping companies discover new business opportunities in the exabytes of data generated by edge devices.

In terms of industry, the IoT trend has given rise to a specific part of the IoT market called the industrial Internet of Things (IIoT) or Industry 4.0. Industry 4.0 refers to the fourth industrial revolution, characterized by the integration of advanced technologies such as AI, robotics, and the IoT into industrial processes. To get an idea of the scale, it was already predicted in 2020, that Industry 4.0 and IoT would connect over 30 billion devices and sensors worldwide to the internet [1].

Countless sensors embedded in a wide variety of IoT devices provide massive amounts of data at an estimated compound annual growth rate (CAGR) of 26% [2]. Data management is particularly important for performing machine learning (ML) on IoT data, and it should be done both in the cloud and at the edge to reduce latency and network traffic [3]. As a result, time series management is now subject to new obligations.

Simultaneously, 80% of the data available is unstructured, with a significant portion being generated in real-time [4]. However, only 18% of organizations are presently capable of making use of this opportunity [5].

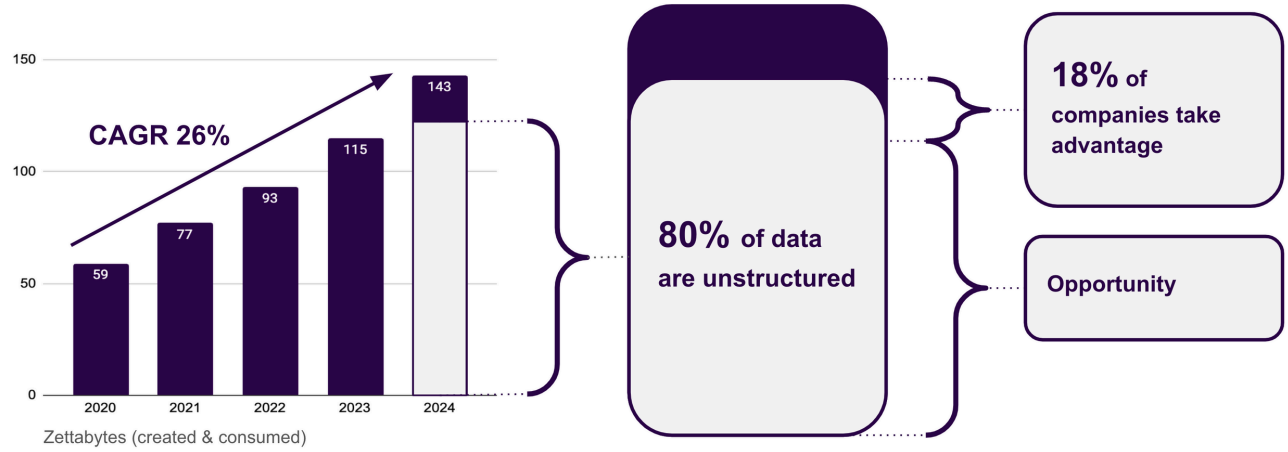
[1] Munirathinam, Sathyan. "Industry 4.0: Industrial internet of things (IIOT)." In Advances in computers, vol. 117, no. 1, pp. 129-164. Elsevier, 2020.

[2] IDC. "IDC's Global DataSphere, 2021," 2021.

[3] Ghosh, Ananda Mohon, and Katarina Grolinger. "Edge-cloud computing for Internet of Things data analytics: Embedding intelligence in the edge with deep learning." IEEE Transactions on Industrial Informatics 17, no. 3 (2020): 2191-2200.

[4] Mishra, Suyash, and Anuranjan Misra. "Structured and unstructured big data analytics." In 2017 International Conference on Current Trends in Computer, Electrical, Electronics and Communication (CTCEEC), pp. 740-746. IEEE, 2017.

[5] Deloitte. "2019 Survey by Deloitte: Analytics and AI-driven enterprises thrive in the Age of With," 2019.



