FLEXIBLE MODEM INTERFACE (FMI) IN SPACE – EXTENDING STANDARDIZED COMMERCIAL SATELLITE COMMUNICATIONS SERVICES TO SPACE USERS

Daniel Zeleznikar
National Aeronautics and Space Administration: Glenn Research Center at Lewis Field
21000 Brookpark Rd, Cleveland, Ohio, 44135, 216-433-3616, daniel.j.zeleznikar@nasa.gov

Abstract

Recent innovations are producing a multitude of advanced commercial satellite communications (COMSATCOM) systems that could deliver massive amounts of SATCOM capacity at a fraction of the current cost while also offering reliability and availability that is critical to achieving mission success for orbiting assets. Recognizing the alignment of commercial capabilities with the National Aeronautics and Space Administration's (NASA) diverse mission requirements, the agency is proactively engaging industry to formulate strategies leading towards a NASA communications architecture that includes advanced commercial capabilities. To fully leverage the expanded space resources, NASA must also address the integration of a diverse set of commercial waveforms into its space terminals. In pursuit of similar goals, the United States Department of Defense (DoD) is leading the standardization of the flexible modem interface (FMI) to address service integration for their tactical terminals in pursuit of a DoD Wideband SATCOM Enterprise.

This paper describes NASA's vision for adapting this FMI standard to work with the Space Telecommunications Radio System (STRS) software-defined radio (SDR) framework. The goal of the space FMI is to provide the capability of using both government and commercial services in the Ka-Band spectrum while meeting the size, weight, and power (SWaP) constraints for spacecraft user terminals. To achieve this, the space FMI approach is software based, allowing space users to get the benefits of the FMI approach without swapping physical modules. Security is also a key aspect for this software integration since data will flow through commercial networks, commercial service providers have their own security mechanisms, and space terminals must be able to securely load proprietary software and firmware needed to access networks on demand. Success of this effort means commercial partners will be able to allow network-compliant implementations to be hosted on STRS-compliant SDRs in space for reliable and capable network access.

1.0 Introduction

U.S. National Space Policy expresses guidelines that promote the use of commercial space capabilities and services to the maximum practical extent. Though NASA spacecraft have traditionally employed communication services and infrastructure maintained by NASA, new commercial entrants into this domain are expected to be capable of providing service to spacecraft that can ultimately reduce the cost of providing communications services to assets in space. With this in mind, NASA's Space Communications and Navigation (SCaN) Office has investigated the use of new architectures and technologies that are leading towards a NASA communications architecture that integrates emerging advanced commercial capabilities. [1] Commercialization efforts will initially pursue opportunities that will allow future NASA missions to deploy flight qualified capabilities for near-Earth users to obtain SATCOM services from commercial providers. Longer-term, NASA will be responsible for the acquisition, management, and costs of future operational satcom services as government assets are retired.

In order to employ these commercial services, both compatible space terminals (including low footprint technology) and compatible ground architectures (including network integration) are necessary. Given that the commercial solutions in development are highly heterogeneous, it is challenging to provide terminal compatibility with every network. However, there would be significant architectural risk in designing a terminal toward compatibility with only a single network provider or even a small subset from a hardware perspective. Therefore, NASA hopes to develop a terminal architecture that is capable of air interface compatibility with commercial networks via software modification only. This paper covers what is required on the terminal side to make this happen from both a hardware and software perspective with a focus on the software and systems that will have to be integrated and tested to be successful.

The next subsection describes the effort led by the DoD called FMI that NASA hopes to learn from and align with, followed by a subsection reviewing NASA's in-house SDR architecture standard called STRS. The second section of this paper describes an approach to integrating STRS-based SDRs and FMI, and technology developments that allow for an SDR-based universal space communications terminal that in convergence with FMI and STRS, offer the ability for NASA spacecraft to use these emerging commercial networks. In section three, unique security considerations for the space terminal platform are discussed. In section four, a list of notional risk-reduction demonstrations is offered. Finally, closing remarks are provided.

