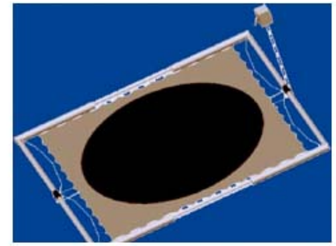
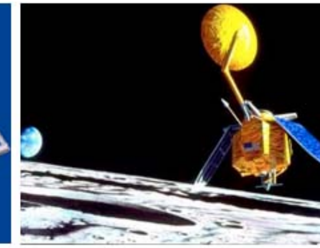
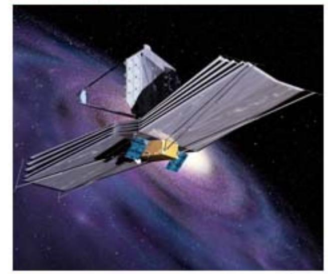
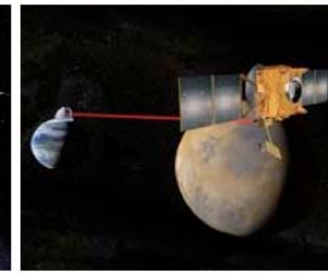
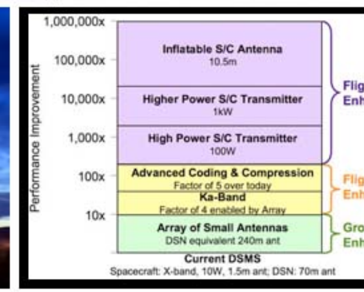
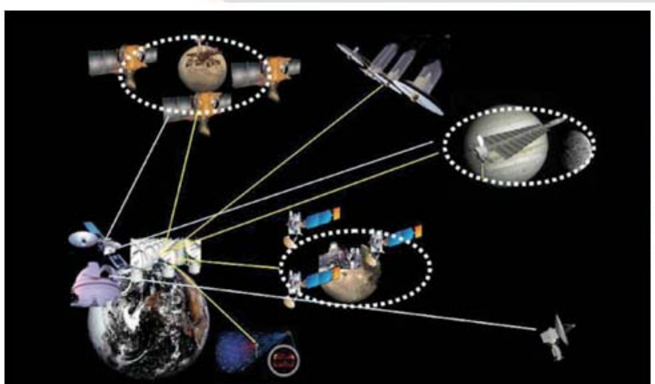
SpaceOps 2008

Delay-Tolerant Networking for Space Flight Operations: Design and Development

Scott Burleigh
Jet Propulsion Laboratory, California Institute of Technology





DTN Design and Development Motivation

- **Large-scale future space exploration will offer complex communication challenges that may be best addressed by establishing a network infrastructure.**
- **The Internet protocols are not well suited for operation of a network over interplanetary distances; a *Delay-Tolerant Networking* (DTN) architecture has been proposed instead.**
- **DTN is now a rapidly growing research field, but most implementations are mainly aimed at supporting applications of DTN technology to terrestrial networking problems. Those implementations are not necessarily suitable for deployment in an interplanetary network.**
- ***Interplanetary Overlay Network* (ION) is an implementation of the DTN architecture that is specifically designed for use in resource-constrained embedded systems, such as interplanetary robotic spacecraft.**

DTN Design and Development
DTN Protocol Implementations

- **Bundle Protocol**
  - DTN2 reference implementation (UC Berkeley)
  - DASM for Symbian mobile phones (U. of Helsinki)
  - C# (.NET) implementation at Georgia Tech.
  - Java implementation at Ohio U.
- **Licklider Transmission Protocol**
  - Java implementation at Ohio U.
  - C++ implementation at Trinity College Dublin, Ireland
  - Experimental flight implementation in C at Johns Hopkins University, Applied Physics Laboratory

DTN Design and Development

Flight Environment Constraints on DTN (1 of 2)

- **Link constraints: wireless links enabling interplanetary network communication are generally slow and are usually asymmetric.**
  - **Limited electrical power, relatively small antennae.**
    - **So signals are weak. This limits transmission to rates on the order of .25 Mbps to 6 Mbps.**
    - **Additionally, reception sensitivity is limited. Reception data rates are typically even lower, on the order of 1 or 2 Kbps.**
  - **So the cost per octet of data on the links is high, and the links are heavily subscribed.**
  - **Economical use of reception and transmission opportunities is important.**

