2015 NDIA GROUND VEHICLE SYSTEMS ENGINEERING AND TECHNOLOGY SYMPOSIUM
AUTONOMOUS GROUND SYSTEMS (AGS) TECHNICAL SESSION
AUGUST 4-6, 2015 - NOVI, MICHIGAN

A MODULAR OPEN SYSTEM ARCHITECTURE STRATEGY FOR ROBOTICS AND AUTONOMOUS SYSTEMS

Michael S. Moore, PhD
Southwest Research Institute
San Antonio, TX

Mark Mazzara
PM Force Projection
Warren, MI

Bill Thomasmeyer
National Advanced Mobility Consortium
Ann Arbor, MI

Frank Lentine
QinetiQ North America
Waltham, MA

Danny Kent
OpenJAUS, LLC
Winter Springs, FL

David Martin
Neya Systems, LLC
Wexford, PA

ABSTRACT

This paper describes the results of a study funded by the National Advanced Mobility Consortium (NAMC) to develop a strategy for establishing interoperability as the norm in military ground robotic and autonomous system (RAS) programs. It briefly provides background explaining the current practices and the reason the study was conducted. It outlines the types of interoperability targeted in ground RAS programs, and describes the findings of a survey of current efforts aimed at creating interoperability through a modular open system architecture approach. It recommends a path forward for creating interoperability in military ground RAS program based on maturing and propagating the ground robotics interoperability profile (IOP) currently being developed and matured at Project Manager, Force Projection (PM FP). Finally it lays out specific steps to be taken and proposes that responsibility for IOP be transitioned to a consortium-style organization as it progresses through an "Iteration and Maturation" phase over the next 3-5 years towards its eventual adoption by an enduring standards body. The views expressed in this paper do not constitute official Department of Defense policy.

INTRODUCTION

This is a crucial time for the future of Robotic and Autonomous Systems (RAS) in general and Robotic and Autonomous Ground Vehicle Systems ("RA-GVS") in particular. Historically, the vast majority of military ground robotics systems have been procured to support immediate operational needs. This has resulted in a number of built-to-purpose, tightly integrated systems that have proven their operational utility and become indispensable. However, as current conflicts wind down and budgets shrink, maintaining and adding new capabilities to such built-to-purpose ground robotic systems will prove an expensive and difficult proposition. The difficulty in tightly integrated, built-to-purpose systems is that they are costly to maintain and extend, and it is not possible to select and integrate "best of breed" components. Moving forward, the community must lessen the life-cycle costs, shorten the technology update cycles, and enable operational flexibility.

In the case of autonomous systems, science and technology (S&T) projects have proven the technological feasibility and potential benefits of autonomy to a range of applications, from automated convoys, to fully self-driving vehicles. Successful transition of autonomous systems into real-world deployment will require development of well-understood, modular packages that can be integrated inexpensively with legacy platforms and worked into military operational processes. The continued development of tightly integrated, built-to-purpose autonomous systems is not a tenable approach, and delays the transition of these technologies.

To mitigate these difficulties, and to move to the next level in deployment of robotic and autonomous systems, the community of interest (COI) must proactively develop and embrace a modular open systems architecture (MOSA) for robotics and autonomous systems. A MOSA will provide a shared architectural framework and a set of interface standards, and will promote modularity, commonality, and interoperability between sub-systems and components. A MOSA for RA-GVS will enable an environment of competition and innovation in the community, and "grease the skids" for developing, integrating, deploying, and maintaining a wide variety of interoperable, robotic, and autonomous ground vehicles and platforms.

In 2010, the Army and Marine Corps ground vehicle community initiated the process of creating a MOSA for RA-GVS when a group operating out of the then Robotic Systems-Joint Program Office (RS-JPO) commenced work on developing a collection of Interoperability Profiles (IOP) for unmanned ground vehicles. IOP is intended to provide program managers (PMs), and eventually others, with a standardized library of physical, electrical, and logical (messaging) interfaces, and a common set of supporting documentation and materials that they can use to define a common interoperability profile, or "instantiation", specific to a certain robotic vehicle or platform. The instantiation specifies which interfaces and interoperability attributes, from among those defined in the overarching IOP, are to be implemented on a particular RAS.

Over the past 5 years, work has focused on developing initial versions of the IOP and evaluating the technical feasibility of utilizing such an open interface standard without sacrificing operational performance. In the meantime, responsibility for IOP has transitioned to an IOP group operating under the Project Manager, Force Projection organization, under the Army's Program Executive Office for Combat Support and Combat Service Support ("PEO CS&CSS"). With the pending release of IOP Version 2.0, the Government will have completed the Initial Development stage, which has resulted in a well-defined set of IOP documents and initial demonstration of technically sound underpinnings for a RAS MOSA.

Definitions

This section provides a set of definitions for several terms used throughout the remainder of the text. Note that many of these are used ambiguously within unmanned systems or focus on an alternate interpretation across different domains. The purpose here is not to provide a complete definitions document, but rather to focus on specific terms that may cause confusion.

- Open Architecture (or Open Systems Architecture): "a type of computer hardware or software architecture […] that allows adding, upgrading, modifying, and swapping components. [It provides] a varied combination of interoperability, portability, and open software standards." [1]
- Open Standard: "standards made available to the general public and are developed (or approved) and maintained via a collaborative and consensus driven process. [They] facilitate interoperability and data exchange among different products or services and are intended for widespread adoption." [2] Note that for purposes of this document, a standard will still be considered "open" if a small fee is required to obtain it, such as Society and Automotive Engineering (SAE) documents.
- Interface specification: a complete, unambiguous, and testable description of an interface. In robotics, this may include physical (mounting points, dimensions, weight), electrical (voltage, current), and logical (software, communication bus) interfaces.
- Interface: "a point where two systems, sub-systems, components, subjects, organizations, etc., meet and interact". [3]
- Modular: "having parts that can be connected or combined in different ways" [4]
- Interoperability: "the predictable performance of a capability across an interface through compliance to a selected set of specifications" [5]
- Platform: the base vehicle or mobility chassis of a robotic system.
- Payload: "a device carried by a [platform], usually in a bay or attached to a hardpoint" [5]
- End Effector: "last link of a manipulator, often modular to accept various tools or instruments"
- Controller (or Operator Control Unit): "A hardware and/or software interface that allows a human to command or monitor one or more unmanned systems."
- Appliqué: "The augmentation of a manned vehicle such that it can be semi-autonomously controlled." [6]

While most acronyms within this paper are defined within the paragraph of their specific use, others are used extensively throughout the document. A partial reference list is provided here for convenience:

- RAS : Robotics and Autonomous System
- UGV : Unmanned Ground Vehicle
- IOP : Interoperability Profiles
- JAUS: Joint Architecture for Unmanned Systems
- AEODRS: Advanced EOD Robotic System
- NAMC: National Advanced Mobility Consortium
- ISR: Intelligence, Surveillance, and Reconnaissance
- MOSA: Modular Open Systems Architecture

