AUVSI's Unmanned Systems North America 2005
June 28-30, 2005
Baltimore Convention Center
Baltimore, Maryland, USA

JAUS Compliant Systems Offers Interoperability across Multiple and Diverse Robot Platforms

M.N.Clark
Harris Corporation, GCSD
Palm Bay, Florida 32901

Abstract- **The Joint Architecture for Unmanned Systems (JAUS) uses a constrained message passing protocol to affect interoperability among diverse systems and components within the subsystem. Two collaborative experiments separated by eight months successfully demonstrated interoperability and dynamic payload registration between dissimilar unmanned ground vehicles from different vendors. Today JAUS transitions with documentation and its members' full support to SAE aerospace standards technical committee AS4 (Unmanned Systems). Since the conception of JAUS in 1995, industry, government, and academic institutions have discovered functional and economic benefits for this interoperability framework.**

Keywords- bandwidth, compliance, dynamic registration, message transport, network environment, payload, technology independence

Introduction

The demand for diverse unmanned vehicles is increasing, as will costs to repair, maintain, and operate those vehicles. The US DOD has published a vision to have 15% of its ground transportation performed autonomously by 2015. By 2020 the US target is 33% of military aircraft be unmanned [1]. The US Coast Guard has a 20-year program to spend $17 billion on an integrated Deepwater System [2] whose focus is on system interoperability. If unmanned vehicles (UV) could share standard command & control messages, procuring agencies could select components from a variety of vendors reducing costs and increasing availability. It is difficult and expensive to integrate systems into an overall infrastructure for command and control. Today's systems should be capable of interoperating and registering their capabilities outside their OEM family. Vehicles designed today are one of a kind. The engineering involved is significant and drives up the costs for design, deployment, and maintenance. A limitation to interoperability is "in-house" developed communication messages. When UVs share a common message set, owner-operators can chose integrating payloads and control units from multiple vendors. In sharing communication architecture, UVs entering the operations space can choose to register their capabilities to a number of potential controllers thereby increasing chances for survivability. JAUS defines a common language enabling internal and external communication between unmanned systems. Vendors who collaborate on joint robot projects find less integration issues when communications are well defined [3, 4].

Origins of JAUS

The JAUS concept began in 1995 as Joint Architecture for Unmanned Ground Systems (JAUGS). By 1996 a JAUGS Working Group (WG) was established along with an initial set of requirements. The WG held its first JAUGS meeting on December 18, 1997 at the National Institute for Standards and Technology (NIST) in Gaithersburg, MD. The most significant accomplishment in 1998 was inviting industry and academia along with all agencies affiliated with the Joint Robotics Program for membership [5]. The WG examined the NIST Neutral Messaging Language (NML), the Society of Automotive Engineers Generic Open Architecture (GOA), and the Multiple Robot Host Architecture (MRHA) for inclusion and/ or reference by JAUGS. In 1999, NIST made significant contributions on NML as did the University of Florida with their MAX architecture [6]. MAX provided the concept of "wrench"; a six-degree of freedom command for mobility and message-based approach to interoperability [7]. By September 2000, The Office of the Under Secretary of Defense embraced JAUGS by clarifying roles and responsibilities among the Joint Robotics program managers [8]. In that month, the WG released version 2.0 of the JAUGS Reference Architecture Specification [9].

First Demonstration

The JAUGS working group experienced considerable membership growth in 2001. The Air Force Research Laboratory hosted a May JAUGS demonstration that included four unmanned systems from three organizations. The demonstration included semi-autonomous and high-speed vehicles with manipulators, video, weapons fire, and payload delivery. This force-protection mission demonstrated JAUGS exclusively on two of four vehicle systems, the AMRADS[1] and Prairie Dog. There were eight computers talking over wireless Ethernet. The demonstration had two JAUGS OCUs for safety issues but operators controlled vehicles with one. The third vehicle became JAUGS compatible 30 days after the demonstration [10, 11].

JAUGS → JAUS: Domain Expands

On 29 August 2002 the Office of the Under Secretary of Defense formally changed the name JAUGS to JAUS. This established JAUS as an upper level design for the interfaces within the domain of unmanned systems. Also in 2002, the JAUS WG created committees for Compliance and Transport Layer documents. The U.S. Army began efforts in requiring JAUS for Future Combat Systems (FCS) [12].

With growing membership and direction from the Joint Robotics Program in 2003, JAUS expanded its development of architectural elements for high-level control and intelligent behaviors. The Operator Control Units and Payloads committee (OPC) formed to address commonality issues. This collaboration of industry, government, and academia finished 2003 with the first of three interoperability experiments [13, 14]. The JAUS membership received a presentation and white paper concerning the issues of bandwidth [15].

