In-Space Networking On NASA's SCaN Testbed

David Brooks[*]
Vantage Partners LLC, Cleveland, OH, 44135, USA

Wesley M. Eddy[†] and Gilbert J. Clark III[‡]
MTI Systems, Cleveland, OH, 44135, USA

and

Sandra K. Johnson[§]
NASA / Glenn Research Center, Cleveland, OH, 44135, USA

The NASA Space Communications and Navigation (SCaN) Testbed, an external payload onboard the International Space Station, is equipped with three software defined radios (SDRs) and a programmable flight computer. The purpose of the Testbed is to conduct in-space research in the areas of communication, navigation, and networking in support of NASA missions and communication infrastructure. Multiple reprogrammable elements in the end to end system, along with several communication paths and a semi-operational environment, provides a unique opportunity to explore networking concepts and protocols envisioned for the future Solar System Internet (SSI). This paper will provide a general description of the system's design and the networking protocols implemented and characterized on the testbed, including Encapsulation, IP over CCSDS, and Delay-Tolerant Networking (DTN). Due to the research nature of the implementation, flexibility and robustness are considered in the design to enable expansion for future adaptive and cognitive techniques. Following a detailed design discussion, lessons learned and suggestions for future missions and communication infrastructure elements will be provided. Plans for the evolving research on SCaN Testbed as it moves towards a more adaptive, autonomous system will be discussed.

I. Introduction

Since 2012, networking research has been conducted in space using the Space Communications and Navigation (SCaN) Testbed onboard the International Space Station[1]. The overall research goal is to advance the state of in-space networking technology through implementation, integration, and on-orbit operations experience with a complex system. Section I of this paper summarizes the motivation and goals and objectives for our networking research on the SCaN Testbed. Section II provides a brief overview of the SCaN Testbed system and the unique characteristics of the system that makes it an ideal testbed for networking research. Section III provides details of the networking design implemented on the testbed. Section IV covers the concept of detailed software instrumentation needed in a networked system of space assets and our experience with the definition and use of appropriate telemetry in the system. Section V summarizes the experiments completed to date, and planned future work. Additional intelligent routing technology that builds upon this work is described in a separate paper[2].

[*] Electrical Engineer V, GESS-3 Communications Group, David.E.Brooks@nasa.gov, Non-member

[†] Chief Technologist, wes@mti-systems.com

[‡] Software Engineer, gclark@mti-systems.com

[§] Electronics Engineer, Information and Signal Processing Branch, sandra.k.johnson@nasa.gov, Non-member

Motivations

To meet future complex missions' space communication needs, the space community recognizes a necessity to develop Space Internetworking (SI) capabilities that increase communication services automation and enhance cross-support between assets belonging to different agencies or other providers. In 2008, the Interagency Operations Advisory Group (IOAG) Space Internetworking Strategy Group (SISG) published recommendations on a strategy for internetworking in space[3]. This fed into the Solar System Internet (SSI) architecture developed within the Consultative Committee for Space Data Systems (CCSDS)[4] by an international collaboration of space agencies. This architecture includes SI services based on Internet Protocol (IP) and Delay Tolerant Networking (DTN).

NASA uses the SCaN's Space Network (SN), Near-Earth Network (NEN), and Deep Space Network (DSN) communication elements to provide dozens of missions' command and telemetry services using physical layer (e.g. relaying analog signals or a serial bitstream) or link layer (relaying space data link frames between end points) interfaces. However, most legacy services were not designed to provide communication paths between other mission(s), communicate between SCaN elements, or communicate with external services through commercial networks and other agency's networks. Currently, SCaN is integrating the three operational network elements[5] to create shared SI services for future missions. These services provide SI data interfaces for customers. As a first step, under the SN Ground Segment Sustainment (SGSS) project, IP-over-CCSDS services will be supported for the first time as a SCaN service[6].

