

Enabling the Interplanetary Internet

*Christopher J. Krupiarz, Edward J. Birrane, Benjamin W. Ballard,
Lotfi Benmohamed, Alan A. Mick, Katherine A. Stambaugh,
and Edward W. Tunstel*

With more complex and numerous missions on the horizon, NASA and its international spacefaring partners are transitioning from dedicated point-to-point communication links to a network-based system. Unfortunately, terrestrial network protocols used on the Internet have limited use in the space environment because of environmental constraints such as light-time delays, transmission disruption, and planetary alignment. NASA is addressing this problem through research into delay- and disruption-tolerant networking (DTN). DTN protocols are specifically constructed to account for the challenged communications networks used in space exploration. In this article, we describe the work we are performing alongside our colleagues at numerous NASA centers to develop DTN technology. This includes APL efforts in researching and deploying DTN network management and routing techniques, combining DTN protocols with onboard storage systems for spacecraft, and using DTN as the underlying network infrastructure for robotic telepresence.

INTRODUCTION

Throughout the era of robotic space exploration, mission designers have traditionally relied on heavily managed and dedicated point-to-point links for communication between spacecraft and ground systems. However, in recent years, the number and complexity of these missions have increased dramatically, thereby placing

an increased burden on an already-stressed communication system. To address this issue, NASA, the European Space Agency (ESA), and other space organizations are transitioning from reliance on these heritage systems to autonomous, network-centric, packet-switched networks similar to the terrestrial Internet. Such systems

have shown great promise. For example, NASA's Mars Exploration Rovers successfully demonstrated this first-generation architecture by transmitting more than 95% of their data through the orbiters Mars Odyssey and Mars Global Surveyor and ESA's Mars Express orbiter.¹ This architecture consists of a relay capability where spacecraft can rely on data to be stored at intermediate "hops" and take different paths back to Earth, similar to how the Internet operates on Earth. Unfortunately, the protocols Internet applications use are not always suitable for the unique environment of space. Conditions such as long light-time delays, intermittent communications, asymmetric communication, and planetary alignment contribute to reducing the effectiveness of terrestrial protocols or render them unusable. To address this, APL is working with NASA to develop and field a set of efficient next-generation network protocols to create an Interplanetary Internet. This architecture enables full interoperability with international missions at Mars, the Moon, and other locations in the solar system. In addition, the protocol suite reduces costs and operations workload, increases science return, and enables new event-driven science missions by automatically adjusting for the challenging space communication environment.

APL'S PAST CONTRIBUTIONS TO SPACE NETWORKING

Past APL space missions have made extensive use of networking technology developed in coordination with the Consultative Committee on Space Data Systems (CCSDS). APL's most recent operational contribution to furthering space networking is the use of the CCSDS File Delivery Protocol (CFDP) on its MESSENGER (MErcury Surface, Space ENvironment, GEOchemistry, and Ranging) spacecraft.² At its launch, MESSENGER was the first deep-space mission to use CFDP, a protocol that is similar to the file transfer protocol (FTP) and enables transmission of files between the ground and spacecraft systems. This effort is notable because it is one of the first times a space protocol autonomously "closed the loop" between a space-borne asset and a mission operations center when transmitting mission-critical science data files. This removed the need for manual intervention by the mission operations team and increased retransmission speed of the data.

After the CFDP development, NASA's Mars Technology Program selected APL to lead the multiyear Mars Next Generation Protocol Suite Project.³ APL built upon its CFDP experience by joining with the NASA Jet Propulsion Laboratory (JPL) to create a cross-domain team including flight and ground software engineers, mission operators, and telecommunications and network specialists to define the next step in Mars network protocol technology. The team forecasted the needs of

the Mars network for the next decade, identified current and future network protocols to meet those needs, modeled and simulated the proposed architecture, and implemented flight-ready versions of the new protocols. The architecture identified for a next-generation Mars relay infrastructure includes protocols currently in use on the Mars Exploration Rovers as well as two developing standards that fall under the umbrella of delay- and disruption-tolerant networking (DTN).⁴ DTN, a major research area in computer networks, is an evolution from an earlier Interplanetary Internet study,⁵ and the technology targets "challenged networks," of which the space environment is a particular instantiation. The two protocols that form the basis for DTN are the Licklider Transmission Protocol (LTP)⁶ and the Bundle Protocol (BP).⁷ LTP is a long-haul protocol that provides a reliable service across a deep-space link. Evolved from the retransmission procedures of the CFDP, LTP ensures that data sent from a spacecraft to the ground or vice versa are received correctly without the additional overhead of manual verification of the data. BP is a standard method for performing store-and-forward data transmission where data are stored for a period of time at intermediate nodes along a network path. This is essential for a space network because typically data from a rover or lander are not transmitted in a real-time relay link. Instead the orbiter must store data until they can be transmitted. In a space relay environment, BP and LTP would be used in conjunction with each other. The flight software would create a bundle of data received from a landed asset, store it for later transmission, and then use LTP to ensure its delivery to Earth. Figure 1 depicts scenarios where incorporation of these protocols into the current architecture would be beneficial.

Using Fig. 1 as a reference, Table 1 lists some of the advantages of using DTN at Mars or other planetary bodies requiring relay communication.

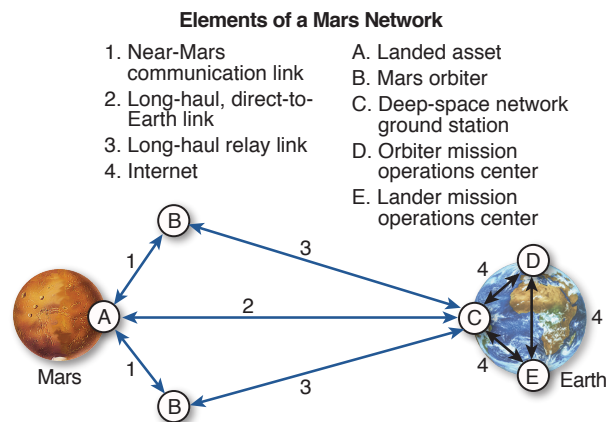


Figure 1. Communication with Mars relies heavily on relay communication.

Table 1. Multiple advantages of using DTN at Mars.

