The Connected Locomotive: Leading IT/Technology

Authors

Mark Kraeling, GE Transportation
Mark is a product architect for the Train Performance group at GE Transportation in Melbourne, FL. He is responsible for integration of networking and applications with worldwide customers’ locomotives and infrastructure. Mark is a regular speaker at embedded systems and software developer conferences worldwide, focusing on the subjects of virtualization, safety-critical systems, and embedded operating systems and their application to the transportation industry. He recently published a book titled Software Engineering for Embedded Systems: Methods, Practical Techniques, and Applications. Mark also actively participates in industry standards development such as those sponsored by the AAR.

David Fletcher, iders Engineering
David is Vice President and Principal of iders engineering, based in Manitoba, Canada. He oversees the research, development, manufacturing and commercial efforts of a 140 person technical team, leading iders innovative efforts in the fields of rail and aerospace transportation systems, structural health monitoring, biomedical and other industrial technologies. Iders is partnered with GE Transportation and has developed a broad range of locomotive, track vehicle and wayside innovations marketed under the GE brand.

Abstract

A key theme among railroad customers is creating “smarter” devices on their assets. This allows better diagnostics and prognostics, which enables better utilization and efficiency. Advances in the industrial and IT technology fields have developed and furthered concepts such as the Internet of Things (IoT), edge management, embedded security, and virtualization.

The Internet of Things (IoT) enables smart devices and sensors to transmit information to a final decision-maker that must deal with this data. On a locomotive, high performance sensors such as accelerometer and vibration devices can help predict failures that may occur sometime in the future. IoT includes using both wired and wireless sensors onboard a locomotive, so that information can be collected and sent across a gateway to either onboard or remote processing. Onboard locomotive platforms that can collect and route IoT data allow suppliers, customers, or third parties to develop their own applications and/or back office to process the information. IoT promotes the thought process of collecting as much data as possible, then look for trends in the data to make better decisions.

Edge Management (EM) is a technology that allows applications, devices, and sensors to be remotely managed by a supervisor. EM is used on devices today such as smart phones where the phone communicates remotely and determines when applications and the operating system are out of date. Transactions occur, and in the case of applications, they can be updated automatically, and for larger items like the operating system, give the user a notification for updating. Systems onboard locomotives are being deployed today with edge management built-in, where software can be updated on the locomotive as new releases are available. Larger components, such as the control system, are updated when a maintainer starts an update process – but the software itself is sent wirelessly and is already present for the maintainer to initiate.
Embedded Security (ES) is a technology that is an ingredient to an overall Cyber Security strategy that is developed for a network. A network typically has different levels, with the largest being how the corporate network connects to other wide area networks. But it also includes the network that is extended to the locomotive. ES is used on the locomotive to do active monitoring of the security of the network, and how traffic comes in and leaves the locomotive. Local security strategies also look for intruders that could be on the network, and not only stops the network intrusion but also has the ability to “black list” or mark traffic from that particular computer as always being blocked.

Virtualization is a technology where applications from various suppliers can run on the same standardized hardware platform. Resources like memory, processing, and communications bandwidth can be allocated to each of the applications, and dynamically adjusted as necessary depending on the operating conditions. Onboard locomotive systems are now shipping with this technology that allows an open platform for applications to be developed. This allows applications to be added to the locomotive without having to “add another box”.

The conference session will cover these four technologies: Internet of Things, Edge Management, Embedded Security, and Virtualization, all shown as running on a locomotive. By moving forward with common, standardized, and open platforms that include these technologies, railroad operators can become more efficient in their operations.
Introduction to Information Technology

Locomotives have progressively become more computer and electronic-oriented. Technologies like pneumatic airbrake systems have been augmented with electronic interfaces. Control systems and displays have appeared – providing the operator with current status and performance information. When the locomotive is delivered to a customer, the electronic systems onboard such as the afore-mentioned brake and control systems, communicate with each other. However, as more advanced features and functionality are required, typically a hardware box with software is added to the locomotive. This “box” will communicate with the systems that it needs to, but may not talk to the systems that it should. If communications or information is needed from outside the locomotive, it is common for additional antennas to appear on the roof or even cell modems exclusively for this added hardware component.

Information Technology (IT) is a field that is dedicated to the effective communication between and management of computers and telecommunications equipment. The IT field focuses on various distribution technologies to transfer the information, and look for ways to standardize these methods and communication. In general, the IT field includes the computation, transfer, and storage of data and information.

Certain IT advancements are already onboard locomotives. Groups like the Institute of Electrical and Electronics Engineers (IEEE) promote standards that can be used across multiple market segments to promote compatibility and have helped to provide IT momentum. Technologies like standard IEEE 802.3 Ethernet connect newer locomotive systems together, and are used for high-bandwidth wired communications. Ethernet speeds of 1 Gb/sec are more readily available today, with Fast Ethernet 100 Mb/sec also being prevalent. IEEE 802.11 (commonly known as WiFi) has also been used for wireless locomotive to yard infrastructure communications. The IEEE has even announced that it is just starting to work on input for electrical standards for wired Ethernet communications, aiming for 1 Tb/sec speeds (1,000 times faster than current Gigabit standard). At this speed, an entire typical BluRay™ High Definition 1080p movie could be downloaded in less than a second.

