REAL-TIME DIGITAL TWINS:

The Next Generation in Streaming Analytics

Better Answers - Faster

ScaleOut Software

CONTENTS

INTRODUCTION

This is the second in a series of ebooks describing a breakthrough new approach to streaming analytics called real-time digital twins. The first ebook, “Enhancing Intelligent Real-Time Monitoring with Streaming Analytics Using Real-Time Digital Twins” introduced this powerful new software technique for boosting situational awareness in live systems and explained how in-memory computing technology ensures fast, scalable performance. In this ebook, we will compare the use of real-time digital twins to conventional streaming analytics, describe its many advantages, and explore applications that can benefit from its unique capabilities. Future ebooks will look at applications for real-time digital twins in depth and give a tour of the software development process.

We invite you to visit scaleoutsoftware.com to learn more about the ScaleOut Digital Twin Streaming Service™, which provides a powerful software platform for hosting real-time digital twins and simultaneously analyzing telemetry messages from thousands of data sources. Get ready for the next generation in streaming analytics that delivers better answers — faster.

CHAPTER 1

Real-Time Digital Twins: A New Approach to Streaming Analytics

The goal of real-time streaming analytics is to get answers fast. Mission-critical applications that manage large numbers of live data sources need to quickly sift through incoming telemetry, assess dynamic changes, and immediately pinpoint emerging issues that need attention. Examples abound: a telematics application tracking a fleet of vehicles, a vaccine distribution system managing the delivery of thousands of shipments, a security or safety application analyzing entry points in a large infrastructure (physical or cyber), a healthcare application tracking medical telemetry from a population of wearable devices, a financial services application watching wire transfers and looking for potential fraud — the list goes on. In all these cases, when a problem occurs (or an opportunity emerges), managers need answers now.

Conventional streaming analytics platforms are unable to separate messages from each data source and analyze them as they flow in. Instead, they ingest and store telemetry from all data sources, attempt a preliminary search for interesting patterns in the aggregated data stream, and defer detailed analysis to offline batch processing, which may take minutes or hours to complete. As a result, they are unable to introspect on the dynamic, evolving state of each data source and immediately alert on emerging issues, such as the impending failure of a truck engine, an unusual pattern of entries and exits to a secure building, or a potentially dangerous pattern of telemetry for a patient with a known medical condition.

In-memory computing with software components called real-time digital twins overcomes these obstacles and enables continuous analysis of incoming telemetry for each data source with contextual information that deepens introspection. While processing each message in a few milliseconds, this technology automatically scales to simultaneously handle thousands of data sources. It also can aggregate and visualize the results of analysis every few seconds so that managers can graphically track the state of a complex live system and quickly pinpoint issues.

Real-time digital twins leverage in-memory computing technology to turn the conventional Lambda model for streaming analytics on its head and enable each data source to be independently tracked and responded to in real time. A real-time digital twin is a software object that encapsulates dynamic state information for each data source combined with application-specific code for processing incoming messages from that data source. This state information gives the code the context it needs to assess the incoming telemetry and generate useful feedback within 1-3 milliseconds.

For example, suppose an application analyzes heart rate, blood pressure, oxygen saturation, and other telemetry from thousands of people wearing smart watches or medical devices. By holding information about each user's demographics, medical history, medications, detected anomalies, and current activity, real-time digital twins can intelligently assess this telemetry while updating their state information to further refine their feedback to each person. Beyond just helping real-time digital twins respond more effectively in the moment, maintaining context improves feedback over time.

Figure 1 illustrates the use of an in-memory computing platform hosting real-time digital twins for multiple data sources. Each incoming message is analyzed within a few milliseconds, and the real-time digital twins create state information that is aggregated every 5-10 seconds for visualization and query.

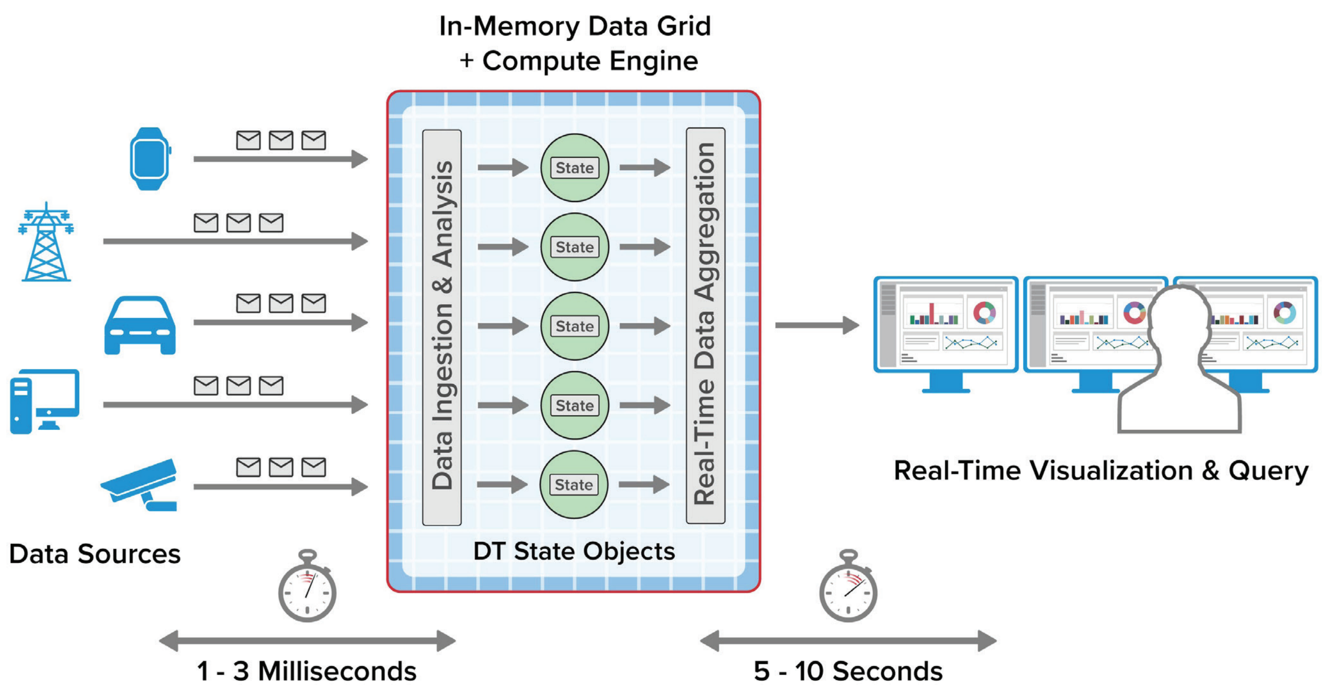

Figure 1: An in-memory computing platform hosting real-time digital twins

The ScaleOut Digital Twin Streaming Service is an Azure-based cloud service that uses real-time digital twins to perform continuous data ingestion, analysis by data source, aggregation, and visualization, as illustrated above. What's key about the use of real time digital twins is that they enable the system to visualize state information that results from real-time analysis — not raw telemetry flowing in from data sources. This gives managers curated data that intelligently focuses on the key problem areas (or opportunities). For example, instead of looking at fluctuating oil temperature, telematics dispatchers see the results of predictive analytics for each truck engine. There's not enough time for managers to examine all the raw data, and not enough time to wait for batch processing to complete. Maintaining situational awareness requires real-time introspection for each data source, and real-time digital twins provide it.