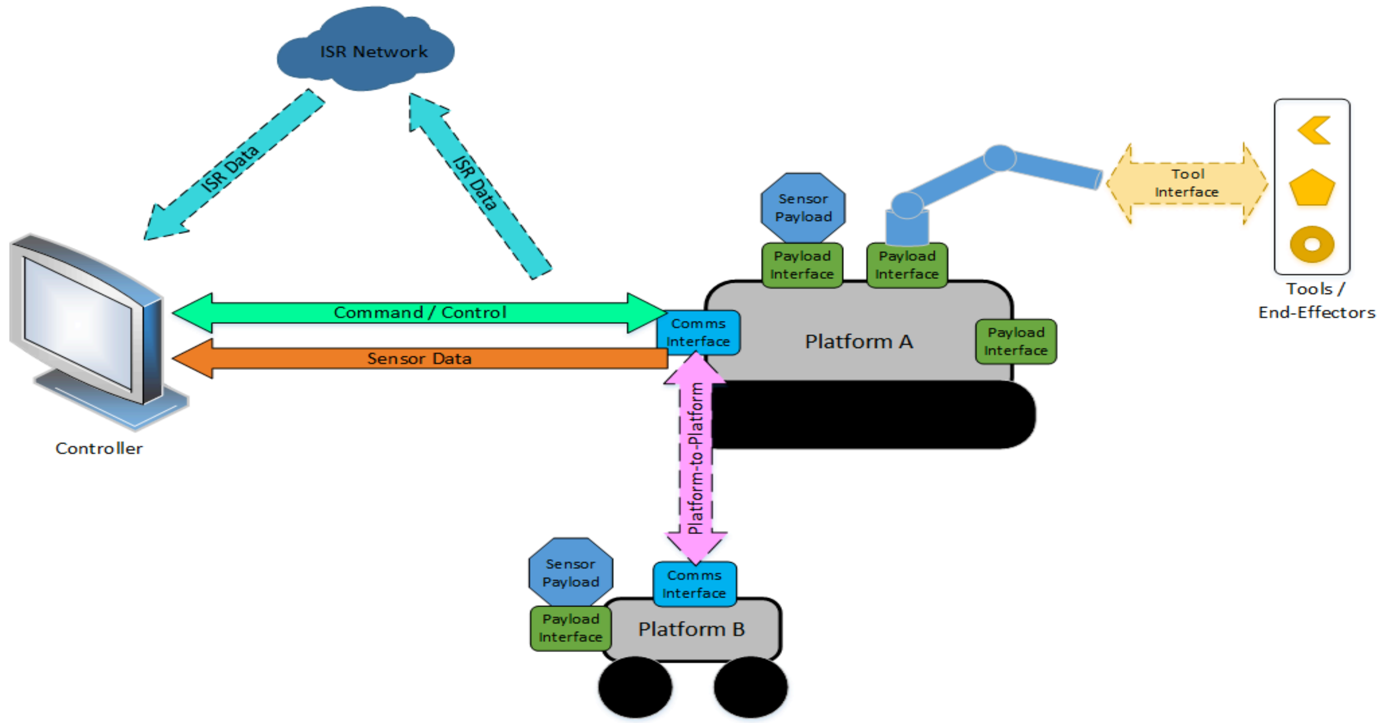

Figure 1. Ground Robotic and Autonomous System Conceptual Architecture

GROUND ROBOTICS AND AUTONOMOUS SYSTEMS INTEROPERABILITY

The Architecture Framework for Unmanned Systems defines interoperability as "the predictable performance of a capability across an interface through compliance to a selected set of specifications" [5]. Therefore, to promote interoperability, an acquisition process must determine the appropriate set of specifications to define the physical, electrical, and logical interfaces between two or more entities. Equally important, however, is that the specific goal of interoperability must be determined. While interoperability generally leads to lower maintenance costs, longer lifespan, and an open marketplace for third party suppliers, specific *use cases* can be considered as representative types of interoperability. These use cases reflect the boundaries between unmanned systems components for which interfaces must be defined.

In-Field Swap of Vehicle Payloads

A common goal of interoperability is the ability to quickly switch out payloads on a robotic platform in the field. For the purposes of this document, "in the field" is considered to be any location in which full engineering and technical services are not available, but swapping may still require a power cycle or minor configuration changes performed by the vehicle operator. This type of interoperability may include functionally equivalent payloads from different manufacturers, or switching to a payload with a different mission function. In all cases, the payloads must have well-defined physical, electrical, and logical interfaces. It is important to note, however, that compliance to well-defined interfaces may still not be sufficient for interoperability. A large, high capacity manipulator arm may be too heavy for a small, backpack-able vehicle regardless of the commonality of interfaces.

Swapping Payloads at a Depot or Maintenance Facility

This use-case is similar to Use Case 4.1 noted above, but may require additional tools, expertise, or time to change payloads. Consequently, the service is not performed in the field, but rather at a depot, forward operating base, maintenance facility, or other location with enhanced services. For example, a new payload may require a software update to be installed on the controller to take advantage of new functionality. Alternatively, additional or complex configuration changes may be required that require more time or specific training.

Common Payload used across Programs

A frequent challenge in military procurements is properly specifying equipment to be used across multiple Programs or even multiple projects within a single Program Office. The goal of this use case is effective implementation of interoperability such that a single payload can be used across multiple programs. This could be achieved by multiple programs coordinating on a single specification for the physical, electrical, and logical interfaces; or by a program adopting a specification already in use. This will likely also result in use case(s) 4.1 and/or 4.2.

Common Design Artifacts used across Services

Similar to 4.3, frequently it is a challenge in military procurements to make use of equipment from programs developed by different services. The goal of this use case is to ensure that common specifications can be developed and used across the different services. Consequently, common electrical, mechanical and logical interfaces should be developed which solve the same (or similar) problems regardless of the service branch.

Tool / End-Effector Interoperability

Most robotic systems incorporate a manipulator used for a variety of tasks. The existing systems frequently have manual or automatic changing of the tools based on the task to be performed and do not allow the user to change tools between different systems. This use case allows interoperability of tools such that new tools can be added to a robotic system without the need to replace the entire manipulator and tools may be swapped between different platforms.

Platform-to-Platform Information Sharing

Fielded robots have historically been tele-operated and have relied upon the concept of a controller communicating directly with a robotic platform. As advances in technology allow for greater levels of autonomy, the need for direct platform-to-platform communication to facilitate autonomous robot teaming will most likely prove necessary. This interoperability use-case would focus on the sharing of information directly between robotic assets without a controller-in-the-loop. A near-term example is autonomous vehicle convoy, in which a lead platform shares its current position directly with followers.

Common Controller for Multiple Robotic Platforms

In most cases, vehicle controllers are mated one-to-one with a robotic platform. Furthermore, the physical design and user interfaces of the controllers vary significantly between vendors, leading to additional training requirements for unmanned system operators. This interoperability use-case would promote a single operator interface capable of controlling multiple, different vehicles.

Note that a concurrent study being executed by the NAMC is largely focused on the issue of common controllers. Consequently, this use case will be considered *out of scope* for the purpose of this document. However, the recommendations presented in this document should not prevent this type of interoperability, and the authors recommend an additional follow-on investigation to merge the final recommendations of that effort with this one.

Cross-Control Interoperability

Traditionally, operator control units have been paired one-to-one with tele-operated robots. Currently if two operators, each an expert in a specific portion of a mission, want to share a single robot they must be co-located at a single operator control unit. This interoperability use-case would allow multiple operator control units to handoff control of a robotic asset, allowing for different operators to share a single platform.

Cross-Domain Interoperability

Existing interoperability standards have generally focused on interoperability within a single domain (ground, air, or sea). On-going areas of research include development of a common controller for both ground and air robotic assets and the sharing of Intelligence, Surveillance, and Reconnaissance (ISR) data between domains. This interoperability use-case would contain all forms of cross-domain interoperability, including a common controller across domains, as well as sharing of ISR data between systems operating within different domains.

Note that the purpose of this study is to focus on interoperability within unmanned *ground* systems. Consequently, this use case will be considered *out of scope* for the purpose of this document. However, the recommendations presented in this document should not prevent this type of interoperability, and the authors recommend an additional follow-on investigation into this area in the future.

Asset to ISR Network Interoperability

Unmanned Ground Systems commonly fielded today are isolated systems, with limited connectivity to large-scale C4ISR networks, such as the Global Information Grid (GiG). As unmanned ground vehicles (UGVs) become an increasing presence on the battlefield, interoperability between the vehicles and wider networks will become critical. Data provided to the upstream network may include telemetry, video, and audio; and may be hosted by the vehicle, a payload, or the controller. Furthermore, as command and control methodologies move away from high-bandwidth human-in-the-loop tele-operation to a more distributed, autonomous paradigm, it may be possible to indirectly control one or more vehicles over the network. It should be noted that many of these ISR systems have well-defined interfaces in place, so interoperability will likely require adoption of these existing standards and architectures rather than new development.