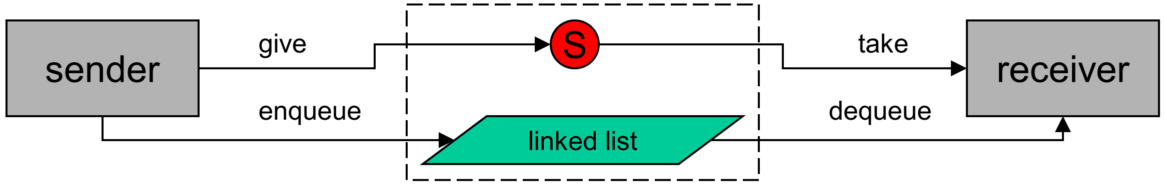
DTN Design and Development

Flight Environment Constraints on DTN (2 of 2)

- **Processor constraints:**
  - Limited electrical power, limited mass allowance.
  - Intense radiation environment, mandating radiation-hardening, which is time-consuming and expensive.
  - Relatively small market, limiting incentive to do rad-hardening engineering for the latest advances in processor technology.
  - So flight processors are always slower than engineering workstations.
  - So the cost per processing cycle is high and the processors are heavily subscribed.
  - Economical use of processing resources is important.
- **Hands-on repair is impossible, so reliability is key.**
  - Predictability enhances reliability, so flight software usually must meet hard real-time deadlines. So real-time operating systems are used, which historically has precluded protected memory: all software runs in “kernel” (rather than “user”) mode.
  - Dynamic allocation of system memory is difficult to predict, so it is typically prohibited except in certain well-understood spacecraft states, e.g., start-up.

DTN Design and Development
ION Design Principles (1 of 2)

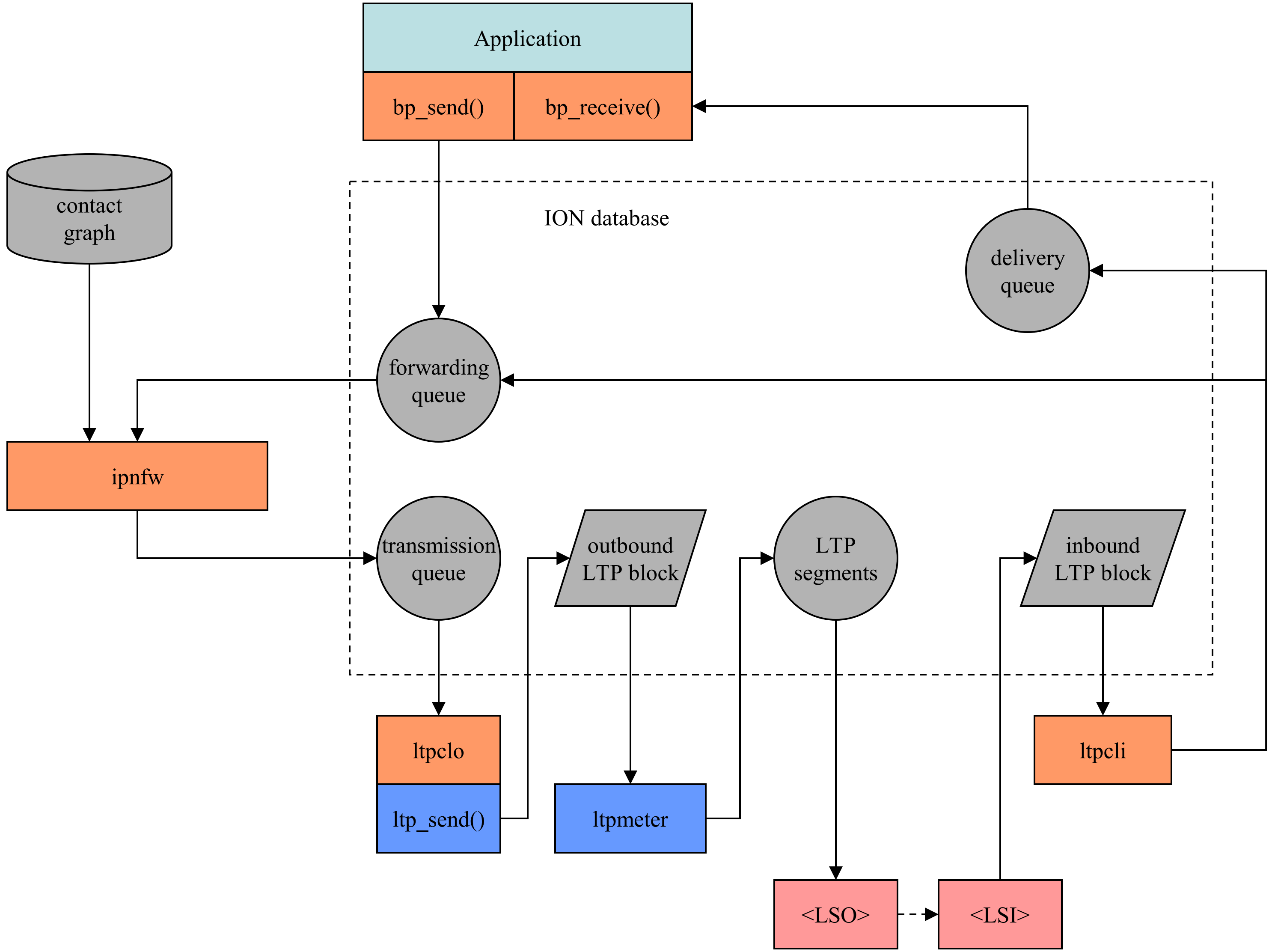
- **Use shared memory.**
  - **Often there's no protected memory, so we have no option.**
  - **But this can be turned to advantage: shared memory is a highly efficient way to pass data between flight software tasks.**