Data growth and organizational utilization of unstructured data

The rise of robotics and IIoT has created a growing need for durable time-series data storage that can be queried efficiently over time. While traditional time-series databases are commonly used for this purpose, cloud-based solutions can become prohibitively expensive due to their complexity and ongoing operational costs [6]. With the increasing computational power and memory available at the edge, there is a strong opportunity to shift time-series storage and analytics closer to where data is generated—improving performance, reducing latency, and lowering costs.

For data-intensive applications, Binary Large Objects, or Blobs, are a storage paradigm that is becoming more and more common [7]. At the same time, a comprehensive amount of control over the data is afforded to the user by the low-level, binary access mechanisms that Blob delivers. This makes it possible to optimize for individual applications, something that structured storage systems like relational databases or key-value stores can't do.

---

[6] Wang, Zhiqi, and Zili Shao. "TimeUnion: An Efficient Architecture with Unified Data Model for Timeseries Management Systems on Hybrid Cloud Storage." In Proceedings of the 2022 International Conference on Management of Data, pp. 1418-1432. 2022.

[7] Matri, Pierre, Alexandru Costan, Gabriel Antoniu, Jesús Montes, and María S. Pérez. "Týr: Efficient Transactional Storage for Data-Intensive Applications." PhD diss., Inria Rennes Bretagne Atlantique; Universidad Politécnica de Madrid, 2016.

Time-Series Databases and Blob Storage:

Modern data architectures are increasingly prioritizing two complementary storage paradigms to address different workload requirements: time-series databases (TSDBs) for high-frequency, analytics-intensive, time-stamped data, and blob storage for scalable storage of unstructured or semi-structured data.

Time-Series Databases:

TSDBs are optimized for high-frequency, time-indexed data (e.g., IoT sensor feeds, telemetry, financial data). Modern TSDBs such as QuestDB, TimescaleDB, and InfluxDB combine SQL-like querying with specialized time-series optimizations:

- High-throughput writes and queries: Efficiently ingest and query large volumes of time-stamped data using columnar storage.
- Built-in analytics: Support for aggregations (e.g., downsampling, moving averages) and integrations with tools such as Grafana for real-time visualization.
- Data Compression: Leverage domain-specific encoding (e.g., delta-encoding, Gorilla compression for floats) to reduce storage costs.

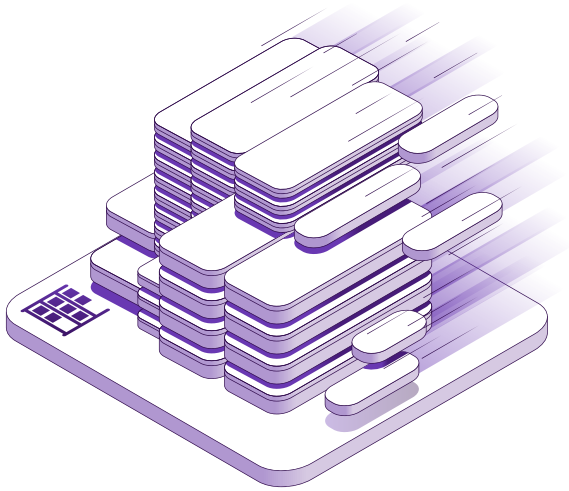
Blob Storage:

For unstructured or semi-structured data (e.g., logs, binaries, Parquet/JSON files), cloud-native blob storage (GCS, AWS S3, Azure Blob Storage) coupled with open formats provides:

- Scalability and cost-effectiveness: scalable storage at a lower cost than traditional block storage with pay-as-you-go pricing.
- SQL analytics on raw data: Tools such as AWS Athena, BigQuery, and Snowflake allow you to query external tables stored in blob storage with SQL access.
- Vendor-agnostic access: Open formats (e.g., Apache Parquet, Iceberg) avoid lock-in and simplify cross-platform data sharing.

Analysis of Existing Solutions:

InfluxDB [11] is largely acknowledged to be the most prominent time-series database globally [12]. A sequential series is used to store data points that have timestamps attached to them. Both fields and tags can be included in a data point's composition. Tags are optional, but fields are required for every data point. Furthermore, InfluxDB supports measurement-specific retention policies to manage data retention duration and replication frequency, allowing users to automate the downsampling and expiration of old data through the combination of retention policies and continuous queries.



