VANTIQ White Paper
VANTIQ.com

Distribution and Federation in Real-Time, Event-Driven Business Applications

Paul Butterworth | May 24, 2017

EXECUTIVE SUMMARY

Real-time, event-driven business applications are taking center stage as the next generation of business applications, supporting the transition of every business to become a digital business. Next generation planning, operations and customer engagement applications that provide optimal, personalized experiences all depend on real-time sensing and near real-time decision making. Such applications must be built on a modern, event-driven business application platform.

Most event-driven business applications need to be deployed in a distributed manner for improved responsiveness, robustness and security. With VANTIQ, an event-driven business application is developed in a single cloud location and then automatically *partitioned,* resulting in the components of the application being distributed to the most optimal nodes for execution whether the nodes are cloud hosted, data center hosted, intelligent devices at the edge, or a combination thereof. Logic will be located where it is the most effective. A wide range of system topologies including star, hierarchical, and peer-to-peer are supported. The provisioning and management of these networks is made automatic and easy to manage by intelligent features built into the VANTIQ platform. Application components can be dynamically changed anywhere in the distributed environment for one or tens of thousands of nodes while the system is running.

VANTIQ easily includes people, as appropriate, into the event-driven business application through direct support for collaboration between systems and humans via mobile devices, more traditional computing devices or emerging voice and video channels. (Refer to VANTIQ's Real-Time Collaborative Systems whitepaper for a deeper discussion of this topic.)

VANTIQ's design goal is to automate as much of the design, provisioning and management of real-time, event-driven business applications as possible so that the development of the systems can focus on the business logic and not the underlying infrastructure. VANTIQ's unique combination of advanced capabilities and seamless integration dramatically improves the speed and efficiency with which event-driven business applications can be constructed, deployed and operated.

Table of Contents

BACKGROUND

Real-Time, Event-Driven Business Applications

In this paper, a fairly expansive definition of a real-time, event-driven business application is used to cover the wide array of modern event-driven applications. Such applications have the general flow of:

- **Input is received** from a number of sensors perhaps over an extended period of time. Sensors may be physical sensors, data streams produced by other enterprise systems or public data streams.
- The sensor **data is analyzed** to produce the events, consisting of information and context, on which automation, recommendation and collaboration decisions are made. Additional context may be extracted from other systems to augment the sensor data.
- The **events are evaluated** in real time to determine the actions that need to be taken. In general, discrete rules and/or machine learning strategies are used to perform the real-time evaluation.
- **Actions are transmitted** to the responsible systems for implementation or human-machine collaboration is initiated with responsible personnel to determine the most appropriate response to the current situation.

In real-time, event-driven business applications, processing is typically better performed local to the device under control, improving response time and reliability. For example, in an industrial setting, managing the position of a materials handling system requires near real-time responses within a few hundred milliseconds. Such response times cannot be guaranteed by a remote decision-making system that may be delayed by thousands of milliseconds if there is a network problem.

All processing is done in a secure environment that carefully manages access to situational data and the ability to initiate control actions.

Sense — Data Acquisition

The notion of a sensor can be very broad including:

- Mobile devices hosting a wealth of sensor data including location, acceleration, audio, video and behavioral patterns derived from the raw sensor data.
- Wearable devices such as watches, activity trackers, health monitors, audio and video headsets.
- Machines including industrial machines, land and airborne transportation, home appliances and any mechanical or electronic equipment that can be sensed and/or controlled. For example, imagine a robot's manipulators instrumented with pressure sensors to vary the pressure applied to objects that may have different crush points.
- Stand-alone sensors deployed in great numbers. For example, moisture sensors distributed across the fields of a farm to minimize water consumption while maximizing growth rates for the crops.
- Video and audio feeds that produce high volumes of what can be considered sensor data. Recognition software is used to determine what the video represents to translate the video into more discrete events on which automation decisions can depend.
- Existing enterprise applications producing streams of transactions.

