BPTAP: A New Approach to IP over DTN

Philip Tsao
ptsao@jpl.nasa.gov
Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Dr.
Pasadena, CA 91109
and
Sam Nguyen
phn19@yahoo.com
Broadata Communications, Inc.
2545 W. 237th St.
Torrance, CA 90505

Outline

- Background
- What BPTAP does and doesn't do
- Technical Details
- Performance Measurements
  - Quantitative
  - Qualitative
- Conclusions
- Acknowledgments

Background

- Traditional Internet protocols have been widely deployed for a variety of applications.
- However such protocols generally perform poorly in situations in which
  - round trip delays are very large (interplanetary distances)
  - persistent connectivity is not always available (widely dispersed MANET).
- Delay/Disruption Tolerant Network (DTN) technology was invented to address these issues
  - Relay nodes "take custody" of blocks of network traffic on a hop-by-hop basis and retransmit them in cases of expected or unexpected link outage
  - Bundle lifetime may be configured for long round trip light times

Deploying Internet Protocol applications across Space Links

- Application Specific Gateway
  - Conceptually simple
  - Cost prohibitive (engineering effort scales linearly with number of distinct applications)
- DTN over IP
  - Traditional Approach
  - Existing applications require significant re-engineering for delay/disruption tolerance
- IP over DTN
  - Existing applications require modest (if any) re-engineering for delay/disruption tolerance

Prior work on IP over DTN

- Ochiai, Shimotada and Esaki were the first to describe encapsulating IP within a DTN.
- Soares, Farahmand and Rodrigues described “Vehicular Delay-Tolerant Networks” (VDTN) that have the capability to aggregate several IP packets into a single DTN bundle to reduce overhead for more efficient message delivery.
- Caro described an IP over DTN implementation that supported a packetized voice application as part of the DARPA Wireless Networks After Next (WNAN) program

- BPTAP is novel by encapsulating Ethernet frames in BP

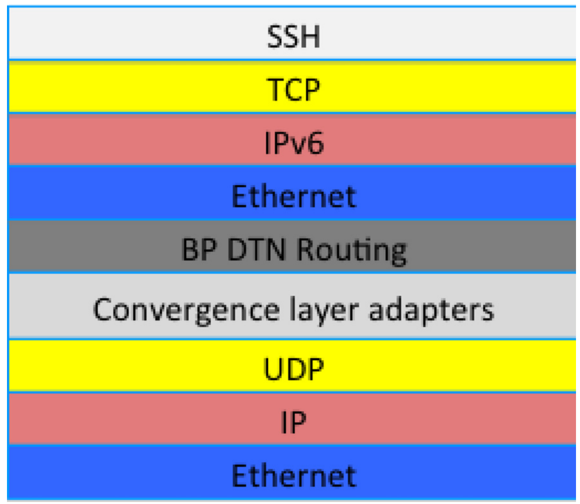
What BPTAP does

- BPTAP encapsulates Ethernet frames within DTN Bundles
  - Ethernet frames may contain IP packets
  - Exploit functionality already implemented in Operating System (DHCP, ARP/NDP, etc.)
- Maps IPv4 or IPv6 DiffServ Code Point to Bundle priority flags to preserve QoS
- Supports an arbitrary number of endpoints
- Supports IPSec
- BPTAP requires kernel TUN/TAP driver support
  - TAP is virtual Ethernet NIC, TUN is virtual IP device
  - TUN/TAP drivers available for UNIX and Windows

