SPACE COMMUNICATION DEMONSTRATION USING INTERNET TECHNOLOGY[1]

Dave Israel
NASA Goddard Space Flight Center

Ron Parise, Keith Hogie, Ed Criscuolo
Computer Sciences Corp

ABSTRACT

This paper presents work being done at NASA/GSFC by the Operating Missions as Nodes on the Internet (OMNI) project to demonstrate the application of Internet communication technologies to space communication. The goal is to provide global addressability and standard network protocols and applications for future space missions. It describes the communication architecture and operations concepts that will be deployed and tested on a Space Shuttle flight in July 2002. This is a NASA Hitchhiker mission called Communication and Navigation Demonstration On Shuttle (CANDOS).

The mission will be using a small programmable transceiver mounted in the Shuttle bay that can communicate through NASA's ground tracking stations as well as NASA's space relay satellite system. The transceiver includes a processor running the Linux operating system and a standard synchronous serial interface that supports the High-level Data Link Control (HDLC) framing protocol. One of the main goals will be to test the operation of the Mobile IP protocol (RFC 2002) for automatic routing of data as the Shuttle passes from one contact to another. Other protocols to be utilized onboard CANDOS include secure login (SSH), UDP-based reliable file transfer (MDP), and blind commanding using UDP.

The paper describes how each of these standard protocols available in the Linux operating system can be used to support communication with a space vehicle. It will discuss how each protocol is suited to support the range of special communication needs of space missions.

KEYWORDS

Internet Protocol (IP), High-level Data Link Control (HDLC), Operating Missions as Nodes on the Internet (OMNI), Space, Shuttle, Mobile IP, Low Power Transceiver (LPT), HitchHiker, Multicast Dissemination Protocol (MDP)

INTRODUCTION

The Operating Missions as Nodes on the Internet (OMNI) project[1] at NASA's Goddard Space Flight Center (GSFC) has been defining and demonstrating end-to-end communication systems for future space missions based on standard Internet technology. In late 2001 the OMNI project was presented with an opportunity to support Internet Protocol (IP) demonstrations with a HitchHiker payload flying in the Shuttle bay on flight STS-107 in July 2002. The mission is called Communication and Navigation Demonstration On Shuttle (CANDOS).

The primary goal of the mission is to test the newly developed Low Power Transceiver (LPT) on-orbit. The LPT is a small, multi-channel, programmable transceiver that was developed by ITT Industries under contract to NASA/GSFC. The OMNI project has been involved in another goal of the mission to investigate the use of standard Internet protocols for communicating with a space vehicle. This paper covers the Internet technologies incorporated in the mission and discusses the operational data transfer scenarios to be examined during the mission.

CANDOS SPACE SYSTEM

The LPT consists of circuit boards in a PC104 form factor and mounted in metal rings. The boards and their associated mounting rings stack together to form the shiny rectangular block in figure 1. For CANDOS, the stack of boards also contains a 686 processor running Linux 6.1, a FastComm ESCC synchronous serial interface card, and a DC-to-DC power supply to convert the Shuttle 28 VDC into 5 and 12 VDC.

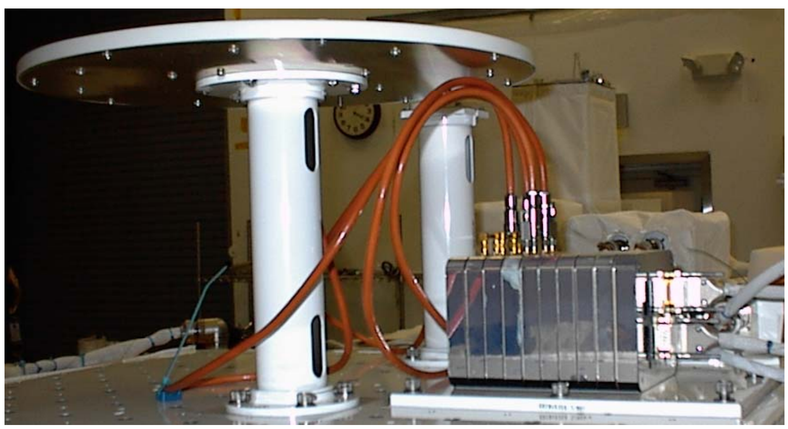

Figure 1 – LPT Transceiver and Processor Slices under High-gain Antenna

The Linux operating system on the CANDOS processor provides standard IP services built into the operating system to provide the following:

- Internet Protocol (IP)[2] addressing support
- User Datagram Protocol (UDP)[3] sockets application programming interface (API)
- Transport Control Protocol (TCP)[4] sockets as another API

The following applications either come standard with the Linux operating system or can be easily downloaded and installed:

- Multicast Dissemination Protocol (MDP)[5] reliable file transfer operating over UDP
- Network Time Protocol (NTP)[6] operating over UDP
- File Transfer Protocol (FTP)[7] operating over TCP
- Telnet[8] remote access over TCP
- Secure Shell (SSH)[9] encrypted remote login and Secure Copy (SCP) file transfer operating over TCP

A key component in the system is a synchronous serial interface card with its RS-422 clock and data lines connected to the data input and output ports on the LPT transceivers. The OMNI project developed a network device driver that allows this serial interface card to be accessed as a network device with IP addresses. When the Linux system wants to transmit a packet, the network driver receives the packet from the Linux IP stack, adds the appropriate Frame Relay[10]/ HDLC[11] header, queues the packet for output, initiates the data transfer, and processes interrupts to complete the transfer.

The combination of a standard Linux operating system with IP and a standard synchronous serial interface with a Frame Relay network driver is the key to a simple space communication system. Using standard IP packet and HDLC frame formats on the spacecraft means the resulting bitstream is easy to interface directly into standard serial ports on commercial routers from vendors such as Cisco, Nortel, and 3com. These ground station routers then use standard Ethernet interfaces to pass data directly on and off the NASA operational communication IP backbone.

One other special piece of software added to the onboard system is a Mobile IP[12] daemon to support the spacecraft end of the protocol. This is a software process that runs continually on the processor and listens on a raw socket for Internet Control Message Protocol (ICMP) packets with Mobile IP advertisements coming up from the ground. Once the Mobile IP daemon responds to the advertisement, the rest of the IP routing work is handled by the ground components. This process is described in more detail at the end of this paper.