Current Mars Architecture	Deficiency in Approach	Benefit of DTN Protocols
Data lost in the transmission from Mars are manually determined and retransmitted. (See link 3 and assets D and E in Fig. 1.)	Complete receipt of data is not guaranteed until data are processed on the ground.	Mission operations overhead costs are reduced because retransmission does not require manual intervention. The delay in the receipt of data by scientists and operations is removed.
Data are not prioritized across instruments or missions on the orbiter. (See asset B in Fig. 1.)	High-priority data wait in a queue to allow lower-priority data to be sent. Relay data take precedence over orbiter data.	Higher-priority data take precedence on the down-link. Priorities can be applied across instruments and spacecraft to enable one piece of software to handle all incoming data streams. Data can be prioritized across spacecraft, instruments, and coordinated observations.
Protocol implementations are proprietary. (See assets A, B, and C in Fig. 1.)	Each NASA mission must develop its own protocol software to interact with the Mars network.	Regardless of the developer of the spacecraft, NASA missions have a robust and reusable set of software for the underlying communication protocols, thus reducing software development costs.
Lander data are sized to match pass. (See asset A in Fig. 1.)	If a portion of data is lost in transmit, those data have to be repackaged for an alternate path.	Pieces of a data product can be sent via multiple orbiters. This allows for outages or missed paths and does not require preplanning to match data product size with path length.
Data from Mars missions external to relay spacecraft flow to the orbiting spacecraft's mission operations center. (See assets C and F and link 4 in Fig. 1.)	The chance of data loss due to hardware failure increases. The approach requires that all relay orbiter locations develop software to process lander data.	Software development costs for individual orbiter missions are reduced. Responsibility for return of lander data no longer belongs to individual orbiter mission operations teams. Instead, data are routed before the lander data reach the orbiter data center.
Retransmission procedures are written in the application code. (See assets A, C, and F in Fig. 1.)	New applications must contain retransmission procedures for lost data.	Software costs are reduced because the application program does not implement retransmission procedures. This simplifies application development and permits new applications to easily make use of the underlying network infrastructure.
Standards are not used across the system.	Software development costs are increased.	Design costs are reduced because new missions can design to a particular standard. NASA missions can use international assets to return data.

Upon completion of the Mars communication investigations, NASA expanded its research into DTN by creating the Space DTN Readiness Project. On this project, APL is teaming with multiple NASA centers including JPL, Goddard Space Flight Center, Marshall Space Flight Center, Glenn Research Center, and Johnson Space Flight Center as well as industry (MITRE and Raytheon BBN Technologies) and academia (University of Colorado at Boulder and University of Maryland, Baltimore County). APL's efforts are focused in two broad areas: DTN research and DTN applications. The DTN research area includes investigating methods for DTN network management (NM), routing, and security. For applications, APL is examining infusing DTN technology into its SmartSSR and robotic telepresence efforts as well as supplying software for JPL's experiments involving the Deep Impact spacecraft. We discuss these efforts in detail in the following sections.

APL'S SPACE DTN RESEARCH

Network Management

For a network to be operationally deployed it must support robust and timely management functions. The characteristics of DTN, however, present significant challenges to providing management tools to the network operator. This section describes ongoing research at APL that defines the NM function for DTN and outlines the systems engineering tasks that must be adopted when implementing interplanetary networks.⁸ APL is the lead on NM for the Space DTN Readiness Project.

The Challenges of Managing "Tolerant" Networks

Space internetworks utilize the BP as the primary data protocol for high-delay networks. BP implements an overlay network that relies on a convergence layer

to glue it to whatever lower-layer protocol stack exists at a network node. Because the protocols at individual nodes in the network differ, the BP may sit atop a transport layer at one node and atop a link layer protocol at another node. However, NM of the BP overlay includes these integrating convergence layers. This lack of clear protocol separation complicates the systems engineering and high-level architectural decomposition of an NM function. Figure 2 illustrates this model, where at the “orbiter control center” DTN protocols exist at the session layer atop the Transmission Control Protocol (TCP), at the transport layer atop LTP, and at the network layer atop Proximity-1.

APL works with multiple NASA centers to define the NM function in the context of space systems. This approach involves bottom-up implementations for existing space systems such as the International Space Station (ISS) and Deep Impact as well as a top-down system analysis of necessary requirements, protocols, and architectures for future deployments such as the next generation of Mars missions in the 2016 time frame. As part of our top-down system work, APL writes specifications and software prototypes for the CCSDS and NASA.

Redefining NM

The most prolific NM model in use today is that provided by the Internet Protocol (IP). In this model, data collectors exist at each node in the network where they locally populate databases and respond to queries for their data from some network operations center. The standard for defining the local database format is the management information base (MIB), and the most popular NM tools for “pulling” management data are the Network Configuration Protocol (NETCONF)⁹ and the Simple Network Management Protocol (SNMP).¹⁰ These tools exploit low-latency data communications to periodically query information from the nodes and collect information in a relatively small number of information sinks within the network. This information typically feeds some visualization of the network

state and provides input to monitoring tools that trigger alarms based on predefined rules (such as data value threshold crossings). These models rely on the ability to discover nodes, negotiate link parameters, and establish communication sessions by using control and data flows through the network. These characteristics do not scale as signal propagation delays or link disruptions increase. At interplanetary distances, signal propagation delays may grow larger than the line-of-sight contact window. For example, the one-way light time between Earth and Mars can range up to 20 min. Contact opportunities lasting less than 40 min cannot exchange a single round-trip datagram. Likewise, when the frequency and duration of link disruption (bad pointing, low energy, or jamming) prevent protocol session establishment, no data cross the link because protocol overhead dominates network traffic. When lost or stale data include health and configuration information, the network manager is unable to effectively mitigate failure cascades. Therefore, any “pull-data-to-central-server” approach cannot react to errors in the network rapidly enough to prevent catastrophic failures or maintain critical quality of service (QoS) requirements.

APL redefines the responsibilities of DTN NM as follows: receive and apply configurations, monitor the local and regional network state, and autonomously configure the behavior of the node. This “intelligent push” model revolves around the configuration of nodes in the network: what data the node collects, how it conditions these data, when these data should be communicated externally of the node, and when the local configuration should be proactively changed by the node itself. This model is similar to the use of MIBs for collecting management data but is extended to include testing new configurations and configuring network fault-protection algorithms at each node. Data conditioning may be as simple as packaging direct local measurements, calculating averages, medians, or delta functions, or performing more complex transformations. Autonomous behavior determines when to send what level of information as

well as how to switch between alternate configurations at the node. In the absence of round-trip communications, NM must include preconfiguration and prescribed actions for far-reaching portions of the network.

