Key Safety Challenges for the IIoT

An Industrial Internet Consortium Technical White Paper

IIC:WHT:IN6:V1.0:PB:20171201

The Industrial Internet is an internet of things, machines, computers and people. Industrial Internet of Things (IIoT) systems connect traditional Operational Technologies (e.g. industrial control systems) with people and traditional Information Technology based enterprise systems, forming larger end-to-end systems. This paradigm shift in traditional industrial control system operations is enabled by the proliferation of low-cost smart devices, ubiquitous network connectivity, rich computing and processing resources and advanced data analytics. These emerging technology advancements have brought transformational business opportunities, shaping sectors such as energy, healthcare, manufacturing, transportation, defense and public sectors both horizontally and vertically.

The trustworthiness of an IIoT ecosystem is essential for the confidence necessary to adopt and use the system. The Industrial Internet Consortium (IIC) defines *trustworthiness* as "the degree of confidence one has that the system performs as expected with characteristics including safety, security, privacy, reliability and resilience in the face of environmental disruptions, human errors, system faults and attacks."[1]

Safety is a critical aspect of trustworthiness and a major concern in many IIoT systems. *Safety* is defined as "the condition of the system operating without causing unacceptable risk of physical injury or of damage to the health of people, either directly, or indirectly as a result of damage to property or to the environment". An increasing number of devices and systems combine hardware, software and connectivity to sense and control the physical world in public spaces, factories, offices and homes. Many of these systems could cause harm to humans, animals or the environment if they did not have designed-in safety mechanisms that mitigate potential risks to a tolerable level. Harm in modern connected systems can result not only from unintentional system defects and random failures, but also from intentional manipulation of the system by a malicious adversary.

While different industrial sectors have long-established approaches to safety, the approaches and standards are still evolving to address new and unique safety challenges that IIoT brings. This white paper articulates four key challenges unique to the IIoT that affect safety characteristics:

- increased security risks due to an increased attack surface,
- convergence of IT and OT,
- pervasive autonomy and
- inadequate regulatory framework and evolving standards.

[1] Industrial Internet Consortium: "The Industrial Internet, Volume G8: Vocabulary Technical Report, Version 2.0", 2017, *http://www.iiconsortium.org/vocab/*

The community must address these concerns for the IIoT to achieve its full potential. We hope to involve stakeholders in understanding the issues and working toward potential solutions, including the adoption of standards under development in ISO/IEC and elsewhere.

This white paper is not exhaustive. Rather, we highlight four key challenges, explain why the current safety frameworks and approaches are inadequate and recommend how the greater IIoT community should address them. Some recommendations may appear to be contradictory. Tradeoffs must be made depending on the context and business goals of the organizations making them.

CHALLENGE 1: INCREASED SECURITY RISKS DUE TO INCREASED ATTACK SURFACE

Security risks related to an increased attack surface expand the safety challenge in IIoT systems. A principal aspect of IIoT is the increased connectivity relative to existing safety-critical systems. At the macro level, demand for more data and linking existing systems into systems-of-systems will increase connectivity, and connectivity requirements will be pushed down to the individual sensors and actuators to close the loop automatically. This increase in connectivity at every level of the system leads to a much larger attack surface that adversaries could potentially exploit to remotely cause unsafe system behavior. Moreover, IIoT systems are becoming more dynamic than traditional safety-critical systems, with participation of many organizations in the management of systems, with access rights assigned across organizations and changing over time. The blurring of traditional IT boundaries between internal and external systems increases risks because policing system boundaries will become more difficult.

Example

Example 1: Robot Safety in the Factory

A robot arm takes product out of the boxes supplied by a human operator and lays them on a conveyor belt where a second robot picks them up and inspects them with a vision camera before they are passed onto a packaging system. To avoid hurting the operator, the robot needs to move with Safe Limited Torque (SLT) and Safe Limited Speed (SLS) to guarantee the robot does not hurt the human operator should they collide.