Open Standards and Licensing

One key area that IT embraces is the use of standardization. First, standardization allows a published method for multiple computers or resources to communicate with each other. Each computer or device can develop and test their hardware and software against that standard, to make sure it works according to the standard. As standards become more pervasive, testing tools are developed to help in this effort. Second, standardization allows choices between multiple suppliers to meet the desired functionality. One example is a USB-based IEEE 802.11 wireless networking card. The supplier must comply with either USB 2.0 or 3.0 standards so it can be plugged into a computer, and must also comply with the IEEE 802.11/x standards to communicate with a wireless router or access point. Consumers can then shop for any added features that have been put in, while also evaluating against cost. But in any case, the purchaser will know that it will plug into any USB computer, and connect to any 802.11/x router or access point. It is important to note that the /x on the end of the standard specifies the service amendments or extensions added to the original standard, such as 802.11n. Having a published method or standard and allowing multiple supplier choices are important to standardization.
For railroads customers in the United States, the Association of American Railroads (AAR) also drives standardization in the railroad industry. The common theme for the AAR is “Setting the Standard for Safe and Efficient Railroad Operations”. Though it is focused on US railroads, much of the equipment that is shipped worldwide for railroads comply with these standards. The AAR was founded in 1934 as a railroad research, policy, technology, and standard setting organization to focus on safety and productivity in the US freight rail industry. Its Full Members include major freight railroads in the US, Canada, and Mexico, as well as Amtrak. Associate and Affiliate Members include non-Class 1 and commuter railroads, as well as suppliers, railcar owners, and consultants.

The AAR sets up committees to focus on standards and technology development. One example is the AAR Railway Electronics Standards Committee (RESC). This committee is responsible for publishing and maintaining the M-9101 “Locomotive System Integration Communications Specification”, for example. This specification is commonly referred to as the “LSI Specification”, and provides communications hardware and software standards for brake to control system communications, fuel tank monitoring, cab signaling, and LOCOTROL® Distributed Power messaging. Specifications like this one promote standardization across the railroad industry.

Another important standard from the AAR is the “Locomotive Electronics and Train Consist System Architecture” document, or S-9101. This standard specifies the onboard architecture and conventions for integrating systems onboard newer road locomotives. It includes the capability of being able to add infrastructure and new features, without displacing existing technology onboard the locomotive. A high level architecture is provided, and then separation of functionality into different groups, and ultimately sub-networks. Each sub-network, or subnet, is connected to specific devices, but all operate together with routing functionality.

Section 8.3 in S-9101 specifies a standard Ancillary Card Cage (ACC) that can be used by any piece of equipment on any of the subnets. The functionality is placed on individual hot-swappable modules that can be placed into the card cage. All mechanical and electrical standards are listed in this S-9101 specification, so that various suppliers’ modules can be incorporated with the other modules. Module examples include high performance processing, storage, and networking that relate directly to IT technologies discussed earlier in this paper. A picture of an ACC with various modules is shown on the next page:

Figure 1: AAR S-9101 Standard 8-Slot ACC Chassis
When integrating onboard locomotive equipment like an ACC, it naturally supports the integration of multiple products, and manages the various networks (or subnets) as listed in the specification. It allows all of the networked modules to use devices like the wireless module or GPS, so that each application does not need its own cellular device or GPS receiver. Using a standard platform like this is the first step to get to the goal of having a “connected locomotive”.

Another key for having a connected locomotive that can use key IT technologies is to create a networked connection. Integration of communications on a locomotive, that any networked device can use, makes more efficient use of communications paths, such as 802.11 WiFi and cellular communications. This also aids the upgrade path for the communications, so that as new technologies like 4G LTE (cellular) are deployed, the devices onboard the locomotive can take advantage of them. There are also solutions that have multiple SIM cards per cellular modem, which allows switching between carriers to make best use of coverage or cellular plans. Communications for the locomotive IT infrastructure include more than just WiFi and cellular, and can include licensed or unlicensed frequencies. Consolidation and integration of the communications for the networked platform allows each device to use these paths based on the device permissions as set by the locomotive owner.

Other open standards exist in the IT industries today that are directly applicable to locomotive electronics deployment. Operating systems like Linux are needed to run software applications that are onboard the locomotive. Developers can use various Linux distributions free of charge, but must comply with the GNU General Public License (GPLv2). In summary, if changes are made to the source code in the operating system kernel, those changes must be submitted back free of charge. Typically, users of Linux use the operating system kernel without changes – especially when developing software for processors that exist in the industry with a Linux software development kit (SDK) already. Applications that are written to use Linux still retain ownership of the engineer and/or company that created them. There is a large community of Linux developers that maintain specific distributions of Linux that can be used like Debian or Gentoo. There are over 250 different distributions of Linux, each specializing in a particular market or user space! Using Linux as the operating system helps keep application software interfaces more standardized, and certainly allows migration possibilities between hardware platforms.

