delay-tolerant protocols, it does provide an excellent visualization and recording tool for the experimenter to collect coordinated networking state for debugging and performance measurement purposes.

2. Networking Configuration

Network configuration is accomplished by synchronizing the setup and configuration files associated with the various hard and software systems comprising the testbed. We separately consider the configuration of the operating system, the configuration of the physical networking layer, and the configuration of DTN support software.

The only important piece of operating system information that must be synchronized amongst assets in the DEN is the time. Significant clock drift is both counter to the operational realities of space systems, which invest quite heavily in this regard, and prevents other aspects of the DEN support system, such as propagation delay queues, from operating. Configuration of the physical networking layer ensures that the physical components of the network are able to communicate outside of their respective organizational boundaries through a Virtual Private Network (VPN). This includes ensuring that there are not subnetwork collisions and that ports are enabled as appropriate.

The DTN support software provides several configuration mechanisms that must be synchronized to make the overlay network function, as listed in Table 4.

Table 4. The network configuration of the DEN specifies protocol, routing, and addressing infrastructure.

Configuration Item	Function
Protocol Exchange	When using the DTN communication protocols (LTP, BP) there are several configurations that must be synchronize throughout the network so that the protocols correctly interoperate. LTP configurations include messaging sizing, retransmission windows, and other timing-related configurations. BP includes synchronization on naming conventions and the interoperability and support for the extension blocks used as extensibility mechanism within the protocol.
Routing Information	The DTN suite uses a graph-based routing mechanism called Contact-Graph Routing (CGR). In this scheme transmission opportunities are decomposed into contacts and the transmission ranges associated with those contacts. These contacts and ranges must match between any pair of transmitters and receivers in the network, and must further match (or at least not run counter to) the configuration of the underlying physical network.
Naming and Addressing	Each node in the overlay network is assigned a name within the overlay network called an Endpoint Identifier (EID). This identifier may be associated with unicast, multicast, or broadcast and it is used to communicate to other nodes at the overlay. However, this overlay exists over one, or more, physical networks. The mapping between the DTN EID and the underlying networking identifier (such as an IP address or a MAC address) must be configured and consistent at each node in the network.

3. Administrative Configuration

The administrative configuration of the network is composed of two major activities: the coordinated administration of machines on the network and the management of security information.

Coordinated machine administration ensures that the machines representing networking nodes are enabled, appropriately configured, and will have the appropriate resources, including disk space, to run the experiment. If there are problems with a particular machine policies are in place to communicate this difficulty. On the DEN, individual institutions manage their own set of machines; while there is a coordination function to ensure that the various administrators communicate with one another, no single entity has administrative privileges across the network.

The physical security of the network is provided by the boundary defense mechanisms at participating institutions. The security of the network layer comes from the establishment of a VPN its associated hardware configurations. Finally, a manually-negotiated key management process exists to share keys amongst various nodes in the network allowing the security mechanisms that operate at the overlay to function.

V. Results

The DEN, as the evolving reference implementation of the SSI testbed, has already been used for testing of DTN-related software and operational concepts. This section outlines the software that has been tested on the DEN, demonstrates a typical configuration of the network, and discusses lessons learned from these efforts.

A. Major Functional Testing

While the DEN has hosted a variety of individual tests during its construction, there have been four major functional tests that have tested not only a software subsystem but the overall testbed infrastructure: BP interoperability, video streaming, and file synchronization.

1. Initial connectivity and interoperability

The first set of tests conducted across the multi-center DEN network used a combination of the NASA reference implementation of DTN protocols for spaceflight software (ION) and the dtnrg-provided DTN2 reference implementation to verify both IP and BP-layer connectivity. This ensured that the various centers' VPN connections were functioning properly and that the firewalls were configured to allow Bundle Protocol traffic. Several issues were uncovered by even this simple test, including differences in the TCP convergence layer implementations and issues related to interoperability of naming schemes. Both of these issues were addressed by later versions of the ION and DTN2 implementations.

2. Bundle Authentication Block (BAB) security testing

This test validated that various nodes in the network, especially those that operated across administrative boundaries, were able to authenticate using the Bundle Authentication Block (BAB) from the BSP suite. The BAB is one of the fundamental security mechanisms in the DTN architecture and provides hop-by-hop authentication of traffic. The test demonstrated that a user without the correct BAB authentication key was unable to inject traffic across the emulated space link. This ensures that 'rogue' users, even if they have access to the ground network, cannot mount a denial-of-service attack against the space link by sending 'junk' traffic to a spacecraft. As with the initial connectivity tests, several interoperability issues between the ION and DTN2 implementations were uncovered and fixed during the course of the testing.

