BRNO FACULTY
UNIVERSITY OF INFORMATION
OF TECHNOLOGY TECHNOLOGY

Excel@FIT 2025
http://excel.fit.vutbr.cz

# IIoT Data Processing Module

Tobiáš Frajka

**Abstract**
We address the gap between heterogeneous industrial protocols and BI platforms by designing a modular IIoT pipeline in line with Industry 4.0 principles [1]. Our .NET microservices run in Docker, communicate via gRPC, and use interchangeable adapters for MQTT, OPC UA, Modbus, and AMQP to normalize time-series data into TimescaleDB [2, 3]. A Next.js dashboard visualizes real-time metrics and logs and allows for changes in system-wide configuration. The system design aims for low latency, high throughput, and reliability in industrial environments, potentially reducing integration complexity for industrial applications. This blueprint bridges edge data capture with analytics and digital twins [4].

*Faculty of Information Technology, Brno University of Technology*
xfrajk00@stud.fit.vutbr.cz

## 1. Introduction

Industrial facilities generate vast amounts of sensor and machine data, yet translating raw signals into actionable insights remains challenging. Existing platforms either target cloud-only workflows or vendor-locked architectures, leaving a "gap" for a lightweight, modular on-premise solution. We propose a data processing module that collects, normalizes, and routes time-series data from PLCs, HMIs, and databases into analytics engines or external archives. Our design emphasizes low latency, protocol flexibility, and seamless integration with business-intelligence tools.

As manufacturing systems become increasingly connected in the Industry 4.0 era [5], there's a critical need for middleware that can handle the heterogeneity of industrial protocols while offering standardized data outputs. The IIoT ecosystem requires not just connectivity but also proper data management to enable broader integration with enterprise systems [6]. Our module functions as this essential bridge, operating at the edge computing layer to provide low latency, reliable data processing before transmission to cloud or on-premise analytics platforms [7].

## 2. System Architecture

Figure 6 shows the high-level concept: input, core, and output microservices communicate via gRPC. Adapters encapsulate protocol logic for protocols like MQTT, OPC UA (already implemented) and Modbus or direct DB connection (planned). Data flows into a TimescaleDB instance within the core, then is served to external systems via a REST-based output module. A Next.js Web App provides a dashboard for adapter status, sensor charts, and system logs (Figure 3–5) and is used as a configuration hub.

The architecture follows a clear separation of concerns, with each component handling distinct responsibilities Figure 7:

- **Input Module**: Manages the protocol adapters and initial data ingestion, with specialized handlers for each protocol's specifics.
- **Core Module**: Contains shared data models, interfaces, and database contexts, ensuring consistency across the ecosystem.
- **Output Module**: Exposes REST APIs and handles data synchronization with external systems.
- **TimescaleDB**: Specialized time-series database that outperforms traditional RDBMSs for IoT data workloads

We chose microservices and containerization to ensure independent scalability and resilience, critical requirements for industrial applications where downtime is costly [8].

## 3. Implementation

- **Microservices in .NET 8**: each module runs in a Docker container for portability and versioning.
- **gRPC communication**: ensures high throughput and low inter-service latency.
- **TimescaleDB**: optimized storage and querying of time-series sensor data.
- **Adapters**: structured via a common interface, each handling basic operations like connection, initialization and data handling and mapping.
- **Web App**: built on Next.js 15 + TypeScript + Tailwind/shadcn/ui; uses React Query and Recharts for real-time dashboards.

Our implementation distinguishes itself from other IIoT solutions through its flexible adapter architecture. Each protocol adapter implements a common interface while encapsulating protocol-specific behaviors, allowing new protocols to be integrated with minimal changes to the core system [9]. This approach facilitates the evolution of the system as industrial communication standards evolve.

Data persistence leverages TimescaleDB's hypertable functionality for automatic data partitioning and efficient time-based queries. This allows the system to handle high-frequency sensor data without degradation in query performance [10]. For data reliability, we implement batched writes with transaction support and automatic retry mechanisms.

