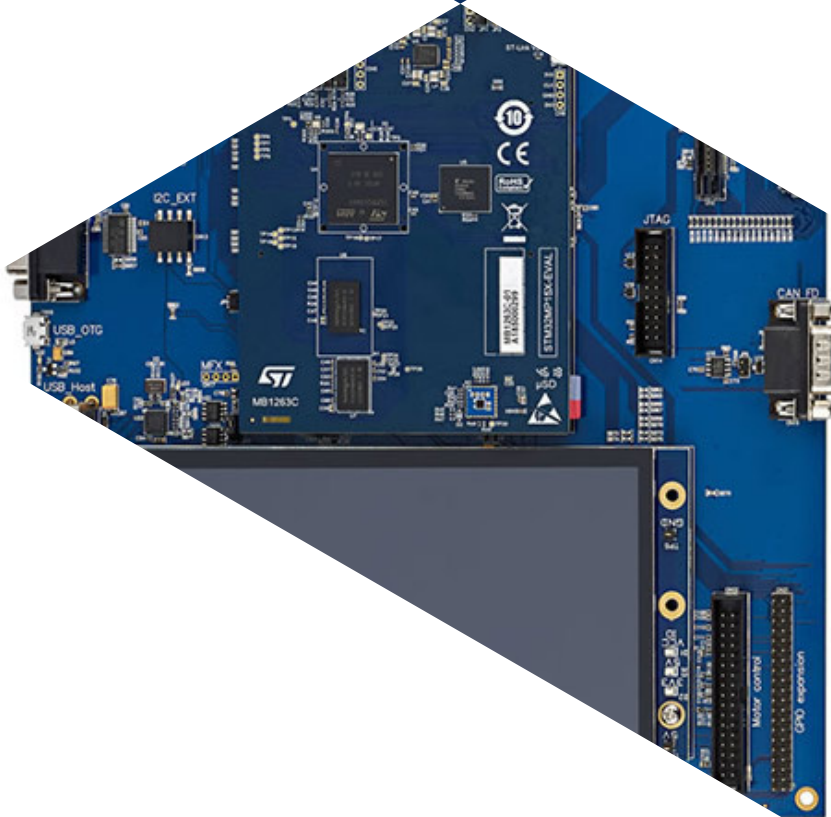


# EXTEND CORTEX-M DEVICE DATA PROCESSING & MANAGEMENT

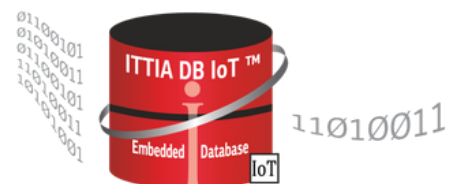
WHITE PAPER

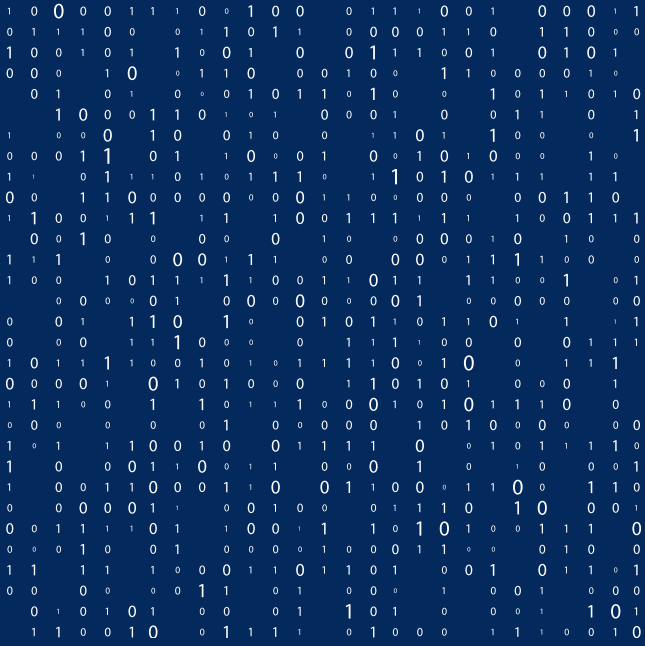
## BRING AN IOT & AI DATA REFINERY TO THE SMALLEST ENDPOINT DEVICES



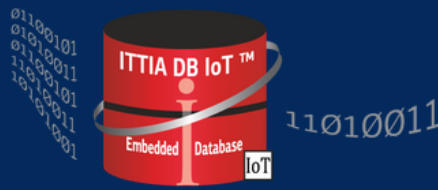
JANUARY  
2024

**ITTIA**

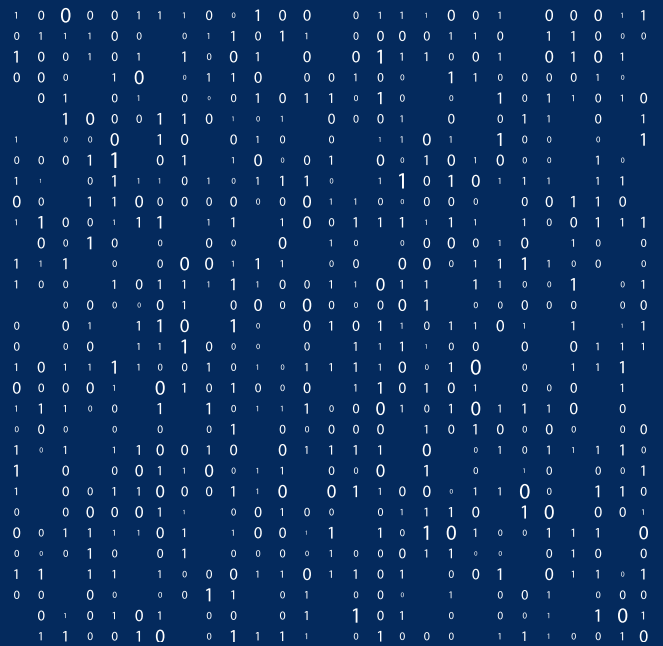




# “An Overview of Modern Options for Data Management and Data Processing for MCU Devices.”



“Capture All Data Points Metrics, Events, Logs and More. Gain Real-Time Insight.”



## Introduction

Edge network device applications must ingest, process, and make sense of data points at a high sample rate. Microcontrollers, MCUs, are a key component for many embedded systems and are becoming more and more responsible for capturing, processing, and managing the data flowing through them. The process starts locally at the devices where manufacturers are looking into innovative opportunities to understand and monetize every bit of data that they can collect.

These are a few challenges presented by real-time data collection and management for MCUs. To overcome these obstacles, we designed ITTIA DB IoT to perform interactive queries on data streams that filter, aggregate, split, and combine data points before that data is stored or transmitted. ITTIA DB IoT is an embeddable database designed to be small enough to run on a microcontroller and offers developers, through this single platform, the option to select the data management capabilities that meet their needs: stream processing, time series data management or relational table data management.

Applications can initiate new data collection campaigns at any time and either receive continuous updates or end the campaign before querying the final results. In this way, data transmission is minimized but important information required to make decisions is prioritized. The database capabilities required to scale up the deployment and management of edge devices and their data workloads are included in ITTIA DB IoT. It combines lightweight data management and processing and is built for microcontrollers to satisfy the demands of data workloads operating in resource-constrained, field-deployed devices.