•

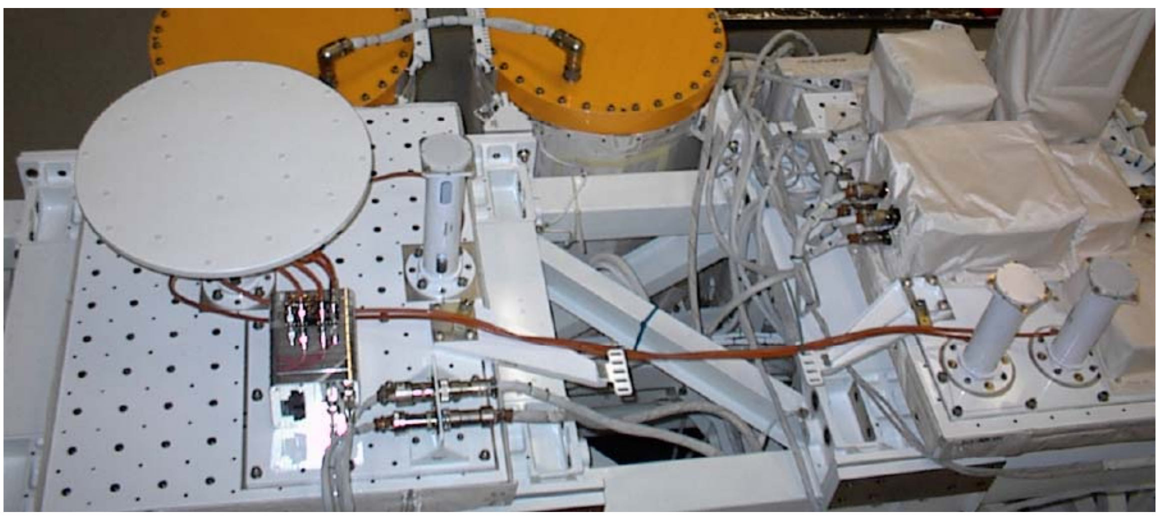

Figure 2 – LPT and Antennas on HitchHiker Truss

The LPT has a high-gain, fixed position, circular 18” antenna shown on the left in figure 2. The three other small disks on columns are a low-gain transmit antenna, a low-gain receive antenna and one GPS antenna.

CANDOS GROUND SYSTEMS

The ground systems for the CANDOS mission will use existing antennas, transmitters, and receivers at NASA’s Ground Network (GN) and Space Network (SN) stations. However, the stations will be upgraded by connecting standard routers to the transmitter/receiver to provide direct connectivity to NASA’s operational IP backbone. Modifications will be made at both

Tracking and Data Relay Satellite System (TDRSS) stations in White Sands as well as at stations at Wallops (WLP) and Merritt Island Launch Area (MILA).

The main component of the upgrade is the addition of standard Cisco routers between the clock and data bitstream interfaces on the transmitters and receivers and NASA's operational IP backbone. Standard router options will be configured to process the HDLC frames and the Frame Relay headers on the router's serial interface. The serial interfaces will make the link to the spacecraft look like any other wide area network (WAN) interface on NASA's backbone.

However, the actual bitstream interfaces at NASA stations vary from station to station and are not always fully compatible with a standard router interface. A conversion device developed at NASA/GSFC will be inserted between the router and the transmitter/receiver interfaces to perform functions such as signal level shifting, convolutional encoding/decoding, and data scrambling/de-scrambling (pseudorandom noise coding/decoding). This process is identical to current satellite modems that are widely used to interface standard routers with commercial communication relay satellites, although some of the coding algorithms are different. A key issue is to segregate the satellite specific coding and forward error correction (FEC) techniques in an external conversion box that is separate from the router. This approach simplifies future growth and change by allowing new coding and FEC solutions to be transparently inserted in the bitstream interface between the RF system and the router with no impact on the router.

Once the HDLC bitstream from the spacecraft enters a router serial interface, the interface locates the frames and extracts the encapsulated IP packets. These packets are them passed out a router Ethernet interface onto NASA's operational IP backbone network. This network is already in place and connects NASA's ground stations.

The control center for the CANDOS mission consists of a collection of workstation and laptop computers with both PC and Macintosh processors. The machines will used the Linux, Window, and MacOS operating systems which all support standard IP software.

END-TO-END DATA FLOW

Significant benefits of using IP technology for future space communications are the extensive end-to-end network architecture available and the wide range of hardware and software components readily available. Internet technologies include an extensive suite of protocols that have all been crafted to work together to provide a very powerful and flexible communication system. Today's space communication systems often use custom protocols on the spacecraft and to the ground and then use Internet technologies to move data around on the ground. A major change on CANDOS is to start all data flows with standard Internet protocols onboard. It will use standard TCP or UDP sockets on the spacecraft and continue the standard Internet formats over the space-to-ground link as well as on the ground. This results in a simple and flexible end-to-end data system.

Figure 3 shows the key technologies used for end-to-end data flows. The LPT processor in the upper left of the diagram is running standard Red Hat Linux version 6.1 with applications such as MDP, NTP, Telnet, FTP, SSH, and SCP. These applications all use UDP and TCP sockets for their application programming interface (API) to send and receive data. They will be interacting with similar applications on the computer systems on the ground.