---

[1] Advanced Mobility Research and Development System (AMRADS)

In March 2004, a JAUS based vehicle participated in a race covering 142 miles of California and Nevada deserts roads [16]. Since no team won the challenge, DARPA will conduct its second Grand Challenge on October 8, 2005.

JAUS → SAE AS-4

An important event in 2004 was the Society for Automotive Engineers (SAE) accepting JAUS as an aerospace technical committee titled AS-4. Its charter is to develop standards for unmanned systems. All JAUS member organizations have transitioned to SAE and commit to a well-defined migration plan. Another JAUS accomplishment was completing a second interoperability experiment [17, 19]. The success criterion was the "ABC" test. A test configuration could be any OCU coupled with any platform that in turn recognized or controlled any payload, [Fig.1]. If we start at the bottom of figure 1 for example, the experiment allowed selecting API[2] OCU and commanding the University of Florida's UGV Mule that in turn, had iRobot's cable spooler as its payload. The OCU will recognize the platform's capabilities and effect actions; including control of the payload. Any of the four OCUs could perform this test.

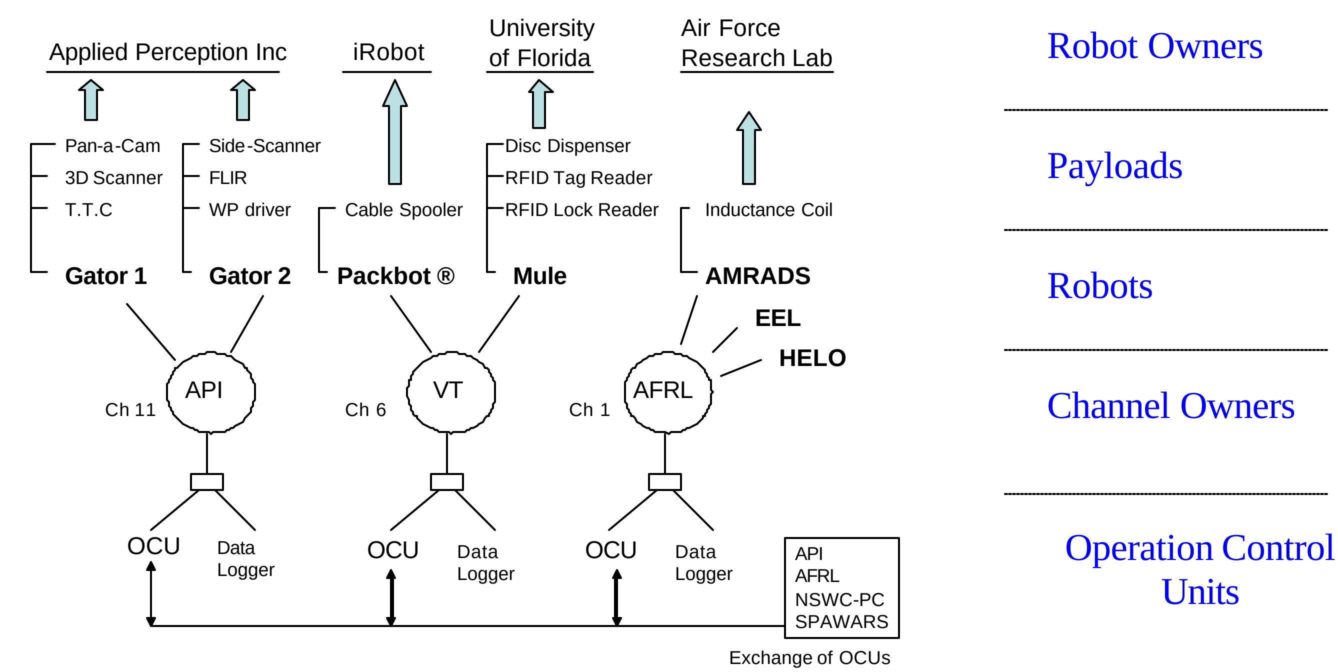

Figure 1 Framework for Payloads, Robots, and OCU Interoperability

Compliance

JAUS compliance is verification that message traffic between elements meets the Reference Architecture Specification [23]. JAUS is a message-based architecture and lends itself well to compliance testing. Elements have three levels of compliance [20]. The Domain Model [24] describes this compliance graphically in Figure 2.

---

[2] Applied Perception Inc (API)

- Level I – Inter-Subsystem
  The requirements between subsystems (i.e., Robot to Robot, Robot to Controller, or Controller to Controller): The purpose of Level I is to support the interoperation of subsystems.