3. Video Streaming

The DEN has also been used to demonstrate the concept of delay-tolerant video streaming over BP. While the concept of streaming video over a high-delay, disrupted link seems counter-intuitive, the concept provides mission operations a valuable service: the ability to view a real-time stream tolerating drop-outs *and* to go back and evaluate a more-complete version of the stream at a later time. The ability to "rewind" a video stream allows operators to view retransmitted frames that would otherwise have been lost in a less-tolerant network. A data repository at the video stream receiver re-constructs streams from frames based on their construction timestamp, and the transmitting video source may offload all reliability concerns to the underlying network. The DEN testing demonstrated the viability of this approach using the DTN protocols and configurations. Notably, the delay-tolerant nature of the transaction allows the "library" of streaming video to continue to accumulate well after the initial video streaming occurred.

4. Long-term File Synchronization

This test demonstrated the ability to perform rsync-style file synchronization across multiple nodes in the DEN with simulated link disruptions. The test was run for multiple months during which periodic underlying directory and file changes, and the performance of both the synchronization utility and the ION suite were reviewed. This test demonstrated the manner in which multiple administrators interact to ensure that DEN machines performed as necessary for long-term tests.

B. Generalized Test Setup

Nodes in the DEN run the ION software and communicate over the terrestrial Internet using a VPN. Certain NASA centers house nodes running on flight hardware and other centers/organizations run different Linux distributions natively and on virtual machines.

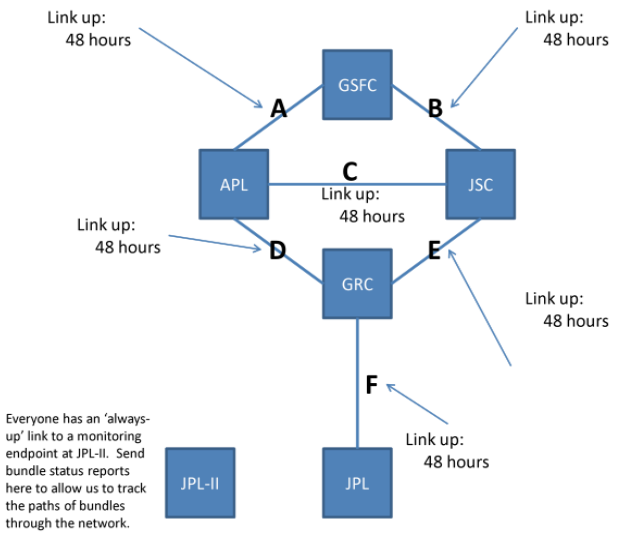

Figure 3. A typical configuration of the DEN involves multiple organizations.

```
#Global Contact Plan

#JPL : 21216
#APL : 21403
#GSFC: 21302
#GRC : 21111
#JSC : 21610

a range   +1 +604800 21212 21111 1
a range   +1 +604800 21111 21610 1
a range   +1 +604800 21610 21302 1
a range   +1 +604800 21302 21403 1
a range   +1 +604800 21403 21111 1

a contact +1 +604800 21212 21111 100000
a contact +1 +604800 21111 21212 100000
a contact +1 +604800 21111 21610 100000
a contact +1 +604800 21610 21111 100000
a contact +1 +604800 21610 21302 100000
a contact +1 +604800 21302 21610 100000
a contact +1 +604800 21302 21403 100000
a contact +1 +604800 21403 21302 100000
a contact +1 +604800 21403 21111 100000
a contact +1 +604800 21111 21403 100000
a contact +1 +604800 21403 21610 100000
a contact +1 +604800 21610 21403 100000
```

Figure 2. Contact plans specify overlay contacts over time.

The network graph illustrated in Fig 2. shows how, at the overlay level, each center is physically connected to at least one other center, but also requires bundle forwarding to at least one other center. Contact graph routing is able to test the routing of bundles between each center, and can also include disruption of links to show how bundles are routed when a direct connection is no longer present, or how bundles are stored when there is not even an indirect route to be used. Each node is configured with a global routing configuration file that specifies the contact times for each node on the network, an example of such a file is given in Fig. 3. The global configuration file sets up the routing table for the network and is essential for when there are multi-hops from node to node, or for when delays and disruptions are introduced into the system. Along with a global configuration file, each node must implement a unique configuration that includes the means for communicating with the other nodes. This file includes turning on or off various features, such as security, and provides the means for the nodes to be able to communicate with one another.

