

# TDengine in the Oil & Gas Sector

---



# Chapter 1

## Introduction

The oil and gas industry operates in some of the most data-intensive and operationally demanding environments in the world. From upstream exploration and production to midstream transportation and downstream processing, vast volumes of high-frequency operational data are generated across wells, pipelines, compressors, refineries, and supporting infrastructure. Managing this data efficiently – while ensuring reliability, scalability, and real-time accessibility – is critical to maintaining safe and profitable operations.

This white paper explores how TDengine provides a modern, scalable foundation for time-series and operational data management in the oil and gas sector. It examines key architectural considerations, deployment models, and practical use cases, illustrating how TDengine supports real-time monitoring, historical analysis, and AI-driven applications across distributed and mission-critical environments.

### 1.1 Data Challenges in the AI-Driven Transformation

As a strategic pillar for driving high-quality business development and technological innovation, digital and intelligent transformation is a complex systems engineering endeavor. While it brings substantial transformation dividends, it also introduces a range of risks and challenges across management, data, security, talent, and ecosystem dimensions. From a data perspective, the challenges are particularly significant:

#### Challenges of Massive Data Ingestion

Oil & gas operations generate vast volumes of time-series data throughout production and operations. Taking one oilfield company as an example: it operates 35 oilfields and 13 gas fields, with a total of 116,357 oil, gas, and water wells, 2,704 stations, and 17 processing (purification) plants, along with 112,000 kilometers of various pipelines. Its production IoT system includes over 370,000 instruments and meters and more than 120,000 control devices. SCADA systems have been deployed across 119 operational zones, bringing more than 90,000 oil, gas, and water wells online. The planned storage capacity for its cloud infrastructure reaches 1.7 PB.

## Expectations for High-Quality Datasets

Within the oil & gas sector, major enterprises are developing large industry-specific models that require high-quality vertical datasets to support professional capabilities in specialized scenarios. Like other forms of professional data, time-series data must undergo large-scale collection, parsing, cleansing, labeling, and enhancement processes. Given the sheer data volume involved, robust and efficient tools are essential to produce high-quality datasets.

Several typical examples include:

### Tight Sandstone Lithology Prediction

**Data Type:** Seismic interpretation data

**Data Source:** Data files

**Data Volume:** 2,000 km<sup>2</sup>, approx. 100 GB

### Intelligent Well Logging Interpretation

**Data Type:** Basic well information, logging data, curve data, interpretation results, core porosity & permeability data

**Data Source:** Group unified logging database; Oilfield regional data lake

**Data Volume:** Conventional, cementing, and electrical imaging logging data totaling approx. 10 TB

### Oil & Gas Reservoir Research

**Data Type:** Reservoir numerical simulation forward-modeling data

**Data Source:** Reservoir numerical simulation systems

**Data Volume:** Approx. 10 TB

### Intelligent Gas Well Production Optimization

**Data Type:** Single-well static and dynamic data

**Data Source:** Regional data lake

**Data Volume:** Approx. 500 MB per well; 27,000 wells total; approx. 15 TB

### Intelligent Optimization of Drilling Parameters

**Data Type:** 1) Engineering logging time-series data 2) Drilling rig data, wellbore structure, drill string assembly, well trajectory, full drilling fluid performance, pressure loss data, drilling daily reports 3)

Engineering modeling calculation results

**Data Source:** EISC

**Data Volume:** 1) Logging data: approx. 12 TB 2) Drilling data: approx. 1 TB, 400+ million rows

### Intelligent Fracturing Decision Support

**Data Type:** Geological and engineering sweet spot modeling data, fracturing execution data, monitoring data, fracturing design documents, fracturing completion reports

**Data Source:** EISC

**Data Volume:** Approx. 3 million data records per well; approx. 2,000 wells in a selected block

### Barriers to Advanced Time-Series Analytics

In AI application scenarios within the oil & gas industry, even in the domains of production and safety alone, the scope spans oil and gas exploration, development and production, renewable energy production, wellbore engineering technologies, refining and petrochemicals, advanced materials, supply chain production coordination, QHSE (Quality, Health, Safety, and Environment), engineering construction, sales, storage and transportation, equipment manufacturing, and production logistics.

Across these areas, researchers have proposed and developed numerous time-series data analytics algorithms with diverse technical characteristics tailored to different scenarios. These algorithms have been widely applied to tasks such as time-series forecasting and anomaly detection. However, for business users, the practical application of such complex algorithms still presents significant barriers.

Taking the operational diagnostics of an atmospheric and vacuum distillation unit (AVU) in refining and chemical production as an example, time-series forecasting capabilities are required to predict key operating parameters and perform anomaly detection. Early warning responses must be triggered for process abnormalities such as vacuum furnace coking, fluctuations in the outlet flow of vacuum gas oil pumps, and level fluctuations in atmospheric distillation columns.

From the initial collection of time-series data through integrated systems to the final deployment and operationalization of predictive models, the path remains long and challenging.

## 1.2 From “Data” to “Assets” to “Productivity”

The digital and intelligent practices of the oil & gas industry have reinforced the industry-wide consensus that “data is an asset,” reflecting enterprises’ deeper expectations for the assetization of data. What companies truly seek is to establish a transformation pathway from “data” to “assets” to “productivity”: leveraging a unified data foundation to convert fragmented data into structured assets, injecting domain knowledge through industry-specific large models, and ultimately generating new digital productivity that supports process optimization, risk early warning, and intelligent decision-making.

## Chapter 2

# High Cost and Low Efficiency in Traditional Data Processing

Time-series data in the oil & gas industry is widely used across energy production, transportation, processing, and sales. A vast number of devices, sensors, and monitoring systems continuously generate real-time time-series data to support production monitoring, dispatch optimization, and fault early warning.

As business data volumes grow rapidly, the number of time-series records has expanded dramatically. Real-time data ingestion has become a bottleneck, while query and analytical performance has significantly slowed, creating new technical challenges.

Traditional relational databases, data historians, NoSQL databases, stream processing engines, and other data infrastructure components have not fully leveraged the unique characteristics of time-series data. As a result, performance improvements have been limited, often relying solely on cluster scaling—adding more compute and storage resources to cope with the load.