An open source core provided by InfluxData is free and includes Telegraph, InfluxDB, Chronograph, and Kapacitor (referred to as the TICK stack) [13]. InfluxDB is schemaless, meaning it does not require a predefined schema. However, it is not designed for traditional CRUD operations. As a time-series database, it is optimized for fast data ingestion and querying, but not for frequent updates and deletions.

OpenTSDB [14], built on top of HBase, scales well horizontally and uses tags to increase data dimensionality but provides millisecond resolution, which may not be suitable for high-frequency real-time applications. In contrast, TimescaleDB [15], an extension of PostgreSQL, breaks tables into smaller chunks, processes data in a column-oriented style, and can compress data, reducing storage and allowing individual column management.

MongoDB [16] is a flexible NoSQL database that stores data in JSON-like documents, ideal for real-time analytics and content management. It supports indexing, replication, and sharding for scalability. However, it may not perform well with high write volumes and complex transactions, and is less optimized for time-series data than dedicated solutions.

MinIO [17] is an open source object store that is compatible with Amazon S3. It is characterized by high performance and scalability, suitable for storing large amounts of unstructured data. MinIO supports erasure coding, bit-rot protection, and high throughput, but is not designed for complex data relationships or transactions.

OpenIO [18] is an open source object storage solution designed for flexibility and scalability. It can be deployed in the cloud, on-premises, or in hybrid environments, offers S3 API compatibility, and ensures data integrity through replication and erasure coding.

[11] "InfluxDB," InfluxData. [Online]. Available: https://www.influxdata.com/. [Accessed: 29-Jul-2024].

[12] Jama Mohamud, Nuh, and Mikael Söderström Broström. "Assessing Query Execution Time and Implementational Complexity in Different Databases for Time Series Data." (2024).

[13] Naqvi, Syeda Noor Zehra, Sofia Yfantidou, and Esteban Zimányi. "Time series databases and influxdb." Studienarbeit, Université Libre de Bruxelles 12 (2017): 1-44.

[14] OpenTSDB. [Online]. Available: http://opentsdb.net/. [Accessed: 29-Jul-2024].

[15] TimescaleDB. [Online]. Available: https://www.timescale.com/. [Accessed: 29-Jul-2024].

[16] MongoDB. [Online]. Available: https://www.mongodb.com/. [Accessed: 29-Jul-2024].

[17] MinIO. [Online]. Available: https://min.io/. [Accessed: 29-Jul-2024].

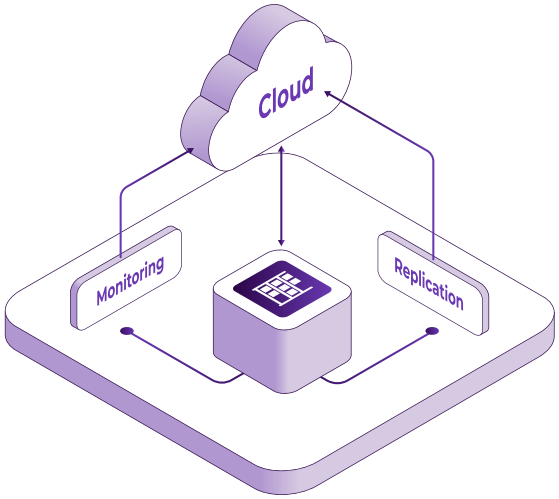
[18] OpenIO. [Online]. Available: https://github.com/open-io/. [Accessed: 29-Jul-2024].

