TELEMETRY IN BUNDLES: DELAY-TOLERANT NETWORKING FOR DELAY-CHALLENGED APPLICATIONS

Scott Burleigh
NASA Jet Propulsion Laboratory, California Institute of Technology

ABSTRACT

Delay-tolerant networking (DTN) is a system for constructing automated data networks in which end-to-end communication is reliable despite low data rates, possible sustained interruptions in connectivity, and potentially high signal propagation latency. As such it promises to provide an inexpensive and robust medium for returning telemetry from research vehicles in environments that provide meager support for communications: deep space, the surface of Mars, the poles or the sub-Arctic steppes of Earth, and others.

This paper presents an overview of DTN concepts, including "bundles" and the Bundling overlay protocol. One possible scenario for the application of DTN to a telemetry return problem is described, and there is a brief discussion of the current state of DTN technology development.

KEYWORDS

Delay, latency, intermittent connectivity, reliable communication, Bundling.

INTRODUCTION

Over the past two decades, Internet communication technology and (in many cases) the Internet itself have enabled rapid growth in the scope and speed of telemetry return from research and industrial equipment. So cost-effective are these capabilities that Internet connectivity is increasingly seen as vital to successful deployment of some kinds of remotely monitored devices (see for example [1]).

However, the Internet protocols are built on a design paradigm of continuous and rapid end-to-end data interchange. They cannot readily be extended into deployment realms in which this paradigm does not hold, due to intermittent connectivity, very low data rates, and/or very long signal propagation delays. Such realms, from which inexpensive and reliable telemetry return may be highly desirable, include deep space, the surfaces of other planets, and extremely isolated terrestrial research sites such as the polar regions and many of the sub-Arctic areas of Alaska, Canada, Scandinavia, and Russia. At the extreme, such as the extended mission of the Deep Space 1 spacecraft, a deep space probe might be in radio contact with Earth for only a single 8-hour period each week, transmitting at only 20 kbps over a distance of 16 light minutes; round-trip communication latency in this scenario would vary from 32 minutes at the beginning of the tracking pass to 160 hours in the final minutes of the pass.

Alternative communication protocols have been developed for these environments, but in many cases those specialized protocols are conversely unsuitable for use within the Internet, where researchers' computers reside. Client software products that implement the protocols may not be widely available or commercially supported, and the protocols themselves may omit security and congestion control mechanisms that are mandatory in the richly connected and heavily used Internet.

Remote research and industrial operations in these configurations might therefore benefit from a single generalized communication architecture that could insulate end-to-end, device-to-researcher application software – and its users – from these variations in protocol capability and constraint. Delay-Tolerant Networking is intended to provide that benefit.

The rationale for Delay-Tolerant Networking and the design concepts of the DTN architecture are discussed in detail in [2]. The balance of this paper first summarizes that rationale and those concepts, then discusses the application of Delay-Tolerant Networking to a telemetry return problem.

RATIONALE

The unsuitability of the Internet protocols in environments that are not characterized by continuous and rapid end-to-end data interchange is most apparent when we consider (a) reliable transport and (b) routing.

Many network applications require reliable data delivery. While forward error correction can reduce data loss substantially, it offers no protection against sustained link loss; some kind of automated repeat request (ARQ) service is required for maximum reliability. The Internet's TCP transport protocol provides just such a service, but over long and intermittent links its performance will typically be unsatisfactory:

- Establishment of a TCP connection normally entails at least one round trip exchange between client and server. Over short-duration, long-latency link episodes, connection establishment may consume most or all of the communication opportunity.

- TCP delivers received data to the application only in transmission order on any single connection, so any loss of data will delay delivery of not only the lost packets but also all subsequently transmitted packets that were not lost.

- The throughput of TCP itself diminishes with increasing round-trip latency due to the manner in which TCP responds to data loss and handles network congestion [3].

