

Deeper Dive into Interoperability and Its Implications for LunaNet Communications and Navigation Services

James Schier^a, Coralí Roura^a, Phillip Paulsen^b, Karl Vaden^b, Jennifer Rock^b, Charles Sheehe^b, Angela Peura^c, Marc Seibert^c, Erica Weir^d

^a National Aeronautics and Space Administration, 300 E St., SW, Washington, D.C. 20546 USA, james.schier-1@nasa.gov, corali.roura-1@nasa.gov

^b National Aeronautics and Space Administration, 21000 Brookpark Road, Cleveland, OH 44135 USA, phillip.e.paulsen@nasa.gov, karl.r.vaden@nasa.gov, jennifer.l.rock@nasa.gov, charles.j.sheehe@nasa.gov

^c Agile Decision Sciences, LLC, 350 Voyager Way, Suite 100B, Huntsville, AL 35806 USA, apeura@asrcfederal.com, mseibert@asrcfederal.com

^d Teltrium Solutions, LLC, 600 Jefferson Drive Rockville, MD 20852 USA, eweir@teltrium.com

Abstract

The Artemis program being developed by United States' (US) National Aeronautics and Space Administration (NASA) is developing the capabilities to return humans to the Moon and establish an initial base camp and associated infrastructure with extensive contributions from international and commercial partners. In planning for cislunar exploration and science missions, space agencies are collaborating to enable communications, networking, and Positioning, Navigation, and Timing (PNT) systems—called LunaNet—to exchange information and provide services to cislunar spacecraft and space systems, thus helping each other to achieve their shared goals. To achieve commonality and lower cost for mutual benefit, the strategy of interoperability is being adopted to help fit all the pieces together and function smoothly. Facilitating interoperability should benefit lunar missions by providing the ability to operate in a collaborative environment similar to the terrestrial Internet. Interoperability allows them to share information, navigate safely despite increasing radio frequency congestion, and follow common processes and procedures for effective joint operations. Unlike prior government-dominated efforts, this ecosystem is expected to include and benefit for-profit (commercial) businesses, non-profit organizations, and academic institutions as active stakeholders. Ultimately, the goal is to enable a cislunar ecosystem of service providers and users to contribute to and/or utilize infrastructure and capabilities to achieve mission objectives that span the full range of human endeavors while supporting a variety of business models. This approach enables a Systems of Systems (SoS), such as a Network-of-Networks, to be sustainable in the context of the LunaNet ecosystem as systems evolve over time in technologies, standards, components, and user applications. This paper reports on the results of an effort to help frame the development of the international LunaNet architecture by providing a canonical definition of interoperability broad enough to meet these needs, examining architectural and operational implications of the definition, and exploring interoperability strategies and tactics to deploy and evolve the services proposed for cislunar exploration and science missions.

Table of Contents

1. Introduction	2
2. Interoperability Definition	3
2.1 Architectural Implications of the Definition.....	4
2.2 Operational Implications of the Definition	5
3. Interoperability Strategies and Tactics	8
3.1 Measuring and Assessing Interoperability	9

3.2 Verifying and Validating Interoperability	16
3.3. Sustaining Interoperability	19
4. Conclusion	23
5. References	24
6. Acronyms	25

1. Introduction

The United States' (US) National Aeronautics and Space Administration's (NASA) Artemis program is developing capabilities needed to return humans to the Moon for the first time since the Apollo missions, this time to stay. The initial ability to land crew and associated systems on the Moon to explore, set up a base camp, conduct scientific research, and test technologies for mining and construction as a new lunar economy begins to take shape. These lunar capabilities will eventually be used to test our ability to send humans to Mars as humanity prepares to expand farther into the solar system.

This new era differs from the Apollo era in key respects. First, NASA is collaborating with a growing number of international partner space agencies. This has proven mutually beneficial for the International Space Station partners at the cost of increasing complexity and interdependencies, with cost and schedule impacts from the associated risks. Second, NASA is being used by the US government as an agent of expanding the US economy—and the global economy with its partners—by investing in developing commercial services and capabilities rather than the government being owner and operator. While the terrestrial commercial satellite communications market developed with little to no interoperability, the desire is for the lunar market to be interoperable from the outset to unlock future growth potential for all participants. This effort to create a lunar market and economy offers new motivation for sustained investment but requires long-term commitment from partnering governments. Third, the concepts for optimizing performance and minimizing cost lead to an integrated System of Systems (SoS) that requires partner systems to interoperate seamlessly and unprecedented cooperation in design and operations.

The overall lunar architecture includes communications, networking, and Positioning, Navigation, and Timing (PNT) infrastructure that supports all of the lunar missions. Community efforts over the past decade have led to defining an integrated approach called LunaNet. LunaNet is envisioned as a set of cooperating networks providing interoperable communications and navigation services for users on and around the Moon based on a framework of mutually agreed-upon frequency bands, standards, protocols, and interface requirements. This framework enables interoperability within all space systems, benefiting lunar space operators and missions regardless of their sponsorship (e.g., governments, academia, commercial). The purpose of this paper is to report the results of a study of the deeper implications of interoperability. This will help frame the development of the international LunaNet architecture by defining interoperability in sufficient detail to investigate the architectural and operational implications of the definition and explore interoperability strategies and tactics to deploy and evolve the services proposed for LunaNet. Figure 1 shows key aspects of the LunaNet architecture including lunar relay and navigation satellites, network terminals on the surface for local area networking, network trunk links to Earth ground stations or Earth-orbiting relays, and network control on Earth. Missions that use LunaNet services (Users) may also have direct links to Earth, may depend partially or completely on LunaNet Service Providers (LNSP), and may also host LNSP packages making them providers as well as users. Several national space agencies are currently collaborating to define LunaNet with a vision of an open architecture that will grow to include many other government and commercial providers and users like the terrestrial Internet.

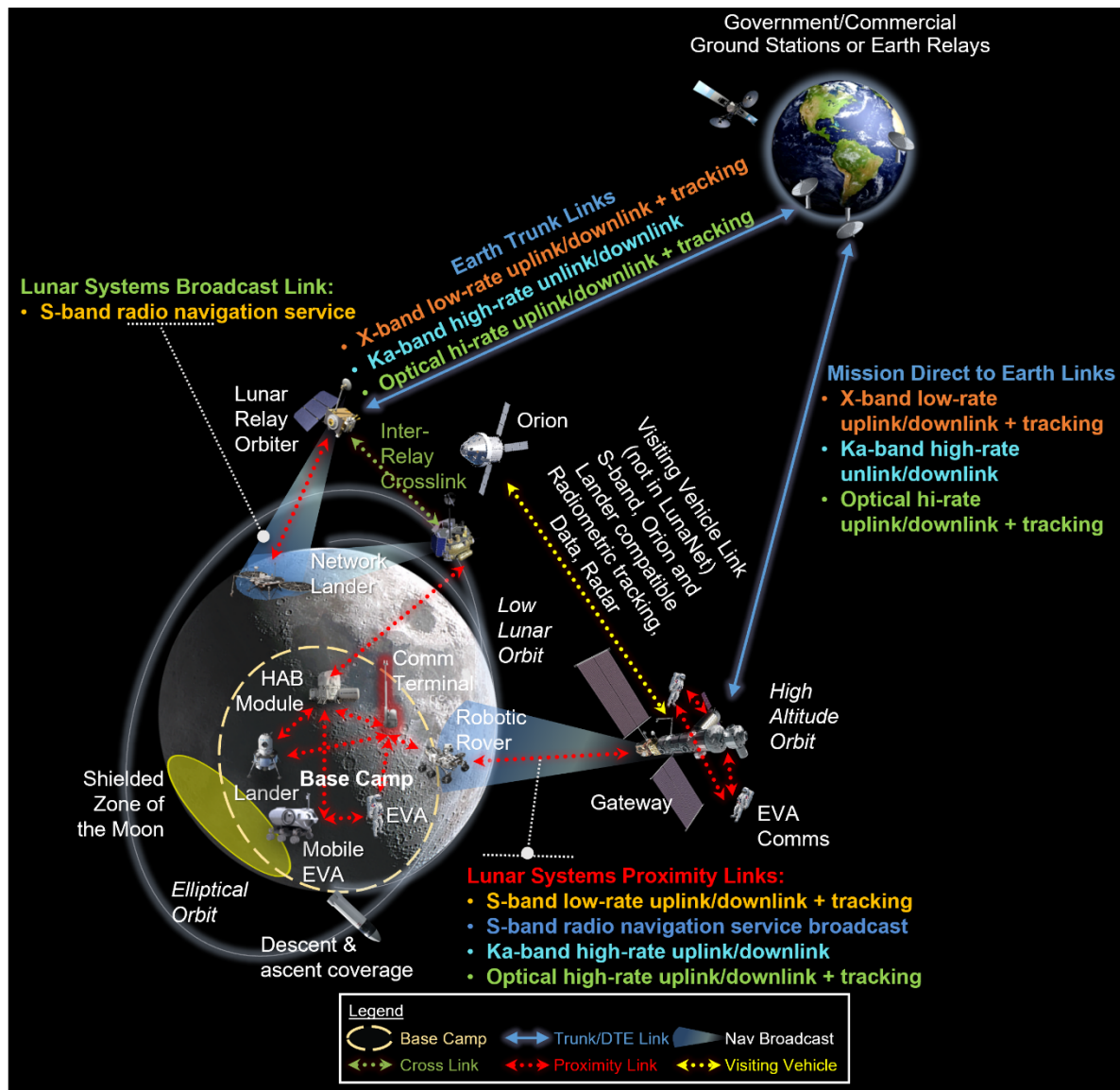


Figure 1. Overview of LunaNet Architecture with Lunar Relay and Navigation Satellites, Surface Terminals, and Network and Mission Links to Earth

2. Interoperability Definition

Interoperability is defined here as the *ability to exchange and understand information* between entities that agree on: a) the *syntax* used to format (encode) the information; b) the *semantic meaning* (content) that represents the information or references the contextual meaning; c) the *means of exchanging* the information; d) the *system context* in which the information exchange occurs; and e) the means of *initiating and sustaining* interoperability across asynchronous changes in entities and their operation.