The TCP and UDP sockets hide lower layer communication details from the applications so the applications don't need to know whether they are running over an Ethernet LAN or a serial space link. The sockets pass packets of data down to the IP layer which adds source and destination addresses.

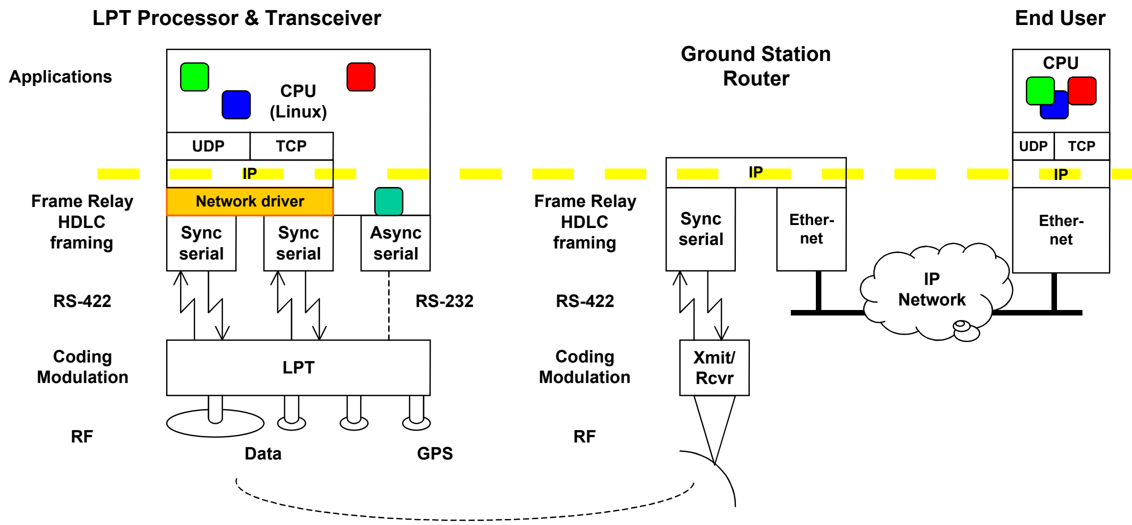

Figure 3 – End-to-End Data Flow

The next step is to send the packets out a synchronous serial interface where they are delivered to data ports of the LPT using clock and data lines with RS-422 signal levels. Getting from the IP layer to the serial interface requires a Linux network device driver that can support the Commtech FastComm PC104 serial interface card used in CANDOS. The card came with a character device driver that supported opening a serial port and sending and receiving individual characters. In order to provide standard IP support, the OMNI project wrote a network device driver that supports a standard Linux interface to accept IP packets from the IP stack. The driver also adds the proper Frame Relay header bytes and sends the resulting frame out the serial card where it gets wrapped in HDLC framing.

The clock and data lines from the serial interface deliver a continual stream of bits to and from the transmit and receive interfaces on the LPT. The LPT performs its coding and modulation functions by operating on this bitstream without any knowledge of the format of the frames being transmitted.

Another major difference between CANDOS and conventional spacecraft is that it uses exactly the same framing and upper layer protocols on both its transmit and receive links. The receive path just reverses the processing of the transmit path. This concept of identical framing on both directions of the communication link is important for future missions which are looking at implementing cross-links between spacecraft.

Today's spacecraft communicate with ground stations and often refer to their data links as the forward link, uplink, or command link, to the spacecraft, and return link, downlink, or telemetry link back to the ground. With space and ground end points this concept works. However, on a spacecraft cross-link, the concept of one format on the forward link and a different format on the return link does not work. This results from the fact that one spacecraft's downlink, or transmit interface, is trying to send data to the forward link, or receive interface on the other spacecraft.

