INTERNATIONAL JOURNAL OF SATELLITE COMMUNICATIONS AND NETWORKING
Int. J. Satell. Commun. Network. 2010; **28**:335–351
DOI: 10.1002/sat.966

Experience with Delay-Tolerant Networking from orbit[‡]

W. Ivancic[1], W. M. Eddy[1,2], D. Stewart[1,2], L. Wood[3,*,†], J. Northam[4] and C. Jackson[4]

[1]*NASA Glenn Research Center, Cleveland, OH, U.S.A.*
[2]*Verizon Federal Network Systems, Cleveland, OH, U.S.A.*
[3]*Cisco Systems Global Government Solutions Group, Bedfont Lakes, London, England*
[4]*Surrey Satellite Technology Ltd, University of Surrey Research Park, Guildford, England*

SUMMARY

We describe the first use from space of the Bundle Protocol for Delay-Tolerant Networking (DTN) and lessons learned from experiments made and experience gained with this protocol. The Disaster Monitoring Constellation (DMC), constructed by Surrey Satellite Technology Ltd (SSTL), is a multiple-satellite Earth-imaging low-Earth-orbit sensor network in which recorded image swaths are stored onboard each satellite and later downloaded from the satellite payloads to a ground station. Store-and-forward of images with capture and later download gives each satellite the characteristics of a node in a disruption-tolerant network. Originally developed for the 'Interplanetary Internet,' DTNs are now under investigation in an Internet Research Task Force (IRTF) DTN research group (RG), which has developed a 'bundle' architecture and protocol. The DMC is technically advanced in its adoption of the Internet Protocol (IP) for its imaging payloads and for satellite command and control, based around reuse of commercial networking and link protocols. These satellites' use of IP has enabled earlier experiments with the Cisco router in Low Earth Orbit (CLEO) onboard the constellation's UK-DMC satellite. Earth images are downloaded from the satellites using a custom IP-based high-speed transfer protocol developed by SSTL, *Saratoga*, which tolerates unusual link environments. *Saratoga* has been documented in the Internet Engineering Task Force (IETF) for wider adoption. We experiment with the use of DTNRG bundle concepts onboard the UK-DMC satellite, by examining how *Saratoga* can be used as a DTN 'convergence layer' to carry the DTNRG Bundle Protocol, so that sensor images can be delivered to ground stations and beyond as bundles. Our practical experience with the first successful use of the DTNRG Bundle Protocol in a space environment gives us insights into the design of the Bundle Protocol and enables us to identify issues that must be addressed before wider deployment of the Bundle Protocol. Published in 2010 by John Wiley & Sons, Ltd.

KEY WORDS: Internet; UK-DMC; satellite; Delay-Tolerant Networking (DTN); Bundle Protocol

1. INTRODUCTION

Delay-Tolerant Networking (DTN) has been defined as the concept of end-to-end store-and-forward delivery, capable of providing communications in highly-stressed or disrupted network

---

*Correspondence to: Lloyd Wood, Center for Communication Systems Research, University of Surrey, Guildford GU2 7XH, England, U.K.

[†] E-mail: L.Wood@surrey.ac.uk

[‡] This article is a U.S. Government work and is in the public domain in the U.S.A.

environments considered 'unusual' from the perspective of the terrestrial Internet [1]. DTN networks can be thought of as operating across varying conditions along several different axes, depending on the design of the subnet being traversed:

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

One way to provide the store-and-forward service in these DTN networks is a new 'Bundle Protocol'. This acts as an overlay to some number of constituent networks [2]. Key capabilities of this Bundle Protocol include:

- Custody transfer—the ability for a bundle agent to take full responsibility for a bundle reaching its final destination.
- Ability for implementations to cope with intermittent connectivity, if required.
- Ability for implementations to cope with long propagation delays, if required.
- Ability to take advantage of scheduled, predicted and opportunistic connectivity (in addition to continuous connectivity).
- Late binding of endpoint identifiers in the overlay bundle network, to network addresses in the underlying constituent networks [3].

The Bundle 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.

In a low-propagation-delay environments, such as may occur in near-planetary or terrestrial environments, bundle agents can use chatty Internet transport protocols, such as TCP, that negotiate connectivity and handshake connections in real time.

In high-propagation-delay environments, such as deep space, DTNRG 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 Internet Protocol (IP).

The Bundle Protocol was originally developed for deep-space use, and was proposed as the core of the 'Interplanetary Internet' for civil space missions. Evaluating the utility of the Bundle Protocol in space has significant bearing on the development of this envisaged space network.

Our experiments with the Bundle Protocol onboard the UK Disaster Monitoring Constellation (UK-DMC) satellite did not have high propagation delays, but were intended to experiment with the proactive fragmentation feature of the Bundle Protocol, which would allow files to be transferred even when they are too large to be completely transferred during a single contact opportunity over a ground station. The experiments also demonstrate the utility of IP for space use, even though it is used in hop-by-hop data transfers to destinations to get the most from the conditions on each local link, rather than in the 'end-to-end' path paradigm found terrestrially. We describe our experiments, draw conclusions about the Bundle Protocol, based on experience gained from those experiments, and briefly summarize other later experiments in space with the Bundle Protocol. These experiments in space have bearing on how the 'Interplanetary Internet' for civil space missions will be developed.

2. THE DMC OPERATING ENVIRONMENT

Low Earth Orbit (LEO) provides a low-propagation-delay environment of less than ten milliseconds one-way delay to ground, with long periods of disconnection between scheduled passes over ground stations.

For the DMC imaging satellites in LEO, contact times consist of five to fourteen minutes per pass, depending on relative positioning of the ground station and satellite track, with one or two available ground station contact times per 100 min orbit.

The ground stations are connected across the public terrestrial Internet, which has different operating conditions (shared, competing, congestion-sensitive, always on) from the private links between satellite and ground station (intermittent but scheduled, and dedicated to downloading).