The sensors can be connected directly to the internet with their own IP communications stack, or they can be indirectly connected to the Internet via an edge node. In the latter case, the sensors themselves may communicate over more specialized protocols such as Modbus or ZigBee with the edge node providing protocol conversion so that the sensors appear as virtual nodes participating in the IoT.

Analyze — Situational Analysis

Once the data has been acquired, a real-time, event-driven business application is responsible for analyzing the data, producing events or situations that represent business or technical conditions that require a response. The application then initiates an automatic response to the current state of the machine or customer, and/or a collaboration between the appropriate operations personnel and the system, to produce the optimal response.

Events and situations are detecting by analyzing the data streams and their context using rules, statistical methods and machine learning. Examples of events or situations that might be detected during analysis include:

- Equipment that is not performing to expectations with conditions such as high temperature or low speed.
- Customers that have arrived at an interesting location in a store or facility. For example, they are standing at a checkout kiosk or a specific merchandise display.
- User is in an unsafe area and needs help.
- The distribution of orders has changed requiring the attention of product management.

Act – Automation and Human-to-Machine Collaboration

Once a situation is detected, a response to the situation is generated. The response may be a response initiated autonomously by the system or a response determined via collaboration among the automation system and the responsible individuals. This is what we call **Action** and represents a large opportunity to both improve business operations or offer new opportunities:

- Providing relevant responses to consumers based on their current situation (items on sale, facility map, emergency response recommendations).
- Respond intelligently to exceptional conditions (close a valve, turn on sprinklers, stop a malfunctioning robot).
- Proactively alert personnel to opportunities/problems based on the current situation (extra delivery trucks available, shortage in part of the supply chain).
- Optimize the user or business resources to improve productivity and/or customer satisfaction (speed up an assembly line, advise sports attendees on the shortest path to their car).

Automated Responses

In response to a situation, an automated response may be taken directly by the real-time, event-driven business application or may be forwarded to a more specialized system for implementation. For example, an action to shut down a machine may be forwarded to the control system that directly manages the machine rather than having the application directly send a shutdown command to it.

Collaborative Responses

For situations where the optimal response may be somewhat ambiguous or where determining the optimal response is beyond the capabilities of the system, a collaboration activity involving the system and the responsible individuals develops the optimal response.

For example, the sensor readings may indicate there is a potential problem with a machine but not provide enough information to automatically decide to shut it down. Instead, the operations team collaborates with the system to review the current data and obtain further information, perhaps via a visual inspection of the machine, to determine if the situation warrants a shutdown of the equipment.

Some cases in which collaboration can produce optimal outcomes:

- Exception situations for which the data streams are inadequate to uniquely define the root cause and determine the best course of action
- Situations in which the operations team is privy to additional information not available to the system
- Situations in which a manual action must be taken on the part of a system that is not controllable online
- Situations in which policies or regulations demand more in-depth analysis of the situation before an action can be taken

Another important class of collaborations notifies interested parties of actions taken and the resulting new state of the system. Notifications can be delivered to other automated systems so that they can independently respond to the situation, or delivered to responsible staff via desktop PCs, mobile devices, and wearable devices. Notifications can also include recommended actions and situational awareness of pending problems.

Refer to the white paper "Real-Time Collaborative Systems", available from VANTIQ.com, that describes collaboration in real-time, event-driven business applications in more detail.

Distributed business applications

Many real-time, event-driven business applications are naturally distributed. In manufacturing environments, Programmable Logic Controllers (PLCs) communicate with area controllers and edge nodes that forward the data to more centralized IT systems. In consumer environments, data may be collected from numerous position sensors, processed locally into logical locations on which immediate automation decisions are made and forwarded to remote systems that optimize the experience for the consumer. Such a wide variety of distributed applications require support for an equally broad set of distributed topologies ranging from devices directly reporting to a central site, to hierarchically structured automation systems, to federated peers collaborating to improve a collection of organizations or businesses.