Entities can include humans, human organizations, communication systems, networks, and other systems that exchange information. Due to communications latency, information exchange also includes information storage and retrieval, which imposes its own solutions to information representation required to preserve semantic content. The context in which the information exchange occurs represents broad social, societal, and organizational information such as reference systems, regulations, policies, legislation, and security constraints. Interoperability in this definition represents ideal or perfect exchange and understanding. Imperfections in data transmission, information representation, and conversion at interfaces introduce errors and noise that contribute to reduction or loss of

intended interoperation. Thus, interoperability is not a simple binary yes/no state; it allows a range of degrees of interoperability that can be designed, assessed, and evolved.

2.1 Architectural Implications of the Definition

This definition enables use of formal systems engineering methods. The International Council on Systems Engineering (INCOSE) defines Model-Based Systems Engineering (MBSE) as “the formalized application of modeling to support system requirements, design, analysis, verification and validation activities beginning in the conceptual design phase and continuing throughout development and later lifecycle phases.” [1] An example of MBSE is illustrated in Figure 2, as evidence that interoperability can be modeled with tools using Object Management Group’s Systems Modeling Language™ (OMG SysML®). [2]

To apply structure and rigor to the methodology described, two defining elements of interoperability—language and architecture—are modeled. To develop the language of interoperability, lexicon and taxonomy are modeled in pursuit of a full ontology, which captures the lexicon within domains and ontological concepts, such as taxonomy, classification, and interrelationships.

These models were developed using multiple commercial tools:

- *Cameo Systems Modeler (CSM)*: A “cross-platform collaborative MBSE environment, which provides smart, robust, and intuitive tools to define, track, and visualize all aspects of systems in the most standard-compliant SysML models and diagrams.” [3]
- *Protégé*: An open-source ontology editor includes logic reasoners to consolidate and flatten taxonomies to define system *semantic content* by providing an *ontological syntax*. [4]
- *Neo4j*: A non-relational graph database capable of modeling elements (nodes), their directional relationships (edges), and the characteristics or parameters associated with each node and edge.

These tools enable a model-based, data-driven approach to making informed decisions on multi-domain SoS. In addition, these tools provide accessible, interactive visualizations that provide insights not easily apparent in dictionaries and other textual documents.

Figure 2 shows a SysML logical architecture (yellow) for Interoperability and an example showing how it could be instantiated (orange). This logical architecture can be instantiated to build current and proposed models of interoperability. The abstract model extends the *syntax* to define terms for hierarchical Capabilities and Assessments, where Capabilities are composed of multiple Sub-Capabilities enabled by Services and Services are enabled by Vendors, who incorporate Technology and Standards that satisfy the requirements. Although not shown, each element can contain parameters, operations, and interfaces while security can be modeled as a combination of internally derived requirements and externally imposed laws, regulations, and policies. The instantiated model defines specific Capabilities decomposed into multiple Sub-Capabilities traceable to Requirements, but with instances in which the system’s *semantic content* and *system context* are defined, and the *means to exchange information* and to *initiate and sustain interoperability* are modeled. Capabilities may need to be refined into both Functions and time-phased Increments. Decomposing Sub-Capabilities into time-phased Increments refines development cycle time for a better integration of activities, while decomposing functionality improves focus on technical characteristics to achieve operational objectives. The blue boxes shown in the figure denote how the five elements of the interoperability definition can be modeled. Subsequent sections show how this enables measurement of interoperable properties, verification of requirements that specify the extent of interoperability at distinct evolutionary points during system operation, and the ability to sustain a specified degree of interoperability over the lifetime of the systems(s) involved.

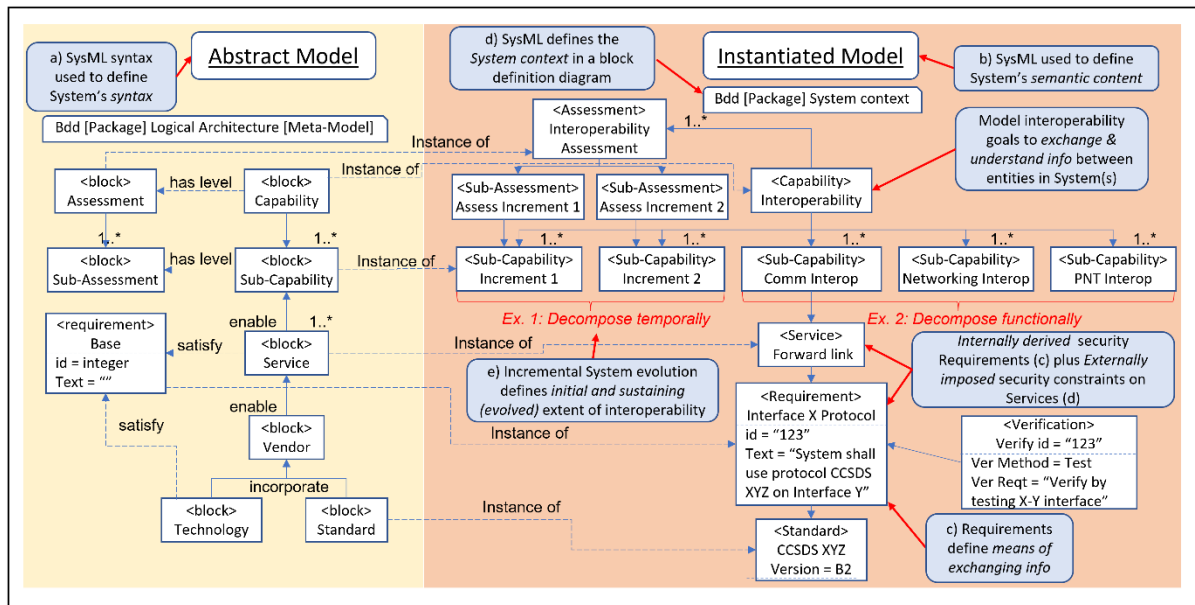


Figure 2. Example SysML Implementation of Interoperability

In this paper, systems planned to be deployed to the lunar surface are used to provide examples of the interoperability concepts discussed.

2.2 Operational Implications of the Definition

This section explains operational implications of the interoperability definition for exchanging and understanding information (definition element c) and sustaining interoperability across asynchronous changes (definition element e), as well as essential characteristics that must occur for interoperability to become an operational capability, including the syntax, semantics, and system context (definition elements a, b, and d).

Exchange and understand information: The Institute of Electrical and Electronics Engineers (IEEE) defines interoperability as “the ability of two or more systems or components to *exchange* information and to *use* the information that has been exchanged” [5]. This has two parts. First, information must be exchanged—transmitted by one entity and received by another. The exchange may be human to human, human to machine (system), or machine to machine. Second, the information must be understood by the receiver as the same information that was transmitted. This definition covers necessary but not sufficient criteria, which is why the three additional criteria are proposed in this paper.

The complexity of the overarching concept can be seen in a space command and control example (Figure 3), where the *system context* in which the information exchange occurs is shown highlighted in orange. Three dissimilar systems constitute part of two interoperating networks. System 1 is a ground-based MOC network and System 2 is a space system network. System 3 acts as the interface between networks and could be a ground or space-based platform. System 1 and System 3 contain the communication subsystems that enable them to be members of Network 1, while System 2 and System 3 contain the communication subsystems that enable them to be members of Network 2. System 3 acts as a network gateway exchanging data between Network 1 and Network 2. Operator 1 resides in System 1; Operator 2 resides in System 2; and System 3 is automated to the extent that its operator is not relevant.

- *Human-Human interoperation* occurs (end-to-end) when information sent by one human is received and understood by another human. This occurs if they: a) speak the same language, e.g., English, with the language-specific *syntax* that defines sentence structure; b) employ words to describe commands using defined *semantics*, i.e., the definitions and concepts of the words; and c) use a *method of information exchange*, e.g., transmitted by

mouth and received by ear. Understanding is successful if the information received is equal to the information sent and both share the same understanding of that information—the *context*. The flight director, Operator 1, gives the direction “Update the landing location to latitude X and longitude Y”. The astronaut, Operator 2, receives the information formatted as a command in English with semantics directing Operator 2 to change the programmed destination of System 2. This presumes that the operators share the *context* of a previously defined coordinate system that defines a geodetic reference frame (e.g., defined by an international standard that describes latitude and longitude for a specific celestial body).

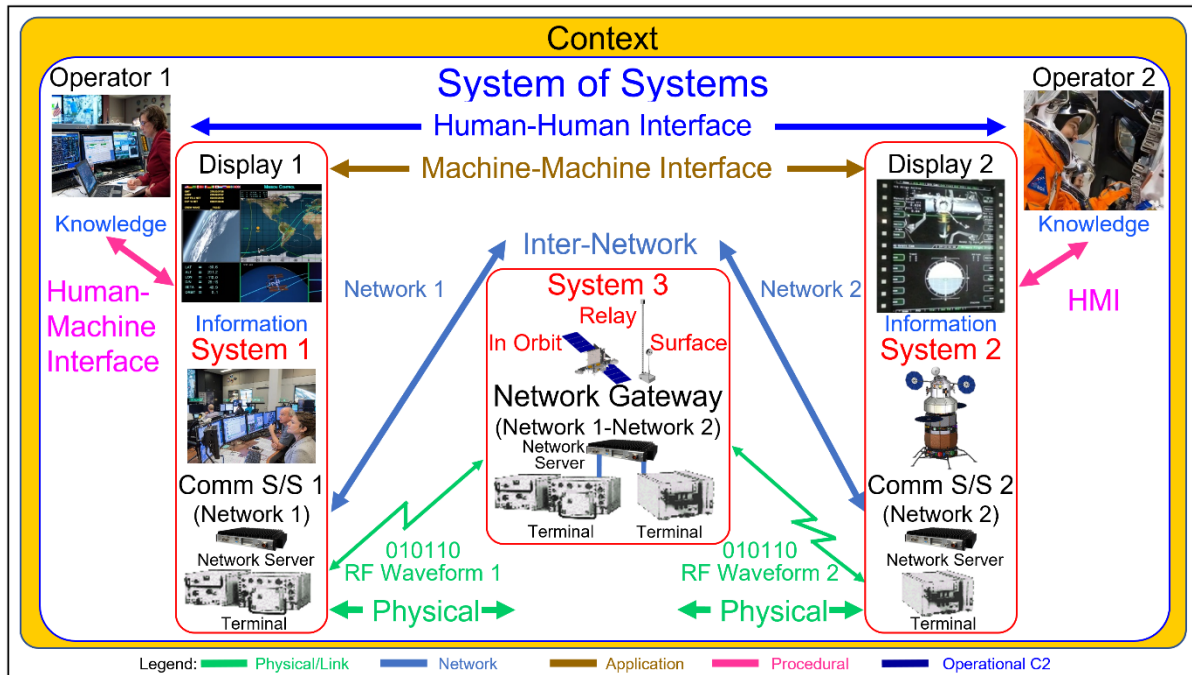


Figure 3. System-of-Systems (SoS) with Interoperability in a Space Command and Control Context