Common Controller for Robotic Payloads

Military robotic assets are primarily conduits to remotely operate payloads or interact with the environment through the vehicle's controller. This use case builds upon the logical, mechanical, and electrical interoperability for payloads as outlined in 4.1 and 4.2. In this case, the vehicle controller shall present to the operator a set of common controls based on the currently attached payload. These controls shall be consistent across various vehicle controllers in such a way that an end-user trained to use the payload on one controller can intuit command and control of that payload on another controller without the need for re-training. This includes the operation of payload-specific capabilities or payloads which may work in concert (such as a pan-tilt visual sensor which may automatically slew to watch the end-effector of a companion manipulator). This use-case increases the value of interoperable payloads and the use of swapping payloads across assets, programs, and services as discussed previously.

Appliqué Systems

While smaller RAS platforms may be built from the ground-up with tele-operation and autonomy in mind, that model is unrealistic for larger platforms. Rather, traditionally manned vehicles used for transport, construction, and shipping may be retrofitted with drive-by-wire kits to enable unmanned operation. These "appliqué" systems may interface directly to a vehicle controller through native on-board networks, such as CAN bus or FlexRay, or they may physically mount to the manned control devices like the steering wheel and gas pedal. Regardless of the underlying approach, the goal is to remove the need for a human driver in the vehicle cab. Since there is only a modest level of standardization between vehicle manufacturers for physical inputs and data bus messages, the goal for an appliqué system from an interoperability perspective is to expose a common set of interfaces for controllers and payloads, regardless of specific vehicle needs. Consequently, it can be considered an abstraction layer, isolating standards compliant interfaces from custom ones. This may be further complicated by safety criticality, as appliqué systems that interface directly to a vehicle controller may have additional constraints for real-time performance and deterministic behavior.

Software Plug-Ins

Previous interoperability use-cases focused on distributed hardware components where interoperability was achieved through well-defined physical and electrical interfaces, along with common messaging formats on a wired or wireless communication bus. Within a device, particularly a Human-Machine Interface (HMI), such as a hand-held controller, interoperability between software elements may be achieved through the use of "plug-in" style architectures. This generally requires a single common software framework that manages multiple plug-ins and a shared Application Program Interface (API) that exposes functionality of the framework, such as video display and input widgets. The framework may further limit the programming language used by the plug-in developer, the operating system, the computing architecture, and even the memory and processing power available.

Note that the purpose of this study is to focus on interoperability between controllers, platforms, and payloads used by ground systems. Consequently, the use case of interoperability *within* a device through software plug-ins is considered *out of scope* for the purpose of this document. However, the recommendations presented in this document should not prevent this type of interoperability in the future, should it be considered necessary by the IOP.

DOD AND INDUSTRY INTEROPERABILTY EFFORTS

This section provides a survey of various efforts to create open architectures and achieve interoperability. The intent is not to create an exhaustive list of all efforts that are similar to this one, or to explore all types of interoperability efforts and categorize them (create a taxonomy). The intent is to provide a small set of relevant examples to provide context for an analysis that will result in recommendations for a business model for a RAS prototype interoperability architecture (PIA).

For each example effort, the following aspects are covered:

- Overview:
    - Goal and focus of the effort
    - Entities that are specified (standardized)
- Managing body that develops the specification
- Modification process
- Business model
    - Stakeholders and COI
    - How the effort is financed (who pays for development, maturation, tools)
    - How the specifications are used by the COI
- Availability and access of specifications
- Availability of reference implementations

  - Cost and access
- Availability of tools
  - Tools for developing requirements against the specifications (selecting options)
  - Tools for defining / documenting / executing compliance tests
  - Commercially available products that implement the specification
- Certification:
  - How compliance / conformance testing works
  - Whether there is a certification authority
- Degree of adoption

This section does not attempt to create a taxonomy or to otherwise classify / group the types of efforts; the survey is done as a simple list of efforts. For that reason the reader may identify commonalities / similarities between some of the efforts. In a later section, categories of efforts are identified based on analysis of these representative efforts.

JAUS

Overview: Joint Architecture for Unmanned Systems (JAUS) is an open, commercial standard that defines logical interfaces between platforms, payloads, and controllers with emphasis on teleoperation of ground vehicles.

Managing Body: SAE International (AS-4 Steering Committee)

Management Type: An open committee made up of individual subject matter experts from industry and Government of any nationality. Voting is reserved for committee members, where membership is solely the discretion of the committee chair.

Modification Process: Change requests may be generated internally by the Committee or by outside organizations. New services (collections of messages) are encouraged to have two independent implementations to demonstrate applicability and interoperability. A Document Sponsor then creates or modifies an Aerospace Standard, which is balloted by the Committee. Configuration Management for the documents is provided by SAE, while the Committee has adopted internal guidelines for version control of the published services to manage backward compatibility between publications. Publication of new or modified services generally takes 18-36 months.

Business Model: Initial development of JAUS was heavily funded by the Joint Ground Robotic Enterprise (JGRE), but direct funding for individual or organization participation has sharply reduced over the past five years. Some leadership positions are currently maintained by Government employees and are often partially funded by the DoD. Generally, however, participation by industry SMEs is at the expense of their organization.

Availability: The documents can be purchased by any person or organization worldwide without restriction for a modest fee (approximately $60 per document).

Reference Implementations: There are no reference implementations or official tools created or endorsed by the SAE committee. Multiple open-source and commercial tools are available from third party sources.

Tools for Compliance Testing: There are no official tools created or endorsed by the SAE committee. At various times over the lifespan of JAUS, DoD offices have created openly available or internal use only tools for evaluating compliance on specific programs.

Certification Authority: There is no authoritative body to certify compliance. Acquisition programs must conduct their own validation.

Degree of Adoption: JAUS has been used in multiple small programs in both the ground and maritime environments. It has moderate adoption among small businesses, as common messages promote rapid integration. Even within this sample, however, custom services not published by the standards body are commonly developed for specific needs or products, but consequently limit interoperability. Adoption from large Prime Integrators has been minimal to date, where the cost associated with compliance is often a cited concern. JAUS forms the basis for the IOP and Advanced EOD Robotic System (AEODRS) efforts, so adoption is expected to rise quickly.

Summary: JAUS started as an ad hoc working group sponsored by the JGRE in the late 90s, and published several versions the JAUS Reference Architecture. In 2004-08, the working group transitioned to the AS-4 committee and moved toward a service-oriented architecture approach. In each publication, the focus of the standard is unambiguous logical interfaces between platforms, payloads and controllers to promote interoperability, generally through the definition of data packets and their expected sequencing on a network. However, as JAUS does not define physical or electrical interfaces, it cannot be used as a stand-alone solution to interoperability. Furthermore, JAUS often introduces a number of options for network architecture and implemented services, which may prevent interoperability between two devices that implement JAUS services but use different network options. Even so, JAUS has been used by multiple DoD Programs of Record with varying levels of success. It should be noted that the standards body is somewhat slow and deliberate, opening itself to criticism of not supporting interfaces for new technologies as they become mainstream.

UGV IOP

Overview: The UGV IOP is a Government controlled standard that extends JAUS to specify custom logical (messaging), physical and electrical interfaces for interoperability of unmanned systems using industry participation.

Managing Body: PM-Force Projection, PEO CS&CSS (formerly RS-JPO)

Management Type: Government-led task groups which incorporate technical recommendations from industry participants. Each industry organization gets a single vote, but the publication of the documents is strictly controlled by Government.

Modification Process: The scope of each version is established early in the development cycle by the Government through the Capabilities Plan. This document provides high-level guidance only, leaving considerable breadth for the technical Working Groups, comprised of Government and industry SMEs. The Working Groups develop and refine input to the IOP documents, which are then balloted and published. A full IOP version cycle lasts approximately 18 months.