## 4. Results

The BP-IIoT module was designed for deployment on edge servers in industrial environments.

- **Modularity**: The system demonstrates effective separation between input adapters, core processing, and output interfaces, enabling independent development and maintenance.
- **Protocol Support**: The implemented MQTT and OPC UA adapters successfully process standard industrial protocol messages into a unified data format.
- **Configuration Management**: The system provides a unified configuration system that maintains consistency across distributed components and supports runtime reconfiguration.
- **Integration**: The standardized interfaces and protocol adapters reduce the need for custom integration code when connecting to industrial systems.

During development, collaboration was established with AGEsoft s.r.o., which provided feedback and testing support. The company evaluated the system using their MQTT and OPC UA simulators in controlled environments. AGEsoft provided positive feedback regarding the system's architecture and potential utility, particularly appreciating the modular design and unified configuration interface. They plan on using the system in their upcoming projects abroad Figures 1 and 2.

The system architecture supports scalability through its microservices design and containerized deployment model. Database performance is optimized for both real-time dashboard visualization and historical data queries using TimescaleDB's specialized time-series capabilities.

Integration testing confirmed that the standardized interfaces facilitate connection to existing industrial systems. The implementation of protocol-specific adapters within a common framework demonstrates the feasibility of the architectural approach for industrial IoT applications.

While comprehensive performance benchmarking in production environments remains part of future work, initial testing indicates the system provides a viable foundation for industrial data collection and processing applications.

## 5. Conclusion

We delivered a complete, open-architecture IIoT data processing module that meets industrial requirements for flexibility, performance, and ease of deployment. Our containerized microservices and interchangeable adapters allow rapid proof-of-concept and production rollouts. Future work includes adding AI/filtering modules and richer security policies (TLS, RBAC) to meet evolving standards.

## 6. Acknowledgments

This work was carried out in cooperation with AGEsoft s.r.o. and supervised by Ing. Jiří Hynek, Ph.D.

## References

[1] Heiner Lasi, Peter Fettke, Hans-Georg Kemper, Thomas Feld, and Michael Hoffmann. Industry 4.0. *Business & information systems engineering*, 6:239–242, 2014.

[2] H. Kagermann, J. Helbig, W. Wahlster, and A. Hellinger. *Recommendations for Implementing the Strategic Initiative INDUSTRIE 4.0: Securing the Future of German Manufacturing Industry ; Final Report of the Industrie 4.0 Working Group*. Forschungsunion, 2013.

[3] Jay Lee, Behrad Bagheri, and Hung-An Kao. A cyber-physical systems architecture for industry 4.0-based manufacturing systems. *Manufacturing Letters*, 3:18–23, 2015.

[4] Z. Lv and E. Fersman. *Digital Twins: Basics and Applications*. Springer International Publishing, 2022.

[5] F. Shrouf, J. Ordieres, and G. Miragliotta. Smart factories in industry 4.0: A review of the concept and of energy management approached in production based on the internet of things paradigm. In *2014 IEEE International Conference on Industrial Engineering and Engineering Management*, pages 697–701, 2014.

[6] Montdher Alabadi, Adib Habbal, and Xian Wei. Industrial internet of things: Requirements, architecture, challenges, and future research directions. *IEEE Access*, 10:66374–66400, 2022.

[7] Weisong Shi, Jie Cao, Quan Zhang, Youhuizi Li, and Lanyu Xu. Edge computing: Vision and challenges. *IEEE Internet of Things Journal*, 3(5):637–646, 2016.

[8] Joyce J. Industrial internet of things (iiot) – an iot integrated services for industry 4.0 : A review. *International Journal of Applied Science Engineering*, 8, 06 2020.

[9] Samer Jaloudi. Communication protocols of an industrial internet of things environment: A comparative study. *Future Internet*, 11(3), 2019.

[10] Sanjana Rathi and N. Jeba. Effective data management and real-time analytics in internet of things. *International Journal of Cloud Computing*, 10:112, 01 2021.