A Universal NM System

A management system that operates in the absence of Internet networking assumptions (link negotiation, node discovery, and bidirectional communication) applies to multiple mission types, includ-

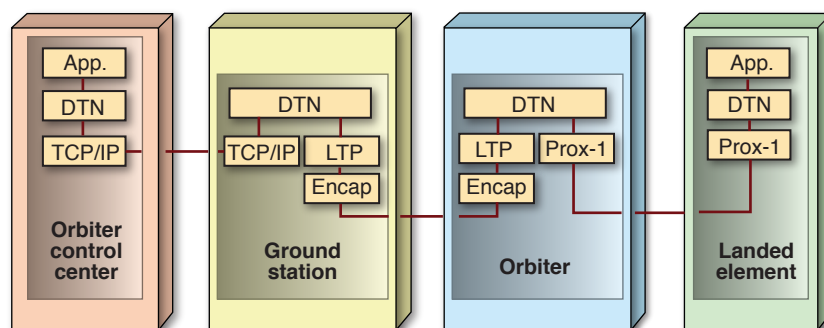


Figure 2. DTN protocols provide a homogenizing “overlay” network for space links.¹¹ App., application; Encap, CCSDS encapsulation packet; Prox-1, Proximity-1.

ing deep space, unmanned underwater vehicles, and unmanned aerial vehicles. Therefore, the DTN NM System (DTNMS) being developed by APL is mission agnostic. Our DTNMS takes a relative view of the term “delay.” We define two important latencies governing management in DTN: the transmission latency between the time a critical datum is produced and consumed and the action latency between the occurrence of an event and a management response to that event. The transmission latency captures both network-use-invariant delays (such as signal propagation delays) and use-driven delays (such as congestion, queuing, and backpressure). The action latency is, intuitively, greater than the transmission latency of the data upon which an action is predicated—we cannot act upon data that have not been received. Quantifying action latency lets the network designer compare the ability to respond to an expressed performance metric for the operational system. When the action latency for a network region exceeds a performance measurement, the region is considered “high delay.” This approach is far more helpful than arbitrary definitions of latency (e.g., 10 ms is low latency and 8 min is high latency).

Because DTN covers networks of both large delay and large disruption, our initial systems engineering task involved determining whether the solutions to these problems overlap. Initial analysis suggests that solving the propagation delay problem inherently solves the link disruption problem and in a scalable manner. For high-delay links, we then examine the transmission latency to determine whether the dominating latency is delay (propagation) or disruption (congestion, backpressure, and queuing). Solutions that ignore delay and solve disruption do so by waiting for the link state to stabilize and reestablish traffic after flushing queues. This, however, assumes that the link state can be inferred and reestablished before the link is disrupted again. Field tests show that when delay is introduced, IP networks experience thrashing in recently restored links. Management solutions that consider delay use an intelligent push model of open-loop control that is based on available information. By not requiring closed-loop, query–response mechanisms and session reestablishment, the management data flow immediately upon link restoration with significantly less thrashing in the network. This approach distributes the NM task across the network by applying autonomy and fault-protection activities at each network node.

An artifact of the distributed DTNMS approach is that the design of the NM function is less influenced by the physical characteristics of the network (data rates, number of nodes, network diameter) and more influenced by the boundaries between high- and low-latency regions. For example, as inferred from Fig. 2, an Earth–Mars network could support three network regions: planetary landers, planetary orbiters, and Earth

stations. Earth stations (ground station and orbiter control center) would be connected via low-latency links and use IPs and human-in-the-loop network operation centers. Planetary orbiters at Mars would be on the far side of high-delay links driven by large-propagation delays and significant link disruptions based on line-of-sight geometries. The NM of these orbiters would be a hybrid approach of human-in-the-loop control from mission control centers and autonomous fault-protection services. Landers on the Martian surface would create a low-latency ground network, perhaps utilizing 802.11 protocols. However, they would also require an autonomous function because they are on the far side of a high-delay link back to Earth. The concept of autonomy and fault-protection services running on landers and orbiters is not new.

Where low-latency and high-latency network regions merge, a gateway is required to translate management protocols and the NM concept of operations (CONOPS). As such, a network region will have its own network manager (some human-in-the-loop and some completely automated). Figure 3 illustrates such a push/pull gateway between a DTN network region and an IP network. In this case, the gateway accumulates pushed data and populates associated SNMP MIBs that may then be pulled from the IP network.

The decision to fully decentralize NM operations to each node versus grouping this capability into relatively few nodes at regional boundaries is part of the engineering of the DTN.

The proposed DTNMS is the result of a bottom-up reuse of appropriate existing technologies combined with top-down review of requirements, scalability, and constraints associated with high-delay environments. We identify four focus areas, illustrated in Fig. 4, that guide the development of a DTNMS: system architecture, data definition and usage, application architecture, and tool implementation.

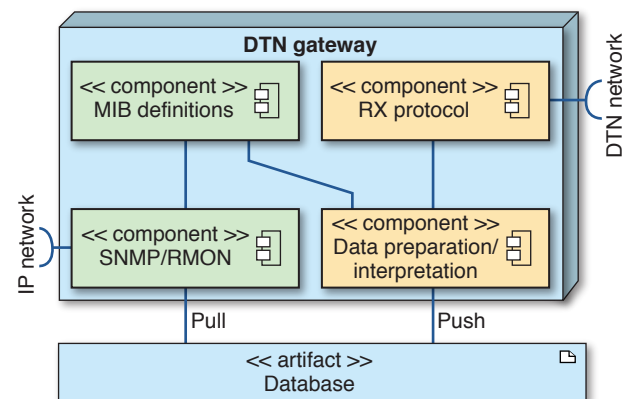


Figure 3. DTN gateways provide an interface between high-latency and low-latency network regions. RMON, remote network monitoring; RX, receiver.

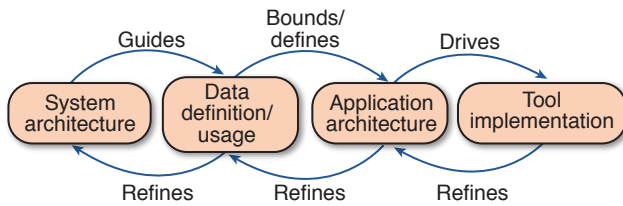


Figure 4. DTNMSs are designed in the context of four critical focus areas.

System Architecture

System architecture incorporates the concepts of the universal NM system presented earlier in this article. By focusing on open-loop control, preconfiguration of assets, and autonomy/fault-protection mechanisms, the NM system is distributed across regions of the network. Although the specific implementations of autonomous NM in DTN remain an open research area, our contribution has been to establish the system architecture within which these algorithms operate. Figure 5 illustrates a proposed architecture for management nodes that focuses on the need to standardize methods of data representation (including NM configurations), fault models that detect errors in applied configuration or changes to the network topology or link schedule, and handlers that apply node-specific actions to regional network events. Our standardization of this NM approach through the CCSDS encourages space agencies such as NASA, ESA, and the Japanese Aerospace Exploration

Agency to construct systems that are compatible at the management and operations layer.

Data Definition and Usage

Data definition and usage describes both the recommended standard formatting used for the internal representation of NM data and the protocols used to exchange NM data. This is conceptually similar to the SNMP/MIB construct but augmented to support more complex data relationships, conditioning/aggregation rules, coherency checks, and autonomy configuration. We replace the term MIB with the term application data model (ADM) and define ADMs for the various protocols envisioned to be active within an interplanetary DTN. Work within the CCSDS specifies the protocols used to exchange both NM configuration and health information. APL leverages experience with engine-driven models of autonomy and telemetry production to provide condensed, efficient representations of these data suitable for implementation within a spacecraft command and data handling system, including the ability to update subsets of ADMs, condensing complex expression to binary representations, and separating data and metadata [such as data tag, identifier (ID), units, and conversion rates].