The simplest architectures are sensors reporting to a central site. Many examples of such systems exist today. A system collecting sensor data from a mobile phone and reporting that data to a cloud service represents a common example of a centralized architecture. These simple *star* architectures represent the bulk of the existing event-driven business applications as they are the easiest to understand and build.

More sophisticated architectures contain additional levels of processing and connectivity. Hierarchical systems are more complex and mimic many existing physical and organizational structures. For example, an industrial IoT system that consists of sensors reporting to local controllers that report to plant-wide controllers that report to divisional headquarters that report to corporate headquarters represents a tree topology. These systems provide both centralized and decentralized monitoring and control. Such systems are more responsive in real-time or near real-time situations. For example, it would be impractical to control factory equipment in real time by collecting the data, shipping it to corporate HQ and having corporate HQ systems determine the next action for the machine. It is far more effective to do such an analysis on the local controller and simply report the situation and the action taken to the plant-wide controllers and, subsequently, to regional and corporate HQ. Faster response times, improved availability and local control make the distribution of the situational evaluation, collaborative decision making and response processing across the hierarchical topology more efficient than moving everything to HQ and making all decisions in a centralized fashion.

Another classic example of hierarchical real-time, event-driven business applications is the use of edge nodes to act as local processors for a collection of sensors and control points with the edge nodes then interacting with more centralized systems.

The most sophisticated distributed real-time, event-driven business applications are peer-to-peer systems where peers are managed by separate organizations. For example, in an electrical demand-response system, the overall system consists of sensors managed by power utilities and sensors managed by utility customers while control of the system is distributed across the utility and its customers. To provide real-time demand-response, the utility system and the customer systems must collaborate. This is accomplished by each system making local decisions and transmitting both the local situation and the local decisions to the other party and then agreeing to modify their real-time behavior based on feedback from each other.

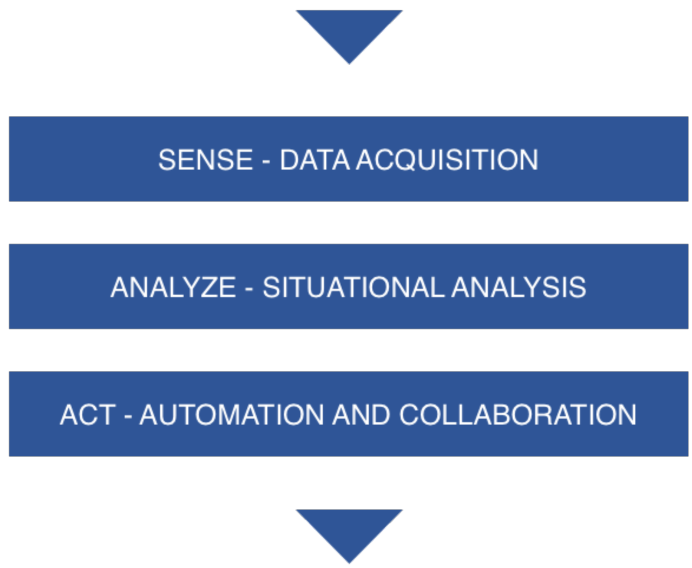
With this introduction to real-time, event-driven business applications in hand, the remainder of this paper focuses on best practices for developing distributed applications as implemented in the VANTIQ system.

VANTIQ PLATFORM

VANTIQ is a Platform as a Service (PaaS) for developing, deploying and operating high performance, distributed real-time, event-driven business applications consisting of:

1. Data Acquisition: Technologies for obtaining data from IoT and enterprise sources, filtering the data and making it available to the automation decision engine.
2. Event and Situational Analysis: Decision engine for analyzing the data in real-time and making business decisions based on the results.