- Standards that define the lunar equivalent to the International Terrestrial Reference System (ITRS) are being developed. The ITRS describes procedures for creating reference frames suitable for use with measurements on or near the Earth's surface by defining a geocentric system of cartesian coordinates using the International System of Units (SI) for measurement. An International Terrestrial Reference Frame (ITRF) is a realization of the ITRS. In the near future, an International Lunar Reference System will be realized by an International Lunar Reference Frame (ILRF) that defines the corresponding coordinate system for lunar longitude and latitude. This will be incorporated into the system model as a requirement to comply with the ITRF when referencing Earth orbits and the ILRF when referencing lunar orbits. These geodetic frames will then be designed into the ground and space systems, incorporated into ground and flight crew training, and used for planning missions. All personnel involved in the SoS will use the common context.
- **Human-Machine Interoperation (HMI)** occurs when information sent by the human is received and “understood” by the machine, i.e., the information is converted from a human format to a machine-readable format. If Operator 1 speaks a command, the machine must have an audio receiver, e.g., a microphone, that converts the operator’s modulated voice into a signal that encodes the same information as the spoken command. System 1 transmits the signal via System 3 to System 2, which translates the command into an audio signal to a speaker that reconstructs Operator 1’s words. Understanding at the HMI-level occurs if Operator 2 hears and understands the command to “Update the landing location”. A second HMI interface occurs in the exchange of visual information. Operator 1 uses eyes to observe System 1’s monitor showing tracking information, which is

transmitted to System 2's display, enabling Operator 2's eyes to see the same tracked objects and achieve the same situational awareness as Operator 1.

- *System-System interoperation* occurs if the systems: a) employ the same information *syntax*, e.g., a message data format; b) understand the *message contents*, e.g., encode and decode the same information in the message; and c) combine that with *means of transmitting and receiving* the message, e.g., via communication subsystems. System 1 may interface directly with System 2 via a Machine-Machine Interface (MMI) if the systems use a common radio waveform and establish connectivity.
- *Network-Network interoperation* occurs when information sent by one network can be received and understood by another network. Figure 3 depicts two networks that do not share a common set of networking protocols. System 1 belongs to Network 1, while System 2 belongs to Network 2. Between them is System 3, which is a gateway—member of both Network 1 and Network 2—capable of translating from the Network 1's protocol stack to Network 2's stack.
- *Physical-Physical interoperation* occurs when information sent by one terminal can be received and understood by another terminal. In the example, Terminals 1 and 2 use different waveforms for communicating and are, therefore, incapable of direct interoperation requiring translation by System 3's gateway. Waveforms are characterized by frequencies, modulation, coding, error correction, and other parameters as well as by the protocol used to control the energy radiated at the physical level to ensure correct transmission of information.

Sustaining interoperability across asynchronous changes: Without the means of sustaining interoperability, interoperability may occur at an instant of time, but it decays due to the asynchronous changes made by Standards Development Organizations (SDO), equipment vendors, service providers (e.g., ground station and relay spacecraft operators), and users (e.g., mission spacecraft) over the asynchronous evolution of systems from initial to final configurations. Sustainable interoperability across modifications to hardware and software requires three criteria:

- *Backward compatibility* allows interoperability between newer and older versions. It exists if information created in an entity with a newer version (with improved *syntax*, *semantics*, or *exchange* capability) can be sent to and understood by an entity operating on an older version lacking those improvements. The older entity can extract the proper subset of information described by the older version of its equipment and can accurately act on the subset of information that it understands.
- *Forward compatibility* allows interoperability between the current version and future versions. This is usually accomplished by defining a roadmap of planned improvements with sufficient specificity that the future upgrade can be implemented without costly retrofitting of the existing equipment. An example in hardware is to define a future capability and scar the current design to allow the future capability to be installed later. A software example is to create interface placeholders or stubs that define the planned syntax and semantics such that the stubs can be replaced with operating software when it becomes available.
- *Version transparency* exists if the receiving system, built using an older version, can then retransmit the information it received to other systems without loss of information present only in the newer version. The older entity must be capable of preserving and transmitting the information that it received rather than the information that it is capable of understanding.

Backward and forward compatibility are separate decisions. One does not imply or require the other, however, backward compatibility has been shown to be easier to sustain if previous versions have been designed to be forward compatible. [6] Backward and forward compatibility and version transparency can be achieved if the information being transmitted across a set of interacting entities of heterogeneous versions can be represented as *self-describing*. Prior technologies designed hardware and software that either had specific

capabilities built in, e.g., to a specific model or version number, or allowed reconfiguration via reprogramming. Both approaches have limited ability to support backwards compatibility or version transparency.

Self-describing information can be implemented in three ways:

- *Internal Representation*: Exchange information that supports a syntax that describes the capabilities embodied in its version, as well as the semantic content to define the specific behavior and performance provided by those capabilities.
- *External Reference to a Shared Schema*: Exchange semantic content using a shared syntax and schema prior to its use in operations enabling the entities to rely on stored information.
- *Hybrid*: Use a hybrid combination of the methods of Internal Representation and External Reference to a Shared Schema. The eXtensible Markup Language (XML), for example, uses a combination of referencing a specific version and defining an internal extension that allows users to create customized extensions to base capabilities. Software-Defined Radios (SDR) now use similar abilities so that a component of any version can interpret the schema for that version or earlier versions while passing on the complete information containing definition of capabilities of later versions.

With or without backward and forward compatibility and version transparency, procedures can be used by cooperating organizations to preserve inter-system interoperability while evolving. An example of this is considered in section 3.3. In all cases, the desired degree of interoperability should be explicitly incorporated into the system model with descriptive information to define the planned context and requirements to specify the desired performance so that it is verifiable and sharable with the community.

3. Interoperability Strategies and Tactics

Interoperability viewed strategically focuses on sustaining operations between entities over the long-term as the System of Systems (SoS) configuration changes dynamically, frequently to introduce new or improved capabilities. Tactically, interoperability focuses on short-term operations between entities (e.g., day-to-day steps and actions) where the SoS configuration is relatively static but flexibility for making operational changes must be supported.

Strategies and tactics for measuring and assessing, verifying and validating, and sustaining interoperability were explored to assist in the deployment and evolution of LunaNet. These strategies and tactics are aimed at supporting full automation, even during modifications, to maintain 24/7 operations.

A complicating factor affecting our ability to implement interoperability that spans multiple systems and participants is self-interest. NASA and the rest of the US government are particularly interested in reducing the cost of developing, testing, flying, and operating systems in space. Currently, there is a huge push to use commercial services and practices to the greatest extent possible and to partner with commercial industry to develop future systems. As a result, it is expected that we will be able to sustain the rapid pace of commercial technological advancement and enjoy the cost savings associated with industrial competition.

Given the small number of participants fielding systems in space and the unique nature of the space environment, it has traditionally been in industry's interest for commercial SATCOM providers to offer "boutique" (one-off) services and proprietary interfaces, operations, and services. By doing so, they hope to capture and maintain a stable portion of the total market. In addition, there are some aspects of their systems that must remain unique. For example, radio frequency spectrum is purchased by SATCOM service providers and is unique to each provider.

From a strategic standpoint, establishing open forum(s) spanning all participants will be an important step in establishing the dialog needed to determine a consensus approach among all stakeholders. This approach has been very successful in other areas, such as the traditional Internet and in cell phone infrastructure development. Recently, several commercial SATCOM providers voluntarily established consortium-based activities, aimed at incorporating common 3rd Generation Partnership Project (3GPP)-compliant services in space. These services can form the basis for common backchannel processes to manage and provide services across providers. The government could also make targeted investments in system development and integration to spur industry to consider the use of alternative approaches in the future.

3.1 Measuring and Assessing Interoperability

The NASA team created a tool in collaboration with MITRE corporation to analyze, measure, and evaluate the interoperability of almost any architecture and the nodes within it. This tool and scoring methodology can be applied to support strategic and tactical interoperability decisions. Objective interoperability scores (0-100%) are produced for both the entire architecture and individual nodes within it, after having performed an interoperability analysis. Results are useful for many purposes, such as identifying optimum upgrades to directly improving the interoperability of a system or SoS.

The goals of interoperability assessment are to provide a “baseline” score for an architecture and the interoperability of individual nodes, expose problem areas of a new and/or future architecture or an active operational system by identifying low-scoring nodes (interoperability bottlenecks), and allow decision makers to identify alternatives or system upgrades to improve performance of either problem nodes or the entire architecture. Additionally, the results allow systems engineers to periodically reassess the architecture and the nodes within it. For example, as the architecture and nodes in the design phase or in operation mature, reassessments can be performed to quantitatively show improvements that have been made or highlight areas that were missed throughout the lifecycle of the architecture.

To perform an interoperability assessment, it must first be decided how to define the architecture being evaluated and which systems or SoS within the architecture are considered “nodes”. Nodes in the architecture are the systems or SoS with the ability to communicate with other nodes across the architecture. The interoperability score of the node and the entire architecture will largely depend on how well the nodes perform this function. Data and reference material are needed as inputs to help understand how the architecture should be decomposed into its underlying systems and low-level components or nodes. The analyst can choose to define an architecture very broadly, as a very large scope like the Artemis SoS, or just a set of systems like the lunar surface architecture. Alternatively, an analyst can define an architecture to be a single system composed of multiple subsystems or components. Consideration must be given to how well the nodes, the systems within them, and the architecture “interoperate,” that is, how they interact, control the parts or equipment of another node, or obtain information from other nodes in the architecture.

Figure 4 shows the process flow to evaluate interoperability elements at the node level and Figure 5 shows the hierarchy of elements that the analyst must define to perform the interoperability assessment. The scope can be defined for the entire architecture, for example for selected systems and/or specific nodes of the entire architecture, as well as for selected systems and/or specific nodes.