Table 1: Comparison of Traditional System

System	Data Structure	Query Language	Performance	Application
InfluxDB	Structured time series data with tags and fields.	InfluxQL, a SQL-like query language, and Flux, a data scripting language.	Good in high write throughput and complex queries	Best for systems that need analytics and real-time monitoring with structured data.
OpenTSDB	Structured time series data with metrics and tags, based on HBase.	Uses a query language specific to OpenTSDB.	Scales well horizontally, making it ideal for very large datasets.	Suitable for long-term storage and analysis of massive time series datasets.
Timescale DB	Based on the relational data model of PostgreSQL.	Uses SQL for querying, leveraging the full capabilities of PostgreSQL.	Optimized for high ingestion rates and complex queries through hyper tables and chunking.	Ideal for applications needing relational data features along with time series data management.
MongoDB	Uses a flexible, document-oriented data model.	Uses MongoDB Query Language (MQL).	Can handle large datasets and high throughput with appropriate indexing.	Fits flexible schema and different data storage demands, including time series data.
MinIO	Stores data as objects in buckets, similar to AWS S3.	Does not have a native query language but supports APIs for data access (like AWS S3).	Optimized for large-scale object storage with high availability and redundancy.	Ideal for applications needing scalable and resilient object storage.
OpenIO	Stores data as objects with a flat structure.	Provides an API for data access but no native query language.	Scales horizontally and supports high availability and redundancy.	Suitable for cloud storage, big data, and backup solutions.

Challenging the Traditional Systems:

In practical scenarios, time series databases are optimized for structured data with well-defined structures, allowing for efficient querying and analysis. On the other hand, object storage systems are designed to handle unstructured data, offering scalability and redundancy, but lacking the specialized querying capabilities required for time series analysis. This separation means that applications often have to choose between the efficiency of structured time series databases and the flexibility of object storage, without a unified solution that can handle both in tandem.



ReductStore [19] is a specialized system created to effectively handle and store unstructured data that is organized based on time. The system is composed of buckets, entries, blocks, and records. Buckets are receptacles that store data entries and possess configurable attributes such as maximum block size, maximum record count per block, and quota types to govern storage restrictions. ReductStore is designed to manage massive volumes of "blob" data, which is unstructured data. Its adaptability and ease of integration with other applications are due to its support for an HTTP API. Applications in edge computing that are performance-sensitive greatly benefit from ReductStore's design, which prioritizes efficient storage and retrieval of binary data.

The ReductStore team is focused on building a storage and streaming solution for managing unstructured data, with the goal of improving the use of raw data for AI processing and overall performance. ReductStore's employees come primarily from the IIoT and robotics industries, and have built a storage management solution from the ground up.

They give customers and organizations access to a wide range of opportunities, including increased efficiency and cost optimization (especially in the cloud by automatically moving data to cold storage). When compared to TimescaleDB and InfluxDB, ReductStore stands out due to significantly better performance for data sizes of 10kB and above. At the same time, ReductStore's time series API and batching capabilities makes it incomparable to other S3 storage options, such as MinIO.

DAQ System for Robotics and IIoT

Robotic systems and IIoT devices generate large amounts of time-sequenced, unstructured data-from images and sensor readings to event logs and LiDAR. In robotics, data is often collected in formats such as ROS bag files or MCAP, which group multimodal data streams for playback and analysis. A DAQ system is responsible for collecting this data, ensuring reliable acquisition at the edge (shop floor in the image below), and making it available for downstream processing and AI development.

ReductStore enables the creation of such a system by allowing local storage of time-indexed binary data with metadata labels. Data can be streamed across multiple tiers - for example, from a robot or edge device to a plant-level aggregator and then to a central or cloud-based store for long-term retention or analysis. Features such as FIFO quotas, label-based filtering, and batch replication make it easy to manage large volumes of data without overwhelming bandwidth or storage resources.

The system follows an ELT (Extract, Load, Transform) pattern: raw data is captured and stored first, with transformation or filtering applied later as needed. This ensures that the original data is always available for reprocessing, model training, or retrospective analysis-especially valuable when data requirements change over time.

ReductStore provides persistence and reliability by writing data to disk and supporting inter-instance streaming, even after a prolonged loss of connectivity.

This allows selected portions of data (based on metadata labels) to be replicated from an edge instance to a factory level aggregator and then to central or cloud storage without risk of data loss or the need for a continuous network connection.