Testing involves either leaving each link to run continually without delay/disruption, or introducing link impairments. For example, the topology specified in Fig. 2 and 3 were in support of the file synchronization test. During this test each center ran DTN protocols for 48 hours and sent messages to each respective center every 15 minutes. At the end of the test, the message traffic was analyzed and graphed to see the bandwidth of messages within the system. A common version of the ION suite (in the case of this test, version 2.4.1 was used) ran at each node. Once each center had properly configured their node, each node was started at 12:00 AM EST by a script running on the machines. During testing, the point of contact at each center monitored the node to watch for any node failures. If the event of a failure, the point of contact restarted the node as appropriate. The initial connectivity test allowed for the restarting of nodes, and a failure presented a good opportunity to simulate the event of a delay within the network. Each center configures their own nodes, which emulates organizations managing the configuration of their own node on the network.

C. Lessons Learned

The initial set of DEN tests provided multiple insights into the greater challenges that will be faced during deployment of DTN on systems, specifically in the realms of network administration and competing administrative domains. The greatest challenges, and lessons learned, are in the non-trivial task of resource management – both human and machine.

For the initial testing, each link was left to run continually without delays or disruptions; however, once delays are introduced a new problem surfaced in the form of timing synchronization. To properly implement contact periods designed during a test, network time must be precise and shared amongst all nodes in the network. However, differing organizations, especially those supporting their own local time servers, encountered enough drift and jitter to provide problems in the network synchronization. Some nodes at come organizations were not

e

connected to any time server and experienced drift over weeks and months of testing. To solve this problem a Network Time Protocol (NTP) server must be housed at one of the DEN nodes from which the nodes synchronize time to in order to start the nodes at a certain time.

A logistics hurdle in assembling all components of the DEN involved assembling points of contact (POC) from each center that would be participating in various tests and scheduling their time concurrently todiscuss the setup and administration of the network. This included juggling both very busy schedules and time zone differences. As more nodes are added to the DEN, and the levels of tests increase, knowing how to manage the personnel running the tests is an important task that can very easily be overlooked. Once all of the personnel resources were assembled, it became apparent that there needed to be a full-time administrator of the DEN at one center which coordinated the efforts of all others. The DEN coordinator then became the global network administrator. This eased the challenge of having each center attempt to create global network configuration files; instead, the DEN administrator would send out the files for each center to load onto their respective machines. The concept of truly distributed network administration, absent a coordinator role, is not recommended based on our experience.

Fault detection and recovery across administrative boundaries provided a different set of challenges. When nodes within a particular organization experienced faults, especially at waypoint nodes, other centers typically discovered the fault first. However, lacking administrative permissions within a different administrative domain, the fault detecting centers could not perform diagnostic analysis. During these outage times, the network available waited for administrators from organizations housing troubled nodes to complete their anomaly resolution processes. This proved to be a bottleneck to testing because of time zone differences and loading of the administration staff. In our experience, having an automatic notification go to a point of contacts e-mail list would help to prevent the lag of checking on the periodic, manual checking of nodes health.

VI. Roadmap

NASA is currently designing experiments to support DTN development in FY12—FY16. The experiments in this timeframe will focus on evaluating the performance of the protocol stack (apps/BP/LTP) in configurations representative of current missions (e.g. communication with the International Space Station, single-hop communication with spacecraft) as well as near-future missions such as 2-hop relay communication. These tests will evaluate whether current routing mechanisms can scale to accommodate realistic contact lists and rates of bundle forwarding when using flight-like hardware. The large-scale tests will also be used to experiment with and further refine the Delay-Tolerant Network Management mechanisms under development.

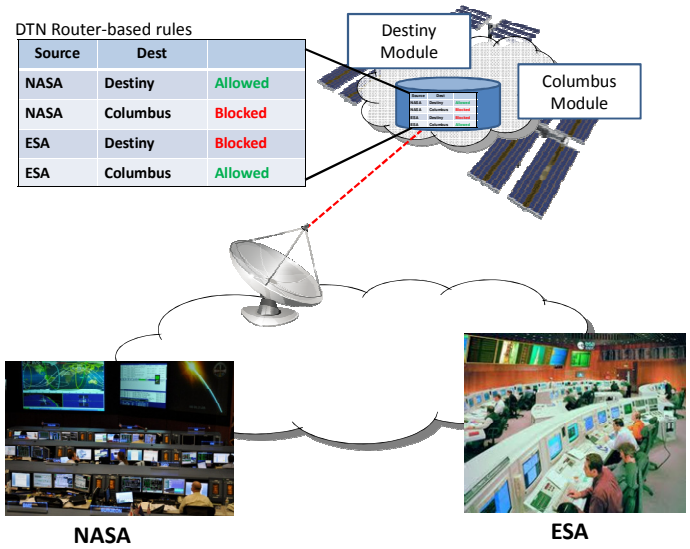

Figure 4. Future test scenarios will incorporate fine-grained security and flight assets, such as the ISS.