Also, the concept of "commands" to the spacecraft and "telemetry" to the ground does not properly describe spacecraft communication with many Internet protocols. For example, a TCP session transferring data to the ground would have TCP ACKs going back up to the spacecraft. However, these ACKs are not really a "command".

CANDOS DATA FLOWS

The goal of the Internet protocol tests on CANDOS is to analyze the performance of numerous standard Internet applications and protocols to operate the LPT. Operations will include monitoring and controlling the processor and radio, as well as retrieving data from GPS experiments performed with some of the LPT receivers. The protocols and applications to be analyzed include:

- Mobile IP – automated route management to the spacecraft
- Multicast Dissemination Protocol (MDP) – UDP based, reliable file transfers from space
- Network Time Protocol (NTP) – UDP based spacecraft clock synchronization
- UDP packets - real-time spacecraft telemetry
- UDP packets – spacecraft blind commanding
- Secure Shell (SSH) and Telnet – TCP based remote login to the spacecraft
- Secure Copy (SCP) and FTP – TCP based reliable file transfer to and from the spacecraft

An important aspect of these protocol investigations is that these will not be highly controlled tests run in a precise laboratory environment. The main goal for all tests will be to record time-stamped performance data on the protocols. This data will then be correlated with Shuttle position and attitude data and RF link measurements to characterize the operational performance of the various protocols. The following subsections describe each of the protocols to be examined.

Mobile IP

In today's spacecraft communication, control centers normally send commands to the ground station/antenna the spacecraft is passing over and the command is uplinked to the spacecraft. The major issue is that the control center must know where to send the commands and address them accordingly. However, as large constellations of spacecraft are deployed, advance planning and scheduling of contacts becomes more complex and expensive and an automated solution for delivering commands to the proper ground station is desirable.

Figure 4 shows an example where a spacecraft has an Internet address (e.g. 100.10.10.18), that is part of a ground based subnet (e.g. 100.10.10.x). Any IP datagrams addressed to the spacecraft address from anywhere on the Internet will be routed using standard Internet routing and will be delivered to the Home Agent ground subnet. However, the Home Agent router needs to know how to forward the packet to the proper ground station to get it to its final destination. This is exactly the same problem encountered by other mobile devices on the Internet such as laptops, PDAs, and eventually automobiles.

The Internet Engineering Task Force (IETF) has developed standards called Mobile IP (RFC 3220) to deal with this problem. These protocols use an initial protocol exchange to allow the mobile device or spacecraft to determine if it is in direct contact with its home subnet and associated home agent software or a foreign subnet and its foreign agent software. If the mobile device is in contact with a foreign subnet, the foreign agent establishes an IP encapsulation tunnel from the home agent to the foreign agent. Then when the control center sends a datagram to the spacecraft address, the packet goes to the home router where the home agent notices that there is a tunnel to the spacecraft via a foreign router. The packet is then sent through the tunnel to the foreign agent which passes it out its serial interface and up to the spacecraft.

This sort of Mobile IP scenario is primarily an issue for sending data to the spacecraft. When any packets are sent from the spacecraft to any ground station, the ground station simply uses the destination address to forward the packets using standard Internet routing rules. One possible exception is if the foreign ground station has additional routing rules, for security reasons, which prevent it from forwarding packets whose source address is not within the foreign subnet. Then the tunneling features of Mobile IP would be needed to encapsulate the spacecraft packets for delivery to their home subnet.