In the ScaleOut Digital Twin Streaming Service, real-time data visualization can take the form of charts, tables, or maps. Dynamic charts effectively display the results of aggregate analytics that combine data from all real-time digital twins to show emerging patterns, such as the regions of the country with the largest delivery delays for a vaccine distribution system. This gives a comprehensive view that helps managers maintain the "big picture." To pinpoint precisely which data sources need attention, users can query analytics results for all real-time digital twins and see the results in a table or map. For example, managers can ask questions like "Which vaccination centers in Washington state are experiencing delivery delays in excess of 1 hour and have seen more than 100 people awaiting vaccinations at least three times today?" With this information, they can immediately determine where vaccine shipments should be delivered first.

Real-time digital twins create exciting new capabilities that were not previously possible with conventional techniques. You can find detailed information about ScaleOut Software's cloud service for real-time digital twins at scaleoutsoftware.com.

CHAPTER 2

Advantages Over Conventional Streaming Analytics

To understand the advantages of real-time digital twins for streaming analytics, we first need to look at what problems are tackled by popular streaming platforms and how these platforms analyze data. Then we can see how real-time digital twins take a different approach that creates a breakthrough in situational awareness.

Conventional Techniques for Streaming Analytics

Most if not all platforms focus on mining the data within an aggregated message stream for patterns of interest. For example, consider a web-based ad-serving platform that selects ads for users and logs messages containing a timestamp, user ID, and ad ID every time an ad is displayed. To give a running indication of which ads are trending, a streaming analytics platform might count all the ads for each unique ad ID in the latest five-minute window and repeat this every minute.

Figure 2 illustrates a typical streaming analytics pipeline available on popular cloud platforms, which could be used to analyze telemetry from applications like the ad-serving example above. This pipeline combines telemetry from all data sources into a single stream which is queried by the user's streaming analytics application. Analytics code often takes the form of a set of SQL queries (extended with time-windowing semantics) running continuously to select interesting events from the stream. Query results are then forwarded to a data lake for offline analytics using tools such as Spark and for data visualization. Query results also might be forwarded to cloud-based serverless functions to trigger alerts or other actions in conjunction with access to a database or blob store.

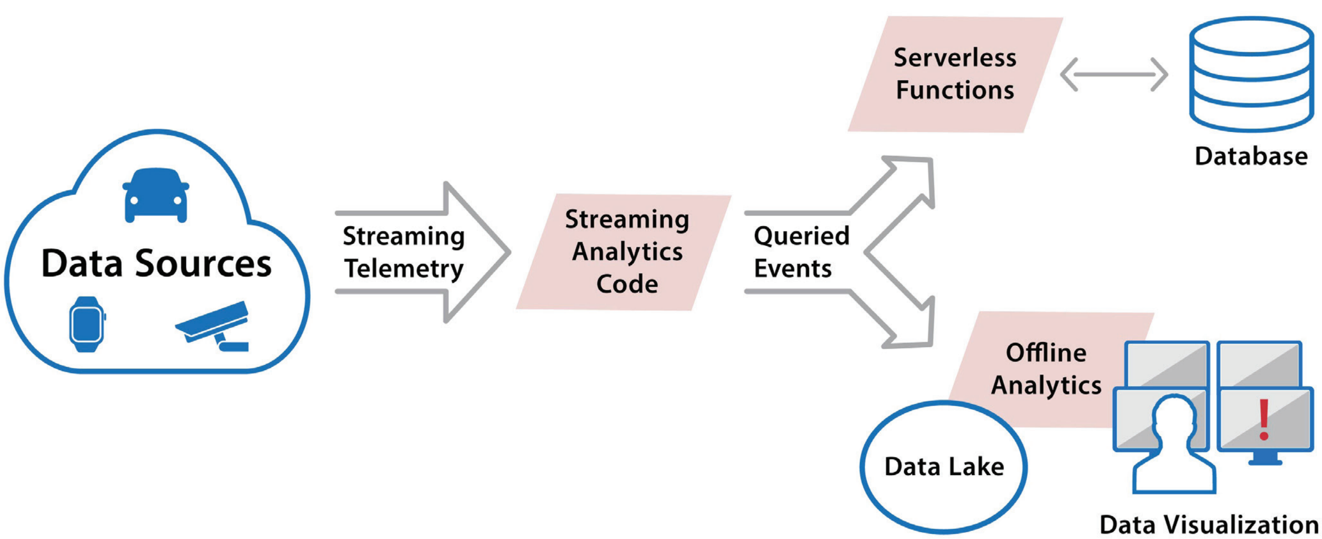

Figure 2: A conventional streaming analytics pipeline

Based on technology from the Trill research project, the Microsoft Stream Analytics platform offers an elegant and powerful platform for implementing applications like this. It views the incoming stream of messages as a columnar database with the column representing a time-ordered history of messages. It then lets users create SQL-like queries with extensions for time-windowing to perform data selection and aggregation within a time window, and it runs at high speed to keep up with incoming data streams.

Other streaming analytic platforms, such as open-source Apache Storm, Flink, and Beam and commercial implementations such as Hazelcast Jet, let applications pass an incoming data stream through a pipeline (or directed graph) of processing steps to extract information of interest, aggregate it over time windows, and create alerts when specific conditions are met. For example, execution pipelines could process a stock market feed to compute the average stock price for all equities over the previous hour and trigger an alert if an equity moves up or down by a certain percentage. Another application tracking telemetry from gas meters could likewise trigger an alert if any meter's flow rate deviates from its expected value, which might indicate a leak.

These techniques are highly effective for analyzing aggregated raw telemetry to identify unusual situations which might require action. For example, if the telemetry is tracking a fleet of rental cars, a query could report all cars by make and model that have reported a mechanical problem more than once over the last 24 hours so that follow-up inquiries can be made. In another application, if the telemetry stream contains key-clicks from an e-commerce clothing site, a query might count how many times a garment of a given type or brand was viewed in the last hour so that a flash sale can be started.

However, other than by observing data in the stream, these stream-processing techniques do not track the dynamic state of the data sources themselves, and they don't make inferences about the behavior of the data sources, either individually or in aggregate. For example, the streaming analytics platform for the ad server doesn't know why each user was served certain ads, and the market-tracking application does not know why each equity either maintained its stock price or deviated materially from it. Without knowing the why, it's much harder to take the most effective action when an interesting situation develops.