Application Architectures

Application architectures provide a template for implementing the DTNMS concept within a flight command and data handling system that operates with

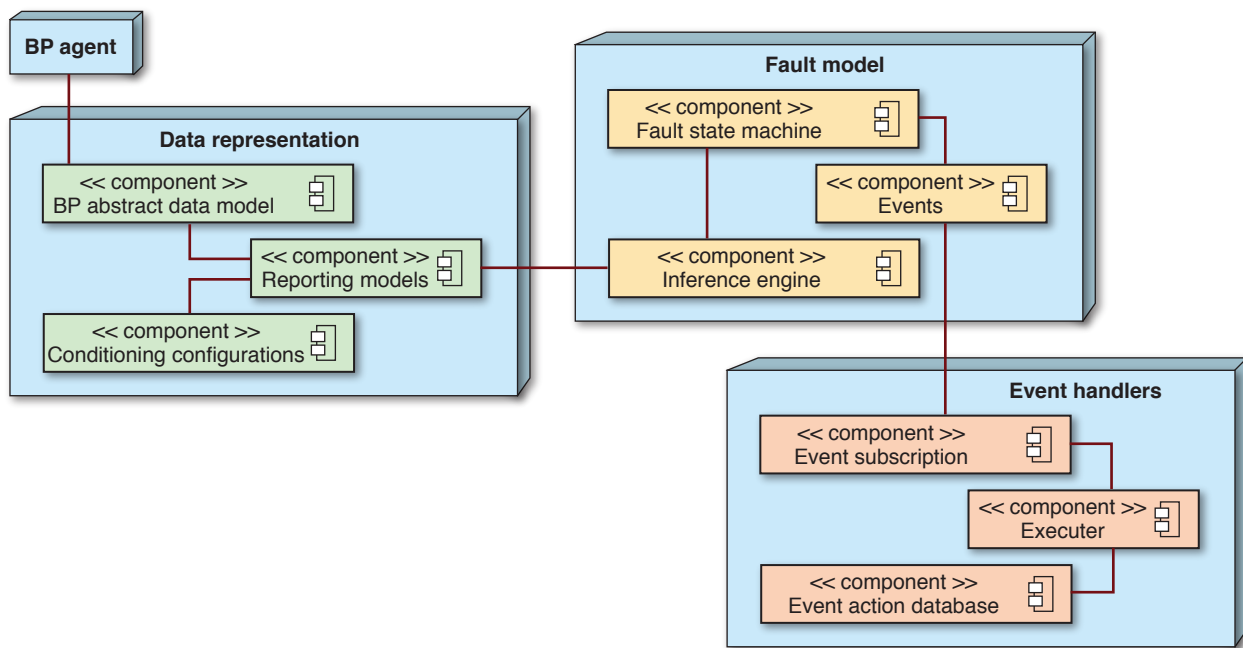


Figure 5. A DTNMS comprises mechanisms for data representation, fault models, and local event handlers.

low power, low central processing unit margin, and low bandwidth constraints. APL builds complex autonomous systems for deep-space missions, including New Horizons, STEREO (Solar TERrestrial RELations Observatory), and MESSENGER. We leverage this experience when designing a recommended application architecture for agencies to adopt in support of the conceptual universal DTNMS. This architecture, shown in Fig. 6, is derived from a flight autonomy subsystem and provides a first-level decomposition of the standardized roles and responsibilities of the DTNMS function.

We recommend two logical stores of data at each DTNMS node: one to hold configuration data and one to hold NM data. Centralized data collection, in the form of ADMs, provides shared storage for a variety of monitoring and reporting tools across the platform. For example, if a BP agent determines that available storage is exhausted because of queuing due to loss of connectivity to a downstream node, the local routing application may wish to remove itself as a waypoint to that node. This computation is simplified by a central database holding ADMs for both the BP agent and the routing application. Further, as ADMs are maintained and versioned over time, it is possible that those two DTN nodes in different states of upgrade may need to exchange information. A centralized data formatting and aggregation function removed from individual applications the onus of handling ADM versioning translations. Versioning and version tracking will become increasingly important with the increase in the size of operational DTN and associated delays.

Telemetry collection refers to the process by which the local node accumulates data into the database either

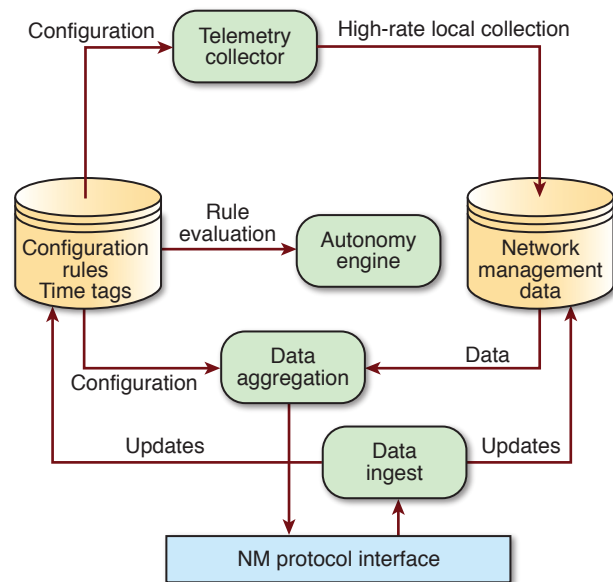


Figure 6. APL’s DTN NM application architecture leverages autonomous spacecraft subsystems experience.

“just in time” in the context of an NM tool execution or, at least, in the background as part of a (high-rate) periodic task. This provides a natural bound on the staleness of ADM data, which is necessary when the data are evaluated in the context of fault protection, performance, and aggregation schemes.

Data aggregation occurs in the context of a management tool that takes local, high-rate collected data and conditions them for transmission over a DTN link. Aggregation implies the use of aggregate functions to summarize high-rate, locally collected data for reduced bandwidth utilization. In addition to aggregation, however, remote data may support other mechanisms to alter the verbosity of the NM data payload. For example, given low-priority, low-bandwidth links, a standard NM payload would be a very compact data sample. As errors at a node are collected, the tool may choose to aggregate over smaller periods or otherwise add additional information to the payload for a larger, but more informative, NM payload.

