Delay/Disruption-Tolerant Network Testing Using a LEO Satellite

Will Ivancic
National Aeronautics and Space Administration (NASA)/Glenn Research Center (GRC), Cleveland, Ohio, United States, wivancic@grc.nasa.gov
Wesley M. Eddy
Verizon Federal Network Systems / NASA GRC, Cleveland, Ohio, United States, weddy@grc.nasa.gov
Lloyd Wood
Cisco Systems, Feltham, United Kingdom, lwood@cisco.com
Dave Stewart
Verizon Federal Network Systems / NASA GRC, Cleveland, Ohio, United States, dstewart@grc.nasa.gov
Chris Jackson
Surrey Satellite Technology Ltd (SSTL), Guildford, Surrey, United Kingdom, c.jackson@sstl.co.uk
James Northam
SSTL, Guildford, Surrey, United Kingdom, j.northam@sstl.co.uk
Alex da Silva Curiel
SSTL, Guildford, Surrey, United Kingdom, a.da-silva-curiel@sstl.co.uk

***Abstract*- Delay/Disruption Tolerant Networking (DTN) "bundles" have been proposed for deep-space communication in the "Interplanetary Internet." This paper describes the first DTN bundle protocol testing from space, using the United Kingdom Disaster Monitoring Constellation (UK-DMC) satellite in Low Earth Orbit (LEO). The mismatch problems between the different conditions of the private dedicated space-to-ground link and the shared, congested, ground-to-ground links are discussed. DTN, with its ability to transfer files on a hop-by-hop basis across different subnets, is presented as a technology that can be used to alleviate this problem. We describe our operational testing, as well as test configurations, goals and results, and lessons learned.**

I. INTRODUCTION

Delay/Disruption Tolerant Networking (DTN) has been defined as an end-to-end store-and-forward architecture capable of providing communications in highly-stressed network environments. To provide the store-and-forward service, a "bundle" protocol (BP) sits at the application layer of some number of constituent internets, forming a store-and-forward overlay network [1]. Key capabilities of the BP include:

- Custody-based retransmission – the ability to take responsibility for a bundle reaching its final destination
- Ability to cope with intermittent connectivity.
- Ability to cope with long propagation delays.
- Ability to take advantage of scheduled, predicted, and opportunistic connectivity (in addition to continuous connectivity).
- Late binding of overlay network endpoint identifiers to constituent internet addresses [2].

The DTN protocol suite is intended to consist of a group of well-defined protocols that, when combined, enable a well-understood method of performing store and forward communications. DTN can be thought of as operating across varying conditions across several different axes, depending on the design of the subnet being traversed:

- low or high propagation delay
- dedicated or shared, congested links
- links with intermittent disruption and outages or scheduled planned links.

In a low-propagation-delay environment, such as may occur in near-planetary or terrestrial environments, DTN bundle agents can utilize chatty underlying Internet protocols, such as TCP, that negotiate connectivity and handshake connections in real-time. In high-propagation-delay environments such as deep space, DTN bundle agents must use other methods, such as some form of scheduling, to set up connectivity between the two bundle agents, and can use less chatty transfer protocols over IP.

Low Earth Orbit (LEO) is a low-propagation-delay environment of less than ten milliseconds delay to ground, with long periods of disconnection between passes over ground stations. For the UK-DMC satellite, contact times consist of 5 to 14 minutes per pass with one or two available ground station contact times per 100 minute orbit – assuming multiple available ground stations. The ground stations are connected across the terrestrial Internet, which has different operating conditions (congestion-sensitive, always on) from the private links between satellite and ground station (intermittent but scheduled, and dedicated to downloading.)

