research

CHECKLIST REPORT 2021

Data Architecture for IoT Communications and Analytics

By David Loshin

Sponsored by:

CLOUDERA

tdwi TRANSFORMING DATA WITH INTELLIGENCE™

JANUARY 2021

TDWI CHECKLIST REPORT

Data Architecture for IoT Communications and Analytics

By David Loshin

TABLE OF CONTENTS

tdwi

Transforming Data With Intelligence™

555 S. Renton Village Place, Ste. 700
Renton, WA 98057-3295

T 425.277.9126
F 425.687.2842
E info@tdwi.org

tdwi.org

This checklist is an update of the original report published in 2018.

Product and company names mentioned herein may be trademarks and/or registered trademarks of their respective companies. Inclusion of a vendor, product, or service in TDWI research does not constitute an endorsement by TDWI or its management. Sponsorship of a publication should not be construed as an endorsement of the sponsor organization or validation of its claims.

FOREWORD

The Internet of Things (IoT) is an architectural paradigm that combines physical devices (with embedded sensors and actuators along with the necessary implementation software), massive network connectivity, data accumulation and analysis, and operational control informed by analytics models that enhance and optimize cross-system interoperation. In turn, a successful IoT deployment can be leveraged to improve the extended system's performance. For example, integrated IoT analytics can help factories reduce or eliminate unscheduled downtime through predictive maintenance, enable manufacturers to improve production quality through continuous monitoring of assembly lines, and anticipate and consequently eliminate power outages across an energy utility's regions by monitoring power usage across a broad network of smart meters. IoT analytics can even alert people to potential health risks by monitoring wearable fitness monitors.

However, systemic objectives for IoT are less about the operation of any single device in the network and more about overall business performance optimization that can be achieved through visibility of all the devices and nodes in the network. The foundation of a successful IoT implementation is a technical architecture that blends network connectivity with an information architecture for streaming, ingesting, filtering, and capturing data. This must be coupled with a means of analyzing the data, creating analytics models, and pushing those models back to the edge nodes in the IoT network.

Executing this vision demands organization and discipline when it comes to data management and oversight, requiring information models (including metadata and searchable services) for simplified integration. Finally, the environment must also integrate continuous monitoring to determine when objectives are being met or when there are opportunities for additional improvements.

This checklist explores some fundamental aspects of the data architecture necessary for IoT success. It examines what is required to enable an environment that can rapidly adapt to the dynamic nature of massive numbers of connected sensors and other endpoint devices, communication and data streaming, ingestion and analysis, and deployment of developed analytics models for automated decision making.

Here are eight key suggestions to evolve your data architecture for an IoT environment.

1 ENVISION, DESIGN, AND GOVERN THE FLOW OF DATA IN MOTION

One goal of IoT analytics is to analyze data collected from diverse, real-time data streams and develop models that improve *global* business performance across the extended enterprise. The logical end state of the IoT environment integrates many prescriptive models whose actions all incrementally contribute to optimizing overall business outcomes.

Reaching that end state requires a careful examination of the proposed IoT network topology and how different connected components interoperate, including:

- **EDGE DEVICES.** Identify the sensors/edge devices at the endpoints of the IoT network, assess their connectivity, and evaluate the generated data streams and how those data streams are communicated. Identify the actuators located at the endpoints and the communication methods for controlling them.
- **EDGE COMPUTING NODES.** In many IoT architectures, localized computing resources are used to accumulate sensor data streams and package them together to be forwarded to a centralized computing and analytics system. In addition, edge nodes provide a site for localized streaming analytics models for oversight and control of the end nodes within an assigned jurisdiction.
- **CLOUD/DATA CENTER.** All data gathered from the hundreds of thousands of edge devices is ingested and processed in real time at a local data center or in the cloud. This data is further pushed into operational stores and data lakes to generate operational insights and build machine learning models.