3. Action: Technologies for sending control information to devices and for notifying external systems and users of the decisions or recommendations for subsequent actions being made by the automation solution. Technologies for managing collaboration between the automation system and the responsible individuals to develop optimal responses to complex situations.



The VANTIQ platform affords high user productivity by supporting the specification of real-time, event-driven business applications at a much higher level than traditional development tools. It also offers a set of tools that make the development, deployment, operation and management of the application as easy as possible. Since the data is externally sourced and the results must be delivered to external systems, a key to high productivity is flexible but simple integration capabilities for sources and devices.

The platform affords high performance by supporting a highly scalable cloud architecture which incorporates a number of performance-specific optimizations.

The platform is fully distributed supporting automated partitioning for the transparent development and optimization of automation solutions that span all layers of the distributed environment.

The solution is grounded in the following design principles.

Design Principles

- Lightweight, supporting simple topologies involving as few as two – or as many as n - participants with no additional moving parts to deploy. For example, no name service or coordination service is required and, by implication, no need to deploy these as additional services
- Incremental expansion supports the addition of new participants in the distribution or federation without system-wide reconfiguration
- Support for both reliable and unreliable delivery of messages among the participants in the distributed topology. This means the system designer may assume full connectivity or intermittent connectivity, explicit flow control or implicit flow control via store and forward messaging
- Simple configuration using the VANTIQ developer's console, command line utilities or the REST service
- Robust VANTIQ security features to secure both the individual components and the interactions between components in a distributed topology
- Transparent deployment of data and logic to any or all nodes participating in the distributed topology
- Support simple integration of both IoT device data streams and enterprise systems. Both streaming and enterprise data must be available to accurately evaluate the current collaboration context and make optimal collaborative decisions based on that context
- Seamless collaboration between the real-time, event-driven business application and the individuals responsible for the operation of the system

DISTRIBUTED MODEL

Topologies

VANTIQ supports a general model of distributed and federated topologies. A distributed VANTIQ application consists of two or more nodes with each node representing a VANTIQ installation. A VANTIQ installation can contain a single service instance or a cluster of service instances. The VANTIQ installations are assembled into a distributed topology when an installation declares at least one "peer" node with which it desires to exchange messages.

VANTIQ installations, by default, are considered independently managed. A node, A, declaring another node, B, as a peer MUST have credentials to access node B. Thus, the system is naturally federated since a node may only exchange messages with another node if it has been granted sufficient rights to perform the desired operation on the peer node. Peering is symmetric. If node B wishes to exchange messages with Node A, Node B must provision Node A as a peer and have sufficient rights to access node A.

Since the peering relationships can be defined between any two nodes, VANTIQ can support any distributed topology. Also, the topologies are implicitly federated since authentication and authorization are independently managed at each node.