Moreover, the proliferation of system components has greatly increased overall system complexity, leading to sharply rising development, operational, and maintenance costs for enterprises.

The following are several real-world cases of enterprises that have used traditional data processing solutions, illustrating the practical difficulties they encountered during implementation and operation.

### 2.1 Relational Database Solution

In the production management system of a leading petrochemical enterprise, Oracle was primarily used to store and process time-series data. However, as usage expanded over time and data volumes continued to grow, challenges related to real-time data storage and processing became increasingly pronounced. Technical optimization efforts eventually reached their limits:

- After storing large volumes of data in Oracle, write and query performance degraded significantly—par-

ticularly for complex queries and large-scale aggregations, where performance decline was even more pronounced.

- The data compression ratio was low, resulting in substantial disk space consumption.
- Under multi-user or multi-application access, concurrency conflicts and locking issues frequently occurred.
- Data partitioning and archiving were cumbersome to manage, and in the event of system failures, data recovery cycles were excessively long.
- In addition, data synchronization efficiency was too low to achieve second-level data replication.

## 2.2 Traditional Data Historian Solution

A large oilfield adopted a “SCADA system + data historian” architecture to collect production time-series data. However, issues such as fully loaded industrial control systems and complex data link chains created significant obstacles during the transition from SCADA systems to an industrial Internet platform:

- Closed and incompatible data interfaces: Multiple operational zones and plants used isolated real-time and historical databases that were not interoperable.
- Low processing efficiency: As business expanded rapidly and the number of measurement points multiplied, the system could no longer meet the requirements for high-throughput ingestion and high-concurrency queries.
- Complex operations and maintenance: Database operations were cumbersome, and data was passively replicated and synchronized across multiple products, making troubleshooting extremely difficult.
- Limited data analytics capabilities: The system was unable to quickly support upper-layer big data analytics and innovative application requirements.
- Difficult edge–cloud synchronization: Data coordination between operational zones and plant-level centers was highly challenging, significantly increasing the complexity of subsequent data governance and limiting the ability to fully unlock data value.

## 2.3 Hadoop-Based Big Data Platform Solution

An industrial Internet project required end-to-end data aggregation from the application layer to a big data platform, including analytical computation and the generation of business reports for application-layer feedback. Typical tasks included calculating per-minute average power curves, aggregating per-minute electricity consumption, and comparing energy usage across different equipment and plants. The system needed to support both real-time stream processing and batch processing.

Using Hadoop as the big data platform introduced numerous challenges for technical teams:

- A “heavy” architecture: The solution required the integration of multiple technology components. For example, HDFS and Hive were used for raw data storage; HBase stored processed data; Kafka served as the messaging middleware for database synchronization; Flink or Spark handled computation; and ZooKeeper was used for distributed coordination.
- High technical complexity: Effective operation required deep familiarity with the Hadoop ecosystem, including data balancing and redundancy strategies, Hive partitioning strategies, and HBase key design principles.
- Difficulty in Controlling and Adapting Business Rules: Take the per-minute power calculation as an example. Initially, the value used was the last reading within each one-minute interval; later, the rule was changed to use the average value within that minute. Changes like these require the big data system to recompute from the ODS layer onward, resulting in a significant amount of additional development effort.

In a centralized monitoring and control project for renewable energy, three regions needed to ingest large volumes of monitoring data from wind turbines, PV inverters, energy storage equipment, and other systems, and perform application-level analytics on this data. The number of measurement points was close to 7 million, with data collection frequencies spanning seconds, minutes, and hours.

The project initially adopted CDH (Cloudera Distribution Including Hadoop) as the underlying data platform to manage massive time-series data—an integrated big data platform built on Hadoop. As time went on and more devices and data were connected, many issues emerged:

- Ingestion performance hit bottlenecks and data loss occurred
- System stability was difficult to guarantee, and recovery after disconnections was slow
- Overall operations and maintenance became extremely challenging.

## 2.4 NoSQL Database Solution

A manufacturer of energy storage equipment used MongoDB to store time-series data, including operational data, second-level data, power data, energy data, revenue data, as well as FFR and FACS data. The workload involved ingesting and storing tens of millions of measurement points across hundreds of thousands of devices, along with real-time querying and analytics of device status. The use cases imposed stringent performance requirements, with response latency expected to be at the millisecond level.

In practice, as the number of connected devices and measurement points increased, MongoDB’s limitations in handling time-series data became increasingly apparent:

- Low storage efficiency and high per-unit storage cost: Compared with purpose-built time-series databases, MongoDB typically achieves lower compression ratios. At massive time-series scale, this leads to higher storage overhead. In addition, MongoDB does not yet support tiered storage, making it difficult to allocate storage resources differently based on data value.

- Limited write performance and long ingestion times for large volumes: For high-frequency (millisecond-level) data ingestion, MongoDB's write performance operates at a seconds-level granularity and is difficult to meet the high-frequency data acquisition requirements of our energy storage devices. In practice, data is generated faster than it can be written.
- Lower query performance and long latency for real-time queries: MongoDB provides broad support for complex queries, but in real-time scenarios where data must be returned immediately, query performance still cannot meet production requirements.
- Complex technical architecture and a heavy real-time stack: Due to MongoDB's limited real-time processing capability, Redis must be introduced in the system architecture to satisfy real-time query requirements. This adds more components and more complex data flows, increasing both application development effort and technical management complexity.

## Chapter 3

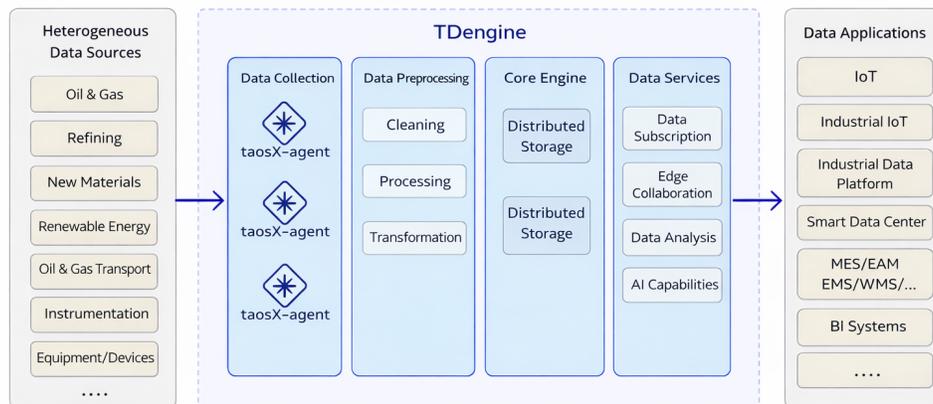
# Introducing TDengine

TDengine is an AI-powered data historian. It is built on a scalable, high-performance time-series database with an industrial data management layer that delivers everything a traditional data historian does – plus AI-powered operational insights.