The system uses light curtains to notify the Safety Programmable Logic Controller (PLC) to put drives into a lower safety-limited torque and speed where collision is more likely.

A specific challenge is to have reliable and secure safety protocols that are appropriate to threats that challenge the safety-critical components on the network, i.e., Safety PLC, Light Curtain, Safety Drives, Safety Emergency Stop, etc. The safety protocol functionality must be verified and validated rigorously to reduce the risks of safety-critical components becoming subverted via their network interfaces. Manufacturing systems currently have safety protocol standards such

as CIP Safety[2] (defined and maintained by ODVA, an IEC standard), and PROFISafe®[3] (also an IEC standard). These standards consider some security aspects. For example, CIP Safety has included security transport using Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS) since 2015. The real challenge is in getting the security aspects deployed in products and in the end user systems–which has not happened to date.

Depending on the intended use, operational environment, threats and criticality of the system, the system will need to meet a specified safety integrity level as well as certain security requirements. The components on the safety network must be designed accordingly to support the overall system requirements. Different levels of certification may be appropriate, ranging from manufacturer self certification to certification by third parties.

Example

Example 2: Building Steam Heating System

The steam source for each building is generated by an array of boilers. The temperature and pressure of each boiler are maintained by a feedback loop, from sensors to a controller implementing control algorithms, thence to actuators on a gas burner to input heat energy.

To optimize energy usage and cut costs, analytics predict heating needs and energy costs in real-time and manipulate the set points of the controller(s) on the boilers.

This scenario opens a number of risks, including the risk that a message to the actuator could be out of range and result in driving the boiler to too high a temperature, causing the boiler to explode. This could be the result of a system fault or a malicious attack, such as an attacker reconfiguring or reprogramming the connected system.

Alternatively, causing the temperature to go too low could cause pipes to freeze, also catastrophic, if less spectacular. Failsafe mechanisms are needed, in case of communication, messaging or algorithm failures. The system should continue to operate safely when the feedback loop is not functioning properly, even if Internet or network connectivity fails. The IEC 61508 Basic functional safety standard employs the 'zero-current principle' for functional safety communications. This means that if a channel fails, endpoints will act to take the system to a safe state, though an attacker could use this feature to generate a ‘denial of service’ attack.

This example shows how extending the system boundary increases difficulties for safety and security protection. Real examples include:

[2] *https://www.odva.org/Portals/0/Library/Publications_Numbered/PUB00110R1_CIP_Safety_White_Paper.pdf*

[3] *https://profibusgroup.files.wordpress.com/2014/03/c11-profisafe-considerations-for-functional-safety-and-security-pete-brown-siemens.pdf*

- Ohio's Davis-Besse nuclear power plant Slammer incident, when a worm came to a nuclear power plant internal network from a consultant's computer through a T1 line.[4]
- U.S.-retailer Target attack whereby the information network was attacked through a 3rd party contractor that provided HVAC services to Target[5].
- Maroochy Water Services insider incident.[6]

The increase of the networked integration of systems and the increasing ability of adversaries to conduct attacks over the Internet requires a new view of security in safety-critical systems designed to meet stringent safety requirements. Not connecting systems to the Internet ("air-gapping") reduces risk, as do firewalls. But the Stuxnet worm that was introduced through a USB memory device shows that air-gapping is not a perfect protection technique in today's world.[7]

IIoT stakeholders must be prepared to implement comprehensive security solutions at each level, from the system of systems down to the individual sensor or actuator. The IIC's *Industrial Internet Security Framework*[8] provides plenty of advice on this topic.

CHALLENGE 2: IT/OT CONVERGENCE

IIoT is driving tighter integration between Information Technology (IT) and Operational Technology (OT). IT assets include the enterprise network/information bus, database services, analytics engines and web services. OT assets include the technology of real-time networks (e.g., industrial Ethernet), programmable logic controllers (PLCs), sensors and actuators.

