ION IMPLEMENTATION OF THE DTN ARCHITECTURE

Training Overview

Theoretical Knowledge

o **Space Communication Challenges**

o **DTN History**

o **DTN Features and Benefits**

o **The ION Implementation**

o **ION Architecture**

o **The basic ION Structure and Functions**

o **ION design principles and constraints**

o **Resource Management in ION**

o **ION Network Operations Concepts**

o **ION Convergence-Layer Adapters**

o **ION Functional Modules**

Hands-On Applications

During this Training Session you will:

- Study and demonstrate a simple ION application: bpsource, bpsink

- Configure a multiple-nodes test network (simulation workstation)

- Learn about trouble-shooting an ION network configuration

- Stretch goal: develop a new ION application to run on the new network

Course Outcomes

At the end of this Training Session you will be able to:

Understand key principles of ION's architecture

Integrate theoretical knowledge of ION into your design thinking

Perform ION infrastructure maintenance

Develop ION-based software applications
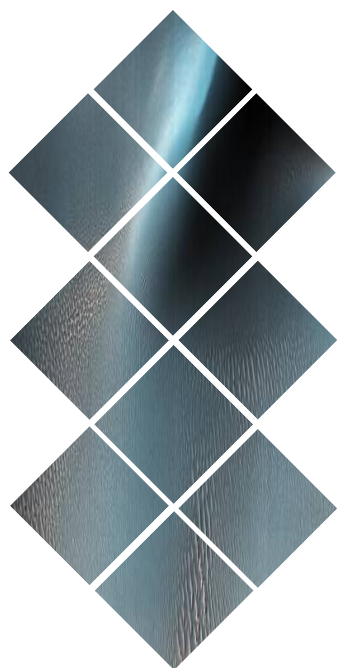
Day 1 Agenda – Morning

Key Topics

o **Space communication overview**

o **Internet communication overview**

o **Interplanetary Internet**

o **Delay-Tolerant Networking (DTN)**

o **Solar System Internet**

What are we talking about?

The **Interplanetary Overlay Network (ION)** software distribution
is an implementation of
Delay-Tolerant Networking (DTN) architecture.

It's difficult to make sense of ION
without knowing something about DTN,
so let's start there.

INTRODUCTION TO DTN

- ◆ **Space Communication Overview**
- ◆ **Internet Communication Overview**
- ◆ **Delay-Tolerant Networking**
- ◆ **Brief History of the Solar System Internet**
- ◆ **DTN vs IP**
- ◆ **DTN Features**
- ◆ **DTN Benefits**

Space Communications Overview

Space Communication Challenges

Extreme distances and high rates of data loss (due to radio signal interference) make communicating between Earth and **interplanetary** spacecraft a challenge.

Constant orbital movement, link handovers, and discontinuous vehicle operations make the challenge even more difficult, even in **near-Earth space**.

Delay- (or Disruption-) Tolerant Networking (DTN) is NASA's solution for reliable, automated "network" communications in space missions.

Space communication latencies

Consider:

- Typical round-trip time (RTT) between two points on the Internet: **100** ms to **300** ms
- Internet RTT via GEO satellite: **480-560** ms
- Distance to ISS (through TDRS): ~71322 km
    - RTT: typically about **1200** ms on Ku link
- Distance to the Moon: ~384400 km
    - RTT: **2560** ms
- Minimum Distance to Mars: ~54.6 Million km
    - RTT: **364,260** ms or ~6 min
- Average Distance to Mars: ~225 Million km
    - RTT: **1,501,000** ms or ~25 min
- Farthest Distance to Mars: ~401 Million km
   - RTT: **2,674,000** ms or ~44.6 min
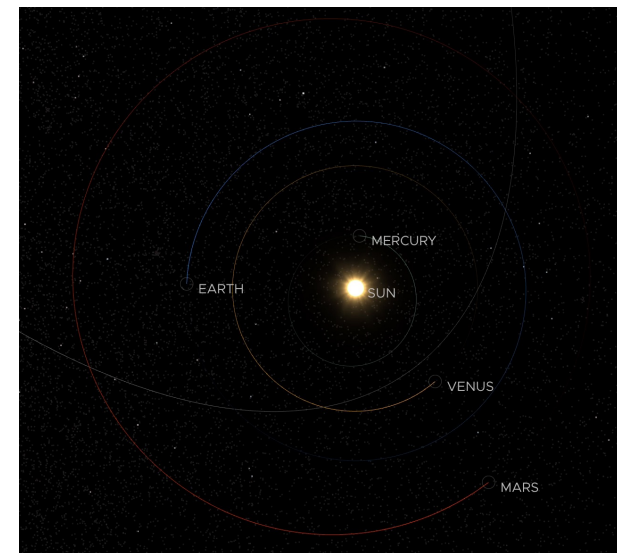
Context in which space communication is being performed

Deep Space Communications Today

- Communication opportunities are scheduled, based on orbit dynamics & operations plans.

- Transmission and reception episodes are individually configured, started, and ended by command. Spacecraft to ground.

- Reliability over deep space links is by management: on loss of data, command retransmission.

- More recently – MER, MSL – we have had managed forwarding through a relay point: commands to Odyssey, MRO, etc.

What's wrong with that?
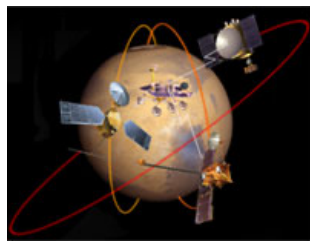
- This mission communications model has worked fine for over forty years; we've done a lot of good science.
- But the status quo is:
  - Labor-intensive
    - Communication operations cost is a large fraction of the budget for each mission.
    - Risk of human error mandates mitigations that further increase cost.
  - Program-limiting
    - Cost and risk increase with the number of links between communicating entities.
    - As cross-links among spacecraft become common (Mars network, lunar exploration Constellation), cost and risk increases are non-linear with increase in the number of spacecraft.

An Alternative

Networks, such as the Internet, automate complex communications. The Internet is used today to conduct scientific investigations. We're already building an **Internet of Things**.

So why not use it for deep space science missions too?

- **Minimize cost (automation, COTS).**
- **Minimize risk (huge installed base).**

High-performance Wireless Research and Education Network (HPWREN)

11 GHz link to Mt. Soledad

California Wolf Center

Optical Fiber Infrasound Sensor

The Internet itself doesn't quite do it

The Internet Architecture (details a little later) is based on the assumptions that:
- Network nodes are continuously connected
- Networks have short signal propagation delays
- Data links are symmetric and bi-directional
- Bit error rates are low

These assumptions are NOT VALID in the Space Environment!
A new sort of network is needed.