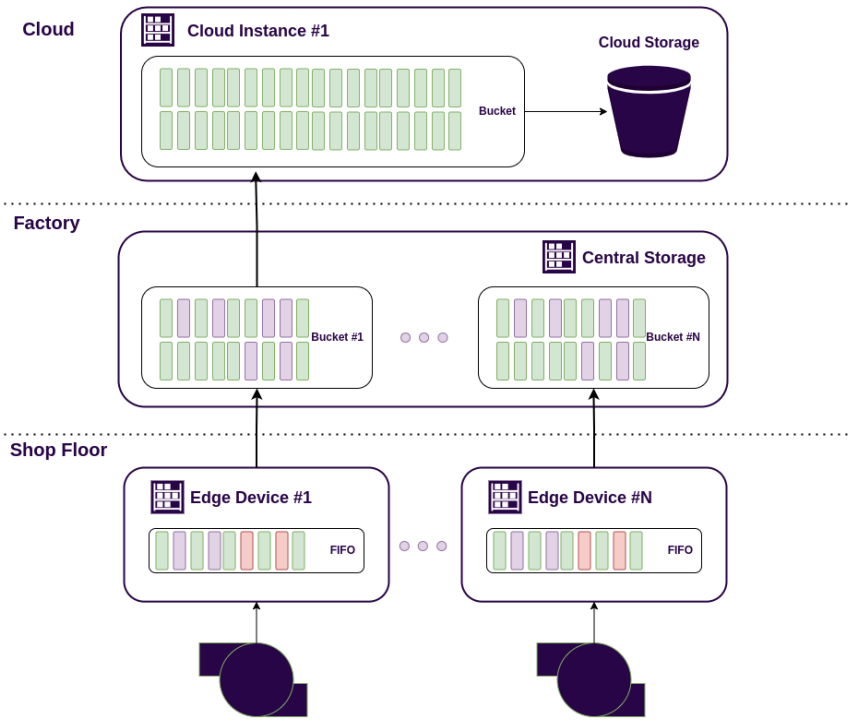
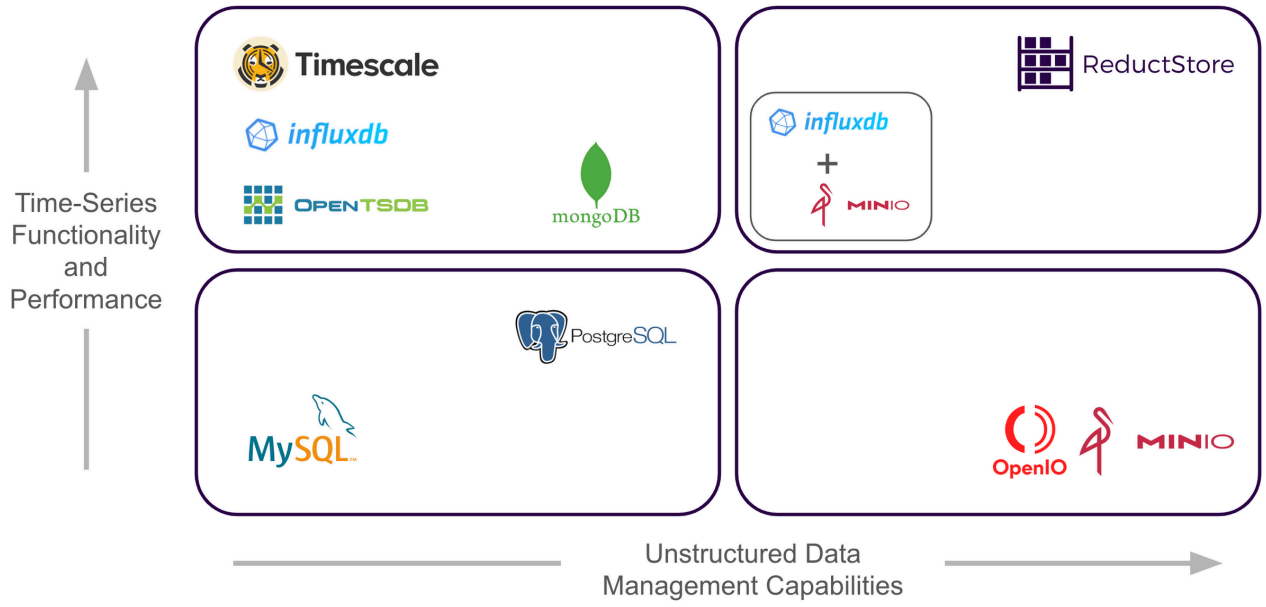