There are many hardware industry standards that are also important to consider when building the connected locomotive. Standards like Com Express specify the mechanical and electrical interfaces for a single board computer, or SBC. The connectors, screw locations, form factor, and signal characteristics for the pin assignments are all specified. There are particular “Types” that are specified for the SBCs, each with a different form factor and connector configuration. As an example, the table below shows the functionality associated with each Type connector:
Table 1: Connector Type Functionality

<table>
<thead>
<tr>
<th>Type</th>
<th>Type 1 AB connector</th>
<th>Type 2 AB/CD connectors</th>
<th>Type 3 AB/CD connectors</th>
<th>Type 6 AB/CD connectors</th>
<th>Type 10 AB connector</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCIe Lanes</td>
<td>Up to 6</td>
<td>Up to 22</td>
<td>Up to 22</td>
<td>Up to 24</td>
<td>Up to 4</td>
</tr>
<tr>
<td>PEG/SDVO</td>
<td>2</td>
<td>1/2</td>
<td>1/2</td>
<td>1/NA</td>
<td>1/1</td>
</tr>
<tr>
<td>PCI</td>
<td>-</td>
<td>32-bit</td>
<td>32-bit</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>IDE Ports</td>
<td>-</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>-</td>
</tr>
<tr>
<td>SATA Ports</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>LAN Ports</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>USB 2.0/USB 3.0</td>
<td>8 / 0</td>
<td>8 / 0</td>
<td>8 / 0</td>
<td>8 / 4</td>
<td>8 / 0</td>
</tr>
<tr>
<td>Display Interfaces</td>
<td>VGA, LVDS</td>
<td>VGA, LVDS, PEG/SDVO</td>
<td>VGA, LVDS, PEG/SDVO</td>
<td>VGA, LVDS, PEG, 3x DDI</td>
<td>1x DDI</td>
</tr>
</tbody>
</table>

Using electronics processing modules that comply with this COM Express standard allows for upgradeability in the future, as well as takes advantage of being able to select from multiple vendors. Another standard that is used in industrial applications is PCI Express. This standard also specifies the mechanical and electrical interfaces for equipment to be compliant. One variant is the Mini PCIe standard, which includes the standardization of smaller footprint SSD mSATA drives. It includes the electrical interface for wireless communications carrier cards as well. Again, the more standards can be pulled into onboard locomotive products, the easier it becomes to upgrade and maintain them, which are key ingredients for IT technology.

Internet of Things (IoT)

One of the latest buzzwords in IT technology is the Internet of Things, or IoT for short. When the question is asked: “What is it?” a multitude of different answers result. One concept is around machine to machine (M2M) communication. Instead of having masters that collect data from slaves and then make decisions, the machines (or computers) themselves communicate with each other. This implies that decision-making is being pushed down to the machines rather than a master orchestrator. Cloud computing also comes into play – where information from (again) multiple machines is stored somewhere on a larger network, like the Internet. This is also discussed when defining what IoT is. The emergence of smart sensors, ever faster and cheaper processors and memory, plus “canned” algorithms and software, enables the conversion of data to information at the point of acquisition which in turn, enables what really is a true IoT system or architecture.

The better definition for IoT is substituting the word “machine” with the word “sensor”. This implies that the sensor itself has the capability to send information, whether it is to an actual machine or to network storage. This interconnection of machines and sensors is what makes the IoT world powerful – more information to make smarter and better decisions. Incorporating sensors into the system as opposed to strictly machines is the key to create IoT. Cloud computer – or the capability to store large amounts of data is important as well. Often companies that use IoT will collect information they have yet to determine is valuable, but post-event can look for trends or signs they can assess in the future to make smarter decisions. Store everything – then decide later what is important and refine it. The following diagram (Figure 2) shows the relation between sensors, machines, and cloud storage:
Building upon the importance of standards discussed in the previous section of this paper, few standards for IoT exist at this point. Companies have applied wireless standards such as IEEE 802.15.4 that can be used for battery-operated or low power sensors. There has not yet been a convergence of the actual protocols and messaging that companies use to communicate over this wireless standard.

A relatable example deals with home automation. For the past few years, companies have discussed smart refrigerators or thermostats that participate in the home network, and allow users, when away from home, to change settings or get status. But this has not taken off in terms of large growth because of a lack of a common standard. A thermostat from one supplier will require that supplier’s network application and, for instance, it will not communicate with the refrigerator of another supplier, requiring different applications to use. There is no way to correlate in one place that the rise in temperature for both during the summer really means that power is not being provided!

Examples of the varying standards in home automation include ZigBee®, Z-Wave®, Insteon®, and X10, just to name a few. These are standards that are effectively competing with each other for this market space. They are not compatible with each other – but there are suppliers that create sensors that meet two of the standards so they can have a single product that could be used in multiple spaces. This leads to cost and complexity – and variance in network topologies further complicates how the devices are used. Google-owned Nest Labs is leading a new consortium of partners to create the Thread Group, which intends to work on another standard for home automation that will use the same chipset and frequency range as ZigBee. This will include a standard and procedure for testing devices against the new standard, before the company is allowed to list their product as compatible. With Google now included into this space, there is a hope that standardization can finally be realized across the home automation market.