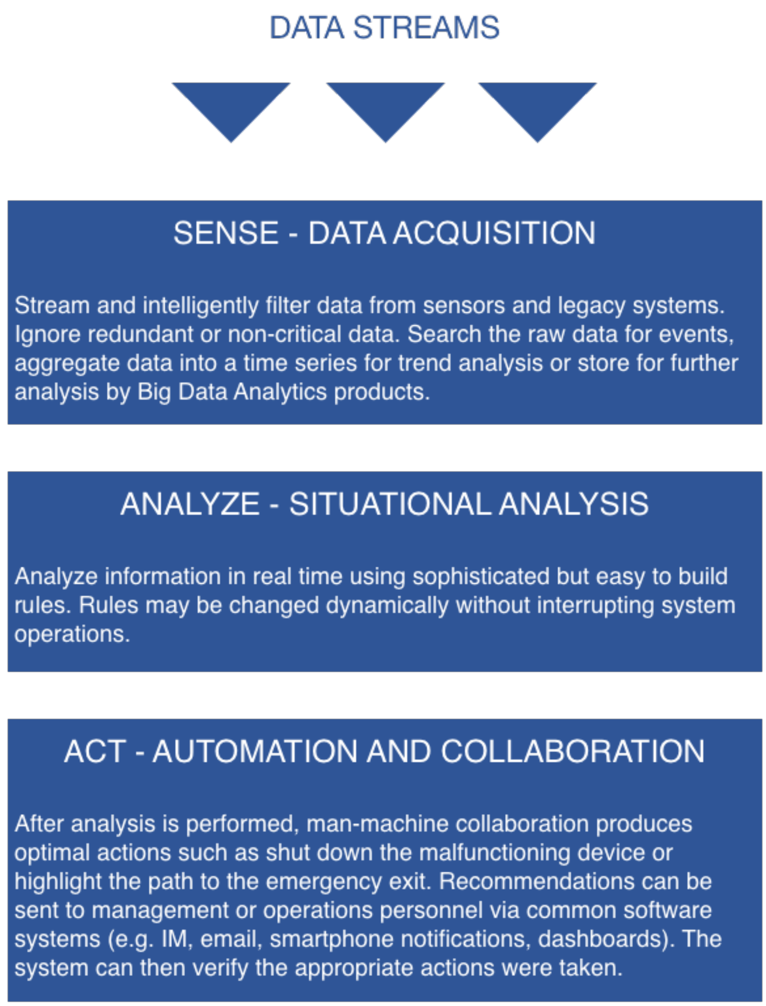
VANTIQ anticipates that initial usage patterns will favor topologies in which all nodes in the distributed system are managed by a single authority. Such systems are typically organized into star and tree topologies:

- Star – consists of a single parent node with an arbitrary number of child nodes.
- Tree – consists of a root node with an arbitrary number of child nodes where each child node may act as a parent for an arbitrary number of child nodes.

As the deployed system becomes more collaborative, more general federated peer-to-peer networks will be constructed. In such a network topology, any node may peer with any other node leading to a general graph structure representing the connections among the nodes. The network model tends to be the most complex since cycles in the graph are possible and the cycles must be handled by any functions that operate on more than one node in the graph. Also, because each VANTIQ node represents an independent system that may require separate credentials the systems naturally generalize to federations among collaborating organizations.

Functional Capabilities

Each node in a VANTIQ application implements a full set of VANTIQ services. All VANTIQ nodes support the key functions of:



Sense — Data Acquisition

The VANTIQ Data Acquisition subsystem can acquire data from a wide array of data sources by using standard protocols such as REST, MQTT, and AMQP. In the cases where older data sources only communicate via proprietary protocols, custom ingest agents can be developed using the ingest SDK.

The data sources may include IoT devices as defined earlier in this paper and enterprise systems that hold context required to evaluate the data flowing from sensors and placing the sensor data in the proper context. For example, if the application is assisting a customer by tracking their location, access to information in the CRM system is required to obtain the customer's profile information and more accurately assess the opportunities to assist the user at their current location. This places a heavy emphasis on the integration of existing systems as part of the application. VANTIQ supplies a wide range of declarative integrations for easily incorporating existing enterprise systems into the real-time, event-driven business application.

This service supports:

- Both push and pull models
- Synchronous and asynchronous models
- RPC (remote procedure call) as well as store and forward messaging systems
- The source may elect to send data by matching documented VANTIQ formats or can choose to have VANTIQ accept the native source format and use the VANTIQ filtering system to convert it to the proper format for internal processing.

With these capabilities, the service makes source integration simple by matching the interaction model and message protocols of the source rather than requiring the source to match VANTIQ messaging models.

Also, VANTIQ supports a model for managing data hosted behind firewalls that do not allow external systems to communicate directly with the data sources. The flexible nature of the VANTIQ ingest system allows such sources to provide data at their discretion rather than requiring the source to respond to an external request that cannot be delivered through the firewall.

Security is always maintained by requiring VANTIQ to use user-supplied credentials to access all data in peer nodes. Thus, every node has complete control in determining which peer nodes are authorized to access the local node.