Earth

Moon

Satellite
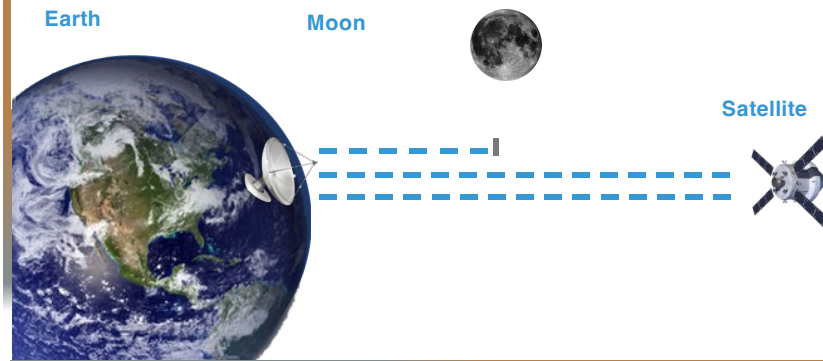
In a space environment:
- Connections can be routinely interrupted
- Interplanetary distances impose delays
- Link data rates are often asymmetric, and some links are simplex
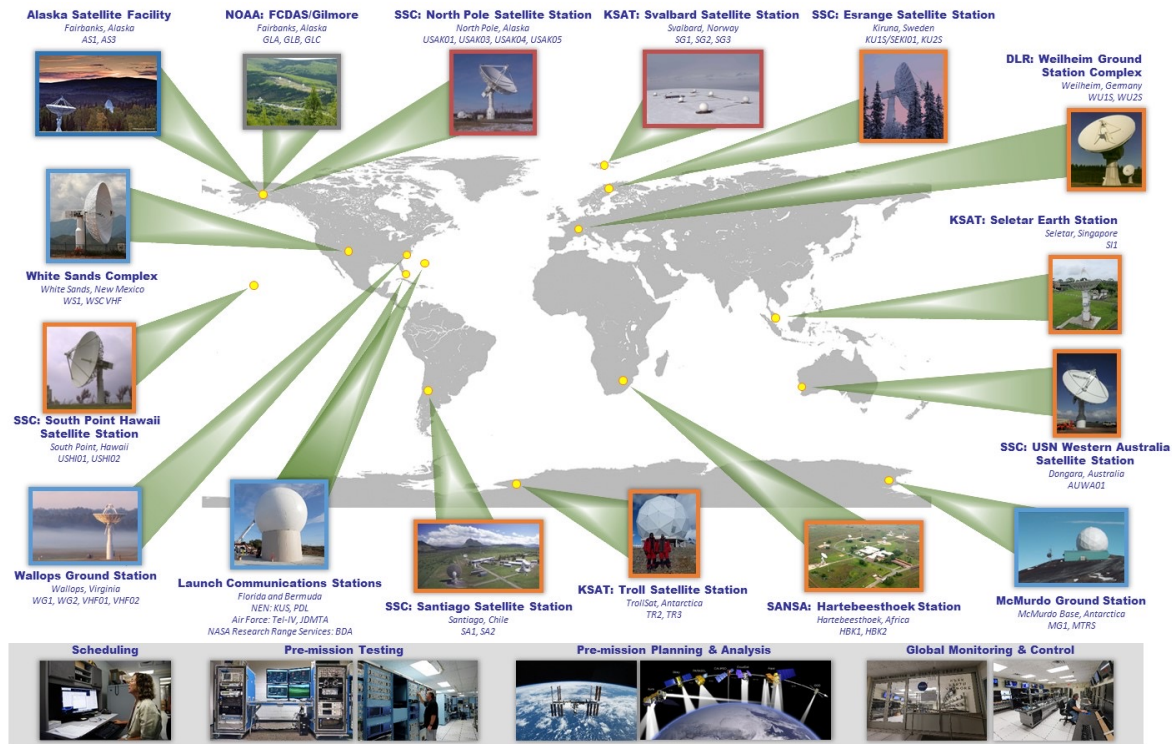- Bit error rates can be very high

But isn't there already a space network?

Yes!

NASA manages **three** communication networks, consisting of distributed ground stations and space relay satellites for data transmission and reception that support both NASA and non-NASA missions.
These are the Deep Space Network (DSN), the Near Earth Network (NEN), and the Space Network (SN).
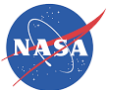
Other national space agencies operate similar networks, and there are commercially operated networks as well (notably, now, from Amazon).
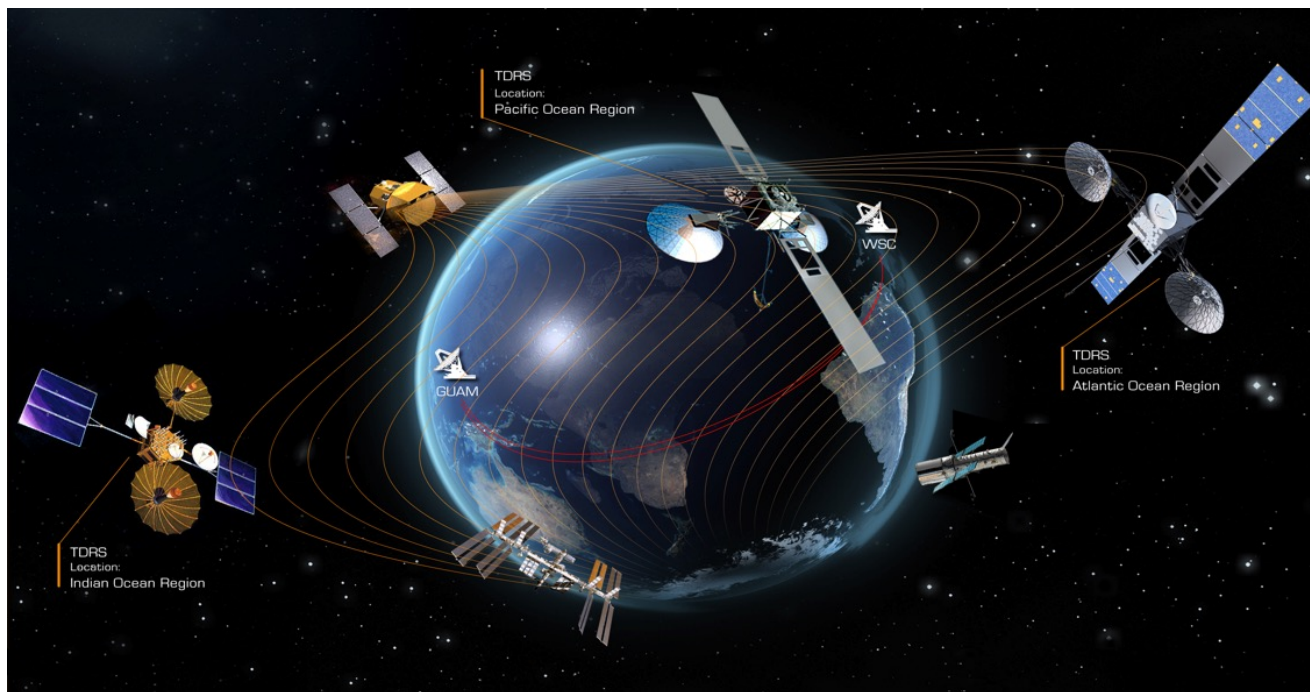
Near Earth Network Map