- **Zero-copy procedures: leverage shared memory to minimize processing overhead.**
  - **Encapsulation in layers of protocol overhead (headers and trailers) can be done by reference rather than by copy.**
  - **The same data object can be shared by multiple tasks, provided reference counting prevents premature deletion.**
- **Portability: this is an unfamiliar programming model, so we must make it easy to develop in an environment with good programming support (e.g., Linux) and then deploy – without change – in the target RTOS environment.**

DTN Design and Development
ION Design Principles (2 of 2)

- **Highly distributed processing**



DTN Design and Development Software Elements (1 of 3)

- **Interplanetary Communication Infrastructure (ICI)**
  - "platform" library insulates ION software elements from the differences among operating systems.
  - Personal Space Management (PSM) enables flexible, dynamic private management of a fixed block of pre-allocated system memory.
  - Memory Manager system enables coexistence of multiple memory management instances (e.g., multiple PSM-managed partitions).
  - Lyst and SmList systems standardize management of linked lists in private and shared memory.
  - Simple Data Recorder (SDR) enables flexible, dynamic private management of a fixed block of non-volatile storage, such as battery-backed memory or a pre-allocated file in a flash file system.
  - Zero-Copy Objects (ZCO) system enables protocol encapsulation by reference rather than by copy and provides a reference counting system to enable safe concurrent access to a single non-volatile storage object by multiple tasks.