Tool Implementation

The tools used to visualize, configure, and archive management information differ on the basis of the requirements of a particular network. For example, standard Internet tools such as PING and TRACEROUTE may have no use in high-delay networks (such as between orbiters and ground stations) but may be useful in low-delay networks (such as between planetary rovers or Earth ground stations). Rather than create a finite list of management tools, we recommend a common platform

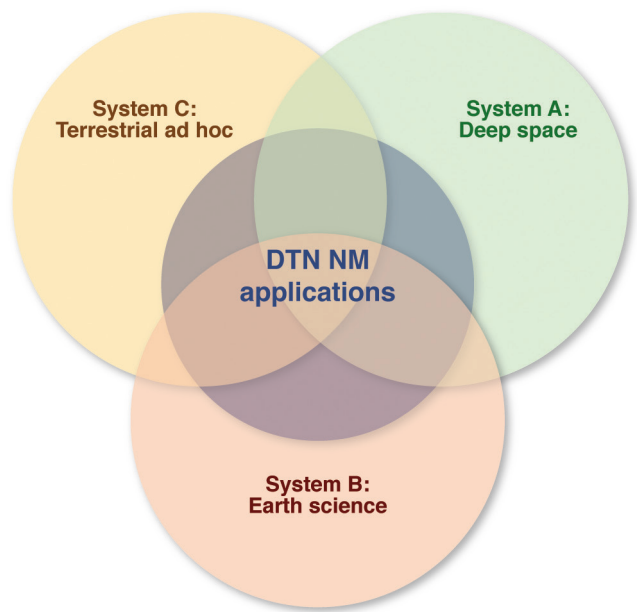


Figure 7. An NM application comprises a subset of tools available to the network engineer.

for interoperability and standardization through the system architecture, data definition/usage, and application architecture work. Assuming such a common platform, a large repository of tools may be evolved over time and applied to individual engineered networks as illustrated in Fig. 7. This approach is especially important for the Interplanetary Internet concept where multiple national space agencies will perform management tool development in the contexts of their own missions while seeking coordination and reuse from others.

Routing

As mentioned above, DTN refers to a wide range of challenged networks where links are intermittent and end-to-end connections cannot be assumed to exist. In such networks, DTN routing automatically handles the forwarding of DTN bundles among bundle agents. Note that the BP specification provides the framework for bundle forwarding but does not define an algorithm for routing. The protocol can be configured to use static routes; however, automated dynamic routing provided by a DTN routing protocol has the advantages of increased scalability and reduced operations and management cost, as well as self-organization and the ability to operate in ad hoc settings.

Routing in the space environment is accomplished by different DTN routing approaches in different parts of the interplanetary network. Indeed, a space DTN network spans multiple regions and covers different bodies such as Earth, the Moon, and Mars. Given the wide range of operating conditions, different region-specific routing protocols are needed. The region-specific protocols optimize the bundle layer and enable better performance in different network deployments, for instance, in the interplanetary region as well as in different planetary regions. Because existing approaches to DTN routing are focused on single routing domains, we explore in this section the protocol enhancements needed to achieve multidomain DTN routing.

Consider a space interplanetary network interconnecting a number of planetary networks as shown in Fig. 8. In the interplanetary part, because of the orbit-

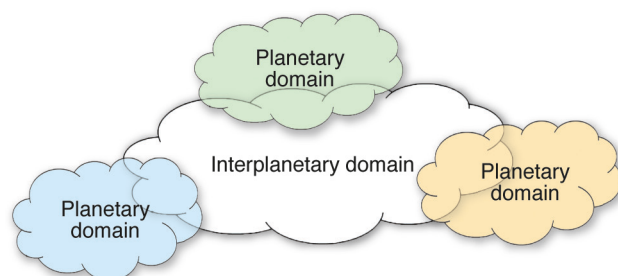


Figure 8. Three planetary network regions and an interplanetary network region.

ing of planets, the link disruption is due to intermittent connectivity with schedules derived from knowledge of planetary trajectories. In this case we can assume that connectivity in the form of pairwise node contacts is available over some time horizon, and a DTN routing protocol that is based on scheduled connectivity would be suitable for interplanetary links.¹² Contact graph routing (CGR) is one such protocol.¹³ CGR leverages the fact that contact information between nodes in space can be predetermined, and it computes routes to deliver data over these scheduled communication contacts that represent future transmission opportunities. Each contact has an associated record of parameters that include the duration of the contact, its link bandwidth, and the approximate distance between nodes on both ends of the contact.

In contrast to the interplanetary region, unscheduled events are behind the disruption in most planetary networks and are mainly due to mobility of the planetary nodes (for example, rovers or astronauts), which is often not subject to predetermined schedules. Therefore, for communication among mobiles on a planet, routing approaches tailored to unscheduled/probabilistic links are more appropriate. There are a number of proposals in this category including Delay Tolerant Link State Routing (DT-LSR),¹⁴ Probabilistic Routing Protocol using History of Encounters and Transitivity (PRoPHET),¹⁵ and MaxProp.¹⁶ These protocols differ with respect to bandwidth and storage requirements as well as forwarding operation in terms of degree of message replication. DT-LSR does not replicate messages and leverages predictions of future link availability as a metric within a link state routing solution. Epidemic routing is based on spreading messages in the network to increase the chances of successful delivery when a route becomes available. MaxProp replicates as many copies of a message as possible with the available resources. However, on the basis of the idea that the mobility in real networks is not completely random, PRoPHET uses an estimation of the delivery probability to select appropriate forwarding nodes without spreading to all encountered nodes.

Because the networks of different domains (corresponding to each planetary region as well as the interplanetary one) use different approaches for routing internally, we need to define the interdomain routing procedure. Border nodes are those nodes on the domain boundary that participate in both planetary and interplanetary routing. These nodes are involved in the interdomain procedure, which includes aggregation for interdomain information exchange, border node selection when multiple nodes are available, and accounting for end-to-end QoS/performance from the per-domain contributions.

To discuss the solution that we propose to address these issues, consider a scenario where DTN routing is based on CGR within the interplanetary domain and

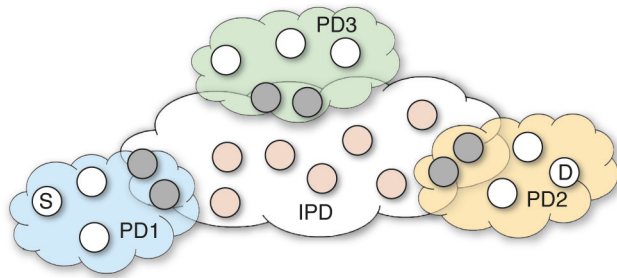


Figure 9. Planetary domains (PD1–PD3) with nodes running DT-LSR, an interplanetary domain (IPD) with nodes running CGR, and border nodes (gray) running both protocols. S, source node; D, destination node (see text for details).

DT-LSR in each of the planetary domains (PD1–PD3) as shown in Fig. 9.

Nodes in the CGR domain know about all node IDs within the domain. In particular, they know about all border nodes and the set of destinations reachable through each one of them. This set is composed of all the IDs of nodes in the planetary domain with which the border node interfaces. This information should be part of the contact record for all contacts that involve a border node. In order to limit the size of these contact records, IDs of nodes within one planetary domain should be assigned in such a way that they can be aggregated, ideally into one aggregate per domain to achieve a strong identifier summary. One approach is to use a (region_id, node_id) naming, which is based on a hierarchical structure, with a corresponding aggregate of (region_id,*) as a summary per region.