Alaska Satellite Facility
Fairbanks, Alaska
AS1, AS3

NOAA: FCDAS/Gilmore
Fairbanks, Alaska
GLA, GLB, GLC

SSC: North Pole Satellite Station
North Pole, Alaska
USAK01, USAK03, USAK04, USAK05

KSAT: Svalbard Satellite Station
Svalbard, Norway
SG1, SG2, SG3

SSC: Esrange Satellite Station
Kiruna, Sweden
KU1S/SEKI01, KU2S

DLR: Weilheim Ground Station Complex
Weilheim, Germany
WU1S, WU2S

White Sands Complex
White Sands, New Mexico
WS1, WSC VHF

KSAT: Seletar Earth Station
Seletar, Singapore
S1

SSC: South Point Hawaii Satellite Station
South Point, Hawaii
USHI01, USHI02

SSC: USN Western Australia Satellite Station
Dongara, Australia
AUWA01

Wallops Ground Station
Wallops, Virginia
WG1, WG2, VHF01, VHF02

Launch Communications Stations
Florida and Bermuda
NEN: KUS, PDL
Air Force: Tel-IV, JDMTA
NASA Research Range Services: BDA

SSC: Santiago Satellite Station
Santiago, Chile
SA1, SA2

KSAT: Troll Satellite Station
TrollSat, Antarctica
TR2, TR3

SANSA: Hartebeesthoek Station
Hartebeesthoek, Africa
HBK1, HBK2

McMurdo Ground Station
McMurdo Base, Antarctica
MG1, MTRS

Scheduling

Pre-mission Testing

Pre-mission Planning & Analysis

Global Monitoring & Control

INTRO

ION AND DTN

For LEO in particular, no problem!

...almost

1. **ISS round-trip latency through TDRS is lengthy, typically over one second. Reliable high-rate communication is hard to achieve.**
2. **Loss of signal (LOS) and acquisition of signal (AOS) cycles interrupt end-to-end connections.**
   a. **Large file transfers (e.g., email attachments) often fail.**
   b. **Download transfers that span Ku-Band AOS periods require special scheduling and scripting.**
   c. **Data lost during LOS must be retransmitted, requiring extra storage and manual intervention.**

All of which can be engineered around, though.

So why aren't we done?

Because these are closed networks
of "bent-pipe" link repeaters only.

They don't include instruments, vehicles,
or scientists' workstations, and they aren't connected.

**What's needed for the space flight missions of the future
is a new "internet", spanning the solar system.**

# Internet Communications Overview

Protocol "stacks" (1 of 2)

In network design, we allocate different functions to different protocols.

Those different protocols work together in a nested fashion, with each one using the services of another and providing service to yet another.

We represent these service relationships graphically in a *stack*, where each layer of the stack corresponds to a function.

In a protocol stack diagram, each layer of the diagram identifies the protocol that performs the function at that layer, relying on the services of the layer below it and providing service to the layer above it.

The Internet protocol stack looks like this:

| |
|---|
| Application |
| Transport |
| Network |
| Data Link |
| Physical |
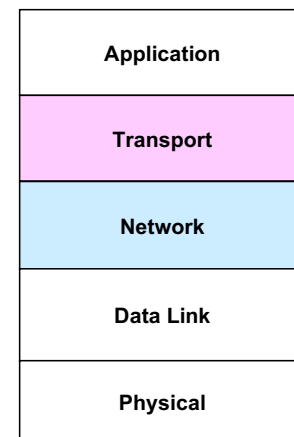
The Internet

Protocol "stacks" (2 of 2)

The stacking of protocols is reflected in the structure of the *protocol data units (PDUs)* that are transmitted over the network:

- A data unit of protocol N is usually formed by prepending a protocol-N header to a data unit of protocol N + 1, which it is said to *encapsulate*.
- The protocols headers in a PDU echo the stack, except rotated 90°.
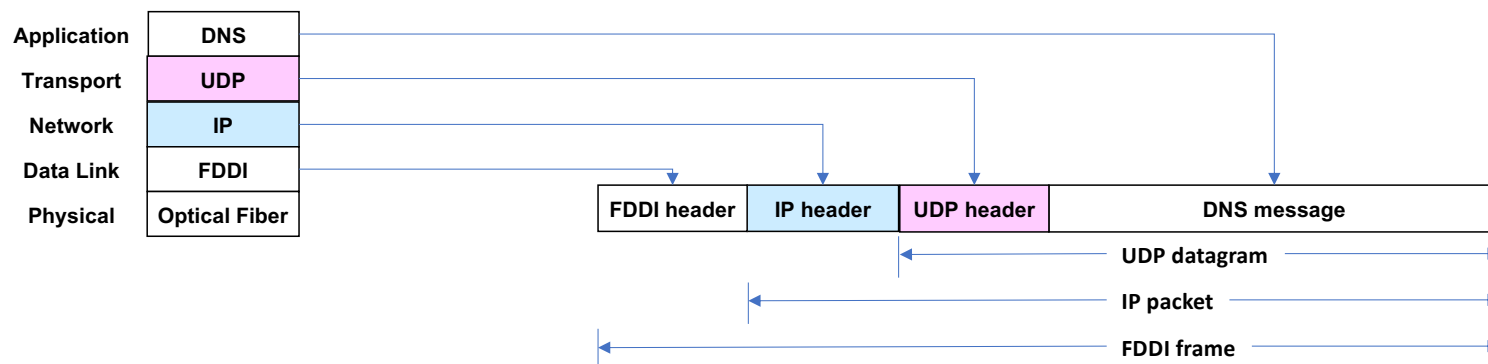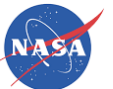
Internet in a nutshell

- **Packet Switching vs. Circuit Switching**
  - **Statistical multiplexing:**
    - **Better use of available resources**
  - **Possibility of congestion and loss**
  - **Smart edges, dumb (fast) core**

- **IP unites individual networks into an internetwork**

- **Autonomous Systems**

- **Performance issues in stressed environments**
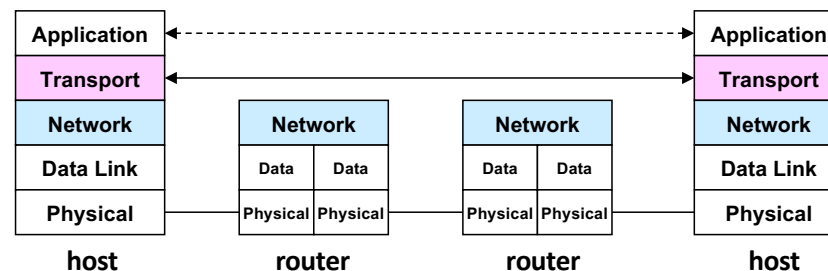
Assumptions and Architectural Decisions