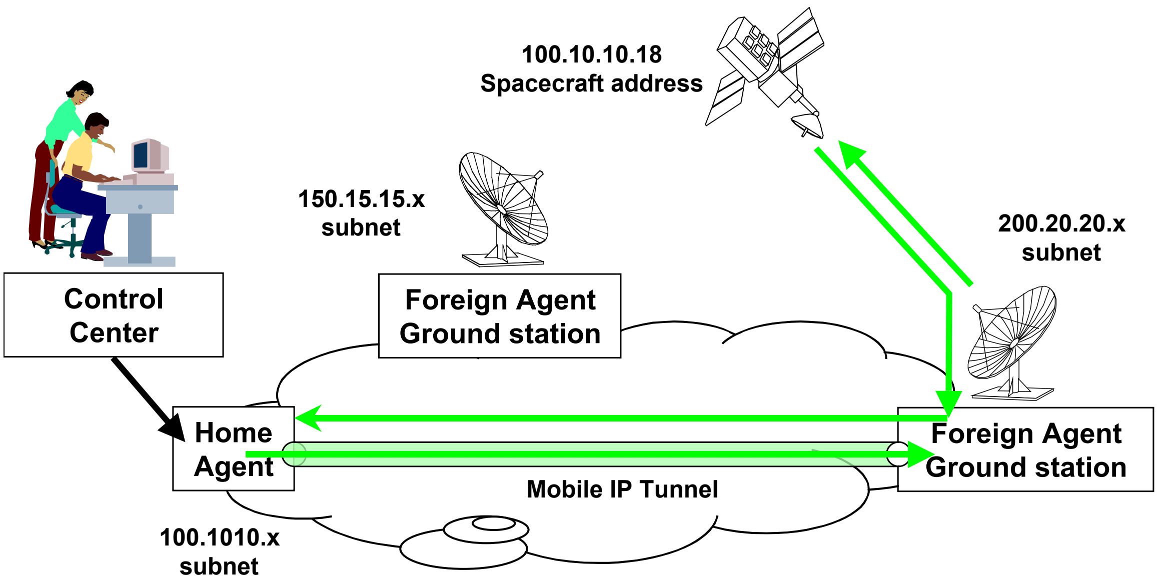

Figure 4 – Mobile IP Routing

These cases have only addressed a spacecraft or mobile host with a single IP address. If the mobile device or spacecraft has a LAN with multiple IP addresses then the problem gets more complex. One solution is for each node on the spacecraft with an IP address to perform Mobile IP registration and set up tunnels for each. However, this does not scale well and causes additional traffic for all of the registrations and additional software for each node. The solution currently being worked on in the IETF is called Mobile Routing. It involves a router that performs all of the Mobile IP operations and none of the nodes on the LAN even realize they are mobile. They simply operate just like they do on a fixed LAN. The research and development in this area is being driven by concepts in which all future automobiles will have onboard LANs with Internet addresses and full mobile Internet connectivity. The size of the automobile market, a potential market for mobile routers, is huge and the commercial research and development investments are substantial. A version of the Mobile Routing protocol is currently available in Cisco routers in version 12.2.4 of the IOS software.

During the CANDOS mission, Mobile IP operation will be carefully monitored to analyze its performance during real space mission conditions. All Mobile IP packets will be captured and router Mobile IP authentication and tunnel management activities will be logged. The time stamps on the data will be compared to the Shuttle position and antenna orientation to understand overall protocol performance especially around the beginning and end of the communication contacts.

MDP reliable file transfers

Future spacecraft system designers are very interested in moving to storage systems that use a standard file system concept and pass data files around using reliable file transfer mechanisms. This approach simplifies spacecraft data storage designs and eliminates most of the data reassembly functions performed in today's level-zero processing (LZP) systems.

Many standard Internet data transfers such as web surfing, email, and file downloads use reliable file transfer mechanisms operating over the TCP transport protocol. However, using a TCP transport layer in space has problems due to TCP's use of continual acknowledgements (ACKs) flowing in the opposite direction of data flow. In searching for ways around this problem the OMNI project found an attractive solution from work by the Naval Research Laboratory (NRL) in reliable multicast file delivery. The OMNI project is also working with NRL to incorporate some additions to the MDP application to provide more robust and automated operation in space environments [13].

