BlackBerry | QNX

Simplifying Software Integration and Safety Certification for Medical Devices

Scott L. Linke

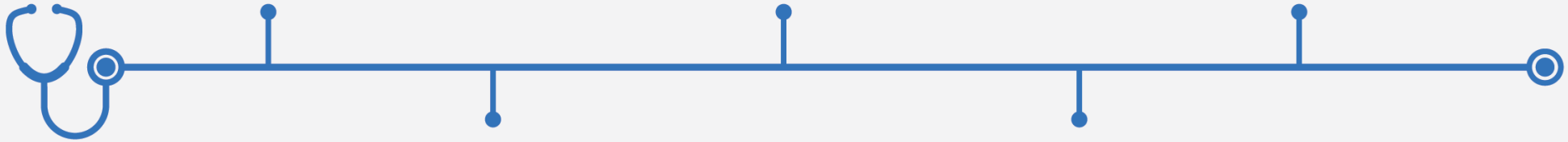
June 25, 2019

Trends for the **Medical Market**

Sophisticated and Connected
With accelerating technology innovation, medical equipment and devices are becoming more sophisticated and more connected

Safety and Security
There is a growing need to improve the approach towards medical device safety and security

FDA Activities
This has been reflected by recent activities and guidelines from the FDA around these two topics



Scrutiny on Software
FDA is placing more emphasis on the software in medical devices

Focus on Cybersecurity
FDA is considering cybersecurity testing to be the responsibility of the medical product manufacturer

Safety – Finding Hazards

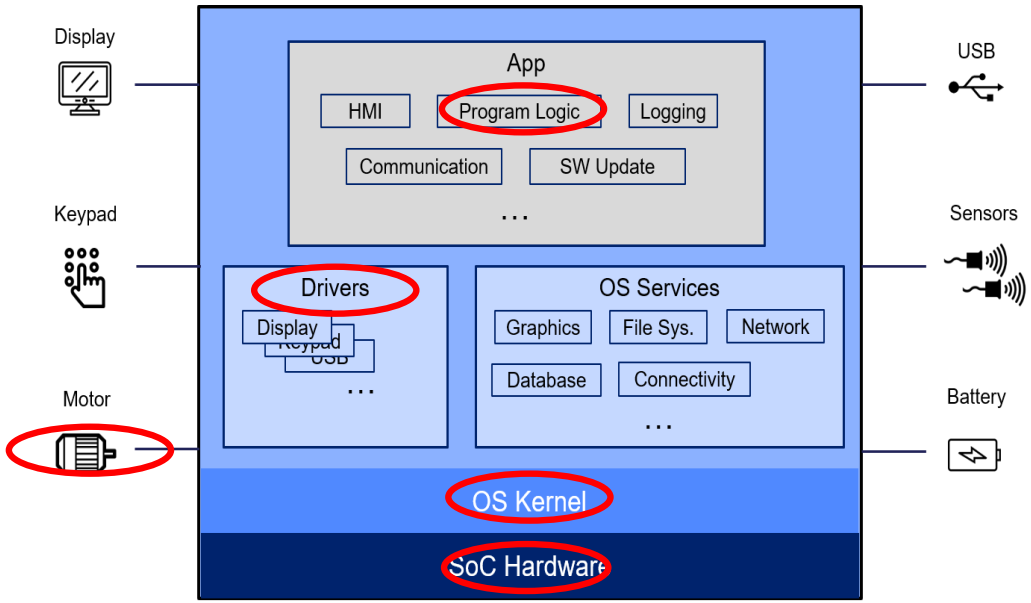
Start with the identification of safety hazards in the system

Example of system level safety hazard:

Device does not perform the prescribed action A within time T after receiving the command

This hazard must be addressed by all the implicated components in the system. As an example, any of the following conditions could lead to the materialization of this hazard:

- The hardware's power unit malfunctions after receiving the command
- A logical error could occur in action *A*
- The operating system does not respond in time within time *T* after receiving the command
- Another action *B* could interfere with the proper execution of action *A*



Safety – Defining Requirements

In order to mitigate the risks resulting from this hazard, safety requirements are defined

Using the example:
Device does not perform the prescribed action A within time T after receiving the command

Risk: The hardware malfunctions after receiving the command
Safety Requirement: The hardware's power unit must have failure probability lower than <threshold>

Risk: The operating system does not respond in time within time T after receiving the command Safety
Requirement: The operating system must have an upper bound for the response time less than T

Risk: A logical error could occur in action A
Safety Requirement: The design of action A must be free from logical errors

Risk: Another action B could interfere with the proper execution of action A
Safety Requirement: Action A must be free from interference from another action in the system

Safety – RTOS Requirements

Zooming in on one of the safety requirements we defined for the RTOS:

Action A must be free from interference from another action in the system

This safety requirement actually translates into multiple requirements for the OS, including:

Run-time Isolation
The code comprising the QOSM SHALL be isolated at run time from faults, errors and failures in other components of the operating system.

