Streaming Data Architectures for IoT Analytics

Authored by Radiant Advisors

Sponsored by AEROSPIKE

Produced by database TRENDS AND APPLICATIONS

Streaming Data Architectures for IoT Analytics

TABLE OF CONTENTS

Introduction 1
- *Example: Healthcare*

Digitization of the World 1
- *Example: Digital Twins*

Streaming Data Architectures 2
- *Example: Industrial Internet of Things (IIoT)*

Business Value with IoT Data and Analytics 3
- *Example: Smart Cities*

Data Architecture Principles for IoT Data and Analytics 4

Two-tiered approach 5

Core component 5

Refined three-tiered architecture 5

Paradigm shifting 6

The edge 6

Database at the edge 7

The Aerospike database 8

Three architecture patterns for scalability 8
- *Perspective: IoT PoCs in the Cloud Learnings*

Necessity of Reliable Operations with IoT Data and Analytics 9
- *Autonomy at the Edge*
- *Recoverability at the Core*
- *Resiliency for Business Continuity*
- *Recognizing and Insulating Against Schema Drift*

Conclusion 11

Streaming Data Architectures for IoT Analytics

by John O'Brien, CEO and Principal Advisor, Radiant Advisors

More and more of our devices and processes are "smart." For example, your smartphone has more processing power than the mainframes that solved the trajectories to navigate our paths to the Moon and back. When we speak about the Internet of Things (IoT), we are talking about the interconnection of thousands of devices with the tremendous computing power available to power Artificial Intelligence and Machine Learning (AI/ML). Intelligence is the sum of all the information across the network of devices, sensors, and the cloud. No longer is data aggregation required to derive intelligence. Rather, real-time insights and decisions at the moment where they can have the most significant positive impact are being derived by streaming data.

Example: Healthcare

Consider the example of connected health care for the management of diabetes. There are patches that can continuously monitor glucose levels and send these back through a smartphone to an insulin delivery system that supplies exactly the right amount to keep blood sugars at the optimal level. It doesn't end there, as the information is sent up to the cloud where continuous Machine Learning algorithms are processing that individual's information as well as plugging that information into larger data sets across individuals to gain new insights on how to manage this complex internal system. There are feedback loops between the patient and the drug delivery system, as well as from the model and continuous learning driven by the larger data sets delivered to the cloud.

Digitization of the World

When we speak of the digitalization of the world, among other things, we are referring to the models driven by sensor data. This is sometimes referred to as "digital twinning." There is the device, the building, the car, or the airplane that exists physically in the world, and then there are the digital models that represent those in the digital world. These devices, and even the systems in our bodies, can now be monitored continuously and understood in terms of digital models. Decisions can then be made to adjust or take corrective actions in the moment.

The models are similar whether we are talking about systems within homes, manufacturing, buildings, cars, trucks, or planes. So what is required to handle the data capture and decision-making in real-time when it makes the most difference?

Example: Digital Twins

Whether designing a digital version of a Smart City, an Industrial Internet-of-Things (IIoT) factory floor, or an aircraft engine, digital twins provide a modeling and simulation option for improved designs and maintenance—both in new initiatives and in retrofitting existing systems. Digital twins can be expected to live in any IoT-enabled facility. For example, in airplane manufacturing, digital twins simulate aircraft jet engines that generate 400k records per second per plane along with 24k parameters to monitor per aircraft—in what Airbus describes as a "sky-wide data analytics platform." For smart cities, digital twins of a municipality are modeled first and then used to aid in building the first generation of that city. These simulation models help improve the design and, more importantly, help IoT-enabled initiatives maintain high efficiency with reduced downtime and encourage cost savings by isolating corrective issues before they arise.