Another limitation of this approach is that it is difficult to separately track and analyze the behavior of each individual data source, especially when they number in the thousands or more. It's simply not practical to create a unique query tailored for each data source. Fine-grained analysis by data source must be relegated to offline processing in the data lake, making it impossible to craft individualized, real-time responses to the data sources.

For example, a rental car company might want to alert a driver if she/he strays from an allowed region or appears to be lost or repeatedly speeding. An e-commerce company might want to offer a shopper a specific product based on analyzing the clickstream in real time with knowledge of the shopper's brand preferences and demographics. These individualized actions are impractical using the conventional tools of streaming analytics.

Advantages of Real-Time Digital Twins

Real-time digital twins easily bring these capabilities within reach. Take a look at how the streaming pipeline differs when using real-time digital twins:

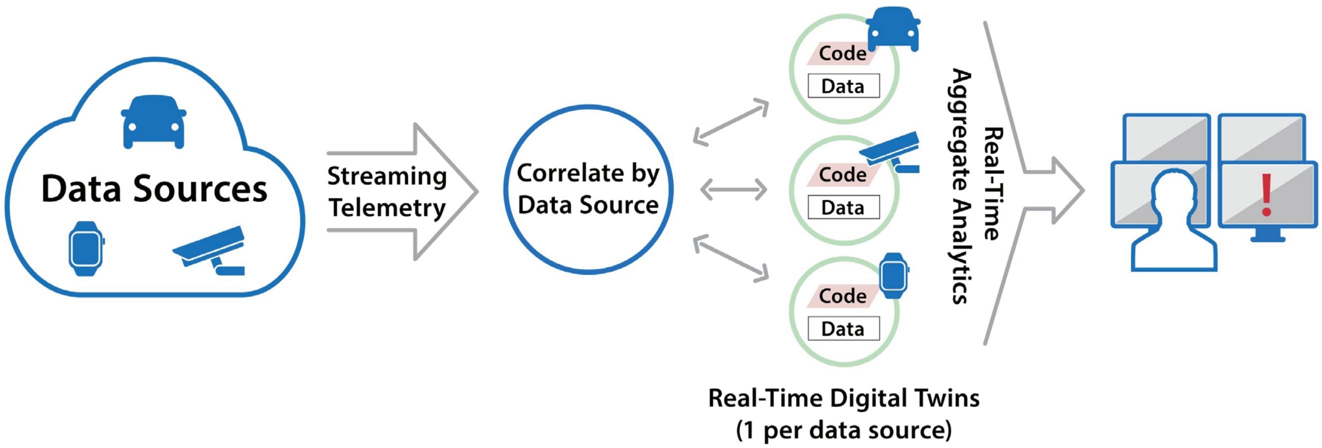

Figure 3: A streaming analytics pipeline using real-time digital twins

The first important difference to note is that the execution platform automatically correlates telemetry events by data source. This avoids the need for the application to select events by data source using queries (which is impractical in any case when there are many data sources). The second difference is that real-time digital twins maintain immediately accessible, in-memory state information for each data source which is used by message-processing code to help analyze incoming messages from that data source. State information enables application code to more fully analyze telemetry information with contextual knowledge about the history and state of each data source.

For example, the rental car application can keep each driver's contract, location history, and the car's known mechanical issues and service history within its corresponding digital twin for immediate reference to help detect whether an alert is needed. Likewise, the e-commerce application can keep each shopper's recent product searches along with brand preferences and demographics in her/his digital twin, enabling timely suggestions targeted to each shopper.

The power of real-time digital twins lies in their ability to provide high quality, fine-grained analysis and immediate responses for thousands of data sources. This software technique is made possible by scalable, in-memory computing technology hosted on clusters of cloud-based servers, which ensures the fast response times and scalable throughput needed to support many thousands of data sources.

Real-time digital twins shift the application's focus from the incoming raw telemetry stream to the dynamically evolving state of the data sources. Applications can incorporate dynamic information about each data source in their analysis of incoming messages, and they can update this state over time. The net effect is that applications can develop a significantly deeper understanding about the data source's dynamic behavior so that effective action can be taken when needed. This cannot be achieved by just looking at data within the incoming telemetry stream.

For example, the ad-serving application can use a real-time digital twin for each user to track shopping history and preferences, measure the effectiveness of ads, and guide ad selection. The stock market application can use a real-time digital twin for each company to track financial information, executive changes, and news releases that explain why its stock price starts moving and filter out trades that don't fit appropriate criteria.

Real-time digital twins also open the door to real-time aggregate analytics that analyze state information across all instances to spot emerging patterns and trends. Instead of waiting for the data lake to provide insights, aggregate analytics on state information held in real-time digital twins can immediately surface patterns of interest, maximizing situational awareness and assisting in the creation of response strategies.

With aggregate analytics, the rental car company can identify regions with unusual delays due to weather or highway blockages and then alert the appropriate drivers to suggest alternative routes. The e-commerce company can spot hot-selling products resulting from social media campaigns and respond to ensure that inventory is made available.

An Example

Consider a trucking fleet that manages thousands of long-haul trucks on routes throughout the U.S. Each truck periodically sends telemetry messages about its location, speed, engine parameters, and cargo status (for example, trailer temperature) to a real-time monitoring application at a central location. With traditional streaming analytics, personnel can detect changes in these parameters, but they can't assess their significance and take effective, individualized action for each truck. Is a truck stopped because it's at a rest stop or because it has stalled? Is an out-of-spec engine parameter expected because the engine is scheduled for service, or does it indicate that a new issue is emerging? Has the driver been on the road too long? Does the driver appear to be lost or entering a potentially hazardous area?

The use of real-time digital twins provides the context needed for the application to answer these questions as it analyzes incoming messages from each truck. For example, it can keep track of the truck's route, schedule, cargo, mechanical and service history, and information about the driver. Using this information, it can alert drivers to impending problems, such as road blockages, delays or emerging mechanical issues. It can assist lost drivers, alert them to erratic driving or the need for rest stops, and help when changing conditions require route updates.