What BPTAP does not do

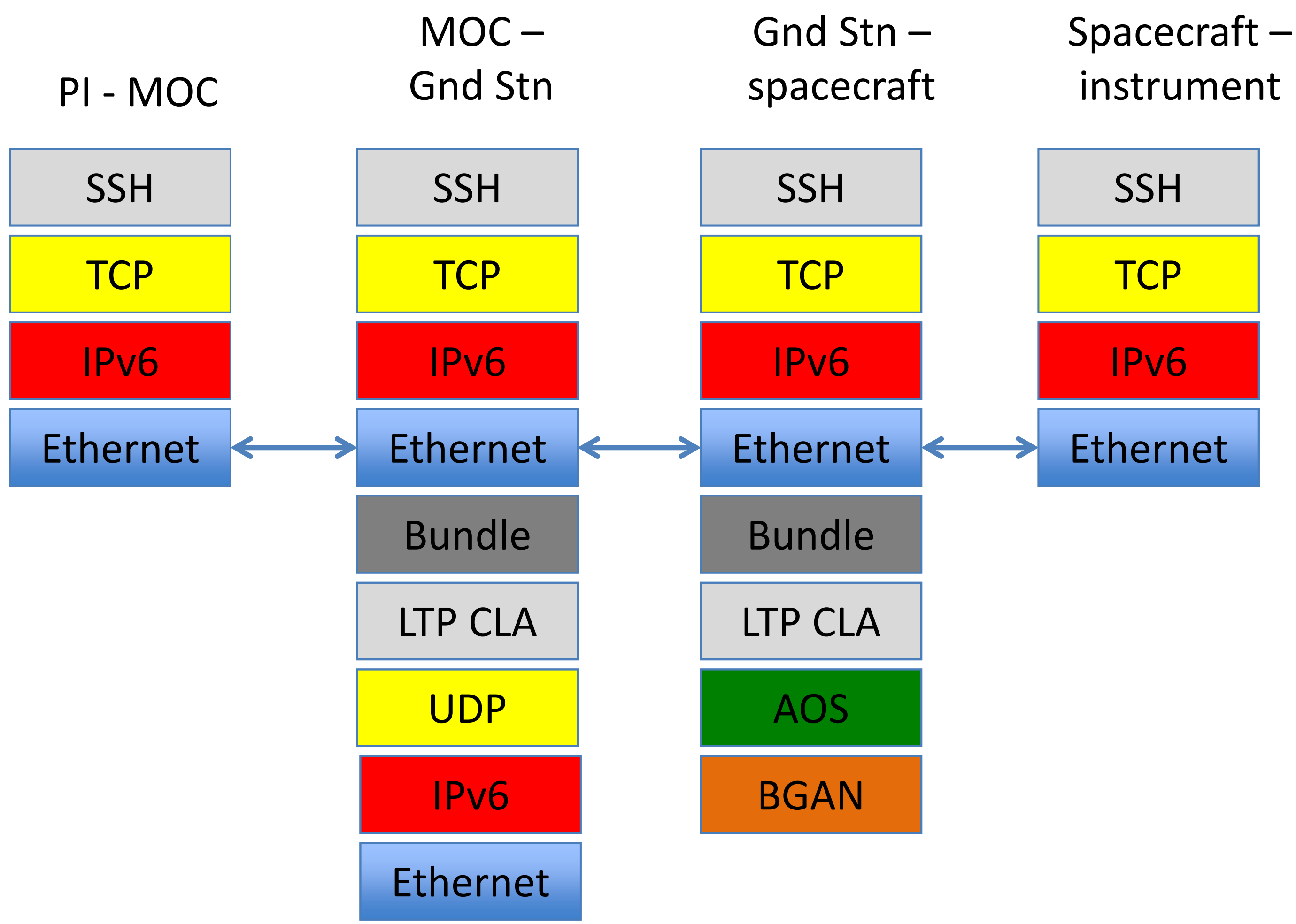
- BPTAP cannot make inherently non delay/disruption tolerant applications magically delay/disruption tolerant
  - Such applications may have short timeout counters that may be exceeded in a long link outage
  - These applications will work correctly when there is no link outage (BPTAP acts as a compatibility shim)
- BPTAP currently has no support for signaling bundle lifetime or custody requirement