Integration between IT and OT implies not only physical convergence but also convergence of expectations and mentalities. Many attributes we typically associate with IT systems will start to be associated with OT systems and vice versa. Physical convergence involves hosting both OT and IT functions on the same platform. For example, using the same network to host the enterprise information bus and real-time control signals could reduce costs.

Likewise, the safety-critical nature of OT systems has shaped how they are engineered and maintained. Safety-critical systems should be developed top-down using rigorous processes for design and implementation through verification and validation. Requirements should be comprehensive, well documented and traceable through each stage of development. Figure 1 shows the typical development model for safety-critical systems.

[4] *http://www.securityfocus.com/news/6767*
[5] *http://securityintelligence.com/target-breach-protect-against-similar-attacks-retailers/#.VMA24UfF-So*
[6] *https://slidedocument.org/maroochy-water-services-case-study-briefing*
https://www.mitre.org/sites/default/files/pdf/08_1145.pdf
[7] *https://securelist.com/analysis/publications/67483/stuxnet-zero-victims/*
[8] *http://www.iiconsortium.org/IISF.htm*

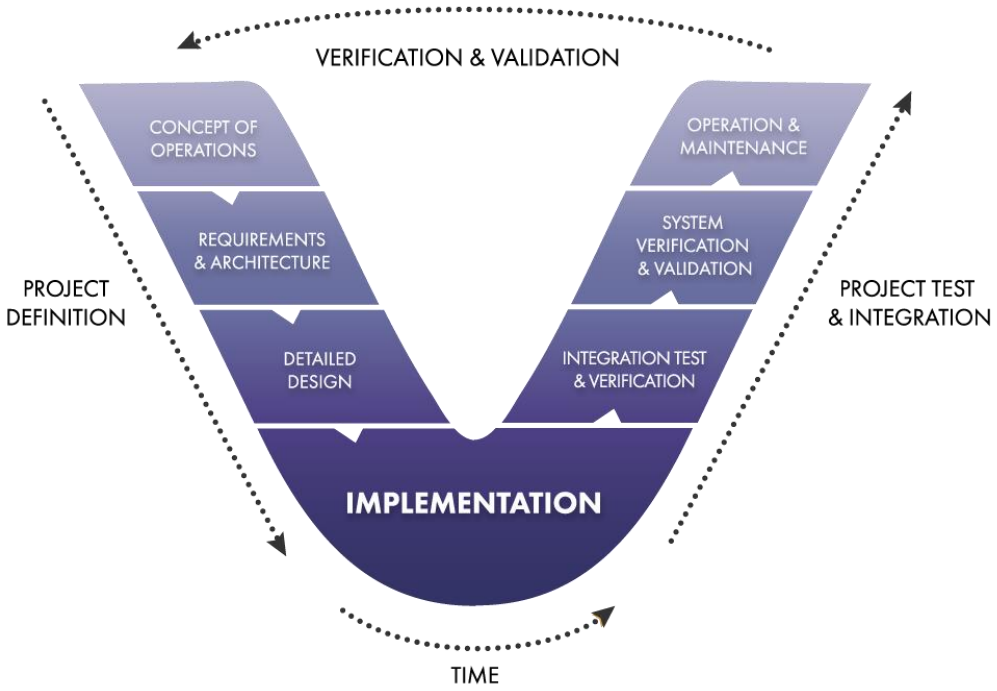

Figure 1: The standard "V" system development model[9]

Moreover, systems designers should explicitly assign safety responsibilities to each system component and consider the control actions a system must implement to actively avoid unsafe situations.[10] For example, IIoT designers should engineer their systems to actively avoid dangerous fault conditions as opposed to simply relying on reliability calculations and failure rates to ensure safe behavior. Due to the increasing complexity of IIoT systems, products cannot rely solely on verification and validation testing to find and remove faults. That said, verification and validation is also important since no system is perfect.

Verification and validation of safety-critical software usually requires extensive testing (for example, the Level A[11] objective of DO-178C[12] requires full branch coverage) and possibly formalized correctness proofs. Moreover, maintenance of OT systems is conservative relative to their IT counterparts. Many OT systems remain relatively static once deployed because any significant change to the system configuration, software or other functions will require new comprehensive verification and validation efforts.