Figure 4 shows a truck communicating with its associated real-time digital twin. (The parallelogram represents application code.) Because the twin holds unique contextual data for each truck, analysis code for incoming messages can provide highly focused feedback that goes well beyond what is possible with traditional streaming analytics:

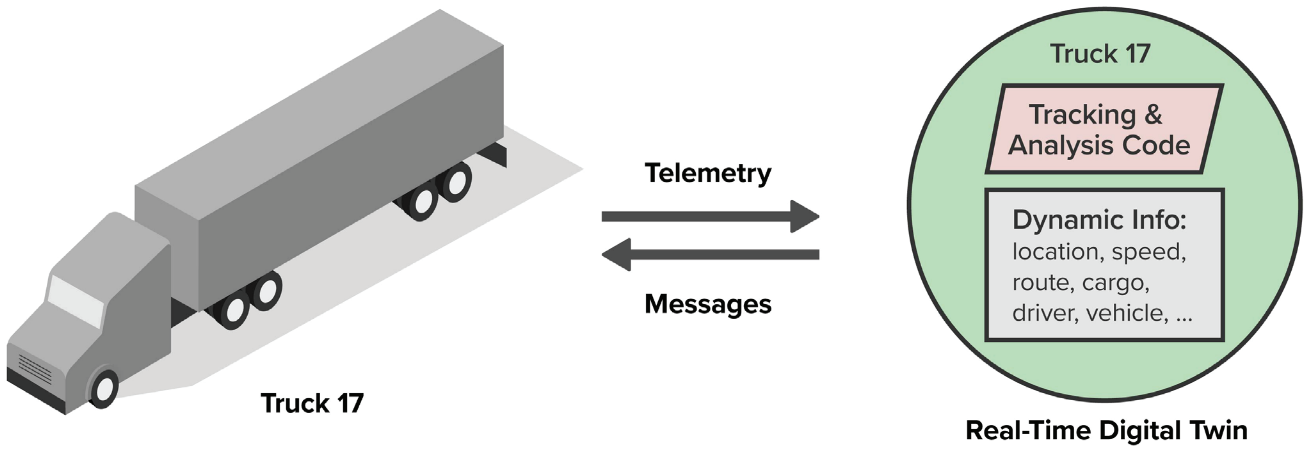

Figure 4: Example of a truck sending telemetry to its real-time digital twin

As illustrated in figure 5, the ScaleOut Digital Twin Streaming Service runs as a cloud-hosted service in the Microsoft Azure cloud to provide streaming analytics using real-time digital twins. It can exchange messages with hundreds of thousands of trucks across the U.S., maintain a real-time digital twin for each truck, and direct messages from that truck to its corresponding twin. The use of real-time digital twins simplifies application code, which only needs to process messages from a given truck and has immediate access to dynamic, contextual information that enhances the analysis. The result is better feedback to drivers and enhanced overall situational awareness for the fleet.

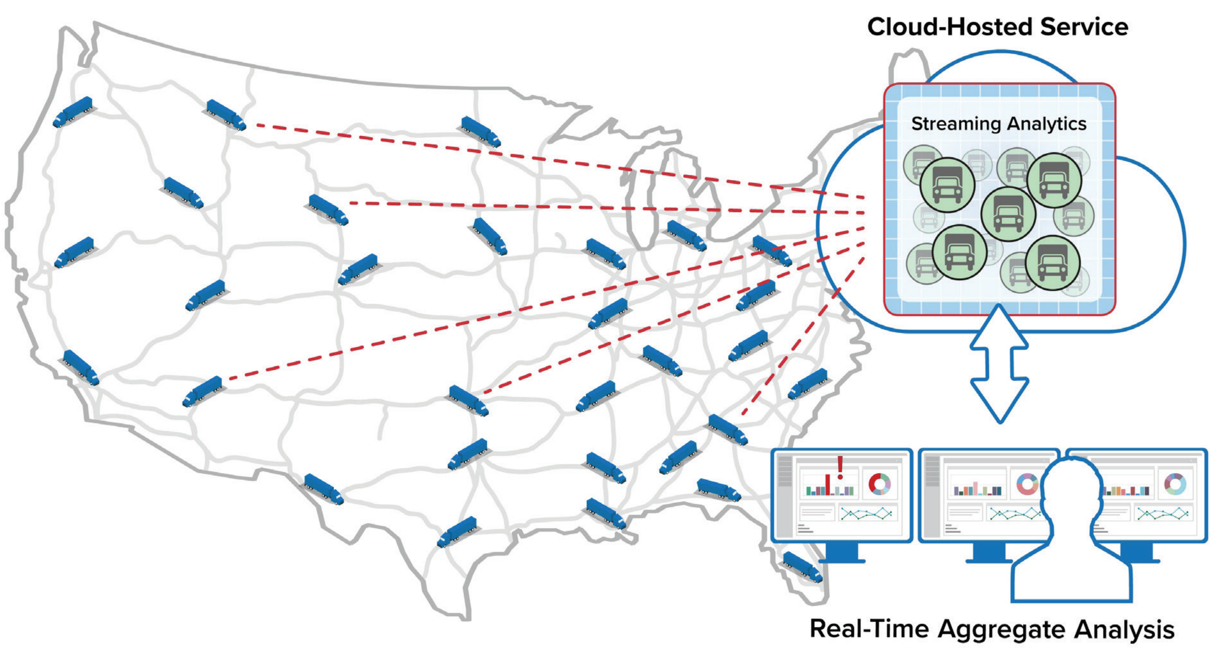

Figure 5: Example of a cloud service using real-time digital twins to manage a fleet of trucks

More Advantages: Lower Complexity and Higher Performance

While the functionality implemented by real-time digital twins could be replicated with ad hoc solutions that combine application servers, databases, offline analytics, and visualization, they would require vastly more code, a diverse combination of skill sets, and longer development cycles. They also would encounter performance bottlenecks that require careful analysis to measure and resolve. The real-time digital twin model running on an integrated in-memory computing platform sidesteps these obstacles.

Scaling performance to maintain high throughput creates an interesting challenge for traditional streaming analytics pipelines because the work performed by their tasks does not naturally map to a set of processing cores within multiple servers. Each pipeline stage must be accelerated with parallel execution, and some stages require longer processing time than others, creating bottlenecks in the pipeline.

In contrast, real-time digital twins naturally create a uniformly large set of tasks that can be evenly distributed across servers. This enables the processing of real-time digital twins to scale transparently without adding complexity to either applications or the platform.

Summing Up

Why use real-time digital twins? They solve an important challenge for streaming analytics that is not addressed by conventional, “pipeline-oriented” platforms, namely, to simultaneously track the state of thousands of data sources. They use contextual information unique to each data source to help interpret incoming messages, analyze their importance, and generate feedback and alerts tailored to that data source.

Traditional streaming analytics finds patterns and trends in the data stream. Real-time digital twins identify and react to important state changes in the data sources themselves. As a result, streaming analytics applications can achieve better situational awareness than previously possible. This new way of implementing streaming analytics creates a breakthrough for a wide range of applications.

CHAPTER 3

Aggregating State Data Created by Real-Time Digital Twins

When analyzing telemetry from a large population of data sources, such as a fleet of rental cars or IoT devices in "smart cities" deployments, it's difficult if not impossible for conventional streaming analytics platforms to track the behavior of each individual data source and derive actionable information in real time. Instead, most applications just sift through the telemetry for patterns that might indicate exceptional conditions and forward the bulk of incoming messages to a data lake for offline scrubbing with a big data tool such as Spark. How can streaming analytics maintain situational awareness without waiting for the delays inherent in big data analytics?