In multicast data delivery, a packet is sent out to potentially hundreds or thousands of receivers. If each of those receivers sent an ACK back for every few packets, the original sender would see a flood of thousands of ACKs coming back for every few packets out. To solve this problem, NRL developed the Multicast Dissemination Protocol (MDP) which uses UDP to allow primarily one-way data delivery with only negative acknowledgements (NACKs) as needed.

On CANDOS, GPS experiments will be performed with the LPT's GPS receivers. These experiments will generate files of data that need to be delivered to the ground. MDP will be used to transfer these data files to the ground. The MDP throttle data rate will be set to maximize bandwidth on the current RF link. File transfers will be monitored to analyze the actual performance of MDP and how well it utilizes the available link bandwidth.

MDP file transfers over primarily one-way, SN demand access link

In NASA's TDRSS system there are often one-way, return-only data paths available. This resource is more available and more easily scheduled than two-way links. MDP is built to primarily send data in one direction with minimal ACKs or NACKs returned. During the CANDOS mission, this capability will be used to deliver files to the ground over a one-way link. When a two-way contact is available any NACKs will flow back up, lost data will be retransmitted, and the reliable file transfer will complete. This usage of a one-way link with intermittent two-way contact is exactly the type of data delivery currently being considered for the Global Precipitation Mission (GPM).

Network Time Protocol (NTP) - time sync

A common operational issue for spacecraft is to keep the onboard clock synchronized to UTC so data can be accurately time stamped on the spacecraft. For the CANDOS mission initial clock management will be performed manually but in later passes an NTP application onboard will be tested with ground NTP servers to automatically set the LPT clock and verify its stability. Initial time setting will be done using an existing time server on NASA's backbone. All of the CANDOS foreign agent routers will be synchronized to the same NTP server and may be used as NTP servers for the spacecraft. This will allow comparing NTP operation across short communication paths at each station versus longer paths by traversing the ground network to get to central NTP servers. Results are expected to be in the 1's to 10's of milliseconds precision.

UDP real-time telemetry and housekeeping data

A normal operation for all spacecraft is to send telemetry and housekeeping data in TDM or CCSDS frames. On the CANDOS mission, UDP packets in HDLC frames will be used instead of CCSDS frames to accomplish the identical function. The LPT processor will normally be sending out a low-rate stream of UDP packets containing status information such as receiver lock, signal levels, temperatures, etc. By sending this data in UDP packets, it can flow to the ground during both one-way and two-way communication contacts. Each packet is completely self contained with one set of status data samples and full IP addressing to allow the ground network to deliver it to its proper destination. If a UDP packet is lost due to bit errors, it will simply disappear. However, since each packet is standalone and there is no session state as there is with TCP, there is no retransmission and no corresponding delay of following data. The next packet propagates on down with minimal network delay.

UDP blind commanding

The LPT transceiver is configured with its receiver normally ON but its transmitter OFF. Typically, spacecraft have either an onboard schedule that turns on their transmitter at a specified time or they receive a command from the ground to turn on their transmitter. For the CANDOS mission, the transmitter can be turned on by manually configuring an IP tunnel to route packets to the proper ground station and sending a UDP packet addressed to the LPT on its blind commanding port. This process is identical to the "blind commanding" scenarios used with today's spacecraft when they do not have a two-way link available.

SSH/Telnet remote login

All of the previous applications described herein use UDP to operate over intermittent and noisy space links. But tests will also be performed with applications that operate over TCP. Standard SSH and Telnet applications will be used to log into the LPT processor from the CANDOS control center to monitor its operation and make software changes. The packets from these interactive sessions will be captured and analyzed with programs like tcptrace to analyze TCP performance over the space link.

SCP/FTP file transfers

Standard SCP and FTP applications will also be used for reliable file transfers when a two-way link is available. These packets will also be captured and analyzed with tcptrace to compare performance with similar MDP file transfers.