1.1 Flexible Modem Interface (FMI)

FMI is a control plane interface standard under development by the DoD in concert with its terminal and service providers that provides the capability to verifiably control and configure a multi-service-enabled satellite communications (SATCOM) access terminal. [2] It is being developed in order to enable heterogeneous SATCOM network roaming capability within the future DoD SATCOM architecture. Enabling such capability is a critical step towards achieving affordability, resilience, and performance objectives of the future architecture that consists of space, infrastructure, tactical, and management segments. Though FMI is principally a terminal interface, it is designed around the desired end-state architecture and exposes the configurable elements that are required to achieve satellite access flexibility, network-to-network roaming, global terminal mobility, situational awareness, network management, and even remote terminal provisioning. [2]

The FMI development and standardization effort addresses the design, integration, operation, and management of a tactical terminal that contains multiple modems (often proprietary) for operating on multiple respective networks. FMI relies on the existence of a flexible terminal. A diagram showing this flexible terminal (taken from the original FMI paper) can be seen in Figure 1. The scope of terminal flexibility and FMI control includes antenna selection, antenna pointing, RF front-end configuration including power control, a multi-waveform modem system, and internal terminal data routing and switching. [2]

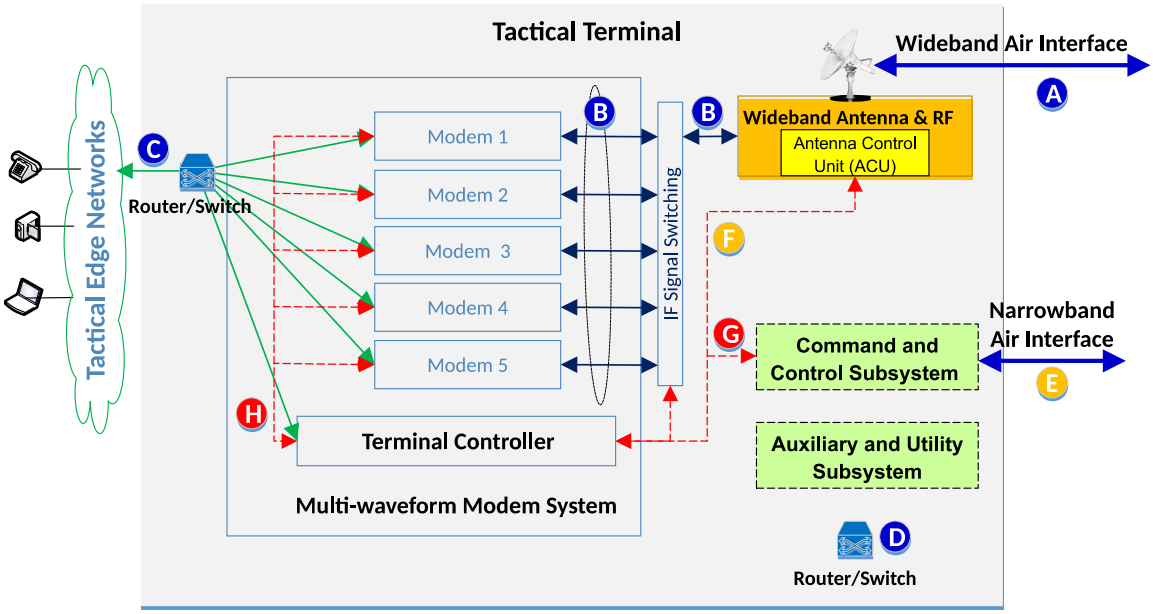

Figure 1 – Flexible Tactical Terminal Diagram. Reproduced from [2]