- TCP retransmission is end-to-end. Since retransmission is possible only when the sender retains a copy of all transmitted data until receipt of acknowledgements from the receiver, the length of time that this transmitted data must be retained – and therefore, if transmission is continuous, the aggregate <u>amount</u> of storage space allocated to these retransmission buffers –

increases as round-trip communication time increases. For telemetry sources with limited local storage resources, this effect may render reliable communication impossible.

The Internet routing system poses additional problems. Routing packets through the Internet is made possible by a family of concurrently operating protocols that continuously compute available routes from each device in the system. The global Border Gateway Protocol (BGP) is based on TCP, so the difficulties posed by TCP afflict it as well. The local routing protocols operating within Autonomous Systems rely on regular reachability reports from agents; transient connectivity losses may delay these reports, causing permanent connectivity losses to be incorrectly inferred. A sequence of transient connectivity losses on multiple segments of an end-to-end path may even be incorrectly interpreted as complete loss of connectivity on the path.

In short, the Internet protocols work very well within the Internet and other Internet-like environments (such as local area networks deployed at remote research sites), but they work far less well in environments characterized by large message exchange latencies – whether those latencies are due to signal propagation delay, transient connectivity loss, or both. This makes them unsuitable for end-to-end communication over paths that include one or more links within such high-latency environments.

DESIGN CONCEPTS

Delay-Tolerant Networking architecture was developed for just such paths. The basic idea is to install a new overlay network protocol at a layer above the transport protocols that work best in each environment, and to expose only that overlay protocol to applications. The overlay protocol utilizes the capabilities of the underlying transport systems as far as possible, bridging between them at the boundaries between communication environments. This enables it to offer applications an end-to-end data transmission service that is both reliable and efficient.

Because the overlay protocol must operate end-to-end, spanning both Internet-like and high-latency environments, its design cannot rely on any end-to-end expectation of:

- continuous connectivity
- low or constant transmission latency
- low error rate
- low congestion
- high transmission rate
- symmetrical data rates
- common name or address expression syntax or semantics
- data arrival in transmission order

Yet for optimum end-to-end performance it must be able to take advantage of any of these favorable circumstances that are present, wherever they are present. The DTN architecture therefore is built on the following fundamental principles:

1. A <u>postal model of communications</u>. Modern communications media tend to be telephonic and conversational in nature because they are based on continuous electronic data exchange over relatively short distances at the speed of light. Long-latency communication takes us back to pre-telephony, to the communication methods and techniques suitable for postal exchange. Absent conversational interchange, each message should constitute a self-contained, atomic unit of work; the receiver's potential questions regarding each message should be anticipated by the sender, and the answers should be "bundled" with the original message. For this reason, the DTN overlay network protocol is named *Bundling* and the messages conveyed via this protocol are termed "bundles".

2. <u>Tiered functionality</u>. The transport protocols designed for each environment already exploit whatever favorable circumstances the environments offer, while operating within their constraints. Bundling does not attempt to replicate all of this functionality; it instead does only the minimum necessary to bridge between environment-specific transport systems.

3. <u>Terseness</u>. Although data rates may be high in some environments, they may be very low in others. It's safest to avoid wasting bandwidth wherever possible.

Informed by these principles, the DTN architecture includes the following structural elements:

- <u>Tiered forwarding</u>. A DTN *region* is a set of DTN communication entities (*nodes*) that can mutually exchange bundles using a common environment-specific transport system; a delay-tolerant network is simply a set of concatenated regions. Each node is addressable by the concatenation of (a) the region ID of the region it resides in and (b) a regional endpoint ID that is meaningful only to the regional transport system. Regional network protocols are used to accomplish bundle forwarding within each region traversed by the end-to-end path from sender to receiver; the Bundling protocol encompasses this transmission in end-to-end forwarding that crosses region boundaries at *gateway* nodes.

- <u>Deferred transmission</u>. Because connectivity may be intermittent, a Bundling node may be unable to forward a newly received bundle to the next gateway (or to the final destination) at the moment of reception. Nodes must be prepared to store received bundles, preferably in some non-volatile medium, until an opportunity to transmit them arises.