Analyze — Situational Analysis

VANTIQ provides a sophisticated suite of services for situational analysis. The system is optimized for processing streaming data in both simple and complex configurations:

- Data from multiple streams can be correlated to assist in situational analysis. The developer uses a simple domain-specific language derived from SQL to specify that an event detected in one stream must come before or after an event in another stream, or both events must happen within a specific timeframe with the events occurring in either order. Even in cases where events do not occur, a common error indicator can be specified in a simple fashion. Event constraints can be composed to any level making the specification of complex conditions simple. For example, an automation system may monitor two sensor streams for a mechanical device with the first stream reporting speed and the second reporting position. If the automation system sends a stop request to the device, it expects to see the speed of the device as read by the first sensor go to zero and the position of the device to remain unchanged once a speed reading of zero has been seen. If the position changes AFTER a speed of zero has been reported an alert is generated. Also, if a position is NOT reported within 30 seconds of a speed of zero being reported an alert is generated indicating a potential failure of the device control system.
- Some of the streaming data is processed immediately or held only for a short time to facilitate time-series construction while other data may represent an extended time series or historical data that must be maintained over longer periods of time. VANTIQ simplifies the use of both transient and persistent data by unifying the abstractions used to represent series and set data in both its transient and persistent form.
- Data is analyzed by discrete collections of rules or by algorithms produced by machine learning systems and subsequently integrated into the application.
- A complete set of services is available to forward data to other nodes in a distributed VANTIQ topology using the SQL-based domain specific language to easily support real-time processing throughout the distributed environment.

Act – Automation and Collaboration

Actions may be applied directly to the internal state of the system. Actions are applied to external devices using source integrations that can deliver the actions to external devices or edge nodes using standard integrations such as REST, MQTT, AMQP and others, or custom integrations.

VANTIQ provides a very powerful model for creating actions or responses that involve collaborations between the application and its users. The collaboration model supports the rapid development of collaborations by composing high level collaboration patterns using a graphical editor. Collaboration patterns include:

- Notification – handle notifications and responses via SMS, EMAIL, push notifications and massaging systems
- Assignment – negotiate assignments of users to tasks
- Location Tracking – significantly simplifies the task of knowing when a user reaches a specified destination
- Recommendations – recommendation courses of action while adjusting to real-time changes in the situation
- Conversation – mediate a conversation among users over third party messaging systems
- Escalation – respond to critical delays in completing tasks.

Refer to the white paper "Real-Time Collaborative Systems" available from VANTIQ.com, that describes collaboration in real-time, event-driven business applications in more detail.

VANTIQ also supplies mobile clients that can be used to easily integrate people into the overall collaborative decision-making process. The clients are designed to support the most natural and efficient interactions possible. Users are automatically notified of situations that need their attention, custom interfaces for each notification supply the user with exactly the information they need. The user can respond by using all the data capture features of the mobile device – videos, photos, audio, location, acceleration, voice with natural language recognition, as well as traditional text entry.

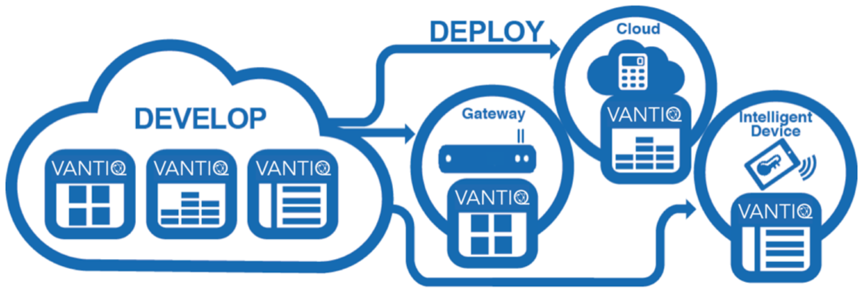
For users with more specialized user interface needs, SDKs are available for use in developing custom user interfaces for environments such as iOS, Android, the web, kiosks, and augmented reality devices.

