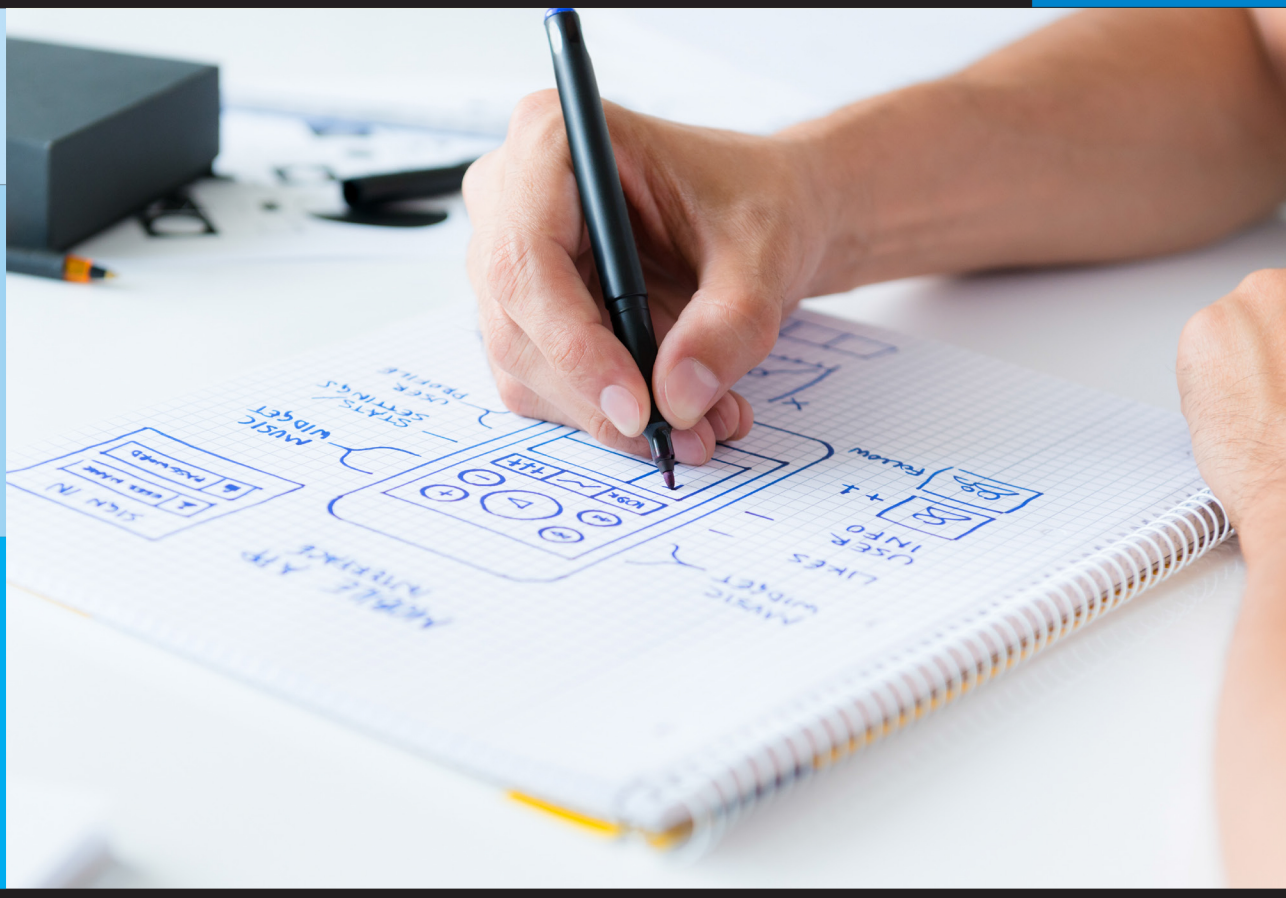
PEGASUS ONE

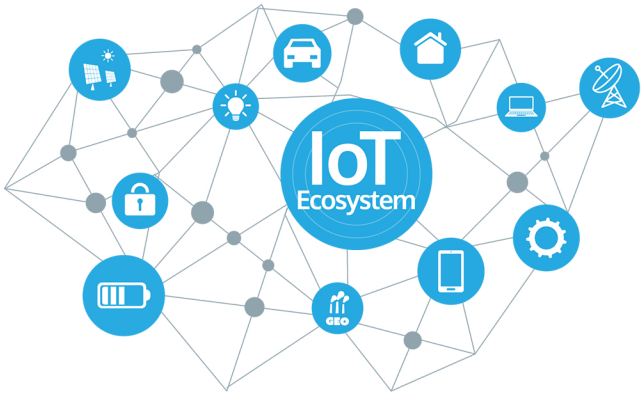
A COMPLETE GUIDE TO IOT DEVELOPMENT



Pegasus One

1440 N HARBOR BLVD #900, FULLERTON, CA, 92835 | 10880 WILSHIRE BLVD., # 1101, LOS ANGELES, CA, 90024

CONTENTS



IoT Development Trend

The fourth industrial revolution is creating an environment in which everything will be perceptible, interconnected, and intelligent. IoT is the cornerstone of this new era. ICT powers the ability of IoT to reshape traditional industries. By integrating the physical and digital worlds. IoT shortens business processes, boosts productivity, and provides better products and services, while, at the same time, unleashing the huge potential for innovation.

In the future, everything will be connected in scenarios more diverse than simply connected people. For example, governments want to make everything intelligent. From street lighting, parking, and bicycles, to water meters, gas meters, manhole covers, fire protection, and environment monitoring, governments hope that IoT will improve quality of life and increase city management efficiency. This will bring a new wave of connectivity services, presenting great opportunities for development. At the same time, the IoT platform is used to integrate the open asset data of different industries. This data may be sourced from water, gas, and electricity meters, intelligent door locks, pet tracking, home security, luggage, vehicles, and so on. Through the unified portal, IoT can make people's lives both smarter and more convenient.

As everything becomes connected and intelligent, the IoT brings huge economic value. It is driving the digital transformation of all industries. From governments and organizations to businesses and local communities around the world, everyone is actively investing in and researching the IoT. They collect, analyze, and apply data generated through the IoT, facilitating the rapid development of all industries.

According to Huawei's Global Industry Vision (GIV) predictions, everything will be brought into a digitalized and intelligentized world where everything is both perceptible and connected. By 2025, it is predicted that 40 billion smart devices will be in use worldwide, with a total of 100 billion connections in public utilities, transportation, manufacturing, medical care, agriculture, finance, and other industries. The IoT promotes digital transformation, creating a digital economy worth US$23 trillion'. With the comprehensive improvement of perception and connectivity capabilities, the IoT connects huge numbers of devices to achieve breakthroughs. It not only creates value from data but is also becoming part of our everyday life.