Using State Information for Aggregate Analytics

State information held within real-time digital twins provides a significant repository of data that can be analyzed in aggregate to immediately spot important trends. With in-memory computing, aggregate analysis can be performed continuously every few seconds instead of waiting for offline analytics in a data lake. State information maintained for each data source and updated as telemetry flows in can be extracted from all real-time digital twins and aggregated to highlight emerging patterns or issues that may need attention. This provides a powerful tool for maximizing overall situational awareness.

Consider an emergency monitoring system during the COVID-19 crisis that tracks the need for supplies across the nation's 6,100+ hospitals and attempts to quickly respond when a critical shortage emerges. Let's assume that all hospitals send messages every few minutes to this system, which runs in a central command center. These messages provide updates on various types and amounts of shortages (for example, of PPE, ventilators, and medicines) that the hospitals need to quickly rectify. Using state information, a real-time digital twin for each hospital can both track and evaluate these shortages as they evolve. It can look at key indicators, such as the relative importance of each supply type and the rate at which the shortages are increasing, to create a dynamic measure of urgency that the hospital receive attention from the command center. All of this data can be continuously updated within the real-time digital twin as messages arrive to give personnel the latest status.

Aggregate analysis can then compare this data across all hospitals by region to identify which regions have the greatest immediate need and track how fast and where overall needs are evolving. Personnel can query state information within the real-time digital twins to quickly determine which specific hospitals should receive supplies and what specific supplies should be immediately delivered to them. Using real-time digital twins, all of this can be accomplished in seconds or minutes, as illustrated in figure 6.

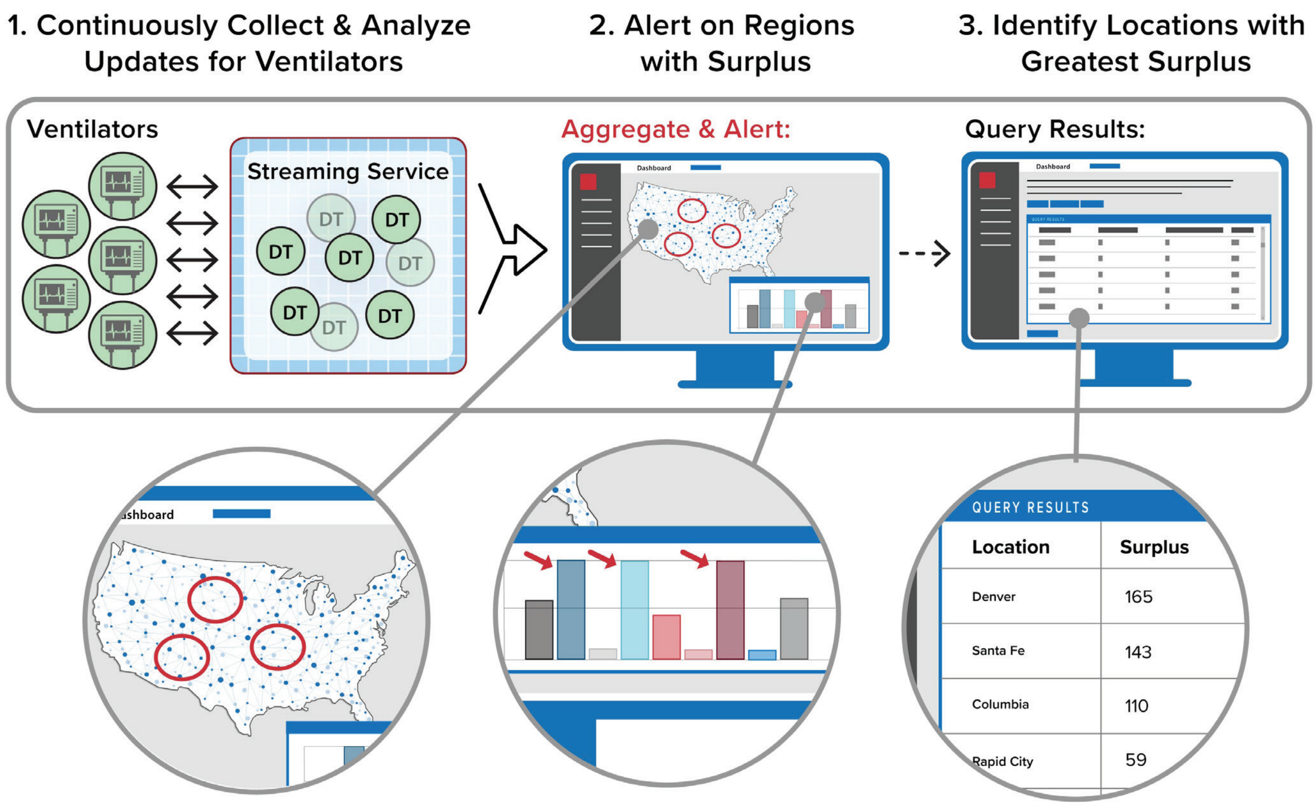

Figure 6: Aggregate analysis of state information in real-time digital twins to identify supply shortfalls

As this example shows, real-time digital twins provide both a real-time filter and aggregator of the data stream from each data source to create dynamic information that is continuously extracted for aggregate analysis. Real-time digital twins also track detailed information about the data source that can be queried to provide a complete understanding of evolving conditions and enable appropriate action.

Numerous Applications Need Real-Time Monitoring

This new paradigm for streaming analytics can be applied to numerous applications. For example, it can be used in security applications to assess and filter incoming telemetry (such as likely false positives) from intrusion sensors and create an overall likelihood of a genuine threat from a given location within a large physical or cyber system. Aggregate analysis combined with queries can quickly evaluate the overall threat profile, pinpoint the source(s), and track how the threat is changing over time. This information enables personnel to assess the strategic nature of the threat and take the most effective action.

Likewise, disaster-recovery applications can use real-time digital twins to track assets needed to respond to emergencies, such as hurricanes and forest fires. Fleets of rental cars or trucks can use real-time digital twins to track vehicles and quickly identify issues, such as lost drivers or breakdowns. IoT applications can use real-time digital twins to implement predictive analytics for mission-critical devices, such as medical refrigerators.

Summing Up: Do More in Real Time

Conventional streaming analytics only attempt to perform superficial analysis of aggregated data streams and defer the bulk of analysis to offline processing. Because of their ability to maintain dynamic, application-specific information about each data source, real-time digital twins offer breathtaking new capabilities to track thousands of data sources in real time, provide intelligent feedback, and combine this with immediate, highly focused aggregate analysis. By harnessing the scalable power of in-memory computing, real-time digital twins are poised to usher in a new era in streaming analytics.