Memory Protection
The QOSM SHALL prevent code running as an application writing to memory used exclusively by the QOSM.

IOMMU
The application guidance provided for engineers deploying the QOSM SHALL require the use of IOMMU-equivalent constraints on any hardware capable of direct memory access.

System Call Robustness
The QOSM SHALL protect itself against improper system calls. In particular it must be impossible for any application code to cause a failure in the QOSM by accidentally or maliciously passing a particular value in the parameter of a system call.

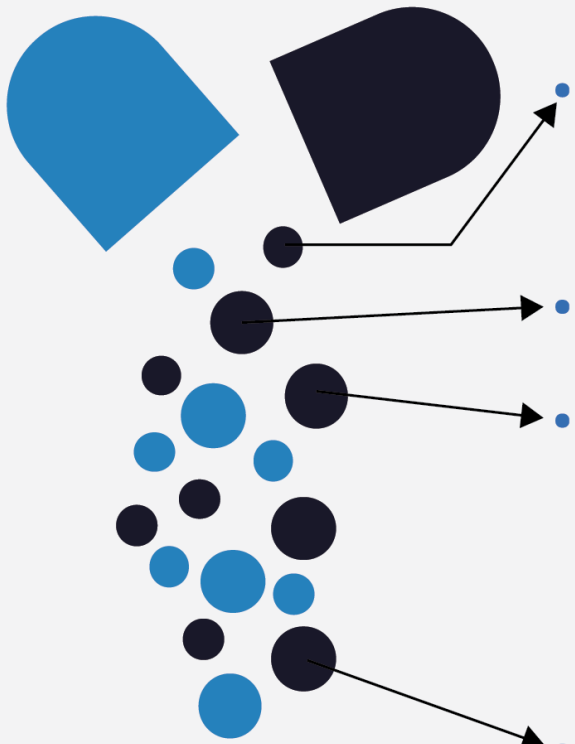
Safety – Selecting Off The Shelf Components

OTS is a Necessity

With the increasing complexity of today's medical devices, the use of OTS (off-the-shelf) components is a necessity

Safety Pedigree

When choosing an OTS component, it is important to understand its safety pedigree



How to select OTS components

- The supplier's past track record in similar type of systems
- Product information
- Standard compliance – the FDA has multiple recognized consensus standards for safety and security, which are listed in their guidelines for the industry
- Supplier Audit

QNX Functional Safety (FuSa) Products

QNX OS for Safety (QOS)

- Certified version of SDP 7.0 to ISO 26262 ASIL-D and IEC 61508 SIL3
- Version 2.1 to be released in August 2019

QNX OS for Medical (QOSM)

- Certified version of SDP 7.0 to IEC 62304 Class C
- Available February 2019

QNX Hypervisor for Safety (QHS)

- First QNX certified Hypervisor, to ISO 26262 ASIL-D and IEC 61508 SIL
- To be released in November 2019. Access and runtime will include QOS 2.1

Black Channel (controlled access)

- Point-to-point safe communication, to be certified to ISO 26262 ASIL-D and released in January 2020

QNX Platform for Instrument Clusters (QPIC)

- Instrument Cluster reference platform, with ISO 26262 ASIL-B Certified Graphics Monitor (Apollo Lake) for tell-tale monitoring
- Released as 1.0 in 2018

SAFERTOS

- Strategic Partnership and product integration with WITTENSTEIN, for MCU devices

What is Functional Safety?

