2011 NDIA GROUND VEHICLE SYSTEMS ENGINEERING AND TECHNOLOGY SYMPOSIUM
VEHICLE ELECTRONICS AND ARCHITECTURE (VEA) MINI-SYMPOSIUM
AUGUST 9-11 DEARBORN, MICHIGAN

BENEFITS OF INTRA-VEHICLE DISTRIBUTED NETWORK ARCHITECTURE

Mr. David Jedynak
Network Centric Systems
Curtiss-Wright Controls Electronic Systems
Santa Clarita, CA

ABSTRACT

Curtiss-Wright has developed an advanced, open system architectural approach to Vehicle Electronics, based on our vast experience in providing military electronics to many programs for ground, sea, and air platforms. This experience has provided Curtiss-Wright with a unique understanding of key architectural concepts which provide for highly successful implementation of specific Vehicle Electronics suites to meet Ground Combat System program and platform requirements. This Open-Standard and COTS based Intra-Vehicle Network Reference Architecture was previously presented the paper "Ground Combat Systems Common Vehicle Electronics Architecture and Applications" (D. Jedynak, et al., 2010) and will be summarized and described in terms of the US Army's VICTORY Architecture in this paper as a foundation for discussion. Clarification is provided for the differences between federated and distributed architectures with regard to function, and how physical and functional system implementations are decoupled. Key Metrics associated with the concepts of Interoperability, Risk Mitigation, Upgradeability / Obsolescence Mitigation, Scalability, as well as Space, Weight and Power (SWaP) optimization are presented. Advanced Concepts are presented, including Commonality, Thermal Management, Cost Optimization, and Warfighter Benefit.

INTRODUCTION

Interconnection and interoperability of systems within the vehicle is a challenging task. As operational requirements, technologies, and missions change, vehicle equipment and functions need to adapt. In traditional appliqué models, physical vehicle equipment and functionality is tightly coupled and inflexible. Curtiss-Wright has developed an advanced, open approach to an Intra-Vehicle Distributed Network Architecture, as was previously presented (D. Jedynak, et al., 2010). As opposed to Federated approaches which tightly couple hardware, interface, and software into standalone appliqués, the Distributed approach allows straightforward upgrade, interconnection, and interoperability of vehicle electronics. The benefits of this approach are shown through key metrics and advanced applications. In addition to technical merits of the network itself, benefits can be realized in commonality, cost optimization, vehicle thermal management, and warfighter utilization.

KEY ARCHITECTURAL CONCEPTS

Although a straightforward assumption when discussing Vehicle Electronics architectures in the context of Network Centric Operations, interconnection of these devices can be extremely complex. The very same potential value of the network described by Metcalfe's Law (*potential value creating interactions for N networked nodes is* N^2-N) can provide potential complexity and risk in the process of interconnecting the nodes. The goal of interconnecting is simple, but the implementation can be chaotic and unstable without a well defined architectural approach. In order to implement successful Intra-Vehicle Distributed Network Architecture, a number of key architectural concepts need to be considered, specifically:

- Interoperability
- Risk Mitigation
- Upgradeability / Obsolescence
- Scalability
- Optimizations

These concepts are discussed below, along with the presentation of key metrics related to each.

Interoperability

The goal of interoperability is the seamless integration of both legacy and new technologies from multiple vendors in a nonproprietary open systems approach to vehicle networks.

Interoperability between old technologies and new technologies is a constant challenge when modernizing or implementing vehicle networks. As opposed to the traditional approach of bolt on appliqués for Vehicle Electronics upgrades, interoperability allows for staged and flexible changes to the overall system in a low risk manner without requiring significant changes to all vehicle systems. The interoperability also allows for resource sharing, flexibility, and functional reallocation based on evolving mission and platform needs, as well as mitigating obsolescence issues.

The challenge of interoperability is careful evaluation of standards compliance, the proper mix of Commercial-off-the-Shelf (COTS) products and the balance of unique or custom solutions encapsulated by standard interfaces. The systems integration task requires a thorough understanding of the specific program and platform requirements, while leveraging experience and agnostic application of interoperable technologies.