- <u>Tiered routing</u>. DTN relies on regional routing protocols for route computation in support of intra-regional forwarding. End-to-end forwarding is subject to the occurrence of connectivity episodes (*contacts*) which may be continuous, on-demand, scheduled, predicted, or wholly opportunistic; new DTN routing protocols are used to develop the necessary contact information at forwarding nodes.

- <u>Tiered ARQ</u>. Regional transport systems are expected to assure reliable point-to-point communication between nodes, effectively advancing the point of retransmission toward the destination at each region boundary to avoid the performance problems inherent in long-latency end-to-end ARQ. In the event of data loss at a retransmission point, a "safety net" ARQ system built into Bundling – *custody transfer* – may optionally be invoked.

- Tiered security. Unauthorized use of DTN infrastructure is minimized by authentication of the bundle sender upon each bundle reception. End-to-end application data confidentiality and integrity are left to the application, with some built-in support from Bundling.

- Tiered congestion avoidance. Regional transport systems are expected to use regionally appropriate measures to control regional congestion. In addition, Bundling detects DTN congestion (local depletion of the resources needed for bundle storage and forwarding) and responds to it by invoking tiered flow control.

- Tiered flow control. Regional flow control may be protocol-based (in the Internet) or managed and rate-based (in deep space). Bundling may exert additional flow control by refusing to accept custody of selected bundles.

- Tiered coding. Regional protocols may encode data at one or more layers of the stack to optimize link utilization. Additionally, Bundling's interfaces to regional transport systems may compress bundle headers in ways that are regionally appropriate.

- Tiered fragmentation and reassembly. Regional protocols are expected to fragment bundles as necessary for regional transmission, e.g., to comply with MTU constraints. Bundling itself may subdivide bundle payloads in recognition of contact duration limits.

- Resilient delivery. A destination application entity (task, process, thread) may not be running at the time a bundle destined for it arrives. In this event, Bundling may retain the bundle locally until the destination entity resumes operation (*deferred delivery*) or may even restart the destination entity itself so that bundle delivery can be completed (*reanimation*).

- Postal service levels. In recognition of the success of more traditional communication systems based on the postal model – postal systems – Bundling similarly offers three postal service levels (low, standard, and high) and three postal notification services: notice of initial transmission, notice of delivery to destination application entity, and report of route taken.

APPLICATION

As an example of the use of DTN technology for telemetry return, consider the following operational scenario (Figure 1).

Connectivity. Two rovers and a stationary lander on the surface of Mars communicate among themselves using an 802.11b-based wireless local area network. The lander also has a UHF transceiver that is sufficiently powerful for communication with two low-orbiting relay satellites that pass overhead at infrequent but predictable intervals. The relay satellites, in addition to their UHF transceivers, have X-band radios for communication with a tracking station on the surface of Earth, which is between 4 and 20 light-minutes distant. The tracking station has an Internet interface, as does the workstation used by the scientist who operates the rovers.

Protocols. The lander, orbiters, tracking station, and workstation all operate as Bundling communication nodes; application software on the workstation issues and receives bundles via the workstation's own Bundling engine, while application software on the rovers uses an RPC-based application programming interface to issue and send bundles via the Bundling engine on the lander. The RPCs exchanged between the lander and rovers over the wireless LAN are conveyed by TCP/IP. The lander and orbiters communicate via the reliable CCSDS Proximity-1 protocol [4], underneath Bundling. The orbiters communicate with the tracking station via the CCSDS AOS protocol [5] with supporting link-layer ARQ, underneath Bundling. Finally, the bundle traffic between the tracking station and workstation is carried by TCP/IP on the Internet.