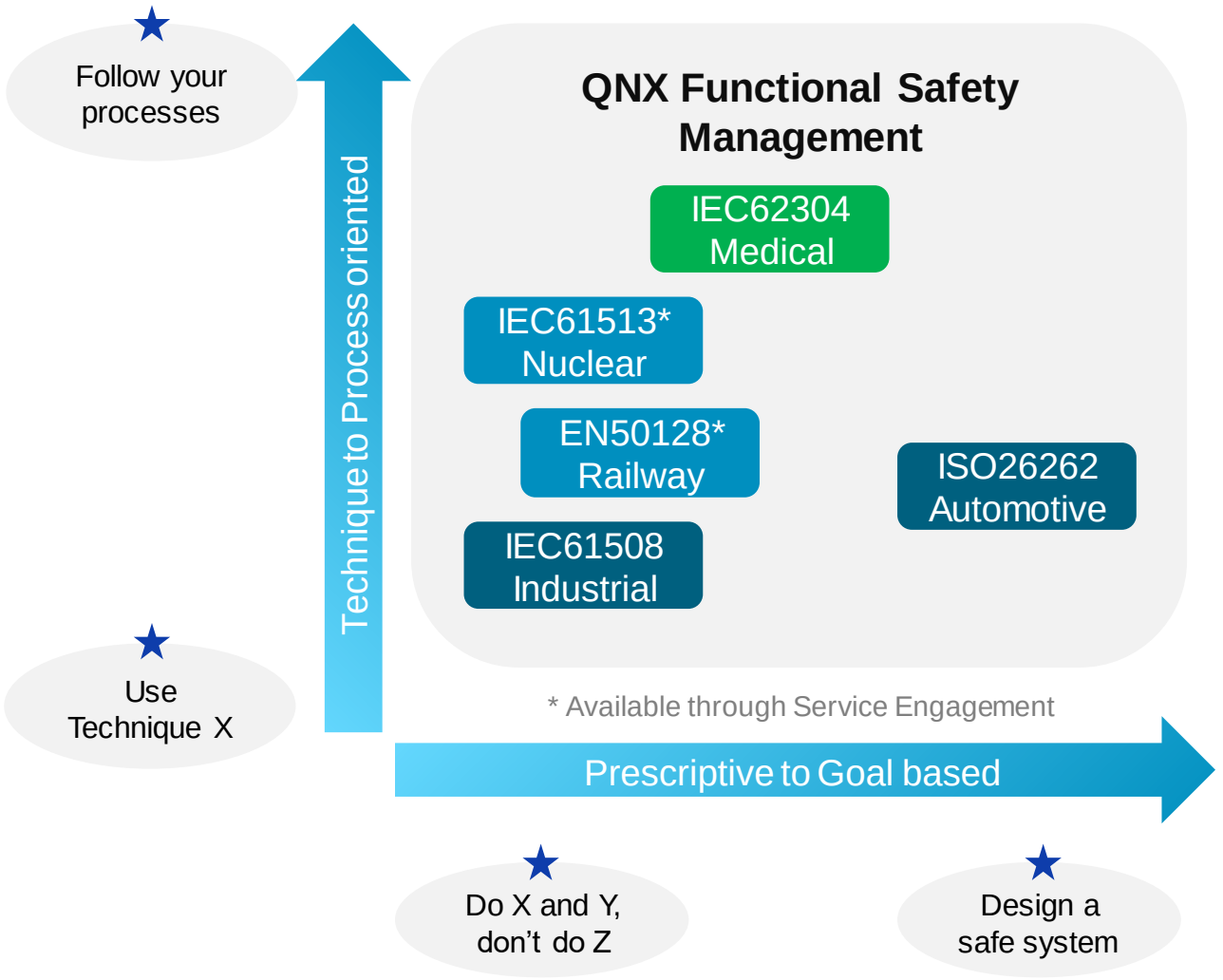
Functional Safety (FuSa) is a QNX pedigree !

- From the standard: “Functional safety is the part of the overall safety of a system or piece of equipment that depends on automatic protection operating correctly in response to its inputs or failure in a predictable manner.”