Business Model: The RS-JPO (now PM-Force Projection) funds several Government personnel and a limited number of contractors to work full or part-time on the IOP. Industry participation in the Working Groups and during the review/ballot process is at the expense of their organization.

Availability: DoD Distribution A

Reference Implementations: There are no official or endorsed implementations or tools, but since the IOP is based on JAUS, multiple third party tool sets exist.

Tools for Compliance Testing: The United States Army Tank Automotive Research, Development and Engineering Center (TARDEC) stood up a UGV Interoperability Lab for compliance evaluation, and has a limited release Conformance Verification Tool in development for self-evaluation. Funding for this effort was Government based.

Certification Authority: Acquisition programs must conduct their own validation against their unique "IOP instantiations". The TARDEC Lab expects to act as a validation authority, hired on behalf of Government or commercial organizations. The output of the testing process is currently undefined, and may fall short of offering full certification.

Degree of Adoption: The IOP has been used in multiple small programs, and will be included in the Request for Proposals (RFPs) for at least two near-term Programs of Record (PoRs). Industry participation has been moderate, with major platform providers and multiple small businesses actively involved. Interest from large Prime Integrators has been muted to date, but expected to increase once larger scale programs are launched. The DoD has stated that the IOP will be the basis for interoperability for all unmanned ground vehicle acquisition programs going forward, so adoption is expected to rise quickly.

Summary: The Interoperability Profile effort seeks to extend JAUS to a full-suite of interfaces suitable for current and near-term unmanned systems programs. While it uses published JAUS services as a baseline for logical interoperability, it offers several additional options for standardized physical and electrical interfaces. Furthermore, it defines unique custom services to deal with capability gaps not covered by JAUS, but required for existing programs. These custom services are based on the JAUS Service Interface Definition Language (JSIDL) and may be adopted by the SAE AS-4 community in the future. The IOP also attempts to reformulate the JAUS services into selectable "Attributes", corresponding roughly to functional capabilities, which may be selected during the acquisition process. The selection of specific Attributes and Attribute Options for a particular program is referred to as an "IOP Instantiation", and becomes the basis for an Interface Control Document (ICD) as well as compliance testing. Similar to JAUS, however, if different programs select different Attributes and therefore differ in their Instantiations, the results may not be interoperable. Consequently, the IOP can be considered an enabler of interoperability with more focus than pure JAUS, but does not guarantee it across all programs. Future versions of the IOP may potentially profile other existing standards, such as the J1939 CAN or others for tactical wheeled vehicle autonomy.

VICTORY

Overview: The Vehicular Integration for C4ISR/EW Interoperability (VICTORY) effort is defining a network-based architecture and set of interface standards for integrating electronics systems ***inside*** of military ground vehicles. The standards include logical (messaging) protocols and physical (signaling) specifications, a shared processing API specification, and an optional physical connector / pin-out specification.

Managing Body: The VICTORY standards support office (VSSO), a group that is lead and funded by the Government, manages the standards. The VSSO includes a core team of government employees, supported by a small set of contractor SMEs. The VSSO hosts and staffs the leadership of a standards body, which includes government and commercial participation. The VSSO leads the development and maturation of the architecture and standard specifications, and provides supporting tools including a reference implementation and the compliance test suite.

An executive steering group (ESG) sets the direction and priorities, and provides funding for the VICTORY effort. The Program Executive Office for Ground Combat Systems (PEO GCS) is the managing partner on the VICTORY ESG, which also includes PEO CS&CSS, PEO Command, Control and Communications-Tactical (C3T), PEO Intelligence

Electronic Warfare & Sensors (IEW&S), United States Army Communications-Electronics Research, Development and Engineering Center (CERDEC), and TARDEC.

Management Type: Government-led task groups incorporate technical recommendations from industry participants. Change proposals are formed and finalized through a consensus building process, and managed by a government-paid core technical team. The Government controls the publication of the documents, and its distribution statement is "D" due to some export-controlled and other sensitive information in the specifications.

Modification Process: Within the VSSO is a change control board (CCB) that manages changes to the specifications. The standard specifications document is updated roughly every 6 months, with additional component type specifications (grouping of interface specifications) and modifications to existing specifications. When a specification has matured and is fully supported in the compliance test suite, then it is marked as a "proposed standard" level. Once matured, it is considered ready for adoption by programs. Once a specification is at this level of maturity, the VICTORY CCB manages any subsequent change, and ensures that any specification that has been adopted has long-term support in the documentation and the tool-set. The technical team works to ensure that different versions of a specification that has been adopted actually still interoperate. The inter-version interoperability concept includes both an earlier client interoperating with a later service, and a later client interoperating with an earlier service within the capabilities that are common to both versions.

Business Model: The organizations that make up the ESG provide funding to the VSSO, which funds the core technical team to lead the standards body, document and mature the specifications into standards, and to develop supporting artifacts for the community. The VSSO assists PMs within the acquisition community in developing requirements language to put into acquisition documents that specifies compliance and defines acceptance criteria for compliant components. The VSSO also assists commercial organizations in interpreting the specifications, and in using the provided artifacts to interpret, develop against, and test compliance with requirements. The VSSO explicitly does not test or certify compliance of, or recommend use of particular products.

Availability: The VICTORY architecture, standard specification, reference implementation, and compliance test suite are all DoD Distribution D, but are open and free of charge otherwise.

Reference Implementations: The VSSO provides a reference implementation for each component type specification, which is updated with every major release of the specifications. As compliance tests are developed, the reference implementations are tested for compliance, and updated until they pass the compliance tests. The compliance test reports are provided with the reference software library as they are completed. This VSSO reference implementation is provided to the community on the VICTORY portal. Instructions for achieving access to the VICTORY portal are available on the VICTORY public web site [7].

Tools for Compliance Testing: An effort was made to ensure that the VICTORY standard specifications are "testable" by providing the community (both government and commercial organizations) with tools that support compliance testing. These tools are referred to as the compliance test suite (CTS). The CTS includes a specification, compliance test plan, compliance test report template, and a compliance test tool plug-in for each component type specification. Compliance test plans provide the measurements to be acquired and the acceptance criteria for determining compliance with the specifications. The compliance test report templates are a machine-readable format for documenting compliance. The compliance test tool (CTT) is the main interface of the community for VICTORY. The CTT can be used to view the specifications, compliance test plans, and compliance test reports. It also automates a large part of the test procedures and automatically generates compliance test reports.

Certification Authority: There is no certification authority set up for VICTORY. Various government and commercial organizations are setting up facilities intended to provide independent, third party compliance testing services (for a fee). The use of a third party compliance testing facility to test and document compliance of a product is up to the PM, but is not required by the VSSO. The VSSO defines compliance requirements in the compliance test plans, and defines the evidence necessary to support a compliance claim in the compliance test report templates. The compliance test tool is provided to the community, and it generates compliance test reports that can be used as evidence of compliance.

Degree of Adoption: VICTORY has defined a set of capabilities that are considered necessary to be implemented on each vehicle for the communities to benefit from the architecture. These are called the "core in-vehicle network (IVN)". Each vehicle program in the PEO GCS and PEO CS&CSS has been asked to develop a strategy and timeline for implementing the core IVN in their vehicles. The programs that currently have an engineering change proposal (ECP), or which are currently in development, have included various levels of the core IVN capabilities for near-term development. Programs that are not currently in development have longer-term plans for implementation. The government furnished equipment (GFE) PEOs have also developed strategies for how they plan to leverage the capabilities of the vehicles, as well as when they plan to provide capabilities that are related to their products. Overall, implementation of the VICTORY standard specifications is underway between the vehicle and GFE PMs, with the earliest deployment targeted for the fiscal year 2017 timeframe.

Summary: VICTORY is providing a network-based architecture, standard interface specifications, and a set of tools to support implementation, specification, and compliance testing for the standards, which define how components and sub-systems will interoperate within vehicles. It is a government lead and funded effort that includes a broad array of government procurement, government research and development, and commercial organizations. VICTORY is currently being implemented across the Army ground vehicle community, and will be fielded in the FY 2017 timeframe. Additional information can be found on the VICTORY portal [7].