Table 2: Comparative Analysis of ReductStore with Other Solutions [19]

	VS TimeScale		VS MongoDB		VS MinIO	
Record Size	Read Speed (%)	Write Speed (%)	Read Speed (%)	Write Speed (%)	Read Speed (%)	Write Speed (%)
1 MB	+671%	+1604%	-30%	+170%	+291%	+936%
100 KB	+603%	+924%	+260%	+420%	+1552%	+1288%
10 KB	+313%	+297%	+1600%	+850%	+6170%	+1629%
1 KB	+28%	+198%	+2300%	+900%	+8310%	+1400%

ReductStore efficiently manages large datasets with features that optimize both time-series functionality and object storage management capabilities.



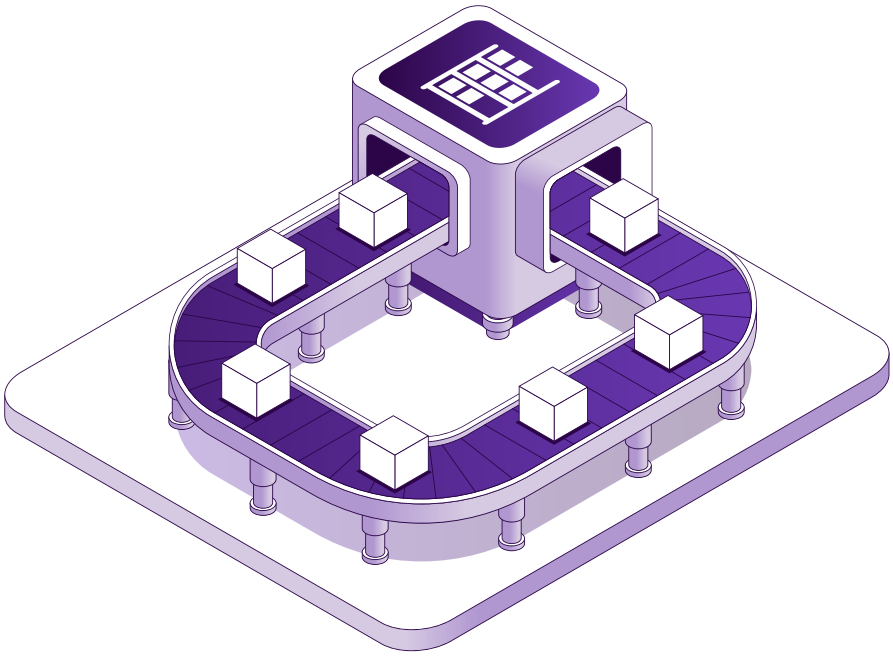
Comparison of database capabilities: Time-Series functionality and performance vs. unstructured data management

[19] ReductStore. [Online]. Available: https://www.reduct.store/. [Accessed: 29-Jul-2024].

Specific features for edge computing:

ReductStore's real-time, first-in, first-out (FIFO) quota system ensures that disk space is optimally managed, preventing shortages that could disrupt operations on edge devices. For example, a strict FIFO quota can be applied to high-frequency vibration data from industrial machinery, image data from computer vision applications, or any complex IoT application where each record is inherently large.

In addition, each record can be associated with labels - typical in AI applications - which are key-value pairs and can hold meta-information about the record. These labels can also be used for filtering and replication. When it comes to network challenges, it also offers a feature to minimize network overhead, especially in high latency environments, by batching data retrieval operations.