- **Key Assumptions**
  - *If* you can get from a source to a destination, you'll be able to get there along a continuously-available path
  - The network topology changes relatively slowly
  - Latency is low
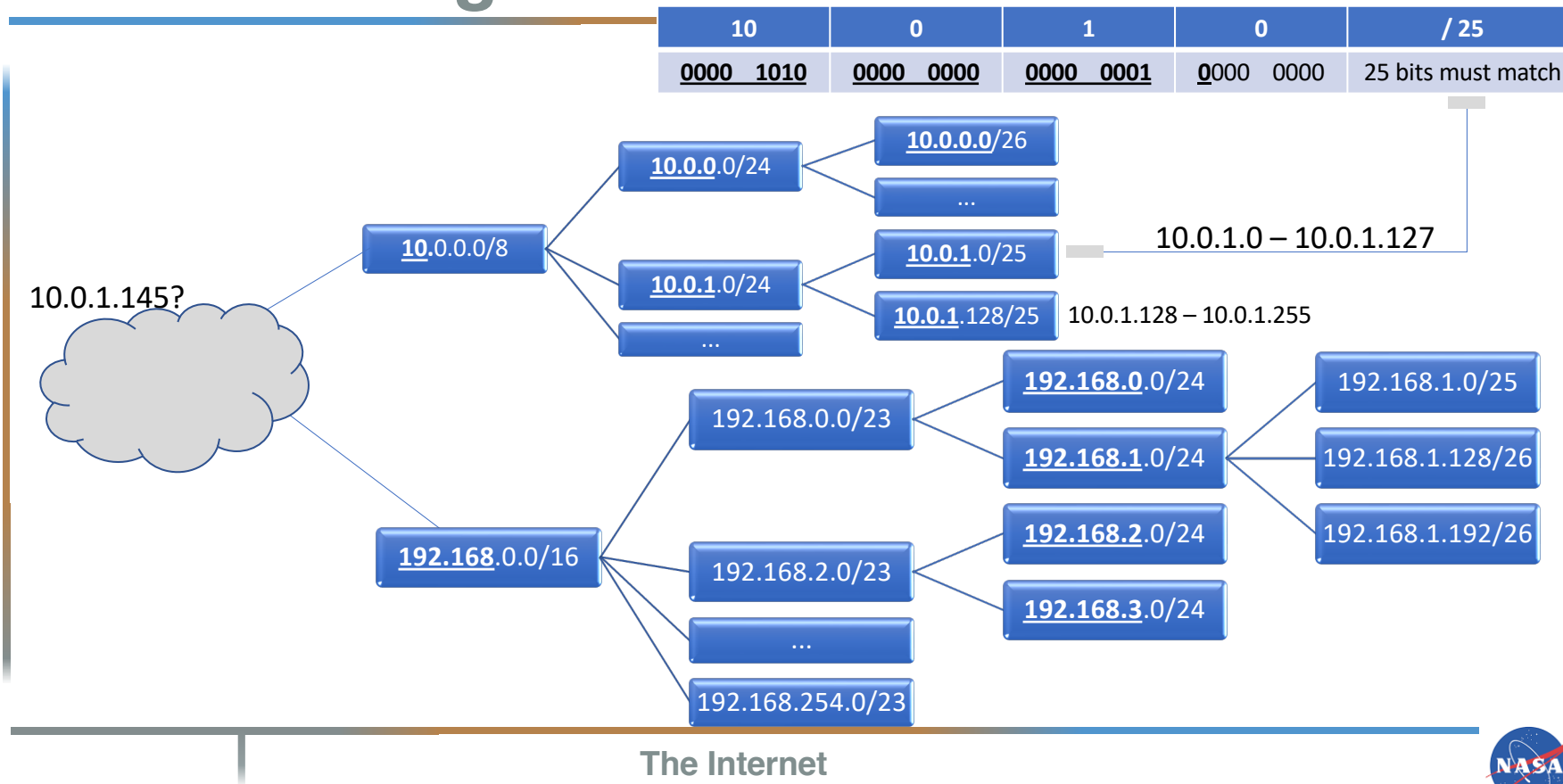 - Round-trips are cheap
  - Packet loss rates are relatively low

- **Internet Architectural Decisions**
  - **Packet switching and statistical multiplexing** (vs. circuit switching and dedicated resources)
    - Sometimes there will be loss
  - **Smart (slow) edges, dumb (fast) core**
 - If you need to know something from somewhere else, ask and be told: **client/server**

IP Forwarding: Names Are Also Addresses

10	0	1	0	/ 25
0000 1010	**0000 0000**	**0000 0001**	**0**000 0000	25 bits must match

10.0.1.0 – 10.0.1.127

10.0.1.145?

- 10.0.0.0/8
  - 10.0.0.0/24
    - 10.0.0.0/26
    - ...
  - 10.0.1.0/24
    - 10.0.1.0/25
    - 10.0.1.128/25   10.0.1.128 – 10.0.1.255
  - ...
- 192.168.0.0/16
  - 192.168.0.0/23
    - 192.168.0.0/24
    - 192.168.1.0/24
      - 192.168.1.0/25
      - 192.168.1.128/26
      - 192.168.1.192/26
  - 192.168.2.0/23
    - 192.168.2.0/24
    - 192.168.3.0/24
  - ...
  - 192.168.254.0/23

The Internet

WHY Names Are Also Addresses…

- **Organizing IP addresses into a tree provides *scalability***

Jan 2019: ~750,000 BGP RIB Entries
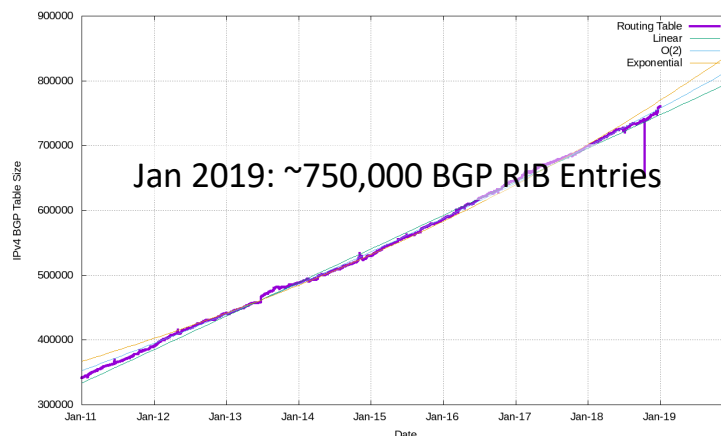
- **But introduces issues with *mobility***
  - **If a node moves in the hierarchy, its address has to change**

- **And issues with *multihoming***
  - **What if I want TWO different names / addresses for the same node?**

