

Development of a Space Universal MOdular Architecture (SUMO)

Bernie F. Collins

Office of the Director of National Intelligence (ODNI)

Acquisition, Technology, and Facilities (AT&F)

Washington, DC 20511

703-275-3525

Bernie.F.Collins@dni.gov

Abstract--This concept paper proposes that the space community should develop and implement a universal standard for spacecraft modularity – to improve interoperability of spacecraft components. Pursuing a global industry consensus standard for open and modular spacecraft architecture will encourage trade, remove standards-related market barriers, and in the long run increase both value provided to customers and profitability of the space industrial sector. This concept paper sets out: (1) the goals for a SUMO standard and how it will benefit the space community; (2) background on spacecraft modularity and existing related standards; (3) the proposed technical scope of the current standardization effort; and (4) an approach for creating a SUMO standard.

TABLE OF CONTENTS

1. INTRODUCTION.....	1
2. BACKGROUND.....	3
3. GOALS.....	3
4. SPACECRAFT MODULARITY.....	3
5. WHY ACT NOW?.....	10
6. EARLY EFFORTS.....	10
7. NEW WORK.....	12

1. INTRODUCTION

This concept paper proposes that there is a global need to improve interoperability of spacecraft components and that a standard for modularity of spacecraft buses is the most effective way to achieve such interoperability. Pursuing a global standard for open and modular spacecraft architecture will therefore encourage trade and remove standards related market barriers. This should both strengthen the space industry by helping them compete in a growing international space market and enable the international market to leverage the capacity of the worldwide industrial base. It will also enable the industrial base to provide more functionality to its customers (government and commercial) at lower cost

because the international marketplace can create an economy of scale that the global customer base will be able to leverage. The resources expended to modify components for a range of specific applications will instead be available to improve sensors and payloads. This universal modular architecture will focus on spacecraft manufacturing stakeholders, including customers, manufacturing primes and tier 2 and 3 payload and component manufacturers. The interoperability focus is data interfaces (including software interoperability) and electrical interfaces; to the extent practical, this effort will avoid definition of mechanical interfaces.²

The challenge at hand is that each spacecraft vendor has developed its own de facto proprietary standard for building avionics, and each vendor design represents a unique solution within the trade space of performance, size, weight, power, budget, and schedule that is optimized for a very specific purpose. Because each spacecraft vendor's architecture and de facto interface standards vary, lower tier manufacturers must integrate their components with custom interfaces and integration efforts, conforming to the higher-tier vendor's optimizations at the cost of more extensive non-recurring engineering (NRE). Standardized interfaces would reduce or eliminate the need for unique integration with each new component or application and thus lower the overall NRE costs for component reuse. Moreover, open and standardized architectures can more easily incorporate innovative and competitive products. Expecting spacecraft vendor and component manufacturers to protect their intellectual property (IP) through industry consensus standards may seem counter-intuitive. Yet – an increasing number of companies and their diverse user community are becoming less tolerant of “useless differentiation” through proprietary interfaces.³ In fact, establishing interface compatibility can free up time and capital to differentiate products through innovation, rather than differentiating them based on their niche within an optimized design space.

¹ U.S Government work not protected by U.S. copyright 978-1-4673-1813-6/13/\$31.00 ©2013 Crown

² Mechanical interfaces may be addressed to some degree; for instance, certain form factors are associated with certain electrical interfaces and may be standardized.