TDengine integrates capabilities for time-series data ingestion, storage, analytics, and distribution, and offers a range of unique technological advantages.

### 3.1 Four-in-One Architecture: TDengine’s Innovative Solution

For time-series data processing in the oil & gas industry, the TDengine team has proposed a “four-in-one” integrated solution:



## Data Ingestion and Aggregation

TDengine aggregates industrial data from diverse sources across oil & gas production environments—including SCADA systems, DCS systems, PLC systems, equipment sensors, data historians, relational databases, and file storage systems—into a unified platform. It supports a wide range of data sources and protocols:

- Industrial protocols: OPC UA, OPC DA, MQTT
- Message queues: Kafka
- Data historians: PI System, Aveva Historian
- Time-series databases: TDengine, InfluxDB, OpenTSDB
- Relational databases: Oracle, MySQL, SQL Server, PostgreSQL
- NoSQL databases: MongoDB
- Text files: CSV

## Data Preprocessing

Before data is ingested into the database, TDengine performs cleansing, transformation, and processing to ensure data quality, enabling centralized monitoring and comprehensive analytics. Leveraging TDengine’s zero-code data ingestion capabilities, industrial data sources can be configured with minimal setup to complete ETL (Extract, Transform, Load) workflows efficiently and reliably.

## Core Engine

TDengine employs an innovative storage architecture to deliver read and write performance more than 10 times higher than conventional databases. It efficiently stores aggregated data using columnar storage, two-level compression, and optimized compression algorithms tailored to different data types, achieving compression ratios over 10 times greater than those of general-purpose databases.

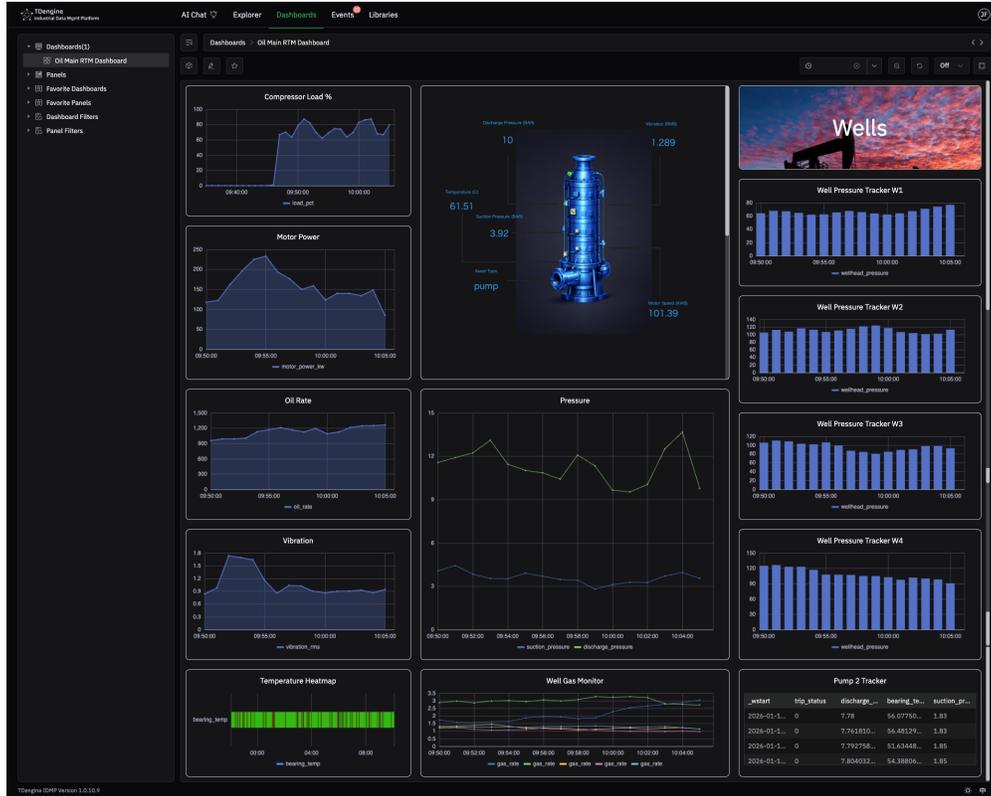
In addition, through time-based partitioning, device-level sharding, and the separation of storage and compute, TDengine provides robust horizontal scalability to support large-scale industrial deployments.

## Data Services

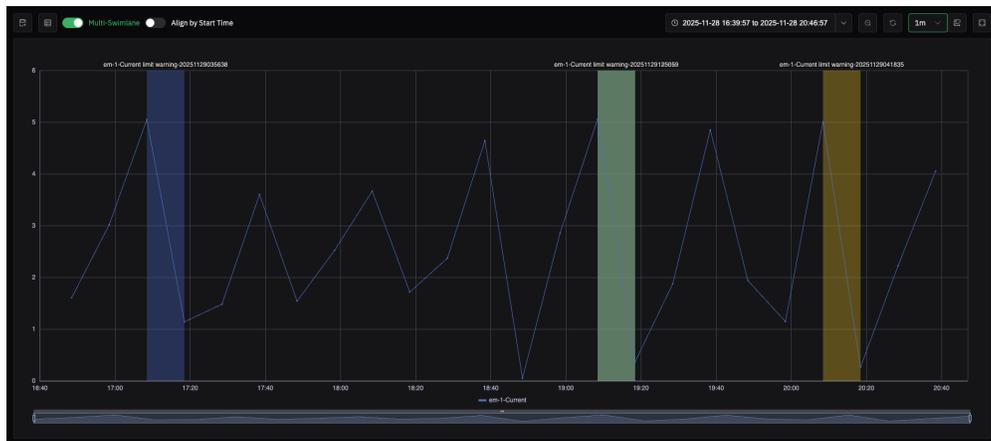
TDengine provides standard SQL query capabilities and built-in stream processing functions. It supports standard JDBC and ODBC interfaces, enabling seamless integration with a wide range of visualization, BI, and AI/ML tools, as well as advanced AI-enabled analytics.