[9] This figure is derived from one from Wikipedia, see *https://en.wikipedia.org/wiki/V-Model*

[10] *http://sunnyday.mit.edu/accidents/safetyscience-single.pdf*

[11] Wikipedia: https://en.wikipedia.org/wiki/DO-178C

[12] RTCA DO-178C, "Software Considerations in Airborne Systems and Equipment Certification," December 2011, https://my.rtca.org/NC__Product?id=a1B36000001IcmqEAC

In contrast, IT systems are usually developed faster with less rigorous processes than their OT counterparts. They often have less rigorously defined requirements, and little or no traceability back from the implementation. Unlike OT systems that can remain unchanged for decades, IT system stakeholders expect more frequent changes to meet evolving requirements. In addition, IT stakeholders accept the need for patches to correct security vulnerabilities and other issues while OT systems require stability due to the costs associated with safety certification. These differing expectations and experiences of the IT and OT communities are a challenge for systems that combine IT and OT.

The engineers responsible for developing OT are often different from the engineers responsible for IT development. Relatively few have significant experience in developing both. Yet both perspectives should be considered together to create a trustworthy and safe system, requiring organizational changes to increase communication, cooperation and understanding.

Managing the tradeoffs due to conflicting requirements and conflicting choices is a challenge. Inconsistencies can arise from the competing demands of security and safety. For example, a door control system should keep all doors in an open state from a fire-safety vantage point but access to some of the doors should be restricted from a security vantage point. This illustrates the need to approach safety and security together at a system level in addition to the component level.

Organizations must be prepared to address the security challenges due to IT/OT convergence that affect safety. First, organizations undergoing IT/OT convergence should attempt, wherever possible, to enforce the non-interference of IT and OT elements that share computing and communications platforms. The IEC 61508 functional safety standard uses the term “non-interference” or “independence” to include *logical separation* while considering both “spatial” and “temporal” aspects. Two components are logically separated if it is impossible for one component to affect the operation of another, even if they are sharing a resource (such as a network, CPU or memory). For example, a *separation kernel* is a type of thin operating system (usually a hypervisor) that ensures software running in one of its partitions cannot interact with software running in another. Separation kernels are simple enough (relative to mainstream operating systems) that their separation capabilities can be rigorously (in some case formally) verified. Another example of logical separation for IT and OT network segments is provided in “ICS Network Architectures”.[13]

[13] “Recommended Practice: Improving Industrial Control System Cybersecurity with Defense-in-Depth Strategies” Industrial Control Systems Cyber Emergency Response Team, Homeland Security, September 2016, https://ics-cert.us-cert.gov/sites/default/files/recommended_practices/NCCIC_ICS-CERT_Defense_in_Depth_2016_S508C.pdf

Second, manufacturers of safety-critical system components should investigate (and be prepared to implement) the types of "IT-like" capabilities users will come to expect, such as firmware updates via the network of their OT systems, while still ensuring safety.

Third, vendors of equipment and software with an IT legacy who want to participate in the IIoT community should familiarize themselves with how safety critical software and hardware is developed, from requirements through validation and verification.

Fourth, an organization should define areas of responsibility and ways of interaction between OT and IT specialists. For example, a computer security incident response team in an IIoT system should include OT and IT specialists.

CHALLENGE 3: PERVASIVE AUTONOMY

Autonomy is the ability of the system to make its own decisions with regards to external inputs and its changing environment and to be able to continue to operate even if disconnected from the network and remote analytics. Typically, autonomous systems are given high-level directives by human stakeholders (e.g., commanding a self-driving car to drive to a certain address), while the low-level behavior directives used to satisfy the command are generated by the system itself with little or no human intervention. Autonomy has the potential to increase efficiency by optimizing resource usage and enabling systems to react to changes in the environment faster than what would be possible if they relied on human input.