³<http://www.worldstandardscooperation.org/newsletters/004/newsletter04.html>

2. BACKGROUND

As the global recession has forced many governments to reduce spending, standards which encourage design and construction of affordable satellites with interoperable components can spawn innovation in both component and overall spacecraft capabilities. Optimizing interoperable components in satellite architectures will eventually reduce a prime contractor's NRE expenses (for example, those associated with data interfaces and software functionality) when new components are introduced into heritage designs. This essentially removes a major barrier to market entry for smaller component suppliers. Concurrently, it enables prime contractors to offer shorter lead times and improved capabilities to the space customer base, both internationally and nationally. It is well documented that standards can produce significant and tangible benefits across: a range of industries; three core business functions (engineering, procurement and production); and key stakeholders – both demand side (customers) and supply side (vendors).⁴ With the help of an international standard written with industry needs in mind, it is expected that government and commercial interests will shift some of their buying power to more affordable and modular spacecraft designs over time – enabling them to invest their limited resources in more sophisticated sensors and payloads. While changing heritage systems to meet the standard will come with a cost, the transition can be eased by the use of adaptors and middleware so that adoption is incremental and targeted.

This concept paper sets out:

1. goals for a SUMO standard and how it will benefit the space community;
2. background on spacecraft modularity and existing related standards;
3. expected technical scope of the current standardization effort; and
4. approach for creating a SUMO standard.

3. GOALS

Our intention is to:

1. Develop a consensus US technical position concerning the desirability of developing a standardized modular architecture for space vehicle avionics that blends the best parts of current standards and any other approaches that may be proposed during the working group process.
2. Build that consensus by consulting and involving leading technical experts from the US satellite community, including the Department of Defense (DoD), NASA and commercial providers.

3. Collaborate to develop the agreed-upon US technical position relative to the requirements for an internationally-standardized spacecraft avionics architecture. The group will initially focus on defining the problem and the desired characteristics of a preferred solution, rather than advancing any particular concrete implementation.
4. Advance the consensus US proposal to the Consultative Committee on Space Data Systems (CCSDS) and eventually, after reaching consensus within CCSDS Working Group, create the necessary international standard(s) that would then be advanced to ISO.

4. SPACECRAFT MODULARITY

This project focuses on modularity because modular architecture supports both interoperability and design for change. Few, if any, major satellites are developed entirely from scratch; they reuse heritage technology, from individual components to entire vehicles. Current industry reuse largely occurs satellite by satellite, at best standardized across an individual prime contractor. Tier 2 and 3 satellite component manufacturers will, over time, develop specialized components for each prime with whom they interact. As a result, primes and their suppliers will be tightly coupled due to the prohibitive NRE costs for multiple vehicle/component interface designs. The goal in constructing a modular architecture is to decouple specialized component designs from vehicle systems engineering, so that:

- component technology advances can be amortized over a larger number of vehicles;
- primes will have more freedom to choose the best technology available; and
- decreased NRE costs will increase space sector profitability.

As an example, consider a spacecraft data bus: MIL-STD-1553B, which is used for the primary data bus in most U.S. military spacecraft, was published in September of 1978. While terrestrial, personal, and mobile computing have seen enormous technological advances, the space industry's technology inertia has impeded technological advancement – even as more modern alternatives are being used in the similar domains of civil aviation and automotive design. This is likely to remain the case as long as components must be engineered to work with one bus specifically. The NRE associated with onboard satellite data interfaces includes more than just ensuring that data can flow across a cable; it includes ensuring that data sent across the satellite bus is sent in the format the bus requires; that it is sent using the right protocol, at the right time; that the timing allowances

⁴ World Standards Cooperation; No. 5, August 2012;
<http://www.worldstandardscooperation.org/>

are compatible with the expectations of the data's receiver; that all of the data that must flow across the bus each computational period can be accommodated; that failure notifications and responses are consistent and maintain vehicle safety; and that all of these considerations have been analyzed and tested to ensure they work in the context of the system at hand. The retest effort alone needed to incorporate a new data bus would be prohibitive.