IoT Development: What Matters The Most

The Data

For an infrastructure made up of tiny little pieces, the IoT can certainly generate huge volumes of data. It's easy to say that the data has to go somewhere, but one of your early decisions is whether that's true. The decision is critical because knowing what you're going to do with your IoT data will determine whether those bits ever leave the edge -- and how long they get to stick around if they make the trip to your enterprise core.

Your decisions on the data will have a major impact on the communications you use, how many tiers the application requires, and just what sort of I/O engine you're going to need on the backend. We'll leave the storage component to another discussion. So, before you go too far in development, figure out the data piece -- and think carefully about just how much data you want to deal with once you've left the edge of the application.

The Platform

If your IoT plans include a network to monitor and control your own systems -- a network that stretches to include a device count in the scores or hundreds -- then an Arduino or a Raspberry Pi platform could work. If, on the other hand, you're going to deploy an IoT system to thousands upon thousands of customers, then you'll need to choose a compact, rugged platform that can easily be gang-programmed (or have the software flashed during manufacturing) and inserted into the device.

The fact is, the platform matters. Small embedded systems are not infinitely interchangeable. Enterprise software developers aren't accustomed to choosing the hardware platform on which their software will run, but for embedded control developers it's part of the game. You'll need to pull the two sides together to think about the long-term implications of whichever platform you choose. Whether you ultimately go with one of the popular prototyping boards, or with a custom design in conjunction with a chip development house, your choice of platform will have a huge impact on both the immediate success of your device and the ability to revise the design in the future.

Security

Security is a great deal for the IoT. Concerns about security are among the biggest reasons for people and organizations to delay implementing IoT systems, and their concern is echoed by many security professionals. In short, many professionals think that the IoT is an unmitigated security disaster -- and there are a lot of IoT devices and systems that do absolutely nothing to prove them wrong.

The solution is to design security into the system from the very beginning. That means ignoring sentences that begin like "No one would want to.." and spending brainstorming sessions getting extra paranoid. How paranoid should you get? How about letting penetration testers have a go at your system? IoT devices are going to collect data on and control some of the most sensitive aspects of home and business. You can bet that hackers want that data. We take all precautions to make sure that they don't get it.

Remember you are connecting many devices that no one ever expected to plug into your physical, virtual or cloud networks. And we're not just talking about consumer endpoints like cars or refrigerators. Essential commercial components such as dialysis machines, injection molding systems and retail merchandise inventory sensors are among the heretofore unconnected and unmanaged endpoints in the industrial IoT. And if you thought patching garden-variety PCs was a chore, think about applying proper security frameworks to these new, diverse endpoints. Your IoT security framework may be the single most important technology decision you will make, in no small part due to its huge regulatory, financial, operational, legal, user experience and brand reputation impact. Your IoT systems must be ready to not only sense the appearance of new kinds of endpoints, but also instantaneously analyze their legitimacy, access and privileges under policies and identity management.

Things

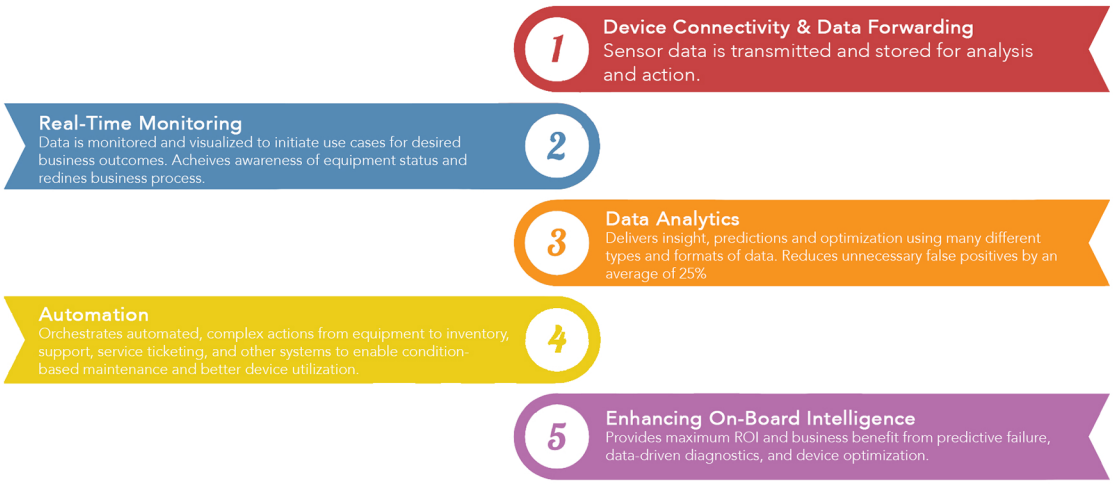
One of the problems in talking about the IoT is that the term means different things to different organizations. Connected programmable residential thermostats? IoT. Fitness trackers? IoT. Networked industrial process control? IoT. You begin to see the problem. The problem does nothing but grow when you hear some people talk as though the IoT is a monolithic, homogeneous whole.

The fact is that the things that make up the IoT matter. The platforms, communications, precision required, and safety margins are different for the different values of "IoT." Think carefully about the nature of the system that you're building and the devices that sit on the perimeter, and don't allow yourself to be convinced that "one size fits all" when it comes to the things you're introducing to the Internet.

The great potential of IoT is going to put strain on legacy infrastructure, so ensure your client is prepared to make infrastructure investments—especially for networking and connectivity. After all, the volume, variety and velocity of data going over both wired and wireless infrastructure are expanding dramatically, with no end in sight. For instance, make sure they have modernized physical network switches and management tools to ensure nonstop flow of information. A steady, reliable and comprehensive flow of all data—structured and unstructured—is essential for the kind of analytics that is going to be a critical component in your IoT use case deployments and measurements, and you can't do that without robust, secure and scalable infrastructure.