AEODRS

Overview: Advanced Explosive Ordinance Disposal Robotic System (AEODRS) is a Government run program that defines an open architecture such that sub-systems can be designed by several contractors and integrated by a Prime Systems Integrator. This architecture definition not only defines the interfaces to payloads, but standardizes the interfaces between core modules within the base platform. The AEODRS definition builds upon and heavily references the SAE JAUS AS4 standards.

Managing Body: Naval Surface Warfare Center Indian Head Explosive Ordinance Disposal Technology Division (NSWC IHEODTD)

Management Type: The Government, with funded support from Johns Hopkins University – Applied Physics Lab (JHU APL) created, manages, and publishes all documentation associated with the program and architecture definition. Program updates are published to fbo.gov and interested parties are required to submit key company information, such as CAGE code, in order to be considered for reception of the documents. The first increment of the program included a pre-Production Representative Module (pPRM) phase run by JHU APL to refine the requirements.

Modification Process: The specifications were initially produced and managed by the Government and its direct contractors. All changes to the specifications or system configuration are controlled by the Prime System Integrator (PSI) until Critical Design Review (CDR) for a given increment of the program. After the CDR, the Government directly controls all configuration changes. In this case, the Program PSI must prepare a formal ECP, classify the change as major or minor, and submit to the Government. The Government manages the process of reviewing, approving, changing, and distributing the specifications.

Business Model: AEODRS is a government Program of Record. The development and management of the specifications are funded by the Government.

Availability: DoD Distribution C – Available to U.S. Government Agencies and their contractors.

Reference Implementations: Full reference designs exist for the Power Capability Module and the Autonomous Behaviors Capability Module. These reference designs are maintained and distributed by JHU APL. These reference designs include printed circuit board (PCB) schematics and physical layouts, mechanical models, software source code, and software executable files.

Tools for Compliance Testing: JHU APL maintains an AEODRS simulation tool which allows individual Capability Modules (CMs) to be tested against virtual CMs in a simulated environment. Additionally, JHU APL maintains a test bed for use by qualified potential module providers. Neither of these testing options provides a certification of compliance.

Certification Authority: No official certification authority currently exists for AEODRS.

Degree of Adoption: AEODRS has been directly adopted by a number of potential CM suppliers who wish to participate in the program. AEODRS preceded and informed IOP, and IOP has accommodated AEODRS such that AEODRS can be considered an instantiation of IOP.

Summary: The AEODRS program seeks to develop a family of Explosive Ordinance Display (EOD) robotic systems which utilize a Government defined open architecture. The AEODRS family consists of 3 robots, referred to as Increment 1 through Increment 3 in order of increasing size. Each system is broken up into a number of Capability Modules, each of which could be provided by different organizations/vendors. In addition to the published JAUS services, custom services are added to promote logical interoperability. Furthermore, detailed electro-mechanical interfaces and constraints are provided for each module to ensure physical and electrical compatibility. Each Capability Module is defined by a detailed ICD and a Module Performance Specification which define not only the interfaces necessary for module-level interoperability, but also a required performance level for each module. Additionally, system-level performance specifications are provided for the program. Significant commonality exists between AEODRS and IOP; however AEODRS provides a more focused, more rigid interface definition for a specific PoR. AEODRS does not attempt to provide standards for capabilities outside the scope of its family of 3 EOD robotic systems.

ROS

Overview: Robotic Operating System (ROS) is a set of software libraries and tools focused on robotics. ROS provides a number of out-of-the-box hardware drivers, software modules, and bridges to other popular software packages (Gazebo, OpenCV, etc).

Managing Body: Open Source Robotics Foundation (OSRF)

Management Type: General management is provided by OSRF with community input driven through message boards, forums and various events (ROSCon).

Modification Process: New capabilities (packages) are developed by the community and shared through standard open-source processes (downloads, repositories, etc). Popular packages can be pulled into the main release schedule and are then maintained by either the original author or another contributor called a “maintainer”. Source code management is done using code repositories, issues trackers, and forums.

Business Model: As an open-source project, development of ROS comes from a wide range of sources, both commercial and academic. Since 2013 the OSRF has had primary stewardship of ROS. The OSRF is supported through corporate sponsors that include DARPA, Google, Bosch and more.

Availability: Freely downloadable from ROS.org, distributed under a BSD open-source license.

Reference Implementations: There is a reference implementation of ROS (currently called Indigo) available for Linux (Ubuntu) through the ROS.org website.

Tools for Compliance Testing: There are freely available tools for development and evaluation of ROS software. There is no defined compliance or certification for ROS nodes and no standardized set of logical interfaces for testing. Each ROS system may have drastically different message sets for the same functional capabilities.

Certification Authority: None.

Degree of Adoption: Wide-spread. ROS has seen significant growth in the robotics market over the past 5 years. This is especially true for programs and institutions focused on research. Many high-profile research programs (such as DARPA ARM-S and the DARPA Robotics Challenge) mandate ROS for use in the systems for interoperability and to leverage existing tools and capabilities.

Summary: ROS started as an extension of research conducted at Stanford University. Willow Garage took that work and extended it into the market through the sales of the PR2 robot. ROS is focused on collaborative robotics software development, pulling on the expertise of the wide community to build solutions to one problem at a time. ROS has emerged as the go-to solution for many researchers in robotics, especially in academia. However, ROS lacks strong standardization of the logical interfaces. ROS supports a common message definition format (ROS .msg files) which defines software for serialization of message content. The format, structure and content of ROS messages, however, are not standardized across the ROS ecosystem. Therefore there is no easy path to interoperability using ROS unless the systems and modules were designed to use the same message set from the beginning.

ORAV

Overview: On-Road Automated Vehicle Standards Committee (ORAV) is an open, commercial standards committee that defines terms (vocabulary) and safe testing guidelines for commercial ground vehicles on public roadways, such as passenger cars.

Managing Body: SAE

Management Type: An open committee made up of individual subject matter experts from industry and Government of any nationality. Voting is reserved for committee members, where membership is solely the discretion of the committee chair.

Modification Process: A Document Sponsor creates or modifies a potential Standard, which is balloted by the Committee. Configuration Management for the documents is provided by SAE.

Business Model: Interest groups and “customers” of the reference architecture include military & contractors, automakers & suppliers, truck companies & heavy equipment (including mining & agriculture), academia and related standards bodies (ORAV human factors, J2735, J1939 & AS4 committees, FlexRay use case committee.)

The primary contributors include engineers and scientists from industry, government, and academia with expertise in systems engineering and vehicle communications architectures. Generally, however, participation by industry SMEs in SAE committees is at the expense of their organization.

Availability: The documents can be purchased by any person or organization worldwide without restriction for a modest fee (approximately $60 per document).

Reference Implementations: In development. At present, only a terminology document is published.

Tools for Compliance Testing: TBD. At present, only a terminology document is published.

Certification Authority: TBD. At present, only a terminology document is published.

Degree of Adoption: Unknown.

Summary: The initial goal of the ORAV committee is to publish an Aerospace Information Report, not a Specification, which defines the levels of autonomy for on-road automated vehicles. In addition, a current work-in-progress provides guidelines for safely testing these systems in a real-world environment, but does not specify standard test procedures or processes. While the overall scope of the committee’s charter includes interoperability for on-road systems, it did not appear to be a near-term focus until a small contingent of IOP industry participants approached the

ORAV committee with some of the work performed under the Autonomous Mobility Applique Systems (AMAS) project. This material does address some of the logical interoperability needs for on-vehicle communication by defining a set of J1939 messages. The SAE ORAV Committee stood up a task force to propose one or more SAE documents for publication based on this work.