Streaming Data Architectures

The architectures to enable us to process data in motion are called streaming data architectures. These architectures are quite different from business analytic models of the past. Typically, this model is a two-speed pattern. First, there is the system at the edge that supports data ingestion at ultra-low latency (within microseconds to single-digit milliseconds) and performs fast local analytics. Then there is the requirement to supply that

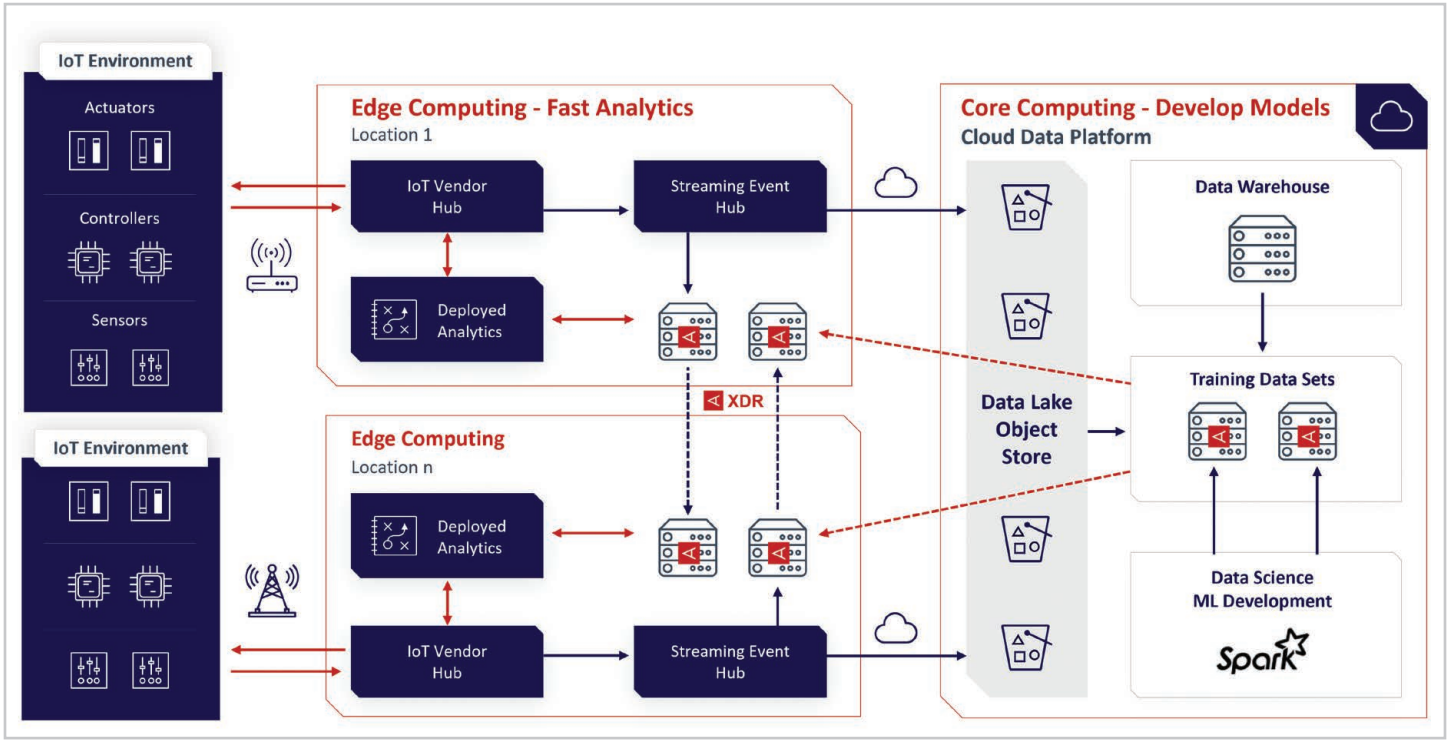


Figure 1 – Example of a functional Streaming Data Architecture

data to the AI/ML systems at the core, to generate and continuously update the models and decisioning systems. This process is commonly referred to as "training the models."

These new applications that drive decisions and actions have to be incredibly reliable. The data architectures must be fault-tolerant, remain current, and provide consistent performance. There is a new mission-critical; it is real-time mission-critical. Data which flows in streaming architectures changes over time, growing as new use cases are found for the information and new devices are added to the network. There is a need for the architecture to be elastic, to grow as needed when there are spikes in information or activity, and to grow as the business opportunity grows.