IoT has now extended to the locomotive market as well. Smart sensors collect data from various locomotive components – including sensors that measure vibration. Vibration sensors that are mounted around moving parts can detect the frequency and amplitude of events, to help identify either imminent mechanical failures or non-optimal operating efficiency. For the locomotive market, this data is typically stored on networked storage on the locomotive as opposed to sending all of the data to the Internet (cloud) or back office. The reason behind this is the cost of wireless data as the locomotive moves around the railroad network. Once the locomotive arrives in a yard where WiFi is present, the data can be uploaded from the network storage to the back office.
Another key component in the locomotive space for IoT is a “sensor gateway”. This gateway exists in home automation as well – it takes the message data that is being received on the sensor network and translates it to packets that are accepted on the machine network. An example is a wireless sensor mesh network that communicates over a protocol on IEEE 802.15.4 (often used for low-power sensors), then converts the data to Gigabit IEEE 802.3 Ethernet that can either be consumed by machines on the locomotive network, sent to network storage, or sent off-board the locomotive if the communication path is acceptable. Figure 3 below shows the locomotive IoT network:

![Figure 3: Locomotive IoT Network](image)

Once the locomotive data is sent to a machine or computer that can process the information, important decisions can be made. One example of a “smart” decision is for replacing parts. If the processor of the smart data determines that a part on the locomotive has either failed, or is about to fail, parts can either be ordered or moved to the correct repair location. Operations can simultaneously be informed that there is a chance that the locomotive asset will not be ready for the next operation, or even be given a chance of success measurement if it is used. Instead of sending all of the sensor data off the locomotive, if an application processor is present, it can then send an email or SMTP-type of message to the correct parties informing them of the failure or imminent failure.

For the locomotive and the Internet of Things, a standard platform can be used that incorporates this functionality onto a single network. The same locomotive standard that was discussed for the Ancillary Card Cage (ACC) in S-9101 already has the pieces necessary for the next evolutionary step in networks. Modules are placed into the ACC with their specific network and IoT sensor functionality. Sensor data is collected, the sensor gateway translates the data, application processing analyzes the data, and network storage keeps the sensor data. The following is a look at how this type of standard platform is used onboard a locomotive:
The diagram above (Figure 4) also shows a wireless module, which could be used for either WiFi communications while in a coverage area, or cellular if not. Message traffic can be restricted so that large transfers of information only take place in a WiFi area, where only status messages or alerts are allowed over cellular. This type of configuration helps restrict the amount of network message activity that might need to be paid for!

The Internet of Things and the number of connected devices is expected to grow at an extremely fast rate over the next few years. BI Intelligence is a company that focuses their research and analysis on connected devices, and released this graph (see Figure 5) showing the number of connected devices labeled the “Internet of Everything”:

---

**Figure 4: Ancillary Card Cage (ACC) Standard Network Capability**

The diagram above (Figure 4) also shows a wireless module, which could be used for either WiFi communications while in a coverage area, or cellular if not. Message traffic can be restricted so that large transfers of information only take place in a WiFi area, where only status messages or alerts are allowed over cellular. This type of configuration helps restrict the amount of network message activity that might need to be paid for!

The Internet of Things and the number of connected devices is expected to grow at an extremely fast rate over the next few years. BI Intelligence is a company that focuses their research and analysis on connected devices, and released this graph (see Figure 5) showing the number of connected devices labeled the “Internet of Everything”:
As advancements occur in IoT, many are readily applied directly to the transportation industry. The large consumer market for automobiles has fueled a lot of smart sensors, and mesh networks as discussed previously. There is much upcoming advancement in car-to-car communications. Currently, cars have sensors that use radar or video to see obstructions or other cars around them. All of the information that is acted upon is local to the car – nothing comes from other cars. General Motors has committed to its first release of car-to-car communications starting in the 2018 model year. Then cars will communicate their status to other cars, as well as receive it. Stoplights and other ground-mounted equipment will also transmit information, so the car can make decisions based on the environment around it. No longer will a car have to make decisions based on local data, coming in from multiple sources.

There is similarity with this scenario and the Positive Train Control (PTC) initiative for railroads in the United States. With this system, the locomotive communicates with wayside signaling devices wirelessly to get aspect states and changes. The PTC system onboard the locomotive can then make sure that train is stopped short of stop signals, or is at the appropriate speed for caution signals. When more information is available to the locomotive, smarter decisions can be made. This could include corrected GPS information, wind/weather conditions, or even possible status information from passing locomotives on what may be ahead.

IoT technologies, the incorporation of smart sensors, and the application to a standard platform for locomotives will lead to much advancement in the locomotive in the future.
Edge Management (EM)

Edge Management (EM) is the technology that involves the managing of edge devices or nodes on a network. An edge device is a networked component that takes information and translates it to a different protocol – in this case, it may be taking temperature and other measurements and translating it into network traffic that can be processed and managed. This works hand-in-hand with IoT: IoT is the larger collection of sensors and the network, and the edge device is the individual component.

One key component of edge management is the capability to remotely “manage” the edge devices and nodes. This management includes updating software, reading software status and health information, and setting parameters that dictate how the device operates.

Updating software is one of the more important features in edge management. When devices that perform a variety of functions, such as data collection or driving of inputs and outputs are deployed to the field, the ability to update software later with bug fixes or feature enhancements is critical. Otherwise, the device would have to physically be removed from the locomotive and replaced with another. It would then go back to a location where it could be updated. Another important aspect of software updating is having the sensor report information that is expected for that particular installation. If a single device could be deployed to multiple locomotive types, and then the sensor could be updated with the software that is appropriate for that locomotive type, the number of spare parts would decrease, and devices could become more interchangeable.