TDengine offers data subscription capabilities, allowing processed data to be distributed in real time to third-party applications. With TDengine, enterprises can easily build cascaded “edge–cloud synchronization” architectures to support distributed industrial scenarios.

In addition, TDengine includes built-in visualization and dashboard capabilities for rapid operational insight without external tooling. Through its semantic modeling and contextualization features, users can organize time-series data into meaningful business entities and digital twin hierarchies, enabling more intuitive exploration and analysis.



Native event detection and event frame support make it easy to capture operational patterns, anomalies, and process windows directly within the data platform.



For business users and engineers, the TDengine Excel Add-In (EAI) enables direct access to real-time and historical data inside familiar spreadsheet workflows, supporting ad hoc analysis, reporting, and data exploration without writing code.

## 3.2 Foundational Architecture Aligned with Industry Needs

With its “four-in-one” integrated solution, TDengine effectively meets the oil & gas industry’s expectations for high-quality processing and intelligent utilization of massive time-series data. This capability stems from TDengine’s advantages as a full-stack time-series data processing platform.

### Simplified Platform

A typical time-series data processing architecture requires not only a time-series database, but also supporting components such as message queues, caching systems, and stream processing engines. To simplify system design and reduce operational costs, TDengine fully leverages the characteristics of time-series data and integrates built-in caching, stream processing, and data subscription capabilities.

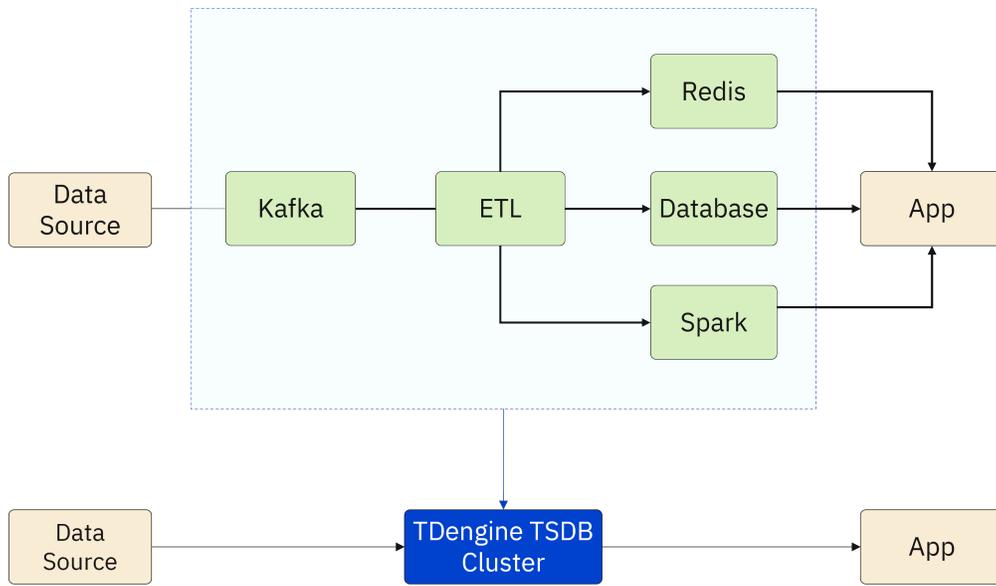
This provides a comprehensive, full-stack technical solution for time-series data processing, positioning TDengine as a streamlined and highly integrated platform. For more details, see: [A Minimalist Solution](#).

### Comparison with Traditional Architectures

In conventional time-series data platform designs, data is first collected and written into a message queue, then consumed and passed through ETL processes for cleansing and transformation. The processed data is subsequently written into databases such as HBase, MongoDB, or MySQL for persistent storage. At the same time, the platform caches the latest data in Redis and forwards it to stream processing frameworks such as Spark or Flink for real-time analytics. The analytical results are also stored in databases. Applications must integrate separately with Redis, the database, and stream processing engines.

By contrast, TDengine provides a streamlined solution for time-series data processing through its built-in caching, stream processing, and data subscription capabilities. This eliminates the need for Kafka, Redis, Spark, Flink, ETL frameworks, or similar tools, significantly simplifying system architecture and reducing overall complexity and operational costs.

A comparison of the logical structures of the two approaches is shown below:



## Chapter 4

# From Data to Services: TDengine Addresses Key Industry Challenges

Through an innovative architectural approach, TDengine resolves three major challenges in time-series data processing within the oil & gas industry:

### 4.1 Ultra-High Performance: Overcoming Massive Data Read and Write Challenges

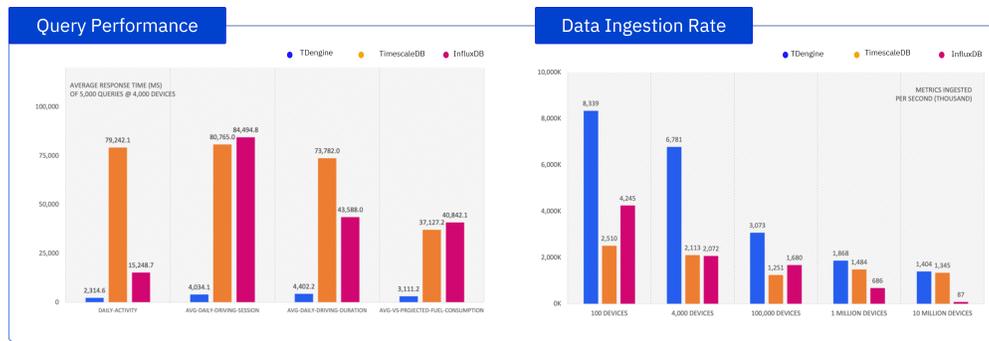
#### High Performance

TDengine can handle tens of thousands of write requests per second, ingest millions of data points, and retrieve over ten million data points per second. Its read and write performance is more than ten times faster than that of conventional general-purpose databases.