NASA will also use the DEN to prototype and demonstrate capabilities being considered for experiments and/or use on the International Space Station. These will include a prototype 'border gateway router' for ISS to allow a single point of contact with the ground and compression of the custody acknowledgements that form the basis for DTN reliability.

In preparation for more extensive international testing and increased use of networked communications, the team is also experimenting with router-based security control measures such as access control / firewall capabilities. These will ensure that policy rules can be imposed to limit the forwarding of certain traffic in the network. Fig. 4 shows an example of a set of router-based firewall rules to prevent unintended commanding of other agencies assets. In the figure, bundles with source endpoints within NASA can only be routed to the Destiny module on board ISS, and bundles with source endpoints in ESA can only be routed to the Columbus module. Such a capability is complementary to the BAB security mechanisms; the BAB keeps unauthenticated traffic out of the network, while router-based security

will keep authenticated traffic from inadvertently being routed contrary to policy. This capability will help assure operators that networked communications can be implemented without necessarily having to check the *content* of every uplinked bundle. Obviously allowing particular forwarding operations under such a policy requires that the consequences of the allowed bundles do not impact overall operation.

VII. Conclusion

The construction of a to-scale testbed for the SSI involves the simulation or emulation of a variety of operational characteristics, including individual link-layer effects, overlay network configurations, and administrative boundary conditions. Constructing a cohesive, controlled, deterministic testbed that addresses each of these areas with the necessary fidelity is a non-trivial endeavor that has been undertaken by NASA. To date, the DEN represents the initial implementation of such a testbed, and it has been used to provide concepts relating to file synchronization and video streaming over DTN protocols; the DEN has also provided interoperability testing for portions of the BP security suite. NASA will continue to evolve the DEN into the fully envisioned SSI testbed by incorporating additional operational hardware and software, including flight assets such as the ISS.

We have described the system-level objectives of the SSI testbed, described the mechanisms for the construction, test execution, and anticipated evolution of the DEN, and listed several lessons learned from this initial construction and test phase. Ultimately, the DEN provides a unique capability for validating DTN concepts in the space environment. Specifically, the ability to increase the TRL of flight software by incorporating flight-like hardware in the current DEN and flight assets in future evolutions of the DEN, provide adopting missions with the metrics necessary to baseline space internetworking models.

Acknowledgments

The authors would like to thank the management and technical members of the NASA Space DTN Project for their hard work and dedication in the construction, maintenance, and evolution of the DEN.

References

[1] Cerf, V., et al. "Delay-Tolerant Networking Architecture", *RFC4838*, April 2007.

[2] Mankins, J. C. 1995, “Technology Readiness Levels”, White Paper, Advanced Concepts Office, Office of Space Access and Technology, NASA, available at: http://www.hq.nasa.gov/office/codeq/trl/trl.pdf (Seen: 2012-04-29).

[3] Brent Chun, David Culler, Timothy Roscoe, Andy Bavier, Larry Peterson, Mike Wawrzoniak, and Mic Bowman. 2003. PlanetLab: an overlay testbed for broad-coverage services. *SIGCOMM Comput. Commun. Rev.* 33, 3 (July 2003), 3-12. DOI=10.1145/956993.956995 http://doi.acm.org/10.1145/956993.956995

[4] GENI. Global environment for network innovations. http: //www.geni.net. Seen: 2012-04-29.

[5] J. Wyatt, et al., "Disruption Tolerant Networking Flight Validation Experiment on NASA's EPOXI Mission," Advances in Satellite and Space Comm., Intl. Conference on, pp. 187-196, 2009 First Intl. Conference on Advances in Satellite and Space Comm., 2009

[6] S. Burleigh, K. Scott, Bundle Protocol Specification, November 2007, http://tools.ietf.org/html/rfc5050

[7] Ramadas, M., and Burleigh, S., and S. Farrell, Licklider Transmission Protocol – Specification, RFC5326, September 2008.

[8] S. Burleigh, Contact Graph Routing: draft- burleigh-dtnrg-cgr-01, July 2010, http://tools.ietf.org/html/draft-burleigh-dtnrg-cgr-01.

[9] J. Segui, E. Jennings, S. Burleigh, “Enhancing Contact Graph Routing for Delay Tolerant Space Networking”, IEEE GLOBECOM 2011.

[10] Symington, S. and Farrell, S. and Weiss, H, and P. Lovell, Bundle Security Protocol Specification, RFC6257, May 2011.