Technical Details: Protocol Stackup

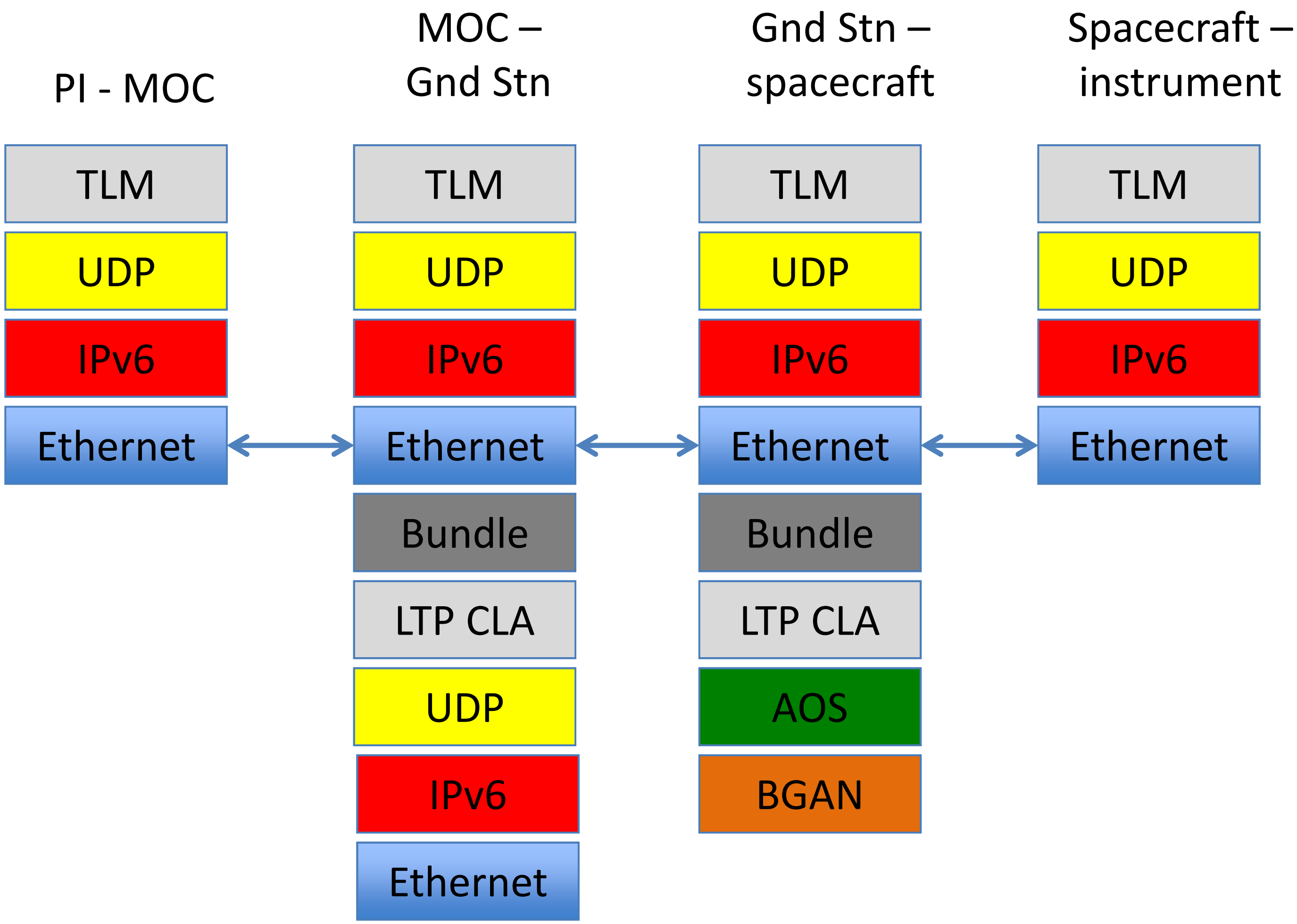



```
0000  08 00 27 5e 18 88 08 00  27 07 1c e5 08 00 45 00   ..'^.... '.....E.
0010  00 be 00 00 40 00 40 11  24 aa 0a 0a 00 fa 0a 0a   ....@.@. $.......
0020  00 78 e7 d6 04 59 00 aa  ed 60 03 01 4b 00 01 00   .x...Y.. .`..K...
0030  81 14 86 c7 eb f1 29 00  06 81 14 10 02 01 00 00   ......). ........
0040  00 00 00 00 81 ae 84 9b  28 01 0a 00 01 09 7d 96   ........ (.....}.
0050  07 c6 d6 a8 2a 62 00 53  80 26 2f 86 dd 60 00 00   ....*b.S .&/..`..
0060  00 00 47 06 40 fc 00 00  00 00 00 00 00 00 00 00   ..G.@... ........
0070  00 00 00 00 01 fc 00 00  00 00 00 00 00 00 00 00   ........ ........
0080  00 00 00 00 02 00 16 c6  a6 a2 4c f5 4b 61 3f ed   ........ ..L.Ka?.
0090  9f 80 18 00 b3 53 c6 00  00 01 01 08 0a 00 1a 22   .....S.. ......."
00a0  56 00 31 9c 4b 53 53 48  2d 32 2e 30 2d 4f 70 65   V.1.KSSH -2.0-Ope
00b0  6e 53 53 48 5f 35 2e 33  70 31 20 44 65 62 69 61   nSSH_5.3 p1 Debia
00c0  6e 2d 33 75 62 75 6e 74  75 37 0d 0a               n-3ubunt u7..
```