Cost Analysis:

> **« ReductStore is a vital part of our infrastructure. It handles terabytes of unstructured data in a production environment. »**
>
> *Michael Welsh, Founder at Metric Space UG*

To illustrate the benefits of ReductStore, let's consider an organization that processes and stores 50 terabytes/month of images on edge devices and in the cloud, with each image averaging 100KB in size. Since 1TB consists of 10 million 100KB images, we can calculate the time to read and write these 10 million images using the following benchmark [20]:

Image Size	Operation	TimescaleDB, image/s	ReductStore, image/s	ReductStore, %
100 KB	Write	491	5026	+924%
100 KB	Read	1,602	11,244	+603%

Write Operations:

TimescaleDB: $\frac{\text{10 million images}}{\text{491 images / sec}} = 5.66 \text{ hours}$

ReductStore: $\frac{\text{10 million images}}{\text{5026 images / sec}} = 0.55 \text{ hours}$

Read Operations:

TimescaleDB: $\frac{\text{10 million images}}{\text{1602 images / sec}} = 1.73 \text{ hours}$

ReductStore: $\frac{\text{10 million images}}{\text{11244 images / sec}} = 0.25 \text{hours}$

Time savings per terabyte:

- Write operation savings: 5.11 hours (5.66 hrs for TimescaleDB - 3.7 hrs for ReductStore)
- Read operation savings: 1.48 hours (1.73 hrs for TimescaleDB - 0.25 hrs for ReductStore)

[20] "Alternative to MongoDB for Blob Data" [Online]. Available: https://www.reduct.store/blog/comparisons/iot/reductstore-vs-mongodb/. [Accessed: 29-Jul-2024].

Reduced edge storage hardware requirements:

ReductStore's blazing speed (10x for writes, 7x for reads vs. TimescaleDB) eliminates the need for costly disk over-provisioning at the edge. For example, storing 1TB of images (10 million 100KB images) in around 30 minutes requires a throughput of ~5,000 images/sec. At TimescaleDB's write speed (491 images/sec), you'd need 11 disks to meet this demand. ReductStore achieves the same result with just 1 disk, thanks to its 5,026 images/sec throughput. We estimate that this 90% reduction in disks reduces hardware costs by $10,000 (assuming that 11 disks require 5 industrial PCs with 2-3 disks/PC at $2,000/PC).

Cloud Storage Savings:

By separating compute and storage, ReductStore leverages low-cost cloud solutions like Google Cloud Storage (~€20/TB/month) and avoids proprietary database fees (€200+/TB/month). For example, storing 50TB/month reduces cloud costs from $10,000 ($200 x 50) to $1,000 ($20 x 50), saving $9,000/month while maintaining scalability.

Total savings:

Combining $10,000 in edge hardware savings with $9,000/month in cloud storage savings, **organizations save $118,000 in the first year, not including additional licensing fees and the complexity of handling multiple disks in parallel**. ReductStore's speed and cost-effective architecture turns storage from a bottleneck into a strategic advantage.

Conclusion:

After careful consideration, we determined that the systems we selected represent the most advanced storage solutions for edge computing. ReductStore's focus on blob data performance, efficient batching, replication and FIFO quota management makes it well suited for applications that require effective storage, streaming and fast retrieval of massive amounts of unstructured data.

ReductStore offers a unique solution tailored for unstructured time-series data, making it a strong contender for robotics and IIoT applications where efficient management of real-time data is critical. Its simplicity, performance optimizations, and focus on binary data along with labels sets it apart from more traditional time-series databases such as InfluxDB, OpenTSDB, and TimescaleDB, as well as object storage solutions such as MinIO and OpenIO.

In the context of modern data acquisition (DAQ) systems, ReductStore provides a practical approach to efficiently ingesting, organizing, and replicating raw data across tiers - from edge instances to factory and cloud infrastructure - supporting scalable, ELT-friendly pipelines and long-term data access.

🔗 LinkedIn - ReductStore

💬 Join us on Discours

✉️ info@reduct.store

🌐 www.reduct.store