Reading software status and health information is also an important feature of EM. This does not mean the actual sensor or input status, but rather the health and status of the software load itself. Information like the software version can be used to determine if an update needs to be performed, or if the information just needs to be logged into a centralized network storage location. The health of the device itself can also be read – whether it is the internal temperature of the chipset that is on the device or whether it is noise seen on signals that are being read. Understanding this status and health information helps provide insight into the measurements that are fed into the system.

The last important feature presented is the setting of parameters on the device itself. The edge management device can send the sensor the parameter set to use based on the overall conditions that the edge manager evaluates. Environmental, locomotive, or system conditions could dictate the resolution of the data being sent, or even the update rate at which the device sends data. The edge manager can dynamically set these parameters on the device, again based on condition. This can help limit the amount of data that is being stored or even help set up alarms or conditional information so the device knows what limits to look for in the evaluation of data.

One example of an edge managed device is an electric meter at a consumer location. The utility company owns the device, and communicates with it to get status information for the amount of electricity being consumed. Options exist where customers can sign up for electric discounts in conditions of high electrical usage, where the utility can selectively turn off appliances such as the air conditioner (HVAC) system or the electric water heater. The electric meter as an edge device also reports electrical usage information back to the utility office, instead of having to send a person to manually “read the meter”. Finally, the electric utility can remotely update the software that is on the meter as new features or alarms are added.
An example of an edge managed device on a locomotive is remotely being able to update the control system software. Because of the complexity with control of the locomotive, this type of edge management is done when conditions are safe to update the locomotive (i.e. not moving down the track). The software package is sent from the back office to the locomotive wirelessly, even while the locomotive is moving, for storage on the locomotive until it is time to do the actual update. This eliminates the need for having a laptop or update device and a copy of the software for someone to manually bring onboard the locomotive.

A notification is then sent to the maintainer or service area, that a software update is available and ready for that locomotive. When there is sufficient time, the maintainer then gets onboard and on the screen walks through a series of questions including whether hand brakes and other items are set correctly on the locomotive. When the maintainer answers the questions affirmatively, the update process is launched with an estimated time to completion. The maintainer can then leave the locomotive and come back at the appropriate time to make sure that screen on the locomotive shows a successful update.

The most important part is notification to the back office that the update was successful. This notification to the back office is typically handled by edge management that is part of the update process for the locomotive. It orchestrates the order of which software components are updated, and manages the overall amount of time it takes to update the parts of the control system being updated. The diagram on the next page shows the software updater (edge management) and the control system devices (edge devices) onboard the locomotive:

![Figure 6: Locomotive Edge Management](image)

In summary, edge management is a key part of any integration with IT technology concepts. Being able to remotely update and manage the software, as well as change the behavior of a device is important when considering components as part of a network system.

**Embedded Security (ES)**

One of the hot topics in the Information Technology world today is the continuing threat of hackers and the ability to seize control of networked assets. Recent stories involving automotive, corporate, and Internet businesses have certainly grabbed headlines. Hackers have demonstrated control over systems
that were not meant to have remote connectivity, stolen corporate data, or inflicted denial of service (DoS) attacks that render online commerce businesses as useless until resolved. Certainly the publicity and threat of attacks is on an increased and heightened awareness level.

What is Cyber Security? Cyber Security, simply put, is the protection of networks, computers, applications, programs, and data from either unsecure or unintended access. Such access could cause harmful, destructive, or unintended access to applications and processes. Cyber Security typically starts at a corporate level. A policy is developed, often with the help of outside experts, and put in place for the entire corporation. As networks within this network are developed, they inherit the security and access policies of the larger network. The security of the network extends to remote networks as well, where either other remote facilities or assets use the same secure network channel to become part of this larger network. The secure policies and strategies that are developed at a corporate level then transition down to the local network level. Computers and devices on this network then become “trusted” and can access the network services requested. The question then becomes – how can the network or network system look for devices that are not operating in a trusted manner.

Embedded security fits well within this philosophy. When separate, remote networks are created within a corporate environment, there may be conditions or operating norms that do not fit within a typical network infrastructure. Once such network is onboard the locomotive. As the locomotive or other mobile asset moves throughout the network, connectivity to the larger corporate network can either be available or unavailable. The orchestrator or central network router then need to make sure it can administer and protect the local network as a whole while being disconnected from the larger network. This requires processing resources, network monitoring and control, and the capability of applying a set of rules when the corporate network isn’t immediately relaying these rules.

There are two types of embedded security. The first is passive. A passive security system does an excellent job of protecting the networked access based on a set of rules. But it does it in a passive manner. If a computer tries to access a particular network TCP port over and over again, a passive security system will block this access every time it is attempted to be accessed. This can go on indefinitely. The second type is active. An active security system not only blocks network access based on the same set of rules, but it actively looks for computers or devices that are attempting to break the rules. So for the same case where a computer is trying to access a particular TCP port over and over again, after a certain amount of times the active security system will block that computer from any future accesses of any kind. This, in effect, blocks that computer out entirely so that it cannot further pollute or “learn” about what ports may be open or closed.