The FMI effort addresses many key challenges for a roaming-capable terminal standard. One major challenge is vendor business models & customer culture inertia. Another major challenge is that the terminals and control protocol must be flexible enough to accommodate the fact that network assets in space are highly heterogeneous with respect to orbits, network management, throughput capability, coverage areas, beam sizes, RF characteristics, operating bands, and network side interference protection techniques. Finally, the various user terminals it intends to work with are disparate, with evolving tactical requirements driving the proliferation of modem divergence. To achieve its goals, FMI's key design features include industry standardization, application layer operation, remote management

capability, extensibility to future networking capabilities, and security with respect to both access and availability. [2]

1.2 Space Telecommunications Radio System (STRS)

STRS is the open architecture developed by NASA for space and ground SDRs, and is the NASA standard for space SDRs. STRS provides a common, consistent framework to abstract application software from the SDR platform hardware to reduce the cost and risk of using complex reconfigurable and reprogrammable radio systems across NASA missions. Figure 2 below shows an illustration of the STRS architecture.

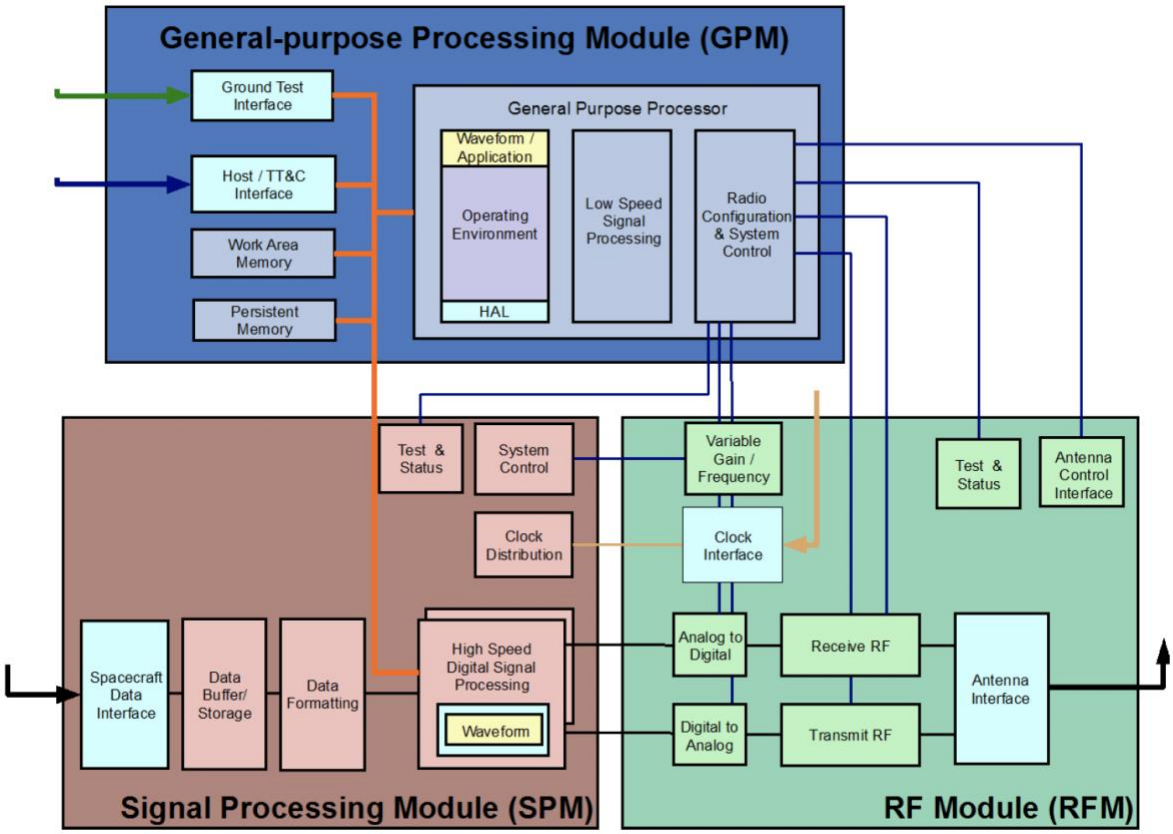

Figure 2 – Modular STRS Architecture. Reproduced from [3]

The architecture contains modular functional processing components, where each module performs unique tasks related to its purpose. Although not shown, other modules such as security modules, network modules, or optical modules may be included in the specification as it matures.