A modular architecture, on the other hand, could define a data bus at a conceptual level and specify a generic interface between any data bus and any component connected to the bus. Such an interface could contain specifics where appropriate, e.g., in the syntactic format of the data being sent from a component over the bus; and outline the form of the interface where variance is permitted but actual values must be known, e.g., the maximum uncertainty in data transmission times. Internal characteristics of the bus, e.g., whether usage is arbitrated by a controller or is allocated a periodic time slot a priori, would be excluded to enable flexibility in designs and adoption of new technologies. The test program for any particular data bus would be required to verify all stated characteristics of the interface, so that component-level test results could be reused and overall vehicle test times shortened significantly. This approach has been adopted with success in civil aviation, under the moniker Integrated Modular Avionics (IMA).

Defining a modular architecture to meet these goals involves four major steps:

1. Determining the scope of interoperability that is desired.
2. Designing the overall structure of the architecture.
3. Specifying the interfaces within the architecture of components that may change independently.
4. Choosing the appropriate level of standardization for vehicle subsystems.

We discuss each of these steps individually in the following sections.

Determining the Scope of Desired Interoperability

A standard to support interoperability through modularity must make effective tradeoffs between interface definition and design flexibility. Standardization comes at a cost: (1) cost to change existing components to meet the standard; (2) overhead cost associated with interfaces that are not tailor-made for the vehicle at hand. Hence defining the *right* set of interfaces – and avoiding over-specification – is essential if the standard is to be adopted.

Table 1 organizes the characteristics of spacecraft buses that could potentially be standardized. For each aspect, it states whether, in this initial concept, that aspect is targeted for standardization, along with rationale for the choice. The Table will be coordinated throughout the space industry in order to achieve consensus.

TABLE 1. INTEROPERABILITY SCOPE

Category			Characteristic	In scope?	Rationale
Functional	Physical	Data bus	Connector	yes	A commonized physical output will allow bus manufacturers to choose a component based on price, performance, and availability rather than because of the cost of the NRE to incorporate the component into the bus's data network. Weight of potential middleware needed for adaptation is low.
			Cabling		
			Shielding		
		Mechanical interface	Bolt patterns	no	Too expensive/over-engineered: benefit is outweighed by the constraints that would be imposed
			Vibration isolation	no	
			Thermal isolation and/or mgmt	no	
		Power interface	Cable type	partial	May specify standards for a few different voltage levels (e.g., 28V, 56V, 112V) to facilitate standardization with some differentiation to avoid excessive inefficiency
			Voltage		
			Shielding		
	Logical/system	Data bus	Data rate	no	Requirement is that data bus can support rate needed by system. MIL-STD-1553B provides an implicit minimum. May increase as new technologies become available; need to leave flexibility to take advantage of advances.
			Jitter	yes	May have different allowable categories but algorithms & message partitioning activity need to know tolerances
			Latency	yes	May have different allowable categories but algorithms & message partitioning activity need to know tolerances
			Quality of service	yes	May have different allowable categories but algorithms & message partitioning activity need to know tolerances
			Protocol	no	Can specify interfaces that support multiple protocols