There are three types of methods that active embedded security need to protect against. The first is the external network look, where the security is protecting the local network from a remote and possibly hostile computer. The second is an internal network look, where the security looks for computers that are breaking policy, or a computer that is behaving in a strange manner relative to the other local computers. The third is looking at “itself”, to make sure that an unsecure image hasn’t been loaded or applications on the file system haven’t been changed to something that is not expected.
The external network look typically involves a heavily active firewall system. The first step is to create a set of rules for the firewall that block all incoming and outgoing network traffic. Then a network administrator goes in and adds in very specific rules for the traffic that is allowed. This could include traffic from a particular IP address, message type (such as TCP or UDP), network port number, and/or a particular destination IP address. This is the most restrictive way to build a firewall security policy and certainly the most effective. External security also involves packet inspection of incoming messages to see if they match data patterns that exhibit destructive behavior. Finally, this external network look will focus on packets from local devices and computers that are outgoing – to make sure they are not going to suspect IP addresses that are not members of the approved network.

The internal network look involves more rogue computer inspection and packet classification. The first type of security that is deployed consists of a “honeypot”. The honeypot is setup to trick other computers that may have incorrectly plugged into a system to think it is one environment when it is really another. Special programs and features are setup at the typical ports (such as ports 22, 80, 443) so that when a local device accesses these ports, they are given an alternate view of reality. As an example, maybe a local threat with a laptop connects to the local network. The user (or hacker) then attempts to log into the routing system over SSH, or port 22. The router under attack then displays login information and may even allow the third password entered, to trick the user to think he is actually in the system. This is a total sham – where traffic is actually re-directed to record any messages that the computer is sending out. This data can then later be used to possibly identify the person, especially if it is a corporate-owned asset. The honeypot and information given to the user does not exist, and the network remains protected.

The internal or “itself” look for an embedded security package is literally looking at its local computing and network environment. This includes setting up secure boot images, so specific security keys are stored in the boot process of the router. If any image is loaded that does not contain the right key, the image itself is not loaded and cannot run whatever rogue or unintended image was installed. Another internal security method that is used includes active monitoring of the executables, libraries, and configurations that are being used on the system. Any unauthorized alteration of these files in any manner can lead to swift action including reporting of the change or even shutdown of the router itself if deemed critical enough to the intended operation of the program.

Whether the “look” of the embedded security is external or internal, the security that is deployed needs to work well together. Threats can come from any source, and the embedded security policy that is
deployed needs to work in conjunction with the larger corporate network when it is available. Indications of active alerts immediately to the back office and then appropriate personnel may catch hackers acts – especially if the attack is local while the locomotive is in a specific location. Data that is gathered during an attack, including what the attacking computer was broadcasting on the network at the time, could also be used to track down and find suspected violators of network policy. Having a strong embedded security option for the locomotive is a definitive IT-type technology that is important in the environment today.
Virtualization

The simplest definition of virtualization is the creation of a virtual (rather than an actual) version of an IT resource, such as the CPU, network interface, or storage. Virtualization allows programmers and users to think of how the program or application is going to be developed, as opposed to worrying about the specific hardware that it is going to be used on. When applications are written in a virtualized environment, they become very portable to other virtualized environments as they are developed, which minimizes the amount of software porting effort that needs to occur. Virtualization also moves away from a single processor, with a single operating system, and a single set of applications, to an environment where one processor can now have multiple operating systems, with multiple application sets being deployed.

A HyperVisor is the term that is often used for the set of software or firmware that manages the virtualization on a device. It manages all of the hardware resources, and sets up virtual protected memory spaces for each of the operating systems that are running. When running in a non-virtualized state, if an operating system crashes, the operating system and its applications reboot to clean up. Nothing runs on the processor until the restart occurs. In a virtualized state, only the application or operating system that needs to restart does, and the other keeps running.

The following is a diagram of a processor that is running in a non-virtualized state which is common today, and a processor that is running in a virtualized state that allows multiple operating systems to run concurrently independent of each other:

![Figure 8: Processor Examples](image)

The left side of Figure 7 above depicts a typical computer – such as a single laptop that boots Windows® as the operating system. If it crashes, it must reboot with any applications that were running. The right side of Figure 7 shows a computer running two operating systems simultaneously, with the protection between them in place.

Another advantage of running in a virtualized state is that resources can be assigned only to the operating systems (OS) that need them. For instance, if there are two Ethernet connections, both can be made visible to OS #1, where only one of the Ethernet connections is made visible to OS #2. This creates a boundary, so that OS #2 cannot generate any traffic on that particular Ethernet connection.

Another resource that can be assigned is virtual CPU cores. If running on multiple core hardware, virtual cores can be assigned to the operating system that is running. For instance, one OS instance may see four virtual cores, where another operating system may only see one virtual core. The number of virtual cores that are assigned to the operating systems can add up to more hardware cores than actually exist.
on the system. Loading is spread out over the cores and threads as the HyperVisor sees fit to do so – maximizing performance when it can.

Applications where virtualization makes sense are:

**Cost Reduction / Consolidation**: Instead of having multiple boxes or computers taking up space and needing to be networked together, the processor requirements at the peak state required are put together into a single processing area. This could include multiple processors for redundancy, but each “box” is given its own operating system and set of priorities to run.