3. THE RATE MISMATCH PROBLEM

Figure 1 illustrates a LEO satellite ground network with a bundle agent sink located at a remote location. The final destination 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, the LEO satellite, to the DTN sink. In this example, the image file is too large to be transferred during one pass over a single ground station. Three passes are required to transfer the complete file to ground. These

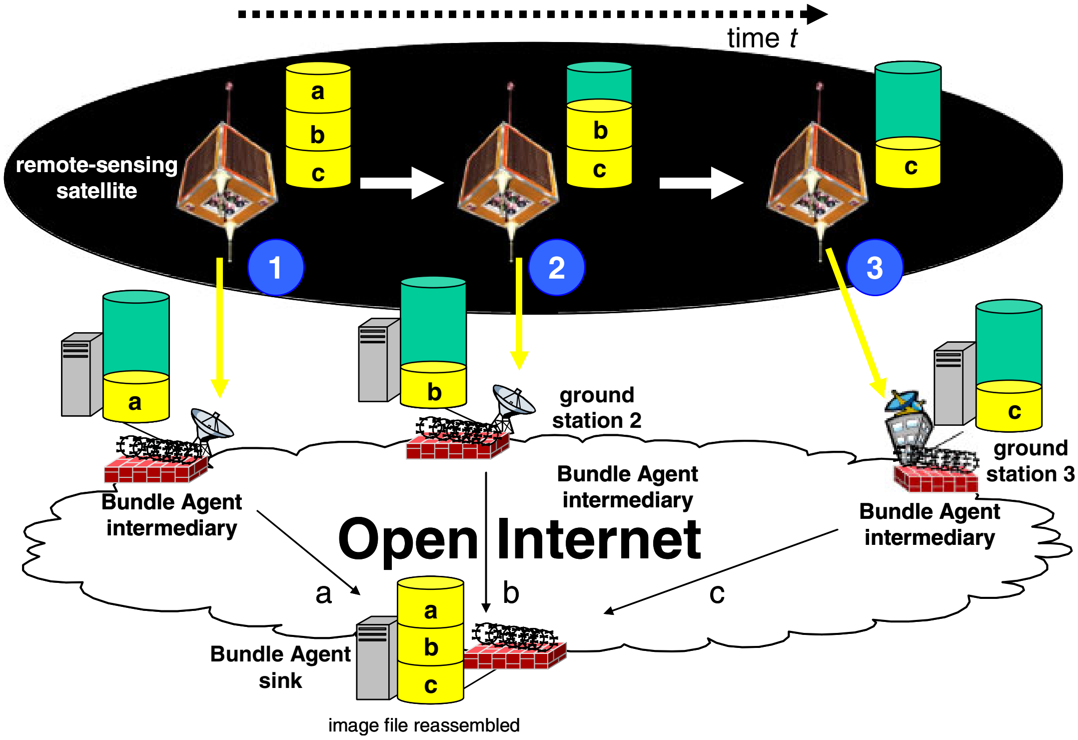


Figure 1. Use of bundling and fragmentation across multiple passes.

could either all be through the same ground station, or could utilize three different ground stations, from left to right in the diagram.

In the example in Figure 1, the minimum time a complete image file could be transferred using a single ground station is a little over 300 min, assuming one pass per 100-min 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 Internet Research Task Force (IRTF) DTN Research Group's Bundle Protocol is one example of a way to provide such functionality to split the path into separate hops and control loops. It can therefore compensate for rate mismatches between the private space-to-ground link and the shared path between ground station and remote destination for the image.

4. CHARACTERISTICS OF THE UK-DMC SATELLITE

The UK-DMC satellite is one of seven similar imaging satellites currently launched into Low Earth Orbit (LEO) 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 two more satellites to be added in the next two years to maintain a continuous on-orbit imaging capability. Although most of 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 buy a requested image [4].

The UK-DMC is primarily an operational imaging satellite and not an experimental satellite. However, Surrey Satellite Technology Ltd (SSTL) has also run secondary experiments onboard the UK-DMC, such as investigating GPS reflectometry [5], and networking experiments have taken advantage of an onboard Internet router [6, 7]. SSTL continues to permit NASA Glenn 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 DTNRG Bundle Protocol 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 512-Mbyte RAM, respectively.

- An uplink of 9600 bits per second and a downlink of 8.134 Mbps—this is highly asymmetric. Both links use the proven IPv4/Frame Relay/HDLC commercial-standard protocol stack developed for space use by Hogie *et al.* [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*, called version 0 based on its version field, running over UDP.

Saratoga version 0 is the existing operational SSTL file transfer protocol, originally developed to replace and improve transfer performance rates over an implementation of CCSDS File Delivery Protocol (CFDP) that was previously used by SSTL. *Saratoga* version 1 is a slightly improved specification, with enhancements to *Saratoga* version 0, which has now been documented publicly as a contribution to the Internet Engineering Task Force (IETF) [11].

Our use of *Saratoga* as a bundle convergence layer to carry DTNRG bundles has also been publicly documented [12].

5. EXPERIMENTAL BUNDLING IMPLEMENTATION

5.1. *Onboard the UK-DMC satellite*

Figures 2 and 3 show how DTNRG bundling is implemented onboard the UK-DMC and in the ground infrastructure.