The Internet

BGP Size Plot from: https://blog.apnic.net/2019/01/16/bgp-in-2018-the-bgp-table/

Routing in the Internet

- IGP vs. EGP
  - Administrative boundaries
  - Scalability

- Some IGPs: OSPF, EIGRP
  - Discovery: who are my neighbors?
  - 'Immediate' routing:
    - Learn the network topology; build forwarding tables
    - Forward based on that topology until something breaks
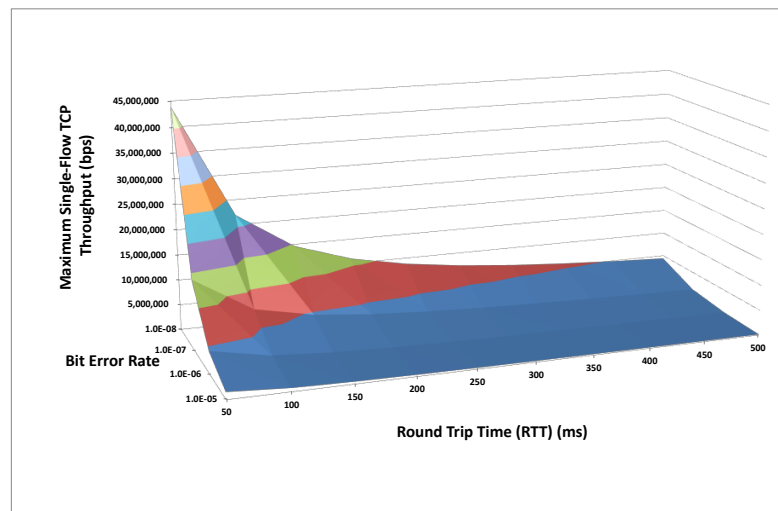 - Generally shortest-path (# of hops) under some metric

Reliable Data Delivery: TCP

- **TCP Service**
  - **Bytestream service**
  - **Reliable**
  - **In-order**
  - **Without duplicates**
- **TCP Mechanisms:**
  - **Acknowledgements**
  - **Window-based flow control**
    - **Congestion control overlay**
- **Important Points**
  - **State must be synchronized to start and end connections**
  - **State retained at endpoints**
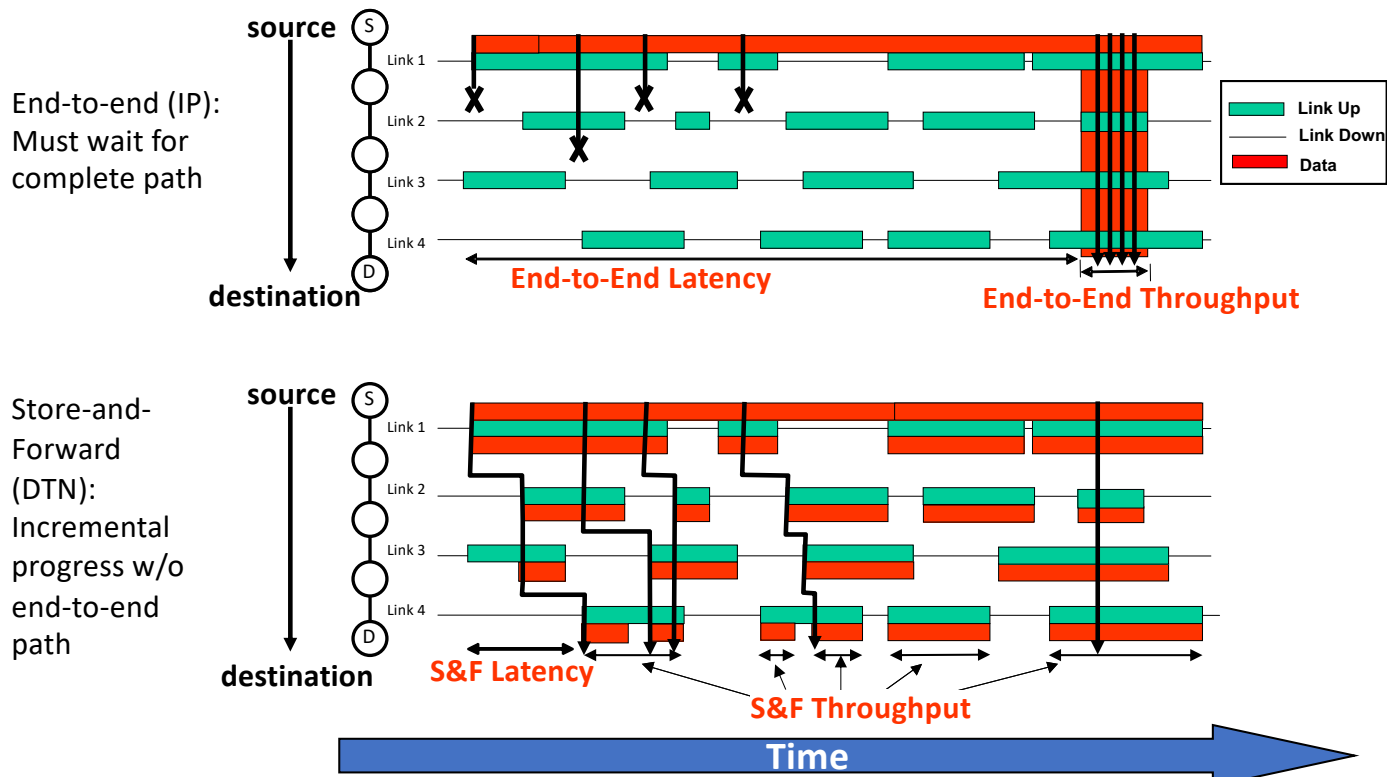 - **Performance issues in stressed environments**

The Internet

TCP Performance

- **UDP and IP as encapsulation mechanisms don't have any sensitivity to latency or loss. But TCP, the Internet's main reliability mechanism, does.**
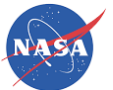
Reliable Data Delivery: TCP vs DTN

End-to-end (IP): Must wait for complete path

Store-and-Forward (DTN): Incremental progress w/o end-to-end path

source
S
destination
D

Link 1
Link 2
Link 3
Link 4

End-to-End Latency

End-to-End Throughput

Link Up
Link Down
Data

S&F Latency

S&F Throughput

Time

DTN Can Reduce Delay and Increase Throughput

Internet is:

- **Usable for LEO, but not ideal:**
  - **Lengthy round-trip latency** through TDRS, so slow.
  - **LOS/AOS cycles** are expensive.
- **Not usable for deep space:**
  - TCP isn't suitable:
    - Connection requires a three-way handshake, consuming 1.5 round-trip times with a 2 minute timeout.
    - In-order delivery means data loss freezes reception for at least one RTT.
  - There's no alternative Internet standard for **reliable transmission** that would work over deep space links.
    - So no standards for **flow control and congestion control.**
  - None of the standard **routing protocols** would work.
    - BGP relies on TCP. Others rely on timers that won't work right.
    - Transient network partitioning mustn't be interpreted as topology changes.
  - No **COTS routers** would work.
    - Interruption of outbound link must cause outbound traffic to be queued rather than discarded.
  - What's left is UDP/IP with static routing: just a less bit-efficient packaging alternative to raw CCSDS packets.

A new sort of network is needed.

Delay-Tolerant Networking

A brief history of DTN

Delay/Disruption Tolerant Networking (DTN) has its origins in work started at NASA in 1998 as an initiative to develop an "Interplanetary Internet".

**Since the mid-2000s worldwide DTN development has been led by NASA.**

In 2010, the Interagency Operations Advisory Group (IOAG), consisting of NASA and other national space agencies, recommended a timely evolution toward a fully operational Solar System Internetwork (SSI), based on DTN as a core service of the SSI.