Category			Characteristic	In scope?	Rationale
		Component (type-specific)	Bus interface	yes	Need components to be able to plug into whatever bus protocol is chosen
			Interface to application software	varies	Team will need to define what components may change, which drives which interfaces need to be standardized
			Sensor Precision/Accuracy	yes	Specific precision/accuracy not defined, but characterization will be required such that algorithms can statically determine whether a given component in a class is within tolerance
		SW infrastructure/ services	OS abstraction (incl. CPU time & space partitioning)	yes	Need to allow components to work with different operating systems
			Network access layer	yes	Need to allow components to work with different network types
			Other services (TBD)	varies	SOIS contains some candidates; these are points of coupling of the applications and are needed if sets of subsystems are to be reused/changed/rapidly assembled
		Intra-satellite data	Command formats	no	Needed, but can be separated from the current effort
			Telemetry formats	no	Needed, but can be separated from the current effort
		Applications	GN&C, C&DH, EPS, etc.	no	Would result in over-specification
		Payloads	General	partial	Payloads will conform to specification of a general component
Nonfunctional	Physical	General	Parts, Materials and Process requirements	yes	Have to standardize on space-qualified components and processes. Leads to universal component certification.
			Component RMA requirements	no	Need system reliability to be within spec but component reliability can vary. Minimum component reliabilities may be necessary in some cases, e.g., switching hardware
		Environmental test requirements	Vibration	partial	Need components certified across “regions” of launch and mission environments to avoid unworkably narrow selection of components for any particular system. May lead to unacceptable weight/power requirements for some components; candidate initial component set to be standardized across regions is torque rods, sun sensors, star trackers, GPS receivers, solar cells, reaction wheels/CMGs, processor boards, and transponders.
			Acoustic		
			Radiation		
			EMI/EMC		
			Thermal/vac		
		Fault interface	Allowable unmasked faults	no	Too difficult to define a priori at this time; push complexity into logical interface, as is currently done
	Logical/system	General	SW & system dependability/RMA requirements	one or the other	Need a way to determine whether a piece of software has been developed and tested to the appropriate level of assurance
			Development processes & process evidence		
			Cyber Security	yes	Vulnerabilities may exist, and may be public, in common elements; could preclude adoption of standard if not addressed
		Fault interface	Component failure notification format	yes	Need to standardize so that fault handling is independent of implementation
			Component failure notification content	no	Push the complexity into higher layers to allow for flexibility
			Software failure responses	no	Push the complexity into higher layers to allow for flexibility
			Format for database of permissible value ranges	yes	Part of electronic data sheet for sensor; allows swapping of sensors without retooling software interface
		Performance	Worst case execution times	no	Do not need to specify this for particular subsystems; can rely on static analysis of integrated system and OS guarantees on time partitioning
			Algorithm precision and accuracy	partial	Requirements algorithms place on component precision/accuracy need to be standardized, but precision/accuracy characterization of software outputs varies too widely to be in scope at this time.
			Quality of service	no	Not typically applicable to domain

Designing the Architecture Structure

Our conceptual architecture integrates both hardware connectivity and logical interfaces. As illustrated in Figure 1, the functionality within the architecture is defined in 4 major layers: individual components; component

interconnection (data and electrical); software services that link the components with the flight software applications; and the flight software applications themselves. Component classes assigned to higher layers use the interfaces of lower layers to decouple their behavior from the specifics of other hardware and software components of the avionics.