The standard importantly defines specific API methods both for calling into the main STRS operating environment (OE), and for the OE to interact with waveforms and devices. From a software implementation perspective, STRS takes the form shown in Figure 3, exposing both application/device APIs and STRS APIs. STRS APIs provide a way for both external systems and internal applications and devices to interact with the overall STRS operating environment (OE).

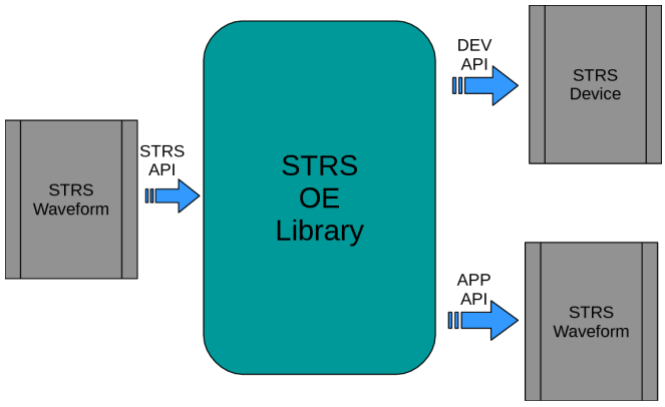

Figure 3 – STRS Software Architecture Interfaces

Compared to other SDR architecture standards, the STRS library aims to be lightweight so that it may be usefully run on

spacecraft with limited resources. As such, the current standard does not require the use of any middleware layers. STRS has been proven in the low-earth orbit spaceflight environment while in use on NASA's SCaN Testbed (STB) SDR demonstration platform. During its tenure on the International Space Station (ISS) from 2012 to 2019, NASA conducted more than 4,200 hours of testing and demonstrated routine reconfiguration of radios in space 888 times. STRS provides a flight-proven SDR architecture with flexibility to support operation of both traditional NASA network waveforms and emerging commercial network waveforms. An STRS waveform repository also exists which promotes the reuse of flight proven software and firmware to reduce cost, schedule and risk for future missions.

2.0 Universal Space Communications Terminal and Standards Convergence

To enable FMI in the constrained space environment, the terminal flexibility must be realized entirely through software. This is directly achievable by implementation of FMI as an STRS application within the STRS architecture. The application would present the FMI-standard interface to both the host spacecraft network and the enterprise ground network for control via in-band communication channel messages. FMI is important because it provides a standardized interface to control which radio waveforms and physical components are active at any given time. STRS is important because it provides a trusted SDR framework for activating waveforms as needed, including modem functions and frequency, timing, and power controls, acting as the software alternative to unrealizable physical hardware changes.

As explained previously, realization of a flexible terminal has many challenges. When realized as a space-based SDR, this becomes even more challenging due to the need for a specific SDR-based waveform implementation using space-qualified hardware and software within the challenging size, weight, and power constraints of a spacecraft. Though a flight-proven SDR architecture has already been established, and an adaptable heterogeneous service control plane protocol is rapidly under development by the DoD, flexible space-rated hardware is needed to meet the challenge. In the realization of this flexible multi-provider capable terminal, low cost electronically-steerable antennas (ESAs), wideband front-ends, and programmable logic devices (PLDs) provide the hardware capability that is necessary. Figure 4 illustrates this clearly from the space radio's hardware component perspective as it communicates to heterogeneous SATCOM networks.