Working with the world's top microchip companies, committed ITTIA DB developers have collaborated to deliver unique, superior commercial grade database for manufacturers of embedded systems. In this white paper, we will share the existing challenges and problems solved by ITTIA DB IoT for developer of MCU applications.

## Benefits of ITTIA DB for Microcontrollers

Most MCUs have severely limited computational resources, including Random Access Memory, RAM, Read-Only Memory, ROM, and flash media. RAM is volatile memory that temporarily stores the data and ROM is non-volatile memory that permanently stores instructions. Flash media is a rewriteable location where data is permanently stored. RAM, ROM, and flash are essential for any MCU software, including real-time database libraries. Therefore, the amount of data and limited resources play a crucial role in a MCU context, and application developers struggle to benefit from on-device data computing. As IoT and AI applications utilize this data, databases for MCUs are becoming more important to leverage the limited specifications of data-driven embedded systems.



Embeddable Edge Data Management

When we designed ITTIA DB IoT, we paid close attention to ensure that applications can smoothly run in such a constrained environment with small RAM and ROM footprint.

### **Real-time data collection campaigns**

Real-time data must be processed and delivered immediately after collection. There can often be no delay in the timeliness of data that is used for navigation, tracking, measurement, etc. For MCUs, such data must also be processed using real-time data management. The surge of real-time data coming from sensors and other data sources can lead to data congestion and delay in the transmission of insights required for decision-making.