Devise a data flow architecture that models the network connectivity—how the information from the different streams is accumulated and managed, how the various data streams are connected through edge nodes to a centralized cloud platform, and how analytics models are created and subsequently integrated back into the network. Account for the realities of bandwidth dynamics, especially designing data flows in an environment where the sensors are not necessarily static, but moving.

Ensure that your selected platform architecture supports the end-to-end data flow aspects of IoT data communication:

- Routing and streaming data from the network endpoints to the centralized analytics platform
- Introducing bandwidth controls (such as back pressure controls and dynamic data package reduction when bandwidth is low or throttling packets until the bandwidth improves)
- Data acquisition, transformation, filtering, and analysis as well as deployment of analytics models

Choose a platform that facilitates the data flow across different types of devices and machines, captures data source and transformation provenance (to support traceability and consistency management), and allows you to control and manage the end-to-end data life cycle across the environment.

2 ENABLE EDGE DATA COLLECTION AND PROCESSING

The beauty of an enterprise IoT analytics strategy is that it can integrate predictive analytics to influence automated decision making in real time via streaming analytics processes. Although the conventional approach to event-stream processing typically involves forwarding data to a centralized environment where a stream processing engine is expected to trigger desired events, this approach likely will not adapt to the IoT environment.

Organizations cannot configure a high-volume IoT network in which the outlying devices need to wait for a latency-filled process to collect the edge data at periodic intervals, push it through an archaic data integration process, and then wait for the analytics results to be forwarded back to the devices along the edge. The value of collected and forwarded data decays over time, and the delays inherent in the network latency may affect an enterprise's ability to take advantage of the predictive analysis within the desired time frame.

Edge data collection, processing, and analysis are necessary. Adopt an enterprise strategy for properly identifying the right places in the IoT environment for integrated stream processing that employs machine learning models to effect prediction to influence system behaviors. Situations that can influence global systemic behaviors can be seated at a centralized platform. However, decision making affecting local operations can benefit from collecting data from the edge devices and handing that data off to a localized stream processing engine in real time.

For example, lightweight edge agents such as MiNiFi (a subproject of Apache NiFi) can be embedded at the edge devices to collect data from the source devices and hand that data to edge nodes that can process the data, monitor for any triggering events, and extract key pieces of information and then route that information to a gateway or a cloud analytics solution. By pushing the analytical stream processing to places in the environment where the consumed data originates, the predictive or prescriptive analyses can be leveraged in real time without being impacted by the latency delays of the round trip to and from a centralized server.

3 SIMPLIFY AND SCALE UP DATA INGESTION

Streaming IoT data reflects the core nature of big data: massive volumes of high-variety data being streamed at very high velocities. These three Vs pose a fundamental challenge for ingesting, processing, and analyzing IoT data, requiring a big data environment encompassing high-performance computing capabilities, distributed file systems, components for ingesting static and dynamic streaming data, data transformation tools, and analytics libraries.

It is naïve to presume that a development team will be agile enough to not only rapidly assemble processes for ingesting and onboarding different streaming IoT data sources but also continuously maintain those processes, especially as the number of sources increases. To take advantage of these big data environments, it is critical to simplify the processes by which data sets and data streams are ingested, filtered (if necessary), persisted, and forwarded to downstream algorithms and tools to create those event-processing models for streaming data that will be deployed across the network.

Look for products that provide a low-code (or better yet no-code) data ingestion engine (e.g., Apache NiFi) that can connect with a wide range of streaming data sources as well as enterprise sources for data enrichment. The availability of prebuilt processors/connectors should help simplify the development team's work building and managing data pipelines, enabling and deploying data pipelines, and orchestrating how simultaneous data pipelines support data consumer requirements. These product features help accelerate development of IoT data ingestion as well as automatically scale. Because IoT data streams often rapidly grow in terms of data volume and velocity, a framework for ingesting and processing incoming data in real time must be able to scale automatically.