Example: Industrial Internet of Things (IIoT)

The terms IIoT and Industry 4.0 refer to the use of IoT devices in manufacturing (both discrete and continuous) and factory floors to further optimize production operations and quality. Rather than a single device to collect information on product quality, in IIoT, a piece of machinery is digitized to measure multiple variables on any piece of that machinery (for example, equipment vibration, friction, tension, power consumption, temperature, noise, and so on). In this way, every piece of factory equipment can be digitized to instrument the entire factory floor with the shared goal of cohesively producing the highest output of product in the most efficient manner. Because it is a highly integrated process, any weak link in the assembly line stops everything else. However, being able to manage the factory floor with early detection of impending issues enables proactive responses, such as redirected workflows for maintenance, and predictive maintenance that can reduce collateral damage to increase overall up-time and quality. This approach makes factories smarter by integrating all IoT data in real-time to optimally balance throughput, machinery and robotic health, and power consumption to deliver the most high-quality products with the least amount of costs in energy, maintenance, and downtime.

Business Value with IoT Data and Analytics

The Internet of Things (IoT) can be thought of as a network of interconnected sensors and processes that integrate with a system of models derived through AI/ML. This system of models allows for decision and action at machine speeds in controlling devices, and systems of devices. The ability to understand and to react earlier reduces risk, cost, and exposes new opportunities within the broader business. Millions of data points are ingested and then supplied to models that are coordinated by higher-level systems. Individual decisions can be made in milliseconds resulting in coordinated systems that can manage our power grids, monitor and manage networks, manage buildings, manage cities, fly planes, and drive our cars.

Business value is realized at the moment that an event (or meaningful combination of events) is recognized and acted upon.

Example: Smart Cities

Smart cities are continuing to gain traction worldwide, accelerated by the continued rollout of 5G

communications. Focused on improving citizens' lives, the initiatives are considered either green-field initiatives for new cities or brown-field initiatives intended to retrofit cities. While green-field cities focus on making the city holistically self-sufficient and optimally managed, brown-field cities choose to leverage IoT to tackle three top priorities: traffic management (reducing congestion and reducing the effect of pollution on air quality), energy consumption (shifting toward requiring less energy production), and improving citizen services and transparency (such as monitoring city government spending, and improving public services like waste management, city lighting, and maintenance of green spaces). In all these endeavors, smart cities leverage IoT to move from reactive to predictive capabilities in city optimization—and even beyond operations to deliver community services, like alert communication, to densely populated urban areas. Smart city builders strive to ensure data security and citizen privacy is protected using anonymized data and protected networks.

Data Architecture Principles for IoT Data and Analytics

The data architectures needed to accommodate processing IoT data go beyond established rules and necessitate new, modern paradigms—ones designed for scalability, capture, and high-performance that align with the steps necessary for developing and deploying AI models through the enterprise for internal business systems, operations, and customer engagement applications.

There are several key tasks that a data architecture for IoT must perform:

1. **Ingestion or capture of data across many sensors.** These data points are streamed over the network and must be buffered so that different windows of data can be supplied to a number of models. The ingestion of many streams of data at high velocity means that high throughput in terms of writes must be supported by the data store. Additionally, the system needs to ensure that collected data in the pipeline is not lost and data integrity is maintained.
2. **Consolidated for processing.** Next, the data across many instances of the device or processes must be consolidated for further processing, e.g., model training, the discovery of new patterns, and opportunities. The faster the database is, the faster models can be trained, re-trained, and the tighter the feedback loops of continuous learning and improvement.
3. **Correlation within the broader ecosystem.** The data architecture is not only about databases but also about connectivity with the broader ecosystem of data movement and processing provided by such technologies as Kafka, Pulsar, Spark, and Flink among others. The vast amounts of historical event data need to be correlated with related data from operational systems and external sources to identify which data elements are useful to enrich the data set for better predictions and decisions.
4. **Scalable and reliable.** The entire system must be elastically scalable to be future-proof. In practical terms, this implies that the technologies are based on distributed systems incorporating clusters of commodity servers connected through networks and using cluster management algorithms to achieve high performance with extreme reliability.

With this in mind, let's look at various architecture elements that support successful data analytics including AI/ML operations.