In later stages of SCaN integration, additional SI services will become available, per the SCaN Systems Requirement Document[7]. The SRD states "new technologies like space internetworking and optical communications will be infused as required to answer the needs of NASA's long-term exploration and science goals. Major objectives are to change the architecture to enable significant reductions in system acquisition and operations costs while improving the SCaN Network's flexibility and scalability to be more responsive to changes in budget and user needs." Space internetworking is defined as "extending Internet-like connectivity throughout the solar system". Requirements for space internetworking are defined at a high level in the SRD and simply state that SCaN shall "provide Space Internetworking services to mission users" and "interoperate with external space networks that are compliant with space internetworking standards."[8]

Goals and Objectives

Implementing networking for space missions is currently difficult because (1) it requires protocol support across mission-developed and SCaN elements, (2) there are not many reusable flight or ground software components available, (3) standards are still being developed in multiple areas, (4) commercial information technology products do not directly support the space mission needs, and (5) operations concepts with networking differ from legacy point-to-point communication services.

The SCaN Testbed mission success criteria included support for networking experiments that include SI protocols. The SCaN Testbed project includes a portfolio of several networking experiments, among its other communications, SDR, and navigation experiments.

Goals of the networking experiment portfolio include:

- Gain long-term operations experience with SI, in order to advance technology readiness using on-orbit flight systems and realistic ground systems
- Produce flexible implementations that can be used beyond SCaN Testbed
- Support network topologies that represent future mission complexity, not just single hops
- Mature the operations concepts and procedures for providing SI services in conjunction with multiple communications layers, radio configurations, front-ends, and other data processing systems
- Integrate networking with realistic on-board data interfaces (e.g. SpaceWire, 1553)
- Include native support for security protocols operating across multiple layers

II. SCaN Testbed System

The SCaN Testbed System architecture, shown in Figure 1, provides Network Experimenters with a space flight element, reconfigurable ground radios, reconfigurable ground elements and a Mission Operation control center. Detailed information on the SCaN Testbed can be found in Reference 9.

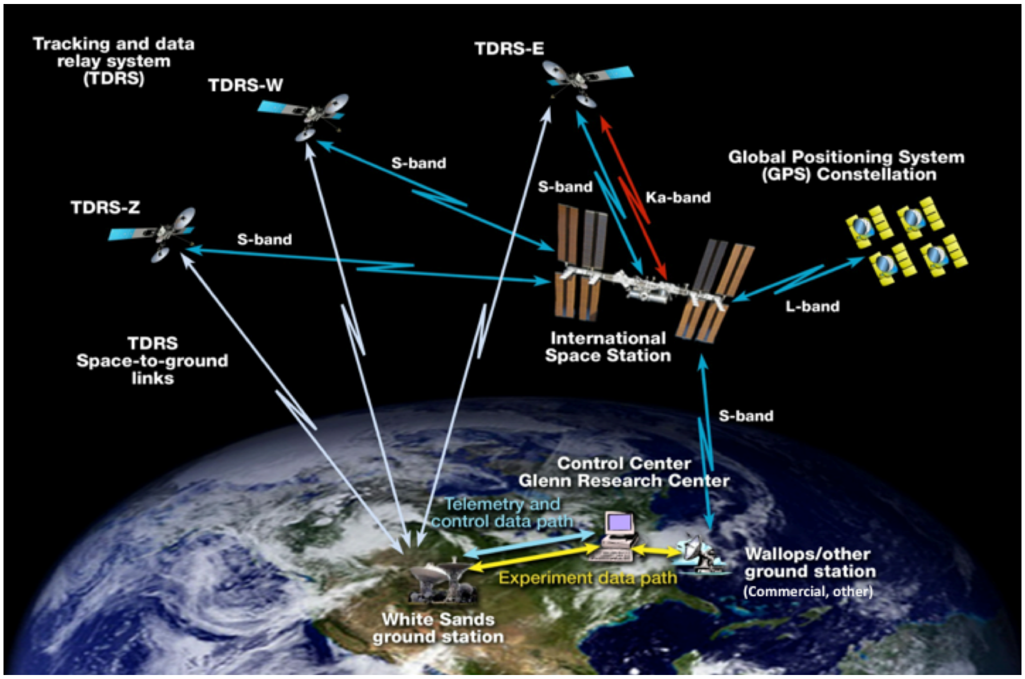
Tracking and data relay system (TDRS)
TDRS-W
TDRS-E
TDRS-Z
S-band
S-band
S-band
Ka-band
Global Positioning System (GPS) Constellation
L-band
International Space Station
TDRS Space-to-ground links
S-band
Control Center Glenn Research Center
Telemetry and control data path
Experiment data path
Wallops/other ground station (Commercial, other)
White Sands ground station

Figure 1. SCaN Testbed System Architecture

There are two paths to access the SCaN Testbed on the ISS. The first path is NASA's ISS payloads' command and control path through the Payload Operations Integration Center (POIC) located at NASA's Marshall Space Flight Center in Huntsville, Alabama. The second path is through the over-the-air experiment communications path from the experimenter equipment to the Software Defined Radio (SDR)s on-board the flight system through NASA TDRS Space Network or an ISS tracking S-band ground station.