This exceptional performance is enabled by three key innovations:

- Data Model: One table per data collection point
- Super Table: Efficient aggregation across multiple data collection points
- Virtual Table: Logical association and querying across multiple data collection points

TDengine fully leverages the characteristics of large-scale time-series data and incorporates an innovative storage engine design that significantly improves both data ingestion and query performance, while also achieving substantially higher compression ratios. Compared with general-purpose databases, TDengine delivers at least a tenfold improvement in read performance, write performance, and data compression efficiency. TSBS benchmark results further demonstrate that TDengine significantly outperforms TimescaleDB and InfluxDB.



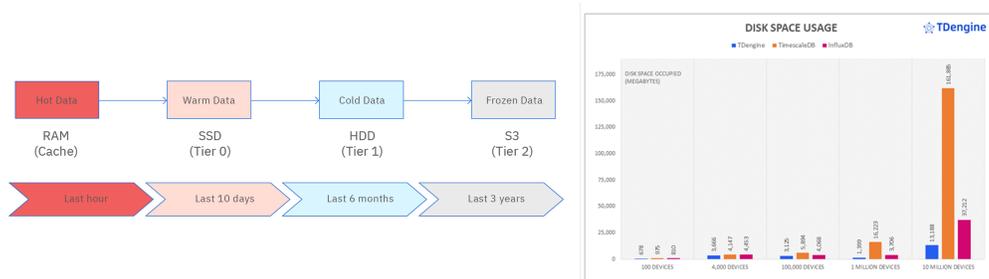
The oil & gas industry generates enormous volumes of data through production IoT systems, often reaching terabyte scale within a single day. As a result, efficient storage becomes critically important. TDengine employs a range of technical strategies to ensure high performance while significantly reducing storage costs to the greatest extent possible.

### High Compression Ratio

TDengine offers multiple compression algorithms with industry-leading compression performance, reducing datasets to as little as one-tenth of their original size.

### Tiered Storage

TDengine supports data tiering and S3-based storage, enabling data from different time periods to be stored in directories mounted on different storage media. This allows data of varying “temperature” levels (hot, warm, cold) to reside on appropriate storage tiers, maximizing compression efficiency and storage utilization while reducing overall costs.



## 4.2 Strict Ingestion and Controlled Output: Zero-Code Data Quality Enhancement

The successful implementation of AI in the oil & gas industry depends on high-quality datasets. However, in real-world IoT and industrial Internet environments, the diversity of equipment is vast, and enterprises often operate multiple sites and stations across different locations. Due to varying construction timelines, software systems deployed at different sites may differ significantly.

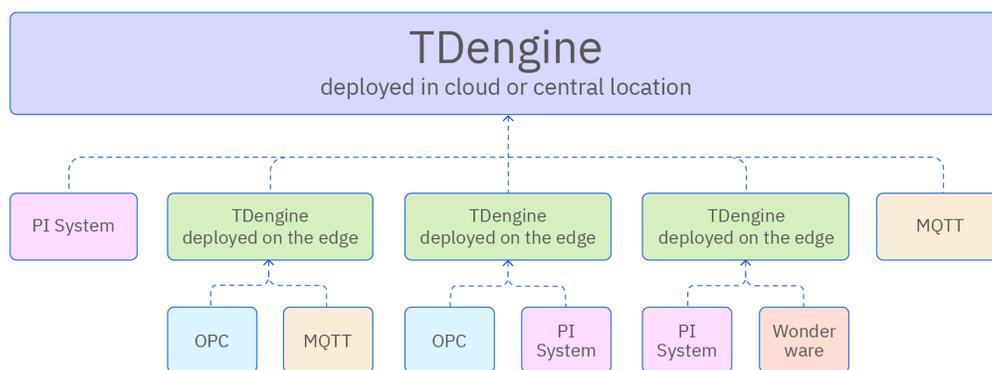
As a result, aggregating and standardizing data from multiple locations becomes critically important. TDengine addresses this seemingly straightforward yet operationally complex challenge through a comprehensive and robust solution.

## Support for Multiple Data Sources

TDengine is equipped with a powerful visual data management tool, taosExplorer. With taosExplorer, users can complete simple configurations directly through a browser and submit tasks to TDengine, enabling zero-code ingestion of data from diverse data sources into the platform.

During the ingestion process, TDengine automatically performs data extraction, filtering, and transformation to ensure data quality before storage. No additional ETL tools need to be deployed, significantly simplifying overall system architecture and improving data processing efficiency.

The following diagram illustrates the system architecture of the zero-code data ingestion platform:



Currently, TDengine supports a wide range of data sources, including MQTT, OPC-UA, OPC-DA, Kafka, and CSV; traditional data historians such as PI System and Wonderware; relational databases including MySQL, PostgreSQL, and Oracle; and time-series databases such as InfluxDB and OpenTSDB.

In addition, TDengine supports various data collectors such as Telegraf, Collectd, and Prometheus, with plans to expand support for additional data sources in the future.

## ETL Capabilities

Because data may originate from multiple heterogeneous systems, differences often exist in physical units, naming conventions, and time zones. To address this, TDengine provides built-in ETL functionality that can parse, extract, filter, and transform data from various source payloads before ingestion. This ensures high data quality and establishes a unified namespace for downstream use.

The key functions include:

- **Parsing:** Use JSON Path or regular expressions to extract fields from raw messages.
- **Field Extraction and Splitting:** Use split functions or regular expressions to extract multiple fields from a single raw field.

- **Filtering:** Messages are written to TDengine only when the defined expression evaluates to true.
- **Transformation:** Define mapping and transformation rules between parsed fields and TDengine super table fields.

## Edge–Cloud Synchronization

TDengine allows one instance to subscribe to data from another TDengine instance, enabling the construction of a cascaded architecture. At the edge, collected data can be stored locally in a TDengine instance. A cloud-based TDengine instance can then aggregate data in real time from one or multiple edge instances through the subscription mechanism.

In addition, TDengine addresses technical challenges such as breakpoint resumption, firewall traversal, and data backfilling. With simple configuration, the system can achieve seamless edge–cloud synchronization, significantly reducing architectural complexity and operational costs.