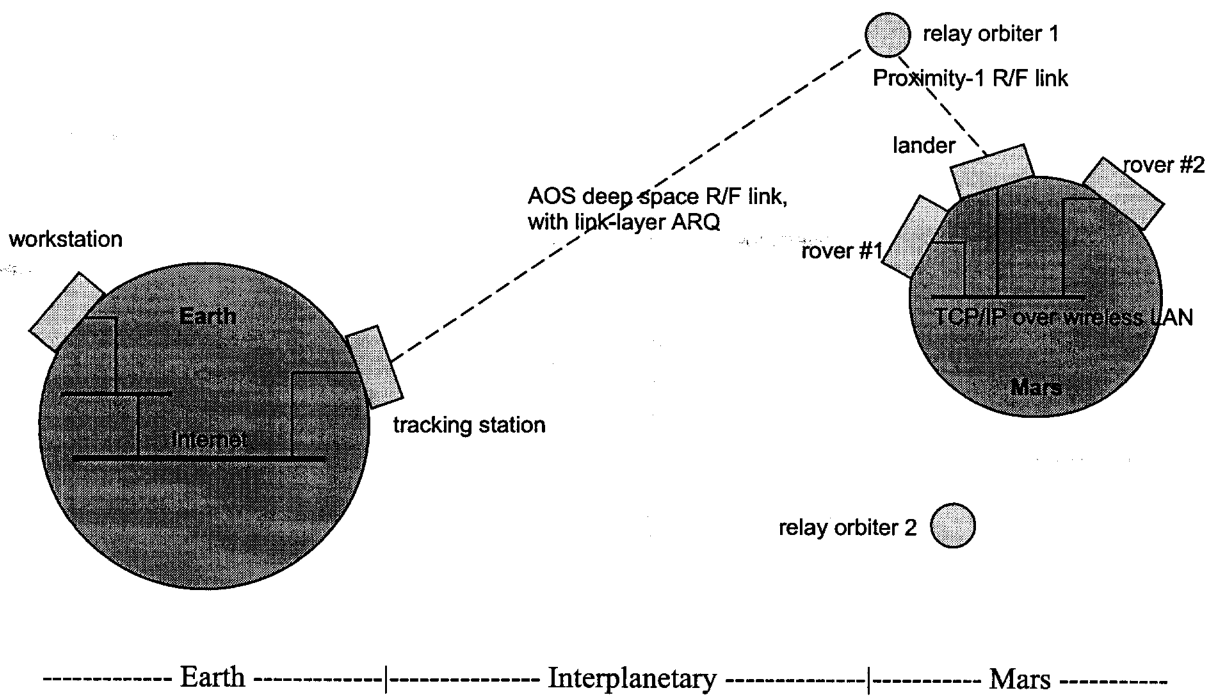

Figure 1. A sample application of delay-tolerant networking.

Application. Each rover periodically captures photographic images, stores the images as files in its on-board computer's file system, and uses the CCSDS File Delivery Protocol (CFDP) [6] to send the files to the scientist's workstation. For each transmitted file, CFDP issues a Metadata protocol data unit (PDU), containing such information as the name of the file and the time the image was captured; a single "file data segment" PDU containing the file's content; and a end-of-file PDU containing a computed checksum for the file. CFDP on the rover passes the PDUs to Bundling on the lander (via the RPC interface) for transmission. Bundling encapsulates the PDUs in bundles and uses the available regional transport systems to forward the bundles reliably, end-to-end: Proximity-1 in the "Mars" region, reliable AOS in the "Interplanetary" region, and TCP/IP in the "Earth" region. CFDP, running on the scientist's workstation, reconstructs each image file from the PDUs delivered in the received bundles and uses the checksums to assure that image integrity has been preserved.

A few points to note:

- Application software – CFDP and the image capture and display systems – operates in complete ignorance of all the underlying complexity of this scenario. Bundling's tiered routing and deferred transmission insulate the applications from the constantly changing topology of the network and the scheduling constraints on the expensive deep space link.

- A continuous, contemporaneous end-to-end path from rover to workstation might never exist, but the store-and-forward nature of Bundling operations makes such a path unnecessary.

- Tiered, point-to-point ARQ shifts from the lander to the orbiters the responsibility for retransmitting data lost over the deep space link. This is advantageous, because storage and processing power are less expensive (and thus likely more abundant) on a spacecraft in Mars orbit than on one on the surface: incident solar energy per square meter is greater, and relaxed constraints on structural robustness leave more of the device's mass budget available for electronics.