CHAPTER 4

Geospatial Mapping for Real-Time Digital Twins

Geospatial mapping of queried state information held in real-time digital twins combined with continuous queries that refresh the map every few seconds can significantly boost situational awareness in streaming analytics applications. For example, figure 7 shows a telematics system for a trucking fleet displaying the locations of specific trucks which have issues (the red dots on the map) in addition to charting aggregate statistics:

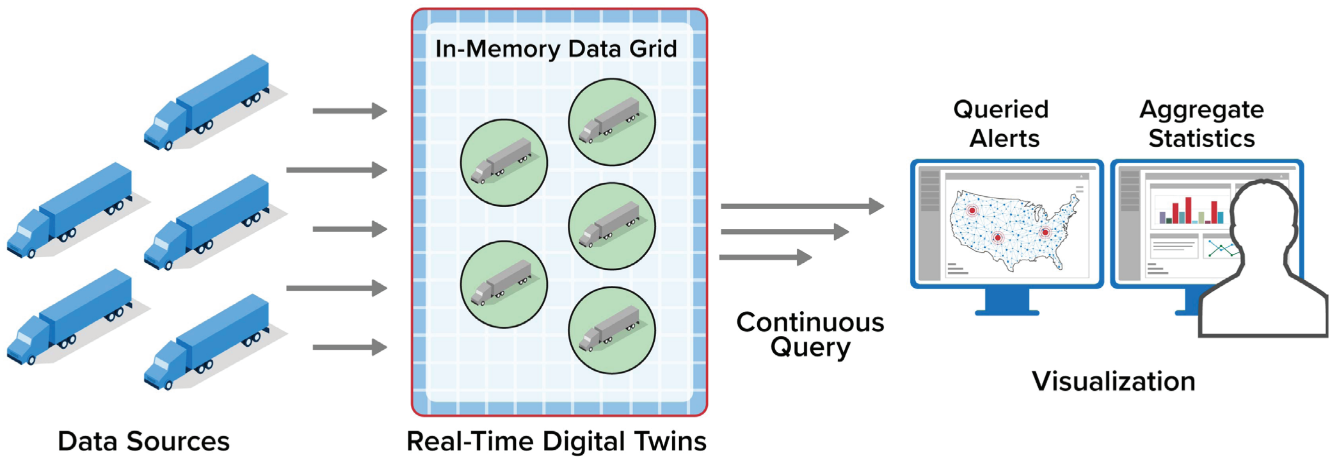

Figure 7: Example of geospatial mapping for continuous queries of real-time digital twins

For applications like this, a mapped view of query results offers valuable insights about the locations where issues are emerging that would otherwise be more difficult to obtain from a tabular view.

As illustrated in figure 8, geospatial mapping shows the results of real-time analytics, not raw telemetry, and these results are continuously updated as incoming messages are processed. For example, instead of displaying the latest oil temperature from a truck, the query reports the results of a predictive analytics algorithm that makes use of several state variables maintained by the real-time digital twin. This declutters the dispatcher’s view so that only alertable conditions are highlighted and demand attention.

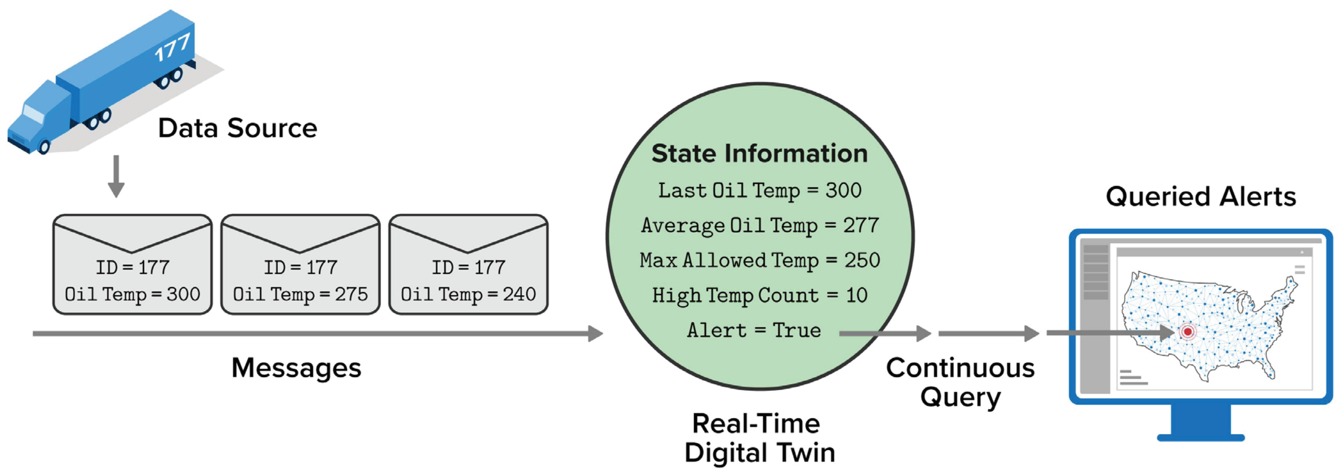

Figure 8: Example of geospatial mapping of queried state information held in a real-time digital twin

Figure 9 shows an example of actual map output for a hypothetical security application that tracks possible intrusions within a nationwide power grid. The goal of the real-time digital twins is to assess telemetry from each of 20K control points in the power grid's network, filter out false-positives and known issues, and produce a quantitative assessment of the threat ("alert level"). Continuous queries map the results of this assessment so that managers can immediately spot a real threat, understand its scope, and take action to isolate it. The map shows the results three continuous queries: high alerts requiring action, medium alerts that just need watching, and offline nodes (with the output suppressed here):