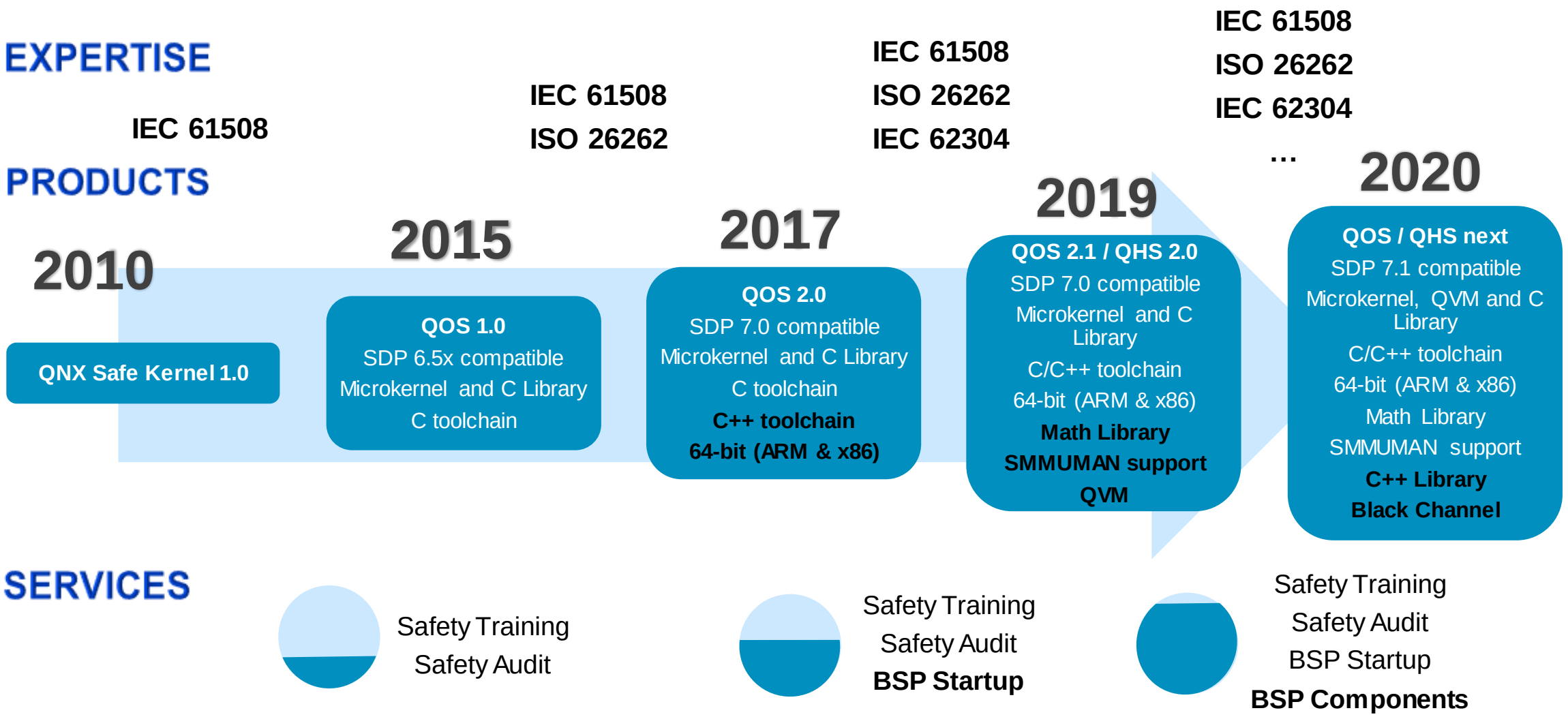
At BlackBerry QNX, we adhere to a wide spectrum of FuSa standards as part of the product development lifecycle

It is in our DNA to follow processes and set safety goals for our products

- We have a long history of proven safety critical product and services delivery that customers can count on.