**Split Workload**: Resources on the platform can be dynamically shared between multiple operating systems and applications. This includes dividing tasks by layer, such as having control elements exist in one operating system at a high priority and data elements exist in another at a lower priority. Cores, network resources, priorities, and storage space can be dynamically allocated based on the operating environment. Locomotive is stopped? Dedicate more data resources. Locomotive is moving? Dedicate more control resources as required.

**Sandboxing**: When a new application is brought onto the locomotive, a “sandbox” can be setup in the HyperVisor environment for it to run. Even if it crashes or does not perform as it should, it is in its own space so that it does not bring down the rest of the more mature applications. In this way, applications can be deployed quickly without risking the rest of the processing environment.

**Failover**: One of the most important aspects of running in a virtualized state with a HyperVisor is being able to failover between one operating system and another. OS #2 can be setup and running, and just checking to make sure that OS #1 is performing as it should. If OS#1 crashes, then OS #2 has all of the software needed to run and can start it immediately. OS #1 would then reboot, and enter into the monitoring function to watch OS #2.

The following diagram on the next page shows the various states discussed above that the virtualized HyperVisor environment can run in on a locomotive:
In order for virtualization to work in a given processing environment, there are some processor features that should be included. Without getting into deep specifics, the processor should support multiple privilege levels, an extended address space, translation of guest physical addresses in hardware, assignable direct guest interrupt management, and a programmable memory protection unit. The latest Intel® line of processors, such as the Core i3, i5, and i7 support all of these modes. The latest ARM® v7/v8 processing cores also support all of them – such as the ARM® Cortex A-Series processors.

Virtualization concepts also extend to smaller 32-bit processors. There are processors that support the modes above, but may run slower than the more powerful (and power hungry!) processors. For this, a concept called Linux Containers was developed. Linux Containers are like a HyperVisor, except all of the applications run on the same operating system. The applications are grouped into “containers” with their own user space and assignable resources. If a container needs to be restarted, the other container is not affected. Running in this type of system decreases the amount of overhead associated with a HyperVisor in that there are no large context switches that need to take place. But it comes at another cost – if the operating system needs to be rebooted or crashes then the containers are rebooted as well. However, it does provide some level of separation between application sets, and should be considered if running on a lower power processor.

There are many examples of virtualization in the IT technology world today. One common virtualization tool that is used is VMware®. This is typically used in a PC desktop environment, and allows the host operating system to create a space for a guest operating system. This would allow someone running Windows 7® (the host) to create a separate environment for Linux to run (the guest) within the operating system. This is traditionally called a Type 2 virtualization environment, as the host is protected from any reboots or crashes that occur with the guest, but the guest is not protected from the host. If the host goes down, guests do as well. A Type 1 virtualization environment has the operating systems running side by side, so they are each protected from the other.
The locomotive can also take advantage of virtualization. Examples that were already provided above were failover, sandboxing, and consolidation.

Failover on a locomotive would be deployed by having multiple virtualized operating system and application sets monitoring each other for health and status. When changes in that status occur, the programs that are still running can take action including operating in a "limp home" state, or send appropriate diagnostics and messaging to flag failed systems. Failover can also be done by having multiple processor farms run designated functions, with the processor modules themselves coordinating what applications need to run and which do not. Once applications are written to use virtualized processors, moving that application from dual-core hardware to quad-core hardware is straightforward and easier to do.

Sandboxing can also be a valuable feature for a virtualized locomotive. This would allow more rapid software deployment, so benefits can be gained earlier than waiting for a prolonged software testing process. If the application and operating system are placed in their own environment, and allocated a minimal set of resources, it will not impact the other applications running in the system. One example is network bandwidth allocation. In a virtualized environment, the HyperVisor can assign parts of the total available bandwidth. The sandbox operating system could be assigned at most 10 Mb/sec on a 1000 Mb/sec link. Other, more mature operating systems can be assigned use of the full 1000 Mb/sec bandwidth. The HyperVisor will restrict the sandbox to always have less than its limit, but not regulate the other operating systems' bandwidth usage. Like the virtual CPUs, the virtual network bandwidth can add up to be more than is actually available.

Consolidation is another benefit to locomotive installations. Instead of the concept of always "adding a box" when a new feature is required, a software package can be deployed to run in the virtualized processor environment. Virtualization allows multiple operating systems to run, and they do not need to be the same type or version. There could be eight Linux operating systems running at the same time as a Windows® operating system. They can all each have their own versions, drivers, and application sets to run.

The following is a diagram of the processor environment on standardized AAR S-9101 hardware, showing multiple processor modules running multiple virtualized operating systems:
The only negative to running virtualization on hardware is whether the HyperVisor itself will take up resources and CPU utilization to manage the operating systems that are running concurrently. This should be evaluated – sometimes through consolidation of disparate hardware costs are going to be reduced so a little extra cost can go into running faster hardware. Today, industrial-grade processor modules that are designed for locomotive use can run even the latest Intel® Core i7 hardware – so there is a path in upgrading processor performance. Protection modes, resource allocation, allowing rapid prototyping of software or sandboxing, and cost savings are all real strong reasons to consider virtualization technology on a locomotive.

Conclusions

This paper is meant to give an overview of how information technology (IT) is already starting to move into the locomotive, and the various feature sets that are available. Open and published standards and operating systems continue to be important, so that multiple vendors can be evaluated and a large user community can be leveraged. These standards are already widely accepted in the IT space, where standardization helps with rapid deployment, better maintainability, and compatibility between multiple devices.