Make It Agile

Enterprise software development has gone full-bore into the agile discipline. Managers and programmers now speak as easily of scrums and stories as of procedures and compile cycles. Embedded design, on the other hand, has long been a world of the "write once and forget it" discipline, in which the common assumption is that the hardware and software will never be touched



IoT is a maturity progression as an industrial organization's business goals evolve. This maturity model illustrates the stages of IoT maturity and a snapshot of benefits gained along the way. Starting with a sound IoT strategy, companies can achieve maximum value upon completion of all five steps

Five Stages of IoT

Stage 1: Device Connectivity and Simple Data Forwarding

Every IoT story starts with a smart, connected device. Often companies begin their IoT progression by outfitting their equipment and devices with sensors that collect all manner of information and data. Often referred to as telematics, machine-to-machine (M2M) or IoT products and solutions, a gateway or similar communications device transmits sensor data from the device to a location where the information can be stored for future use. Most of these systems operate with very limited communications bandwidth, particularly in industries with remote or highly mobile environments. This means only a small subset of the data produced is actually collected. Even in situations where bandwidth is not an issue, the cost of transmitting and storing such vast amounts of data, which may or may not be important, can be prohibitive.

All of this is a crucial first step in enabling devices to form the foundation of an IoT solution. Without the device data, none of the subsequent stages are possible. However, simply adding sensors and intelligence to equipment and collecting data requires investment but doesn't produce business benefits. A system that performs some type of monitoring or analysis is required to begin to extract value from the device data.

Stage 2: Real-time Monitoring

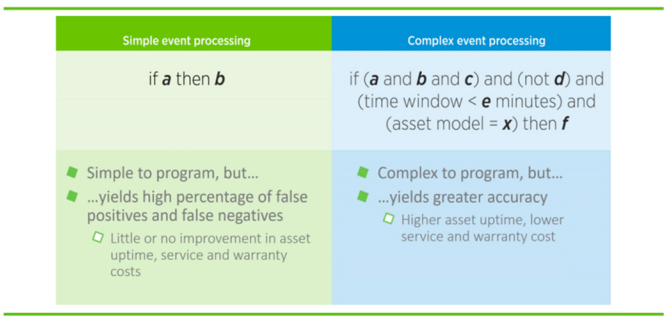
As data is collected, it must be monitored in real-time and visualized to begin enabling the use cases that lead to desired business outcomes. Common use cases for industrial equipment are: condition-based maintenance to improve operational efficiency and reduce service costs, device utilization information to help guide future product design and improve regulatory compliance, and IoT device management for enhanced device integrity and lower operating costs. Monitoring and alerting can help companies gain awareness of equipment status and start to adopt and refine business processes that improve outcomes.

In many cases, the monitoring system is a dashboard that provides basic information, data visualization, and simple alerting. For instance, the dashboard displays an alert when a temperature gauge exceeds a specified threshold so the person monitoring the dashboard can take steps to diagnose and service the issue.

At this stage, data is often called "actionable data," but the action must be taken by humans, who simply can't stay on top of the vast amount of monitoring and alerts that take place on the flood of data generated as real-time monitoring is rolled out to more and more equipment. One of the biggest hurdles along the way is that the dashboard-human operators approach cannot scale effectively. Software needs to perform monitoring and analysis – not humans.

With real-time monitoring, companies can see some level of condition-based maintenance; however, using this approach often results in an unacceptably high rate of false positives or false negatives. These are error codes or conditions that indicate a problem but when the equipment is serviced no problem is found (false positives), or error conditions that are not correctly flagged (false negatives). Basic dashboard solutions simply cannot detect complex conditions and events as they attempt to apply simple logic to complex equipment.

To extract real value from machine data, complex event processsing capabilities in the form of data analytics are required. These go beyond simple event processing (if a then b) to apply rules and analyze multiple data sources to gain valuable insight and truly actionable data.



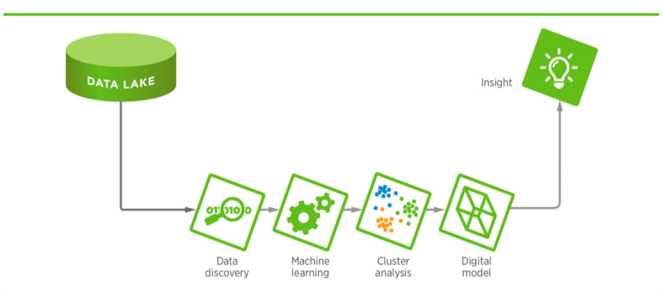
Stage 3: Data Analytics

Data analytics can deliver insight, predictions and optimization, and reduce unnecessary false positives by an average of 25%, but culling insights from rivers of data is an involved process. In order to support IoT, many different types and formats of data are likely to be needed. To effectively use all these different data sources, an extra ingest step is required to align the data; for instance, transforming all temperature data from Fahrenheit to Celsius or enriching data to give context. There are several elements required for a successful data analytics system:

Data discovery: First and foremost, it's important to have the right kinds of data to support the desired business use cases and outcomes. The appropriate types of data can be identified and collected by determining whether additional sensor data is necessary to provide essential device information, and what business systems may require integration. Only then should complex event processing capabilities for in-depth analysis and actionable insight be applied.

Machine learning: Algorithms can then be applied to the large pool of data to accomplish much of the heavy lifting required to identify correlations and patterns that may have been done manually in the past.

Cluster analysis: Once machine learning is applied to the data, groups of equipment that behave similarly can be identified to help understand how the environment is working.



Digital model: This is a representation of how the equipment behaves. Beyond the basic anatomy of a particular piece of equipment with a given number of sensors, the behavior of how those sensors interact with each other leads to insight.

Taken together, these elements provide truly valuable insight that allows appropriate actions to be taken – in many cases autom-

atically. In the example of condition-based maintenance, the ability of complex event processing logic to examine historical data and other contextual or system information (such as equipment specifications) in concert with machine sensor data provides much more accurate profiles and event prediction than human operators ever could.

The insights gleaned at this stage further a company's ability to make progress towards several use cases: predictive failure for increased asset uptime and elimination of false negative and positive reports, condition-based maintenance, device optimization for improved asset performance, and device utilization.

While insights alone may provide more sophisticated dashboards and a better picture of what's occurring in the organization's environment, the system is still reactive – not proactive. The human scalability challenge outlined in Stage 2 steadfastly limits IoT success as companies often underestimate the fire hose effect of events, conditions, and data streams generated by a growing amount of connected complex equipment. Automation solves this problem.

Stage 4: Automation

Once rules and automation are applied, complex actions can be orchestrated across multiple areas such as integration into inventory, support, or service ticketiticketing systems. Additionally, there may be data collection rules that can be changed. For instance, if the device is healthy, less data is collected and transmitted, but when the rules monitoring the system's health start to determine that conditions are approaching a potential anomalous condition, data fidelity can be increased by collecting more data. Rules can be enabled for improved safety, as well. If the system recognizes a specific set of conditions, it can shut a machine down without waiting for a control room operator to respond.