II. THE RATE MISMATCH PROBLEM

Figure 1 illustrates a LEO satellite ground network with a DTN Bundle Agent sink located at a remote location. The final remote location for the downloaded imagery could be a satellite control station and office or a laptop 'in the field' with wireless connectivity – it really doesn't matter. In this example, an image is to be transferred from the DTN source,

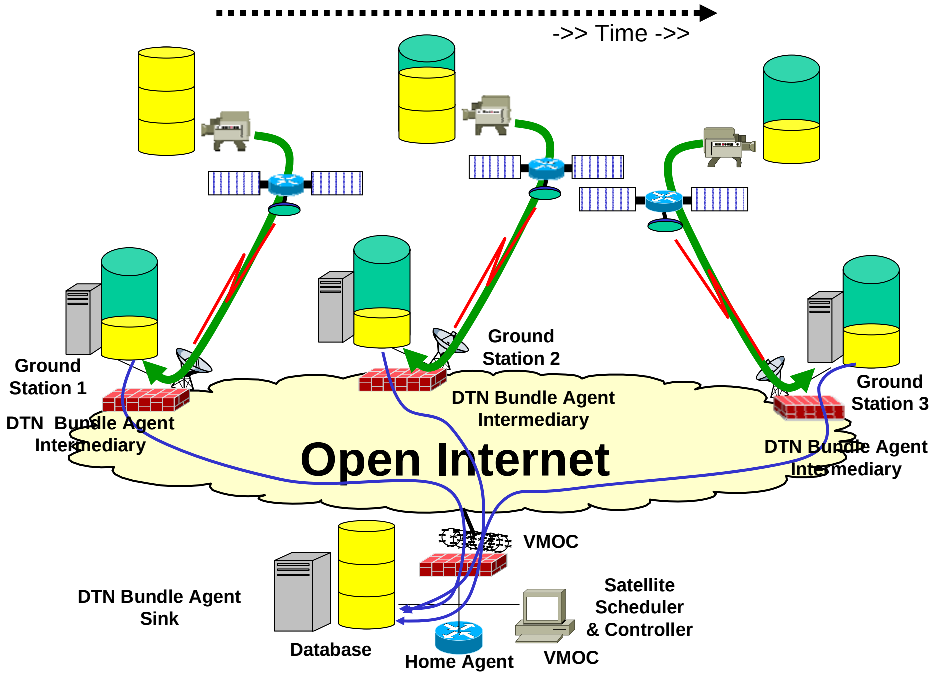

Figure 1 - DTN Ground Network

the LEO satellite, to the DTN sink. In this example, the hypothetical image file is too large to be transferred during one pass over a single ground station. Rather, three passes are required to transfer the complete file to ground. These passes could all be via the same ground station or could utilize three different ground stations. The minimum time a complete image file could be transferred using a single ground station is a little over 300 minutes, assuming one pass per 100-minute orbit. However, using three different ground stations, the entire image could be downloaded in a fraction of an orbit, by downloading fragments of the image to each ground station and reassembling the complete image file on the ground.

If some type of rate-based file transfer is used between the sink and source, problems will arise if ground link capacity does not match or exceed the rate of the space-to-ground link; the transfer becomes limited by any bottleneck in the path. In order to increase the download rates across each link, the transfer can be split into multiple separate hops, where the download is stored and forwarded locally across each hop – note, this is the situation whether using a single ground station or multiple ground stations.

The requirement is to get the image off the spacecraft as efficiently as possible, as spacecraft pass time is the major constraint, and then transfer separately across the different environment of the terrestrial Internet afterwards. The DTN BP is one example of a protocol that provides such functionality, and can thus compensate for rate mismatches between the private space-to-ground link and the shared path between ground station and remote destination for the image.

III. UK-DMC CHARACTERISTICS

The UK-DMC satellite is one of five similar imaging satellites currently launched into low Earth orbit in similar sun-synchronous planes. It was launched in September 2003, with a design lifetime of five years. This imaging constellation continues to grow, with at least four more satellites to be added in the next two years to maintain a continuous on-orbit imaging capability. While these satellites are government-owned, the UK-DMC satellite is also used to provide imagery for commercial resale when not otherwise tasked in imaging campaigns or supporting disaster relief. Anyone may request an image and pay the associated costs [3].