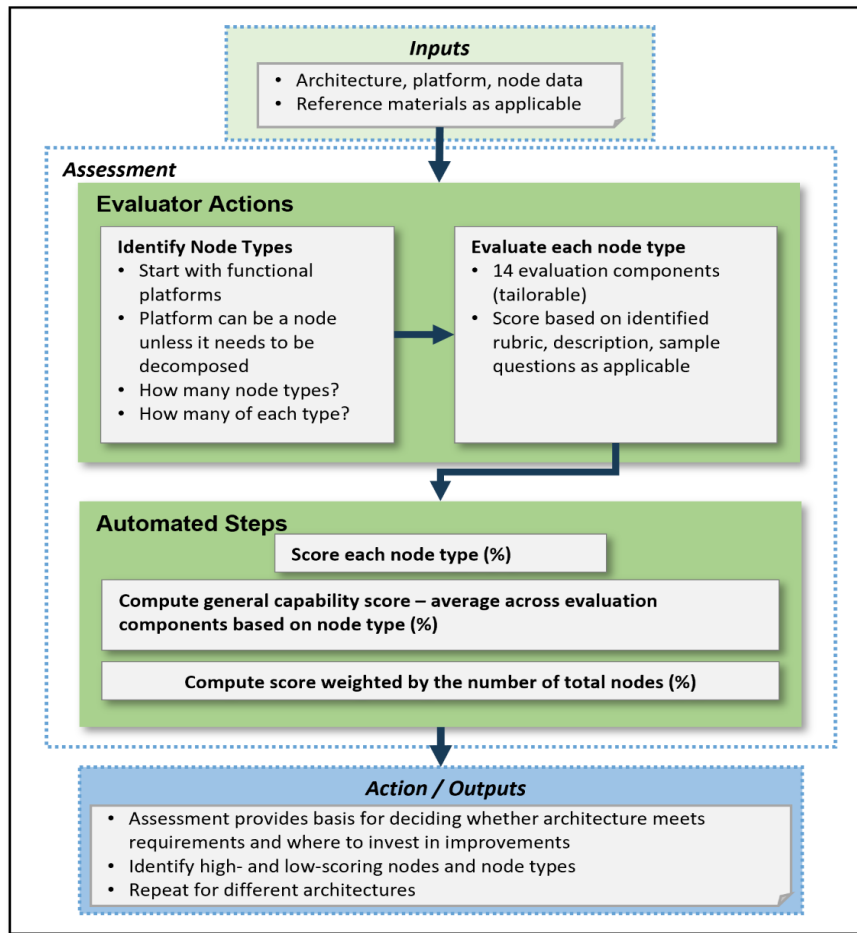


Figure 4. Assessment Process for Node-level Interoperability Elements

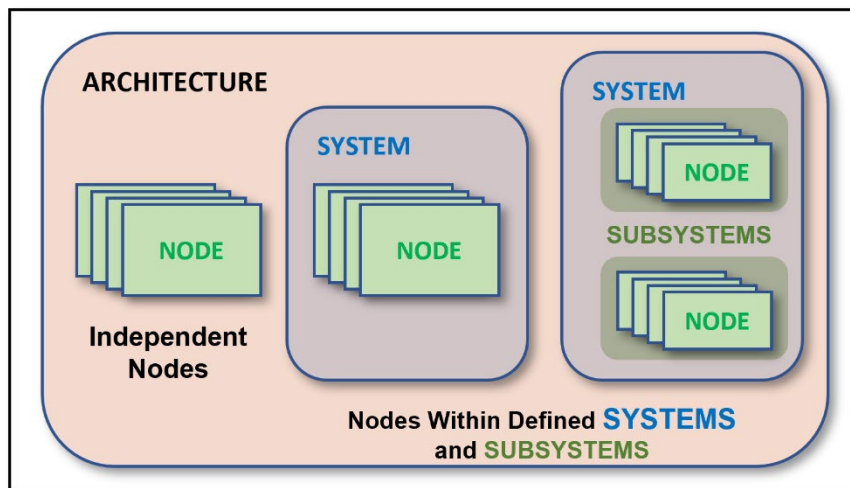


Figure 5. Analyst-defined Assessment Elements

Figure 6 depicts an example of how an architecture can be decomposed from a large-scale SoS, such as an Artemis lunar mission, to an evaluation of interoperability on a single lunar element, such as the Exploration Extravehicular Mobility Unit (xEMU or space suit). This allows decision makers to identify systems that are not meeting interoperability objectives and increase their capabilities so that they better contribute to mission success. The evaluation can begin once it is decided within the architecture at which level(s) to define the

nodes. There is great flexibility in how an architecture can be decomposed, depending on the desired strategic and tactical perspectives and approaches.

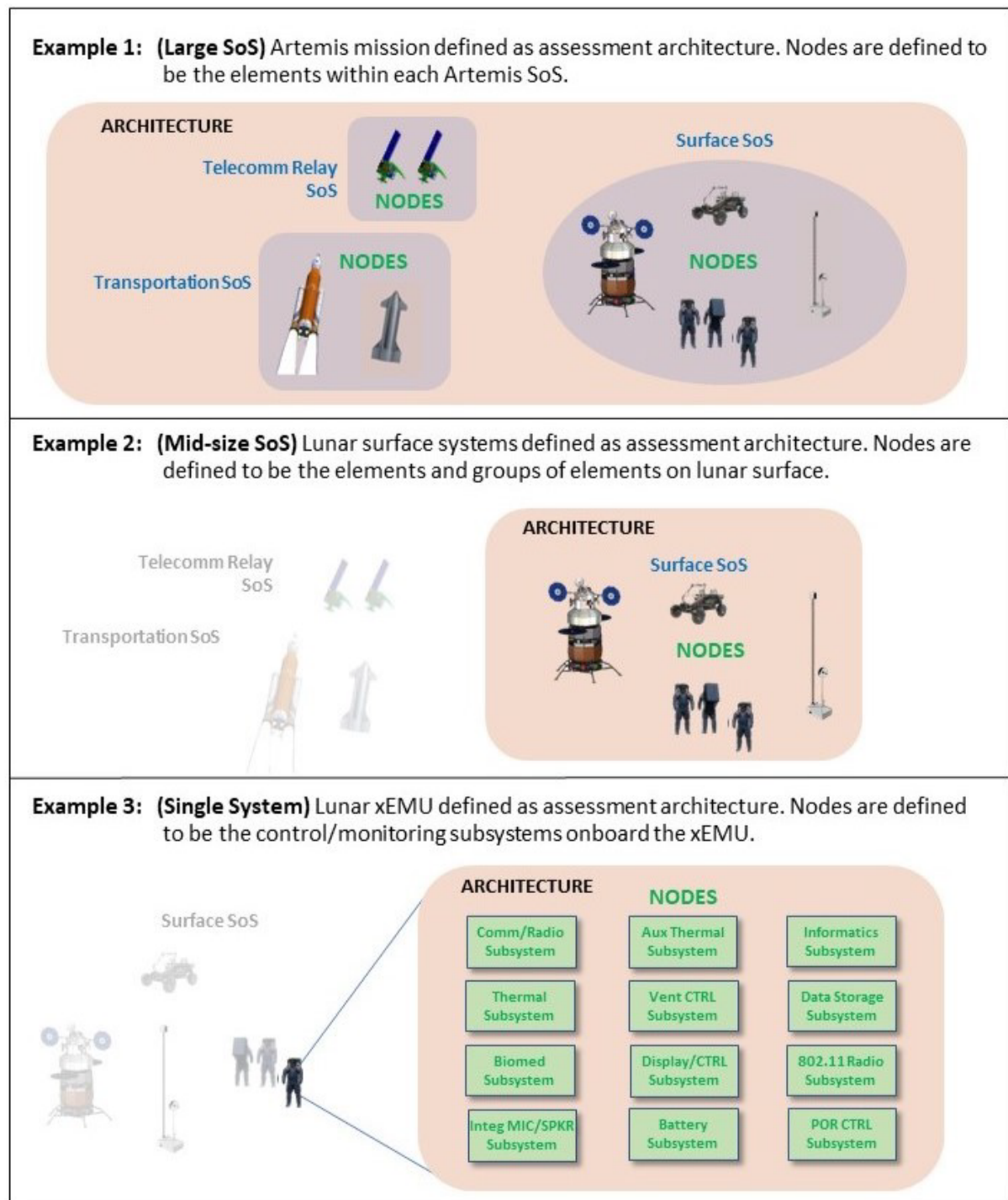


Figure 6. Three examples of Architecture and Node identification

Assessing node-level interoperability: The scoring criteria used in the tool to make the baseline assessment for node-level interoperability are as follows:

- **General communication or networking capability:** The ability to network or communicate is the most fundamental requirement to enable telecommunications interoperability. In this assessment we refer to all forms of communication as “networking”. At a minimum, every interoperable node must be able to network or communicate in a standard manner with interfacing nodes within the architecture. As examples, a network can be as simple as a

series of “signal fires” with one message type spanning mountain tops, an 1800’s telegraph network with a defined set of “dots” and “dashes” representing characters to interoperate, or as complex as a large constellation of commercial satellites, clusters of elements on the surface of Moon or Mars, or an interoperable deep space optical network of elements depending on each other to interact with Earth. If a system in the architecture cannot network or communicate in a standard way to other nodes in the architecture, its interoperability score is zero and the analyst can end the assessment here. If a node can network or communicate in some way, then an assessment can proceed to the next step.

- **Node networking/communication protocol capability:** Scoring the specific networking capabilities of each node is the next assessment step. Basic networking abilities and protocols today allow minimal interoperability between nodes and are scored at an average level in the assessment. Nodes with advanced network protocols provide greater interoperability and receive higher assessment scores. These nodes help other nodes in the architecture increasing the capacity and availability of all nodes reduce latency for all nodes by providing more path options for information to move, thus overall strengthening the architecture’s security, resilience, and redundancy and the nodes within it.

Assessing the robustness of networking protocols is an important part of the assessment. Assessing robustness is based on how a system or node communicates across a network through protocols that it currently uses and that may be used in future upgrades. Using well-established Internet, industry, and government standards and protocols across all networking layers ensures the highest level of interoperability and, therefore, the highest scores for a node. Using non-standard, military, proprietary, or agency-unique protocols achieves a minimum level and score, as these characteristics severely limit interoperability. Emerging standards with backward compatibility provide broader interoperability capability than government, military, or proprietary protocols, but fall short of the reliability, security, and resiliency of a well-established and broadly employed protocol standard, so they are scored accordingly.