FuSa at QNX – Ever Increasing Scope



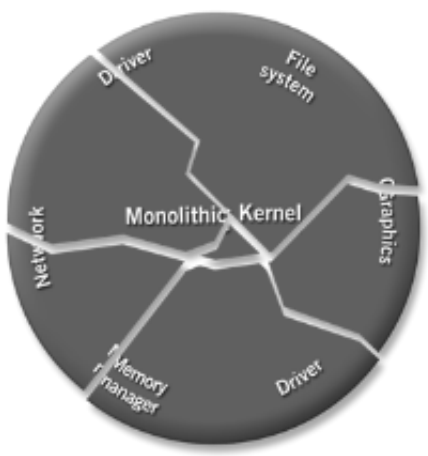
QNX FuSa Value Proposition

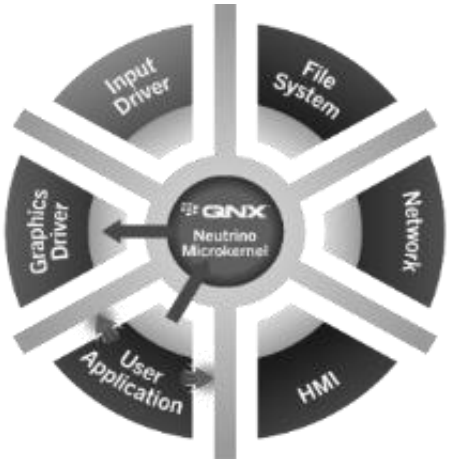
- Best in class support for Functional Safety (FuSa), mixed-criticality and virtualization
- Products certified to ISO 26262 ASIL-D (Automotive), IEC 61508 SIL3 (Industrial), IEC 62304 Class C (Medical)
- Applicable to other markets and standards such as EN 50128 (Railway), IEC 61513 (Nuclear)
- FuSa products assessed by certification body (TUV Rheinland), undergo fault-injection tests and other stringent validation methods
- Products are subjected to continuous safety impact analysis
- Simplified integration to FuSa items delivered as ISO 26262 SEooC to reduce cost
- Expanding product certification scope and engineering services improves stickiness

Safety Case (Historical)

Spatial Separation

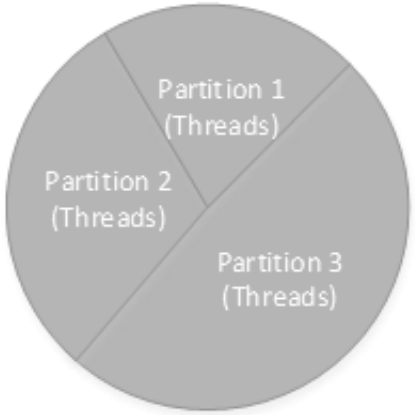
The QOS microkernel architecture separates critical OS components into their own protected memory partitions, unlike a monolithic OS that places them all together. Reduces attack surface.





Temporal Separation

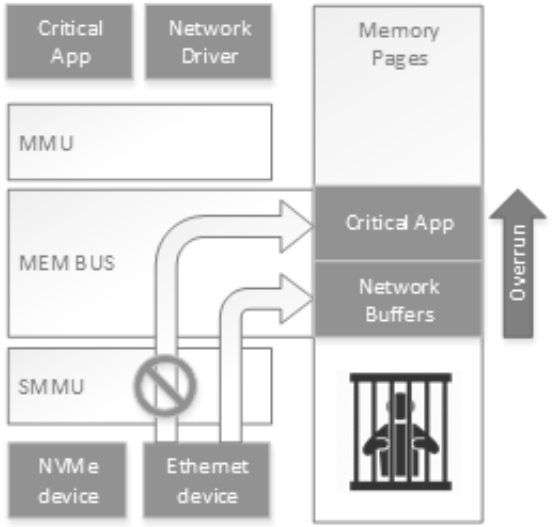
The QOS Adaptive Partitioning System (APS) supports CPU time partitions to limit CPU usage from misbehaved or rogue applications and/or services to starve safety critical applications.



Safety Case (Additions)

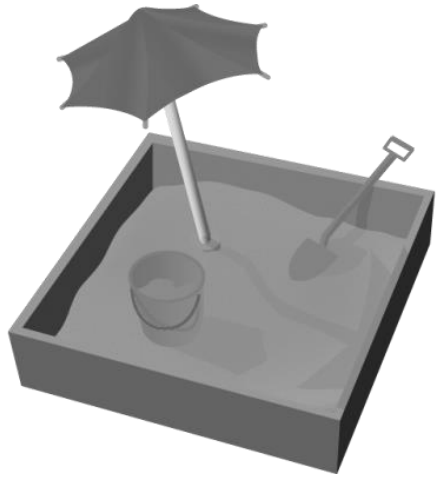
Bus Master Caging (QOS 2.1)

QOS and QNX integrate SMMU support, and allow bounding of memory accesses by bus-mastering device, preventing unintentional or malicious access to safety critical memory.