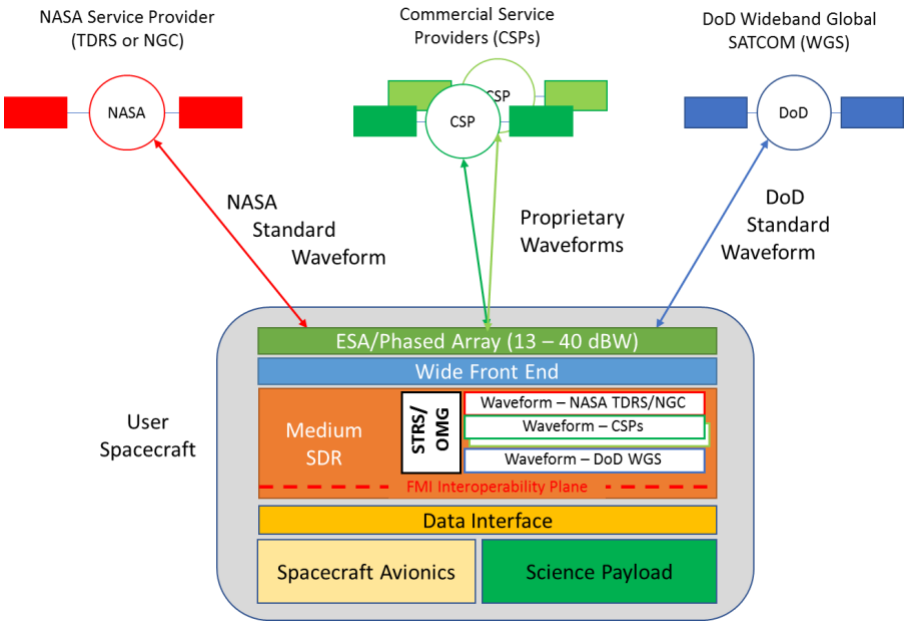

Figure 4 – Space-based Flexible Terminal with STRS

Meeting the challenge with respect to ESAs, recent technology development and maturity in miniaturization has led to a variety of phase-array antenna offerings including in the Ka-Band with the promise to provide rapid steering and even arbitrary beam synthesis capability in a package that is amenable for space applications. A front-end encompasses filtering, amplification, and frequency conversion. Applicable wideband versions of this for space require miniaturization of high-bandwidth amplifiers and mixers, and configurable networks of filter banks. Realizations of this form in terrestrial environments already exist, but must be hardened to meet the harsher requirements for space. On the modem side, PLDs have provided capability for high bandwidth waveforms for many years now in space.

3.0 Security Challenges

With the migration to an architecture that includes commercial infrastructure-perhaps even multiple commercial parties simultaneously-there are a number of security aspects that must be considered and components that must be appropriately secured since some elements in the system are no longer considered as intrinsically trusted. Each additional security provision will also bring certain costs and complexities. With respect to an FMI-enabled space terminal utilizing an SDR, there are unique aspects above and beyond those that the FMI control interface and accompanying data interface address. This includes protection of network-enabling proprietary software and firmware designs that run on the shared SDR components, procedures for securely developing and deploying software updates, integration of vendor-compatible security components (potentially including specialized hardware components), and management of sensitive operational data. This section will provide a brief description of each of these security aspects, and how each aspect might be addressed on a standardized SDR platform.

While some layers of proposed commercial network stacks are based on open standards, it is expected that certain commercial innovations needed to access networks will be proprietary in nature, thereby offering customers enhanced services and the corresponding network providers a marketplace advantage. It is therefore important that these design innovations be protected from discovery and/or unauthorized use by other parties. Achieving the appropriate level of intellectual property protection is possible on the proposed shared resource-constrained platform using an architecture such as STRS with additional security extensions, though vendors must cooperate to achieve consensus on the exact additional security requirements that will provide the level of protection they expect. Nonetheless, it is expected that the available software libraries and programmability features already in existence on modern computing platforms can provide all the security capability that will be required in a low-overhead manner.

For example, proprietary vendor software and firmware can be signed and encrypted, providing secrecy and authenticity of running programs on the platform if needed. Table 1 shows the equivalent level of features that certain forms of software and firmware provide with respect to obfuscation, auditability, compiler optimization capability, cryptographic accessibility, and portability.