Protocol standards for a unique type of node must be compatible with other nodes in the architecture or interface via a network gateway that converts between protocols. For example, all nodes in a network with information routing functionality must use the same minimum level of routing protocols. These nodes may also employ protocols beyond this capability, but a minimum set of protocols must be present across all nodes of a similar type in the network to achieve basic interoperability. Protocols and standards are often dramatically improved over time, and security issues in the standards are continually eliminated as they are found. It is very important that an interoperable node and/or system can keep up with the evolving form of communication between other nodes in the architecture, as they adapt and advance with standards and technologies.

A fully interoperable node in the network and architecture can be upgraded so its internal library of protocols and standards keeps pace with changes to other nodes in the architecture. A node that cannot be upgraded becomes isolated and obsolete. Extreme isolation due to inflexible protocols not only deprecates the node entirely, but also means that its contribution to the architecture’s performance degrades over time.

- **Node resilience:** Nodes in a network must be able to survive, operate, and adapt to dynamic networks and potentially hostile nodes or networks within or outside their own architecture. To obtain a good score in the resilience component of interoperability, nodes must be evaluated for the following areas:
 - **Capability to communicate frequently:** This capability is applied either through the use of its own systems or via relay through other, more capable, nodes in the architecture. This is critical to both receiving important system patches for fast-moving security risks, as well as performing interoperability-supporting tasks such as relaying time-sensitive information for another node in the architecture that has no other communication path options.

- Diversity of networking paths to a node: This capability is best illustrated with an example: a node with two radios operating on independent frequencies using different protocols, receives a higher score due to its dissimilar redundancy.
- Patching capability of a node: Interoperable nodes are essentially “good citizens” in the interoperable architecture by staying up to date. This requires that the nodes’ hardware and software be designed to receive new and updated networking protocols, and to quickly receive and implement patches for security issues affecting the node.
- Ability to prioritize information: This capability includes information handling and Quality of Service (QoS) capability. Interoperable nodes that can transmit their own information or “relay for others” the most time-sensitive information first and the lower priority information later, receive a high score for this category in the interoperability assessment. Any network node that receives high priority information from another node and takes its time transmitting the information behind lower priority information is detrimental to the resiliency and performance of an architecture.

Figure 7 includes the evaluation groups such as general networking, node protocols, and node resilience, as well as the evaluation component, description, scoring concept notes, and scoring framework. Analysts can make modifications to the scoring rubric itself, such as modifying the scoring algorithms to add weighting factors to individual scoring elements. When employing weighting factors (e.g., adding a new column with a weighting factor and including it in the scoring functions), an analyst can choose to increase the importance of individual scoring criteria (rows) such as “Real-time Security Patching/Resilience” compared to other scoring criteria. Similarly, an analyst can also add an entirely new scoring criterion (row element), such as “Affordability to Upgrade”, and also include this metric into the node and architecture scoring algorithms. Analysts should consider this tool as a good starting point for measuring any system, but also consider customizing the scoring rubric to match unique performance, organizational, or management needs.

An example of the node assessment results of a hypothetical Artemis-III mission and systems, including the Human Landing System (HLS), which will transport astronauts to the lunar surface, and the Lunar Surface xEMU elements, are shown in Figure 8. The HLS lander and Extravehicular Activity (EVA)/xEMU elements evaluated include the number of nodes, element scores, and notes from the analyst.

Assessing architecture-level interoperability: Once all nodes in an architecture are assessed and scored, groups of the same node type are combined and scored with other node groups using a mathematical algorithm to produce a normalized composite Interoperability score (0-100). The mathematical algorithm consists of scoring the overall architecture by averaging the scores of all groups of nodes and averaging those results across the averages of all groups of node types. This allows addition of an objective weighting factor that considers how many nodes of each type are present. Figure 9 shows the architecture preliminary scoring results for the hypothetical Artemis III mission overall.

INDIVIDUAL NODE/SYSTEM SCORING					
Evaluation Groups	Evaluation Component	Evaluation Component ID	Description	Scoring Concept Notes	Scoring Framework
General Networking	Can the node network?	GN-1	Does the node have connectivity or is it isolated from network layer? Ex: Is the soldier/plane/tank/astronaut/rover on the network? Can the command or sensor information be put on the network at all?	Three level score: 0/1/2. Accounts for limited networking functionality such as a device that may not have wireless connectivity but can "dock" for a wired connection – which implies time-dependency (see temporal availability). 0 = no connectivity; 1 = partial or dependent; 2 = connected.	0 / 1 / 2
	If the node can network, what kind of protocol is used (context independent)?	NP-1	Protocol standards levels vary from well-established industry/internet/government standard, to emerging to less ideal proprietary or unique standards that could be DoD, civil, or commercial in nature. What level of standardization is present at network layer?	Four level score to reflect benefits of well-established standards. If industry standard = 3, if emerging = 2, if proprietary = 1, if none = 0. Standards in use must be captured in notes in order to score subsequent category.	0 / 1 / 2 / 3
Node Protocols	Are protocol standards commensurate with the context of the node's functions (context dependent)?	NP-2	What level of standardization is present at transport layer?		0 / 1 / 2 / 3
		NP-3	The node may use a well-established standard, but if the other nodes in the architecture use a different standard, the individual node is not interoperable. Network layer?	Is the node compatible with the protocol in use across most of the other nodes? Threshold for majority is >50%. Yes = 1; no = 0.	0 / 1
	Protocol Flexibility/ Modification Capability?	NP-4	Transport layer?		0 / 1
		NP-5	Is the device modifiable or capable of adopting new protocols during campaign/operation; both in the sense of the design of the system being capable/supportable of protocol modification and the associated resources? A change is any action taken to change the capabilities/functions present in a protocol layer. At the network layer?	Can updates technically be pushed to the node while in operations -- meaning the node can accept changes, regardless methodology? Yes = 1, No = 0.	0 / 1
		NP-6	At transport layer?		0 / 1
		NP-7	What is the mean time to modify protocols (whether the node is modifiable during operations or not)?	During operations +2; out of ops +1; not realistic/not feasible = 0.	0 / 1 / 2
	Store-and-forward enabled at network layer?	NP-8	Does the node have store and forward capability at the network layer -- through standard or proprietary protocol?	If equipped with a store and forward protocol of any time = 1; unit discards packets when there is no ability to transmit information = 0	0 / 1
	Does the node have adequate store and forward capacity?	NP-9	For the given operational environment, has the node been equipped or is it capable of storing sufficient information as physical link connectivity cycles through AOS/LOS periods?	If capable of tolerating temporal outages: has memory capacity to store all critical data until next link is available = 2; some store and forward capability but limited capacity, e.g., may not be able to store all data = 1; none = 0.	0 / 1 / 2
Node Resilience	Node/System Temporal Availability	NR-1	What proportion of time is the device/node/system available?	Always "on" (>95% availability) = 2; "on" but moderate limitation (75-95% availability) = 1; intermittent or unreliable (<75% availability) = 0.	0 / 1 / 2
	Path Diversity/ Resilience	NR-2	For a given node, is there more than one end-to-end path? (Must assume a notional destination – "farthest node".)	Two+ paths capable = 1; only one path possible = 0.	0 / 1
	Real-time Security Patching/Resilience	NR-3	Ability to real-time upload/resolve security issues: deploy patches, upgrades, etc. Focus is on hardware/software capability vs. operational preference or "comfort" level.	If yes = 1; if no = 0.	0 / 1
	Quality of Service	NR-4	Is the node capable of prioritizing the information in its outgoing traffic?	If yes = 1; if no = 0.	0 / 1

Figure 7. Scoring Rubric for a Node in the Assessment Tool

ARTEMIS 3 NODES – NOTIONAL EXAMPLE RESULTS						
Eval Component ID	HLS Lander		Lunar surface xEMU			
	(Assuming general lander capabilities)		UHF Subsystem		WIFI and Informatics Subsystem	
	No. of nodes of this type →	1	No. of nodes of this type →	2	No. of nodes of this type →	2
	Score	Notes	Score	Notes	Score	Notes
GN-1	2	N/A (notional score)	1	Uses Crit-1 radio: 2-way voice, downlink, TLM	2	Uses Crit-3 radio: bi-directional IEEE 802.11 networked
NP-1	3	Assuming CCSDS and IPv4	1	N/A (notional score)	3	N/A (notional score)
NP-2	3	N/A (notional score)	1	N/A (notional score)	3	N/A (notional score)
NP-3	1	N/A (notional score)	0	N/A (notional score)	1	N/A (notional score)
NP-4	1	N/A (notional score)	0	N/A (notional score)	1	N/A (notional score)
NP-5	1	N/A (notional score)	0	Radios must be modified on the ground	1	N/A (notional score)
NP-6	1	Assumed to be capable by crew or ground	0	N/A (notional score)	1	N/A (notional score)
NP-7	2	N/A (notional score)	1	N/A (notional score)	2	N/A (notional score)
NP-8	0	N/A (notional score)	0	N/A (notional score)	0	N/A (notional score)
NP-9	0	N/A (notional score)	0	N/A (notional score)	0	No known store and forward protocols in use
NR-1	2	N/A (notional score)	2	Crew expected to operate within UHF radio range	1	N/A (notional score)
NR-2	1	N/A (notional score)	1	N/A (notional score)	1	N/A (notional score)
NR-3	1	N/A (notional score)	0	N/A (notional score)	1	N/A (notional score)
NR-4	1	N/A (notional score)	1	N/A (notional score)	1	N/A (notional score)
	86%	NODE INTEROPERABILITY SCORE	36%	NODE INTEROPERABILITY SCORE	82%	NODE INTEROPERABILITY SCORE

Figure 8. Sample HLS Lander and Lunar Surface xEMU Scoring Results in the Assessment Tool

It is possible to modify the assessment tool to apply organizationally or situationally unique weighting factors to node and architecture scoring rubrics. For example, the scoring rubric could be modified to include relative cost or mission criticality levels for each node or for key nodes. Applying weighting factors to the assessment can be used to steer the assessment towards program priorities.