Figure 2 shows an overview of the flight system, which consists of three SDRs, a flight computer, and an RF subsystem with five antennas. All three SDRs communicate with the flight computer via a SpaceWire interface for data exchanges and 1553 or SpaceWire for command and telemetry. The three zenith-facing antennas enable communications with NASA's Tracking and Data Relay Satellite System (TDRSS) via S- and Ka-band. An L-band receive only system provides GPS receive capabilities. The only nadir-facing antenna provides communication with an ISS tracking ground station operating in the S-band. The two S-band SDRs are connected to the three S-band antennas via a switch matrix, which allows for simultaneous connection of the two SDRs with any two of the three antennas.

The SCaN Testbed system has ideal characteristics for the development of network technology:

- Multiple reprogrammable nodes in space and ground, including SDR waveforms, a flight S950 computer to run VxWorks applications, ground x86_64 computers to run Linux applications and ground FPGA link processing
- The ability to communicate to multiple nodes based on time and position. This ability is needed to perform such tests as downloading a file that is distributed over multiple nodes. Neighbor awareness and opportunistic contacts can be demonstrated with this capability.
- Simultaneous communication paths over SN and NEN data links that assist in exploring intelligent networking concepts and "hardware in the loop" simulation.

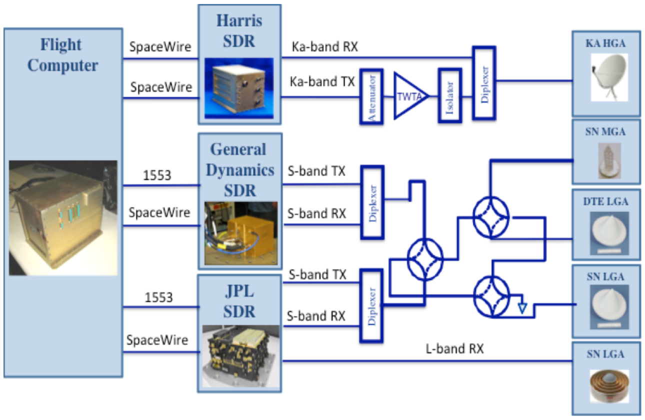

Figure 2 SCaN Testbed Flight System

- Reconfigurable space testbed that provides more system level diagnostics then typically available in an operational space node.
- Semi-operational environment. All software on the flight system is “Class C” which has been documented and tested more rigorously than typical laboratory software so code reuse is easier to justify with less auditing.

III. Network System Design for the SCaN Testbed

SCaN Testbed’s networking design goal is to create a robust, flexible, and expandable service to explore new technologies such as Intelligent Routing concepts described in Reference 1 in a realistic scenario. The testbed provides an integrated system with ground network nodes, flight network nodes, multiple communication paths, simultaneous link operation, CCSDS Space Link Extension (SLE) support, reprogrammable flight radios and reprogrammable nodes. The following section describes the evolving capability of the SCaN Testbed, from a basic platform for physical layer protocol development to a robust, networked system ideal for further studying advanced networking techniques

SCaN Testbed System at Launch

The July 2012 SCaN Testbed launch window constrained the schedule to only develop and test a minimal set of pre-launch capabilities that included:

- Ability to perform on-orbit updates to avionics flight software, experimenter software and Software Defined radios
- Testing of the RF and physical layer development platforms for Experimenters
- Point-to-point physical and bit layer service between the SDRs and the ground data interfaces at NASA’s Glenn Research Center (GRC) in Cleveland, Ohio through the Space Network
- Primary path Command and Telemetry services with SCaN Testbed’s Mission Operation Center located at GRC.