These are a few challenges presented by real-time data collection and management. To overcome these obstacles, we designed ITTIA DB IoT to perform interactive queries on data streams that filter, aggregate, split, and combine data points before that data is stored or transmitted. Applications can initiate new data collection campaigns at any time and either receive continuous updates or end the campaign before querying the final results. In this way, data transmission is minimized but important information required to make decisions is prioritized.

### **Real-time data streaming and monitoring**

For MCUs, real-time analytics is a discipline where a significant volume of data points must be continuously ingested, indexed, analyzed, and either transmitted or stored on the MCU. Data points must be monitored and searched, filtered, aggregated, and processed according to specified queries.

With ITTIA DB IoT for MCUs, developers can create firmware applications to dynamically aggregate, filter, split, and combine real-time data into a sequence of processed events. This enables sensor devices to run continuous analysis, make sense of every bit of raw data, identify alarming occurrences, and emit/store valuable findings.

### **Data categorization and organization**

The main purpose for embedding ITTIA DB IoT with MCU application is to organize the data and benefit from data classification. The practice of grouping data according to its qualities and traits and then allocating class labels that characterize a collection of properties that apply to the matching data sets is known as data classification. This is a well-known practice for enterprise systems in the absence of restrictions. The objective with MCUs is to give unstructured, raw data meaningful class properties and enable them to organize and extract insights from a collection of data points.

ITTIA DB IoT queries facilitate classification by enabling analysis at the origin of each data point, inside MCU applications.



## **Storing clean data**

With the era of IoT and AI upon us, MCUs—like many other computing systems—must ingest a great volume of data and only store what is important, after cleaning the raw data. What does MCU data cleaning entail? The act of making sure data is accurate, consistent, and useful is known as data cleaning.

MCU application can clean data with ITTIA DB IoT either by discarding some data before it is stored or through data retention policies that remove old data. The advantages of clean data are numerous. When several data sources are combined into a single dataset, MCU-level data cleaning eliminates a significant amount of undesired raw data storage and reduces power costs associated with data transmission.

## **Time series data management**

New applications developed for embedded systems often store various data points sampled at different rates. As an embedded time-series database, ITTIA DB IoT is optimized to store and access associated pairs of timestamps and values. As a result, MCU applications can efficiently measure values from sensory apparatuses, effectively query any combination of captured data points and analyze how they change over time. The main value of ITTIA DB IoT time series is enabling MCU application to monitor, store and analyze real time data at every point.

## **Data integrity and scalability**

As data produced by IoT devices gains more and more influence over decision-making processes, it becomes increasingly important for MCU devices to make prompt, confident, accurate decisions. Data produced by sensors and other devices is enabling deeper connections between operations and decision-making components. Since these data-driven diagnostics are utilized to support systems intelligence and decision-making capabilities, their integrity is essential.

Furthermore, data integrity becomes even more crucial as machine learning and artificial intelligence (AI) are used to support decision-making more and more often. To ensure that AI models accurately reflect the reality that the system is forecasting, they must be grounded in high-quality training data, and their inferences must be supported by high-quality operational data.

ITTIA DB IoT offers data integrity by ensuring that data is stored and processed according to the data model specified by MCU application designers. When data is stored, write operations are carefully sequenced to prevent data corruption.

In addition, ITTIA DB IoT provides a path for device manufacturers to scale up MCU applications to different levels of computing by enabling database features and functionality only when additional resources are available.