The UK-DMC is not solely an experimental satellite. However, SSTL has also run experiments onboard the UK-DMC such as investigating GPS reflectometry [4,5] and networking experiments have taken advantage of an onboard Internet router [6,7]. SSTL continues to permit NASA to utilize the UK-DMC satellite for experimentation with new forms of networking.

The UK-DMC satellite's onboard payloads include:

- The Cisco router in Low Earth Orbit (CLEO). CLEO has been used for network testing and is its own experiment to simply show that a commercial-off-the-shelf router could survive and function in orbit. CLEO is not used for DTN bundle testing.
- Three Solid-State Data Recorders (SSDRs)
    - one SSDR based around a StrongARM Processor, supporting the onboard GPS

    reflectometry experiment.
  - two SSDRs with Motorola MPC8260 PowerPC processors, supporting the imaging cameras. One of these SSDRs is used for DTN testing. These run the RTEMS operating system, which supports the POSIX API and BSD sockets. These have a constrained operating system firmware size limit of 1 MByte, and storage capacities of 1 GByte and 512MByte RAM respectively.
- There is an uplink of 9600 bits per second, and downlink of 8.134 Mbps – this is highly asymmetric. Both links use the proven IPv4/Frame Relay/HDLC encapsulation developed for space by Keith Hogie [8]. IPv6 has been tested over these links, using the onboard CLEO router [9,10]. The IP-based transport protocol used for downloading images is SSTL's original implementation of *Saratoga*, retroactively called version 0, running over UDP. *Saratoga* version 0 is the existing operational SSTL file transport protocol, originally developed to replace and improve transfer performance rates over an implementation of CCSDS CFDP that was previously used by SSTL. *Saratoga* version 1 is an improved specification, with enhancements to *Saratoga* version 0, which has now been documented publicly as a contribution to the IETF [11].

IV. EXPERIMENTAL BUNDLING IMPLEMENTATION

a. Onboard the UK-DMC satellite

Figure 2 illustrates how DTN bundling is implemented onboard the UK-DMC and in the ground infrastructure. *Saratoga* acts as a bundle transport 'convergence' layer on the space-ground link. Only the bundle forwarding portion of DTN was implemented onboard as a simple networking "shim" since available code space is constrained, and a goal was to have the onboard DTN implementation be transparent to normal UK-DMC operations, living side-by-side with the existing operational code in a non-disruptive manner. This was considered acceptable for testing as the UK-DMC acts only as a source of DTN data, and does not need to receive and parse bundles from elsewhere.

Thus, the DTN-bundle-receiving intelligence only needed to be present in the ground station implementation of the *Saratoga* client and the DTN bundle agent. The *Saratoga* client in the ground station queries the UK-DMC satellite for a directory of files, and then requests any files with a ".dtn" extension and an associated satellite image file. (File naming conventions are discussed in detail later.) The satellite image file and associated metadata files are transferred to the ground, where the *Saratoga* client reassembles the bundles and then presents them to the full DTN bundle agent – full DTN-2 bundle agent implementations were used both at the ground station and the final DTN destination [12]. Finally, to demonstrate proactive fragmentation, the DTN fragments were