## 4.3 AI Agents: Bridging the Gap from Data to Decision-Making

Advanced analytics and AI application scenarios in the oil & gas industry are extensive. Researchers have proposed and developed a wide variety of algorithmic models with distinct technical characteristics tailored to different use cases. How to rapidly validate these algorithms and models on assetized data foundations has become a key constraint in bringing AI into practical deployment.

TDengine supports AI-enabled advanced data analytics, helping accelerate the validation and operationalization of AI models and bridging the gap between data and actionable decision-making.

### Advanced Data Analytics

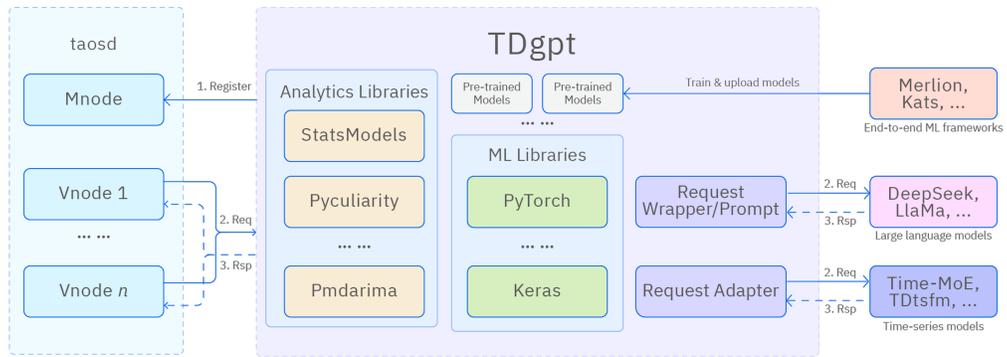
TDengine provides an open solution for industrial data, supporting both public cloud and on-premises deployments. It delivers open, cost-effective, and horizontally scalable real-time analytics capabilities. Through SQL and time-series extensions, TDengine offers built-in analytical functions and supports real-time stream processing.

It can seamlessly integrate with a wide range of third-party tools for batch analytics, real-time analytics, reporting, AI/ML, and visualization—such as Grafana, FineReport, Yonghong BI, and Power BI. For more details, see: [Advanced Data Analytics](#).

### AI Agent for Time-Series Data Analytics

TDgpt is TDengine’s built-in AI agent for time-series data analytics, providing capabilities such as time-series forecasting, anomaly detection, data imputation, and classification. It supports dynamic switching among multiple model algorithms, which can be invoked through a single SQL statement.

An open SDK enables developers to integrate custom AI algorithms or models, making them accessible to TDengine users across industries.



## Chapter 5

# Practical Cases: Empowering Digital Transformation in Oil & Gas

TDengine has extensive practical implementations in the oil & gas sector. This section highlights several representative application scenarios.

### 5.1 Sinopec: Transitioning from SCADA Systems to an Industrial Internet Platform

#### Project Background

A large oil and gas field enterprise has established the largest oil and gas production IoT system in China. The system includes over 370,000 instruments and meters and more than 120,000 control devices. Domestic SCADA systems have been deployed across 119 operational zones, bringing more than 90,000 oil, gas, and water wells online, with a coverage rate of 96.7%, while achieving 100% digitalization of stations. This has enabled regionally centralized real-time monitoring and management of critical production data.



However, in the area of time-series data acquisition, the architecture based on “SCADA systems + data historians” presented challenges such as fully loaded industrial control systems and complex data link chains. These issues became significant obstacles during the transition from SCADA systems to an industrial Internet platform:

- Closed and incompatible data interfaces: Multiple operational zones and plants used isolated real-time and historical databases that were not interoperable.
- Low processing efficiency: As business expanded rapidly and the number of measurement points multiplied, the system could no longer meet the requirements for high-throughput data ingestion and high-concurrency queries.

- **Complex operations and maintenance:** Database operations were complicated, and data was passively replicated and synchronized across multiple products, making troubleshooting extremely difficult.
- **Limited data analytics capabilities:** The system was unable to respond quickly to upper-layer big data analytics and innovative application needs.
- **Difficult edge–cloud synchronization:** Data coordination between operational zones and plant-level centers was highly challenging, significantly increasing the complexity of subsequent data governance and preventing the full realization of data value.

## **TDengine Solution**

Based on the group’s unified industrial cloud architecture and centered on direct cloud-based acquisition of oil and gas production data, the enterprise advanced its industrial Internet construction. A customized industrial time-series data platform was developed, and representative application scenarios were implemented to address issues such as fully loaded industrial control systems and complex data link chains, enabling a gradual transition from SCADA systems to an industrial Internet platform.

The approach to collecting oil and gas production time-series data underwent a significant transformation.

## **System Deployment Overview**

### **Oil and Gas Production Data Acquisition**

- 45,407 oil wells connected
- 16,012 water wells connected
- 15,344 gas wells connected

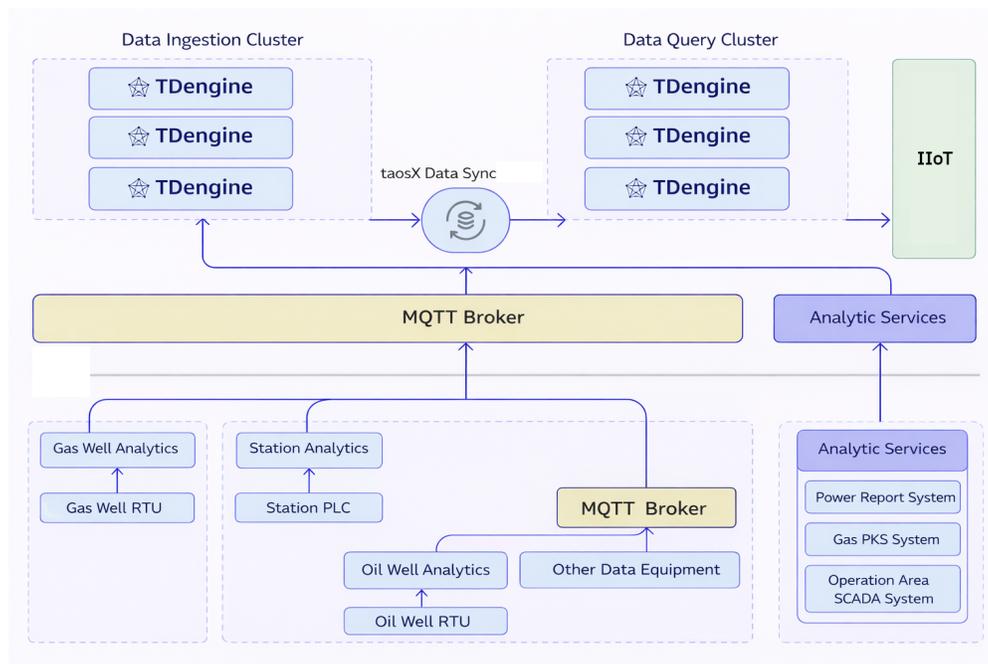
### **renewable energy Data Acquisition**