Security

VANTIQ includes an extensive set of security features for protecting applications developed in its environment:

- Access to the system via the REST interface requires authenticated credentials. The generated access tokens are valid for a limited time to reduce the opportunities to steal and re-use the tokens.
- Any actions requested must be authorized by the authorization system. Access controls can be coarse or fine-grained depending on the requirements of the automation system. Fine-grained authorization may specify access controls down to a single property of a single object managed by VANTIQ.
- Access to sources operates under credentials supplied by the user. Therefore, no external system can be accessed without the initiating user being properly authorized. All credentials are encrypted for increased security.
- Encrypted communications with external sources are supported where the external source supports encrypted connections.
- Properties managed by VANTIQ may be encrypted before being moved to permanent storage.
- Communications with client applications are encrypted via TLS.
- Communications among VANTIQ services are encrypted via TLS.
- All security events are logged for subsequent auditing.
- Rules can be applied to the audit logs to automatically detect suspicious activities.

Provisioning



Example of Star Environment

Key to high productivity is an efficient provisioning and management architecture. Distributed systems managed by a single authority generally operate best with centralized provisioning and management. This applies to most instances of star and tree topologies. Network topologies tend to be managed by multiple authorities obviating the need for centralized management. However, to make the federation effective, collaborative models are required for provisioning and management.

The main provisioning and management functions supported in the distributed environment:

- Deployment of core VANTIQ runtime services (installation) to remote nodes.
- Configuring the distributed topology.
- Provisioning of VANTIQ application artifacts to remote nodes. The current artifact types to be provisioned include:
  - Types
  - Rules
  - Procedures
  - Collaborations
  - Analytical Models
  - Sources
- Monitoring the nodes participating in the distributed system for:
  - Health
  - Performance

    - Security

A general approach to provisioning and management is used for easier integration with other development and operational toolsets. All provisioning and management functions are available via:

- REST API
- Command line tools
- VANTIQ system portal

Several features of the provisioning system are optimized for use in distributed topologies:

- VANTIQ application artifacts may be defined on one node and automatically provisioned onto other nodes within the distributed environment either automatically by the VANTIQ system or under user control.
- VANTIQ artifacts may be changed and dynamically re-provisioned at any time.
- Capabilities are available to deploy definitions without activating them in preparation for activating the entire set of definitions once they have been fully deployed and checked.
- A versioning system is employed so that multiple versions of a definition may be created, selectively deployed and selectively activated in a context-dependent fashion. For example, a new version of a rule set may be developed and deployed that implements a new business optimization. However, you may not want to subject the entire national user community to the optimization until you have more experience with it under live conditions. You constrain the system to use the old version of the rule set for all users except users in Nebraska. The Nebraska users use the new version. Once the version has been fully validated, the activation constraint is modified to enable it for all users.
- Declaration statements for VANTIQ artifacts can be issued from a rule. This supports sophisticated installations deploying and modifying rules automatically based on potentially complex internal situations. As a simple example, when a new node is added to the distributed environment, rules can be established to automatically deploy the

relevant definitions to that node. If the network bandwidth between nodes changes, the allocation of situation detection processing between the nodes can be automatically readjusted to reflect the increase (or decrease) in bandwidth.

VANTIQ supports the provisioning of all nodes participating in a distributed topology whether the topology is a:

- Star
- Tree
- Network

As mentioned previously, provisioning may be centralized at one master node or distributed, partially or fully, across the distributed topology. Key to the VANTIQ provisioning model is that all nodes can execute independent provisioning actions. For example, if node A provisions an artifact into node B, node B may subsequently provision that artifact into Node C, if C is one of its peer nodes. Provisioning can continue until the artifact has been propagated to all nodes in a star or tree. In a network containing cycles, the system detects cycles to stop such provisioning if an attempt is made to provision an artifact onto a node on which the artifact is already provisioned.