ARCHITECTURE / SUMMARY SCORING				
Evaluation Groups	Evaluation Component	Description	Scoring Description	Score Result
General Networking	Architecture Connectivity Resiliency	Simplified representation of the amount of "connectivity" - (e.g. are nodes connected)	Sum of node type scores over total possible	92%
	Architecture Protocol Standardization at Layer 3	The degree to which nodes use common/accepted standards at Layer 3	Sum of node type scores over total possible	89%
Architecture Protocols	Architecture Protocol Standardization at Layer 4	The degree to which nodes use common/accepted standards at Layer 4	Sum of node type scores over total possible	89%
	Architecture Protocol Uniformity at Layer 3	The degree to which nodes are compatible at Layer 3	Sum of node type scores over total possible	83%
	Architecture Protocol Uniformity at Layer 4	The degree to which nodes are compatible at Layer 4	Sum of node type scores over total possible	83%
	Architecture Protocol Flexibility at Layer 3	The degree to which nodes can change protocols at Layer 3	Sum of node type scores over total possible	67%
	Architecture Protocol Flexibility at Layer 4	The degree to which nodes can change protocols at Layer 4	Sum of node type scores over total possible	67%
	Architecture Protocol Flexibility-Timeliness	Timeliness feasibility of updating protocols (both layers)	Sum of node type scores over total possible	83%
	Architecture Delay Tolerance - Capacity/Storage	The degree to which nodes are store and forward enabled	Sum of node type scores over total possible	17%
	Architecture Temporal Resilience	The degree to which nodes on the network are available	Sum of node type scores over total possible	92%
Architecture Resilience	Architecture Diversity Resilience	The degree to which nodes on the network have path diversity	Sum of node type scores over total possible	100%
	Architecture Security Response Resilience	The degree to which nodes on the network have ability to update / be responsive to security patching	Sum of node type scores over total possible	67%
	Architecture QoS	Simplified representation of the quality of networking across the nodes/systems in the architecture	Sum of node type scores over total possible	100%
NOTE: This is a sample result and includes scoring from all nodes that were evaluated as part of the architecture, not just for the HLS lander and the xEMU suit.			General Capability Score - Average Across Evaluation Components Based on Node Type Performance>>	79%
			Architecture Score Weighted By the Number of Total Nodes by Type >>	76%

Figure 9. Architecture Scoring Results for the Artemis III Mission Evaluation Sample

3.2 Verifying and Validating Interoperability

For complex systems, a key part of the engineering process is to develop, maintain, and document system-level verification and validation (V&V) spanning the entire project (concept to flight) to:

- Ensure that the conceptual design adequately considers all the interoperability implications related to the fully integrated system as a key element of a SoS.
- Determine how closely a design element adheres to the envisioned system Concept of Operations (ConOps).
- Ensure that the system(s) meets all design requirements.
- Uncover system workmanship defects to prevent failure.
- Ensure there are margins built into the design to accommodate future updates and modifications.
- Ensure that the final, fully integrated, end-to-end SoS, operates correctly and produces the expected data products for each anticipated set of operational conditions.

For stand-alone flight projects, the V&V process is exercised iteratively for every aspect of the system design process, beginning during conceptual development, moving through each critical milestone, and ending shortly before flight. Complex systems with incrementally deployed capabilities may require more than one round of V&V. The V&V processes are similar in nature, but have fundamentally different objectives:

- Verification shows compliance with requirements—that the system can meet each “shall” statement using various methods, e.g., test, analysis, inspection, and/or demonstration.

- Validation shows that the system accomplishes the intended purpose in the intended environment, i.e., that it meets the expectations of the customer and other stakeholders. Validation uses different procedures than verification but may use the same methods.

A verified and validated baseline and proper configuration controls are needed to avoid later modifications that could be costly or cause significant performance issues. V&V methods may differ between incremental phases as capabilities and designs advance.

For SoS flight projects, V&V becomes more complex, particularly when mission-unique, non-interoperable services are included. These capabilities allow commercial vendors to offer unique “niche” services that often require exclusive interfaces, data flows, and operations concepts, putting users of these services at risk of becoming dependent on single vendors and incurring higher costs. From an interoperability standpoint, this may be acceptable, providing the unique services do not introduce issues with the interoperable core services. Whether operating in a single system or SoS environment, organizations can develop their V&V plans more productively from a set of requirements shared by stakeholders and formed using good systems engineering practices. Figure 2 is intended to give a glimpse into how interoperability can be formally verified and validated in this manner using SysML.

The strategy of requiring open standards for interoperability may be at odds with current initiatives to use “Public-Private Partnerships” (PPP) for acquisition. Commercial providers routinely develop and fly new systems that do not rely on extensive government-drafted requirements or test procedures and treat proprietary capabilities as competitive discriminators. If the government does not include specific interoperability requirements in its procurements and/or chooses to limit its procurements to only high-level requirements that lack the specificity needed to achieve interoperability, upfront cost estimates may appear to be lower, but can have unanticipated consequences in terms of cost and schedule during system development, testing, and operations as unexpected interoperability issues arise. These cost and schedule implications have traditionally driven the government to take a more active role in the areas of interoperability requirements, procurements for applicable systems, and ConOps development from the beginning. The era of PPPs may make it necessary for the government to integrate more aggressive strategies and tactics into the V&V process for the commercialization of interoperable services.

Changes in Service Commercialization/Comingled Services: To reduce costs while improving overall end-to-end system responsiveness and latency, there has been a recent push to comingle space system services, operations, and data among service and data providers, (e.g., NASA, commercial, and international space agency partners). As a key element of this effort, there has been a particular emphasis on obtaining commercially provided SATCOM services wherever possible. Each commercial SATCOM service provider offers a set of services ranging in interoperability compliance from full to none, driving the need to establish community-wide standards for interoperability among providers and users to enable user systems to autonomously “roam” or transition between providers. This approach functions best when using open standards and protocols—increasingly of commercial origin—in all systems expected to interoperate, whether through contractual means or voluntary adoption.

Based on discussions with service providers, the high cost of space systems coupled with the business environment creates challenges in convincing SATCOM service providers to adopt interoperability with competitors. Absent financial incentives, a compliance-driven acquisition strategy, coupled with use of some form of industry-led consortium to encourage voluntary consensus on common use cases, requirements, V&V approach, and coordinated testing, may be needed.

The government can provide value-added services for such an arrangement by establishing and moderating a level playing field that spans all providers without perceived bias, particularly by providing common test capabilities and metrics. Alternatively, a purely

commercial solution can be considered for V&V testing and certification if it is widely adopted.

Changes in Automation Accommodation: Automation holds the promise of reducing overall system latency and cost (largely through a reduction in labor), while making it possible to conduct rapid response science in ways that were previously impossible. Without interoperable systems, automation may be limited within each system and expensive to implement across systems, particularly with multiple providers. Unfortunately, “although several technology demonstrators for highly automated systems already exist, there is a severe lack of cost-effective, commonly accepted V&V methods and the need for tools supporting these methods”. [7] Space systems automation is in its infancy with respect to interoperability, making achievement of goals in both areas challenging.

Changes in Security: Emerging cyber threats to systems in space, coupled with changes in technology, capabilities and new co-mingled commercial, civil, and defense systems and services, require dramatic changes in space security for architectures and systems. For this reason, integrating cybersecurity into all development phases and ensuring cybersecurity throughout the life cycle is essential for space systems. Open architectures like the Internet have now addressed standardization of security capabilities. While the alarming growth in the number and severity of cyber incidents is forcing more attention to developing systems that are both secure and interoperable, security remains an area that is prone to secrecy as an approach that can sacrifice interoperability.

From a V&V security standpoint, the potential for bad actors to inject malicious code into space systems, seize control, or disrupt operations needs to be tested. The extent to which this potential vulnerability exists in the architecture can have impacts ranging from service interruptions to complete loss of the system. Red team testing—which uses ethical hacking to identify breaches to an organization’s security system and pits a team of adversaries (the Red team) against an organization’s security team (the Blue team)—of the end-to-end, multi-system security scheme in relevant conditions could be performed as part of V&V to identify vulnerabilities and eliminate them prior to system operations. [8]

Security is driven by risk-cost/benefit analysis. Security requires a large amount of pre-shared information, as well as structure and capabilities. End-to-end SoS security requires that security policies and processes overlap, but policies and system capabilities differ between systems. For example, Internet of Things devices have minimal security capabilities, while host-based computing platforms have many more capabilities. Security V&V is performed within and across systems’ boundaries because the interoperating parties must mitigate the risks of interaction with each other at system boundaries to enable end-to-end data flow.

If System A wants to talk with System B, some form of external coordination must take place since the domains do not overlap. Systems A and B must protect against attack or intrusion by each other as well as by external entities. Interoperability between the systems will require a shared common context (threat environment) and a shared trust mechanism for security. For example, stakeholders need a shared method for authenticating users and staff. The extent of trust between different system owners is driven by the degree to which such mechanisms can be verified by Red team attacks in addition to traditional V&V.

Security is required by laws and defined by regulatory groups and these requirements are constraints on a system, for example:

- *General Data Protection Regulation* is a European regulation implemented to enhance European Union citizens' control over the personal data that can be legally exchanged. [9]
- *Personally Identifiable Information (PII)* uses data to confirm an individual's identity. Some PII is necessary to establish authentication but must be protected from internal and external attempts to acquire such data. [10]

- **Protected Health Information (PHI)** is any health information that includes any of the 18 elements identified by the Health Insurance Portability and Accountability Act (HIPAA). In the lunar scenario, this applies to crew health data. [11]

The ISO/IEC 27000 series of Information Security Management Systems (ISMS)-related standards is one avenue for establishing trusted, shared security mechanisms within a SoS.

3.3. Sustaining Interoperability

In this section, sustainment is divided into governance and implementation. Governance is essential to develop and sustain interoperability. It is defined here so the reader recognizes its complementary nature, but will be treated in a future paper.

In 2022 the Interagency Operations Advisory Group (IOAG)—which promotes interoperability between civil space agencies—established the Committee to Study LunaNet Governance, which defines governance in its Terms of Reference [12] as: “The initial working definition of LunaNet governance is *the development and application by governments, the private sector and civil society, in their respective roles, of shared principles, norms, rules, decision-making procedures, and organizations that shape the evolution and use of LunaNet in the context of the relevant international legal framework.*” This is based on the definition of Internet governance established by the United Nations-initiated *World Summit on the Information Society*. [13]

Implementation approaches to sustain interoperability are addressed from a strategic point of view at the international architecture level along with coordinating the evolution of interoperable services.

Sustaining Interoperable Architecture: A stable ecosystem sustains interoperability when the community of stakeholders defines the interoperable architecture and maintains that architecture as it evolves over time. Evolution may be due to changes in user community needs, standards revisions, technology improvements, introduction of new capabilities, and obsolete capability retirement.