Rules must be dynamic in nature. It's a widely held belief that analytics is a "once and done" or infrequent exercise. The reality, particularly in the world of complex heavy equipment, is that analysis is only as good as the moment it is put into place. At the outset, the system collects data, performs analysis, gains insight, and turns it into a set of rules that gets applied to the real-time environment. Everything goes along smoothly.

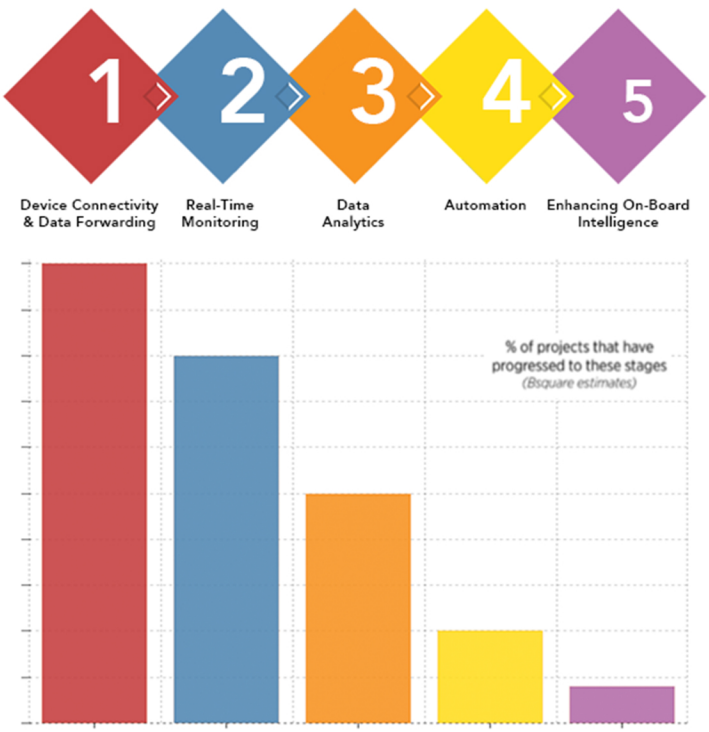
But fast-forward to 30, 60, or 90 days later and rule effectiveness begins to drift. In many cases equipment may be operating in challenging environments and can be affected by external factors. There may be hardware or software revisions that change equipment behavior enough that the rules aren't as applicable as they were during the original analysis. Automation can help manage rule drift.

Organizations completing this stage can achieve the full benefit of several use cases, including condition-based maintenance and device utilization – and for some businesses, this may be sufficient. Significant benefit also can be realized for predictive failure, data-driven diagnostics, and device optimization. Yet companies can gain additional value across these use cases, as well as IoT device management, by making their equipment and devices even smarter.

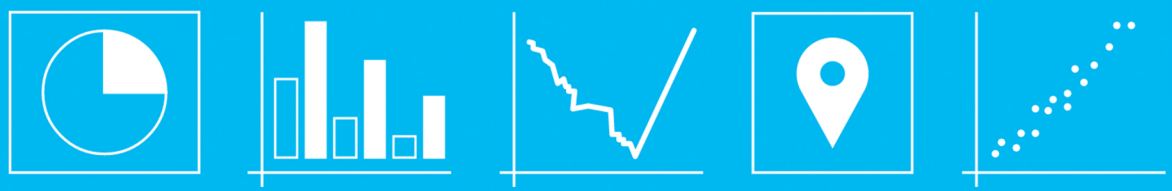
Stage 5: Enhancing On-Board Intelligence

Distributed intelligence, edge computing, edge intelligence, edge analytics. These terms all describe the same fundamental concept – processing data on or very close to the connected equipment in addition to functions performed in gateways or the cloud. Rather than moving the data to the logic, on-board intelligence brings the logic to the data. In the case of complex industrial machinery, much of today's connected equipment already has computing capabilities that can be tapped to perform data analytics and automation directly on the equipment, in real time.

By adding analytics and automation to equipment, all data can be analyzed for greater accuracy. In a typical scenario where limited data is transmitted for analysis, decisions are being made on a tiny percentage of the available data. By having 100% of the data available, results are faster, more accurate, and eliminate the need, and associated costs, to transmit and store unnecessary data.



On-board intelligence brings the IoT maturity model full circle, allowing industrial organizations to gain maximum ROI and business benefit from predictive failure, data- driven diagnostics, and device optimization. Further, true IoT device management becomes a reality as on-board intelligence monitors for conditions in order to identify events and then automates actions directly on the equipment for better predictive accuracy and more rapid response time. It also enables valuable functionality when equipment loses connectivity. For example, machinery can be rapidly and automatically shut down when there is an unsafe condition. Other important benefits are the enablement of over-the-air (OTA) software updates for improved device integrity and OTA operational configuration to dynamically alter asset behavior as requirements and environments change.



Important Considerations and Challenges

When it comes to building successful IoT implementations, security, development, app design and operation processes must be considered from the beginning. Ongoing security and privacy concerns must be forever managed. And the right infrastructure must be not only secure, but elastic and agile enough to deliver innovation for many years to come. Tactical mistakes in designing and deploying this architecture can create massive headaches that are costly to fix; indeed, they can be more costly than building the infrastructure properly the first time.

Given these important considerations, let's take a look at several challenges to address early on to successfully design and deploy a winning IoT initiative as based on our years of experience building cloud- based IoT architectures across industries and for companies of all sizes.

Infrastructure

Many enterprises try to build their IoT back-end systems on-premises, typically in a virtualized private cloud. They'll manage their virtual workloads, and manage the IoT platform and most of their network capacity in-house. There are significant challenges to this. The first is having to build enough capacity to handle the maximum potential bursts in usage that could arise. The networking and connectivity planning also have to be managed carefully.

Configuration and version management is another potential challenge in that they are needed to help ensure code meets and maintains security and operational requirements. If something were to break without the operational processes and supporting technology to manage and maintain version and configuration control, the question "what changed" would go unanswered, greatly impacting problem resolution and project success.

Additionally, failing to design the right IoT infrastructure from the beginning often underpins organizations having to make future, rushed decisions to compensate for inadequate design. For instance, not having adequate elasticity can spur additional, and harried, capacity plans to quickly overcome that constraint. Such rushed decisions typically lead to overspending and a fragmented

architecture.

Other types of mistakes that can be easily made once the infrastructure is off-course include developers realizing there is so much extra capacity in the data center that they are less rigorous in their work, and design oversized systems and deploy bloated code. Unnecessarily long, slow, and wasteful of resources, bloated code can create ongoing maintenance headaches that often result in missed deadlines and a loss of agility.