Autonomy presents at least two safety challenges that we examine further. First, autonomy changes how safety responsibility is divided between human operators and the system. Second, sophisticated autonomy typically requires responding to dynamically changing circumstances and often involves the application of machine learning and artificial intelligence techniques that will themselves present verification challenges. Artificial Intelligence and machine learning is not easily quantified in terms of fault analysis for a Failure Mode and Effects Analysis (FMEA).

How does shifting responsibilities from humans to the machine present a safety challenge? Human operators help maintain safety by detecting an impending unsafe situation and driving the system outside of its envelope or violating some normative rule. For example, if a driver notices that another car in front of them is behaving dangerously, they may judge the safest option is to violate traffic law (such as illegally cross into another lane) to avoid an accident.

Historically, because human operators have had the capability to make (and apply) these sorts of judgments, systems designers have not had to specify safety judgments as part of the system's requirements. However, as machines take on more decision-making responsibilities, their stakeholders must be able to define the appropriate judgments and tradeoffs, incorporate them into the systems requirements, and figure out how to ensure that systems will faithfully satisfy those directives.

How do current artificial intelligence and machine learning technologies present verification challenges? The behavior of autonomous systems will be driven by software, but the type of software used is not amenable to traditional verification approaches. Consider, for example, so-called "deep learning" (a buzzword for systems built around very large artificial neural networks (ANNs) systems that have recently been demonstrated to be successful at computer vision, natural language and digital signal processing tasks). Each artificial neuron in an ANN is a simple weighted activation function: If the input to the neuron triggers the activation function, that neuron will fire an output and propagate a signal to any neurons connected to that output. ANNs are trained by presenting the network with a large number (usually millions or more) of labeled training examples (e.g., images annotated with the objects present in the image). Learning algorithms then iteratively adjust the weights in the ANN until the network's accuracy (i.e., its ability to correctly pair an input with the correct label) is maximized.

Deep learning itself has vulnerabilities and is subject to attacks since assumptions are often made in the models on which it is based. If security and safety assumptions are incorrect, this can invalidate the learning model. For example, a system could describe attacker behavior with a state machine assuming an attacker will first "scan for vulnerabilities", then leverage these vulnerabilities as part of an attack, and finally exploit the system for other ends (e.g., to extract information, damage a machine or ask for ransom). If an intrusion detection system assumes that the first step will always be a scan, then when an attacker does not start with the expected scan, the attack may not be detected by the security monitoring system. Another example of a deep learning assumption is that training is for good purposes, yet it is possible for an attacker to train the system to learn inappropriate behavior. Assumptions must always be considered from a safety and security viewpoint.

Validation and verification are appropriate for deep learning systems too, but it is paramount to avoid faults first by reducing complexity and systematic errors by appropriate requirements and design activities. Appropriate design work should be combined with a high degree of verification coverage. Although it may be fairly straightforward to achieve high coverage for deep learning software using traditional metrics like branch coverage, traditional coverage metrics at the unit level do not relate well to the emergent behavior of the software system when that system comprises millions (or even billions) of artificial neurons. Indeed, many artificial intelligence and machine learning techniques result in autonomous software systems that are effectively black boxes to the human developer. How do we ensure that these complex pieces of software will produce safe behavior?

To meet the first challenge, the stakeholders of autonomous IIoT systems must engage with one another and come to a consensus on which safety judgments and tradeoffs are appropriate for the autonomous system to make on its own. To meet the second challenge, the IIoT community must invest in research and development for verification of autonomous systems.

CHALLENGE 4: INADEQUATE REGULATORY FRAMEWORKS AND EVOLVING STANDARDS

Many safety-related standards and regulatory frameworks were designed over many decades ago and are still applicable to IIoT systems in many aspects. Now they face the challenge of the increasing number of components and interfaces in an IIoT system. Various standards and regulatory bodies currently take into account important security-related considerations and how they impact safety regulation and compliance in IIoT. Both of the domains of safety and security must be considered. IIoT systems should take into account well proven and tested safety standards for the components and interfaces, as well as security standards.