Figure 10 depicts a simplified concept for maintaining configuration control of the architecture via a notional Architecture Board (AB). Members of the Architecture Board would be subject matter experts in various aspects of the architecture including operations, security, global and local architecture, Provider/User interaction, communications and networking technologies, etc. Participation in this Board must be broad and allow a wide assortment of stakeholders access to information that defines the architecture, the status and details of changes being considered, educational and training material, access to the accumulated body of information related to the architecture for historical and trend analysis, and other material.

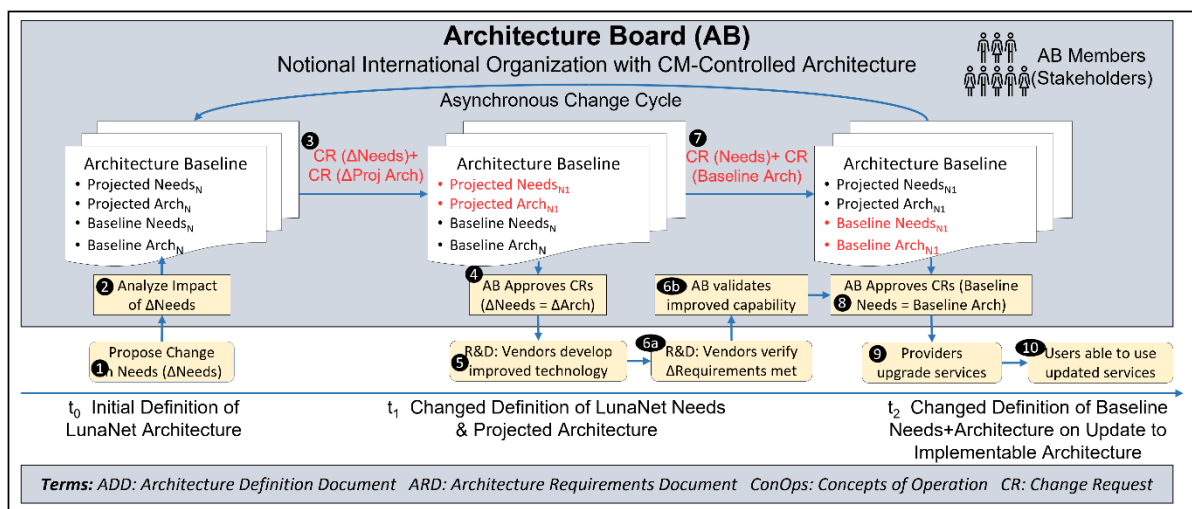


Figure 10. Sustaining Interoperable Architecture at the Architecture Control Level

The change control process outlined in Figure 10 assumes that the Architecture Board and a stable baseline exist at time t_0 . It begins (Step ❶) with stakeholder(s) proposing and introducing information describing changes in needs perceived to be emerging or in development such as increased demand for specific services, advances in technology, or operation under different scenarios or market conditions. The AB must be able to draw on expertise capable of evaluating the impacts of these proposed changes in needs. The evaluation (Step ❷) must validate the proposed changes and ascertain whether one or more viable approaches exist (or can be developed) capable of meeting those needs. This includes reviewing the Architecture Baseline (Step ❸) and the current state of capacities and capabilities over the set of Providers to quantify differences. Assuming a favorable outcome in this analysis, the result yields proposed changes to the Architecture Baseline (Step ❹) to capture the change in Projected Needs. This would include documenting analysis of alternatives and test results. The result is captured in a Change Request (CR) that proposes specific changes from the baselined Projected Needs_N and Projected Architecture_N—the accumulated result of prior approved CRs—that will produce an updated baseline including Projected Needs_{N+1} and Projected Architecture_{N+1}. This CR advances to the AB to formally approve the CR, thus recognizing the Users' and/or Providers' revised needs and the architectural response to meet them. The Projected Architecture should be defined independently of the technologies that may be used but may reflect the capability that improved technology is anticipated to provide.

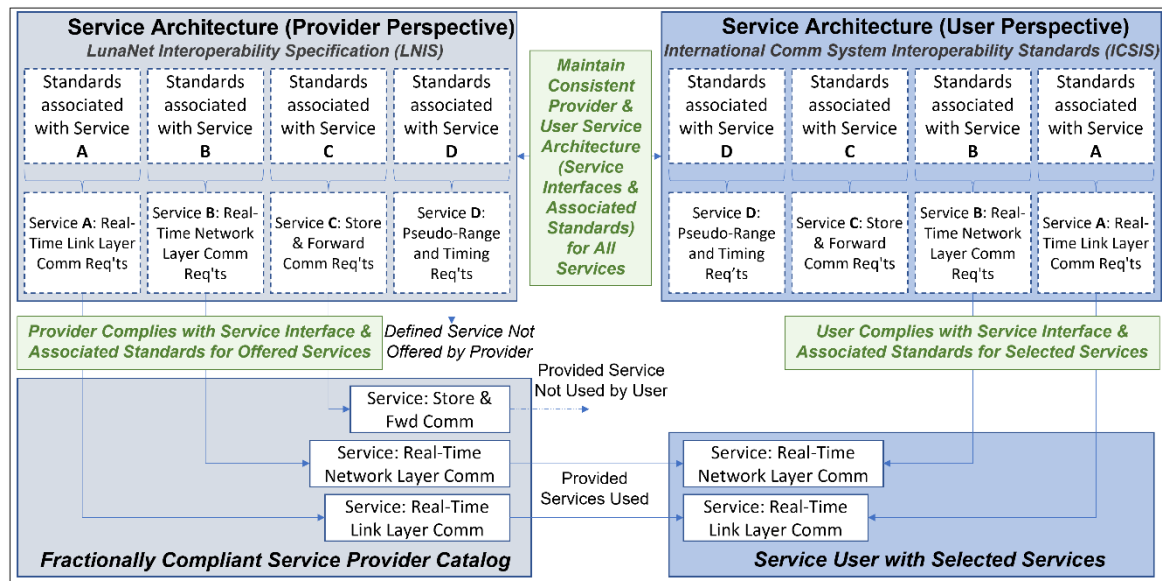
Industry responds to the revised Projected Needs by investing in Research and Development (R&D) that may generate competing means of meeting the needs (Step ❺). One or more feasible solutions emerges and is verified by industry (Step ❻a) as complying with the Projected Architecture, possibly with proposed modifications, since real technology may differ from anticipated architecture. Industry submits proof of their ability to the AB (Step ❻b) and the AB again relies on evidence to validate industry's claims. The AB generates a second set of proposed changes to the baseline via a CR that will revise the Baseline Needs_N and Baseline Architecture_N (Step ❼). The AB approves the CR (Step ❸) producing the new Baseline Needs_{N+1} and Baseline Architecture_{N+1}. This enables the Provider community to upgrade their systems and begin providing the enhanced services (Step ❾) at the performance level required to meet the increased User and/or Provider needs. Users can request services from Providers offering enhanced services (Step ❿).

Architecture and Service Flexibility: Another important aspect to consider in a sustainable interoperable architecture is that interoperability must allow the needs of both service providers and users to be shared, even if they are not identical. The architecture must be defined in a way that establishes a shared context between providers and users, allowing providers to make decisions about the services they plan to offer to users. The provider may decide to implement the full set of defined services, offer a subset of those services, or offer a subset of services as its initial capacity with the plan to expand its service delivery in the future. The concept that the provider and user can independently decide to implement subsets of non-identical services defined in the architecture is known as *fractional compliance*.

Figure 11 depicts the relationship between the provider-oriented service architecture—based on and using as an example the LunaNet Interoperability Specification (LNIS) (upper left), the user-oriented service architecture—based on and using as an example (ICSIS) (upper right), the provider's decisions about the services to be offered to users (lower left), and the user's decisions about which services to use (lower right). The service architectures of the provider and user must be complete, define all currently agreed services, and be consistent with each other. These service architectures do not contain information about temporal dependencies, such as phases or increments.

The provider's design in the lower left demonstrates the provider's decision to offer Services **A**, **B**, and **C**. The provider elects not to provide Service **D**. Consequently, the requirements in the service architecture associated with Service **D**, including any applicable standards, do

not apply to the provider. The provider does not have to verify compliance with the requirements of Service **D**. Therefore, the provider is fractionally compliant, that is, the provider fully complies with the requirements of the services it plans to offer and must verify that its system meets with those requirements. The provider does not have to comply with requirements associated with services that will not be offered to users.



The user's design in the lower right demonstrates the user's decision to use only Services **A** and **B**. The user elects not to use Service **C** even though the provider offers it. The user does not have the option to use Service **D** as it is not provided (unless an alternate provider exists). Therefore, the user is fractionally compliant, that is, the user fully complies with the requirements of the services that it decides to use and must verify that compliance. The user does not have to comply with requirements associated with services that are not used.

A valid option, not shown in Figure 11, is to define a single Service Catalog. This Service Catalog could provide the user with the same options as providers, or it could explicitly define the requirement for providers to comply with all requirements associated with a particular service, while users are only required to comply with a selectable subset.

Sustaining Interoperable Operations: Interoperability is more than an abstract ideal. It requires effort to develop and sustain over the life of a SoS that evolves over time. Concepts associated with sustaining interoperability are integral to the definition. Aspects related to sustenance must be feasible, affordable, and incorporated from the outset of establishing the SoS. This section describes key concepts for sustainable interoperability and demonstrates their feasibility.

Figure 12 and accompanying Table 1 show a notional process used to maintain interoperability with little or no service interruption while propagating a revision to one service. Interactions occur between a SDO that owns the standard(s) that define interoperable services offered by a set of Service Providers to a set of Service Users. These users are treated as space systems. They are launched with a configuration of hardware (HW) and software (SW) with no assumed in-space servicing capability. Consequently, flight SW can be modified from the ground-based Mission Operations Center, while flight HW may not be modifiable or have limited modification ability, e.g., using Field Programmable Gate Arrays. The ability to modify the flight system's configuration in space constrains the ability to evolve the space system in operation, e.g., via an SDR. For this example, only the flight SW is assumed to be modifiable and service modifications are similarly constrained.