Vendor Confusion

The field of IoT vendors has expanded rapidly, and is continuing to grow daily. There are cloud infrastructure and platform providers that are making IoT solutions available. Networking equipment and appliance makers are doing the same, and there are a slew of dedicated point-product IoT vendors offering their tools as well. Gartner expects the IoT market to provide new market revenue of $300 billion by 2020. You should expect turbulence ahead as many of these smaller vendors are acquired and big standards battles are fought over time.

Connectivity

Network connectivity is also crucial. The local area wired and wireless networks must be correctly designed, and they must be able to scale their capacity for the amount of traffic from IoT devices that they'll be carrying. When thousands of endpoints are transmitting, utilization can become quite high. And as is most often the case today, connectivity must extend far from the local network and into fields, factory floors, offices, and even roadways. Enterprises need to design their network and applications so that latency never becomes a problem, and so that bandwidth is adequate and expandable.

Keeping Good Processes In Place

When it comes to IoT, maintaining quality development processes is crucial. Here, agile testing methods and approaches to IT management such as DevOps are essential. Not only do such approaches keep new apps and features moving forward at a brisk pace; they also ensure software quality stays high. Updating and fixing IoT software is often costlier and more cumbersome than fixing or, to avoid duplication, updating software on servers and traditional endpoints.

Security

All of the same challenges that apply to traditional software apps, servers, and endpoints apply to IoT devices and the data they generate. These devices can be breached, succumb to denial- of-service attacks, and the traffic can be snooped upon, copied, and even spoofed. None of those risks go away — in fact, the enterprise attack surface expands.

A successful IoT architecture can scale, and will offer a development pipeline that can provide for innovation, data analysis, and take whatever the future can throw at it.

IoT Security Threats And Challenges

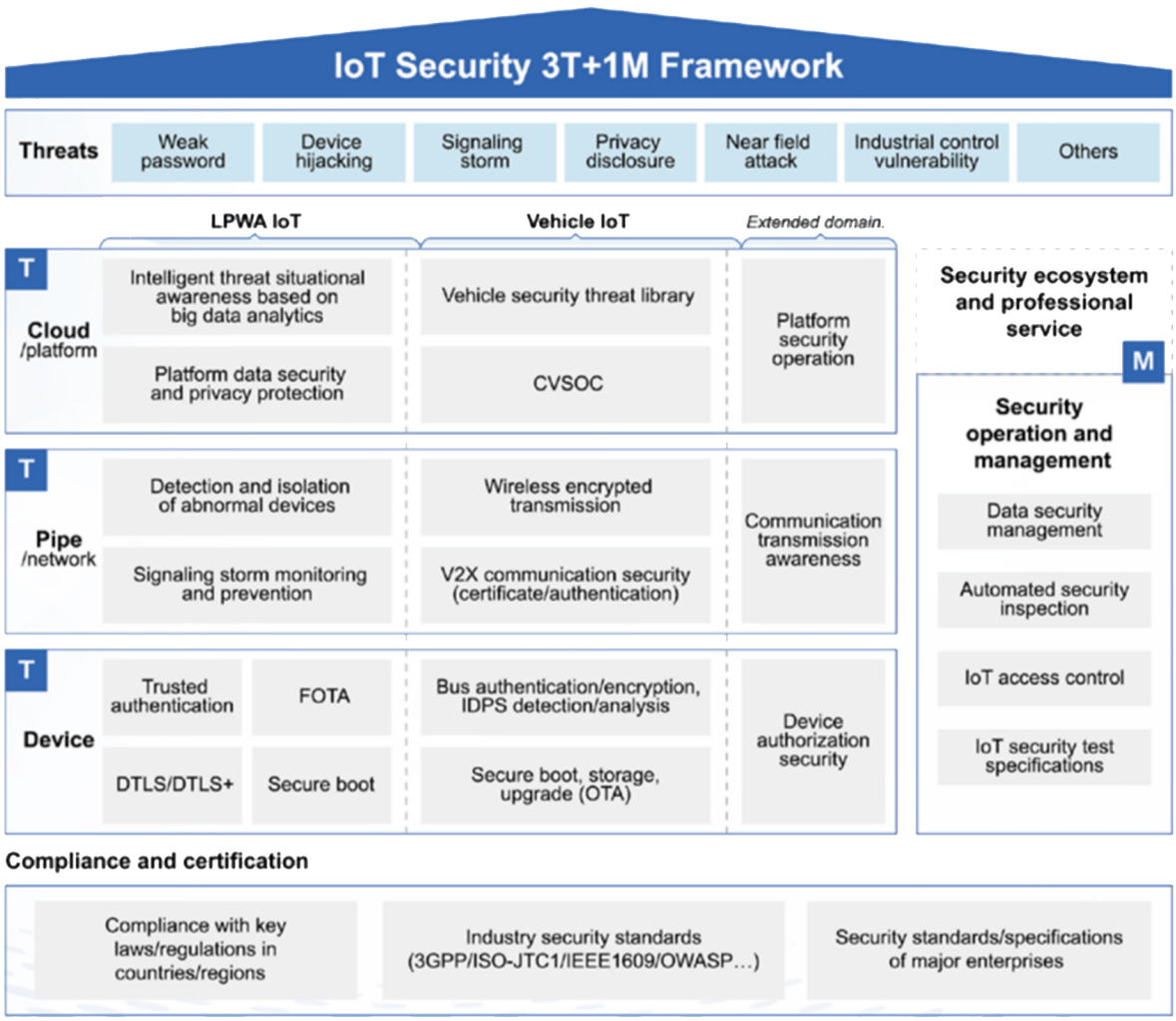
One of the many values of that IoT is that it is driving the digitization of all industries; however , the IoT brings with it new security threats dues to new technology applications.

As the tools used in attacks become more sophisticated, Machine Learning (ML) and Artificial Intelligence (AI) will compound attack -defense confrontation. Although AI can be used to rapidly detect new security threats, it can also be used to launch attacks. The technical barriers for implementing attacks become lower. loT devices, including refrigerators, vacuum cleaners, water meters, and street lights, will become potential targets for attack. By 2020, Gartner predicts that more than 25 percent of identified attac-ks in enterprises will involve the 1012. In addition, devices at different physical locations and network layers are connected to each other, breaking up the boundaries of traditional network security and generating more attack vectors. Attacks can be launched at different locations to target different layers, creating a springboard effect where attackers can leverage small vulnerabilities to open up larger ones. Of note is that attackers have transformed the way in which they launch attacks. Attacks that were once launched on vulnerable devices are now being launched on legitimate devices. Attackers now use automation tools to simulate authorized operations on legitimate devices, which are then exploited as a springboard to launch attacks.