Table 1: Software/Firmware Security and Features

General Form	Protection	Hardware Description Language (HDL)	Compiled Languages (e.g. C/C++)	Portability & Compiler Optimization
Source Code	None	Plain HDL	Plain C code	Unrestricted
Encrypted Source Code	Third-party Use	Encrypted HDL (only compiler can decrypt)	Encrypted C (can decrypt w/key)	Restricted only to key holders
Pre-compiled Binary	Design Obfuscation	Pre-compiled netlist, Bitstream	Binary Shared Object	Restricted to originally targeted device architecture and compiler settings
Encrypted Binary	Design Obfuscation, Third-party Use, Reverse Engineering	Encrypted netlist, Encrypted Bitstream	Encrypted Binary Shared Object	Restricted to originally targeted device architecture and compiler settings only to key holders

At runtime, proprietary software can be invoked as a separate process with reduced platform privileges and a private memory space. Access to dedicated hardware processing devices can be controlled by the STRS OE through a low overhead STRS API middleware layer. For dedicated processing hardware such as FPGAs, partial reconfiguration features [4] can be used to load new hardware processing as needed on-the-fly. Within the STRS architecture, security extensions could be implemented by simply using a security-enhanced OE implementation and STRS wrappers provided by the government around designs that implement the STRS standard as it exists today, enabling backwards-compatibility with existing STRS-compliant applications as well. Figure 5 shows a representative formulation of secure STRS on a UNIX-based operating system (OS) utilizing standard OS features. This formulation also delineates provider deliverables from government-provided support components, including keys that would need to be provided in order to enable encrypted binaries if deemed necessary.