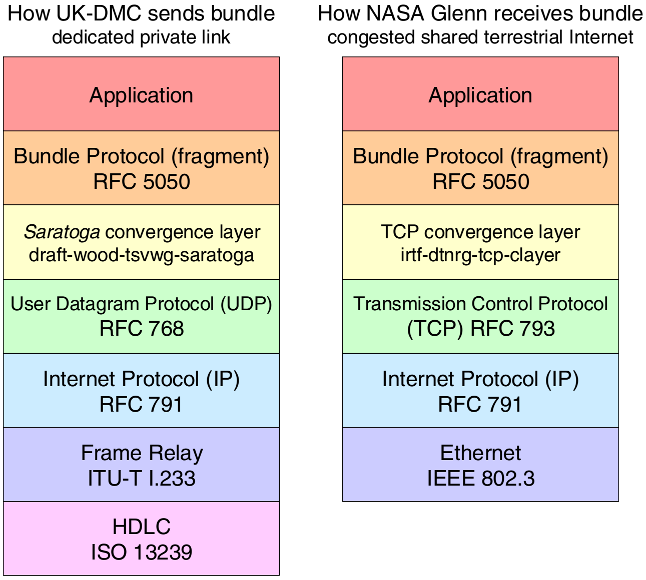


Figure 2. The protocol stacks used for these Bundle Protocol tests.

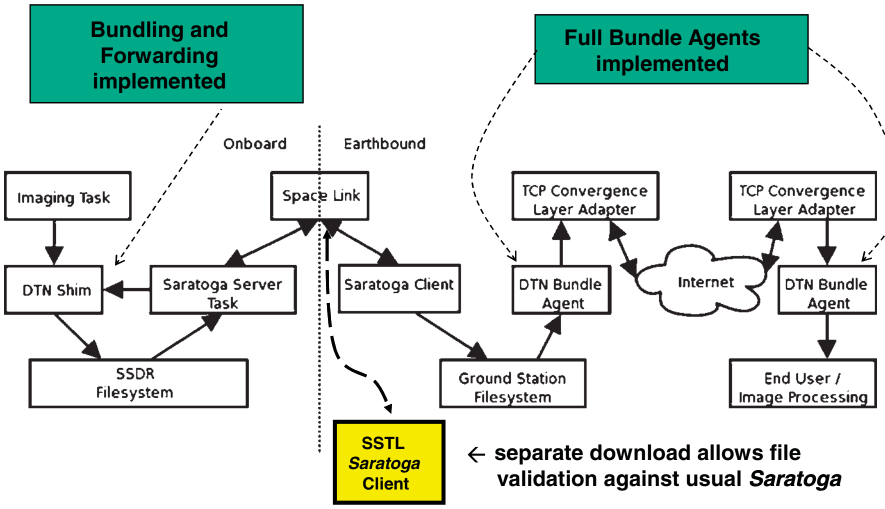


Figure 3. How bundling was implemented for downloads from the UK-DMC satellite.

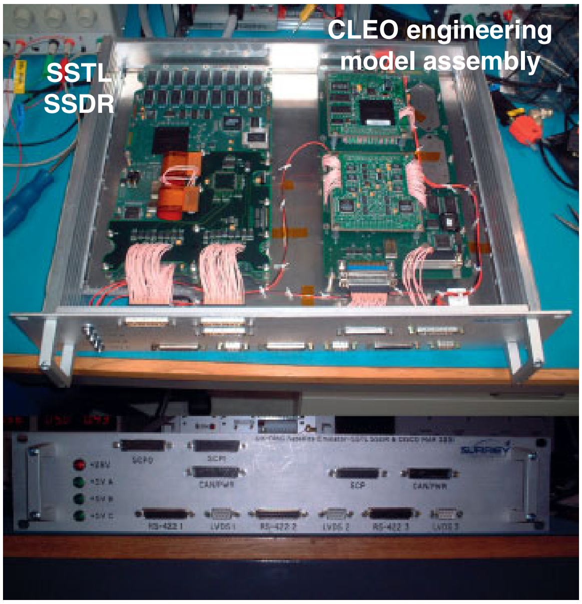
The *Saratoga* implementation (at the time of experimentation, the operational version 0, rather than the later, publicly documented version) acts as a bundle transport 'convergence' layer on the space-ground link. Only the bundle-forwarding portion of the Bundle Protocol was implemented onboard as a simple networking 'shim' as available code space is constrained. A goal is to have the onboard Bundle Protocol 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.

The DTN bundle-receiving intelligence only needed to be present in the ground station implementation of the *Saratoga* client and the bundle agent. The *Saratoga* client in the ground station queries the UK-DMC satellite for a directory of files, and then requests any bundle metadata files with a '.dtn' extension and an associated satellite image file that contains the payload used to construct the bundle. The satellite image file and the associated metadata files are transferred to the ground, where the *Saratoga* client reassembles the bundles and then presents them to the full bundle agent—full DTN2 *dtnd* bundle agent implementations were used both at the ground station and the final DTN destination [13]. Finally, to demonstrate proactive fragmentation, the bundle fragments are reassembled at the final DTN destination.

Deploying bundle functionality on the satellite required that all the new pieces of that functionality were first implemented and tested on the ground against emulated pieces of the rest of the operational system.

5.2. Ground development and testing

Figure 4 shows the DTN ground testbed, where bundling over *Saratoga* was prototyped, with a schematic diagram given in Figure 5.



a. top view, before adding fans and heatsinks.
b. front view of ports

Figure 4. CLEO ground-based testbed.