For nodes in a planetary domain, each node knows about all IDs within the domain through DT-LSR's exchange of link state advertisements (LSAs), and traffic to any destination outside the domain is handled by one of the domain's border nodes. Each border node is known to all nodes in the domain by the LSA it generates, which indicates that it is a border node and thus can be used as a “default” egress node to reach any destination outside the domain. Note that nodes on a domain boundary act as border nodes that participate in both CGR and DT-LSR. They are configured to handle scheduled contacts with CGR and all other contacts with DT-LSR.

In summary, the routing of a given DTN bundle proceeds as follows: For a bundle that originates at node S in planetary domain PD1, is destined to node D in planetary domain PD2, and must travel through the interplanetary domain (see Fig. 9), each of the three domains involved makes a routing decision. In domain PD1, from the structure of the ID allocation the source node S knows that the destination D is outside of its domain when D has a different region ID and that it

therefore needs to route the bundle to one of the domain border nodes. When there is more than one border node, the simplest approach is to select one according to some metric internal to the domain. A more appropriate approach requires determining the edge-to-edge paths in the interplanetary segment having the best QoS and injecting them through each DT-LSR domain, allowing the selection of the most appropriate border node for use in reaching a destination. Because the interplanetary segment is expected to dominate the end-to-end path, this second approach results in more efficient path selection. In this case each border node maintains the current best path and associated metric such as latency to reach each of the other domains and inserts this information in its LSAs. Note that CGR can be augmented to perform this edge-to-edge QoS assessment. Once the bundle gets to the appropriate border node in PD1, we use CGR to forward to the destination border node in PD2 through the interplanetary segment of the end-to-end path. For the third and last segment (within domain PD2), once the bundle gets to the egress border node, it simply gets delivered by the local DTN routing mechanism (DT-LSR in this case) to its destination D.

APL'S SPACE DTN APPLICATIONS

SmartSSR

As we have seen, in the context of a network subject to delays and reconfigurations, standard routing protocols are incapable of effectively establishing complete routes. These standard protocols try to first establish a complete end-to-end route before forwarding the data. In a network that lacks continuous connectivity, a “store-and-forward” approach is more effective. Data are incrementally moved and stored throughout the network in advance of establishing a complete end-to-end route. This suggests that DTN routers should be coupled with the capability to store large amounts of data. For space applications, this is the approach taken by the SmartSSR DTN router.

The SmartSSR is a solid-state recorder (SSR) that is being developed under an Independent Research and Development program by APL specifically for space applications. This component combines a large NAND (Negated AND) flash array with a general-purpose processor in order to host several closely related functions in a way that efficiently provides their services to the other components of the spacecraft. It is based on the LEON3FT processor and uses SpaceWire interfaces to communicate with other spacecraft components. Figure 10 displays the SmartSSR prototype development board.

The SmartSSR manages the spacecraft data through the Space Flash File System (SpaceFFS). The SpaceFFS

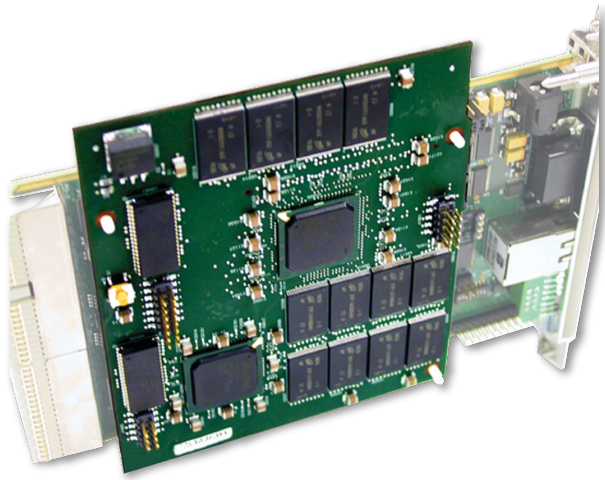


Figure 10. The SmartSSR prototype board.

is APL's adaptation of a commonly available flash file system to the special requirements of the space operational environment. The DTN router capability, as well as several other capabilities, is provided by JPL's interplanetary overlay network (ION) implementation of the standard DTN protocols. Combined, the SmartSSR DTN router is capable of providing the following:

- A high-capacity, high-performance, solid-state data recorder that serves as a local area (spacecraft) network file system server

- Delay-tolerant routing that is based on the ION implementation of the DTN BP
- CFDP file transfers and file management using the ION CFDP
- A remote gateway for publish/subscribe distributed computing using the ION asynchronous message service protocol

The JPL ION implementation of the standard DTN protocols is a fundamental component for the DTN routing capability to APL's SmartSSR as shown in Fig. 11. The ION software is implemented using a layered approach that exploits standard software interfaces. The interplanetary communication infrastructure layer provides common functionality to other ION packages through general-purpose libraries. A key component is the "zero copy objects" capability, which, in turn, is used to build linked lists of data objects representing the data and tasks that need to be performed. Subsets of these are included in the "traffic database," which is built on the simple data recorder (SDR) component. The SDR provides a "persistent object" store and a transaction-based "checkpoint" capability.

The ION's DTN traffic database is built on the SDR component. Because the SDR is an abstraction based on the common properties of a SSR, this component forms the link between the SmartSSR and the ION software. By using the SpaceFFS as the underlying backing store for the SDR component, the ION software's DTN routing capability is easily integrated into the