A focus on safety for industrial machines has become more prevalent and there are well-established standards for how network and machine safety should be implemented. The standards defined for the safety of industrial systems vary greatly across vertical markets such as discrete Industrial Automation, Process Control, Automotive and Aerospace (e.g., IEC61508 is a basic safety standard used to derive industry-specific standards such as IEC62061, ISO13849-1 for Automation, ISO26262 for Automotive). As an example, industrial machine drives must adhere to various safety-motion profiles that have clear requirements on specific functions: safety-limited speed, safe direction, safety-limited torque and so on.

Due to the importance of security in general, security regulation is also important to IIoT systems. Security standards for industrial applications already exist (e.g., IEC 62443 series) and are evolving. While the relationship of safety and security standards must be considered, each of these domains still has primary responsibility for the relevant core issues.

One important desired capability of IIoT system components is plug & play interoperability. The goal of plug & play interoperability is to enable systems operators to assemble and integrate a new system for use quickly. For example, a medical provider could combine a set of medical sensors, actuators and control algorithms on the cloud to automate the delivery of certain therapies.

While safe systems can be created by combining plug & play components appropriately, scaling existing certification processes is a challenge because these frameworks are not explicitly designed with plug & play interoperability in mind. The current best practice is to use protocols designed with safety certification in mind (e.g., IEC61508 CIP Safety, ProfiSafe), and then also certify/regulate the entire system. This process adds time and extends the duration of project lifecycles, negatively impacting manufacturers' product time-to-market (i.e., first production run capability).

Just as current engineering processes approach safety, and safety verification and validation (V&V) from the perspective of integrated systems, current regulatory frameworks are set up for whole systems, or sets of specific component combinations and system configurations (as shown in Figure 2). In contrast, safety certification organizations (like UL, TÜV SÜD) have certification programs for components, including software and software library certifications. It is up to the

integrator to ensure these are used correctly and to test the final system and to meet regulatory requirements. The prime contractor does the regulatory submission and is responsible for communicating a safety argument (and associated V&V evidence) to the regulatory body.

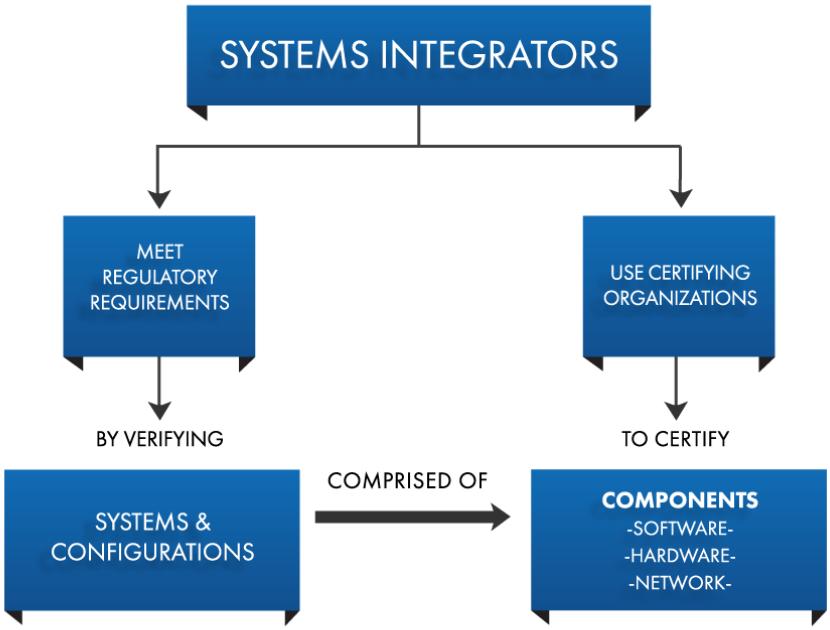

Figure 2: Role of System Integrators and Certification Organizations

Unfortunately, current regulatory frameworks are not well adapted to regulate an ecosystem of IIoT components. The frameworks do not currently scale to regulate effectively all the combinations of components that might be included in an IIoT system. For example, the current US Food and Drug Administration (FDA) regulatory process for medical devices has provisions to approve devices designed to work with other devices via the so-called accessory rule. This rule lets manufacturers seek approval for a device that is expressly designed to work with another device. In such a case, a modular ultrasound machine and its sensor modules can receive approval (and be sold) separately (with the machine itself being the parent device and the sensor modules being accessories) as long as the manufacturer(s) indicate that both machine and sensor modules are designed to work together as a pair and that the manufacturers provide the necessary documentation in their regulatory submission.