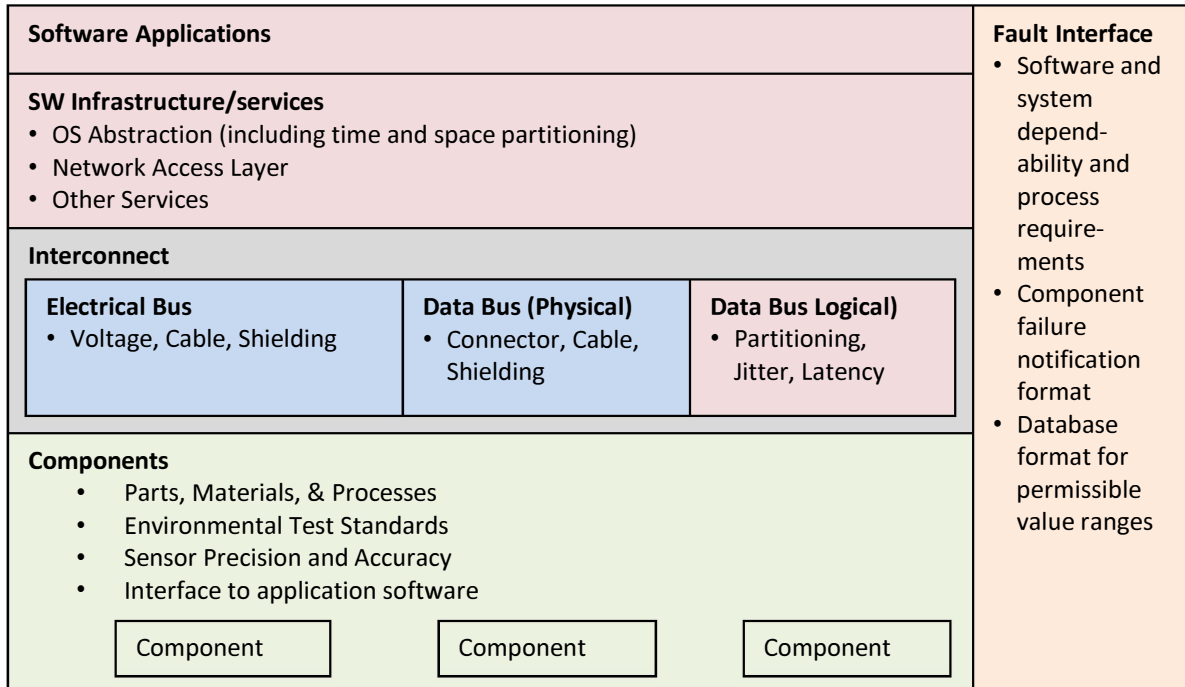


FIGURE 1. ARCHITECTURE LAYERS

Table 2 shows the scope (from Table 1) of each layer and references existing standards that the committee can use as a springboard for its activities.

TABLE 2. SCOPE AND REFERENCES FOR ARCHITECTURE LAYERS

Component/ component class	Scope	SPA reference	SAVOIR reference	Other References
Hardware	Data bus connector, cabling, shielding Electrical bus cable type, voltage, shielding Sensor precision, accuracy	xTEDS SPA Ontology Standard SPA 28V Power Service Standard SPA Physical Interface Standard SPA Logical Interface Standard	SOIS green book sections 5, 2	
Avionics Infrastructure	Data bus interface, jitter, latency	SPA SpaceWire standard SPA System Timing Standard SPA Networking Standard	SOIS green book, sections 4,2	

Component/ component class	Scope	SPA reference	SAVOIR reference	Other References
Computing Services	Operating system abstraction Network access layer Other services, incl. data handling, fault detection, device interface Component failure notification format		SOIS green book, sections 3,2,7	
Applications and Payloads	Format for DB of permissible value ranges Algorithm precision and accuracy Flight software interface		SOIS green book, section 7 (minimal)	
Test, Reliability and Mission Assurance	PMP requirements Vibration, acoustic, radiation, EMI/EMC, thermal/vac test requirements SW & system dependability & process requirements	SPA Test Bypass Extension		MIL-STD- 1540 on environ- mental testing
Cybersecurity	To be refined by experts		CCSDS green books “The Application of CCSDS Protocols to Secure Systems” and “Security Threats against Space Missions” SOIS green book, section 6	

Specifying Architectural Interfaces

Each architecture module (component, layer, service, software application, etc.) that falls within the committee scope (as defined in Table 1) will be worked individually by experts on the committee. The goal is to characterize each module well enough that its interactions with other modules can be defined and standardized as an interface.

Each pair of modules that interact directly within the architecture will be considered for interface standardization. The pieces to be defined during the interface specification process for a sample pair of modules (Module A and Module B) are illustrated in Figure 2. To create A’s interface, the subject matter experts on module A would:

- Define common and variable features that A should be able to support
- State common requirements that A will place on its interfaces (i.e., requirements that other modules need to meet)
- Define the interfaces that A can provide to others (i.e., the requirements that A is able to meet)

- Meet with the subject matter experts on module B to iterate between what B can provide and what A needs, until the “right” interface (in terms of efficiency and flexibility) is reached.

The team working B would concurrently do the same, developing B’s capabilities and needs so that the iteration step would produce a result for the pair. The same team could potentially work both A and B, if the team had the necessary breadth of expertise.