Interoperability is provided through the use of open standards. When custom or proprietary products or technologies are required, application of the architecture properly encapsulates and isolates them from the overall system thus maintaining the system interoperability, scalability and obsolescence mitigation goals.

Interoperability Metrics are defined as follows (High Scores are better):

Metric	Scoring
Vendor Adoption	More than 3 = +2 2-3 = +1 Only 1 = -1
Message Set	Extensible = +2 Open Standard = +1 Proprietary / Custom= 0 Fee Licensed Proprietary = -1
Network & Transport Layers	Open Standard = +1 Proprietary = 0 Fee Licensed Proprietary = -1
Physical & Data Link Layers	Open Standard = +1 Proprietary = 0 Fee Licensed Proprietary = -1
Connector / Pinout	Open Standard = +1 Proprietary = 0 Fee Licensed Proprietary = -1
Overall	Summation of Scoring (max 7)

Risk Mitigation

The architecture mitigates obsolescence risk by allowing for encapsulated and isolated changes to the system without affecting the system as a whole. This allows for staged approaches to changes which can be incorporated when technologies and components are sufficiently mature. At the same time, a flexible architecture will allow for the inclusion of prototypes and commercial equipment in the system to take the place of rugged endpoints during development and demonstration phases anticipating the qualification of rugged components. By using open standards and nonproprietary component interfaces, new elements can be integrated into the system with minimal risk. Reuse of existing components from other systems is also low risk because the component interfaces are well understood and easily integrated into a new system. A robust tools set of message set monitoring, generation, and simulation further reduces the risk by providing tools for arbitrarily interacting with the network.

A significant benefit of risk mitigating architectural goals is reduced cost and reduced time to deploy an entire architecture, as well as an upgrade or expansion of existing architecture. Downgrade integration of proven technologies when newer technologies are unstable or scarce can be performed as well, further providing risk mitigation at an operational level.

Risk Mitigation is provided by adhering to a flexible architecture, utilizing nonproprietary open systems, and providing interoperability with legacy networks and devices, including interoperability between multiple versions of the open standard message sets.

Risk Mitigation Metrics are defined as follows (High Scores are better):

Metric	Scoring
Commercial Sources of Endpoints	Mass Market = +3 Specialized Market = +2 Custom / Building Blocks = +1 None = 0
Versioning in Message Set	Run-time version adaptation = +1 Version incompatibility possible = 0
Communication Interface Tools Availability	Endpoint Simulator Tools = +3 Message Generation Tools = +2 Monitoring Tools = +1 No Tools Available = 0
Overall	Summation of Scoring (max 7)

Upgradeability and Obsolescence Mitigation

The architecture must provide clear paths to upgrade due to obsolescence and evolving mission needs. As with interoperability and risk mitigation, upgradability is provided through the open standards and interfaces, which provide both future and backward compatibility for components. Given overall platform development schedules which may last multiple years, ensuring upgradability exists in the architecture allows for straightforward technical refresh of capabilities as commercial and industrial technologies advance and are adapted for rugged applications.

Upgradeability and Obsolescence Mitigation is provided by adhering to open standards and interfaces, with Metrics defined as follows (High Scores are better):

Metric	Scoring
Versioning in Message Set	Run-time version adaptation = +1 Version incompatibility possible = 0
Message Set Compatibility	Extensible = +3 Forward Compatible = +2 Backward Compatible = +1 Incompatible between versions = 0
Compatibility of Network & Transport Layers	Forward Compatible = +2 Backward Compatible = +1 Incompatible = 0
Compatibility of Data Link & Physical Layers	Forward Compatible = +2 Backward Compatible = +1 Incompatible = 0
Compatibility of Connector / Pinout	Compatible = +1 Incompatible = 0
Overall	Summation of Scoring (max 9)

Scalability

Architectural scalability provides for a common implementation which can be modified for different variants or different missions of a platform, such as commander vehicles versus personnel vehicles. In order to support multiple different platforms which leverage a common heritage and similar (or subset) operational requirements, yet have significantly different SWaP-C constraints, the architecture needs to function in a similar manner, whether for minimal installations (such as for a light vehicle) or for a high capability installations (such as for a heavy combat vehicle). This allows for common components, training, logistics, and leveraging of low risk, mature products.