Report of the
Interagency Operations Advisory Group
Space Internetworking Strategy Group

Recommendations on a Strategy for
Space Internetworking
November 15, 2008

The Elevator Pitch

DTN is a digital communication networking technology that enables data to be conveyed reliably among communicating entities *when round-trip times may be highly variable and/or very long*.

Data transmission is done automatically and reliably even if one or more of the network links in the end-to-end path between those entities is subject to very long signal propagation latency and/or prolonged intervals of unavailability.
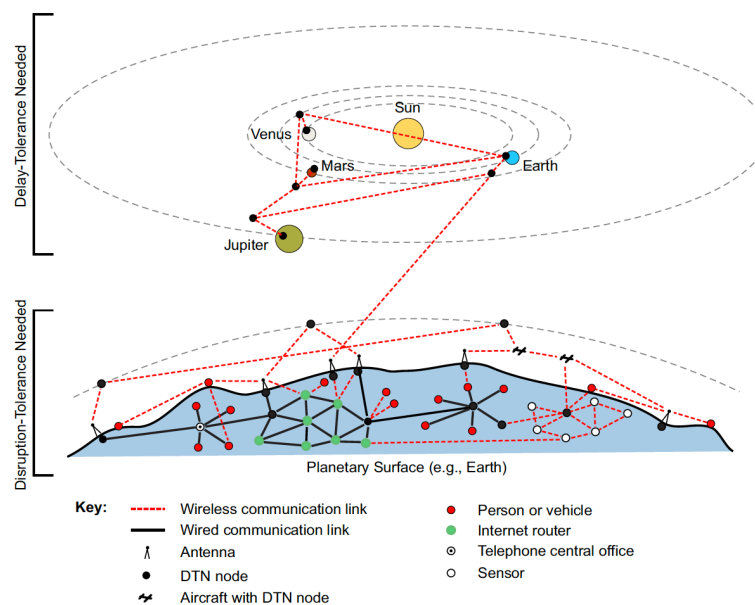
Refer to *Delay- and Disruption-Tolerant Networks (DTNs)* - A primer

Definitions and Premises (1 of 2)

Nodes

- A DTN *node* is an engine for sending and receiving data - an implementation of the DTN *bundle protocol.*
- Applications utilize DTN nodes to send or receive application data units (ADUs) carried in *bundles*. Each node is a member of one or more groups called DTN *endpoints.*
- A DTN *endpoint* is a set of DTN nodes (but often includes only a single node). A bundle is considered to have been successfully *delivered* to a DTN endpoint when some minimum subset of the nodes in the endpoint has received the bundle without error.
- Bundles are forwarded from a storage place on one node to a storage place on another node, along a path that eventually reaches the destination.

Definitions and Premises (2 of 2)

Store-and-forward message switching

- DTN uses this model to overcome the problems associated with intermittent connectivity, long or variable delay, asymmetric data rates, and high error rates. Each forwarding node stores bundles locally – <u>within the network</u>, not at the source – until they have been received by the next node in the path.

Convergence layers

- DTN protocols rely on underlying *convergence-layer* protocols for the physical transmission of data between topologically adjacent nodes in the network. Different convergence-layer protocols may be used on different segments of the end-to-end path, to accommodate variations in the network environment.

DTN vs IP: Expected Environment

IP

Short Round-Trips: Small and relatively consistent network delay—milliseconds, not hours or days—in sending data packets and receiving the corresponding acknowledgement packets.

Low Error Rates: Relatively little loss or corruption of data on each link.

Continuous, Bidirectional End-to-End Path: A continuously available bidirectional connection between source and destination to support end-to-end interaction.

Symmetric Data Rates: Relatively consistent data rates in both directions between source and destination.

DTN

Long or Variable Delay: Long propagation delays between nodes and variable queuing delays at nodes contribute to end-to-end path delays that can defeat Internet protocols and applications that rely on quick return of acknowledgements or data.

High Error Rates: Bit errors on links require correction or retransmission of the entire packet. For a given bit error rate, fewer retransmissions are needed for hop-by-hop retransmission than for Internet-type end-to-end retransmission.

Intermittent Connectivity: DTN allows data to be transmitted even when there is poor connectivity.

Asymmetric Data Rates: The Internet supports moderate asymmetries of bi-directional data rate for users with cable TV or asymmetric DSL service. But if asymmetries are large, they defeat conversational protocols.