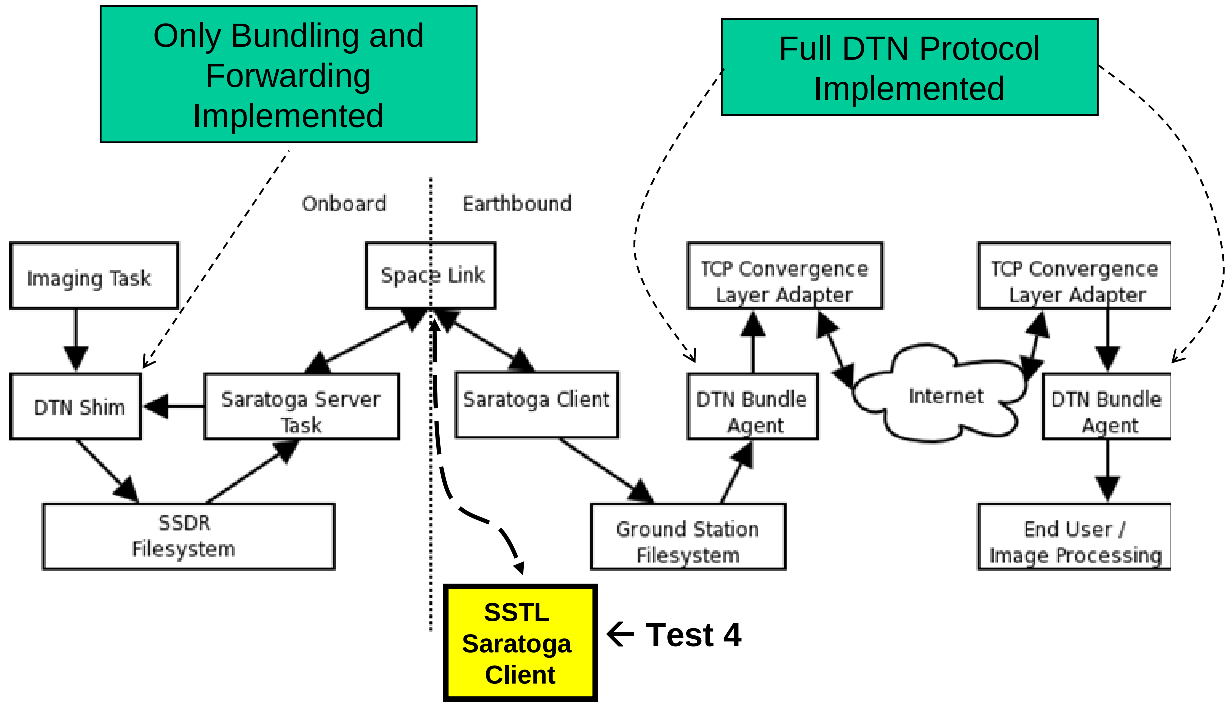

Figure 2 - UK-DMC Implementation

reassembled at the final DTN destination.

b. Ground development and testing

Figure 3 shows the DTN ground testbed, where bundling over *Saratoga* was prototyped, with a schematic diagram given in Figure 4. This development testbed, which reused the CLEO ground-based testbed duplicating in-orbit UK-DMC hardware, requires:

- The PowerPC-based Solid-State Data Recorder (SSDR) that resides in the Cisco router in Low Earth Orbit (CLEO) engineering model, where the bundle file is generated.
- A channel emulator that emulates the 9600 bps uplink and the 8.134 Mbps downlink. This uses a Spirent SX-14 data link simulator to provide channel delay and bit-error-rate emulation independently on both the uplink and downlink.
- A DTN bundle agent acting as the ground station. This bundle agent queries the DTN source onboard the SSDR for files and bundles using the SSTL *Saratoga* version 0 file transport protocol.
- A remote sink for DTN bundles – another bundle agent.

All network layer communications used IPv4, with the simulated space/ground data link implemented using Frame Relay/HDLC.