- Level II – Inter-Nodal
  The requirements between nodes (i.e., payload to payload, payload to on-board controller): The purpose of Level II is to support the interoperation of nodes.

- Level III – Inter-Component
  The requirements between components (i.e., component to component): The purpose of Level III is to support software source-code reuse.

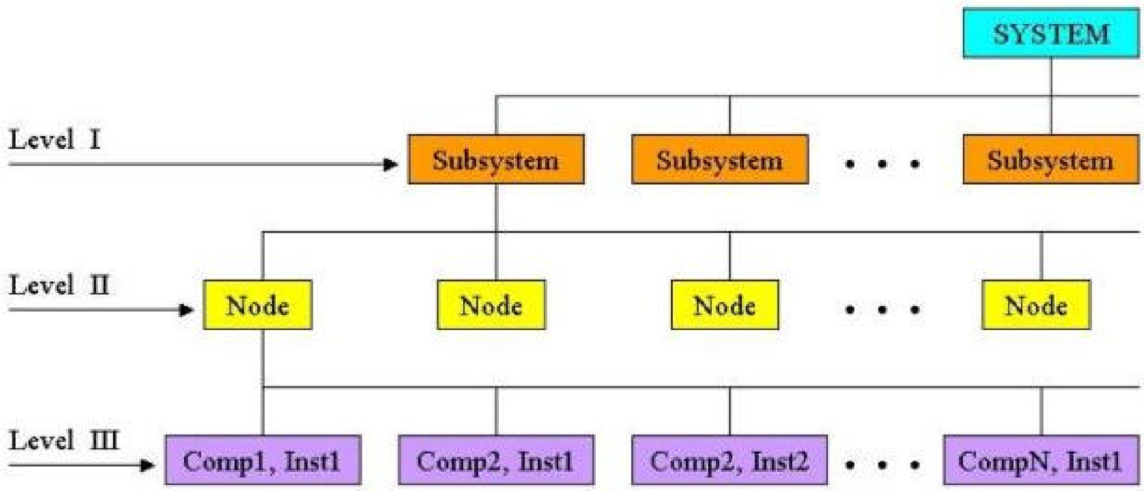

Figure 2: Compliance Levels

Verification can be independent and checked at any one of the three levels. Compliance at one level does not infer compliance at another.

The engineering directorate of AMRDEC continues developing a software utility called the JAUS Compliance Tool Suite (JCTS) [18]. Beta testing begins July 2005 with an expected release in January 2006. The JAUS community will use JCTS for system diagnostics and as an aid for compliance verification. By early to mid 2006, the OPC committee plans the third experiment to demonstrate JAUS/AS4 advanced autonomy capability combined with Robot/OCU/Payload interoperability. Capabilities include mission planning and execution, cohesive world model generation, and advanced payload interoperability. In addition, OPC-3 will evaluate new infrastructure protocols such as the proposed IPv4 and LSS[3] transport layers along with dynamic registration. OPC embarks in mid 2005 on a series of pre-experiments thereby increasing team effectiveness for the third experiment.

[3] Low Speed Serial Transport

Conclusions

The JAUS community looks forward to a mature compliance tool and rich documentation set; dynamic registration to control diverse payloads; network transport that accommodates high and low band widths. JAUS provides a common message set for designers of any platform to interoperate. When vehicles announce their capabilities and understand messages from authorized control units, we reach new levels of interoperability while lowering acquisition, maintenance, and time to deploy costs. DOD endorsement, demonstrations, collaborative experiments, commercial products and an entry into the DARPA Grand Challenge vindicate the framework's scalability and value. With beta version available and an official release of JCTS in early 2006, users have a robust diagnostic tool and means to check message compliance. The OPC will continue experiments that ensure SAE-AS4 continues addressing issues of unmanned systems interoperability using standard messages.

Acknowledgments

The author recognizes R.W. English "Woody", chairperson of AS4 and past OPC chairperson for his JAUGS-JAUS history and document review. His significant contributions in [7, 9, 11, 15, 19, 20, 21], and timeline collections are greatly appreciated.

References

[1] **Working Group on Security and Space Robotics:** The European Robotics Platform (EUROP)-April 13, 2005- http://www.cas.kth.se/europ/EUROP/europ-sec-spc-050413.pdf

[2] **The Integrated Deepwater System (IDS)-Ensuring Interoperability in the Acquisition of a Total System of Systems:** CDR Paul J. Roden, USCG & CDR Douglas J. Henke, USCG, American Society of Naval Engineers Conference February 2002.