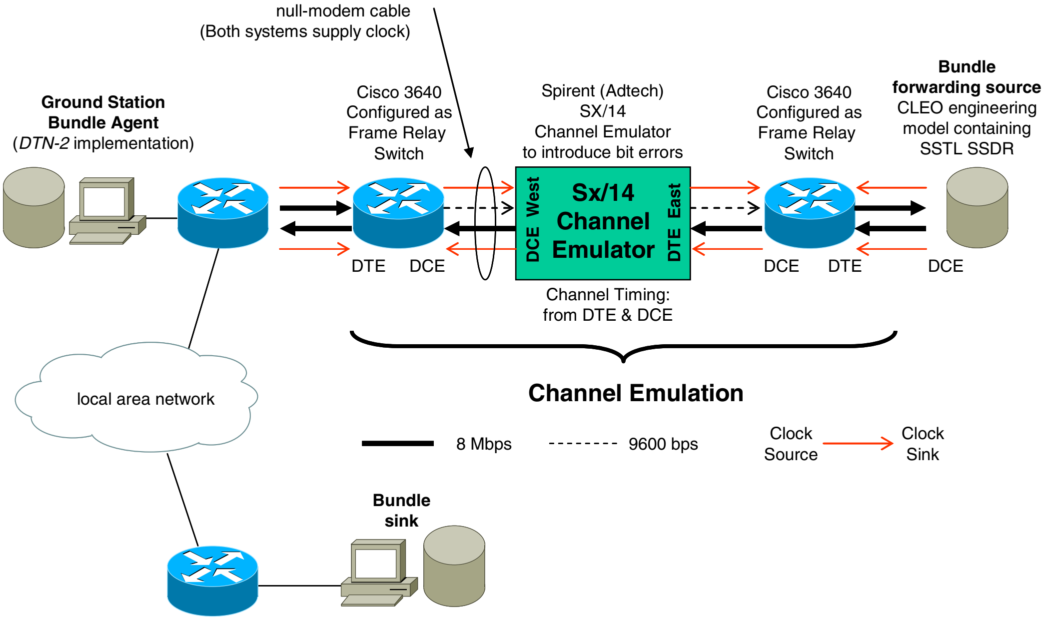


Figure 5. NASA Glenn's DTN testbed.

Published in 2010 by John Wiley & Sons, Ltd.

DOI: 10.1002/sat

This development testbed, which reused the CLEO ground-based testbed duplicating in-orbit UK-DMC hardware, contains:

- The PowerPC-based SSDR that resides in the CLEO engineering model, where the bundle file is generated by reading data from an emulated satellite imaging device.
- 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 bundle agent acting as the ground station, which queries the DTN source onboard the SSDR for files and bundles sent using the *Saratoga* version 0 transfer protocol.
- A remote sink for bundles—another bundle agent.

All network-layer communications used IPv4. The simulated space/ground data link was implemented using Frame Relay and HDLC to match the real space/ground link as closely as possible.

We also deployed bundle agent software at several remote ground stations to create a hub-and-spoke topology around NASA Glenn Research Center (GRC)'s bundle agent, to gain experience with managing bundle agent deployment on the scale needed for coordinating multiple ground stations for cooperative fragmented large file transfers.

5.3. Overall goals of these bundle experiments

The goals of the experiments were to:

- demonstrate that NASA Glenn's code additions can coexist with SSTL's code without affecting normal SSTL spacecraft or ground station operations;
- demonstrate bundle transfers from the UK-DMC satellite to SSTL and NASA Glenn and
- demonstrate proactive fragmentation of bundles to allow downloads across multiple passes.

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

Demonstrating normal DTNRG bundle transfers verifies DTN operation and shows that *Saratoga* can also be used as a bundle convergence layer. Proactive fragmentation allows the download to tolerate disruption between satellite passes, and is required to perform large file transfers over multiple passes and multiple ground stations.

6. BUNDLING TESTS FROM ORBIT

To efficiently run as many bundling tests as possible during a single satellite contact time, an analysis was performed to determine the optimal satellite image size to take. During a ten-minute pass over a ground station, just over 600 Mbytes of data can be transferred from the UK-DMC satellite; this varies with the elevation and duration of the pass over the ground station. Calculations suggested that, in the likely pass time available, an image size of approximately 160 Mbytes would

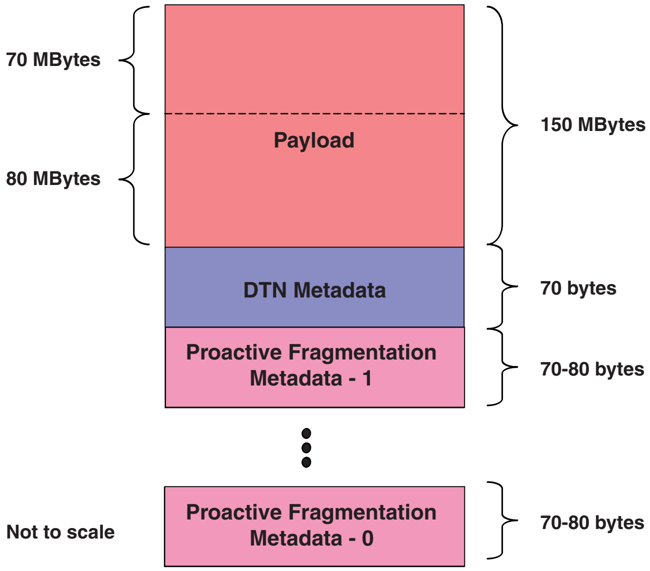


Figure 6. Bundles on the UK-DMC.

allow us to carry out a full 160-Mbyte file transfer, a 160-Mbyte bundle transfer and two 80-Mbyte bundle fragment transfers during a single satellite pass (single continuous contact).

Figure 6 shows how bundles were created onboard the UK-DMC satellite. When the image was acquired, the large 150-Mbyte image was stored in the SSDR and automatically named by the operating system. Small metadata files were created by our modifications to accompany the image files.

6.1. Initial January 2008 on-orbit tests

Partially-successful tests of bundling image files over *Saratoga* were carried out on 25 January 2008. Three UK-DMC satellite passes were taken to test the latest NASA/Cisco/SSTL firmware code supporting *Saratoga*/DTNRG bundling. Four tests were performed:

- Basic image file download, using existing *Saratoga* file transfer techniques (NASA GRC's implementation of *Saratoga* version 0).
- Download of that file as a DTNRG bundle.
- Download of the same file, using proactive fragmentation with 80-Mbyte preconfigured fragments, by creating additional small files containing metadata information (Figure 7).
- 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.