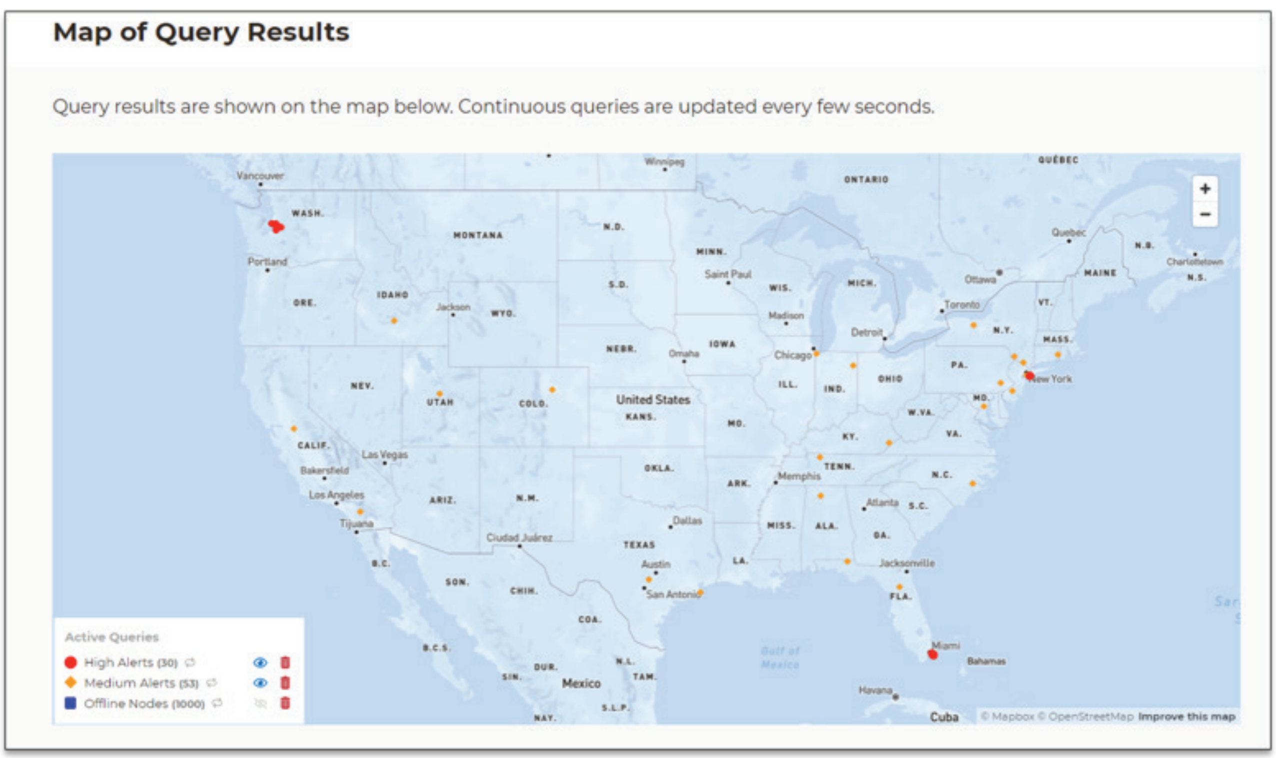

Figure 9: Geospatial mapping of alerts in a security application for a power grid

In this scenario, a high alert has suddenly appeared in the grid at three locations (Seattle, New York, and Miami) indicating a serious, coordinated attack on the network. As shown in figure 10, by zooming in and hovering over dots in the graph, users can display the detailed query results for each corresponding data source. Within seconds, managers have immediate, actionable information about threat assessments and can quickly visualize the locations and scope of specific threats.

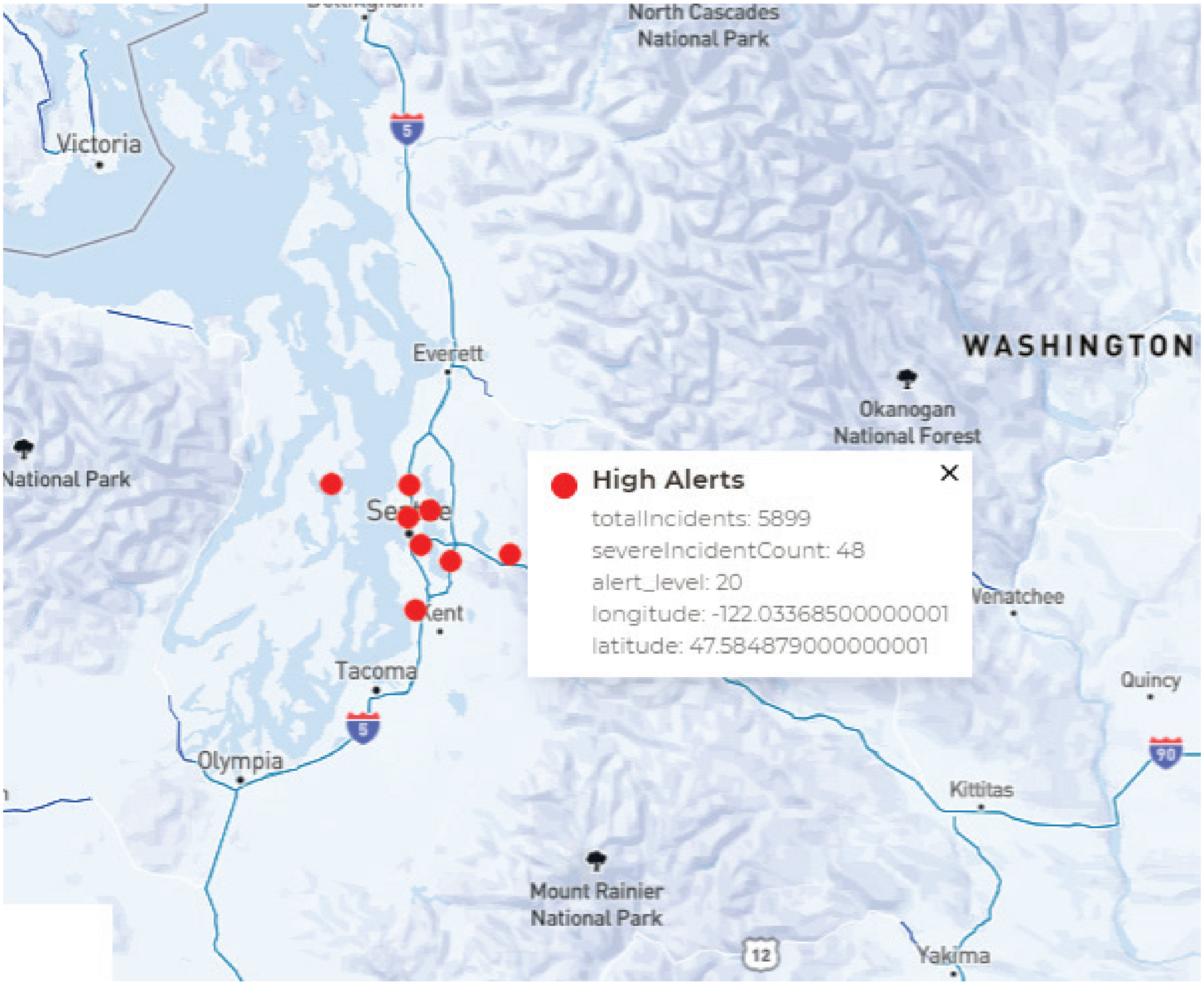

Figure 10: Examining the details of query results displayed in a map

In applications like these and many others, the power of in-memory computing with real-time digital twins gives managers a new means to digest real-time telemetry from thousands of data sources, combine it with contextual information that enhances the analysis, and then immediately visualize the results. This powerful technology boosts situational awareness and helps guide responses much better and faster than was previously possible.

CHAPTER 5

Digital Twins and Real-Time Digital Twins: What's the Difference?

Digital twins have historically been used in the field of product life-cycle management (PLM) to model the behavior of individual devices or components within a system. Digital twins assist in their design and development and help lower costs. A digital twin model of a device simulates both the device's behavior and its interactions with other components in the system. For example, it might periodically emit telemetry messages that mimic those from a real device, and it might also accept incoming commands that modify its actions. To maximize their usefulness, these models are designed to realistically emulate the behavior of their real-world counterparts. This requires that the model incorporate state variables and timing algorithms that accurately implement device behavior.