Figure 3 – CLEO ground-based testbed

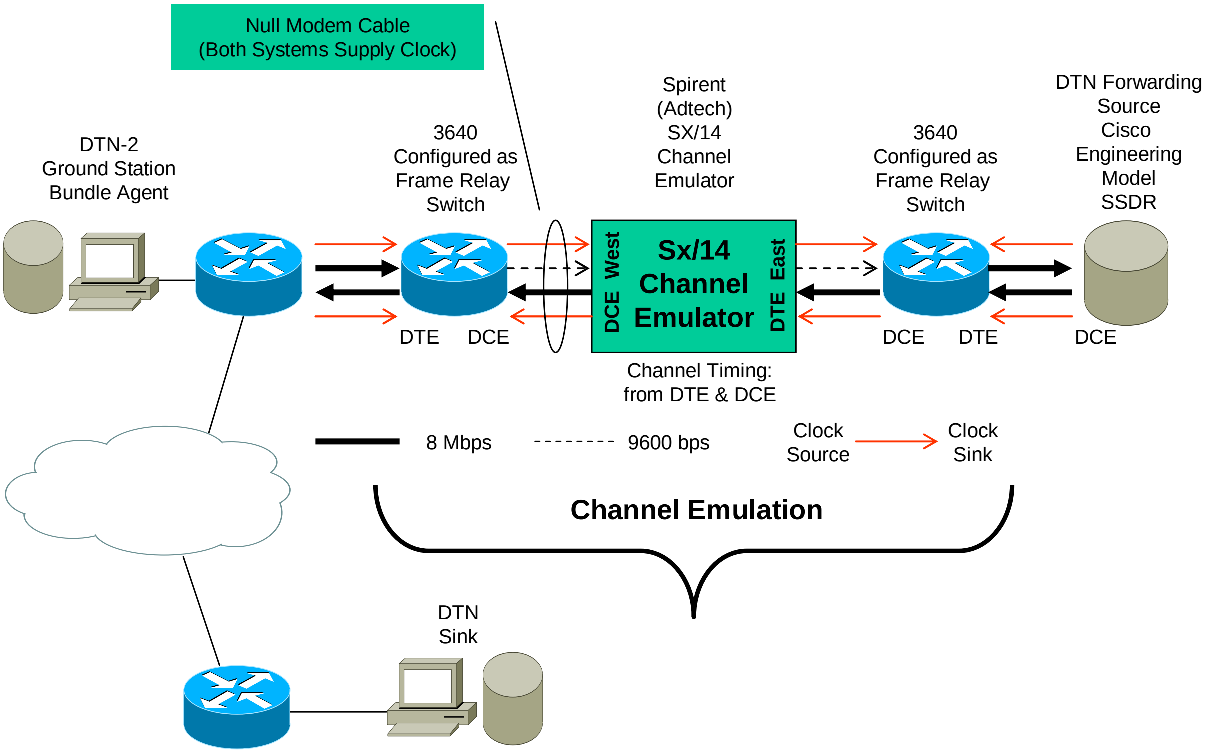

Figure 4 - DTN Testbed

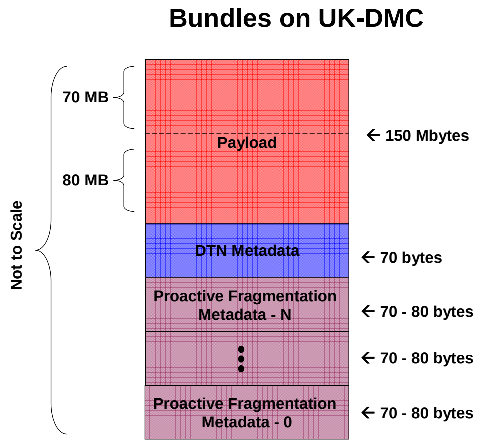

Figure 5 - Bundles on the UK-DMC

c. Overall goals of these DTN experiments

The goals of the experiments were to:

(1) Demonstrate that DTN code and general SSTL code can coexist without affecting normal SSTL spacecraft or ground station operations;
(2) Demonstrate DTN bundle transfers from UK-DMC to SSTL; and,
(3) Demonstrate proactive fragmentation of DTN bundles.

The ability to run DTN bundling without affecting normal SSTL operations would enable the DTN bundling code to remain loaded as part of the operational system. NASA will not need to take the UK-DMC out of normal operations for dedicated experimental use. This lack of impact on normal imaging operations will result in significant cost savings for future tests and demonstrations.

Demonstrating normal DTN bundle transfers verifies DTN operation and shows that *Saratoga* can be used as a bundle convergence layer. Proactive fragmentation is required to perform large file transfers over multiple ground stations.

V. TESTS

a. Test Configuration

In order to efficiently run as many tests as possible during a single satellite contact time, an analysis was performed to determine the optimal satellite image size to take.