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

For test 2, the satellite image file, DU00076pm, and associated bundle metadata file for the full bundle, DU00076pm.dtn, were received by the *Saratoga* client on the ground and presented as a

DOI: 10.1002/sat

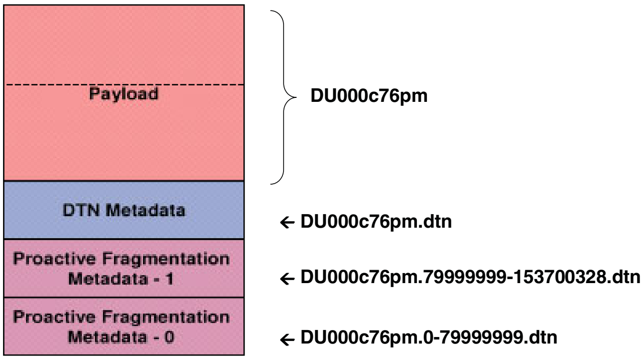


Figure 7. File naming convention.

full bundle to the bundling agent, *Bundling-SSTL*, at SSTL's ground station. This was resent as a full bundle across the Internet to the NASA Glenn Research Center DTN sink, *Bundling-GRC1*, using the TCP convergence layer implemented in the DTN2 *dtnd* implementation [14].

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 DTNRG bundle agent, *Bundling-SSTL*. This bundle fragment was then automatically transferred from *Bundling-SSTL* to *Bundling-GRC1* using *dtnd*. The second proactive fragmentation bundle was not retrieved. On 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 that 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 bundle sink had the correct file size, but that the file 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 differed for each file. These errors in assembling the bundle at the destination went entirely undetected by the Bundle Protocol.

SSTL performed an additional control test, test 4, where the ground station computer running the GRC bundle agent and *Saratoga* client was replaced by one with SSTL's normal *Saratoga* client (Figure 3). That copy of the 150-Mbyte image was downloaded without errors.

On the first pass, tests 1 and 2 were successful regarding the operation of the Bundle Protocol and the ability to either use either *Saratoga* for straight file transfers or *Saratoga* with bundling to transfer DTNRG bundles between the UK-DMC payloads and the ground, demonstrating bundle delivery from space. Also, the DTN2 forwarding agent, *Bundling-SSTL*, was able to automatically forward the DTN bundles to a DTN2 bundling agent at NASA Glenn Research Center, *Bundling-GRC1*. It was then possible to extract the image file from the bundle.

The post-test analysis revealed a number of minor problems in the experiments conducted. The reconstructed bundle payload and image file (tests 1 and 2) did not match. The 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, to resend packets errored and dropped during the start and end of the pass, were not being performed properly. This programming problem was later fixed and tested extensively on the ground testbed using the channel emulator to introduce bit errors.

A programming problem was also found in the DTN2 code implementation put on the SSTL bundling agent in the ground station, as a bundle became stuck in a temporary file and was never transferred to GRC.

6.2. Successful August 2008 on-orbit tests

An unsuccessful bundle image download was carried out during two passes on 26 August, using an older code version that led to corrupted fragments.

Replacement code, with a bugfix giving correct fragmentation offsets, was then uploaded to the UK-DMC's SSDR. A remote sensing image swath over South Africa was taken on 08:27 UTC on 27 August 2008. Successful download tests, with reassembly of that proactively-fragmented image file downloaded over two passes, were carried out that morning and are the first successful uses of the Bundle Protocol from space [15]. In these successful tests, the image taken by the UK-DMC satellite's cameras was stored as a single bundle as well as proactively fragmented into two bundles onboard the UK-DMC's SSDR, as previously shown in Figure 6. These bundle fragments were then downloaded during two passes over SSTL's ground station, to a bundle agent living on a computer donated by NASA Glenn. That bundle agent then forwarded the bundle fragments over TCP to NASA Glenn Research Center, in Cleveland, Ohio, where the fragments were reassembled into a 150-Mbyte file containing the raw sensor data.

That file was then returned to SSTL for post-processing to generate the final image. Figure 8 shows the resulting image of Southern Africa. The Cape of Good Hope and False Bay are to the west. This is a false-color image; vegetation is red, whereas the Karoo desert, inland on the plateau, is gray.

The image data was also downloaded using SSTL's standard operational method, using *Saratoga* version 0 only, for comparison with the bundle delivery method and validation of the bundle delivery.

We noticed some minor differences in operation and performance between the NASA Glenn and SSTL implementations of *Saratoga*.

The NASA Glenn *Saratoga* implementation can currently time out and reset to requesting the start of the file, rather than the left edge of its window. This needs to be fixed for efficient resumption of disrupted transfers via *Saratoga*.

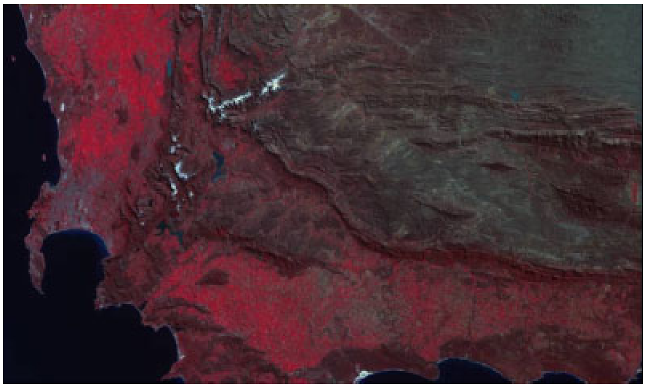


Figure 8. Image delivered via bundles: first useful sensor data delivered from space via the Bundle Protocol.

The more mature SSTL *Saratoga* implementation performs slightly more efficiently by combining selective negative acknowledgements for nearby blocks even though some unnecessary data resend results. This technique avoids congestion of the bottleneck 9600-bps uplink, leading to better download performance when the bit error rate is high, which is mostly at the start and end of passes when the satellite is at a low elevation.