Two-tiered approach

As mentioned earlier, the architecture as seen above in Figure 1 supports continuous streaming with a two-tiered approach for both handling data ingestion from IoT devices and consolidated accumulation of historical IoT data. This includes both event data and polling data snapshotted at predetermined intervals. Data scientists need vast amounts of historical IoT event and actuator data, correlated with other related data from operational systems and external sources, to iteratively identify which data elements contribute to predicting (or identifying) an outcome that can help mitigate business problems or achieve business goals. These statistical processes require a scalable high-performance database to support development and testing. Once these AI models are developed, they can be deployed to the edge of the company's computing infrastructure with the goal of being closest to IoT devices that can effect change. The IoT streaming architecture at the edge is unique in its purpose to independently support analytics while ensuring IoT data is continuously streaming to the core of the data architecture.

The core IoT data architecture supports AI model development and deployment with a high-performance database that sustains multiple persistence options with streaming data ingestion and integration.

Core component

The main framework of this architecture is an independently scalable, high-performance core component where all data is streamed in from nodes on the computing edge. Those nodes are localized, fully self-reliant components of the architecture that perform data collection from IoT devices, store data locally, and **execute analytics on the spot**, then funnel data back into the consolidated, cloud-based core data lake architecture so data scientists and other analysts can build data sets to develop and retrain holistic analytic models. Having any and all data available improves the quality and stability of any analytic model—not building models at the edge but **applying models at the edge to take action**. (Local rules can be set up to enhance or override the output of analytic model processing at the edge).

Refined three-tiered architecture

This refined architecture is made up of three tiers: streaming IoT devices and sensors, the edge computing node as a local collection of resources, and a centralized cloud-based platform to perform history and build new models. It is important to note that a streaming-first mindset for data architecture must focus on continuous data flows and also have the capacity to reroute incoming, asynchronous data into buffering pools. Tackling data in the smallest unit as event records or event data sets is a key principle in architecting for scalability. Because data never stops being generated, architects must become masters of redirecting to manage the flow of data in streaming architectures. Data may pool but never dams, and data pipelines have to be able

to catch up faster than they fill up. At scale, this architecture allows for automation and self-healing capabilities without compromising data loss.

IoT data collection and embedded analytics exist at the edges of the infrastructure while streaming data to a consolidated database at the core. Acceptable databases must have an optimized integration with leading event streaming hubs (such as Kafka, JMS, Spark, and Pulsar) while also providing high-performance queries for data correlation or aggregate functions.

Paradigm shifting

The paradigm shift away from a centralized data processing architecture to a distributed one can be challenging to grasp. A distributed processing architecture is necessary to bring all IoT device data together, correlate data, perform analytics, and then take action. However, the turnaround time for this process can become bloated when the framework in place adds latency across a network from the moment of data capture, until the data is sent, reviewed, processed, analyzed, and results returned. In order to expedite decision-making, this new architecture benefits from an edge computing node. Here, a distributed architecture works best on the cloud because the cloud is completely scalable and elastic.

The edge

At the edge, network connectivity is needed to connect to the data, as well as commodity com-