In the pass time available, an image size of approximately 160 Mbytes would allow us to run a full 160-Mbyte file transfer, a 160-Mbyte DTN bundle transfer, and two 80-Mbyte DTN bundle fragment transfers during a satellite pass (single continuous contact).

For the first attempt at DTN testing, SSTL instructed the UK-DMC satellite to acquire a 150-Mbyte image over the Gulf of Khambhat, India at ~04:35 UTC on 25 January 2008.

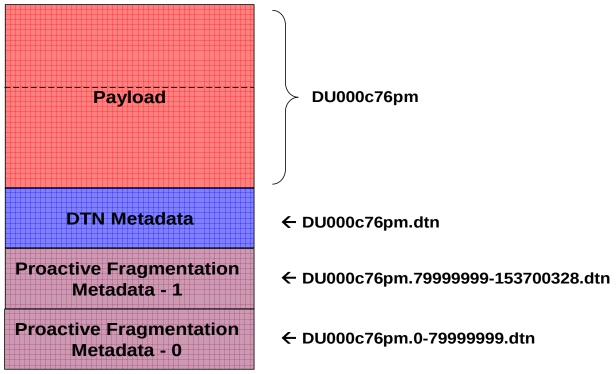

Figure 6 - File Naming Convention

b. Bundles on the UK-DMC satellite

Figure 5 shows how bundles were created onboard the UK-DMC satellite. When the image of the Gulf was acquired, the large 150-Mbyte image was stored in the SSDR and automatically named by the operating system.

The SSTL naming convention is to use a 10 character name for the recorded image. Here, the name was DU000c76pm. As well as this file, the DTN shim created three additional files of approximately 70 to 80 bytes. These files are the DTN bundle headers containing the DTN metadata. The first DTN bundle header contained metadata for the entire 150 Mbyte file. The second two DTN files contained DTN metadata used for proactive fragmentation. The arbitrary convention used to name the metadata files was to use the default system name with an extension of ".dtn" added to the full bundle name. For the fragmentation dtn metadata files, the system default name along with the start and stop file offset and the ".dtn" were used. For the 150-Mbyte satellite image, this resulted in two proactive fragment metadata files shown in figure 6:

DU000c76pm.0-79999999.dtn, and
DU000c76pm.79999999-153700328.dtn

Note, the image was not duplicated; only a small amount of additional metadata and filespace was required to perform proactive fragmentation.

c. *Results of DTN Tests*

Three UK-DMC satellite passes were taken to test the latest NASA/Cisco/SSTL firmware code supporting *Saratoga*/DTN bundling. The passes occurred as follows:

07:54 - 08:07 UTC	28 degrees maximum elevation.
09:31 - 09:45 UTC	45 degrees maximum elevation.
11:12 - 11:21 UTC	5 degrees maximum elevation.

Four tests were performed:

1) Basic image file download, using existing *Saratoga* file transfer techniques (GRC's implementation of *Saratoga* version 0)
2) Download of the same image file as a DTN bundle.

3) Download of the same file, using DTN proactive fragmentation with 80-Mbyte preconfigured fragments.
4) Normal file transfer using SSTL's workstation and SSTL's implementation of *Saratoga* version 0. This provided an operational control to be compared with the first three experiments [Figure 2].

For test 1, the satellite image file, DU00076pm, was received at the SSTL ground station in Guildford, England using NASA Glenn Research Center's implementation of *Saratoga* version 0. This file was then transferred to NASA GRC over the internet using normal file transport protocol (FTP).

For test 2, the satellite image file, DU00076pm, and associated DTN metadata file for the full bundle, DU00076pm.dtn, were received by the *Saratoga* client on the ground and presented as a full bundle to the bundling agent, Bundling-SSTL, at SSTL ground station. The bundle was automatically sent as a full bundle to the NASA Glenn Research Center DTN sink, Bundling-GRC1.