It is important to emphasize scale—IoT implementations can be complicated by high data volumes and the speed at which they are ingested into the enterprise. Fast, scalable ingestion is needed to get the real-time streams to the analytics engine ASAP. A common model to support high-volume data ingestion to connect with real-time analytics engines is to leverage a streams messaging solution such as Apache Kafka. This enables event-driven architectures to accommodate applications of varying speeds and latencies to work together seamlessly in a streaming architecture.

4 LEVERAGE INTELLIGENT STREAM PROCESSING

Any IoT environment is going to involve many types of devices, embedded sensors, and computing nodes, all sharing one feature in common—the continuous generation and communication of data. Analyzing data ingested from multiple, simultaneous data streams not only allows you to develop a holistic perspective of what is happening across the environment, it also contributes to a richer analysis that can improve the precision of analytics models to be directly embedded within the network.

There are two aspects of stream processing that are integral to the success of an IoT application. First, an IoT analytics application must be able to handle the ingestion and capture of many streams of fast-moving data. Second, the resulting analytics models must be deployed across multiple tiers in the IoT network to continuously monitor the data streams, supporting automated decision making and generating notifications that drive actions to improve the business.

Data streams in an IoT network are more than just sources of information conveyed to a centralized server—they fuel the continuous analyses at different locations across the grid of devices. Yet there is fluidity among the participants in an IoT network—new devices join while others are disconnected, along with intermittent and unexpected changes in interface specifications. Therefore, data stream handling requires a greater level of sophistication and intelligence to ensure coherence, synchronization, validation, and integration.

Above and beyond the mechanics of data stream ingestion, consider these aspects of intelligent stream processing:

- **PROFILING.** This includes data value frequency analysis and interpolation of formats and structure that can be validated against documentation (if any exists)
- **SEMANTIC METADATA.** Augment the structural metadata with data element definitions and contexts
- **CURATION.** Data curation is the process of assembling, organizing, managing, and ensuring the usability of a collection of data streams

Intelligent stream processing blends technologies for streaming (such as Apache Flink), developing and mapping data flows (such as Apache NiFi), and scalable parallel processing (such as Spark) with data discovery and profiling tools.

5 ENSURE DATA QUALITY AND PROTECTION WITH UNIFIED SECURITY, LINEAGE, AND GOVERNANCE

Ensuring data quality, protecting sensitive data, and continuously monitoring compliance with an exploding array of data privacy laws and regulations require systemic data intelligence. Institute data governance to ensure acceptable integrity of data transmitted across all stages in the network. Combining structural and object metadata with data lineage, an organization can address handling of sensitive information such as PII that needs to be masked or encrypted.

Data protection is typically enforced using perimeter security and firewalls, data policies for protecting sensitive personal data or confidential intellectual property data, obfuscation methods such as encryption or data masking, and role-based access controls. The scale and breadth of IoT across a massive number of bidirectional, point-to-point data flows, however, makes extending the data protection perimeter across the massive number of nodes particularly complex. In addition, realize that the sensors and devices in the network may operate in imperfect environments, leading to sensor data that is incomplete or flawed.

In the context of IoT, data security is critical. Institute centralized management for unified governance and protection of your IoT data in a way that offers network-wide enforcement of security and governance policies. Use technologies such as Apache Ranger, Apache Atlas, NiFi, or MiNiFi to orchestrate, manage, and validate data across the full IoT data pipelines or even push the security and protection to the edges, nodes, and endpoints in the network.

Use data lineage tracking and data provenance methods and tools to provide insight about where a piece of data came from, who touched it, what changes were made to it, and its final destinations. Does this sound simple? Maybe, but in practice it requires discipline to document lineage data and keep it fresh due to the dynamic quality and volume of data streams. Many streams will ultimately converge at a centralized location for analysis, but it would be difficult to constantly manually inspect the data to ensure its validity, especially if transformations have been applied at different stages within the network. Capturing data lineage, provenance, and details about data flows in your metadata catalog lets you quickly trace emergent anomalies to their sources for assessment and remediation.