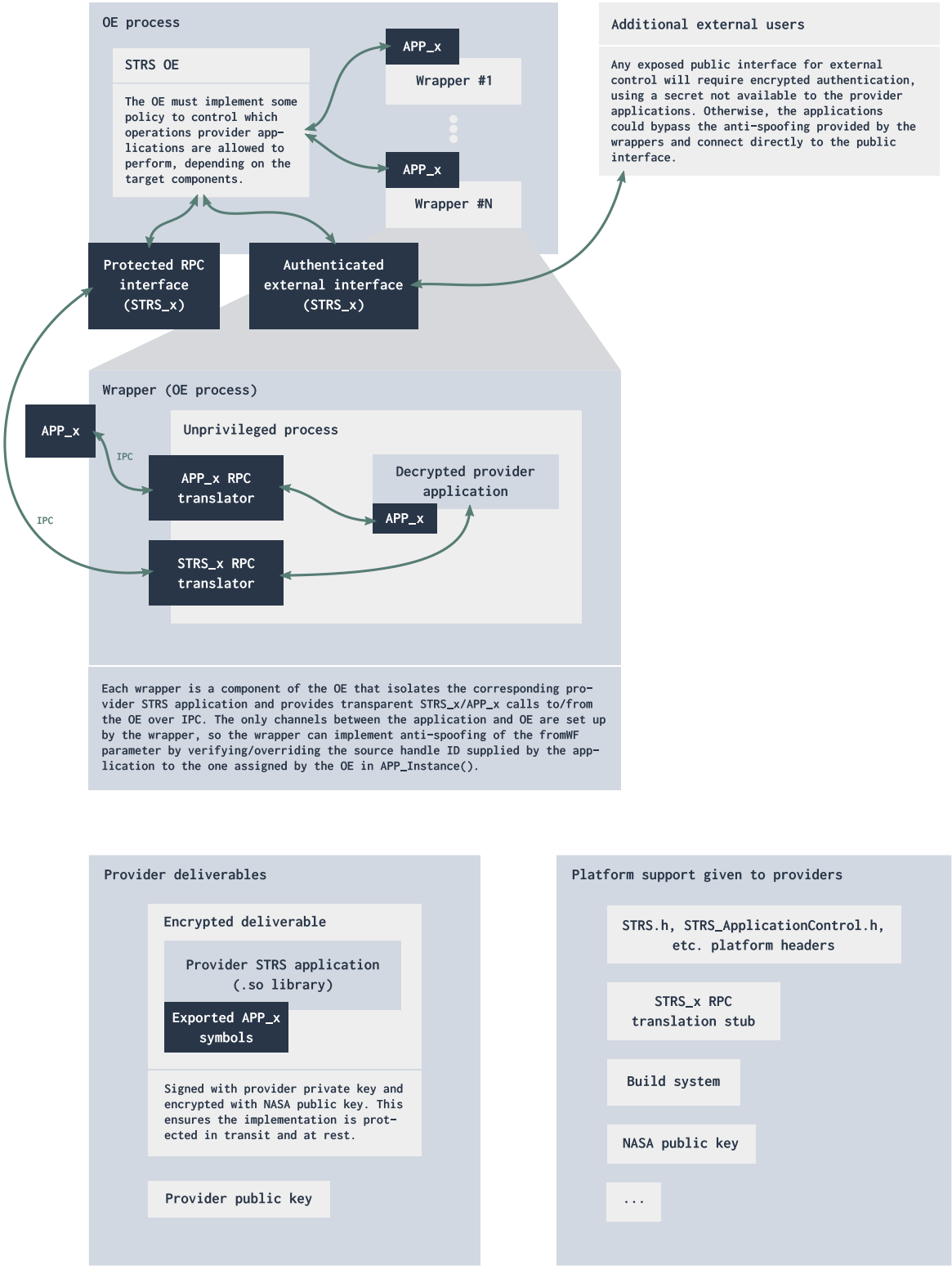

Figure 5 – Notional approach to secure STRS architecture with guest waveform applications

Just as vendors will test their designs and updates before deployment to ground user terminals, vendors and perhaps the end user in this case must be able to do the same to ensure proper operation on secure STRS SDR platforms. In order to make the development process more robust and trusted, a developer operations framework can be adopted with many of the industry standard automated steps of testing, analysis, staging, and ultimately deployment. It may also be possible to tie-in vendors existing development flows for various ground terminals. The traceability in this type of flow through to a

customer engineering platform would provide mission users with the level of confidence and demonstrated maturity they desire for over-the-air updates.

One aspect that may be more unique from one vendor to another is how network authentication and associated user provisioning is performed. In cellular user equipment, SIM (subscriber identification module) cards are traditionally used for this, requiring physical card changes in order to use networks unaffiliated with a home subscriber network, which is not amenable to a space platform. Even with the advent of the programmable SIM card, constant reprogramming for many networks at the rate that a space platform could roam may prove impractical. Therefore, other compatible means of providing the necessary network authentication should be determined-new STRS API methods might be required to support this. Similarly, there must be a way for sensitive operational parameters such as session keys to be especially hardened from third-party access within a secure STRS implementation. A dedicated secure crypto-processor such as the internationally standardized Trusted Platform Module (TPM) [5] may provide for this, and perhaps also serve as a more extensible but compatible device for network authentication. Additional STRS API methods might be required for a restricted STRS application to access this device.

As has been discussed, securing an FMI-enabled multi-service capable SDR platform will require additional work. In an effort to gain commercial vendor support for the proposed architecture, NASA will attempt to ease these burdens by seeking commercial cooperation and acceptance on security extensions to the SDR standard and providing wrappers, reference designs, and technical support. A more rigorous study of this topic including formulation in a threat and risk-based analysis is underway in a parallel NASA effort.

4.0 Risk-Reduction Demonstrations

The path to realization of the proposed universal space terminal requires cooperation from many parties, and should involve a series of demonstrations to root out potential issues that could occur in different parts of the system. Many of these demonstrations can be realized with off-the-shelf hardware from the ground to provide protocol standard maturity and risk reduction. In April 2019, NASA, in concert with DoD and Hughes Network Systems, completed a demonstration to show that FMI could be used with existing STRS-enabled spacecraft radios by interfacing with FMI-enabled ground-networked spacecraft control elements. [6] In this demonstration, a local terminal operator exercised control of an on-orbit spaceborne radio via the terminal operator interface (TOI) using the FMI-standard message set.