- 34 photovoltaic well sites integrated
- Pilot application of energy management and control data at 1 station

### **Development of Typical Application Scenarios**

- Established an associated gas comprehensive utilization application scenario
- Deployed cloud-based configuration tools to enable the transition of monitoring models from operational zone level to plant-level management

The deployment architecture of the TDengine time-series data platform is illustrated in the following diagram:



## Application Outcomes

- **Linear Business Scalability:** The system supports horizontal scalability. As business data volume increases, processing capacity can be expanded through horizontal scaling to accommodate growing workloads.
- **Separation of Storage and Compute:** By separating write and query workloads, and decoupling storage from computation, the architecture reduces the risk of single points of failure and enhances overall system stability and reliability.
- **Real-Time Data Push:** Time-series data can be shared and pushed in real time without relying on external message queues or middleware. TDengine supports subscription not only to raw data, but also to aggregated, summarized, or transformed data. By simply defining the data topic to be shared, TDengine can push subscribed data to authorized consumers.
- **Parallel Computing:** Through a DAG (Directed Acyclic Graph) execution engine, user-defined transformation operations are first compiled into a logical execution plan and then converted into a physical execution plan. The DAG scheduler decomposes computation tasks into multiple stages, executing tasks in parallel within each stage to significantly improve execution efficiency.

## 5.2 CNPC: From Traditional Historian to TDengine

### Project Background

A large oilfield initially used conventional data historians to collect field data. However, these traditional software solutions proved insufficient when it came to data analysis capabilities. In an effort to improve production management, the user migrated data to a relational database, represented by Oracle, with the goal of leveraging these databases as the central platform for data aggregation and analysis. However, as the volume of oilfield data skyrocketed, the client faced two major challenges:



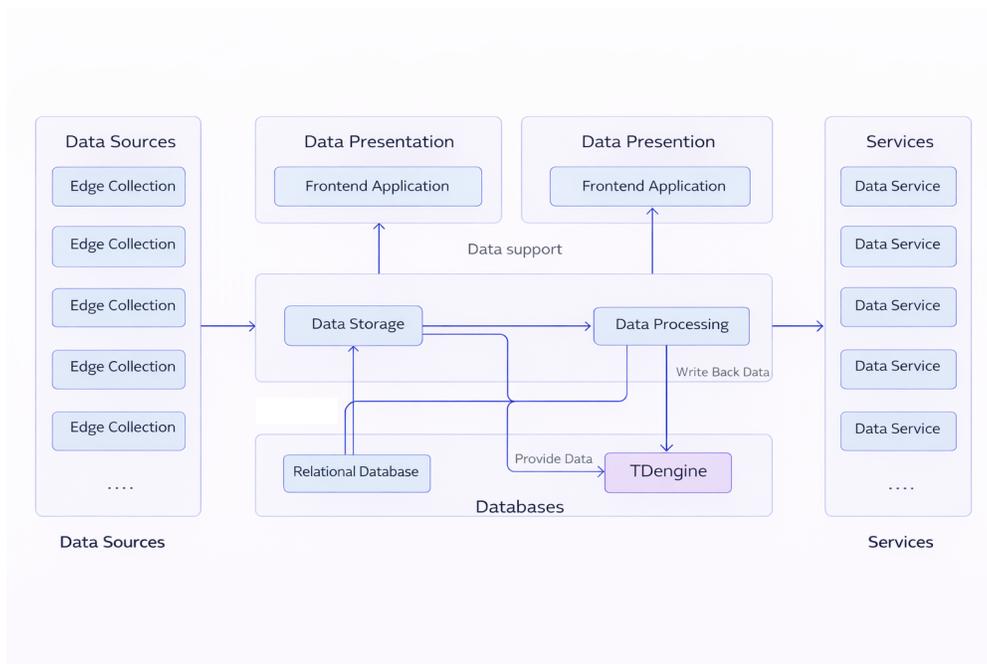
- Rapid growth in data acquisition volume
- Significant increase in data acquisition frequency

In this context, traditional relational databases began to show a series of issues and bottlenecks in data processing:

- High server resource requirements: More than 40 servers were required to store data across the entire oilfield; the existing data volume reached nearly 200 TB, with an annual increase of 22 TB.
- Low data processing efficiency: Data insertion performance was limited; large-scale historical data queries frequently timed out.
- Difficult database operations and maintenance: Backing up historical data was challenging; deleting expired data was cumbersome and inefficient.