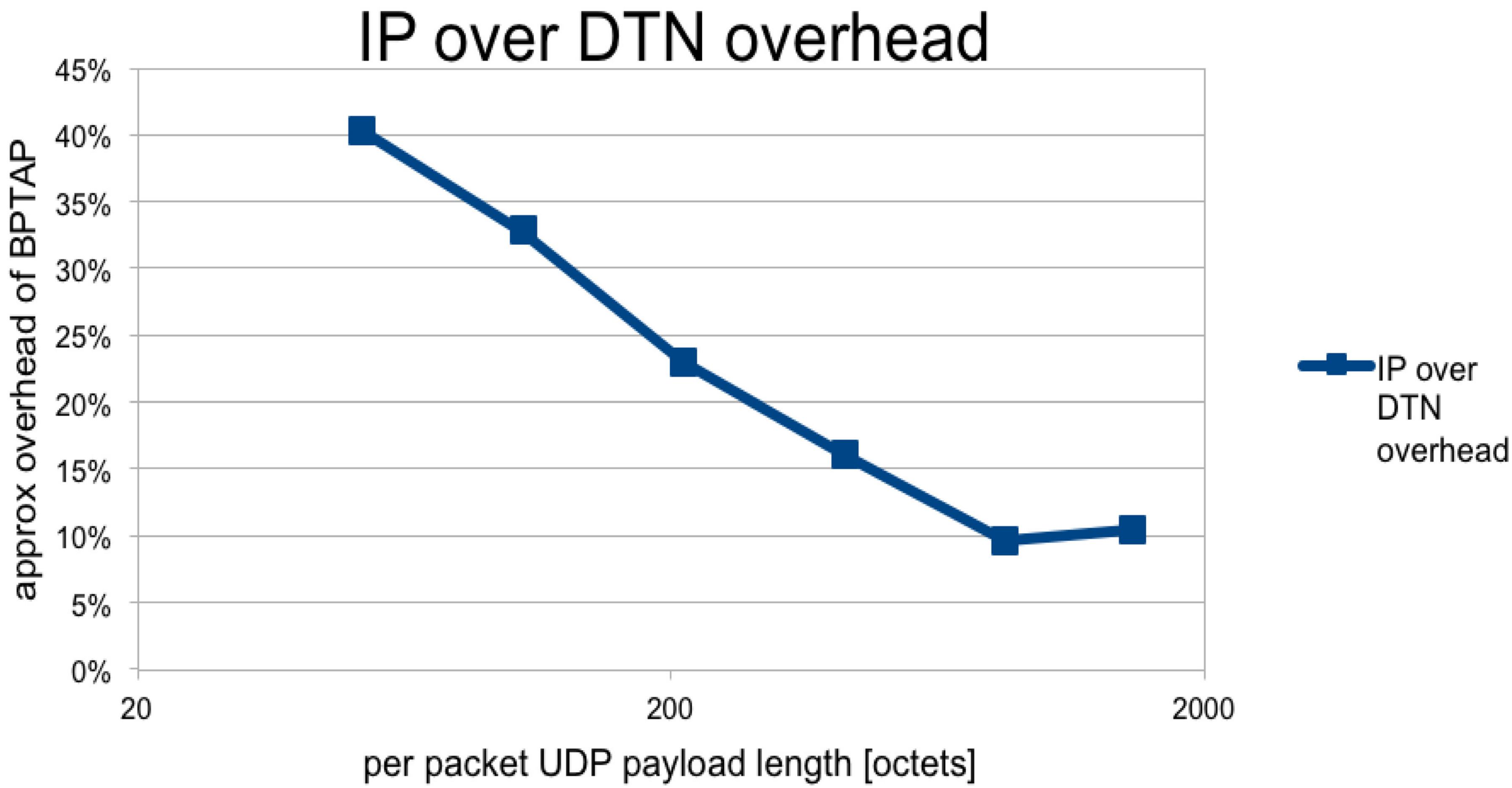
Example Scenario (1)



Example Scenario (2)



Quantitative Performance: What is the "cost" of BPTAP?