CONCLUSIONS

This paper is being prepared before the actual mission so measured results are not currently available. The tests performed and results collected during the mission will focus primarily on examining protocol operation in a space environment. Actual performance of the protocols is expected to vary widely based on each contact's RF link error characteristics. Since the LPT antennas are not steerable, they are always pointed directly up out of the Shuttle bay. Much of the time they will not be aimed directly at TDRSS or a ground station antenna depending on the Shuttle orientation. This should provide a wide range of RF link performance that will provide an excellent variety of test conditions.

The use of HDLC framing in space is expected to work very well based on current experience from tests performed using Internet protocols through TDRSS, on the UoSAT-12 spacecraft, and in simulations performed by ITT Industries for NASA/GSFC. The CANDOS mission will allow collecting more information on HDLC performance with both SN and GN stations across a full range of orbital and antenna pointing scenarios.

CANDOS will also provide an excellent opportunity to collect performance information on the operation of the Mobile IP protocol in these same orbital scenarios. An area of special interest will be to monitor Mobile IP performance at the beginning and end of contacts to see how it performs when the spacecraft is close to the horizon. Current missions normally wait until the spacecraft is at least 5 degrees above the horizon before attempting communication. Since Mobile IP operates over UDP and only requires a few packets it should be able to operate closer to the horizon.

Tests performed using MDP over a one-way, return link configuration will provide useful information for future missions like GPM. A primarily goal of testing applications like MDP and NTP is to identify ways to allow future spacecraft to operate more automatically with less commanding required from ground personnel.

Using TCP based applications such as SSH/SCP and Telnet/FTP will provide insight into their performance over noisy and intermittent links. During the middle of a contact they should operate fine. The more interesting tests will be to determine if it is reasonable to use them near the beginning and end of contacts.

ACKNOWLEDGEMENTS

The end-to-end Internet protocol data flow design and software configuration described in this paper was carried out by personnel from Computer Sciences Corporation working for NASA's Goddard Space Flight Center under contract GS-35F-4381G S-36130-G, with additional efforts and support contributed by individuals from ITT Industries and other GSFC organizations. The work was performed for the LPT CANDOS project through funding provided by NASA Headquarters, Code M. The authors would also like to thank GSFC code 450 for their pioneering IP work on the South Pole TDRSS Relay (SPTR) project[14].

REFERENCES

1 http://ipinspace.gsfc.nasa.gov/

2 "Internet Protocol, DARPA Internet Program Protocol Specification", Internet Engineering Task Force RFC-791, September 1981

3 "User Datagram Protocol", Internet Engineering Task Force RFC-768, August 1980

4 "Transmission Control Protocol", Internet Engineering Task Force RFC-793, September 1981

5 J Macker, R B Adamson, "The Multicast Dissemination Protocol (MDP) Toolkit", IEEE, 1999, http://mdp.pf.itd.nrl.navy.mil/MdpToolkitOverview.ps.gz

6 "Network Time Protocol (Version 3) Specification, Implementation and Analysis", Internet Engineering Task Force RFC-1305, March 1992

7 "File Transfer Protocol", Internet Engineering Task Force RFC-959, October 1985

8 "Telnet Protocol Specification", Internet Engineering Task Force RFC-854, May 1983

9 "SSH Protocol Architecture", Internet Engineering Task Force – draft-ietf-secsh-architecture-12.txt, January 2002

10 "Multiprotocol Interconnect over Frame Relay", Internet Engineering Task Force RFC-2427, September 1998

11 "High-level Data Link Control (HDLC) - Frame Structure", ISO-3309

12 "IP Mobility Support for IPv4", Internet Engineering Task Force RFC-3220, January 2002

13 Jim Rash, Ed Crisuolo, Keith Hogie, Ron Parise, "MDP: reliable file transfer for space missions", Earth Science Technology Conference 2002, Pasadena CA, June 2002

14 Israel, David j., "The South Pole TDRSS Relay (SPTR)." Astrophysics From Antarctica, ASP Conference Series Volume 141, 1998: 319+