7. ISSUES ENCOUNTERED IN THE CURRENT BUNDLE PROTOCOL DESIGN

Our practical experience, recounted in this study, and other detailed analyses have enabled us to identify a number of problems with the current design of the Bundle Protocol.

In this study, we summarize some of the significant problems with the Bundle Protocol that we have encountered during our practical testing. These and other problems and related issues are discussed in greater detail elsewhere [16].

7.1. Reliability, error detection and checksums

We earlier described problems encountered in our January 2008 testing due to the lack of error checking in the Bundle Protocol.

The current published 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 easily determine if the bundle information received at each node was received error-free or not.

Error detection is a very basic networking concept that was overlooked in the Bundle Protocol design. The design of the bundle architecture completely ignores the well-known *end-to-end principle* [17].

Without useful error detection, the Bundle Protocol's custody transfer mechanism cannot guarantee that a node taking responsibility for final delivery of a bundle has actually received an uncorrupted copy of that bundle to send on.

The Bundle Protocol is intended to permit delivery of errored content, as some applications may find it desirable to receive errored content rather than no content at all, in the case where a bundle is corrupted [18]. However, the basic Bundle Protocol does not protect its own header data, nor does it satisfy the needs of applications that do not inherently tolerate payload bit errors and that expect a transport to provide reliability.

Leaving error recovery up to the applications is only possible when the applications are tightly coupled across the network, with a tight control loop for resends of errored data. DTN networks, by their *ad hoc* nature, are loosely coupled, and there may not be any direct communication or control loop between applications at end nodes, requiring increased assistance from the network to improve performance—in line with the end-to-end principle.

We have proposed a workaround extension to the Bundle Protocol to add reliability into the existing protocol infrastructure. This is to use the bundle security specification and to wrap the bundle using a reliability-only cipher (a null-keyed hash function construction) rather than relying on a security cipher (a keyed Message Authentication Code or signature algorithm) that provides a reliability check as a side effect of security [19]. However, this bundle security specification was not implemented onboard the UK-DMC satellite. Using the existing bundle security protocol to support reliability also has some drawbacks, as discussed in detail elsewhere [16].

To provide a measure of reliability checking, we have now implemented an optional MD5 checksum for the *Saratoga* protocol, which can be used to compare hash values of files before and after downloading. The MD5 computation can take several minutes to run over a large file hence is likely to be used sparingly onboard. Given that image data is often downloaded in 'one shot' before being deleted to make room for new images, and post-processed heavily with human inspection, the need to resend image files with slight corruption is minor, although knowing where that corruption may lie in the image data would be useful. However, overall reliability checking becomes very important when e.g. uploading code to be executed.

7.2. Time synchronization problems

A clock synchronization problem was experienced during initial ground testing. All 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 Network (USN) in Alaska. When performing initial bundle transfers from SSTL to GRC to USN, it was noticed that the machine clocks had drifted far apart enough to result in the bundle creation time stamps being out of synchronization. The bundles were therefore rejected due to mismatches in system times leading to unexpected expiry of the bundles. Once the machines were resynchronized, bundle transfers operated correctly. Bundle expiry times could have been increased and set further into the future to tolerate this clock slippage, but this would not have prevented the problem of bundles being sent 'from the future' to a node with a slow clock.

Our initial ground testing made clear that network time synchronization is critical for the Bundle Protocol, which assumes that all communicating bundle nodes share a common, synchronized, understanding of local UTC time. This is probably not a reasonable requirement for many DTN networks. Many DTN networks will have non-deterministic time-varying topologies making time synchronization more difficult. Furthermore, the Bundle Protocol may be running on low-end hardware in *ad hoc* networks in highly-stressed environments. The requirement that all DTN networks running the Bundle Protocol must be synchronized to enable interoperation is not necessarily one that is either practical or deployable. Network robustness is sacrificed by this design choice.

With scheduled LEO passes over a ground station, it is necessary to know what the time is to support the pass opportunity. However, in our initial CLEO/Virtual Mission Operations Center testing, nodes in the field at Vandenberg were still able to operate with clocks set several minutes adrift; the loosely-coupled architecture tolerated this [7].

Expecting DTN nodes with loosely-coupled *ad hoc* connectivity to be tightly coupled with respect to their understanding of clock time has interesting ramifications. A side effect of requiring shared use of UTC time is that it would not be possible for a node to learn the correct time using the Bundle Protocol, as its bundles sent asking for the time are likely to be judged expired or invalid and be discarded, based on their erroneous timestamps. Another protocol would be required to do clock 'housekeeping'. Another concern is that for nodes 'in the field' for a long time (decades), some way of communicating newly-decided leap seconds would be required to prevent clock drift that would eventually inhibit transfers of bundles with short expiration times.

Problems with a shared universal clock were articulated at the 71st IETF meeting in March 2008. Others have noted similar problems in experiments funded through DARPA and other programs [20].

8. OTHER LATER TESTS IN SPACE

The Bundle Protocol was later tested in space in October and November 2008 by NASA's Jet Propulsion Laboratory. The Deep Impact Network Experiment (DINET) was conducted onboard the Deep Impact comet probe, in cooperation with the Extrasolar Planet Observation and Deep Impact eXtended Investigation (EPOXI) project [21].

In those later experiments, small images were uploaded to the spacecraft, where a bundle agent acted as a relay, and then returned to a terrestrial network.

File transfers were conducted using the Bundle Protocol and the Licklider Transmission Protocol (LTP) over the existing network infrastructure of the spacecraft, which uses CFDP [22]. The resulting network stack structure is shown in Figure 9, which can be compared with Figure 2.