The output of the task force will include use cases, requirements, and a modular functional architecture that identifies and defines interfaces between the modules at a logical level including the definition of logical message sets at the interfaces between modules. The final product will include architecture and message definitions that will inform other standards bodies. In addition the task force will offer recommendations for updates to JAUS (SAE AS-4) and the SAE J1939 committees. They also intend for groups to liaise with other groups (Vehicle Architecture for Data Communications, Functional Safety, DSRC, J2735 etc.).

The products will not mandate a single way of building a system nor will it guarantee that the components of a system are designed in a safe/reliable/cyber-secure manner. Rather, it will suggest opportunities for standards that will increase interoperability.

Furthermore, output of this task force will not necessarily result in a new standard. Rather, the report(s) will provide non-normative information or guidance for use in future standards activities. Furthermore, the activity will produce a shared body of knowledge and definitions that will assist in the development of future standards.

DDS

Overview: Data Distribution Service (DDS) is a middleware solution that aims to provide scalable, real-time, dependable and high-performance data exchange using a publisher / subscriber data model. DDS defines a language for describing the data to be shared (data model) and the ***topics*** under which it will be published. The intent is to remove the responsibility of packaging, transporting, and managing data from the application programmer. DDS is managed by the Object Management Group (OMG).

Managing Body: Object Management Group (OMG)

Management Type: The Object Management Group is an international, open membership, not-for-profit technology standards consortium. OMG hosts four technical meetings throughout the year. These meetings give OMG members and interested nonmembers the opportunity to collaborate in a centralized location, learn about technology standards products and processes at tutorials, and attend special information day events on current trending hot topics. While technical meetings provide a centralized location for Task Forces and Working Groups to work together, they are merely checkpoints with the bulk of the work between members taking place electronically via email, teleconferences, and on wikis.

Modification Process: OMG standards are maintained by a Special Interest Group (SIG) assigned to a particular standard. The Data-Distribution PSIG (DDSIG) is a subgroup within the OMG chartered in order to coordinate, guide, and promote the use and evolution of Data-Distribution technology. The SIG mission statement is to:

- Foster cooperation between implementers and users of Data-Distribution technologies.
- Clarify user requirements and coordinate the evolution of the DDS specification, influence related specifications, and catalyze new specifications.
- Identify opportunities to further enhance and integrate DDS with other distributed-computing standards and help develop necessary collaboration / interoperation specifications.
- Educate, guide, and assist the community in the use of DDS technologies.
- Promote and evangelize the use of DDS technologies in the marketplace and seek additional opportunities for the technology.
- Establish and maintain active coordination with appropriate OMG task-forces and external organizations in support of the preceding goals.

Business Model: There is a free implementation of the DDS standard (openDDS), and multiple vendors provide software implementations of DDS along with services for utilizing their libraries. The main driver for leveraging a middleware such as DDS is to make software applications less dependent upon the underlying transports, and to achieve interoperability between software applications developed by different organizations. In order to do this, a program must specify the data model (set of topics to be exchanged) to be used and require that software components utilize the DDS middleware and that set of topics. It is not clear whether such requirements are testable.

Availability: The DDS standards documents are freely available through the OMG website.

Reference Implementations: There are several reference implementations available from commercial entities, both closed-source and open-source. OMG does not sponsor or develop a Reference Implementation. These implementations have been shown to be compliant and interoperable assuming a common topic model, as expressed in a DDS-specific Interface Definition Language (IDL), is used for exchange.

Tools for Compliance Testing: It is not clear if there are methods of testing compliance with the DDS standard. It should be noted that compliance with the DDS standard does not by itself guarantee interoperability between components, as DDS does not standardize the type of data (topics), but specifies how to describe the data and exchange data based on a set of common topics. In order to achieve interoperability, a program must specify both DDS compliance and a set of topics to be exchanged.

Certification Authority: Unknown.

Degree of Adoption: DDS is used in a variety of large-scale distributed systems for both real-time data reporting and less demanding systems which make use of the easy setup and QoS features. DDS has seen use in the Robotics community for a number of projects and bears a similarity to ROS. In fact, with ROS 2.0 (scheduled for 2015), the ROS community is looking to replace their custom data distribution layer with a DDS solution and adopt a hybrid of DDS IDL and ROS .msg formats for topics.

Summary: DDS is a middleware solution that aims to provide scalable, real-time, dependable and high-performance data exchange using a publisher / subscriber data model. DDS is managed by the Object Management Group (OMG). The standard was first published in 2003. DDS simplifies the task of transporting data across networks for a collection of entities. Entities publish data to a topic in which other entities can subscribe to that topic to get updates. The DDS handles the transfer mechanics of encoding data, decoding data, message addressing, etc. DDS features fine and extensive control of QoS parameters, including reliability, bandwidth, delivery deadlines, and resource limits. DDS topics are described using an IDL, which is also managed by OMG. There is no standardized set of DDS topics. Interoperability between different DDS implementations has been shown to work, however common topic models are a prerequisite.

FACE

Overview: The future airborne computing environment (FACE) effort is developing an architecture and framework to promote portability of software components in airborne avionics systems. As FACE is primarily focused on software portability and interoperability, it is mostly a software-centric architecture.

Managing Body: FACE is managed by the FACE consortium, which is hosted by the Open Group™. Member organizations pay a fee to participate and access the work products (documents). There are multiple levels of membership, including Associates, Principals, and Sponsors. The amount of influence increases with the financial commitment. Members include government and commercial organization [1].

The FACE consortium consists of an advisory board and several sub-committees, including enterprise architecture (EA) team, business working group (BWG), technical working group (TWG), and the FACE UCS alignment liaison [9].

The government organizations that represent the "customers" of FACE include NAVAIR and Army PEO Aviation. However, these organizations do not have the power to steer the consortium. Organizations associated with academic institutions participate, but mostly do so as funded representatives of government organizations. For example, the Vanderbilt University Institute for Software Integrated Systems (ISIS) is not a formal member, but is funded by NAVAIR to lead the development of the specification and associated toolsets.

It is notable that FACE is part of the Army common operating environment (COE) effort, and represents the airborne real-time and safety critical (RTSC) computing environment (CE). FACE is not targeting any ground-based platforms.

Management Type: FACE is a (mostly) traditional consortium, in which members pay to join, and have varying influence on the direction and content of the results, depending upon financial commitment.

Modification Process: The TWG is responsible for developing and maintaining the technical specifications and defining conformance requirements. Member organizations have varying levels of control over the contents of the specifications, depending upon the level of membership. According to the frequently asked questions, any member of the Open Group can provide comments during the final review process, which occurs before the Open Group publishes documents [2].

Business Model: The FACE BWG charter is to "Develop, Implement and Communicate Attractive Industry-Government Business Models that Incorporate the FACE Vision and Mission." The BWG is responsible for communicating the FACE consortium goals to DoD and industry, and promoting the use of the FACE standards in procurements. They also define the policies for conformance with the FACE standards. Programs write FACE conformance requirements into RFPs, but it is not clear how these conformance requirements are verified, or what documentation must be provided to support claims of compliance. Conformance verification tools are not yet available, and a certification authority has not yet been set up (see below).

Availability: Most documents are Distribution A (unlimited distribution).

Reference Implementations: At least one reference implementation is being developed with government funding outside of the FACE and Open Group organization. The status of the reference implementations is not known at this point. It is not clear whether the Open Group will manage the reference implementations once they are available.

Tools for Compliance Testing: Tools for conformance testing are being developed, funded by the Government. The tools are not yet available, but otherwise the status of the tools is not known. The management of the tools will likely depend upon whether a verification authority (VA) organization is identified (see below).

Certification Authority: FACE uses the concept of conformance as opposed to compliance, which implies that not only are the interfaces that exist on the component compliant with the FACE specifications, but a component must implement all of the relevant FACE interfaces [3]. Conformant implies that all of the features in the architecture specification are implemented in accordance with the specification, but interfaces may still exist that are not defined by the specification. Compliant implies that all interfaces implemented are as defined by the specification, but it may be that not all interfaces defined by the specification are implemented. This subtlety is not merely academic; conformance is important when describing a framework or operating environment, but compliance is more relevant to components that will leverage a framework, but not be a required part of it. For that reason, requirements for FACE framework components should be based on conformance.