Scalability is provided by using standard components with sizing and growth considerations as part of the architecture decisions. Scalability Metrics are defined as follows (High Scores are better):

Metric	Scoring
Operational (Runtime) Scalability	Run-time scaling = +1 Design Time scaling = 0
Infrastructure Nodes Required for 3 or more endnodes	None = +1 At least 1 = 0
Infrastructure Nodes Scalability	Stackable = +1 Separate Segments Only = 0
Packaging	Flexible Packaging of Functions = +1 Federated Functions = 0
Addressing	Dynamic Possible = +1 Fixed Only =0
Address Space	>50 = +1 <50 = 0
Overall	Summation of Scoring (max 6)

Optimizations

A balanced approach of low-risk hardware integrated with readily available COTS technologies provides the best SWaP, low risk, and cost optimized solutions for Vehicle Electronics.

Fully COTS systems, although considered low-risk, generally have non-negligible SWaP-C; furthermore, the integration risk of multiple COTS components as opposed to one or two custom elements tends to be higher than expected, as COTS components do not always lead to tightly integrated and optimized high level solutions.

Fully custom systems may provide the best SWaP (and sometimes cost) optimization, but usually do not provide time-to-production schedule nor risk mitigation; however, in some cases some functions are not available as COTS, dictating the custom approach.

The use of various building block design elements (functional modules, reusable programmable logic) provides significantly lower risk and faster time-to-production for custom modules in a larger system.

A balanced approach of COTS standards, components, and technologies mixed with proven low-risk building-block design elements creates SWaP-C optimized modules and systems with low integration risk and fast time-to-production. Optimization Metrics are defined as follows (High Scores are better):

Metric	Scoring
Form and Function dependence	Decoupled = +1 Tightly coupled = -1
Building Block Availability	Mass Market = +1 Specialty Market = 0 Sole Source = -1
Overall	Summation of Scoring (max 2)

INTRA-VEHICLE DISTRIBUTED NETWORK ARCHITECTURE

Conceptual Architecture

The Conceptual Architecture is shown in Figure 1. It shows the Vehicle Electronics using a Network Fabric which provides connections to Operator Interfaces, Processing Elements, Storage, and other subsystems, either through native to legacy interface converters, or directly to native subsystem interfaces.