A major consideration in remote provisioning is how to maintain consistency between the artifact definitions on the local node and the corresponding artifact definitions on the peer nodes. VANTIQ supports a range of strategies for maintaining the consistent provisioning throughout the distributed topology and for recovery from failures occurring during the provisioning activities.

Management

The management system supports the operation of a distributed VANTIQ event-driven business application:

- Reports the current health of each node in the VANTIQ system using standard metrics captured by the VANTIQ infrastructure.
- Reports performance metrics focusing on response times, throughput and capacity throughout the VANTIQ system.
- Maintains security logs for auditing security sensitive actions.

Automation rules can be applied to the security logs to automatically identify security incidents and respond by sending notifications or temporarily reducing authorized actions available to the suspect users or accounts. Similarly, actions can be specified based on performance metrics generated by the system to optimize system performance and resource usage.

Performance and Availability Considerations

In most cases, performance and availability can be improved by performing computations close to the data and relying on fewer nodes participating in the computation of a single answer. For example, if the system consists of edge nodes communicating with sensors and controllers on one side and a set of cloud services on the other, it is best if the bulk of the data is initially processed and time critical decisions made on the edge notes. Otherwise, network delays will be incurred communicating with the cloud servers in the best case and unbounded downtime will be incurred in the worst case if Internet connectivity is lost or the cloud services are down.

If more than one node participates in computing a value or answering a question, the system will only work if both nodes are operational. This can be mitigated by connecting the two nodes in an asynchronous fashion via reliable messaging allowing one to be down while the other processes. Of course, this can result in uncertain delay times during periods in which one node is down.

VANTIQ facilitates optimization of data movement and decision making throughout the distributed environment by transparently moving artifacts, such as types and rules, to the optimal nodes either automatically or under user control. This facility alone provides a large increase in productivity because the functional allocation decisions can be made and changed dynamically. This allows rapid experimentation to determine the best performing allocations and simple reconfiguration as the load or topology of the system changes.

With processing allocated to the optimal nodes in the distributed environment, there is still a need to move potentially large amounts of data among the nodes. For example, sensor readings may be read, smoothed and summarized by an edge node but the summarized data still needs to be moved to cloud servers for use in more global optimization decisions and for forwarding to predictive analytics systems that will subsequently analyze the data looking for new business insights. With VANTIQ's support for a wide range of high performance messaging systems that can be configured as the links between nodes in a distributed environment, optimal data transmission architectures can be constructed. As with everything else in VANTIQ, the link definitions can be configured dynamically and changed at any time. Both reliable and unreliable, synchronous and asynchronous protocols are supported, IoT optimized messaging such as MQTT and AMQP and standard REST protocols among others. The user can choose the best protocol(s) for each pair of peer nodes in the distributed topology.

With VANTIQ, the system architect has complete control over the style of interaction between the VANTIQ nodes. Also, because of the dynamic nature of VANTIQ, the messaging bindings can easily be changed dynamically.

SUMMARY

VANTIQ provides higher levels of abstraction whereby the underlying complexities (distributed environments, IoT, mobile, collaboration, etc.) are hidden. The development of event-driven business applications can focus on the business logic rather than infrastructure and the challenges of new technology.

Real-time, event-driven business applications are naturally distributed systems so they can achieve the necessary scale, resiliency and security. VANTIQ automates the partitioning of the logic and provisioning to where it is the most effective.

To keep up with the pace of business, real-time, event-driven business applications will continue to evolve rapidly over time. To satisfy this requirement, VANTIQ offers comprehensive facilities to provision then monitor and adapt the deployed systems.

With the VANTIQ platform all enterprises can leverage their existing skill sets to augment legacy systems of record and quickly build custom, real-time, event-driven applications to power their digital business.