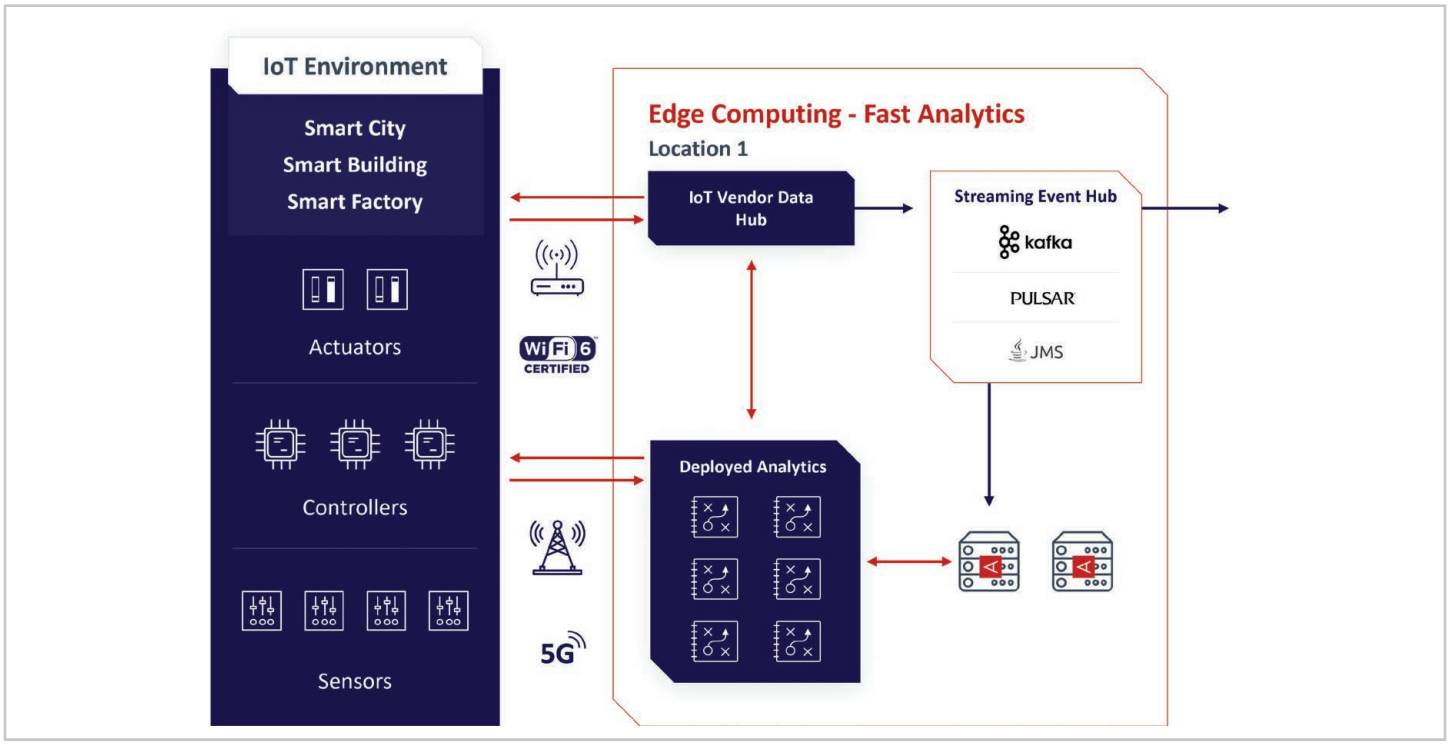


Figure 2. Edge Computing environments enable analytics with the fastest response times

puting power and a limited amount of storage. An important point: it is not necessary to retain historical data at the edge. This is flushed through to the central platform. However, a best practice is to store at least five to seven days' worth of data on-site and use this to estimate how much storage is needed, how many CPUs, and how much memory is required in order to ingest/upkeep. This redundancy helps insulate against an outage that challenges network reliability if, for example, an independent edge node becomes disconnected from the central processing cloud and can't ingest data.

The core of the IoT architecture requires a high-performance, scalable database platform to support data scientists in developing analytics models for ML and AI with Apache Spark.

Database at the edge

It is essential to discuss the necessity of the database within the context of this architecture. Streaming data hubs like Apache Kafka strive to place the least amount of data transformation to increase reliability at scale. However, they struggle when trying to maintain the current state of data, or an integrated operating window of data, such as a "rolling 60 minutes" or "past 24 hours." There are a variety of workarounds but having an industrial-grade database that accepts streaming data ingestion without any scale limitations, serves high-performance queries, and stores additional relevant data at the edge is necessary. The database should retain at least seven days of history as a streaming buffer for the consolidated core data platform.