Embeddable Edge Data Management

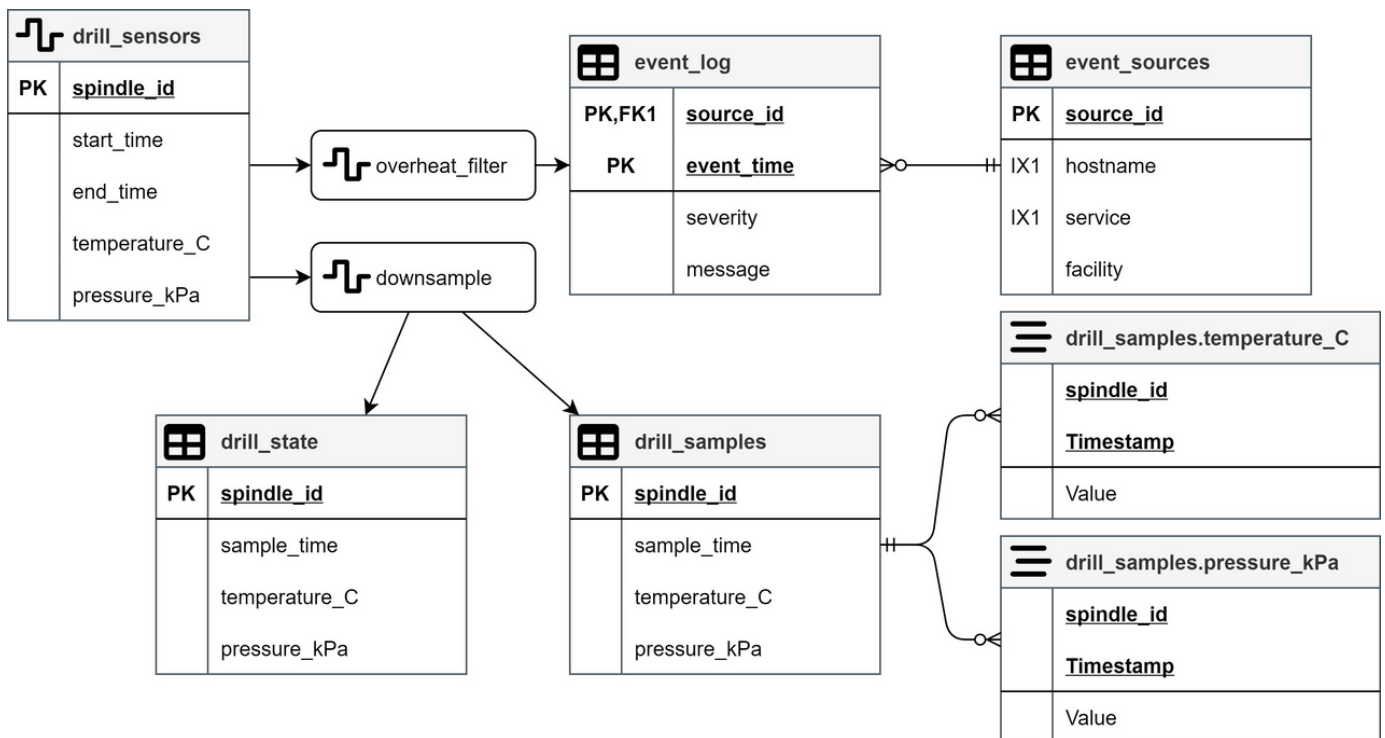
## Use Case: Gain Insight by Streaming Data with Cortex-M Device in an Industrial Drill

Device manufacturers must design software to process a high volume of time series data in real time with high throughput. For real time events, the device must react with low latency. Many decisions need to be made concurrently in multiple threads, tasks, and processes while processing results are captured to local flash media.

In industrial automation, complex machines continuously produce and consume data about a series of real-world tasks. For example, an industrial drilling machine must monitor the temperature of the drill bit and apply a cooling fluid to prevent overheating. If the temperature exceeds a certain threshold, the machine must stop operating to protect the equipment. After a certain amount of use, the drill bit must be replaced, and other maintenance must be performed from time to time.

Many of these activities can be automated by embedded software installed on the industrial drill. On a Cortex-M device embedded in an industrial drill, ITTIA DB IoT can be used to monitor drill bit temperature and cooling fluid pressure, log significant events, such as overheating, and analyze stored data for preventative maintenance:

- Measure temperature and pressure at a high sample rate.
- Update recorded sample values once per second.
- Respond to high temperature events with predictable, low latency.
- Query data recorded in the event log..



## **Accuracy, completeness, consistency, and validity**

Data quality, which is essential to all data governance activities involving embedded systems, quantifies how well a dataset satisfies requirements for accuracy, completeness, and validity. Databases built for MCUs offer data quality that the system needs for making the best data-driven decisions.

An embedded system can trust MCU data and use it to make better decisions only when the quality of the data satisfies the requirements for the use for which it is designed. Nevertheless, databases designed for MCUs aid in the diagnosis of underlying undesirable data when a principle isn't satisfied.

As manufactures move toward incorporation of automation and artificial intelligence (AI) into their processes, data quality and database are becoming more and more important.

## **Data ownership**

The quantity of data produced at the edge with MCU devices is growing exponentially. Data is therefore very important, and it is becoming less and less practical to handle large datasets in a single location due to cost, performance, and privacy concerns. It is desirable and feasible to shift the computing close to where the data originates, with the suitable edge database, rather than relocating all raw data to a centralized location where computing is abundant. Database for MCUs can become a natural extension of cloud computing and has enormous cost savings potential. The MCU data management system is much more appealing when combined with qualities like data ownership, decreased latency, optimum bandwidth consumption, offline or autonomous operation, and adherence to regulatory or security criteria depending on physical location.