This prudent project goal, to develop and verify the necessary components of the system to assure future software and firmware upgrades, enabled ISS deployment success by maximizing limited labor resources towards prelaunch hardware and software testing of critical payload components. It also met a project goal of requiring that additional capability, such as Networking, occurred after post launch checkout of the baseline system.

The waveforms on the three Software Defined Radios, primarily implemented in the reprogrammable FPGAs on the radio, are compatible with the TDRS Space Network[10] with limited CCSDS Advanced Orbiting System (AOS) standards implemented[11]. This basic capability provided a physical layer data link between ground serial interfaces and the flight computer's Spacewire interface. Waveforms were upgraded after launch to support additional link layer processing and advanced high rate, bandwidth efficient and adaptive waveforms[12, 13].

CCSDS Encapsulation Packet Service

The ability to exchange variable length data packets between two service access points (SAP) within the SCaN Testbed system provides flexibility to mix commodity-networking protocols (e.g. IP protocol suite) with space networking protocols (e.g. CCSDS frameworks) using commodity hardware and mainstream operating systems such as Linux, VxWorks, and RTEMS. Fig. 3 highlights the five flight and five ground "space to ground" Service Access Points (SAP) available using the TDRS Space Network and direct to earth (DTE) RF paths for space to ground end points.

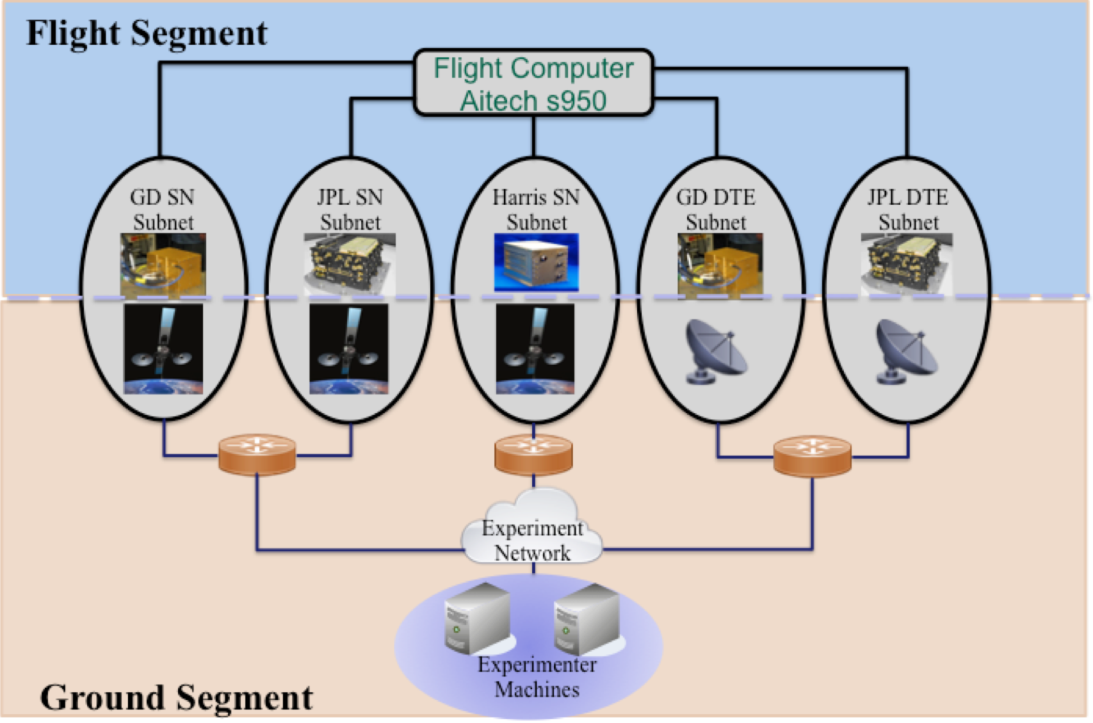

Figure 3. SCaN Testbed Point to Point Link Overview

CCSDS specifies a number of services to enable end-to-end data flow[14]. The Encapsulation service[15] specifies a communications service to be used by space missions to transfer protocol data units that are not directly transferred by the Space Data Link Protocols over a ground-to-space or space- to-space communications link. The data units transferred with this service are encapsulated in either Space Packets or Encapsulation Packets (ENCAP). Figure 4 shows the data flow from bottom to top for a prototol entity at the receiving end. The figure identifies data handling functions performed by the protocol entity and shows logical relationships among these functions. Depending on the services actually used for a real system, not all of the functions are required to be implemented in the protocol entity.