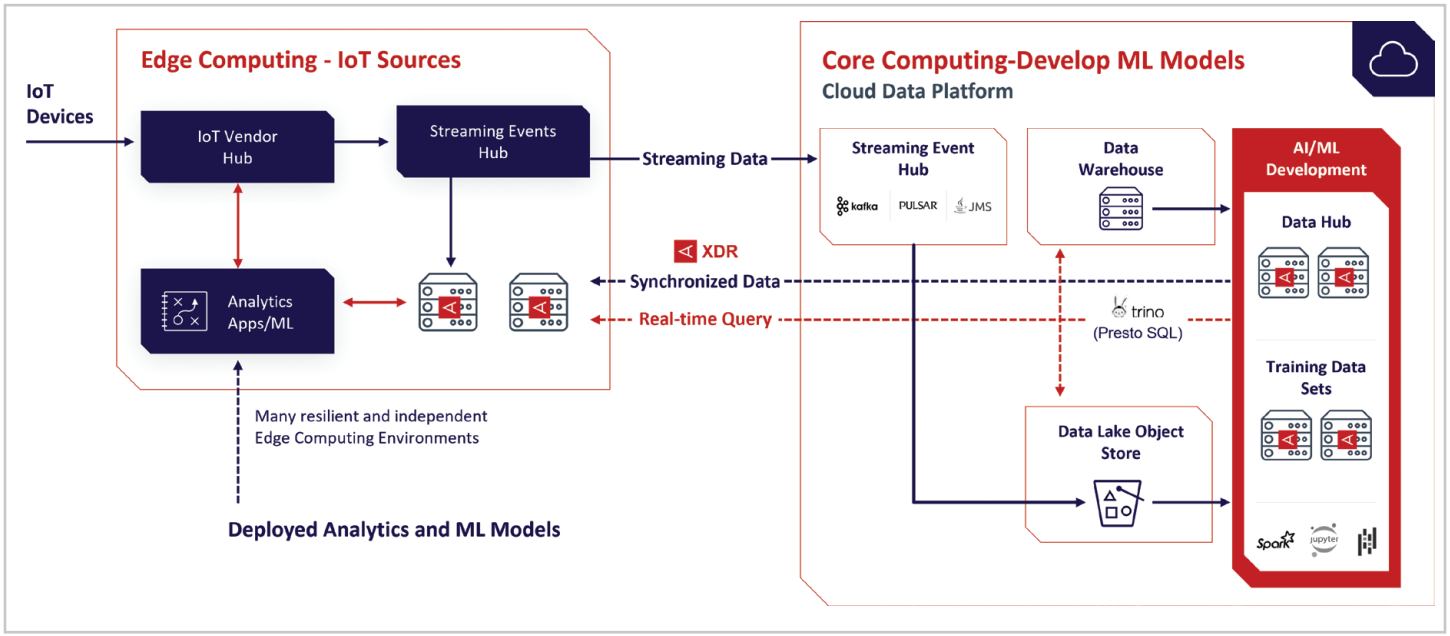


Figure 3. Edge Computing Nodes Stream Data to a Core Cloud Data Platform for AI/ML Development

The Aerospike database

The Aerospike high-performance database comes with all of the capabilities discussed above. In addition, it offers real-time database replication among edge nodes for fault tolerance and also between edge and core to maintain "freshness" between data captured and data used for model generation. Aerospike as the core data platform enables data scientists to work with large data sets with the high performance required for iterative model feature reduction and correlation analysis. A cloud data lake should serve as the data repository for all enterprise data assets. In contrast, the database access layer performs the analytics functions needed in data science, AI/ML, and enterprise data hubs. With a ubiquitous SQL interface and database security, data unification can also be accomplished with federated query engines, such as Apache PrestoSQL/Trino, which can connect and execute queries that integrate databases on IoT edge nodes, the cloud, on-premises operational systems, or data warehouses.

Three architecture patterns for scalability

There are three architecture patterns for scalability which should be considered as part of the enterprise IoT data strategy. IoT proof of concept (PoC) projects and data science experimentation are commonly being carried out on cloud platforms and have become quite useful in gaining experience in cloud-native architecture and principles. However, when operationalizing an IoT project with an increasing number of deployed IoT devices and sensors with low latency requirements of sub-second predictive and prescriptive analytics, data traveling across the network adds measurable delays. Edge computing nodes may emerge as needed to meet these new requirements, but a consistent repeatable distributed architecture from the beginning will improve manageability and reliability. Finally, there are some instances of hyper-scalable IoT data architectures where the edge computing nodes collect too much data to feasibly send across a network and edge node data storage becomes inevitable. In some cases, reduced or distilled data can be streamed to a centralized core data lake in the cloud, but there will also be the capability of distributed database queries of the edge computing databases for essential applications and data scientists.