## **Security: Encrypt, Authenticate, and Authorize Data Access**

The Internet of Things (IoT) hugely magnifies embedded system security challenges, since devices must connect and communicate to each other, as well as the Internet. Intruders and bot networks have many new opportunities to enter, and unprotected data stored on IoT devices is vulnerable. What can you do to protect device data?

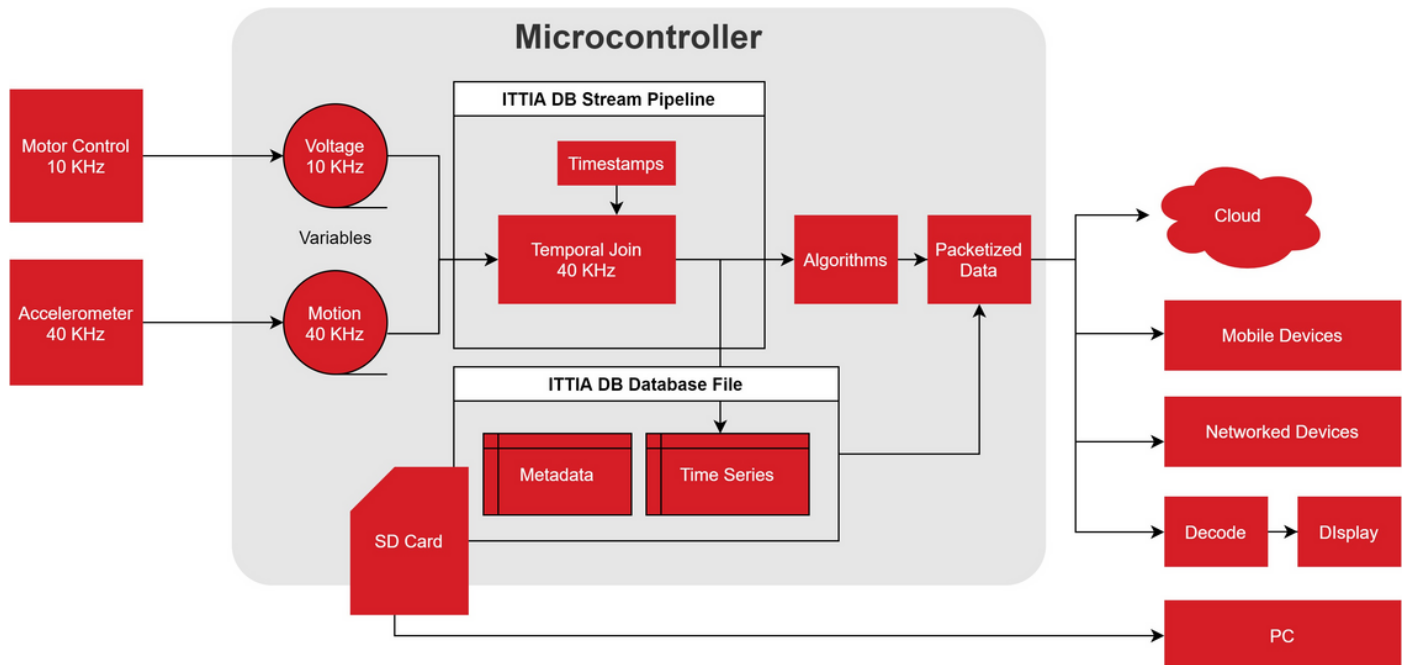
A database with layers of protection will include encryption and ability to support hardware security layers. These security principles are proven and well established,



## Use Case: Sensor Fusion by Synchronizing Timestamped Variables on a Device

Combining data from high frequency data sources is not trivial for data that originates from the Internet of Things. Sending every sampled value to the Cloud consumes a significant bandwidth cost, but processing data on IoT devices themselves requires synchronization of the data sources before algorithms are run on Cortex-M MPUs. Design decisions, such as where to store the data and the medium of communication, will further limit the rate data can be processed.

Consider a ceiling fan that combines data from voltage and motion sensors to identify correlations that affect efficiency and lifetime. By measuring voltage from the motor control at 10 KHz and motion data from an accelerometer at 40 KHz, a Cortex-M microcontroller can achieve sensor fusion. ITTIA DB's temporal join operator combines data from various timestamped data sources within a configurable time interval. The results are sent to various algorithms and stored as time series in an ITTIA DB database file. The results are sent to various algorithms and stored as time series in an ITTIA DB database file.