6 EMPLOY EMBEDDED STREAMING ANALYTICS ACROSS SLIDING WINDOWS

One beautiful aspect of an IoT system is how it unites a variety of always-on devices and integrates intelligent, continuously monitoring event-processing agents that trigger appropriate actions to optimize business process outcomes. Although developing embedded analytics that monitors continuous data streams is complex enough, typical approaches sometimes neglect one of the most critical (yet obvious) aspects: overseeing the status of the systemic IoT infrastructure. This comprises the persistent information about the collection of connected device states, especially in terms of internal variables or reported environmental measurements (such as a device's temperature, ambient air quality, humidity, etc.). Integrated and embedded analytics augment event-stream processing by monitoring the collection of IoT data streams for patterns of changes that should trigger actions.

Embedded analytics accounts for both the immediate snapshot of the streaming data as well as how the collective state of the environment has changed within a defined recent time frame. For example, detecting a rise in temperature along one manufacturing assembly line might generate an alert to check that line's status, but simultaneous temperature increases across multiple lines in the same building over a five-minute span might generate an alarm that the building is on fire.

These real-time analytics environments track time-series data, maintain history, and adapt according to two different duration windows. Streamed messages within a *tumbling window* are blocked by a time frame, such as all device readings within a 10-second interval. Any message will be assigned to a single tumbling window. In a *sliding window*, the time frame of a specified duration slides across the sequence of messages. A sliding window of 15 seconds with a sliding interval every three seconds allows sets of messages to be evaluated every three seconds but also allows messages to belong to more than one sliding window.

Sliding windows allow a more consistent way to continuously monitor infrastructure status. Look for technologies that optimize the incorporation of sliding windows into your event-stream processing by caching or staging streamed data in memory yet do not introduce artificial delays into the data flow.

7 MAKE THE EDGE INTELLIGENT AND SELF-RELIANT

The deployment of, and reliance on, autonomous devices (including vehicles, machines, or the many devices in factories) as well as self-assessing, self-healing devices is on the rise. A naïve vision of an IoT environment presumes a massive field of devices streaming data to a centralized platform that ingests and analyzes data to create analytics models. Although centralized monitoring and management may be important, enterprises are increasingly adopting use cases where autonomy of the edge is more relevant.

Attempting to centralize the processes for analyzing data and deploying embedded analytics models is unwarranted. Aside from the resource costs and the complexity of remote monitoring, the massive number of streams, high traffic, and the network latency do not allow for analytics to generate notifications that can be pushed back to the edge at the scale necessary to provide real-time decision making.

Realize that the data from each of the devices must stream through the computing nodes that form the edge of the IoT network on the way to the central destination, and in many cases the models used to monitor and regulate the devices in an environment are best deployed in proximity to those devices. For example, although models for monitoring energy consumption might be centrally created, based on aggregated data streaming from all the devices in a collection of smart buildings, decisions about device power regulation in each building really only depend on the data streams generated by the devices in that specific building.

Push your machine learning models directly to the edge by directly integrating the models at the edge nodes of the IoT environment. Make the edge more intelligent so that decisions can be made right at the edge itself. Because the edge nodes are much closer to the collection of devices they govern, pushing your analytics models to these edge computers reduces implementation complexity, improves system performance, and, most important, speeds the time for taking actions.

8 FINE-TUNE YOUR ANALYTICS ITERATIVELY

Every IoT initiative begins with a business purpose, and the analytics generated from such IoT implementations need to match up to the continuously evolving needs of the business initiative. Analytics models are integrated across different sites within the IoT network (particularly at the edge) to analyze data streams and generate alerts about emerging risks (such as imminent part failure) or identify emerging opportunities for improvement (such as increasing production at particular factories to meet growing product demand).

However, how can you tell if your models are working to achieve the desired business objectives? Aside from an ongoing review of the degree to which insights from IoT analytics are providing true business impact, here are some practical steps that can be used to iteratively fine-tune your analytics.

- **REVIEW MODEL PERFORMANCE.** Your analytics methodology should analyze the collected data streams and produce multiple models that presumably improve decision making at the appropriate locations in the IoT network. Introduce A/B testing by embedding different produced models, collecting data about the recommendations, and comparing the outcomes.