Perspective: IoT PoCs in the Cloud Learnings

Radiant Advisors has worked with companies that embarked on successful IoT PoCs that were built out entirely on the cloud leveraging cloud elasticity and many available services and automation. IoT data collected into the cloud where streaming data pipelines performed analytics and made decisions about predictive maintenance or next best actions. While this basic "consolidated architecture approach" works in concept, deploying it in production may eventually fail to scale and most of the development would have to be rewritten. Instead, edge computing infrastructure should be leveraged as "agents" to process IoT data closest to the devices and their actions. There should also be a focus on how to ingest data from tens, or even hundreds, of thousands of devices all streaming data in real-time that also need to react to the data in a moving window and an integrated environment. As an example, consider how devices in a smart building might behave: measuring temperature, humidity,

air quality, sound levels, light, and other inputs. While this data is meaningful, to make an HVAC system more efficient, additional external variables—like light filtration, outside weather conditions, etc.—should also be correlated. Further, these variables are not fixed and make up a "living system" as weather and light change throughout the day. At some point, the building becomes optimally efficient through machine learning more so than reacting to thresholds which is unique to the building's location and usage. At its core, IoT is all about devices talking to each other to make decisions about an optimal environment without humans getting involved.

Necessity of Reliable Operations with IoT Data and Analytics

IoT systems must be reliable in terms of always-on consistent performance and resiliency. In the past, a mission-critical system such as check processing could be down for a few minutes and then "catch up." If the system is piloting an airplane or driving a car, it cannot fail, and it cannot fall behind in its processing of the myriad of decisions that have to be made continuously. This is why the systems of choice for IoT are distributed, redundant, and elastically scalable.

Key points regarding resiliency and what it means to be always on:

a. **Local database requirement:** The systems running on or near the devices require local databases for ongoing operations during networking failures. Local databases support streaming analytics providing a moving window of IoT data. Data is typically held for a set period of time, such as seven days. A Time-to-Live (TTL) is set for the data and the system deletes the data when that time is reached.

b. **Core database redundancy:** The core database must be redundant and able to absorb bursts of data from multiple sources. This means that "headroom" must exist in terms of ingestion and processing power. Systems must be redundant across hardware racks with independent power and network sources. Many systems are now redundant across data centers. This places a requirement on the databases to deal with consistency, either through synchronous or asynchronous replication.

In addition, there are a few more things to consider.

Autonomy at the Edge

Autonomy at the edge is needed with enterprise class databases for ongoing operations during network connectivity failures with the core data architecture. A local database is required to support local analytics with a moving window of IoT and reference data. As suggested above, this database should be configured for at least seven days of data storage if connectivity to the core database is unavailable. Event hubs like Apache Kafka and local databases are set up with a combination of carefully designed checks and balances for purging data after seven days, rather than processing deletes after core database subscribers have received the streaming data.

While streaming data is a preferred way to send data to the core, the database at the edge node is important for deployed analytic routines that need to access data locally. A database at the edge will also support

replicated data sets from the core, such as mastered reference data or key entity information, and buffering data as a backup to a streaming process allows the database to fill in gaps in the event of any missing data. Similarly, having a database at the edge also enables much easier data access with distributed queries from core applications if needed (for example, during a network outage, users can query a remote database).

Recoverability at the Core

Recoverability at the core will require a high-performance database that can perform a "catch up" process for IoT edge data streams at a minimum of 3x normal processing speed in order to recover from an interruption in data processing. Within the edge architecture, a data collection hub, such as JMS or Apache Kafka, can collect raw IoT data feeds; streaming data applications at the edge can integrate and enrich the data sets into the database and for the downstream core data lake.

This streaming data requires data pipeline developers to ensure the core database has a mean-time-to-recovery that can be reliably calculated. For example, a three-day network outage would be recovered in one day of catch-up stream processing. This requires that very little data transformation can be performed on the primary streaming data ingestion pipeline. Ultimately, recoverability depends on the IoT data. This can be extracted from the database, or—as a secondary option and fail-safe for resiliency—extracted from the raw data stream.

Resiliency for Business Continuity

Resiliency includes leveraging databases with active-active *and* active-passive multi-geographic processing capabilities. Therefore, a database at one edge node can replicate its data to another database on a different edge node and remain available if the first edge node fails or there's loss of network with the core data platform. Likewise, master reference data at the core can be replicated to an edge node location to support analytics at the edge. Replication can be used as a way to move data, or a streaming data hub can be employed.

