



# Naming in Delay-Tolerant Networking

Scott Burleigh

Jet Propulsion Laboratory

26 July 2012



# Overview

---

- Why we need the concept of “node ID” in DTN.
- Where node IDs should replace endpoint IDs.
- Candidate notations for node IDs.



# Nodes and Endpoints

---

- In DTN (RFC 5050 Bundle Protocol spec):
  - A *node* is any entity that can send and receive bundles, comprising:
    - Application agent
    - Bundle protocol agent
    - Convergence layer adapters
  - An *endpoint* is a set of nodes identified by a string, an *endpoint ID* (EID). An endpoint can encompass any number of nodes, and a node can be a member of (*registered in*) any number of endpoints.



# Node Identification

---

- Each node is required to be a member of at least one “singleton” endpoint, so the EID of that endpoint effectively identifies that node so long as the node is registered in that endpoint.
- However, nodes are not required to be permanently registered in any endpoint; registrations can be commenced and terminated (RFC 5050 section 3.3).
- So, for a given node, no string is guaranteed to identify that node throughout its existence.



# Why This Matters

---

- Bundles are delivered to applications at endpoints.
- But most other operations on bundles are actually performed with reference to nodes, not endpoints:
  - Custody transfer occurs between nodes, not endpoints.
    - For example, RFC 5050 section 5.10 says “The conditions under which a node may accept custody of a bundle....”
  - In practice, we route on the basis of nodes, not endpoints.
    - Delivery predictability in PROPHET, for example, relates to nodes.
  - All of the Bundle Security Protocol mechanisms actually operate on the basis of nodes, not endpoints.
    - For example, section 1.2 of RFC 6257 defines "security source" and "security destination" as nodes.



# Problems

---

- Trying to use endpoint IDs to serve as node IDs errs in two ways:
  - It's technically infeasible, as noted above. It currently “works”, but only by convention: the foundation on which many critical DTN functions rests is really just good will and out-of-band consensus on the part of node administrators.
  - It's overkill. Endpoint identifiers must be fine-grained to enable demultiplexing to multiple applications in the application agent. Node identifiers can be coarse, as they only need to identify nodes. The extra information in an EID wastes bandwidth when bundles are transmitted.



# Solution

---

- Let's add the concept of an explicit "Node ID" – a unique and permanent node name, of which each node must have exactly one – to the BP specification, defining it in a new section of the specification corresponding to section 4.4 defining "Endpoint IDs".
- Let's replace "endpoint ID" with "node ID" in the DTN protocol specifications wherever the entity under discussion is in fact a node rather than an endpoint.



# Where Would Node IDs Be Used?

---

- Bundle primary header
  - Current custodian
  - Source (any entity that can send a bundle is a node)
  - For most unicast traffic, paired with accompanying demux to form Destination and/or Report-to endpoint ID
    - Override in an EID extension block where necessary.
- BSP abstract security block
  - Security source
  - Security destination
- Network management protocols, routing protocols



# What Would a Node ID Look Like?

---

- We could use ASCII or internationalized character strings, as is currently done by convention when the “dtn” URI scheme is used for EIDs.
- We could use variable-length integers, compressed in SDNVs, as is done when the “ipn” URI scheme is used for EIDs and CBHE is used at the convergence layer.
- Numeric node IDs are more efficient for processing, storage, and transmission, but harder to remember.
- Either way, need a universal, central node ID registry.