DTN/BP can run over an existing Internet Protocol (IP) stack or can operate as an Internetworking protocol in its own right.
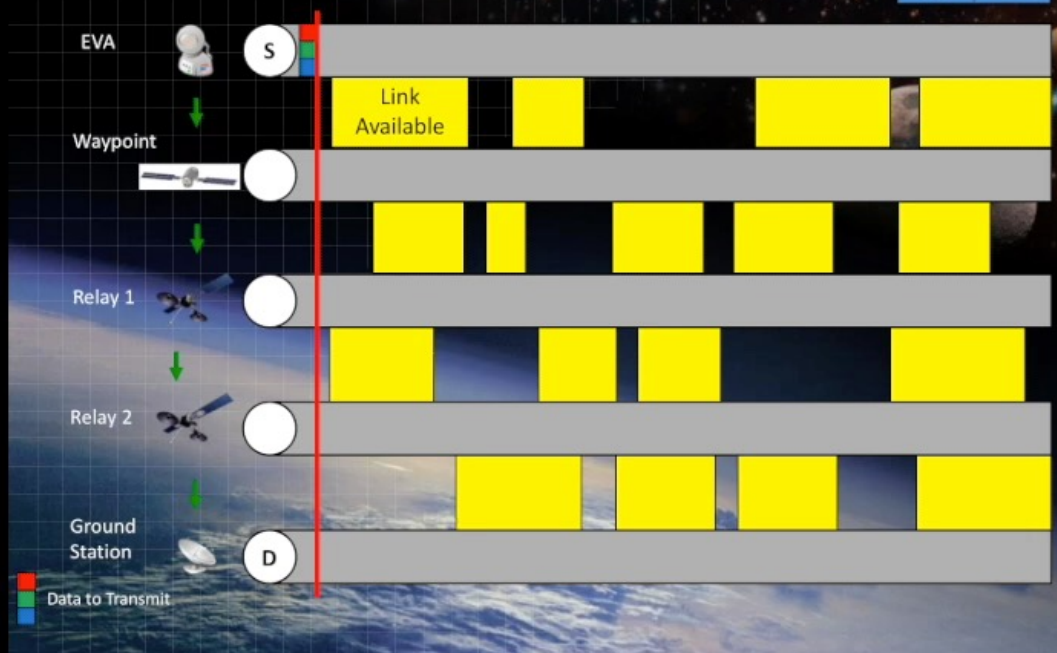
End-to-End IP, PEPs, and DTN
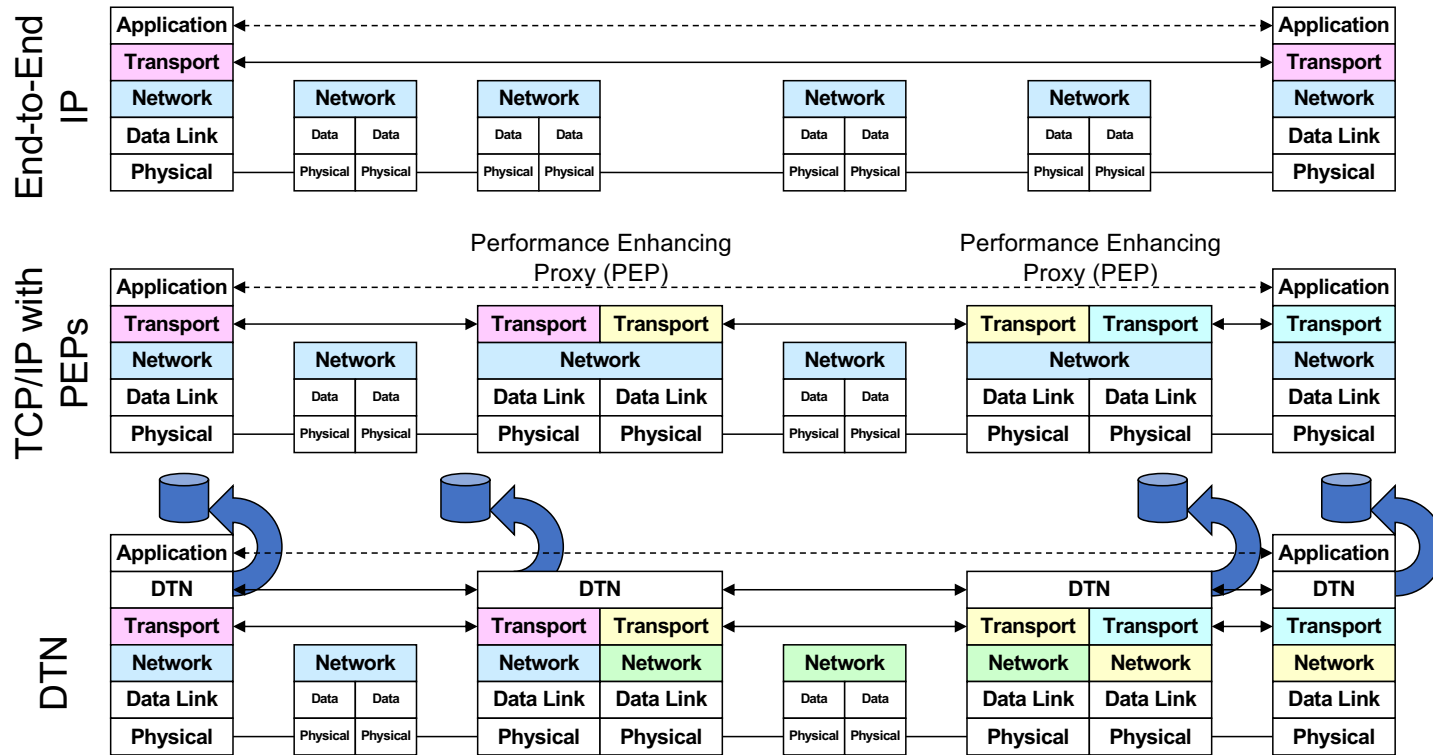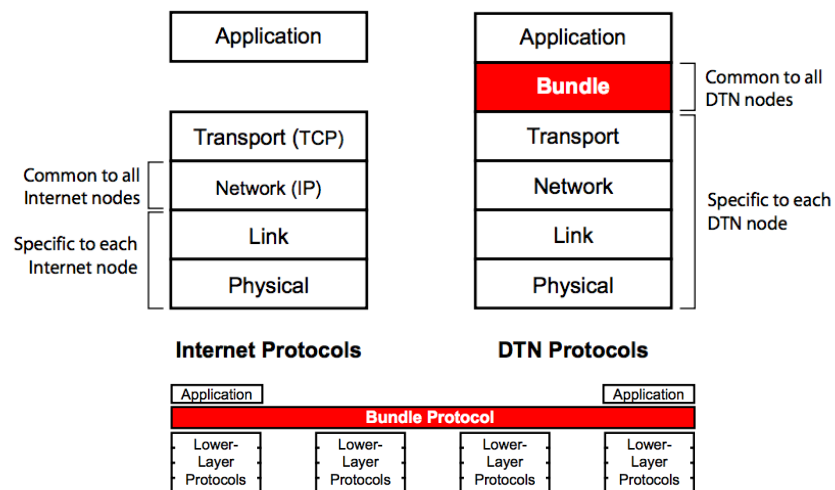
DTN vs IP: the Model

The DTN architecture is much like the architecture of the Internet, except that it is one layer higher in the protocol stack.

The DTN analog to the Internet Protocol (IP) is called "Bundle Protocol" (BP).

Refer to *Delay and Disruption-Tolerant Networks (DTNs) A Tutorial - Version 3.2*

DTN is standardized by the Internet Engineering Task Force (IETF) in RFC4838 and the Consultative Committee for Space Data Systems (CCSDS) in 734.1-B-1 and 734.2-B-1, supported by open-source software that can help users implement the new capabilities.

DTN Features (1 of 2)

The DTN protocol suite was designed for disrupted, mobile, airborne, and near-Earth and deep-space communications, featuring:

Internetworking layer	• The DTN Bundle Protocol is designed to overcome the problems associated with intermittent connectivity, long or variable delay, asymmetric data rates and high error rates.
Intra-network retransmission	• BP "custody transfer" and reliable convergence-layer protocols improve efficiency of end-to-end delivery by leveraging local link information and tighter local control loops.
Disruption-tolerant routing services	• Can take advantage of scheduled and expected future connectivity, in addition to current connectivity, and can interoperate with or without a terrestrial routing infrastructure.
Quality of service mechanisms	• Provide user control over the order in which traffic is served by the overlay internetworking layer and are independent of the underlying network segments.

DTN Features (2 of 2)

Security features	• Protect the infrastructure from unauthorized traffic and provide standard end-to-end security capabilities (e.g. integrity, confidentiality) and 'over-the air' key management.
Network management system	• Configure, monitor, and provide accounting for traffic passing through the system in asynchronous fashion, tolerating long network round-trip times.
International standardization	• IETF & CCSDS

DTN Benefits

Improved Operations and Situational Awareness

> The DTN store-and forward mechanism and automatic retransmission assure data delivery and reduce the need for ground-based commanding.

Interoperability and Reuse

> The standardized DTN protocol suite enables interoperability of international and commercial partner communication assets, encourages commercialization, and allows NASA to reuse technology to support future missions.

Space Link Efficiency, Utilization and Robustness

> DTN enables more reliable and efficient data transmissions resulting in more usable bandwidth. DTN also improves link reliability by multiplexing across network paths.

Security

> The DTN Bundle Protocol Security mechanisms enable integrity checks, authentication and encryption.