- Bundle fragmentation and reassembly optimizes utilization of the radio contacts between the lander and orbiters. At the end of a contact, when the last bundle (e.g., file data content PDU) to be sent is too large for complete transmission before loss of link, that bundle can be automatically divided for partial transmission on the current contact, with the balance to be transmitted at the beginning of the next contact – possibly with the other orbiter. The two resulting fragmentary bundles are forwarded independently through the network and reassembled at the final destination node (the scientist's workstation), the original CFDP PDU is extracted from the reassembled bundle, and the image file is reconstructed in the usual way.

DEVELOPMENT STATUS

An Internet Draft describing the DTN architecture in detail was offered for peer review in March of 2003, and an initial specification for the Bundling protocol has been drafted. Prototype implementations of Bundling and supporting software have been developed, although not all functionality is yet in place. The prototype implementation source code and the architecture and protocol specification drafts are freely available for download at the web site of the Delay-Tolerant Networking Research Group of the Internet Research Task Force: http://www.dtnrg.org.

The end-to-end prototype system has been demonstrated to be tolerant of system reboots and of simulated regional connectivity lapses lasting up to 60 minutes, and we know of no structural obstacle to tolerating much longer interruptions.

CONCLUSION

As the above example illustrates, Delay-Tolerant Networking provides a simple yet powerful framework for telemetry return from environments characterized by a variety of highly challenging communication constraints. Although the original motivation for its development was support for the operation of research robots in deep space, DTN is potentially applicable to terrestrial research and operations scenarios where the absence of continuous high-speed Internet connectivity poses similar problems: undersea exploration, email delivery to remote Arctic villages, pipeline management, and others.

ACKNOWLEDGMENTS

Delay-Tolerant Networking grew out of Interplanetary Internet research that began in early 1998. The authors of that research and of the DTN architecture that is descended from it include Vint Cerf of MCI, Kevin Fall of Intel Corp., Adrian Hooke and Leigh Torgerson of JPL, Bob Durst and Keith Scott of MITRE Corp., Howard Weiss of SPARTA, and Eric Travis of Global Systems and Technologies.

The research described in this paper was in part conducted at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

REFERENCES

[1] Schaefer and Sytsma, "Telemetry and the Internet", Western Snow Conference 2001, Sun Valley, Idaho, April 16-19, 2001.

[2] Burleigh, Fall, et al., "Delay-Tolerant Networking – An Approach to Interplanetary Internet", IEEE Communications Magazine, New York, Vol. 41, No. 6, June 2003.

[3] Padhye, Firoiu, Towsley and Kurose, "Modeling TCP Throughput: A Simple Model and its Empirical Validation", ACM SIGCOMM '98 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, Vancouver, British Columbia, September,1998.

[4] CCSDS 211.0-B-1 Proximity-1 Space Link Protocol, Blue Book Issue 1, CCSDS, October 2002.

[5] CCSDS 701.0-B-3 Advanced Orbiting Systems, Networks and Data Links: Architectural Specification, Blue Book Issue 3, CCSDS, June 2001.

[6] CCSDS 727.0-B-2 CCSDS File Delivery Protocol (CFDP), Blue Book Issue 2, CCSDS, October 2002.

Scott Burleigh – biography

Scott Burleigh has twenty-nine years of experience in software development, of which the past seventeen have been spent at NASA's Jet Propulsion Laboratory. In 1988-89 he developed one of the Laboratory's earliest systems based on Internet technology, a data distribution server that supported near-real-time analysis of science instrument data returned from the Voyager 2 encounter with the planet Neptune. He has subsequently worked on high-speed telemetry data acquisition and management systems, optimization of machine learning algorithms, distributed real-time spacecraft simulation, an embeddable object persistence system, software fault tolerance, and sensor webs. He developed the first implementation of the CCSDS File Delivery Protocol, and he is a member of the Delay-Tolerant Networking Research Group of the Internet Research Task Force. He holds a B.A. in English Literature from the State University of New York at Albany.