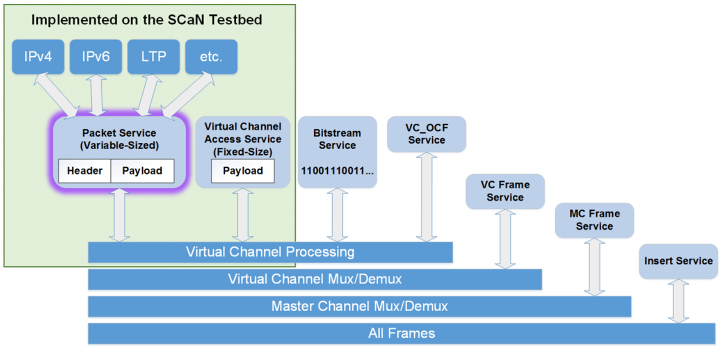

Figure 4. AOS and ENCAP implemented on SCaN Testbed

The services and applications shown in Figure 4 within the green box were implemented on the SCaN Testbed. We implemented the CCSDS Encapsulation Packet service on the three ground computers (Linux Centos7) and the Flight Computer (WindRiver VxWorks 6.3), supporting several ENCAP packets using the Payload Identifier Data (PIDs). The ENCAP Internet Protocol Extension (IPE) service provides the most flexibility for Network Experimenters. This service provides an Application Programmers Interface to exchange variable length ENCAP packets over the five possible experiment paths shown in Fig. 3.

IPv4 and IPv6 subnets can be established between the flight computer and the associate ground gateway[16]. For user applications like DTN over UDP or TCP, the ground SAP can act as an IPv4 router for other ground Network Experimenter computers and provide a flexible platform to explore dynamic link selection with COTS routing protocols and other SAPs such as a Space Link Extension (SLE). ENCAP LTP service allows DTN over LTP users to exchange Bundle Protocol (BP) bundles directly between the flight and ground SAPs. ENCAP PRIVATE service provides infusion of upcoming intelligent networking concepts for emerging data standards without impacting the use of legacy standards between the SAPs. Using AOS Transfer Frames with different Virtual Channels (VCs), different user services can be multiplexed over the same physical data link. For example, this allows a more complex mission to use two encapsulation services and a bit stream service over the same PHY link.

The SCaN Testbed system has access to the fixed length AOS Transfer Frame fields through various software application interfaces, including within the FPGA-based waveform on each SDR, on the FPGA-based system on the ground, and within the networking system processor. The AOS Transfer Frame fields include the Primary Header fields (VCIDs, Spacecraft ID and Virtual Channel Frame Counter) and Data Field (user data). Our flexible implementation could support multiple VCs, implement security extensions, frame error checking or even change AOS Transfer Frame length to investigate link efficiencies over more complex network topologies. Complex topologies can include using an SDR SAP and flight computer SAP on two different VCs over the same radio link. To our knowledge, the optional AOS Transfer Frame “Insert Zone” field is not currently used in missions; therefore, the SCaN Testbed radios do not explicitly support this field but this capability could easily be added.

Each fixed-length AOS Transfer Frame Data field contains a Packet Zone that transports our variable length Encapsulation Packets (ENCAP) as Multiplexing Protocol Data Units (M_PDUs). Each sent packet zone contains either an idle frame, a larger ENCAP packet segment that spans multiple AOS transfer frames, single ENCAP packet or multiple ENCAP packets. For an efficient ENCAP implementation for real missions and to explore protocol performance and issues, we fully utilize the available user data for transport by supporting multiple ENCAP packets into one packet zone. For expediency, some implementations support a single ENCAP packet per packet zone so the packet zone isn't fully utilized for small packets. For these implementations, larger packets requiring segmentation across two or more AOS Transfer Frame packet zones cannot be supported.