Note: Aerospike can provide *synchronous* Active-Active clusters/datacenters.

Recognizing and Insulating Against Schema Drift

It is realistic to expect that IoT devices will collect data in different data structures among multiple vendors for the same IoT function and across multiple model numbers from the same vendor/generation of product. Thus, schema drift from IoT data generation and application changes over time will eventually impact downstream systems and analytics.

Flexibility of NoSQL databases allows for schema drift volatility over time. It is good practice to identify a common set of core data elements that can be typed and structured with a database and support the flexibility of additional data elements which varies from device to device. With high-volume streaming data, interruptions and downtime present unaffordable issues. Therefore, having the flexibility to absorb and process changes, or adjust the system without interrupting the data flow, insulates streaming architecture with the dynamic capability needed for business continuity.

Conclusion

IoT requires always-on consistent high performance at any scale. The data architecture has to support extremely low latency, high availability (nay absolute availability) and must scale gracefully.

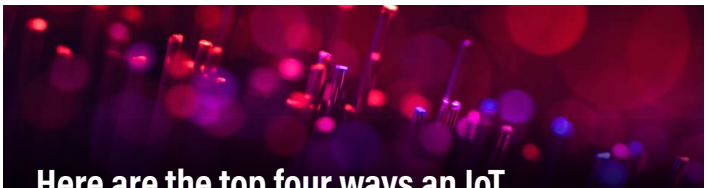
The IoT systems of today are multi-tiered. At the edge, supporting fast ingestion of data and rapid inference based on access to a window of data in motion; at the core, often in the cloud, the ability to process vast amounts of data to train models and to discover new insights. These tiers must be connected by high-speed data pipelines that can enrich and deliver the right data to the right applications.

The distributed architectures of today's non-relational databases coupled with data delivery systems like Kafka and Pulsar and the distributed processing by systems like Spark and Flink give us the elements to compose these systems.

The business value derives from greater efficiency, more reliable systems and devices through better management of them, and compression of risk. Making better decisions faster at the point in the process closest to the issue or opportunity changes business outcomes. The future is in real-time aka the "power of now."

Aerospike is one of the few proven databases/data platforms that meets the requirements of modern streaming architectures.

Overall, Aerospike's unique ability to be a high-performance, resilient edge database while also supporting centralized, consolidated, large-scale parallel data processing for data science and ML development capabilities allows it to fulfill the dual-mode requirements of a streaming IoT architecture.



Here are the top four ways an IoT architecture benefits from Aerospike:

1. **Achieves low-latency and high-performance at the edge:** The Aerospike database meets the requirement for streaming data ingestion necessary for up-to-date and near real-time data and analytics.

2. **Can specify multiple tiers for data persistence:** Aerospike optimizes for edge computing performance. This includes data and/or indexes persisted in-memory or on flash drives available on commodity edge computing servers.

3. **Meets the resiliency requirements needed for continuously streaming data environments:** Aerospike has the ability to persist data in distributing locations, thereby making recoverability a built-in configuration for data architectures that will inevitably have to mitigate failures.

4. **Flexibility to be deployed both at the edge and at the core:** Aerospike's ability here and a highly efficient data replication feature (cross-datacenter replication, XDR) working between them make it highly suitable for this kind of use cases.

About John O'Brien:

John O'Brien is Principal Advisor and CEO of Radiant Advisors. A recognized thought leader in data strategy and analytics, John's unique perspective comes from the combination of his roles as a practitioner, consultant and vendor CTO.

About Aerospike:

Aerospike unleashes the power of real-time data to meet the demands of The Right Now Economy. Global innovators and builders choose the Aerospike real-time, multi-model, NoSQL data platform for its predictable sub-millisecond performance at unlimited scale with dramatically reduced infrastructure costs. With support for strong consistency and globally distributed, multi-cloud environments, Aerospike is an essential part of the modern data stack for Adobe, Airtel, Criteo, DBS Bank, Experian, PayPal, Snap, Sony Interactive Entertainment, The Trade Desk, and Wayfair. A global company, Aerospike is headquartered in Mountain View, California, with offices in London, Bangalore, and Tel Aviv.