For test 3, proactive fragmentation, the first proactively-fragmented bundle file from the UK-DMC was received on the ground by the *Saratoga* client, the fragmentation bundle was reconstituted and presented to the DTN bundle agent, Bundling-SSTL. This bundle fragment was then automatically transferred from Bundling-SSTL to Bundling-GRC1 using DTN. The second proactive fragmentation bundle was not retrieved. Upon further investigation, the directory and the syslog file onboard the UK-DMC indicated that the first fragmentation metadata file was created, but not the second. Post-experiment analysis showed SSTL's operating system limits file names to 32 characters. This is a settable parameter. The file name, DU000c76pm.79999999-153700328.dtn, is 33 characters long and thus the file was not created.

Initial results showed all image files reconstructed at the GRC DTN bundle sink had the correct file size, but the contents did not match as there were long strings of zeros in various places in each file. The placement of these long strings of zeros was different for each file. SSTL performed an additional 'control' test, test 4, where they removed the GRC bundle agent and *Saratoga* client and replaced that machine with SSTL's normal *Saratoga* client machine [Figure 2]. The result was that SSTL downloaded the 150-Mbyte image without errors.

On the first pass, tests 1 and 2 were successful regarding operation of DTN and the ability to either use either *Saratoga* for straight file transfers or *Saratoga* with bundling to transfer DTN bundles between the UK-DMC payloads and the ground, demonstrating bundle delivery from space. Also, the DTN-2 forwarding agent, Bundling-SSTL, was able to automatically forward the DTN bundles to a DTN-2 bundling agent at NASA Glenn Research Center, Bundling-GRC1. It was then possible to then extract the image file from the DTN bundle.

d. Post-Test Analysis

The post-test analysis revealed a number of minor problems in the experiments conducted. The reconstructed DTN bundle payload and image file (tests 1 and 2) did not match. The DTN bundling and forwarding worked, but there was a problem in the NASA GRC implementation of the *Saratoga* client regarding filling holes in missed data. Retransmission requests were not performed properly. The programming problem has since been found and fixed.

A programming problem was also found in the DTN-2 code implementation put on the SSTL bundling agent, as one bundle became stuck in a temporary file and was never transferred from SSTL to GRC.

VI. OTHER KNOWN PROBLEMS AND ISSUES

a. Reliability, error detection, and checksums

The current Bundle Protocol specification does not address reliability, in that it has no checksum support for error detection and rejection of corrupted bundles. That means that one cannot determine if the bundle information received at each hop was received error-free. Error detection is a very basic networking concept that was overlooked in the bundle protocol design. The current proposed workaround is to use the bundle security specification and to wrap the bundle using a reliability-only cipher rather than a security cipher that provides a reliability check as a side-effect of security [13]. However, the bundle security specification was not implemented here. Thus, there were no reliability checks. If checksums had been implemented as part of the core DTN bundle specification, the "holes to fill" implementation problem would have been discovered early on, and corrupt bundles would not have been transferred through our entire DTN network.

b. Time synchronization problems

During initial ground testing it became apparent that network time synchronization is critical for DTN, which assumes that all communicating DTN nodes understand local UTC time. This is probably not a reasonable requirement for many DTN networks, as most DTN networks will be nondeterministic. Furthermore, DTN is a network overlay at the application layer that may be running on top of ad-hoc networks in highly stressed environments. The requirement that one can synchronize DTN networks is not necessarily practical or deployable. However, in this scenario, with scheduled LEO passes over a ground station, it is necessary for everything to know what the time is to support the pass opportunity. The question is – how much clock drift should be permissible?

The synchronization problem was experienced during initial ground testing. All DTN bundle agents were originally configured and tested at NASA GRC in Cleveland, Ohio. One bundle agent was sent to Guildford, England. A second was sent to Universal Space Networks (USN) in Alaska. When performing initial DTN bundle transfers from SSTL to GRC to USN, it was noted that the machine clocks had drifted sufficiently enough to result in the bundle time stamps being out of synchronization. The DTN bundles were therefore rejected due to time-stamp mismatch. Once the machines were resynchronized, DTN transfers operated correctly. This problem was articulated at the 71st Internet Engineering Task Force meeting in March of 2008. Others have noted similar problems [14].