As the loT enters a more pragmatic and operational phase, industry customers are aware of the importance of loT security. Gartner predicts that worldwide spending on loT security will reach USS1.506 billion in 2018, a 28 percent increase' over the US $1.174 billion spent in 2017. Different commercial sectors face a wide range of different threats. For example, the Internet of Vehicles (IW) may face completely different threats and security challenges compared to those facing intelligent street lighting. loT security needs to move from single products to end-to-end solutions and eventually to the entire security architecture. The evolving security architecture is used in future business scenarios, such as Smart City, smart energy. smart transportation, smart manufacturing/industry, smart life and autonomous driving.

loT security is involved in Low Power Wide Area (LPWA) networking, the loV, industrial loT, wearable devices, and other industries. In the loT ecosystem, numerous loT devices generate and use massive amounts of data. The pipe ensures transmission security

of highly concurrent data, and the cloud and IoT platform provide support for a wide range of IoT applications. These support systems and applications may become potential targets of malicious attacks. The 3T+1 M security architecture focuses on the security features of the device, pipe, cloud, and platform to address the security threats at the sensor, network, and application layers in the IoT.



Source: Huawei IoT Security Whitepaper 2018 - Evolving Security Architecture

Security-in-depth strategy can be developed and executed with active participation of various players involved with the manufacturing, development, and deployment of IoT devices and infrastructure. Following is a high-level description of these players.

IoT Hardware Manufacturer/Integrator: Typically, these players are the manufacturers of IoT hardware being deployed, integrators assembling hardware from various manufacturers, or suppliers providing hardware for an IoT deployment manufactured or integrated by other suppliers.

1. **Scope hardware to minimum requirements:** The hardware design should include the minimum features required for operation of the hardware, and nothing more. An example is to include USB ports only if necessary for the operation of the device. These additional features open the device for unwanted attack vectors that should be avoided.

2. **Make hardware tamper proof:** Build in mechanisms to detect physical tampering, such as opening of the device cover or removing a part of the device. These tamper signals may be part of the data stream uploaded to the cloud, which could alert operators of these events.

3. **Build around secure hardware:** Build security features such as secure and encrypted storage, or boot functionality based on Trusted Platform Modules (TPM). These features make devices more secure and help protect the overall IoT infrastructure.

4. **Make upgrades secure:** Firmware upgrades during the lifetime of the device are inevitable. Building devices with secure paths for upgrades and cryptographic assurance of firmware versions will allow the device to be secure during and after upgrades.

IoT Solution Developer: The development of an IoT solution is typically done by a solution developer. This developer may be part of an in-house team or a system integrator (SI) specializing in this activity. The IoT solution developer can develop various components of the IoT solution from scratch, integrate various off-the-shelf or open-source components, or adopt solution accelerators with minor adaptation.

1. **Follow secure software development methodology:** Development of secure software requires ground-up thinking about security: from the inception of the project all the way to its implementation, testing, and deployment. The choices of platforms, languages, and tools are all influenced with this methodology.

2. **Choose open-source software with care:** Open-source software provides an opportunity to quickly develop solutions. When you're choosing open-source software, consider the activity level of the community for each open-source component. An active community ensures that software is supported and that issues are discovered and addressed. Alternatively, an obscure and inactive open-source software project might not be supported and issues are not likely to be discovered.

3. **Integrate with care:** Many software security flaws exist at the boundary of libraries and APIs. Functionality that may not be required for the current deployment might still be available via an API layer. To ensure overall security, make sure to check all interfaces of components being integrated for security flaws.

IoT Solution Deployer: After an IoT solution is developed, it needs to be deployed in the field. This process involves deployment of hardware, the interconnection of devices, and deployment of solutions in hardware devices or to the cloud.

1. **Deploy hardware securely:** IoT deployments may require hardware to be deployed in unsecure locations, such as in public spaces or unsupervised locales. In such situations, ensure that hardware deployment is tamper-proof to the maximum extent. If USB or other ports are available on the hardware, ensure that they are covered securely. Many attack vectors can use these as entry points.

2. **Keep authentication keys safe:** During deployment, each device requires device IDs and associated authentication keys generated by the cloud service. Keep these keys physically safe even after the deployment. Any compromised key can be used by a malicious device to masquerade as an existing device.

IoT Solution Operator: After the IoT solution is deployed, it requires long-term operations monitoring, upgrades, and maintenance. These tasks can be done by an in-house team that is comprised of information technology specialists, hardware operati-

ons and maintenance teams, as well as domain specialists who monitor the correct behavior of the overall IoT infrastructure.

1. **Keep the system up-to-date:** Ensure that device operating systems and all device drivers are upgraded to the latest versions. Keeping operating systems up-to-date helps ensure that they are also protected against malicious attacks.

2. **Protect against malicious activity:** If the operating system permits, install the latest anti-virus and anti-malware capabilities on each device operating system. This practice can help mitigate most external threats. You can protect most modern operating systems against threats by taking the appropriate steps.

3. **Audit frequently:** Auditing IoT infrastructure for security-related issues is key when responding to security incidents. Most operating systems provide built-in event logging that should be reviewed frequently to make sure no security breach has occurred. Audit information can be sent as a separate telemetry stream to the cloud service where it can be analysed.

4. **Physically protect the IoT infrastructure:** The worst security attacks against IoT infrastructure are launched using physical access to devices. One important safety practice is to protect against malicious use of USB ports and other physical access. One key to uncovering breaches that might have occurred is the logging of physical access, such as USB port use.

5. **Protect cloud credentials:** Cloud authentication credentials used for configuring and operating an IoT deployment are possibly the easiest way to gain access and compromise an IoT system. Protect these credentials by changing the password frequently, and refrain from using these credentials on public machines.

IMPLEMENT "SECURITY BY DESIGN"

To achieve end-to-end security within an IoT solution, security must be a priority across all of the layers of your IoT architecture. You need to think about security as a cross-cutting concern in your IoT architecture, rather than as a separate layer of your IoT architecture to be tackled in isolation at the end. With so many devices connected, the integrity of the system as a whole needs to be maintained even when individual devices or gateways are compromised. Ensure that your architecture supports multiple layers of defense. Also, ensure that your IoT solution can identify and neutralize devices that become compromised, such as by using gateways to isolate vulnerable devices and to monitor communications and usage patterns to detect anomalies.