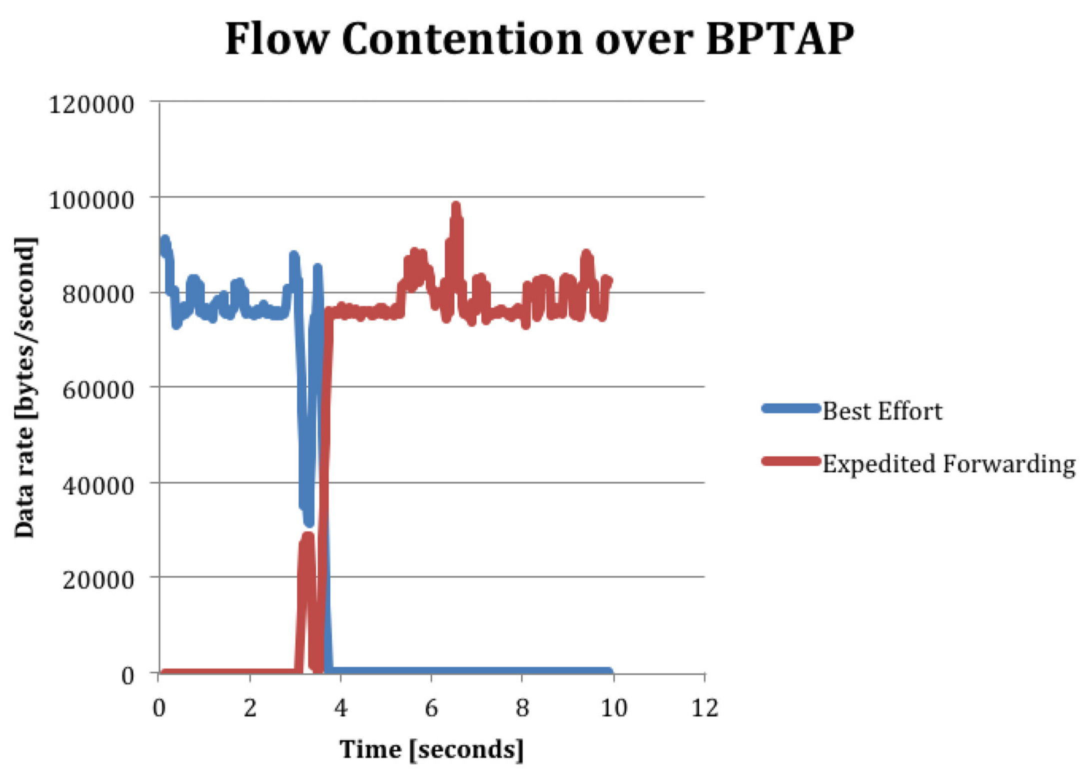


Throughput and Latency

Test Scenario	Throughput (P2020 to PC)	Round Trip Latency (PC to P2020 to PC)
TCP	94.1 Mbits/sec	N/A
TCP over DTN	36.9 Mbits/sec	N/A
UDP	95.7 Mbits/sec	N/A
UDP over DTN	44.5 Mbits/sec	N/A
ICMP	N/A	.082 ms
ICMP over DTN	N/A	1.49 ms

"P2020" is 1.2 GHz e500v2 PowerPC SPE dual core running Wind River Linux 4.2 and "PC" is 3.16 GHz Core 2 Duo running Ubuntu Linus 10.04