- **FINE-TUNE DATA COLLECTION.** Consider whether your models can be improved by adding new variables.

- **REVIEW YOUR ANALYTICS CYCLES.** Assess the frequency and volume of data collection—does your process need a greater number of instances to refine the model?

- **MODEL REFRESH.** How frequently are you reviewing the outcomes and looking to improve operational performance? Consider introducing a cadence for model refresh and refinement—choose the models that have the most desirable outcomes and see if they can be incrementally improved.

- **IDENTIFY THE BEST NETWORK LOCATION.** It is increasingly desirable to push automated decision making to the right locations in the IoT network where it can provide the optimal results. See what models can be pushed to the edges to provide high-quality real-time results.

Continually reviewing the quality of the outcomes associated with the deployment of embedded and integrated analytics allows you to iteratively fine-tune the system to achieve the right kind of results that meet your business purposes.

AFTERWORD

This checklist has reviewed a number of key suggestions for evolving the data architecture for an IoT environment. Visualize the intent of the IoT network and use tools that help continuously refine the network's topology and map out the data flows. Develop a metadata and services catalog that logs information to simplify device integration. These ideas, as well as versioning of data schemas and services, together support an agile approach to determine where and how to process the data streams and rapidly support ingesting data streaming from and back to the dynamic collection of devices. Institute data governance practices that allow consistent enforcement of data policies for data quality, timeliness, security, and protection at all points in the network—from the endpoints through the edge nodes, all the way to the centralized system.

We recommend that IoT data architecture embrace the types of technologies that support these suggestions, specifically a data services catalog, data source integration, integrated security controls, data governance/stewardship, and data life cycle management. This will support the ingestion, processing, and analysis of massive numbers of data streams, resulting in analytics models that guide profitable actions with automated decision making. Incorporating continuous performance monitoring will allow your analysts to improve their machine learning algorithms and provide a positive feedback loop to refine and improve those models.

ABOUT OUR SPONSOR

CLOUDERA

cloudera.com

At Cloudera, we believe that data can make what is impossible today possible tomorrow. We empower people to transform complex data into clear and actionable insights. Cloudera delivers an enterprise data cloud for any data, anywhere, from the edge to AI. Powered by the relentless innovation of the open source community, Cloudera advances digital transformation for the world's largest enterprises. Learn more at cloudera.com.

ABOUT TDWI RESEARCH

TDWI Research provides industry-leading research and advice for data and analytics professionals worldwide. TDWI Research focuses on modern data management, analytics, and data science approaches and teams up with industry thought leaders and practitioners to deliver both broad and deep understanding of business and technical challenges surrounding the deployment and use of data and analytics. TDWI Research offers in-depth research reports, commentary, assessments, inquiry services, and topical conferences as well as strategic planning services to user and vendor organizations.

ABOUT THE AUTHOR

David Loshin, president of Knowledge Integrity, Inc., (www.knowledge-integrity.com), is a recognized thought leader and expert consultant in the areas of data management and business intelligence. David is a prolific author regarding business intelligence best practices, as the author of numerous books and papers on data management, including *Big Data Analytics: From Strategic Planning to Enterprise Integration with Tools, Techniques, NoSQL, and Graph* and *The Practitioner's Guide to Data Quality Improvement*, with additional content provided at www.dataqualitybook.com. David is a frequent invited speaker at conferences, web seminars, and sponsored web sites and channels including TechTarget and The Bloor Group. David is also the program director for the Master of Information Management program at the University of Maryland's College of Information Studies.

David can be reached at loshin@knowledge-integrity.com.

ABOUT TDWI CHECKLIST REPORTS

TDWI Checklist Reports provide an overview of success factors for a specific project in business intelligence, data warehousing, analytics, or a related data management discipline. Companies may use this overview to get organized before beginning a project or to identify goals and areas of improvement for current projects.