DINET implemented a full DTN store-and-forward relay, including automated routing using Contact Graph Routing (CGR) [23] and compressed bundle header compression (CBHE) [24] to compact bundle headers. CGR requires *a priori* knowledge of all contacts, which is not unreasonable for a deep-space network. CBHE requires use of a highly-simplified naming scheme that is applicable to a small deep-space backbone.

DINET was a successful experiment showing the applicability of bundling and automated routing to deep space networks. Bundle sizes sent were limited to 64 Kbytes; therefore, thumbnail images were used as data, uploaded from Earth, and relayed back to Earth. As DINET was an add-on experiment and was not to interact with mission-critical flight code, it was not given access to onboard sensors. CCSDS protocols were used for the space links (Figure 9).

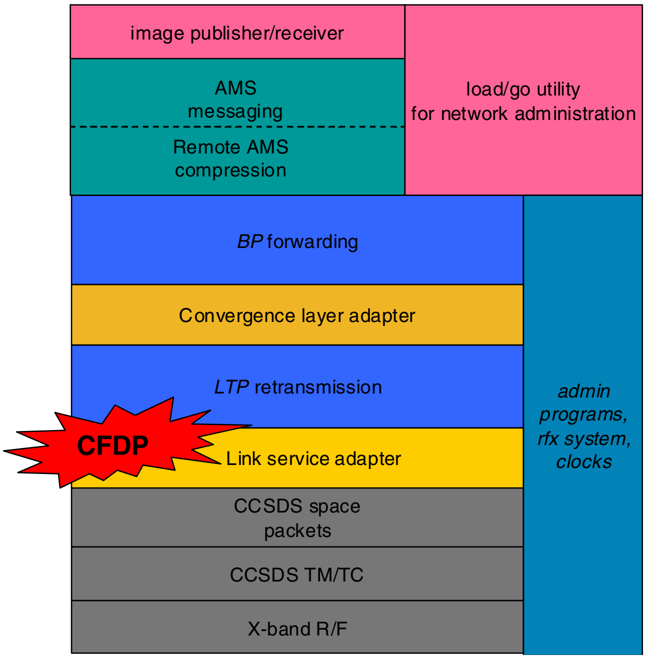


Figure 9. DINET Deep Impact stack [25].

The Internet Protocol was not used onboard Deep Impact or in the space portion of DINET, although these experiments were hailed as the start of the 'Interplanetary Internet' [26].

Deep Impact's clock drifts considerably in the cold of space, and had to be reset before each DINET test [27]. This has accord with the problems that we experienced with lack of clock synchronization adversely affecting bundle use.

Both the UK-DMC and DINET bundling experiments leveraged existing available network stacks that already supported packet-based transmission. Both the DMC's *Saratoga* and Deep Impact's CFDP installations were modified to support the Bundle Protocol, either by carrying bundles directly or by carrying bundles within LTP.

Neither experiment implemented the Bundle Security Protocol onboard the spacecraft.

9. CONCLUSIONS

DTN Bundle Protocol transfers have now been successfully demonstrated from the orbit with the download of sensor data in proactively fragmented bundles.

This has demonstrated the ability to download data across multiple satellite passes, despite the disruption and link loss experienced between those passes.

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

Our practical experience gained with implementing and operating the Bundle Protocol from the orbit enables us to consider aspects of the Bundle Protocol's design.

The lack of integrity checksums for reliability checks in the Bundle Protocol and the need for network time synchronization were shown to be real deployment issues during our first tests, and we are investigating new checksum mechanisms for the Bundle Protocol and ways to remove the protocol's dependence on clock synchronization.

The addition of a common Bundle Protocol overlay can facilitate more automated routing of data and increase interoperability for network-centric operations between organizations and assets. We hope that the problems with the Bundle Protocol that we have experienced and identified will be addressed in the later versions of the DTN architecture and Bundle Protocol specifications.

The DMC satellites and their use of IP for imaging transfers provide working operational examples of effective use of IP for sensor networks. This allows easy integration with the terrestrial Internet for data delivery. This mission-critical use of the *Saratoga* protocol and IP to carry sensor data performs well on a daily basis, without requiring the Bundle Protocol.

ACKNOWLEDGEMENTS

We thank the DTNRG for its work on the Bundle Protocol, Michael Demmer for his implementation of DTN2, and TIME Magazine for mentioning this work as one of its inventions of the year [28]. We also thank the anonymous reviewers for their comments improving this paper.