[3] **Robotic rescue: firms developing two GPS enabled robotic vehicle prototypes designed for wounded soldier extraction and sentry/reconnaissance:** Marty Whitford,- GPS World Nov 2004

[4] **Applications suitable for unmanned and autonomous missions utilizing the Tactical Amphibious Ground Support (TAGS) platform:** Paul J. Lewis, Mitchel R. Torrie, Paul M. Omilon -2004 SPIE conference on Unmanned Ground Vehicle Technology VI in Orlando, Florida.

[5] **PSA#2076**: Commerce Business Daily Issue April 17, 1998

[6] **Fuzzy Control for Autonomous Ground Vehicles:** Jeffery S. Wit, Carl D. Crane III, David G. Armstrong II -American Nuclear Society, Ninth Topical Meeting on Robotics and remote Systems, Seattle, WA March 4-8, 2001 AFRL-ML-TY-TP-2001-0014

[7] **A Modular, Scalable, Architecture For Unmanned Vehicles:** Armstrong, D., Crane, C., Novick, D., Wit, J., English, R., Adsit, P., Shahady, D. - Proceedings of the Association for Unmanned Vehicle Systems International (AUVSI) Unmanned Systems 2000 Conference, Orlando, Florida, July 2000.

[8] **MEMORANDUM FOR ROBOTICS PROGRAM MANAGERS:** Andrus Vilu, Deputy Director Strategic & Tactical Systems – Office of the Under Secretary of Defense, Washington, DC. September 2000

[9] **Joint Architecture for Unmanned Ground Systems (JAUGS) Volume II version 2.0** Draft 26 Oct 1997

[10] **Past and Present Robotics and Related Efforts:** SPAWAR Systems Center San Diego Code 2371 Oct 2001

[11] **JAUGS Working Group minutes**- Sandia National Laboratory July 19, 2001 http://www.jauswg.org/archive/wg_minutes/MM%2017-19Jul01.pdf

[12] **Robotic Architecture Standards Framework in the Defense Domain with Illustrations Using the NIST 4D/RCS Reference Architecture -** Hui-Min Huang, James Albus, Jeffery Kotora, and Roger Liu- *2003 IEEE/RSJ International conference on Intelligent robots and Systems*, Las Vegas, NV, October 27-31 2003

[13] **JAUS Operator Control Unit (OCU) interoperability expe riment: preparation and results:** Gray, Sarah A.; Harrison, Joseph F.; Smith, Brian G. *SPIE, Volume 5422-47* April 12-16 2004.

[14] **JAUS Experiment Successful, Robotics Update** :- SPAWAR Systems Center, San Diego, CA-March 2004/vol 4 number 1

[15] **JAUS Working Group Minutes**, Madison, Alabama- February 11-13 2003

[16] **Planning and modeling extensions to the Joint Architecture for Unmanned Systems (JAUS) for application to unmanned ground vehicles:** Robert Touchton, Daniel Kent, Tom Galluzzo, Carl D. Crane III, David G. Armstrong II, Nick Flann, Jeff Wit, Phil Adsit, - *SPIE Proc. 5804-16: Unmanned Ground Vehicle Technology VII,* Orlando, FL; 29-31 March, 2005

[17] **JAUS Implementation: Robots Gather for Successful Interoperability Experiment:** M.N.Clark,", *Robotics & Automation Magazine,* IEEE March 2005, vol 12, Issue: 1

[18] **JAUS Compliance Tool Suite-ver 0.4b(Beta Test) July 13, 2005:**Wes Brown- *JAUS Working Group Conference*- Salt Lake city, UT January 18, 2005 http://www.jauswg.org/working_group/Jan05/tabs/Tab%20I%20-%20Brown.pdf

[19] **OCU & Payloads Committee Experiment #2 Report**: JAUS Documentation: released August 31 2004

[20] **JAUS Compliance Specification**, ver 1.1- March 10 2005 - http://www.jauswg.org/baseline/CS%20V%201.1%2010Mar05.doc

[21] **Specifying autonomy levels for unmanned systems: interim report:** R. W. English, Titan Corp. and US Army AMCOM and JAUS; H. M. Huang, E. Messina, National Institute of Standards and Technology; R. Wade, U.S. Army Aviation and Missile Research, Development and Engineering Ctr.; J. Albus, National Institute of Standards and Technology: *SPIE- Orlando FL April 12-16 2004 [5422-52]*

[23] **JAUS Reference Architecture**, 3 volumes August 13, 2004 http://www.jauswg.org/baseline/refarch.html

[24] **JAUS Domain Model**, ver 3.1 April 9 2004 http://www.jauswg.org/archive/domain_model/Domain%20Model%20v3.1.doc