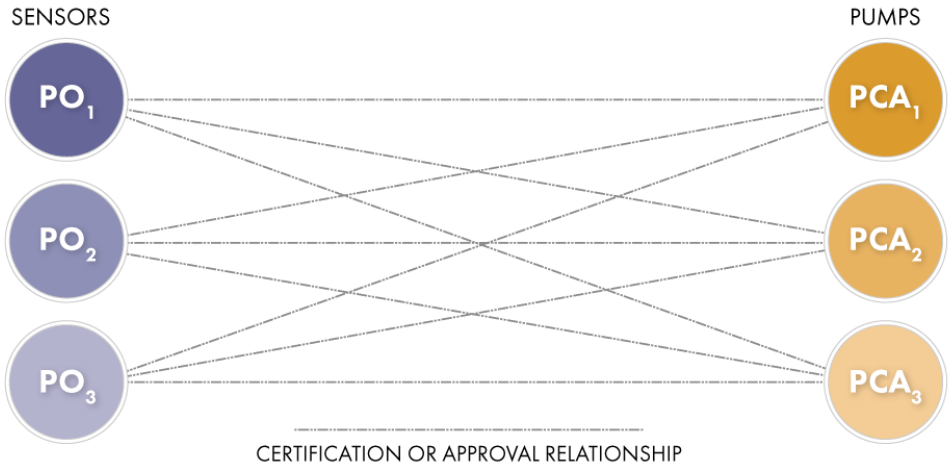

Figure 3: Example medical device ecosystem with 3 sensors and 3 pumps. There are 9 combinations that require independent regulatory filing under "pair-wise" regulation.

The FDA accessory rule and regulatory process is an example of "pair-wise" regulation: Each time a manufacturer (or set of manufacturers) wants to market a pair of medical devices composed into a new system, they need to create a new regulatory submission. For example, pair-wise regulation is required for every combination of Pulse Oximeter (PO) sensors with Patient Controlled Analgesia (PCA) infusion pumps (Figure 3). As long as the ecosystem of devices to be composed is small, pair-wise regulation is tractable. Unfortunately, in IIoT the number of possible device combinations explodes exponentially with respect to the number of devices in the ecosystem. In general, pair-wise regulation is hugely burdensome for both the manufacturers and the regulatory agency: Each regulatory submission usually takes significant resources to prepare and review.

How can regulatory frameworks be designed to minimize regulatory burdens, enable large ecosystems of devices, and still provide the level of safety assurance and protection expected by society? To overcome existing regulatory burdens and help foster a large and vibrant IIoT ecosystem, industry and regulatory bodies should be prepared to move from system and pair-wise regulatory frameworks to approaches that scale with a larger number of interconnected components.

One possible approach to enabling scalable regulation is via a contract-based approach, where a component manufacturer would specify constraints on the interface(s) of the component(s) that their component is designed to work with. These components would actively check that they are being composed with other components whose contracts satisfy those constraints. Regulatory submissions would provide evidence and arguments that the component behaves safely when composed with other components that have been designed for connection using those contracts[14] (i.e., that the compositions of behaviors described in the contracts is not unsafe).

This approach is in contrast to pair-wise regulation, where combinations of specific components are directly regulated. Contract-based regulation regulates components with respect to an explicitly stated contract. This contract would specify information such as types of data the component can send and receive as well as how the component *behaves*[15] when it sends or receives data. With this change, a Pulse Oximeter Interface (POI) and a Patient Controlled Analgesia Interface (PCAI) are added to the previous example (Figure 4).