The ENCAP service relies on many elements to provide a diverse toolkit for applications (e.g. DTN), cognitive networking research, addressable network protocols research (e.g. IP protocol suite or DTN) and test tools (e.g. eping ENCAP RTT calculator, iperf and ping). These elements include SCaN Testbed launch SDR data links, commercial standards, operating systems (WindRiver VxWorks 6.3, Linux and RTEMS), serial links (RS644, RS422), IEEE standards (Gigabit Ethernet), IETF standards (IP, UDP, TCP, etc), CCSDS AOS standards, and COTS hardware.

Later experiments involving new network security protocols and in-space software implementations of cryptographic algorithms have also built upon the CCSDS Encapsulation and IP-over-CCSDS services. An implementation of the NSA’s IPsec Minimal Essential Interoperability Requirements (IPMEIR) has been uploaded, supporting network layer security for IP. Additionally, extensions to ION for DTN-layer security have also been uploaded, using Suite-B cryptography algorithms, and with the capability to run nested on top of IPMEIR, so that security at multiple scopes (hop-by-hop and end-to-end) can be supported across a DTN path.

Reuseable software components implementing the standards and integrating the waveforms and applications were developed for the SCaN Testbed. A portion of these reuseable components are shown in Fig. 5. These reuseable components include

- Interface code for porting ION to VxWorks
- SDR software waveforms with AOS Transfer Frame extraction
- Payload Avionics Software Experimenter API for payload services not provided by VxWorks such as exchanging SpaceWire packets with the SDRs or telemetry
- VxWorks and Linux ENCAP network drivers using OS configuration tools
- COTS and open source test tools like iperf and bping
- Ground and Flight link processing tasks to convert external PHY-TM-AOS into ENCAP
- Ground FPGA Experiment Front End Processor (EFEP) to process SDR specific PHY data over synchronous serial RS644 interfaces
- Linux kernel driver for Synchronous Serial to AOS TFs required for the EFEP