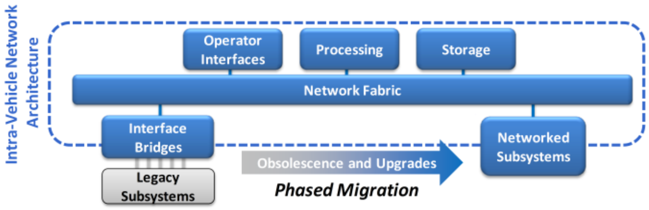

Figure 1: Intra-Vehicle Network Architecture Concept

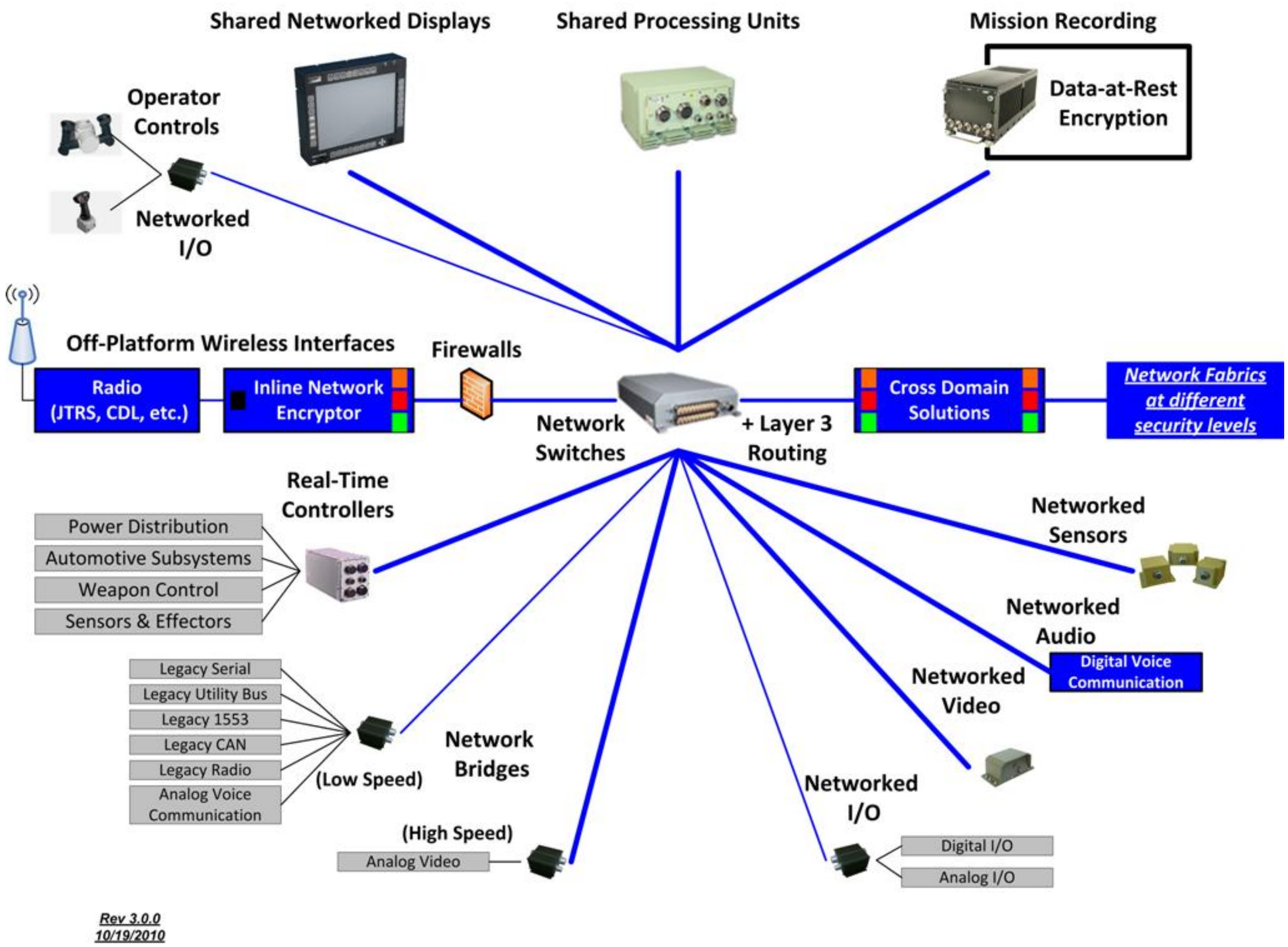

Figure 2: Intra-Vehicle Distributed Network Architecture

Reference Architecture

Curtiss-Wright's Intra-Vehicle Network Architecture (shown in Figure 2) for distributed computing and interfaces can meet the requirements of vehicle programs, incorporating Open Systems Architecture goals and leveraging COTS components. Key building blocks include general-purpose and specialized computing clusters, network switches, real-time controllers, network centric sensors / effectors, I/O concentrators, and legacy bridges. In addition, the architecture shows synergy with the VICTORY Architecture standards (www.victory-standards.org).

Physical versus Functional Architectures

The critical characteristic of the Intra-Vehicle Network Architecture is the distributed nature of the functionality. This does not necessarily mean that the physical implementation of the functionality is distributed, but that the functions themselves are encapsulated and distributed across a network infrastructure. Use of well defined network protocols to form the network infrastructure can be physically embodied in many ways, including combinations of:

- Inter-process communication on a single processing core
- Inter-process communication between multiple virtual machines on single or multiple core
- Inter-process communication between multiple physically separate cores residing in a single enclosure
- Inter-process communication between multiple physically separate cores residing in multiple enclosures

This benefit is significant compared to Federated approaches which require tight coupling of associated processes in single enclosures, often on a single processing module with directly connected I/O. A Distributed network approach removes the tight coupling of function with form, allowing flexibility to implement the physical architecture without significant regard to functional constraints.

Scoring of Intra-Vehicle Distributed Network

The Intra-Vehicle Network Reference Architecture is scored, based on the previously presented metrics, as follows. The Aggregate Score is 27 of 31 (expected 29 of 31 given wide adoption).