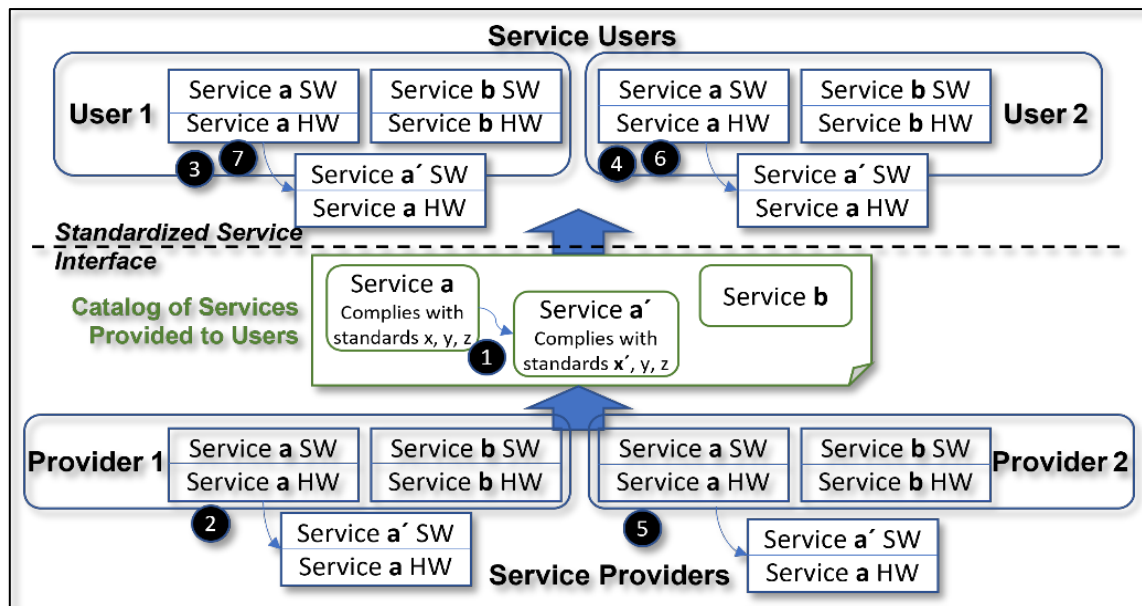


Figure 12. Process Preserving Interoperability Through Asynchronous Change

Table 1. Steps in Process of Preserving Interoperability Through Asynchronous Change

Scenario Assume Users 1 and 2 have SDRs and have Service Level Agreements (SLA) with both Providers 1 and 2 to use Services a and b .		
Provider 1 → Users	Step ❶	SDO revises its standard from Revision x to Revision x' , which impacts Service a . Results: The revised standard is published and the user community is notified. There is no impact to current Service a between Providers 1 and 2 and Users 1 and 2.
	Step ❷	Provider 1 revises its software to deliver Service a to Service a' . Results: Provider 1 revises its software to deliver Service a' , per standard x' , and announces availability of Service a' to users, while continuing to provide Service a .
	Step ❸	User 1 converts from using Service a from Provider 1 to Service a' . Results: User 1 modifies its software from receiving Service a to Services a <u>or</u> a' , and then changes its SLA with Provider 1 to receive Service a' . User 1 uses Service a from Provider 2 and Service a' from Provider 1. No impact occurs to current Service a between Providers 1 and 2, and User 2.
	Step ❹	User 2 converts from using Service a from Provider 1 to Service a' . Results: User 2 modifies its software from receiving Service a to receiving Services a <u>or</u> a' , and then changes its SLA with Provider 1 to receive Service a' . Users 1 and 2 use Service a from Provider 2 and Service a' from Provider 1. Again, no impact occurs to current Service a between Provider 2 and User 2.
Provider 2 → Users	Step ❺	Provider 2 revision from Service a to Service a' . Results: Provider 2 modifies its software from delivering Service a to a' , while continuing to provide Service a to Users 1 and 2.
	Step ❻	User 2 converts from Service a to Service a' from Provider 2. Results: User 2 modifies its software from receiving Service a to Services a <u>or</u> a' , then changes its SLA with Provider 2 to receive Service a' . At this point, User 1 uses Service a from Provider 2 and Service a' from Provider 1.
	Step ❼	User 1 converts from Service a to Service a' from Provider 2. Results: User 1 already has software updated to handle Service a' . It updates its SLA with Provider 2 to change from Service a to Service a' . User 2 uses Service a' from Provider 2.

Additional scenarios can be devised to accomplish a “make before break” approach to sustainment that preserves interoperability throughout the evolution process. However, there are many ways to break or interrupt interoperability by not following a well-designed process or by ignoring the consequences of losing interoperability.

4. Conclusion

Interoperability has been converted here from a high-level concept into an engineering entity amenable to systems engineering methods. This enables systems to be modeled, developed, tested, operated, and evolved with high assurance of achieving quantifiable and verifiable goals.

Is interoperability worthwhile? While commercial terrestrial SATCOM has virtually no interoperability among providers, NASA and its international partners envision the emerging cislunar environment establishing—from the very beginning—interoperable communication, networking, and PNT services offered by multiple providers for a rapidly growing set of user missions. Interoperability is a game-changer for the cislunar community **if it can be initiated and sustained** due to the many benefits it offers, including:

- Opportunity for commercial providers to pursue the entire lunar market as users come to depend on interoperable services.
- Increased capacity, coverage, availability, and reliability of communication services with lower latency for users because of the flexibility to use any or all providers, as well as enhanced ability to recover from off-nominal conditions.
- Higher accuracy PNT service for position and orbit determination for users by combining broadcast signals from all providers allowing better situational awareness and contributing to future space traffic management.
- Creation of a competitive cislunar market with benefits of access to capital for further investment, stimulus for R&D, and robust management of costs, schedules, and risks.

LunaNet will offer new services in space for space systems, rather than over space links for terrestrial systems. Interoperability in LunaNet will have positive impacts on lunar space operators and potentially benefit lunar space missions, regardless of government, academic or commercial sponsorship, by enabling broad use of new capabilities, including Delay/Disruption Tolerant Networking with in-orbit data storage, distributed processing and cloud functions, Internet of Things, new science services, and support for autonomy in mission systems.

There are other systems engineering aspects that play an important role in interoperability that have not been covered, such as the sustainability of operations and governance aspects associated with an international interoperable and SoS architecture. As work for LunaNet progresses, these areas and others not included in this document will be developed.

5. References

- [1] INCOSE Guide to the Systems Engineering Body of Knowledge (SEBoK), v. 2.9, November 20, 2023, [https://sebokwiki.org/wiki/Model-Based_Systems_Engineering_\(MBSE\)__\(glossary\)](https://sebokwiki.org/wiki/Model-Based_Systems_Engineering_(MBSE)__(glossary))
- [2] OMG Systems Modeling Language, <http://www.omg.sysml.org/>.
- [3] Dassault Systèmes, “Cameo systems modeler,” 2022, <https://www.3ds.com/products/catia/no-magic/cameo-systems-modeler>
- [4] Musen, M.A. The Protégé project: A look back and a look forward. AI Matters. Association of Computing Machinery Specific Interest Group in Artificial Intelligence, 1(4), June 2015. DOI: 10.1145/2557001.25757003
- [5] IEEE, “A Compilation of IEEE Standard Computer Glossaries,” New York, Standard 1990.
- [6] backward compatible (backward compatibility), Katie Terrell Hanna, TechTarget, <https://www.techtarget.com/whatis/definition/backward-compatible-backward-compatibility>
- [7] Enable-S3, Testing and Validation of Highly Automated Systems: Summary of Results, May, 2019, https://www.tugraz.at/fileadmin/user_upload/Institute/IHF/Projekte/ENABLE-S3_SummaryofResults_May2019.pdf.
- [8] “Red Teaming: How Red Team Testing Prepares You For Cyberattacks”, CrowdStrike, JJ Cranford, July 7, 2023, <https://www.crowdstrike.com/cybersecurity-101/red-teaming/>.
- [9] Data protection in the EU - European Commission (europa.eu), https://commission.europa.eu/law/law-topic/data-protection/data-protection-eu_en
- [10] Guidance on the Protection of Personal Identifiable Information | U.S. Department of Labor (dol.gov), <https://www.dol.gov/general/ppii>
- [11] Summary of the HIPAA Privacy Rule | HHS.gov, <https://www.hhs.gov/hipaa/for-professionals/privacy/laws-regulations/index.html>
- [12] Committee to Study LunaNet Governance Terms of Reference (ToR), June 12, 2023, https://www.ioag.org/Documents/Committee%20to%20Study%20LunaNet%20Governance%20ToR_for%20signatures%20-%20signed.pdf.
- [13] Tunis Agenda For The Information Society, WSIS-05/TUNIS/DOC/6(Rev.1)-E, November 18, 2005, <https://www.itu.int/net/wsis/docs2/tunis/off/6rev1.pdf>, clause 34.

6. Acronyms

3GPP	3 rd Generation Partnership Project	ITRS	International Terrestrial Reference System
AB	Architecture Board	LNIS	LunaNet Interoperability Specification
ADD	Architecture Definition Document	LNSP	LunaNet Service Provider
AOS	Acquisition of Signal	LOS	Loss of Signal
ARD	Architecture Requirements Document	MBSE	Model-Based Systems Engineering
CCSDS	Consultative Committee for Space Data Systems	NASA	National Aeronautics and Space Administration
ConOps	Concept of Operations	OMG	Object Management Group
CR	Change Request	PHI	Protected Health Information
CSM	Cameo Systems Modeler	PII	Personally Identifiable Information
CTRL	Control	PNT	Positioning, Navigation, and Timing
DTE	Direct to Earth	POR	Pressurized Oxygen Regulator
EVA	Extravehicular Activity	PPP	Public-Private Partnership
HAB	Habitat	QoS	Quality of Service
HIPAA	Health Insurance Portability and Accountability Act	R&D	Research and Development
HLS	Human Landing System	RF	Radio Frequency
HMI	Human-Machine Interoperation	SATCOM	Satellite Communications
HTML	HyperText Markup Language	SDO	Standards Development Organization
HW	Hardware	SDR	Software Defined Radio
ICSIS	International Communication System Interoperability Standards	SI	System of Units
IEC	International Electrotechnical Commission	SoS	System of Systems
IEEE	Institute of Electrical and Electronics Engineers	SW	Software
ILRF	International Lunar Reference Frame	SysML	Systems Modeling Language
INCOSE	International Council on Systems Engineering	TLM	Telemetry
IOAG	Interagency Operations Advisory Group	UHF	Ultra High Frequency
IP	Internet Protocol	US	United States (of America)
ISMS	Information Security Management Systems	V&V	Verification and Validation
ISO	International Standards Organization	xEMU	Exploration Extravehicular Mobility Unit
ITRF	International Terrestrial Reference Frame	XML	eXtensible Markup Language

The authors certify that they have no conflicts of interest.