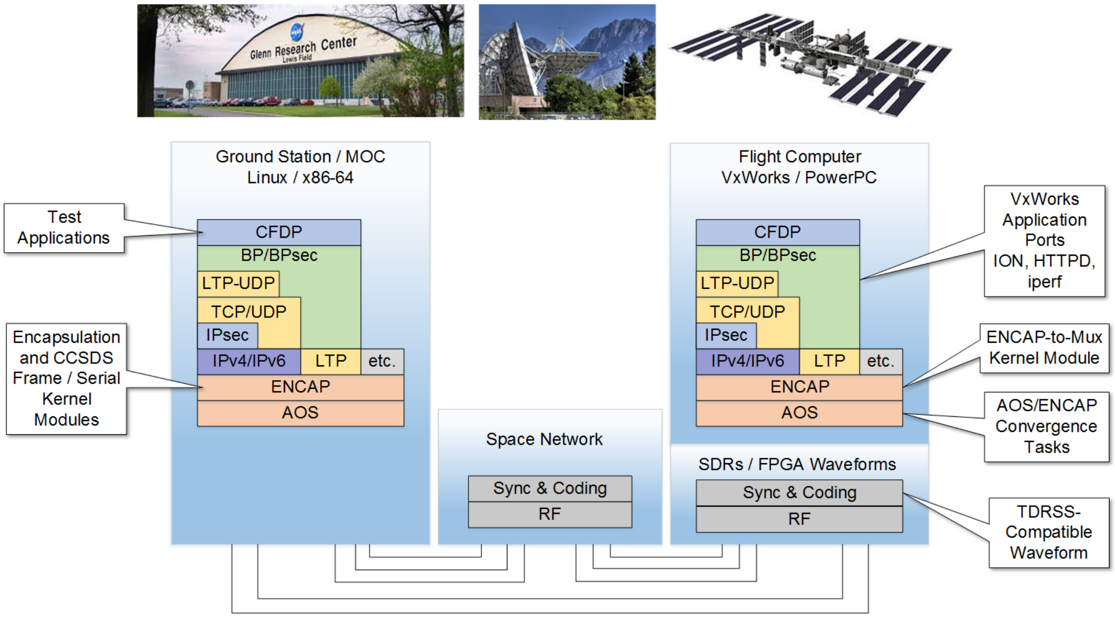

Figure 5. Reuseable Software Components

Delay Tolerant Networking / ION

The Interplanetary Overlay Network (ION) DTN software has been used by several other NASA projects in the past to support DTN technology development efforts[17-19]. Support for DTN on the SCaN Testbed began with an attempt to port the ION release version 3.1.3 to the VxWorks 6.3 operating system used by the SCaN Testbed flight computer. While DTN could also be supported on the three radio platforms, by implementing DTN on the flight computer we can utilize any of the available radio links to the Space Network or ground stations. ION contains many capabilities and features that can be removed (e.g. CFDP, AMS, etc) without impacting the core DTN capabilities. Initially, we only included a fraction of the ION code in our port, as we were testing the feasibility of making it fit within the RAM and CPU cycles available on SCaN Testbed flight computer. The flight computer is simultaneously running other software to manage the SDRs, Antenna Pointing System (APS), and other system functions.

Initially, the important features that we were able to include were:

- The core infrastructure of ION with the Interplanetary Communications Infrastructure (ICI), Software Data Recorder (SDR), and Zero Copy Objects (ZCO) libraries
- Support for the Bundle Protocol (BP) and use of the Interplanetary Network (IPN) endpoint identifier scheme for bundle forwarding
- Support for Licklider Transmission Protocol (LTP) as a convergence layer adapter
- Support for the basic UDP and TCP convergence layer adapters, over IP links

Additionally, there was interest from potential experimenters and collaborators in running LTP natively over the CCSDS space links on the SCaN Testbed; however, the stock ION code only supports running LTP over IP-based networks, by encapsulating LTP in UDP. The stock code works directly with the IP-over-CCSDS implementation on the SCaN Testbed, but when the DTN nodes are directly connected at the CCSDS data link layer, the extra UDP/IP encapsulation is inefficient and unnecessary. We added adapter shim code that allows the LTP implementation in ION to run directly over any of the CCSDS data links on the SCaN Testbed. This was very easy to do, since the ENCAP service on the Testbed includes support for additional protocols beyond simply IP, and has programming hooks for encapsulating and decapsulating other protocols from the ENCAP packets contained in AOS data link frames defined earlier in this paper.

After succeeding in these initial porting efforts, we continued to integrate more DTN features and ported updates from the ION 3.3.1 and 3.4 codebases, including CFDP, basic Network Management (NM) and Streamlined Bundle

Security Protocol (SBSP) support. We selected the features expected to be needed for follow-on DTN experimentation on SCaN Testbed and likely to be required for the spacecraft nodes in future DTN networks .

IV.   Software Instrumentation

All missions share a common requirement to exchange information with mission specific end points that share data transit services. In the past, these missions would manage the complete end-to-end communication system including static network configuration, private data storage, direct RF data link management, manual scheduling and labor-intensive activities that drives communication costs. Going forward, mission designers will focus on exchanging mission-specific information (when, what and where) between stakeholders using shared cognitive communication services to reduce communication costs and provide redundancy.

Sharing cognitive communication services, with missions having diverse communication Quality of Service needs, require integration of complex communication systems (near-earth, deep space, Cubesats, and ground), cross-layer coordination, machine directed (learning) for autonomous link management, agents with various level of intelligence, information-sharing and other cognitive elements. Each communication system must provide suitable software instrumentation and automation elements that include space platforms, near earth orbit relay satellites, dynamic data control plane, dynamic data management control plane, interplanetary relay satellites, RF link scheduling, earth-based commercial networking services, control plan management and other elements. These systems may have a local scope (e.g. data buses), directly reachable communication relay nodes (e.g. TDRS or other spacecraft) or indirect access to other communication systems.

Appropriate instrumentation is key to achieving system-wide knowledge and control for networked missions. SCaN Testbed pre-launch capabilities provided limited instrumentation (basic physical link statistics) to investigate launch SDR waveform RF performance and data link with the Space Network. Later, we gradually added more instrumentation such as round trip time, packet counters, OS counter, ENCAP RTT tools, applications for statistic recording and bit level statistics. The goal of the added statistics is to understand issues found during the integration of the various SCaN Testbed elements into an "end to end" SCaN Testbed system to enable the baseline Networking ENCAP service as described above.

Our current SCaN Testbed instrumentation points provide a rich platform to explore cognitive communication service research, development and testing. These areas are highlighted in Reference 2. Additionally, the flexibility of the SCaN Testbed will allow us to add more instrumentation in the SDRs through the application layer.