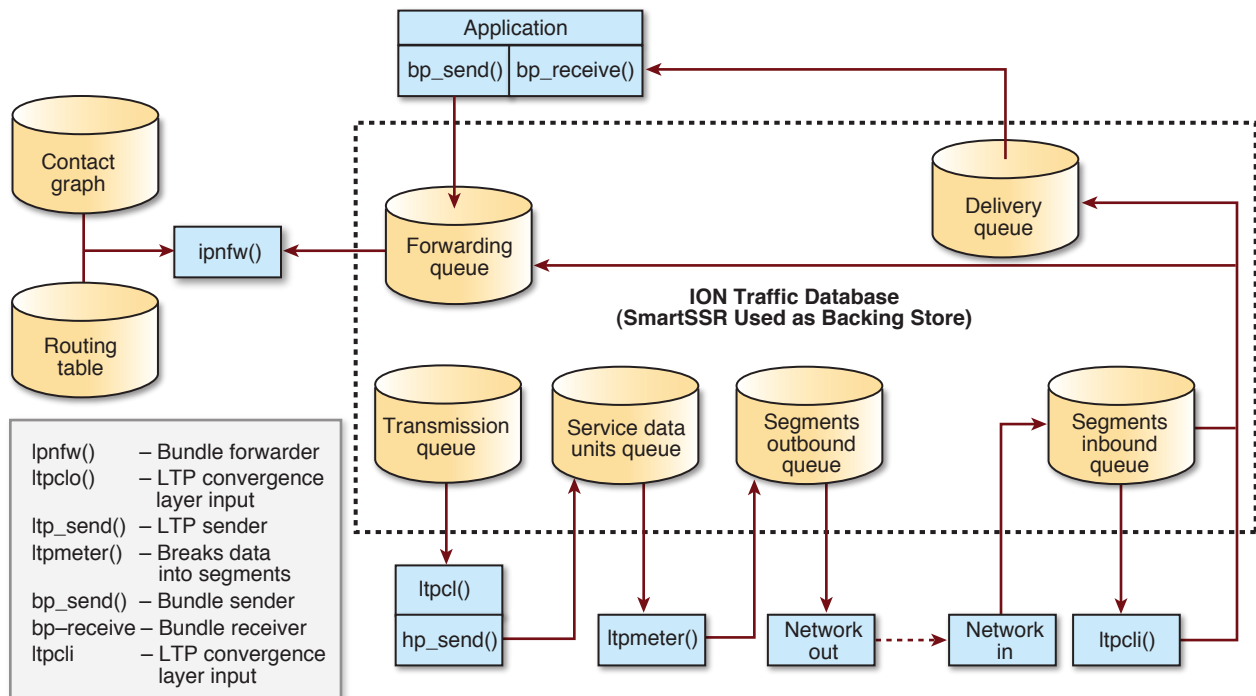


Figure 11. ION serves as the mechanism for DTN operations.

SmartSSR. This forms the basis for DTN capability on any spacecraft that incorporates the SmartSSR as its data recorder.

Telepresence

Future missions will call for astronaut teleoperation of robotic systems on planetary surfaces from orbiting spacecraft as well as human teleoperation of surface robotic systems on the Moon from Earth. The impact of DTN on such telerobotic missions is of strong interest; particularly interesting is determining what it would take to achieve end-to-end communications and what limitations or advantages can be expected from DTN. Telepresence, or the degree to which a human operator feels present in the remote environment of the robot being teleoperated, may be a mode of operation for the mission. The feeling of being present in the remote environment is facilitated by feedback to the operator of sensory data including not only visual images of the robot's local environment but forces, torques, and other tactile sensations representing what the robot is experiencing. The degree to which immersive telepresence is desired or can be achieved depends on various factors including the mission CONOPS, the distance between the operator and the robot, and communications time. DTN can play a role throughout the end-to-end communications network or in certain segments of the end-to-end network.

One way to test DTN in telepresence applications is by leveraging work done for the Defense Advanced Research Projects Agency (DARPA) Revolutionizing Prosthetics Program, which has many collaborators across APL. That work has been leveraged toward advancing teleoperation technology for human capability projection through dexterous robotics with application to military explosive ordnance disposal. Two prototype robotic arms developed under the DARPA program were integrated with a torso and stereoscopic vision atop a mobile robotic vehicle to create a teleoperated Dexterous Robotic Platform for advanced explosive ordnance disposal. The same telerobotics and telepresence technology is applicable and extensible to space mission scenarios. To study end-to-end operations involving telerobotics for a different application in the very different environment of future space missions, the Dexterous Robotic Platform was configured to operate using DTN protocols and infrastructure. Specifically, we configured the Dexterous Robotic Platform, including immersive telepresence and teleoperation capability, for communication over a DTN link to perform a suite of geologic sample acquisition and sample handling tasks while exercising various DTN link attributes. The robotic system consists of a bimanual configuration of anthropomorphic robot arms with a stereoscopic head, all mounted on a mobile robotic base. Its teleoperation



Figure 12. Remote teleoperations enables robotic replication of user movements.

interface consists of an arm motion tracker for setting arm positions and a CyberGlove for setting wrist and hand poses. A head tracker, which is connected to two webcams and pan/tilt motors on the head of the Dexterous Robotic Platform, also provides the user with stereo vision feedback. Figure 12 shows the setup for this experiment.

The DTN configuration investigated for this effort introduces the DTN protocols on the command link between the operator and the robotic platform using the ION protocol suite, the same software used for the SmartSSR Project. Figure 13 shows the data path (outer lines) and the physical path in place for the demonstration. The DTN portion of the network consists of three DTN nodes: A, B, and C. Node A receives user datagram protocol packets from the controller and transfers the data into DTN bundles. These bundles are transferred to node B via DTN and then on to node C. Once on node C, the software extracts the bundle data and places them in a user datagram protocol packet destined for the robotic platform. Additional software on node B provides the ability to introduce a variable network time delay to simulate communication signal transmission

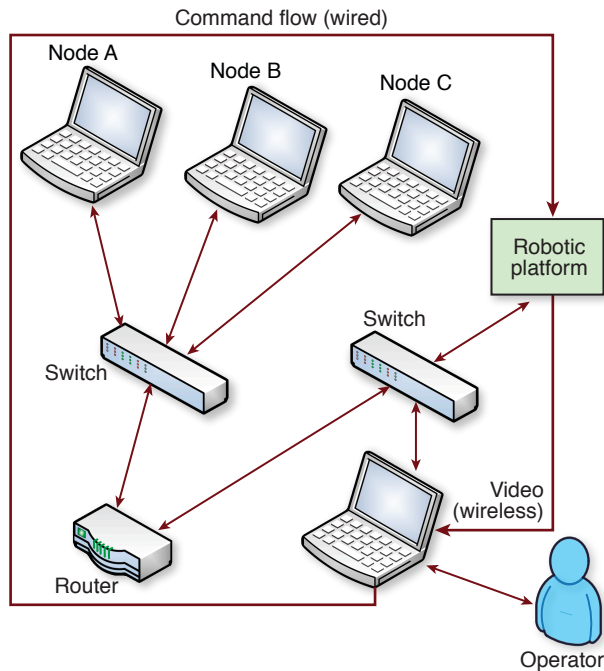


Figure 13. DTN command flow and physical data path used for the DTN/robotics demonstration.

over various distances. The control software generates packets at a rate of 20 Hz. Video feedback on the operation is returned on a separate wireless link.

A number of scenarios or CONOPS could be adopted to explore the advantages and limitations of DTN for telerobotic space exploration missions. The trade space includes the robot location, operator location, mode of operation, robot task, and communications time delay or other network characteristics. For initial cooperative work, we focus on scenarios where the Dexterous Robotic Platform performs visual inspection and dexterous handling of rock and soil samples. For each task, we collected metrics for three CONOPS scenarios beginning with a baseline of no added delay. The second scenario introduced a delay of 0.5 s to represent robotic control between Earth orbit and the surface of Earth. The final scenario mimicked teleoperation from Earth of a robotic system on the lunar surface by introducing a delay of 2 s. The nature of our demonstration is relevant to studies that are gaining interest and funding both within NASA and abroad.