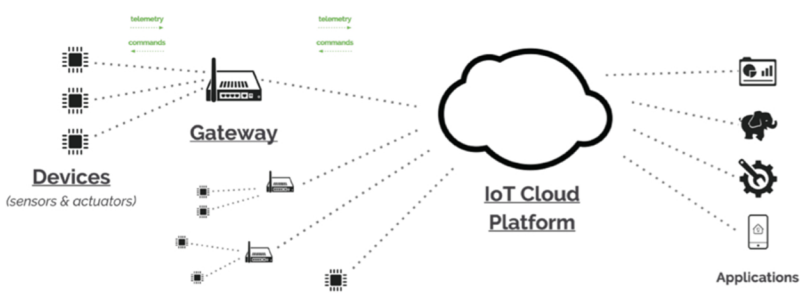
You should adopt standards and best practices for these aspects of your IoT infrastructure:

- Device, application and user identity, authentication, authorization, and access control
- Key management
- Data security
- Secure communication channels and message integrity (by using encryption)
- Auditing
- Secure development and delivery

IoT Architectures

A typical IoT solution is characterized by many devices (i.e. things) that may use some form of gateway to communicate through a network to an enterprise back-end server that is running an IoT platform that helps integrate the IoT information into the existing enterprise. The roles of the devices, gateways, and cloud platform are well defined, and each of them provides specific features and functionality required by any robust IoT solution.

One of the biggest challenges you face when you are planning Internet of Things (IoT) solutions is dealing with complexity. A typical IoT solution involves many heterogeneous IoT devices, with sensors that produce data that is then analyzed to gain insights. IoT devices are connected either directly to a network or through a gateway device to a network, which enables the devices to communicate with each other and with cloud services and applications.

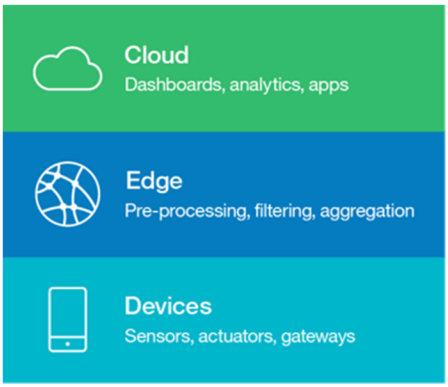


ADOPT A LAYERED ARCHITECTURE

An architecture describes the structure of your IoT solution, including the physical aspects (that is, the things) and the virtual aspects (like services and communication protocols). Adopting a multi-tiered architecture allows you to focus on improving your understanding about how all of the most important aspects of the architecture operate independently before you integrate them within your IoT application. This modular approach helps to manage the complexity of IoT solutions.

For data-driven IoT applications that involve edge analytics, a basic three-tiered architecture, captures the flow of information from devices, to edge services, and then out to cloud services. A more detailed IoT architecture would also include vertical layers that cut across the other layers, like identity management or data security.

1. **Devices layer:** The components in the device layer include physical sensors and actuators that are connected to IoT devices and the IoT devices themselves. Sensors and actuators on their own are typically not considered to be "smart" devices, but sensors and actuators often connect either directly or wirelessly over technologies like Bluetooth LE or ZigBee to IoT devices that have more processing capabilities.



   Some IoT devices communicate directly with related cloud services and apps. However, it is common for IoT devices to communicate upstream through gateways, which are intermediate devices that have slightly more processing power than the basic IoT devices. Although they don't always have sensors attached directly, gateway devices play an important role in the data acquisition process. They can perform basic analog-to-digital conversions, scaling, and other normalization of the raw sensor data readings.

2. **Edge layer:** The Edge layer (shown as the middle tier of Figure 2) relates to the analytics and pre-processing services that are located at the edge of the network. Edge analytics occurs in real time (or near real time) by processing the stream of data at the point where the data is collected as it comes in from the sensors. Basic pre-processing tasks like filtering and aggregation of data are performed at the edge, and then key pre-processed data is transferred upstream to cloud services and applications for further processing and analytics.

3. **Cloud layer:** After the data has been prepared, it is sent upstream for further processing, storage, and use within cloud applications, in the cloud layer (shown as the top tier of Figure 2). The cloud applications that perform the data processing are often complemented by mobile apps and web-based client applications that present the data to end users and that provide access to tools for further exploration and analysis through dashboards and visualizations.

STACK FOR CONSTRAINED DEVICES » SENSORS AND ACTUATORS

The "Thing" in the IoT is the starting point for an IoT solution It is typically the originator of the data, and it interacts with the physical world Things are often very constrained in terms of size or power supply; therefore, they are often programmed using microcontrollers (MCU) that have very limited capabilities The microcontrollers powering IoT devices are specialized for a specific task and are designed for mass production and low cost. The software running on MCU-based devices aims at supporting specific tasks. The key features of the software stack running on a device may include:

1. **IoT Operating System:** Many devices will run with 'bare metal', but some will have embedded or real-time operating systems that are particularly suited for small constrained devices, and that can provide IoT-specific capabilities.

Key Characteristics for IoT Stacks

Loosely coupled: Three IoT stacks have been defined but it is important that each stack can be used independently of the other stacks It should be possible to use an IoT Cloud Platform from one supplier with an IoT gateway from another supplier and a third supplier for the device stack

Modular: Each stack should allow for the features to be sourced from different suppliers.

Platform-Independent: Each stack should be independent of the host hardware and cloud infrastructure For instance, the device stack should be available on multiple MCUs and the IoT Cloud Platform should run on different Cloud PaaS.

Based on open standards: Communication between the stacks should be based on open standards to ensure interoperability

Defined APIs: Each stack should have defined APIs that allow for easy integration with existing applications and integration with other IoT solutions

2. **Hardware Abstraction**: A software layer that enables access to the hardware features of the MCU, such as flash memory, GPIOs, serial interfaces, etc
3. **Communication Support:** Drivers and protocols allowing to connect the device to a wired or wireless protocol like Bluetooth, Z-Wave, Thread, CAn bus, MQTT, CoAP, etc , and enabling device communication

4. **Remote Management:** The ability to remotely control the device to upgrade its firmware or to monitor its battery level.

STACK FOR GATEWAYS » CONNECTED AND SMART THINGS

The IoT gateway acts as the aggregation point for a group of sen- sors and actuators to coordinate the connectivity of these devices to each other and to an external network An IoT gateway can be a physical piece of hardware or functionality that is incorporated into a larger "Thing" that is connected to the network For example, an industrial machine might act like a gateway, and so might a connected automobile or a home automation appliance.

An IoT gateway will often offer processing of the data "at the edge" and storage capabilities to deal with network latency and reliability For device to device connectivity, an IoT gateway deals with the interoperability issues between incompatible devices A typical IoT architecture would have many IoT gateways supporting masses of devices .