IP DiffServ QoS enforced by DTN

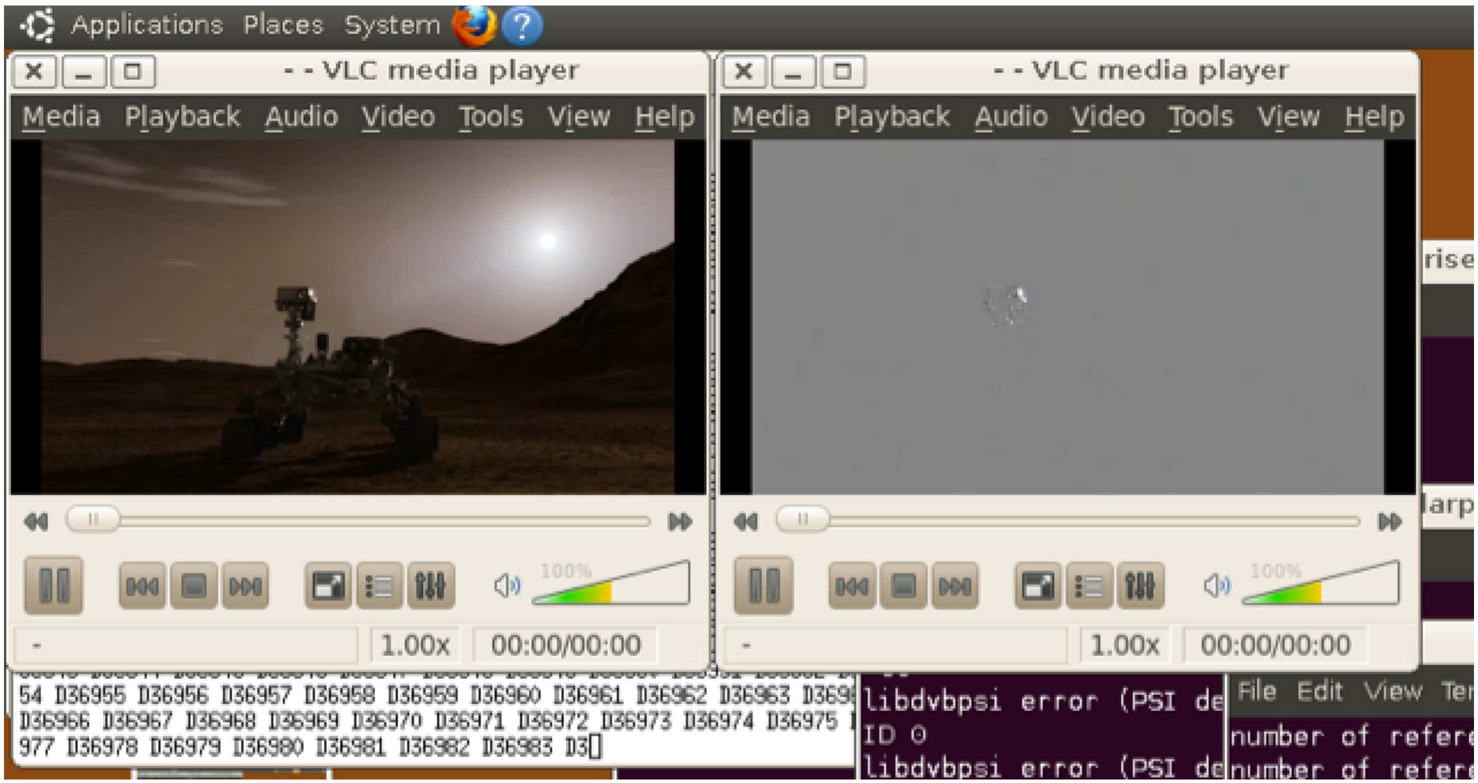


“Best Effort and “Expedited Forwarding” IP DiffServ Code Points were mapped to “Bulk” and “Expedited” bundle control flags

Qualitative Performance: Video over BPTAP

Five Second Outage

IP over DTN | IP



Distributed Real-time Applications

BPTAP can support real time distributed simulation applications

Conclusions

- BPTAP provides an Ethernet-like interface for IP applications to access a DTN
- IP Diffserv code points are mapped to Bundle control flags
- BPTAP does not make non delay/disruption tolerant applications delay/disruption tolerant
- BPTAP allows common off-the-shelf IP applications and tools to run over DTN
- Impact on overhead, latency and throughput modest

Acknowledgments

This work was carried out at the Jet Propulsion Laboratory, California Institute of Technology, and was funded by contracts with the Defense Advanced Research Projects Agency, QinetiQ North America and the National Aeronautics and Space Administration. The authors would like to thank Richard Borgen, Scott Burleigh, John Choi, Loren Clare, Antonio Garcia, David Green, Nuha Jawad, Joe Klein, Norman Lay, Tim Loomis, Joel Marmet, Clayton Okino, Fabrizio Pollara, Joshua Schoolcraft, John Seguí, Leigh Torgerson and Cindy Wang for their valuable suggestions and generous assistance.

BACKUP

Example Implementation