Interoperability, assuming VICTORY Message Sets:

Metric	Scoring
Vendor Adoption	To be assessed (score 0), expected wide (score 2)
Message Set	Extensible = +2
Network & Transport Layers	Open Standard = +1
Physical & Data Link Layers	Open Standard = +1
Connector / Pinout	Proprietary (not defined) = 0
Overall	4 of 7 (expected 6 of 7)

Risk Mitigation, assuming VICTORY message sets:

Metric	Scoring
Commercial Sources of Endpoints	Mass Market = +3
Versioning in Message Set	Run-time version adaptation = +1
Communication Interface Tools Availability	Endpoint Simulator Tools = +3
Overall	7 of 7

Upgradeability and Obsolescence Mitigation, assuming VICTORY message sets:

Metric	Scoring
Versioning in Message Set	Run-time version adaptation = +1
Message Set Compatibility	Extensible = +3
Compatibility of Network & Transport Layers	Forward Compatible = +2
Compatibility of Data Link & Physical Layers	Forward Compatible = +2
Compatibility of Connector / Pinout	Incompatible (not defined) = 0
Overall	8 of 9

Scalability, assuming VICTORY message sets:

Metric	Scoring
Operational (Runtime) Scalability	Run-time scaling = +1
Infrastructure Nodes Required for 3 or more end nodes	At least 1 = 0
Infrastructure Nodes Scalability	Stackable = +1
Packaging	Flexible Packaging of Functions = +1
Addressing	Dynamic Possible = +1
Address Space	>50 = +1
Overall	5 of 6

Optimization, assuming VICTORY message sets:

Metric	Scoring
Form and Function dependence	Decoupled = +1
Building Block Availability	Mass Market = +1
Overall	2 of 2

ADVANCED CONCEPTS

Commonality

Commonality across ground vehicles requires the proper segmentation of subsystems and interfaces to allow for commonality. A traditional Federated Superset Approach, merging the subsystem interfaces of multiple vehicles into a one-size-fits-all set of Line Replaceable Units results in a high degree of complexity and instability of Interface Control Definitions (ICDs) across the Family of Systems. An illustration of the problems, complexity, and instability of this approach are shown in Figure 3.

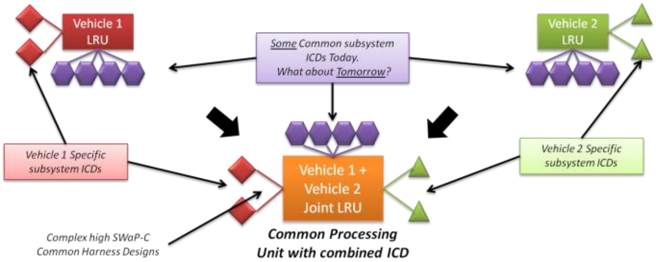

Figure 3: Federated Superset Approach to Commonality - Not Recommended

This approach is not recommended since it requires all the individual vehicles and subsystems to be designed, implemented, and maintained in concert for the life of the vehicles in order to ensure required changes for one vehicle or subsystem do not affect the commonality with the other vehicles. Furthermore, the central LRUs with common interfaces for all the subsystems of the entire set of vehicles will be SWaP-C overburdened in order to meet the specific requirements of all vehicles.

The modern, scalable, and interoperable approach is the Distributed Subset Approach, which builds upon well defined and well selected interfaces between well defined subsystems, as shown in Figure 4. This modular approach creates commonality around well defined subsets of interfaces in all vehicles using standardized distributed network interfaces, such as the VICTORY standards.

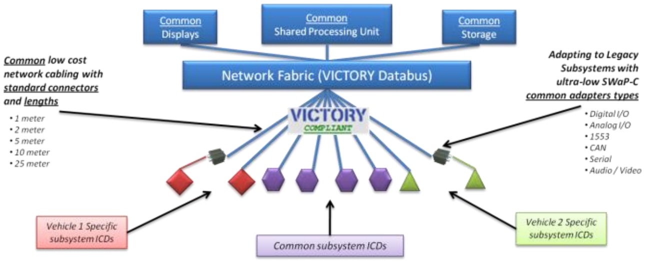

Figure 4: Distributed Subset Approach for Commonality - Preferred Solution

In this approach, modules, such as Common Processing, Displays, and Storage only need network connections in any vehicle. Common and Specific vehicle subsystems attach using standardized interfaces and cabling. Key to this approach is sets of open standard interfaces between the subsystems, allowing the vehicle integrators to build upon these well defined interfaces between standard building blocks.