With incorporation of these technologies, there are reduced costs as well. Integration of standardized hardware and centralization of communications technologies yield lower capital and operating costs. Instead of having multiple cellular modems, antennas, and systems that cannot talk to each other the systems can become more integrated and share information. Redundancy that is required in the system
can be done with these technologies as well – with either multiple paths for communication or use of multiple processors with virtualization to balance the workload.

The Internet of Things continues to have a strong growth path in the next coming years, with more devices and, most importantly better, smarter sensors moving into the market space. Transmitting, collecting, processing, and storing this data will have significant growth in the transportation industry.

Edge management was also discussed, where edge devices and components can be remotely managed and updated. As the sensors and devices become more intelligent, their state and even operational parameters can be viewed and changed, and allow the customization of the sensor conditions based on the operating environment.

Embedded security is an essential building block when building any locomotive network. By serving as the foundation of a network design to prevent outside and inside network intrusions, it becomes an important part of the overall Cyber Security strategy. Keeping the locomotive network secure, and being able to report and publish alerts on the state of security continues to be a leading embedded technology.

Finally, virtualization has started to be deployed in the locomotive space. Getting away from the “add a box” solution to getting additional functionality has been going away, and instead open, standardized platforms are used to consolidate processing and storage. Operating software in a virtualized state allows easier upgrades, assignable priorities, and protection between applications on the locomotive.

The Connected Locomotive: Leading IT / Technology!
References


Kraeling, Mark and Van Ackeren, Peter. “Virtualizing the Locomotive with the ARM®-based QorIQ Processor”. FTF Conference, Austin, TX. As presented and published on June 25, 2015.


The Connected Locomotive: Leading IT / Technology

Mark Kraeling / David Fletcher
June 6, 2016
Speaker Background

Mark Kraeling

- GE Transportation
  - Products for Locomotive Onboard
  - Based in Melbourne, FL
- Product Manager/Architect
  - Wired/Wireless Communications
  - Linux and ARM®-based Designs

Agenda

- What are we addressing?
- Industry standard solutions
- Internet of Things and the Industrial Internet
- Edge Management
- Embedded Security
- Virtualization
- Conclusions

What are We Addressing?

- Safety, Operations, Mechanical and other locomotive groups want to add products and solutions to the locomotive
- Often times, these “boxes” are not integrated together, and are point solutions
- Industry Trends: Standardization, Networked Functions, and Coordination of Information

What Needs to Run?

- Wireless
- Communications Security
- Checking
- Consist
- Communications
- Power
- Management
- Fuel
- Management
- Camera
- System
- Network
- Management
- Reporting
- Security
- Checking
- Consist
- Communications
- Customer
- Applications
- Locomotive
- Applications
- Rail
- /Wayside
- Integrity
- Monitoring
- Precision
- Location
- Network
- Storage
Initial Locomotive Applications

• Locomotive ownership is complex
• Lack of coordination between internal groups

AAR S-9101 Rationalization

• Creation of the AAR s9101 standard
• Built around open, robust Linux Networking
• Modularity
• Integration
• Modules

Industry Standard Solutions

Industry Standards

• Processor Commonality
• Operating Systems
• Open Source / Customer Configurability

The Industrial Internet (aka Internet of Things)
Industrial Internet

- Smart Sensors have the capability of reporting information to a variety of destinations, including to the back office
- Some information may be acted upon immediately

Internet of Things: Locomotive

- For IoT to be a complete solution and have the most impact, all of the locomotive systems must be able to share information and work together to form conclusions
- Also important to coordinate with other locomotives

Edge Management

- System wide locomotive asset management
- Common communications, data language, database, with proprietary subsets
- Optimize edge asset utilization and sharing between organizations

Edge Management

- Information “to” the device
  - Software updates
  - Configuration files
  - Containers
  - Operating parameters / orders
- Information “from” the device
  - Periodic data
  - Alerts and associated data
  - Log files
Embedded Security

Protect the Network
- Passive Protection
  - Block network traffic based on type, port, etc
  - Will protect based on configuration file, “forever”
- Active Protection
  - Continuously looks for attacks from local or remote networks
  - Can blacklist offending device / alert back office
  - Network packet inspection
  - Create “HoneyPot” to provide false information

Protect the Device
- New Processor Technology
  - Secure boot / trusted platform architecture
  - ARM® Trusted Zone
- Monitor File System
  - Look for unauthorized changes to configurations
  - Track acceptable files / remove offenders

Virtualization

What is Virtualization?
- Virtualization – an abstraction layer that enables running multiple guest operating systems on a single system, implemented using hardware and software technologies
- A hypervisor is a software component that creates and manages virtual machines which can run guest operating systems

Why Virtualization?
- Cost Reduction
- Flexibility and Scalability
- Reliability and Protection
What is Virtualization on a Locomotive?

A key ingredient for The Mobile Data Center

Conclusions

• Industry standards enable compatibility, ease of upgradability, and technology advancement

• IT Technology is important for locomotives, enabling
  • Industrial Internet
  • Edge Management
  • Embedded Security
  • Virtualization

• The Connected Locomotive – the customer's IT mobile data center

Thank You!