VII. SUMMARY AND CONCLUSIONS

Delay-tolerant networking bundle transfers have been demonstrated from orbit.

The DTN bundling shim onboard the UK-DMC and the ground station *Saratoga* client and bundle reconstitution mechanisms should continue to operate without affecting normal UK-DMC operations, giving NASA access to an operational DTN testbed in orbit.

Some minor software implementation bugs regarding retransmission of errored packets and file name limitations were identified and have since been corrected and ground tested. We are awaiting further on-orbit testing opportunities.

The lack of integrity checksums in the Bundle Protocol and the need for DTN network synchronization have shown to be real deployment issues during our initial tests. We hope that these architectural issues will be examined in future versions of the DTN architecture and bundling specifications.

REFERENCES

[1] K. Scott and S. Burleigh, “Bundle Protocol Specification,” IETF RFC5050, November 2007.

[2] V. Cerf *et al*., “Delay-Tolerant Network Architecture,” IETF RFC 4838, April 2007.

[3] DMC Imaging International: http://www.dmcii.com/

[4] Martin Unwin, Scott Gleason and Michael Brennan, “Sensing Ocean, Ice and Land Reflected Signals from Space: Results from the UK-DMC GPS Reflectometry Experiment,” ION GNSS 18th International Technical Meeting of the Satellite Division, 13-16 September 2005, Long Beach, California.

[5] Scott Gleason *et al*., “Processing of bistatically reflected GPS signals from low Earth orbit for the purpose of ocean remote sensing,” *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 43, No. 6, pp. 1229-1241, June 2005.

[6] Will Ivancic, Phil Paulsen, Dave Stewart, Dan Shell, Lloyd Wood, Chris Jackson, Dave Hodgson, James Northam, Neville Bean, Eric Miller, Mark Graves and Lance Kurisaki, “Secure, Network-Centric Operations of a Space-Based Asset: Cisco Router in Low-Earth Orbit (CLEO) and Virtual Mission Operations Center (VMOC),” NASA Technical Memorandum 2005-213556, May 2005.

[7] Lloyd Wood, Will Ivancic, Dave Hodgson, Eric Miller, Brett Conner, Scott Lynch, Chris Jackson, Alex da Silva Curiel, Dan Shell, Jon Walke and Dave Stewart, “Using Internet nodes and routers onboard satellites,” International Journal of Satellite Communications and Networking, volume 25 issue 2, pp. 195-216, March/April 2007.

[8] K. Hogie, E. Criscuolo and R. Parise, “Using standard Internet Protocols and applications in space,” Computer Networks, special issue on Interplanetary Internet, vol. 47 no. 5, pp. 603-650, April 2005.

[9] Will Ivancic, Dave Stewart, Lloyd Wood, Chris Jackson, James Northam and James Wilhelm, “IPv6 and IPSec Tests of a Space-Based Asset, the Cisco router in Low Earth Orbit (CLEO),” NASA Technical Memorandum 2008-215203, May 2008.

[10] Lloyd Wood, Will Ivancic, Dave Stewart, James Northam, Chris Jackson and Alex da Silva Curiel, “IPv6 and IPsec on a satellite in space,” conference paper B2.6.06, 58th International Astronautical Congress, Hyderabad, India, September 2007.

[11] Lloyd Wood, Jim McKim, Wesley M. Eddy, Will Ivancic and Chris Jackson, ”Saratoga: A Scalable File Transfer Protocol,” work in progress as an internet-draft, draft-wood-tsvwg-saratoga-01.txt, February 2008.

[12] DTN reference implementation, October 2007 release, available from http://www.dtnrg.org/wiki/Code

[13] Wesley M. Eddy, Lloyd Wood and Will Ivancic, “Checksum Ciphersuites for the Bundle Protocol,” work in progress as an internet-draft, draft-irtf-dtnrg-bundle-checksum-02, March 2008.

[14] Wesley M. Eddy, “DTN Time Sync Issues,” email to the IRTF dtn-interest mailing list, 1 April 2008, and subsequent discussion.