The ESA has started planning the proposal of a flight experiment and test bed called METERON (Mars End-To-End Robot Operations Network). METERON aims to pave the way for human exploration of Mars (and the Moon) by demonstrating immersive telepresence in microgravity between the ISS and Earth-based teleoperators. NASA is teaming up with ESA to support the METERON Program, and APL could potentially serve as one of the DTN surface nodes used for preparation work and testing. Similar activities not directly focused

on DTN include the Canadian Space Agency's Avatar Project involving ISS astronauts teleoperating robots on Earth and NASA's plan to evolve humanoid robotic capability in space by flying its Robonaut 2 system on the ISS. To assess how telerobotics can improve efficiency, productivity, and science return for human exploration missions, more technology demonstration missions between Earth and the ISS as well as between Earth and the lunar surface will be necessary in the near future.

CONCLUSION

APL is part of a NASA-wide team that is defining a future Interplanetary Internet using protocols that fall under the umbrella of DTN. The project is focused on extending space network technology to future missions as well as terrestrial applications. Over the coming years, this will include more flight demonstrations as well as research into NM, routing, security, and flight software. Additionally, APL is working with mission operations and scientists both within and outside APL to define how DTN can increase data return, reduce mission costs, and enable new mission concepts.

REFERENCES

- ¹NASA Jet Propulsion Laboratory, *Mars Exploration Rovers*, <http://www.jpl.nasa.gov/multimedia/mer-yr5/> (accessed 23 July 2010).
- ²Krupiarz, C. J., Burleigh, S. C., Frangos, C. M., Heggestad, B. K., Holland, D. B., et al., "The Use of the CCSDS File Delivery Protocol on MESSENGER," in *Proc. 7th International Conf. on Space Operations (SpaceOps 2002)*, Houston, TX, paper AIAA-2002-T5-35 (2002).
- ³Krupiarz, C. J., Jennings, E. H., Pang, J. N., Schoolcraft, J. B., Segui, J. S., and Torgerson, J. L., "Spacecraft Data and Relay Management Using Delay Tolerant Networking," in *Proc. 9th International Conf. on Space Operations (SpaceOps 2006)*, Rome, Italy, paper AIAA-2006-5754 (2006).
- ⁴Burleigh, S. C., Fall, K., Cerf, V. G., Durst, R. C., Scott, K., et al., "Delay Tolerant Networking—An Approach to Interplanetary Internet," *IEEE Commun.*, pp. 128–136 (2003).
- ⁵Cerf, V., Burleigh, S., Hooke, A., Torgerson, L., Durst, R., et al., *Delay-Tolerant Network Architecture*, Internet Engineering Task Force Request for Comments 4838, <http://tools.ietf.org/tools/rfcmarkup/rfcmarkup.cgi?rfc=4838> (April 2007).
- ⁶Ramadas, M., Burleigh, S., and Farrell, S., *Licklider Transmission Protocol—Specification*, Internet Engineering Task Force Request for Comments 5326, <http://tools.ietf.org/html/rfc5326> (Sept 2008).
- ⁷Scott, K., and Burleigh, S., *Bundle Protocol Specification*, Internet Engineering Task Force Request for Comments 5050, <http://tools.ietf.org/html/rfc5050> (Nov 2007).
- ⁸Consultative Committee for Space Data Systems, *Rationale, Scenarios, and Requirements for DTN in Space*, Informational Report CCSDS 734.0-G-1 (2010).
- ⁹Enns, R., *NETCONF Configuration Protocol*, Internet Engineering Task Force Request for Comments 4741, <https://tools.ietf.org/html/rfc4741> (Dec 2006).
- ¹⁰Levi, D., Meyer, P., and Stewart, B., *Simple Network Management Protocol (SNMP) Applications*, Internet Engineering Task Force Request for Comments 3413, <http://tools.ietf.org/html/rfc3413> (Dec 2002).
- ¹¹Birrane, E. J., and Cole, R. G., "Network Management of Disruption-Tolerant Networks: A Systems Engineering Approach," in *Proc. 11th International Conf. on Space Operations (SpaceOps 2010)*, Huntsville, AL, paper AIAA-2010-2206 (2010).
- ¹²Merugu, S., Ammar, M., and Zegura, E., *Space-Time Routing in Wireless Networks with Predictable Mobility*, Georgia Institute of Technology Technical Report GIT-CC-04-07, pp. 1–13 (2006).

¹³Burleigh, S., "Dynamic Routing for Delay-Tolerant Networking in Space Flight Operations," in *Proc. 10th International Conf. on Space Operations (SpaceOps 2008)*, Heidelberg, Germany, paper AIAA-2008-3406 (2008).
¹⁴Demmer, M., and Fall, K., "DTLSR: Delay Tolerant Routing for Developing Regions," in *Proc. ACM SIGCOMM Workshop on Networked Systems for Developing Regions (NSDR)*, Kyoto, Japan, pp. 1-6 (2007).

¹⁵Lindgren, A., Doria, A., and Schelen, O., "Probabilistic Routing in Intermittently Connected Networks," *Mob. Comput. Commun. Rev.* 7(3), 19-20, 2003.
¹⁶Burgess, J., Gallagher, B., Jensen, D., and Levine, B. N., "MaxProp: Routing for Vehicle-based Disruption-Tolerant Networks," in *Proc. 25th Conf. on Computer Communications (INFOCOM 2006)*, Barcelona, Spain, pp. 1-11 (2006).

The Authors

Christopher J. Krupiarz is a member of APL's Principal Professional Staff in the Space Department. He initially led APL's DTN efforts and now provides consultation on flight software applications for DTN architectures. **Edward J. Birrane** is a member the Senior Professional Staff in the Space Department. He is the overall lead for APL's research in DTN for space applications. **Benjamin W. Ballard** is a member of the Principal Professional Staff in the Space Department.

He consults with APL scientists and mission operations personnel as well as external sponsors to define applications for DTN. **Lotfi Benmohamed** is a member of the Senior Professional Staff in the Milton S. Eisenhower Research Center. He leads APL's effort in DTN routing. **Alan A. Mick** is a member of the Senior Professional Staff in the Space Department. He leads the SmartSSR effort. **Katherine A. Stambaugh** is a member of the Associate Professional Staff in the Space Department. She led the DTN telepresence demonstration project.

Edward W. Tunstel is a member of the Senior Professional Staff in the Space Department. He is the lead on DTN/robotics interfaces. For further information on the work reported here, contact Christopher Krupiarz. His e-mail address is christopher.krupiarz@jhuapl.edu.



Christopher J. Krupiarz



Edward J. Birrane



Benjamin W. Ballard



Lotfi Benmohamed



Alan A. Mick



Katherine A. Stambaugh



Edward W. Tunstel