Thermal Management

A key constraint of federated systems is that processing and the resultant thermal management required is constrained by the clustered physical location of the devices. In vehicles with active thermal management systems, such as liquid cooling loops, the concentration of heat from processors can be managed efficiently; however, in vehicles without active thermal management systems, that tight coupling of a federated system can be problematic. The distributed architecture allows the physical clustering of the processing (and resultant heat) to be spread out in a manner appropriate for the vehicle's lesser capabilities of thermal management systems, such as cold-plates or natural convection.

Open Standards and Costs

Open standards, such VICTORY, provide the Department of Defense, vehicle integrators, and subsystem vendors the path forward for both modernization, growth, scalability, as well as reduced integration costs. Furthermore, given a marketplace of standards compliant and interoperable solutions from multiple vendors, products and technologies will compete to optimize SWaP-C, performance, lead-times & availability, at various price-points, similar to other high technology markets. Similar to the open standards of VITA (VME, VPX, OpenVPX, etc.), vendors can focus on subsystem innovation, instead of expending valuable time and resources on defining non-valued-added interface protocols and message sets. This will provide the warfighter with rapid access to emerging technologies at a pace similar to these other high technology markets. In these adjacent markets, the common interface standards of technologies such as USB, HDMI, and various mobile phone standards allow hardware vendors to innovate and users to select the right mix of technology for use without technical interface constraints. Use of open standards defining the outward interfaces of technology modules allows vendors to innovate and compete with proprietary methods, designs, technologies, and software to provide differentiation in value (e.g. performance, cost, SWaP), while ensuring interoperability and compatibility within the marketplace of similar components. This key differentiation between open standard interfaces and proprietary internal design of modules is critical for fostering and encouraging the continual improvement of industry solutions in terms of both cost and performance.

Warfighter Utilization

The ultimate benefit of the Intra-Vehicle Distributed Network Architecture is in its utilization by the warfighter. The ability to encapsulate functions on the network allows for a number of training and operational benefits.

With regard to training, the architecture allows for encapsulation, simulation, and isolation of functional blocks. Embedding training can take place within a vehicle by insertion of simulated (or previously recorded) network traffic between the distributed functions. During development of vehicle systems, software simulations of functions can be integrated with actual physical functions, providing earlier access for user juries and usage analysis prior to completion of all functional blocks.

In operation, the architecture ultimately allows for a higher degree of vehicle robustness and flexibility, situational awareness, and prognostics / diagnostics. The distribution of functions across the network infrastructure provides for physically separated redundant fail-over designs, straightforward cannibalization of damaged vehicles to return lost functions, and ability to create ad-hoc mission packages with little to no integration effort. The rich set of data on the network infrastructure from multiple devices and sensors provides significant data fusion and analysis opportunities, potentially increasing situational awareness of both the vehicle and environment. This same data can be analyzed either in real-time or post-processed to both predict and diagnosis vehicle failures, potentially increasing the operational availability of the vehicle for the warfighter.

CONCLUSION

Curtiss-Wright's Intra-Vehicle Distributed Network Architecture provides an open, flexible, capable, scalable, and robust approach for vehicle electronics. It provides a clear path forward to high value interconnections on vehicles, whether through incremental modernization efforts or new fully native network designs. The decoupling of physical forms and functions through the transition from Federated to Distributed systems removes a critical constraint from vehicle design.

Well defined network interfaces provide significant benefits in interoperability, risk mitigation, upgradeability / obsolescence mitigation, scalability, and optimizations. Beyond these foundational benefits, the architecture allows for rapid development of applications for multiple vehicle types, which in turn allows for the rapid discovery and leveraging of commonality, both at the sub-component and architectural building-block level. Thermal management strategies, acquisition costs, warfighter training and operations are all benefited by the distribution of functions across the network architecture.