FACE has not yet identified a conformance verification authority, but a request for proposals for candidate VAs has been published. The status of that procurement is not known at this time.

Degree of Adoption: Based on the FACE web-site, FACE is referenced in multiple procurements [4]. Many of those are for technology demonstrations and FACE-related tools, but programs of record are referencing the specification; including Harrier II Upgrade, Army Multi-Function Imagers for Rotary Wing (MFI4RW) Applications, and the United States Special Operations Command (USSOCOM) Remote Readout Unit. It is not clear, however, whether any programs have reached the implementation stage, or when that will happen.

Summary: FACE is developing an architectural framework that leverages layered middleware and model-driven architecture (MDA) with the goal of software portability in avionics systems. FACE is managed by the Open Group FACE consortium, and the Government is providing funding to support the development of the specifications, reference implementations, toolsets, and possibly a conformance verification authority. Reference implementation and tool development are ongoing, and there has been an RFP indicating that a conformance verification authority may be set up.

FACE a layered middleware architecture, in which software interfaces are defined at multiple levels so that the software at each layer can be made independent of the layers below and above; a concept known as "platform independence". FACE defines interfaces to multiple segments (or layers), including; operating system, input / output services, platform specific services, transport services, and portable components segments. FACE is different from traditional middleware architectures which define fixed application programming interfaces (APIs) to each segment. FACE also leverages the MDA, a specific flavor of model-driven development (MDD) promoted by the OMG, to create a modeling framework for modeling data and component interfaces, and generating software and configuration files. The modeling framework is based on an avionics software domain specific modeling language defined using the OMG meta-object framework (MOF), and tools in which designs are modeled in terms of the FACE modeling language. FACE requires most software components to use the FACE Transport Services API for communication, which requires FACE data models for the data sent/received.

Other Efforts

Additional standards, programs, and efforts that could have relevance to this effort, but which are not included in the current scope of this document include:

- Multi-robot Operator Control Unit (MOCU)
- NATO Standardization Agreement (STANAG) 4586
- UAS Control Segment (UCS)
- UAS IOP
- Connected Vehicle Reference Implementation Architecture (CVRIA)
- http://www.iteris.com/cvria/index.html
- IEEE RAS Map Data Representation WG and IEEE's RAS Ontology Working Group
- (http://www.ieee-ras.org/about-ras/governance/industrial-activities-board/standards-committee)
- MOOS (http://www.robots.ox.ac.uk/~mobile/MOOS/wiki/pmwiki.php/Main/HomePage)
- *Control Architecture for Robotic Agent Command and Sensing (*CARACaS)
- (http://www.techbriefs.com/legal-footer-127/3251-npo-43635)
- 4D/RCS
- Autonomous Capabilities Suite (ACS)
- (http://www.public.navy.mil/spawar/Pacific/Robotics/Pages/ACS.aspx)

Analysis

The information provided above serves to illustrate some of the major efforts to apply standards to the RAS and related domains. The domain of military RAS is as broad and diverse as the many standards that attempt to quantify it.

IOP is in many ways similar to some of the other standards presented. However, it also differs from some in terms of scope, goals and direction.

Some of the standards discussed above are not in-fact standards at all, but instead rely on de-facto standards and commonality, which is coordinated among specific vendors (such as DDS and ROS). Others are focused on a very specific use case and do not provide enough coverage for many RAS topics; for example VICTORY is focused on communication within a vehicle system without regard to interoperability for C2 links, sensor data or ISR Networks. Still others, such as JAUS, are part of the IOP standard by incorporation.

The goal of IOP is not in contrast with the goals of the standards above; in-fact, many of the other standards and efforts can be used within IOP by inclusion if appropriate (for example JAUS, VICTORY, DDS, etc.). IOP's focus is on providing well-defined attributes which can be down-selected to define an IOP Instantiation with unambiguous interfaces for both hardware and software interoperability. This approach enables the IOP to select existing, and develop new best practices and standards with respect to a given domain, class, or purpose for the RAS.

RECOMMENDED PATH FORWARD FOR IOP

For IOP to fulfill the promises of a fully operational MOSA, continued work is required to further develop, fully validate, demonstrate, and harden the current operational prototype standard. Infrastructure must be developed and implemented to transition the initial IOP into a readily accessible, open standard. The National Advanced Mobility Consortium (NAMC) was tasked by the Office of the Secretary of Defense (OSD) Joint Ground Robotics Enterprise (JGRE) with developing a strategy for how Government and industry might work together to further advance the current IOP standard.

Accordingly, the NAMC formed a team of MOSA and RAS subject matter experts (SMEs), organized a group of Government and industry stakeholders, and engaged a NAMC Community of Interest (COI) to devise a recommended strategy. The team constructed and documented a set of goals, assumptions, and guidelines for a "business model", performed a survey of existing efforts that develop modular open systems architectures for interoperability, performed analysis of the available approaches to identify the model that aligns best with the IOP and the goals of the RAS domain, then developed a recommended strategy. The proposed strategy, described herein, calls for phasing IOP into a next, ***Iteration and Maturation*** stage of development in order to mature the standard, propagate its use, and develop and implement the tools, processes, mechanisms, and infrastructure needed for IOP to subsequently evolve and transition into a fully-supported, broadly adopted, and widely accepted operational standard in both Government and commercial markets.

Figure 2. Proposed Stages for Deploying IOP

The ***Iteration and Maturation*** stage is expected to last three to five years, depending on several factors, including the availability and timing of the required investment. It is assumed that that IOP will remain predominately DoD-centric during the initial parts of this stage. Efforts will focus on achieving a number of key objectives:

- Transitioning from Government-only to dual Government/industry ownership and control of the IOP standard, and implementing a formally managed standards process
- Continuing to develop, harden, and extend the IOP interfaces, specifications, and documentation and releasing new versions
- Creating tools and processes that will assist program managers and the vendor community to understand and use as well as specify, document, and verify compliance with the standard
- Demonstrating and validating its utility and value by using it to develop and deploy a large number and wide range of interoperable RA-GVS
- Garnering wider community support and investment, propagating its adoption as the de facto, interoperability standard for all RA-GVS across the services, and determining the potential commercial market interest in adopting and utilizing the standard
- Developing a long term plan to subsequently transition IOP, upon conclusion of the Iteration and Maturation stage, to a fully-supported, widely accepted, operational standard

Summary of Recommendations

It will take time to garner support for the proposed organization and to identify funding to support it. Considering the scope of the effort to transition IOP from Government to joint Government/industry control, separate short- and long-term recommendations have been made. By implementing the recommendations contained in this report, the team believes that the RAS community can further transition IOP into use and reap the benefits promised by modular open systems architectures: increased interoperability, rapid innovation, lower life-cycle costs, and enhanced capabilities in the field.

Figure 3 provides a graphical representation of the proposed government structure to support the transition and adoption of IOP.