With these clearly defined interfaces, modules can then be developed independent of each other, and may include the variability planned for them, without impacting one another. In other words, A’s outputs and expected inputs should be independent of B’s internal design, and vice versa. The specific definitions of the interfaces between modules in the SUMO architecture will constitute the SUMO standard.⁵

⁵ It is acknowledged that full implementation of a CCSDS standard will require two successful prototype demonstrations.

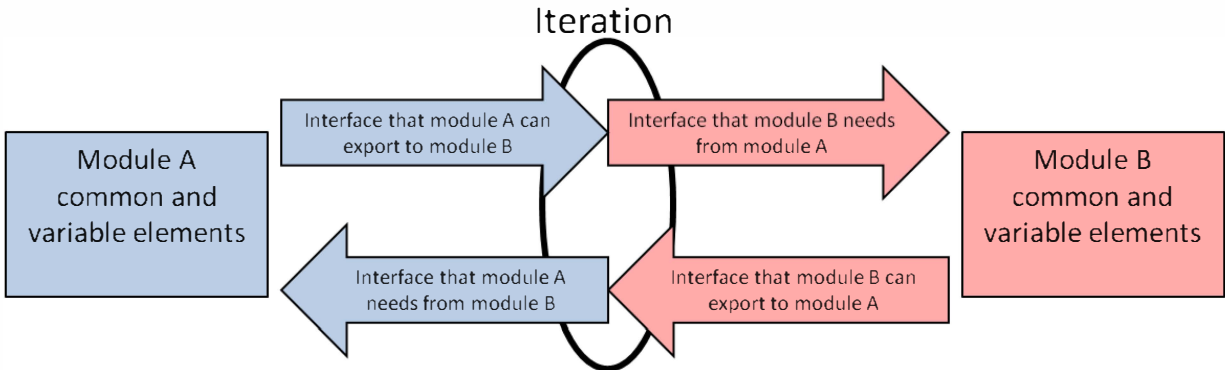


FIGURE 2. DEFINING MODULE INTERFACES

5. WHY ACT NOW?

A unified and internationally standardized open architecture for space vehicle components will support U.S. space component and payload manufacturers, as well as the global space community more generally, by removing market barriers and providing a “level playing field” on which internationally-interoperable products can be based. Moreover:

- There is a strong international desire to eliminate proprietary and regional standards that introduce market barriers which undermine trade.
- There is significant economic and strategic motivation among satellite component manufacturers to reduce NRE expenses. By establishing an internationally standardized modular architecture for space vehicles, satellite component manufacturers would reduce duplicative design and testing. As a result, certification procedures could allow for more time and capital to focus on design and performance improvements.
- A goal of the *US National Space Policy – 2010* is to “promote a robust domestic commercial space industry” and “foster fair and open global trade and commerce through the promotion of suitable standards and regulations that have been developed with input from U.S. industry.”

6. EARLY EFFORTS

Air Force Research Laboratory (AFRL) Efforts – focus on extremely short satellite integration timelines.

AFRL has developed a set of standards for a Space Plug and Play Architecture (SPA) in response to the need for extremely short design, fabrication, integration and test schedules. SPA’s primary goal is to reduce satellite

development phases from months and years to only a few days. SUMO targets the problem of reducing satellite development cost and schedule without requiring the very short timelines supported by SPA. SUMO’s less strict development timeline requirements may result in a different architecture; SUMO’s resulting standard set may not call out for self configuration and organization of components – the key plug and play element within SPA. The CCSDS Spacecraft Onboard Interface Services (SOIS) working group has made substantial recent progress leveraging the AFRL SPA, scoping standard data interfaces, and designing specific Electronic Data Sheets (EDS) for a range of components. The EDSs will automate the inclusion of the components in the satellite data-bus and accelerate integration and, more importantly, enhance rapid reconfigurability of the bus. The SUMO team has refined the phases of the SUMO Implementation Plan to include the coordinated development & adoption of EDSs for components -- based upon manufacturers’ Interface Control Docs (ICDs) we can leverage AFRL’s SPA & CCSDS efforts for quick and early results.