ScaleOut Software has extended the concept of digital twins beyond PLM for use in real-time streaming analytics within live, mission-critical systems. A real-time digital twin continuously ingests telemetry messages from a single device and analyzes this telemetry to produce effective, timely feedback and alerting. In order to identify emerging patterns (for example, signs of an imminent failure or the need for an alarm), it maintains dynamically evolving state information about the device, such as a record of significant abnormal events in its recent operating history, known issues, maintenance history and schedule, and the operating environment. All of this information helps put telemetry in a richer context and leads to better analysis and more effective feedback.

When viewed from the perspective of a digital twin model, a real-time digital twin does in fact implement a model of the real-world device or data source. However, its model is designed not to emulate the device's characteristics for observation by external components. Instead, it is designed to model the device's internal behavior based on updates from incoming telemetry so that predictions and alerts can be made. In short, it is designed to provide streaming analytics. As illustrated in figure 11, by encapsulating message-processing code and state information about each device, a real-time digital twin can perform this function.

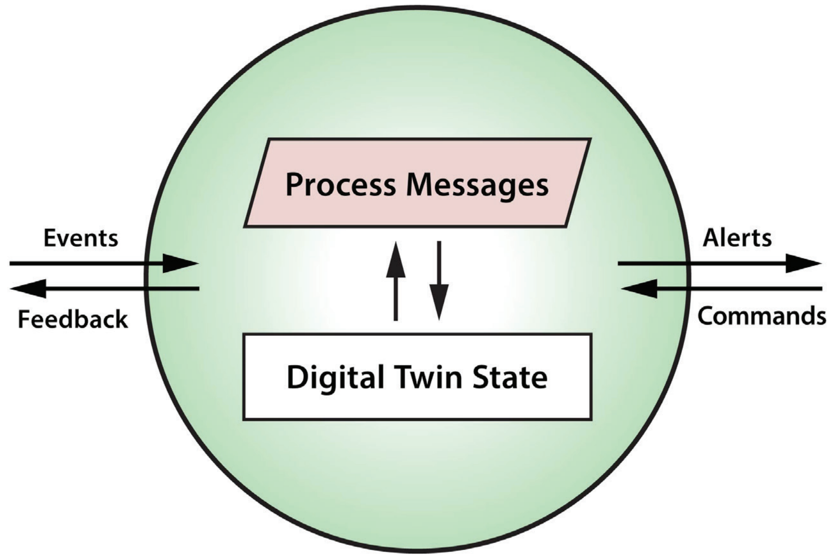

Figure 11: A real-time digital twin showing message-processing code and state information

Because real-time digital twins perform streaming analytics instead of device emulation, they can be used to track data sources far more complex than physical devices, such as generators or engines. For example, they can be used to track the online shopping behavior of ecommerce shoppers to assist in making product recommendations. Their streaming analytics function only needs to track those aspects of the data source which are relevant to the required feedback and alerting. For ecommerce, this might include tracking brand preferences, desired price ranges, and recently viewed product features. While real-time digital twins play a more restrictive role than PLM digital twins, they can be applied in a wide range of applications tracking highly complex data sources.

The key differences between digital twins and real-time digital twins can be summarized as follows:

Characteristic	Digital Twin	Real-Time Digital Twin
primarily used in	product development and simulation	live systems for situational awareness
behavior triggered by	timing the device's actions	incoming telemetry messages
state variables	implement the device's behavior	provide context for streaming analytics
messages	are emitted to model device's telemetry	are received to analyze the device's behavior

What's intriguing about these related concepts is that they can be used together within a single system for purposes of simulation. In particular, digital twin models can be used to simulate devices and generate telemetry for analysis by real-time digital twins. This enables the quality of streaming analytics within real-time digital twins to be assessed and optimized prior to their deployment within live systems. Figure 12 shows a depiction of three digital twins sending messages to their real-time digital twin counterparts:

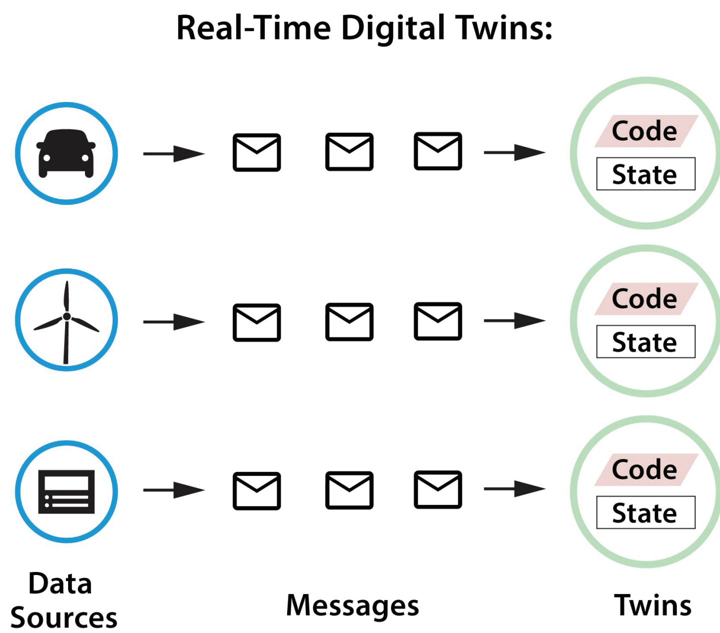

Figure 12: Combining device models with real-time digital twins in a simulation

The ScaleOut Digital Twin Streaming Service provides a powerful cloud service for hosting real-time digital twins and simultaneously analyzing telemetry messages from thousands of data sources. Its ability to automatically correlate incoming messages by data source and keep state information instantly available enables real-time digital twins to respond within a few milliseconds. Integrated, aggregate analysis of key state variables across the entire population of real-time digital twins completes within seconds and runs continuously. This gives real-time monitoring applications a powerful new tool for constantly assessing the health of thousands of devices and maximizing situational awareness.

ScaleOut Software

ScaleOut Software develops leading-edge software products for hosting real-time streaming analytics that maximize situational awareness. Its stream-processing platform and cloud service enable the creation of real-time digital twin models of IoT devices that provide immediate feedback, alerting, and integrated, data-parallel analytics. This platform transparently scales to handle millions of data sources and enables an entire population of data sources to be continuously analyzed to quickly identify and respond to broad trends, such multiple intrusions or cascading failures. Real-time digital twins are useful in a wide variety of applications, including intelligent real-time monitoring, IoT, security, safety, transportation, healthcare, and many others.

Founded in 2003, ScaleOut Software offers a wide range of in-memory computing products that help mission-critical enterprise applications store and analyze live, fast-changing data. Its product portfolio includes ScaleOut StateServer®, an award-winning, linearly scalable, and highly available in-memory data grid for distributed caching, and ScaleOut StreamServer® DT, an on-premises software platform for stateful stream-processing with integrated data-parallel computation on live data. Please visit www.scaleoutsoftware.com for more information on ScaleOut Software's technologies and products.