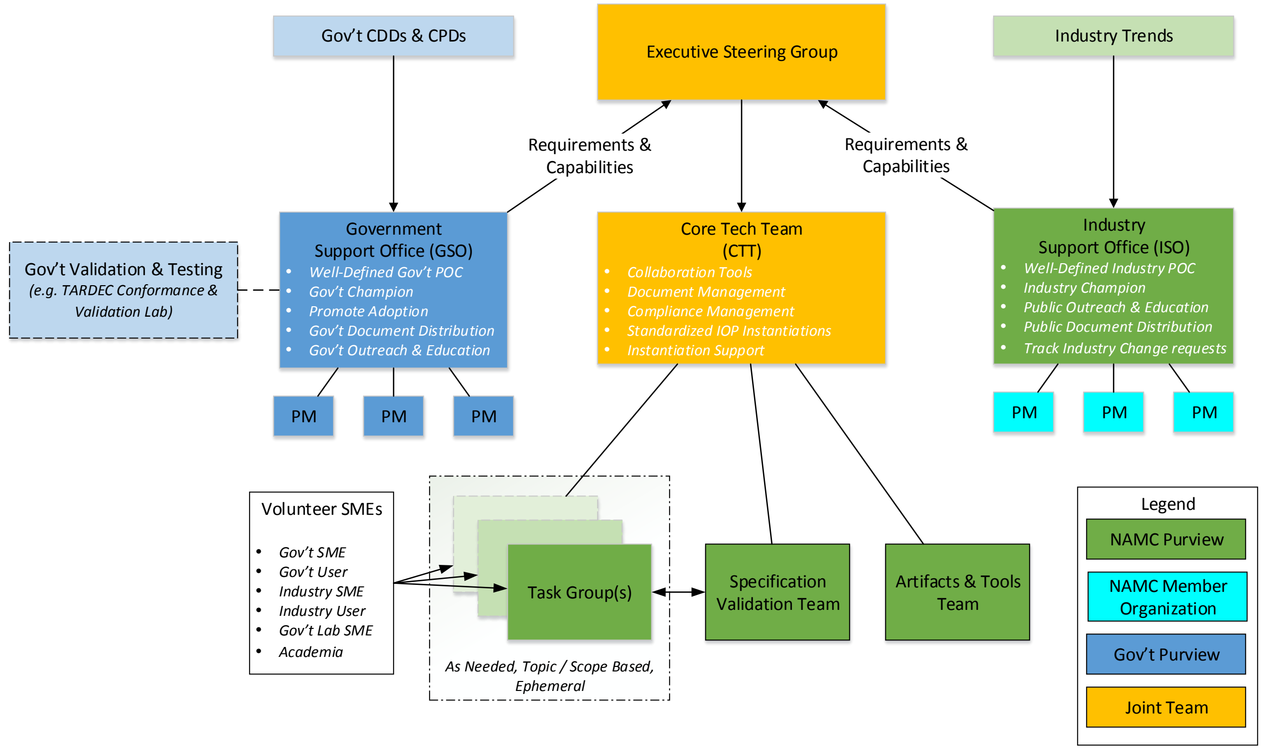

Figure 3. Proposed IOP Governing Structure

Long Term Activities

A number of long term activities are proposed for the purpose of transitioning IOP to the business model described within this document. The recommended long term activities to transition the IOP to the proposed business model include:

1. Establish a Joint Government/Industry Governance & Management Structure

The Executive Steering Group will provide high-level leadership and guidance for the overall effort by prioritizing needs, establishing direction, and managing funding. The recommended structure includes an Executive Steering Group consisting of representatives from both Government and the industry community as represented by the NAMC (i.e. the industry representatives would be appointed by and function under the auspices of the NAMC). One of the primary motivations for moving to a joint Government/industry structure is to facilitate the potential future adoption and use of IOP in commercial markets.

2. Invest in the Support Infrastructure Required to Propagate the Standard

Under the guidance and management of the Executive Steering Group, develop and implement the structure, resources, and processes required to promote and support the standard. This would include two organizations:

<u>Government Support Office (GSO)</u> – responsible for championing IOP to Government PMs, technical managers (TMs), and organizing and maintaining a Government COI. The GSO would be responsible for undertaking various activities intended to promote and propagate the use of IOP, including:

- Educating TMs, PMs, and the COI on IOP and facilitating access IOP documentation and tools.
- Assisting and supporting PMs and TMs efforts to use IOP to define their own instantiations, and establish compliance processes.
- Working with PMs, TMs, and the COI to obtain their input and feedback on future IOP enhancements.

<u>Industry Support Office (ISO)</u> - responsible for:

- Promoting IOP to NAMC members, educating them on IOP, and garnering their support.
- Organizing and maintaining an industry COI from among NAMC members.

- Providing the industry COI with access to the IOP materials, tools, and documentation.
- Providing rudimentary support to members of the COI.
- Soliciting and obtaining input and feedback from industry on desired enhancements and improvements to IOP.
- Publicly championing IOP and providing public access to certain IOP materials.
- Soliciting interest and support for IOP from companies involved in commercial markets.

3. Invest in the Technical Infrastructure Required to Advance the Standard

Under the guidance and management of the Executive Steering Group, develop and implement the structure, resources, and processes required to further mature the standard. Specifically, it is recommended that responsibility for all technical aspects of IOP be transitioned to a **Core Technical Team (CTT),** consisting of both Government and industry personnel. The CTT will be responsible for:

- Developing and validating new versions and releases of IOP.
- Defining and supporting a set of standard instantiations for specific classes of systems.
- Producing the software tools needed to develop and test for compliance with program-specific instantiations.
- Providing a web-based portal where both Government and industry members of the IOP community can access the documentation and tools and exchange information.
- Other related tasks and activities.

Short Term Activities

In addition to the long term activities proposed above, the team recommends a set of steps that should be taken in the near term to lay the pathway and make the case for executing the recommended plan. The recommended path forward for the near term is for the Government to provide the funding needed to:

1. Garner the Required Government & Industry Backing

Begin now to socialize and build consensus among both Government and industry leaders on the concept of transitioning IOP to joint Government/industry ownership and control. Further research the options for doing so, and develop a more detailed plan for how to ramp up the investment in IOP and establish the recommended infrastructure. Commence the effort to budget and secure the required Government funding; based on direct experiences and lessons learned from other efforts, unless the leadership of the support offices and the core technical are funded, it is unlikely that the recommended strategy will succeed.

2. Invest in the Support Infrastructure Required to Advance the Standard

Establish industry and government champions who will serve as the Industry and Government Support Offices in the near term. It is recommended that the GSO come from within the IOP group currently operating under PM Force Projection and that the pilot industry support office function under the NAMC.

3. Develop Baseline IOP Instantiations

Form a dual Government/industry working group to define and maintain standardized IOP instantiations for one or more classes of RA-GVS for research and development purposes.

4. Investigate IP, Copyright, and Distribution Concerns

Investigate any IP, copyright, and distribution issues related to current IOP and test tools (e.g. the TARDEC Conformance Verification Tool (CVT)) that would need to be addressed prior to transition to the joint Government/industry organization.

5. Set up a Website to Facilitate Distribution of IOP Materials

Set up a Website to distribute IOP materials and facilitate a discussion forum for the Government and industry COIs.

6. Ownership of IOP Working Groups

Transition responsibility and administration of the current industry working group which supports the IOP to the NAMC.

7. On-going, Iterative Development of the IOP

Continue developing IOP, including commencing work on V3.0

CONCLUSIONS

This paper has presented an in-depth analysis that was performed to create a strategy for transitioning IOP from a set of interface specifications that has been developed by TARDEC into a fully adopted standard that will promote interoperability in military ground and autonomous systems, and potentially in industrial applications. The recommendations are phased in a way that significant progress can be made in the near term with a moderate level of funding, and incremental progress can be made toward the goal of modularity and interoperability in these systems.

REFERENCES

[1] Clifton A. Ericson, II (12 April 2011). Concise Encyclopedia of System Safety: Definition of Terms and Concepts. John Wiley & Sons. p. 272.

[2] http://www.itu.int/en/ITU-T/ipr/Pages/open.aspx, accessed 20 February 2015.

[3] Oxford Dictionary

[4] Merriam Webster Dictionary

[5] SAE AIR5665, Revision B. . "Architecture Framework for Unmanned Systems". . Feb 5, 2015.

[6] UGV Interoperability Profile (UGV IOP) – Appliqué Profile Version 2 Draft for Ballot.

[7] http://www.victory-standards.org

[8] http://www.opengroup.org/face/member-list

[9] http://www.opengroup.org/face/structure/subcommittee-charters

[10] http://www.opengroup.org/face/faqs

[11] http://pubs.opengroup.org/architecture/togaf9-doc/arch/chap48.html

[12] http://www.opengroup.org/face/procurement-news