More recently (2011), AFRL introduced Monarch (Modular Open Network Architecture) as a refinement of the earlier SPA concept. This version emphasizes a simplified implementation of the suite of SPA technologies, more optimized for large, high-assurance spacecraft with less emphasis on rapid development.

European Space Agency (ESA) Efforts – Space Avionics Open Interface aRchitecture (SAVOIR) – an initiative to increase standardization and reuse within the space avionics community.

SAVOIR is an initiative to federate the space avionics community, bringing customers, primes and suppliers together in order to improve the way that the European Space community builds avionics subsystems. The primary outputs of SAVOIR are:

- A reference avionics architecture for spacecraft platform hardware and software

- A set of avionics external and internal interface specifications
- The definition of high-level architectural building blocks
- The detailed functional specification of selected architectural building blocks
- Implementations of selected building blocks at high TRL levels
- Process definition and assessment.

SUMO will be addressing a similar challenge to SAVOIR, and will build on the existing architectural effort, but may result in a more simplified architecture that addresses less of the detailed software functionality than SAVOIR currently does.

Commercial Aerospace Efforts – Integrated Modular Avionics (IMA) – a commercial avionics architecture that supports component modularity given shared computing resources.

IMA is an architectural trend in commercial avionics systems that reduces the design cost inherent in aviation's more traditional federated architectures. The basic principle of IMA is that several functions (even of different criticality levels) can share common computing resources⁶. In federated architectures, each function executes exclusively on its dedicated computer system, incurring duplicative space, weight, power and maintenance costs for the dedicated hardware. Co-locating multiple functions on the same set of resources, however, introduces concerns about the potential for fault propagation across functions. IMA supports cross-functional fault containment (mitigating the hazards arising from collocation) through time and space partitioning of the computing platform as described in the ARINC 653 Standard. Resource sharing introduces different roles and responsibilities for the software executing on the platform. RTCA DO-297, Integrated Modular Avionics Development Guidance and Certification Considerations, describes the roles and responsibilities for IMA programs based on best industry practice.

NASA Efforts – Core Flight System (CFS) – a reusable software framework providing basic functionality but allowing for mission tailoring.

The NASA Core Flight System (CFS) project has two major goals: (1) to reduce spacecraft development and integration times, similar to SPA, but in weeks rather than days; and (2) to reduce spacecraft development cost by allowing for reuse of much of the flight software functionality across multiple hardware architectures. To achieve hardware architecture and implementation independence, the CFS focuses on standardizing interfaces at several system layers. The CFS is the software

framework in NASA's approach to spacecraft plug and play; it enables CFS components to be added and removed without rebuilding the software or even shutting down a running system.

NASA Efforts – Common Avionics Architecture (SpaceAGE bus) – independent board-level functional building blocks that can be mixed and matched to define an avionics box.

Defined to be the hardware analog to the software CFS, the SpaceAGE bus provides flexibility in defining avionics box functions. The architecture relies on a card frame mechanical approach to intra-box interfaces that eliminates the backplane and mechanical box chassis to reduce NRE. Boards are externally harnessed together via a common hardware building block called the HUB from which all cards are interconnected via a non-blocking crossbar switch capable of bridging between different NODE protocols. A serial communication physical layer is defined as well as the minimal set of generic signals necessary for space avionics intra-box interfaces. The Common Avionics Architecture is agonistic to protocols and box-to-box interfaces.

Department of Energy (DOE) (Sandia National Laboratory) – Joint Architecture Standard (JAS) – a network-based payload architecture that is scalable, reliable, and reusable.