Quality of Service

> The DTN protocol suite allows for different priority levels to be set for different data types, ensuring that the most important data are received ahead of less important data.
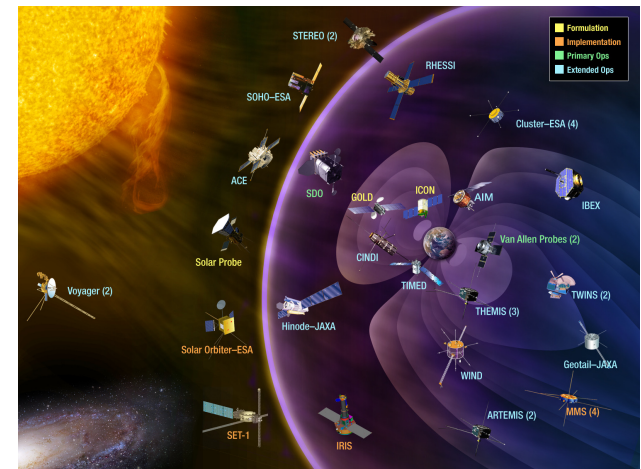
Case Study: DTN & Terrestrial Applications

The SSI can be viewed as a very large (in both distances and numbers) Internet of Things (IoT) that need to communicate:

- Orbiters, landers, probes, sensors, robots – many more "things" than humans.
- When the opportunity to communicate arises, DTN allows for automated, assured communications.
- With growth in the commercialization of space, we need more automation.

The evolving terrestrial IoT doesn't typically need immediate response and constant communication:

- Wearables, home automation, trending data, etc.
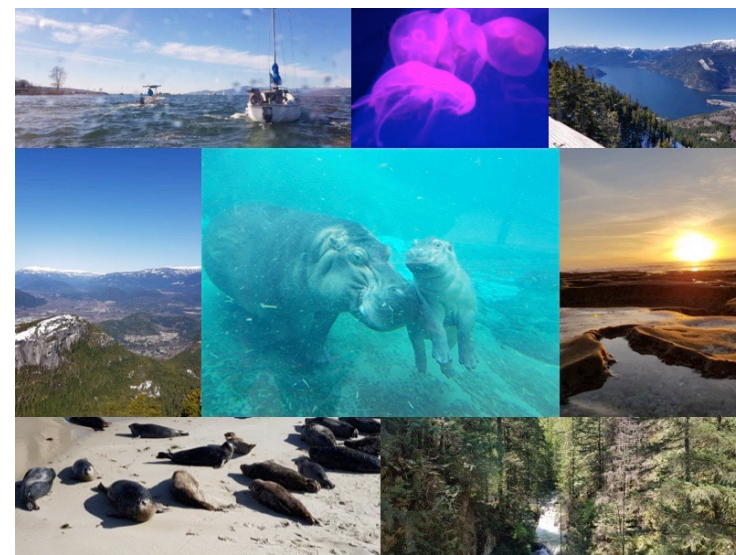- Some node storage is necessary for DTN, but that is fairly cheap.

Case Study: DTN & Terrestrial Applications

Other terrestrial use-cases could benefit from DTN:

o **Sea-going vessels**
o **Outdoor explorers**
o **Wildlife research, etc.**

As the terrestrial IoT grows (in number and in geographic distribution), more infrastructure will be required to maintain all of the communication links. DTN is <u>an alternative to growth in infrastructure</u> => automated, opportunistic, efficient use of existing links with assured delivery.

Topics discussed this morning:

- o **Space communication overview**
- o **Internet communication overview**
- o **Interplanetary Internet**
- o **Delay-Tolerant Networking (DTN)**
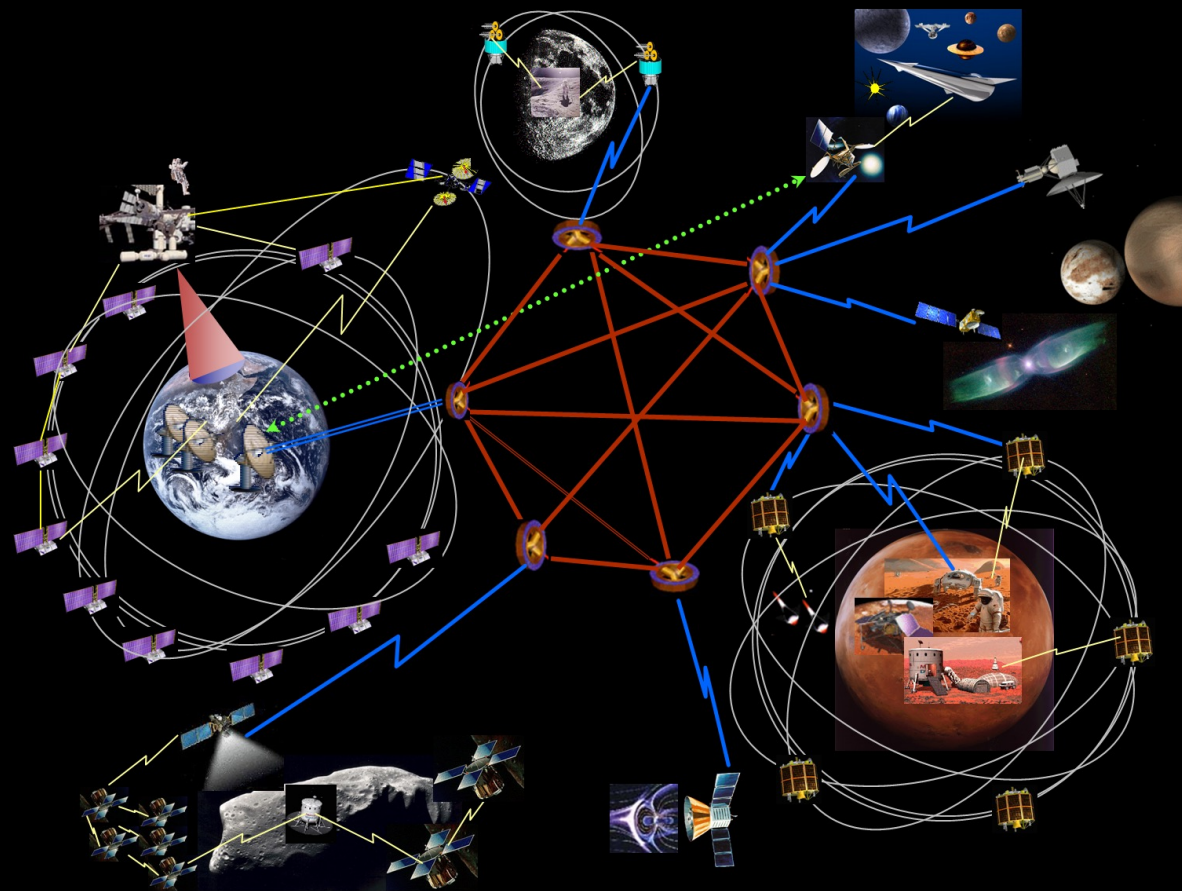- o **Solar System Internet**

Any questions?