Additional demonstrations should be themed and phased to address particular challenges, and could be organized as follows:

Theme A: Feasibility
- STRS-based SDR + FMI App + NASA traditional WF App to/from NASA network simulator, followed by NASA network test from ground
- STRS-based SDR + commercial WF APP to/from commercial network simulator, followed by commercial network test from ground

Theme B: Security
- “Secure” STRS-based SDR + “secure” FMI App + “secure” NASA WF App to/from NASA test device
- “Secure” STRS-based SDR + “secure” commercial WF App to/from test device at commercial site
- “Secure” STRS-based SDR + “secure” commercial WF App, network test from ground

Theme C: FMI Network Integration
- “Secure” STRS-based SDR + “secure” FMI App + “secure” commercial WF App + “secure” NASA WF App, network roaming test from single location, perhaps DoD site
- SW Deployment test to remote terminal site, uploading “secure” commercial WF App

Testing the above components with off-the-shelf hardware will provide an opportunity to hone the systems and demonstrate true network capability in parallel with the maturation of spaceflight hardware.

5.0 Conclusion

The emerging COMSATCOM industry presents opportunities for NASA's future communications architecture to simultaneously offer enhanced services to users and reduce costs. The convergence of a capable and secure implementation of the STRS architecture standard, the development and maturity of the FMI control plane protocol on both the terminal side and throughout ground support networks, and the maturation of hardware to support a universal space communications terminal provides the capability to allow for interoperability from NASA assets on these new networks. Success of this effort means commercial partners will be able to allow network-compliant implementations to be hosted on STRS-compliant SDRs in space for reliable and capable network access, helping to support their commercial business case with both NASA and DoD users, a valuable clientele. The path to realization should be organized and focused on retiring specific development risks, both in hardware and software. Other FMI adopters may subsequently leverage the STRS-based space FMI to reduce costs for their terminals.

Acknowledgements

The author extends thanks to the local STRS development team of Lou Handler, Joe Hickey, and Mick Koch for informative conversations about the details of the STRS architecture standard, including advice for how it could be secured in practice. Thanks also go to the Air Force Space and Missile Systems Center for leading the FMI effort, and especially opening their conversations to NASA on the development of the standard, including access to developer implementation resources for experimentation purposes.

References

[1] National Aeronautics and Space Administration. "Next Space Technologies for Exploration Partnerships-2 (NextSTEP-2). Omnibus Broad Agency Announcement, Appendix G: Space Relay Partnership and Services Study." Released: Friday, September 18, 2018. Solicitation Number NNH16ZCQ001K-SRP.

[2] Joseph Vanderpoorten, Kevin Zhang. "Flexible Modem Interface – Enabling DoD Wideband SATCOM Enterprise**."** MILCOM 2017**,** Track 4 - System Perspectives. IEEE 2017.

[3] National Aeronautics and Space Administration. "Space Telecommunications Radio System Architecture Standard." NASA-STD-4009A. Approved 14 March 2018.

[4] Xilinx. "Vivado Design Suite User Guide: Partial Reconfiguration." UG909 (v2018.1) April 27, 2018.

[5] International Organization for Standardization, International Electrotechnical Commission. ISO/IEC 11889-1:2015. "Information Technology – Trusted Platform Module Library – Part 1: Architecture." August 2015.

[6] Robyn Atkins, Gerry Jansson, Nancy Kemper. "Flexible Modem Interface (FMI) On-orbit Experiment with SCaN Testbed Demonstrates Service Interoperability Baseline between US Government Agencies." 25th Ka and Broadband Communications Conference. Sept 30 – Oct 2, 2019.