Sandia and Los Alamos National Laboratories are currently developing a new architecture that will underlie future Nuclear Detonation Detection System (NDS) payload designs, regardless of host platform. JAS is a modular, node-based architecture that uses standard hardware designs, high-speed serial data interfaces, and reusable hardware and software IP. It uses a layered architectural design to isolate the software from hardware change impacts, and uses a service-based design to support flexible reuse of software to meet new payload needs. JAS offers COTS-based development and test environment for rapid system demonstration.

7. NEW WORK

Evaluate existing and emerging standards that address spacecraft component interoperability for data and electrical interfaces.

Develop a collaborative approach to gain a consensus US technical position that is likely to evolve into an international standard. It is important that the global marketplace be considered; hence we must eventually work with the larger international space community to forge international agreement on the best approach given the various standard-based architectures that are currently being developed. A first and vital step in securing such an

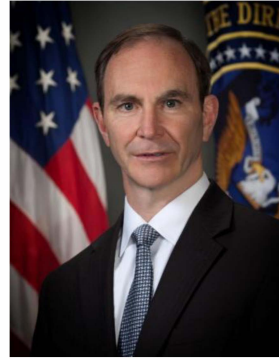
⁶ Airlines Electronic Engineering Committee, ARINC specification 653: Avionics application software standard interface. Aeronautical Radio, Inc., January 1997.

international agreement is to craft a technical consensus position across the US space community.

REFERENCES

- [1] American Institute of Aeronautics and Astronautics; "Space Plug-and-Play Architecture Standards Development Guidebook"; Draft August 2011.
- [2] CCSDS – Consultative Committee on Space Data Systems (<http://public.ccsds.org/about/default.aspx>).
- [3] CCSDS – "Spacecraft Onboard Interface Services" – Draft Informational Report; CCSDS 850.0-G-2; Draft Green Book; April 2012.
- [4] WSC - The World Standards Cooperation; (<http://www.worldstandardscooperation.org/>)
- [5] Francis, N., Collier, P., and Lyke, J. "Optical Networking for Aerospace Systems Provisioned Through Plug and Play Avionics", Proceedings of the 2010 AIAA Infotech Conference, 20-22 Apr 2010 Atlanta, GA. (paper)
- [6] Lanza, D., Vick, R. and Lyke, J., "The Space Plug- and-Play Avionics Common Data Dictionary - - Constructing the Language of SPA ", Proceedings of the 2010 AIAA Infotech Conference, 20-22 Apr 2010 Atlanta, GA.
- [7] Lyke, James A.; "U.S. Air Force's Plug-and-Play Satellites"; IEEE Spectrum; August 2012.
- [8] Taylor, Chris; "Spring CCSDS meetings Opening Plenary" (power point); ESA Estec; April 2012.
- [9] Vera, Alonzo, Sibley, M, Ardalan, S., Avery, K., and Lyke, J. "Appliqué Sensor Interface Module Based on 90nm Rad- Hard Structured Application- Specific Integrated Circuit", Proceedings of the 2010 AIAA Infotech Conference, 20-22 Apr 2010 Atlanta, GA.

AUTHOR BIOGRAPHY



Bernie Collins currently serves as the Senior Advisor for the Director of Science and Technology in the Acquisition, Technology and Facilities Directorate for the Office of the Director of National Intelligence. In this capacity he leads activities to evaluate the space industrial base and to improve satellite acquisitions. He is developing and coordinating an innovative approach to acquiring satellites through industry consensus standards. Mr. Collins has extensive experience in the acquisition, development and operation of satellites. He was manager of two successful satellite programs and mission director for satellite operations. He also led overhead architecture studies and supported several independent program assessments and red teams. Mr. Collins is the recipient of the National Intelligence Medal of Achievement and the National Reconnaissance Office's Medal of Distinguished Performance. He received his Bachelors of Science in Electrical Engineering from the University of New Hampshire and his Masters of Science in Electrical Engineering, as a distinguished graduate, from the Air Force Institute of Technology. He is certified as a Program Manager Level 3.