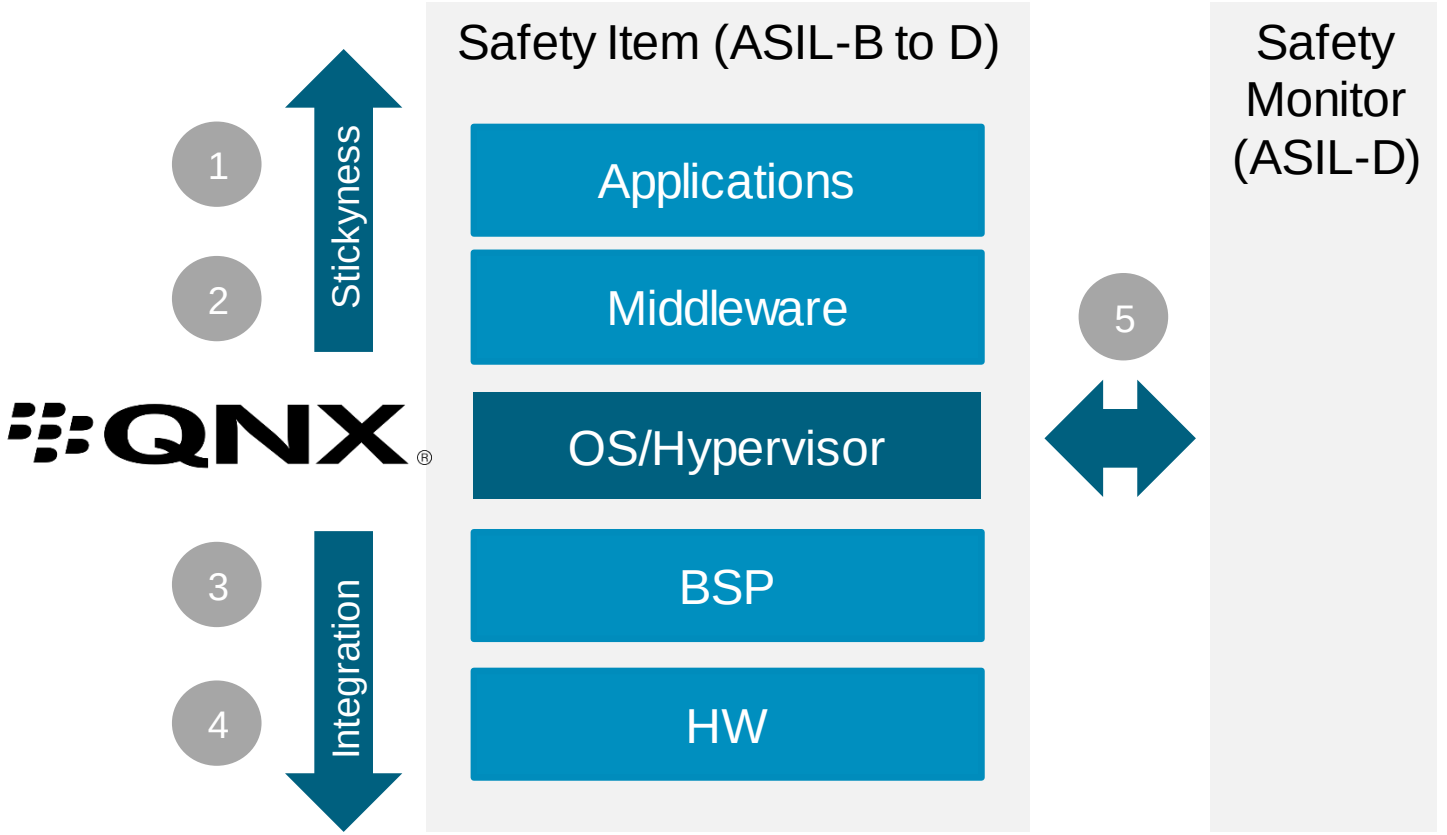
Virtualization (QHS 2.0)

QHS allows OSes to run inside a VM container. It provides freedom from interference between guests, between host and guest, the ability to virtualize safety critical devices and implement a Local Design Safe State (DSS).