### TDengine Solution

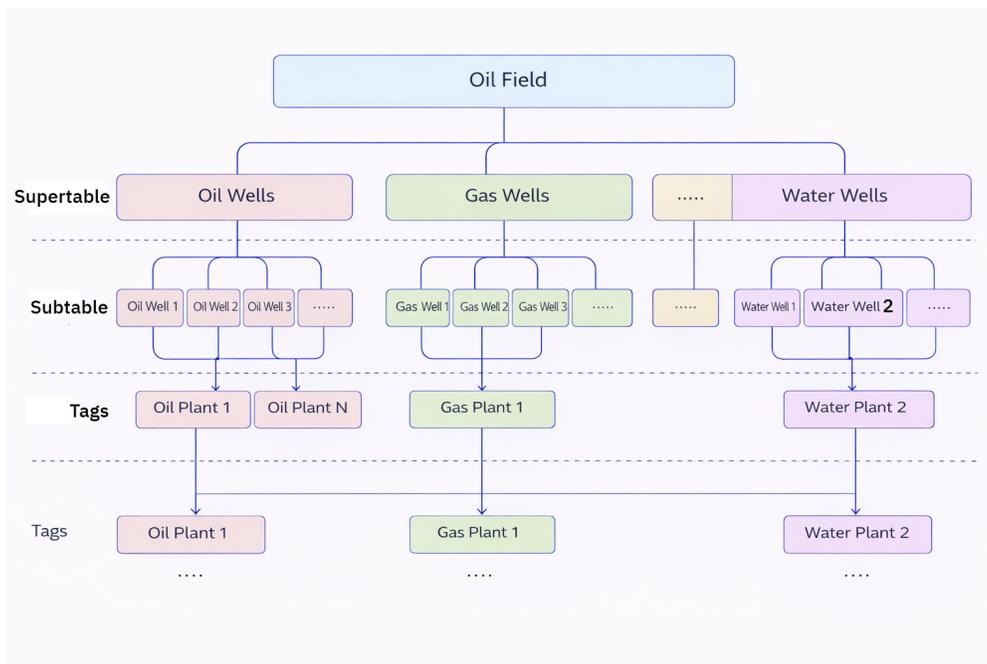
By adopting TDengine as the foundational data platform, it plays a critical role in the development of the smart oilfield. It provides a data-driven foundation for automated data acquisition and control, while delivering strong data support for the continuous optimization of production and operations.



TDengine’s innovative design concepts of “one table per data collection point” and “super tables” significantly improve the efficiency of time-series data ingestion, querying, and storage.

During the project transformation, the client categorized 28 types of monitored objects, including oil, gas, and water wells, flow meters, valve groups, pumps, tanks, and compressors, and created tables accordingly. Taking oil wells as an example, the client first defined the required data items and their respective collection frequencies. Then, a dedicated table was created for each individual oil well. Each table was assigned corresponding static tags, such as the name of the oil production plant and the responsible business unit.

This design not only ensures refined data management and efficient retrieval, but also greatly simplifies data organization and maintenance.



To meet the diverse data requirements of AI research, data mining, predictive equipment maintenance, and other advanced applications, the client frequently needs to aggregate real-time oilfield data from plant-level systems to the company level, and then further consolidate company-level data at the group or business unit level.

As illustrated below, this process places extremely high demands on data timeliness and synchronization. TDengine’s outstanding performance ensures the smooth and reliable operation of this critical data integration workflow.



### Application Outcomes

Following the full migration from Oracle to TDengine, the project achieved significant optimization results, reflected in the following aspects:

- Over 40 Oracle clusters were seamlessly replaced with 9 TDengine clusters.
- Data ingestion performance improved substantially, while hardware resource consumption was reduced, resulting in higher overall resource utilization.
- The cluster supports online horizontal scaling, enabling smooth future expansion and ensuring long-term scalability and forward compatibility.

- Flexible lifecycle management allows for simplified handling of expired data, improving the efficiency and convenience of data governance.
- Achieved synchronization throughput of 5 million measurement points per second, meeting the stringent real-time requirements of edge–cloud synchronization scenarios and ensuring efficient data flow and utilization.

## 5.3 PCS System of a Leading Petrochemical Enterprise

### Project Background

In the production management system of a leading petrochemical enterprise, Oracle was primarily used to store and process time-series data. However, as the system operated over time and data volumes continued to grow, challenges related to real-time data storage and processing became increasingly prominent, and technical optimization reached its limits:

- After storing large volumes of data in Oracle, write and query performance degraded significantly, especially for complex queries and large-scale aggregations.
- The compression ratio was low, resulting in substantial disk space consumption.
- Under multi-user or multi-application access, concurrency conflicts and locking issues frequently occurred.
- Data partitioning and archiving were cumbersome, and recovery cycles after system failures were excessively long.
- Data synchronization efficiency was too low to achieve second-level replication.

### TDengine Solution

In the oilfield PCS architecture, real-time data is aggregated from management-zone SCADA systems into branch-level time-series databases (with three-year retention). Through distributed cluster synchronization, data is then consolidated into a central-node time-series database for permanent storage.

Business data is stored in relational databases and centrally managed by the headquarters node, which distributes data downward based on branch-level application requirements. Within this architecture, relational and time-series data are ingested into data lakes at both the branch and headquarters levels, enabling synchronization between the enterprise data lake and the headquarters data lake.

Considering the positioning of the data lake and the PCS system's requirements for high-frequency real-time access and data integrity, the adopted technical approach first aggregates real-time data at the central node, where it is centrally managed and coordinated across branch nodes. The data is then synchronized to the data lake, which provides unified external data services (excluding PCS) to downstream systems.

Deployment Architecture Upgrade: In line with edge–cloud synchronization architecture requirements, container-managed clusters were established for the central node and enterprise nodes, including the container management cluster, application cluster, database cluster, and data processing cluster. Using distributed multi-cloud container cluster management capabilities, the solution enables unified application release, centralized database management, and standardized system operations and maintenance.

## **Application Outcomes**

After migrating oil and gas field data to TDengine for time-series storage, the solution effectively met the requirements of the operational scenarios. Enterprise-level distributed nodes achieved high read and write performance with reduced storage costs. The central node enables seamless data consolidation with enterprise nodes and supports flexible, scalable expansion. In addition, the platform provides unified external data services to downstream systems.

## Chapter 6

# Conclusion

As oil and gas operations continue to generate increasing volumes of high-frequency, distributed, and mission-critical data, traditional data architectures struggle to keep pace with modern performance, scalability, and cost requirements. At the same time, the growing importance of real-time monitoring, predictive maintenance, and AI-driven optimization demands a more flexible and future-ready data foundation.

TDengine provides a modern time-series and industrial data platform designed to meet these challenges. With scalable architecture, efficient storage, real-time analytics capabilities, and support for hybrid deployments, TDengine enables oil and gas organizations to unify operational data across assets and sites while maintaining reliability and control. By moving beyond legacy historian-centric models, companies can reduce complexity, improve visibility, and create a stronger foundation for advanced analytics and intelligent operations.

As the industry continues its digital transformation journey, adopting open, scalable, and AI-ready data platforms will be essential to sustaining operational excellence and long-term competitiveness.