V.   Conclusions

Along with SDR research, the SCaN Testbed was designed for networking experiments, and support for networking was one of the mission success criteria. Since launch and installation onboard the ISS, several successful networking activities have been performed. This paper highlights the CCSDS Encapsulation service, and the IP and DTN protocol implementations that are built upon it. These technologies are key to the Solar System Internet (SSI) concept. Flight experience, implementations of the relevant standards, and protocol integration lessons have resulted from the SCaN Testbed networking research, and will be useful in future work building the SSI. The reusable flight and ground software with heritage from SCaN Testbed can be a part of future missions and SCaN infrastructure projects deploying the SSI. As our research advances to cognitive networking (Reference 2), we are continuing to use the SCaN Testbed as an on-orbit proving-ground for networking technology in space, and capitalizing on its unique capabilities as an on-orbit testbed for new technology.

VI.   References

[1] Reinhart, R.C., Lux, J.P, “Space-based Reconfigurable Software Defined Radio Testbed aboard International Space Station”, *SpaceOps 2014 conference*, Pasadena, CA, May 2014.

[2] Clark, G.J., Eddy, W. M., Barnes, J., Johnson, S. K., Brooks, D. E., “Architecture for Cognitive Networking within NASA’s Future Space Communications Infrastructure,” *34th AIAA International Communications Satellite Systems Conference*, Cleveland, Ohio, October, 2016 (to be published).

[3] “Report of the Interagency Operations Advisory Group Space Internetworking Strategy Group Recommendations on a Strategy for Space Internetworking,” November 15, 2008. URL: http://cwe.ccsds.org/ioag/Final%20Products/SISG%20Report%20v1.4%20FINAL.pdf

[4] “Solar System Internet (SSI) Architecture,” CCSDS 730.1-G-1, Green Book, July 2014.

[5] Tai, W., Wright, N., “NASA Integrated Space Communications Network”, *SpaceOps 2012*, Stockholm, Sweden, June 2012. URL: http://spaceops2012.org/proceedings/documents/id1275818-Paper-001.pdf

[6] Younes, B. A.; Schier, J. S, “Space Communications and Navigation (SCaN) Network Architecture Definition Document (ADD) Volume 1: Executive Summary, Rev. 2”, URL: http://www.nasa.gov/pdf/675092main_SCaN_ADD_Executive_Summary.pdf

[7] “Space Communications and Navigation System Requirements Document, Revision 3, September 2014.

[8] SCaN-STUDY-AC3-SI, Analysis Cycle 3: Space Internetworking Trade Study Report, June 2012.

[9] "SCaN TESTBED PROJECT SCaN Testbed Flight and Ground System Description" GRC-CONN-DOC-5022 RevA. URL: https://spaceflightsystems.grc.nasa.gov/sopo/scsmo/scan-testbed/potential-experimenter-information/

[10] Space Network User’s Guide, Revision 10. Greenbelt, Maryland, August 2012.

[11] “AOS Space Data Link Protocol,” CCSDS 732.0-B-3, Blue Book, September 2015.

[12] Downey, J. A., Downey, J. M., Reinhart, R. C., Evans, M. A., Mortensen, D. J., “Bandwidth-Efficient Communication through 225 MHz Ka-band Relay Satellite Channel”, *34th AIAA International Communications Satellite Systems Conference*, Cleveland, Ohio, October, 2016 (to be published).

[13] Mortensen, D., Adaptive Coding and Modulation Experiment with NASA's Space Communication and Navigation Testbed, *34th AIAA International Communications Satellite Systems Conference*, Cleveland, Ohio, October, 2016 (to be published).

[14] “TM Space Data Link Protocol,” CCSDS 132.0-B-2 Blue Book, September 2015.

[15] “Encapsulation Service,” CCSDS 133.1-B-2, Blue Book, October 2009.

[16] “IP over CCSDS Space Links,” CCSDS 702.1-B-1, Blue Book, September 2012.

[17] Scott, K. and S. Burleigh, "Bundle Protocol Specification", RFC 5050, November 2007.

[18] Ramadas, M., Burleigh, S., and Farrell, S., "Licklider Transmission Protocol Specification", IETF RFC 5326, 2008.

[19] ION, Interplanetary Overlay Network, URL: http://sourceforge.net/projects/ion-dtn/