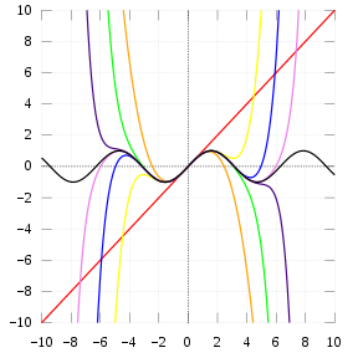
FuSa Expansion Strategy

1. Improve safety artifacts, deliver useful recommendations and safety concepts to customers
2. Expand integrations of 3P safety stacks/services
3. Grow BSP certification capabilities
4. Acknowledge self-test libraries and Silicon IP as part of Safety BSPs
5. Embrace heterogenous computing



Software Qualification (New)

Libm – Innovative (QOS 2.1)



Retrospectively Certified to ASIL-D

Problem:

- Old SOUP implementation

Solution:

- Mathematical validation of calculation accuracy

LibC++ - Labor and Automation (Future)

Standard C++ Library certified to ASIL-B

Problem:

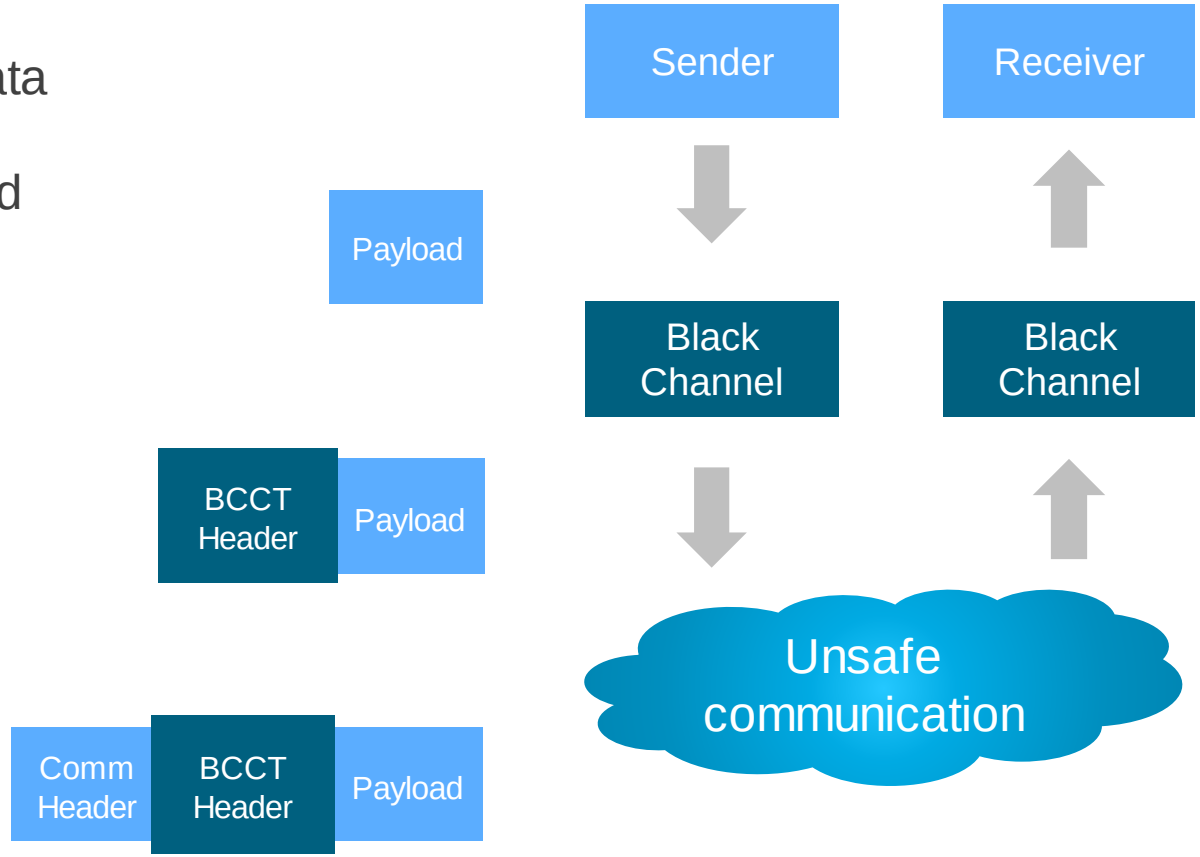
- 1000 page spec
- 4500 functions
- Large gap in testing coverage

Solution:

- Several engineers and tools

Black Channel Communication Technology (New)

- GA release in January 2020
- Safety features, up to ASIL-D, to protect data passed point to point using communication software and hardware NOT safety certified (i.e. Ethernet, UDP, DDS, QSPI, etc)
- "Safety bag" that allows for integrity checking, authentication, detection of data loss and other measures (defined in IEC 61784-3 and AUTOSAR) outside of traditional communication hardware and software.
- Cost reduction for customers to certify communication components for their system safety case.



Q&A