REFERENCES

1. Fall K. A delay-tolerant network architecture for challenged Internets. *Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM)*, Karlsruhe, Germany, August 2003; 27–34.
2. Scott K, Burleigh S. Bundle protocol specification. *IETF RFC5050, Experimental*, November 2007.
3. Cerf V *et al*. Delay-tolerant network architecture. *IETF RFC 4838, Informational*, April 2007.
4. DMC Imaging International. Available from: http://www.dmcii.com/.
5. Gleason S *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* 2005; **43**(6):1229–1241.
6. Ivancic W, Stewart D, Shell D, Wood L, Paulsen P, Jackson C, Hodgson D, Northam J, Bean N, Miller E, Graves M, Kurisaki L. 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. Wood L, Ivancic W, Hodgson D, Miller E, Conner B, Lynch S, Jackson C, da Silva Curiel A, Shell D, Walke J, Stewart D. Using Internet nodes and routers onboard satellites. *International Journal of Satellite Communications and Networking* 2007; **25**(2):195–216.
8. Hogie K, Criscuolo E, Parise R. Using standard Internet Protocols and applications in space. *Computer Networks (special issue on Interplanetary Internet)* 2005; **47**(5):603–650.
9. Ivancic W, Stewart D, Wood L, Jackson C, Northam J, Wilhelm J. 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. Wood L, Ivancic W, Stewart D, Northam J, Jackson C, da Silva Curiel A. IPv6 and IPsec on a satellite in space. *Conference paper B2.6.06. 58th International Astronautical Congress*, Hyderabad, India, September 2007.
11. Wood L, McKim J, Eddy WM, Ivancic W, Jackson C. Saratoga: a scalable file transfer protocol. *Draft-wood-tsvwg-saratoga-04* (Work in progress as an internet-draft), October 2009.
12. Wood L, McKim J, Eddy WM, Ivancic W, Jackson C. Using Saratoga with a bundle agent as a convergence layer for delay-tolerant networking. *Draft-wood-dtnrg-saratoga-06* (Work in progress as an internet-draft), October 2009.
13. DTN reference implementation, October 2007 release. Available from: http://www.dtnrg.org/wiki/Code.
14. Demmer M, Ott J. Delay tolerant networking TCP convergence layer protocol. *Draft-irtf-dtnrg-tcp-clayer-02* (Work in progress as an internet-draft), November 2008.
15. *UK-DMC satellite first to transfer sensor data from space using 'bundle' protocol*. Surrey Satellite Technology Ltd press release, 11 September 2008.
16. Wood L, Eddy WM, Holliday P. A bundle of problems. *IEEE Aerospace Conference*, Big Sky, Montana, March 2009.
17. Saltzer JH, Reed DP, Clark DD. End-to-end arguments in system design. *ACM Transactions on Computer Systems* 1984; **2**(4):277–288.
18. Fall K, Farrell S. DTN: an architectural retrospective. *IEEE Journal on Selected Areas in Communications* 2008; **26**(5):826–828.
19. Eddy WM, Wood L, Ivancic W. Checksum Ciphersuites for the bundle protocol. *Draft-irtf-dtnrg-bundle-checksum-05* (Work in progress as an internet-draft), October 2009.
20. Eddy WM. DTN Time Sync Issues, email to the IRTF dtn-interest mailing list, 1 April 2008, and subsequent discussion.
21. Wyatt EJ, Burleigh SC, Jones RM, Torgerson JL, Wissler SS. Disruption Tolerant Networking Flight Validation Experiment on NASA's EPOXI Mission, *The First International Conference on Advances in Satellite and Space Communications (SPACOMM 2009)*, Colmar, France, July 2009;187–196.
22. Sanders FA, Jones G, Levesque M. Transfer of files between the deep impact spacecrafts and the ground data system using CFDP: a case study. *IEEE Aerospace Conference*, Big Sky, Montana, March 2007.
23. Burleigh SC. Dynamic routing for delay-tolerant networking in space flight operations. *SpaceOps 2008, AIAA 2008-3406*, May 2008.
24. Burleigh SC. Compressed Bundle Header Encoding (CBHE). *Draft-irft-dtnrg-cbhe-04* (Work in progress as an internet-draft), February 2010.
25. Burleigh SC. First look at the deep impact DTN experiment (DINET). Presentation to the *IRTF DTN Research Group, 73rd IETF meeting*, Minneapolis, 20 November 2009; 3.
26. *NASA Tests First Deep-Space Internet*. NASA JPL press release 2008-216, 18 October 2008.
27. Burleigh SC. First look at the deep impact DTN experiment (DINET). Presentation to the *IRTF DTN Research Group, 73rd IETF Meeting*, Minneapolis, 20 November 2009; 11.
28. Jeremy C *et al*. TIME's best inventions of 2008: ♯9 The Orbital Internet. *TIME Magazine* 179(19), 10 November 2008.

AUTHORS' BIOGRAPHIES

William Ivancic (wivancic@grc.nasa.gov) is a senior research engineer at NASA's Glenn Research Center, where he directs research into hybrid satellite/terrestrial networking, space-based Internet and aeronautical Internet. Will is leading a research effort to deploy commercial-off-the-shelf (COTS) technology into NASA missions, including the Space Exploration Initiative, the Airspace Systems Program and the Aviation Safety Program. Will holds BS and MS degrees in electrical engineering.

Wesley M. Eddy (wes@mti-systems.com) is a systems engineer with MTI Systems, working on NASA projects. When this paper was written and submitted he was a network engineer with Verizon Federal Network Systems on contract at NASA's Glenn Research Center (GRC). He is co-chair of the IETF TCPM Working Group and of the IRTF's Internet Congestion Control Research Group (ICCRG). He holds an MS degree in Computer Science from Ohio University.

David Stewart (dstewart@grc.nasa.gov) is a communication engineer at Verizon. David specializes in RF and wireless communication networks. For the past eight years, David has supported the development and deployment of a mobile-router testbed at NASA Glenn, as well as deployment of early-field-trial aeronautic and maritime mobile networks. David holds a BEE degree in electrical engineering.

Lloyd Wood (L.Wood@surrey.ac.uk) is a Chartered Engineer with experience in computing, networking and aerospace. As a space initiatives manager in Cisco Systems' Global Government Solutions Group, Lloyd had responsibility for CLEO, the Cisco router in Low Earth Orbit. He spent some years contributing to the Internet Engineering Task Force and modifying IOS, Cisco's router software... so he's gone on to fly his own code in space. Working with colleagues at NASA Glenn Research Center and Surrey Satellite Technology Ltd, Lloyd achieved the first tests from space of IPv6 and of the delay-tolerant networking bundle protocol for the 'Interplanetary Internet.' He gained his PhD from the Centre for Communication Systems Research at the University of Surrey, where he researched internetworking and satellite constellations, and to where he has returned as a research fellow.

James Northam (J.Northam@sstl.co.uk) is Ground Systems Group Manager at Surrey Satellite Technology Ltd. His earlier experience with SSTL lies in imaging systems. He holds a degree in electronic engineering from the University of East Anglia.

Chris Jackson (C.Jackson@sstl.co.uk) is a principal engineer responsible for ground systems and operations at Surrey Satellite Technology Ltd. Chris has worked at SSTL for more than 14 years, where he has been involved with flight and ground software systems, space-to-ground protocols and flight operations for more than 20 space missions.