IoT gateways are becoming increasingly dependant on software to implement the core functionality The key features of a gateway software stack include :

1. **Operating System:** Typically a general purpose operating system such as Linux.

2. **Application Container or Runtime Environment:** IoT gateways will often have the ability to run application code, and to allow the applications to be dynamically updated For example, a gateway may have support for Java, Python, or node js.

3. **Communication and Connectivity:** IoT gateways need to support different connectivity protocols to connect with different devices (e.g. Bluetooth, Wi-Fi, Z-Wave, ZigBee, Thread). IoT gateways also need to connect to different types of networks (e g Ethernet, cellular, Wi-Fi, satellite, etc ...) and ensure the reliability, security, and confidentiality of the communications.

4. **Data Management & Messaging:** Local persistence to support network latency, offline mode, and real-time analyt- ics at the edge, as well as the ability to forward device data in a consistent manner to an IoT Platform.

5. **Remote Management:** The ability to remotely provision, configure, startup/shutdown gateways as well as the applications running on the gateways.

STACK FOR IOT CLOUD PLATFORMS

The IoT Cloud Platform represents the software infrastructure and services required to enable an IoT solution An IoT Cloud Platform typically operates on a cloud infrastructure (e g OpenShift, AWS, Microsoft Azure, Cloud Foundry) or inside an enterprise data center and is expected to scale both horizontally, to support the large number of devices connected, as well as vertically to address the variety of IoT solutions The IoT Cloud Platform will facilitate the interoperability of the IoT solution with existing enterprise applications and other IoT solutions. The core features of an IoT Cloud Platform include :

1. **Connectivity and Message Routing:** IoT platforms need to be able to interact with very large numbers of devices and gatew- ays using different protocols and data formats, but then normalize it to allow for easy integration into the rest of the enterprise.

2. **Device Management and Device Registry:** A central registry to identify the devices/gateways running in an IoT solution and the ability to provision new software updates and manage the devices.

3. **Data Management and Storage:** A scalable data store that supports the volume and variety of IoT data.

4. **Event Management, Analytics & UI:** Scalable event pro- cessing capabilities, ability to consolidate and analyze data, and to create reports, graphs, and dashboards.

5. **Application Enablement:** Ability to create reports, graphs, dashboards, ... and to use API for application integration.

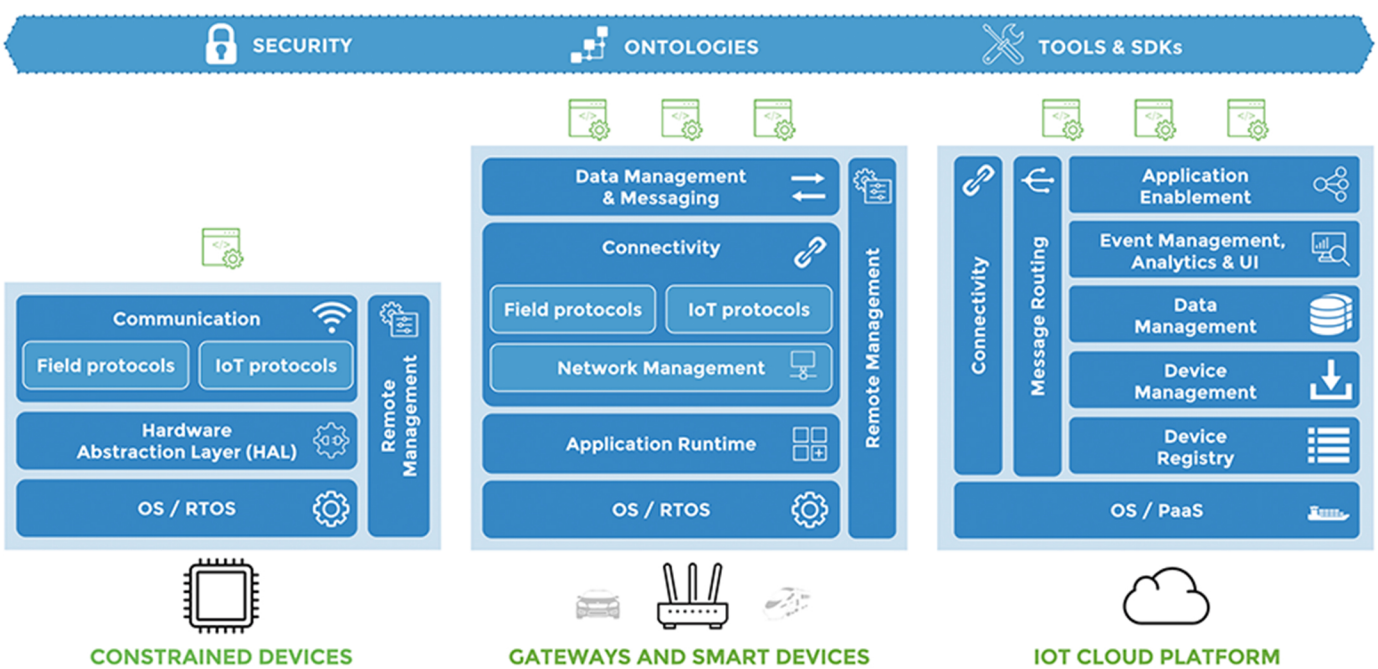
CROSS-STACK FUNCTIONALITY

Across the different stacks of an IoT solution are a number of features that need to be considered for any IoT architecture, including:

1. **Security:** Security needs to be implemented from the devices to the cloud Features such as authentication, encryp- tion, and authorization need be part of each stack.

2. **Ontologies:** The format and description of device data is an important feature to enable data analytics and data interoperabi- lity. The ability to define ontologies and meta- data across heterogeneous domains is a key area for IoT.

3. **Development Tools and SDKs:** IoT Developers will require development tools that support the different hardware and software platforms involved.



Source: The three stacks required for IoT Architecture - Eclipse Foundation

Pegasus One is a privately held, professional IT services company and a **Microsoft** and **Amazon Partner** with its U.S. headquarters in the Southern California and development offices in Mexico and India. We provide custom software development and consulting services in various technologies including Microsoft Solutions, IoT, Mobile and BI. We are a CMMI Level 5 company with high focus on quality and ROI.

email: info@pegasusone.com
phone: +1 (714) 485-8104
corporate office: 1440 N Harbor Blvd #900, Fullerton, CA, 92835
sales office: 10880 Wilshire Blvd., # 1101, Los Angeles, CA, 90024

ISO 9001 QUALITY ASSURANCE

Agile Alliance

amazon web services | Partner Network
CONSULTING PARTNER

Microsoft CERTIFIED
Partner