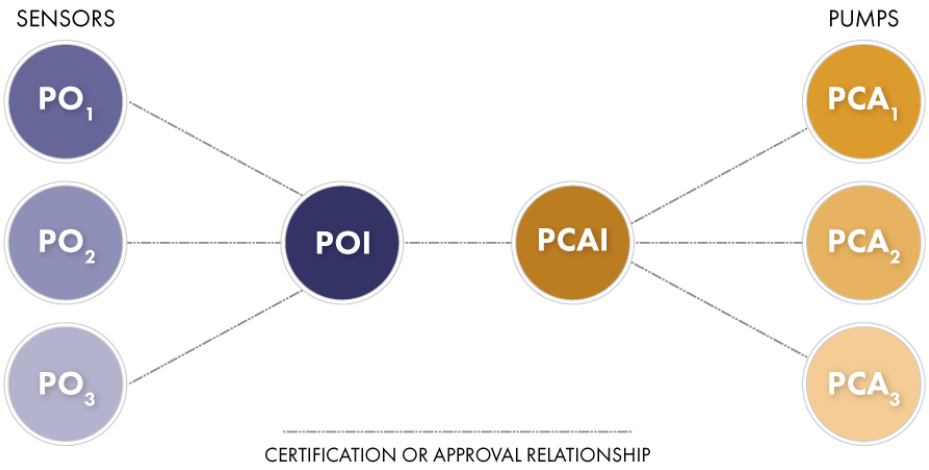

Figure 4: Interface-based, component-wise regulation of a PCAI ecosystem of 3 sensor devices and 3 pump devices. Only one regulatory submission per device is needed.

[14] Towards Assurance for Plug & Play Medical Systems:
http://repository.upenn.edu/cgi/viewcontent.cgi?article=1839&context=cis_papers

[15] A Modal Specification Approach for On-Demand Medical Systems:
http://repository.upenn.edu/cgi/viewcontent.cgi?article=1804&context=cis_papers

Work on this concept is progressing in standards and with regulatory bodies. For example, Annex D in IEC61508 outlines the concept of a Safety Compliant Element and identifies the interface and integration requirements and assumptions (i.e., contract) related to an element or product that must be understood and followed by an integrator to assemble a system with those elements that still can meet the safety requirements. Furthermore, industry and academic research groups are working with the FDA to understand how the regulatory frameworks for medical devices could be modified or extended to support such an approach[16]. Indeed, the FDA has recently released guidance for manufacturers of interoperable medical devices anticipating the types of information that could or should be specified within a contract on an interoperable medical device[17].

WHAT NEXT?

Achieving safety and security will require management and design efforts created to avoid faults and build-in safety and security in all phases of the project. Verification and validation, the use of safety-compliant elements, adoption of security best practices and a review of the overall system and its components are all important practices to achieve a system that meets appropriate safety and security requirements. This all implies a safety and security in-depth strategy with a view toward the overall result.

This white paper has outlined a number of issues that require strong consideration. The questions are challenging and will require further work on trustworthiness, including an understanding of the tradeoffs related to aspects such as security and safety. We encourage interested parties, including those working in both IT and OT to collaborate to find joint solutions.

[16] Medical Device Safety Interoperability Working Group. Pre-IDE Submission for Integrated Medical Systems. http://mdpnp.org/uploads/MDISWG_Cover_letter_and_FDA_Pre-Submission_I120162_Supplement.pdf

[17] Design Considerations and Premarket Submission Recommendations for Interoperable Medical Devices *https://www.fda.gov/downloads/MedicalDevices/DeviceRegulationandGuidance/GuidanceDocuments/UCM482649.pdf*

A publication of the Safety Task Group, authored by Qinqing Zhang (Johns Hopkins University), Andrew King (University of Pennsylvania), Frederick Hirsch (Fujitsu) and Semen Kort (Kaspersky Lab).