## Performance and Footprint

IoT device applications usually perform many different tasks on data stored in the database. Some tasks perform best when run in parallel, allowing long-running activities such as synchronization to complete without first stopping normal operations. Tasks can be performed by a single application with multiple threads or tasks.

Database performance and footprint are major challenges with any MCU development discipline. Processing data closer to the source reduces the amount of data flowing to and from the primary network, leading to lower latency and better throughput.

ITTIA DB IoT for MCUs provides development flexibility so you can design data models, database schemas, and application code to solve complex queries with high performance, small footprint and enable rapid application development.

ITTIA DB IoT a highly concurrent database, offers applications to work with multiple queries and supports reads and updates to the database at the same time. Database provides development flexibility so you can design data models, database schemas, and application code to solve complex queries with high performance and enable rapid application development.

## Affordability

When it comes to MCUs, traditional flat file data management and free open-source solutions are both costly, mostly from the significant upfront expenditures associated with customizing the database and integrating it with real-time operating systems, hardware, etc. Thousands of dollars may need to be spent up front to cover these and other related integration and customizing expenses. In contrast, lower initial and long-term team expenses can be achieved with ITTIA DB IoT, special database built for MCUs that can silently embed and run on the device. Additionally, there will be significant cloud data storage savings, as a significant amount of data is now processed and managed locally on the device. Real-time data queries may also completely eliminate the need to transfer data from a device to other systems and the cloud, saving significant cost.

## Integration Challenges

Building embedded MCU applications introduces a distinct set of difficulties, including scalability, real-time performance assurance, integration and interoperability problems, and more.



Embeddable Edge Data Management

Applications created for MCUs are invariably components of larger systems; examples include digital watches, smartphones, automobiles, and automated industrial equipment. These systems typically feature integrated data management software, real-time operating systems, hardware, IDEs, and other components. In terms of databases, these embedded solutions need to respond instantly in all situations, within the time frame given by the design and function with constrained memory, processing power, and energy availability. Furthermore, since processors, sensors, and other hardware components might change over time, embedded software must be resistant to changes in its working environment. The criteria of autonomy and portability for embedded software are very difficult. Ensuring that all the different components are working together and efficiently is the key challenge. This is not a simple task, and manufacturers must constantly exercise caution when choosing the database that will enable integration with other software and hardware.

### **ITTIA DB IoT for MCUs**

The member of the ITTIA DB family made specifically for microcontrollers is ITTIA DB IoT, which supports time series, relational tables, and data streams in a single portable software library. Developers are free to choose any combination of these features. ITTIA DB IoT is designed to offer microcontrollers real-time data processing and data management for a variety of data points, including metrics, events, logs, and more. Performance, compact size, and adaptability are the significant features of ITTIA DB IoT for microcontrollers.

### **Conclusion**

Whether dramatically improving an autonomous robot's production capabilities, keeping up with a building's temperature in real time, or monitoring personal health, manufacturers greatly desire a reliable data management solution. As product requirements and hardware have evolved, the data management needs are becoming more obvious. A wide variety of microcontrollers can benefit from a modern embedded Edge database, like ITTIA DB IoT.

ITTIA DB IoT enables manufacturers to closely focus on data management and processing, starting from design and development, and continuing all the way through deployment and production.

Contact ITTIA today and learn how ITTIA DB IoT efficiently solves your greatest MCUs data management challenges.



## **Disclaimer**

Information in this document is provided solely to enable system and software implementers to use ITTIA products. No express or implied copyright license is granted hereunder to design or implement any database management system software based on the information in this document. ITTIA reserves the right to make changes without further notice to any products described herein.

ITTIA makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does ITTIA assume any liability arising out of the application of or use of any product, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. Statistics and parameters provided in ITTIA white papers and data sheets can and do vary in different applications and actual performance may vary over time.

All operating parameters must be validated for each customer application by customer's technical experts. ITTIA and the ITTIA logo are trademarks or registered trademarks of ITTIA L.L.C. in the U.S. and other countries. All other product or service names are the property of their respective owners.

Copyright (c) 2024 ITTIA L.L.C. All rights Reserved. References in this document to ITTIA products and services do not imply that ITTIA intends to make them available in every country.



Embeddable Edge Data Management