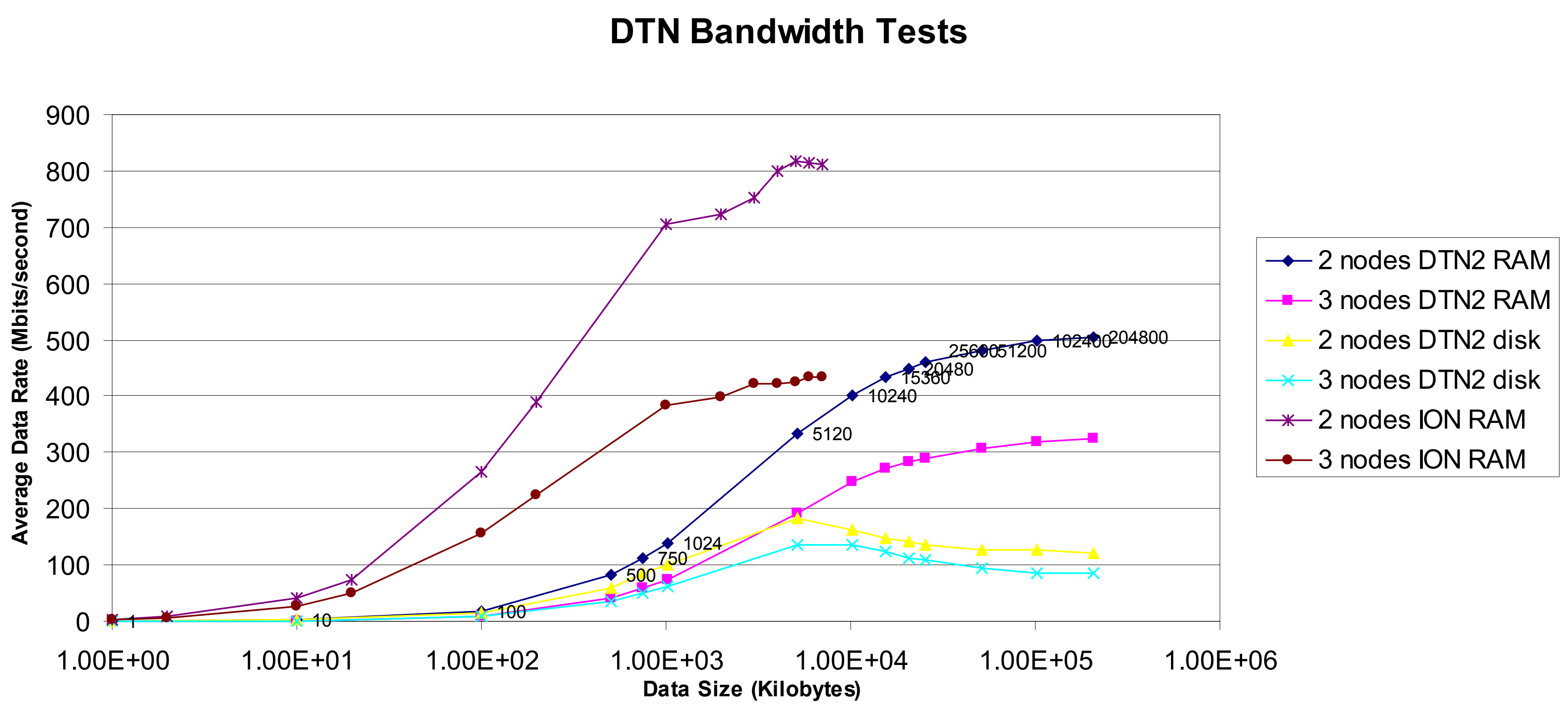
DTN Design and Development Software Elements (2 of 3)

- **Licklider Transmission Protocol**
  - **Full implementation of the LTP spec as developed by the DTN Research Group.**
  - **Additional features:**
    - **Aggregation of multiple service data units into a single block, to minimize the volume of acknowledgment traffic over highly asymmetric links.**
    - **Implements delay-tolerant, non-conversational flow control based on limiting block size and the number of transmission sessions that can be in progress concurrently.**

DTN Design and Development Software Elements (3 of 3)

- **Bundle Protocol**
  - Full implementation of the BP spec as developed by the DTN Research Group. (RFC 5050)
  - Includes support for:
    - Prioritization of data flows
    - Bundle reassembly from fragments
    - Flexible status reporting
    - Custody transfer
  - Additional features:
    - Rate control provides support for congestion forecasting and avoidance.
    - Bundle headers are compressed, to reduced protocol overhead and improve link utilization.
  - Also includes an implementation of Contact Graph Routing, a system for dynamic routing over interplanetary links.

DTN Design and Development Benchmarking results



ION flight software footprint: about 708 kilobytes including SDR database management system.

(These tests were run on a gigabyte Ethernet – don't expect this kind of performance in flight!)

DTN Design and Development Future Work

- **Plan to add an implementation of unacknowledged ("Class 1") CCSDS File Delivery Protocol (CFDP) that is integrated with ZCO for high performance.**
- **Studies for a delay-tolerant Network Time Protocol, enabling automatic synchronization of clocks among all nodes of an interplanetary DTN, are under way.**
- **Need to add an implementation of Bundle Security Protocol.**