Asynchronous Management Architecture (AMA)

Edward Birrane
Edward.Birrane@jhuapl.edu
443-778-7423

JOHNS HOPKINS UNIVERSITY
Applied Physics Laboratory

Motivation

We cannot deploy challenged internetworks until we can manage them.

- Use cases for DTNs are emerging:
  - Handle signal propagation delay (space and some underwater).
    - *Mostly for space and underwater scenarios*
  - Handle frequent link disruptions.
    - *Mostly for disaster and some vehicular scenarios.*
  - Handle frequent link-access disruptions.
    - *Mostly for oversubscribed/congested links.*
    - *Link removed as matter of policy/administration and not physics.*
- All preclude human-in-the-loop network management
  - Nodes operate on “far side” of delayed/disrupted links.
  - Disruptions occur from attenuation, tasking, power, and pointing
  - Network management starts looking more like fault management.
    - *Maintain ability to relay information from critical assets both on-board and remotely without access to direct operator intervention.*

History

- Examined uniqueness of the problem, 2011-2013
  - Some early pubs defining the problem as related to DTN
    - *Birrane, E, & Cole, R. (2011). Management of Disruption-Tolerant Networks: A Systems Engineering Approach.*
    - *E. Birrane, S. Burleigh, V. Cerf, "Defining Tolerance: Impacts of Delay and Disruption when Managing Challenged Networks,"*
    - *E. Birrane, H. Kruse, "Delay-Tolerant Network Management: The Definition and Exchange of Infrastructure Information in High Delay Environments"*
  - Reviewed popular engineering approaches
    - *Autonomous fault protection schemes*
    - *Mobile code and scripting schemes*
    - *Spacecraft telemetry schedules*
    - *Deterministic rule-based expert systems*
- Delay-Tolerant Network Management Protocol (DTNMP) 2013
  - Published to DTNRG, Initial implementation by NASA
  - Utility outside of NASA network management
- Renamed as Asynchronous Management Protocol (AMP) 2015
  - Submitted as set of IDs to DTNWG.
  - Extracted AMA as set of requirements/properties.

How Do We Manage Networks Today? Do we need a new thing?

Low-latency approaches to network management fail to scale with increasing delays and disruptions.

- Rich set of evolving capabilities
  - Simple Network Management Protocol (SNMP).
    - *Pull model of information from managed devices.*
      - Support for “traps” to push unreliable notifications of pre-defined events.
  - Network Configuration Protocol (NETCONF).
    - *XML-based, session-based remote-procedure call (RPC) interface for node configuration.*
  - Remote Network Monitoring MIB (RMON).
    - *Mechanisms for exchanging network monitoring data.*
- Poor scaling with delays, disruptions, or commanding
  - Focus on getting data to operators.
  - Less focus on in-situ response options.
  - Reliance on scripting and mobile code which is not always a deployment option.

Service Definitions

Performance monitoring (reporting) is one of many network management requirements.

- Parameterized Control
  - Detection, diagnosis, reporting, correcting failures.
  - Example: Monitor-Response Autonomy
- Configuration.
  - Update the behavior of functions within the network remotely.
  - Update/reconfigure systems based on local state and time.
- Administration.
  - Apply access control lists, security settings, and other methods filtering management function by role.
- Reporting
  - Report network conditions to operators and other nodes in an internetwork based on local state and time.

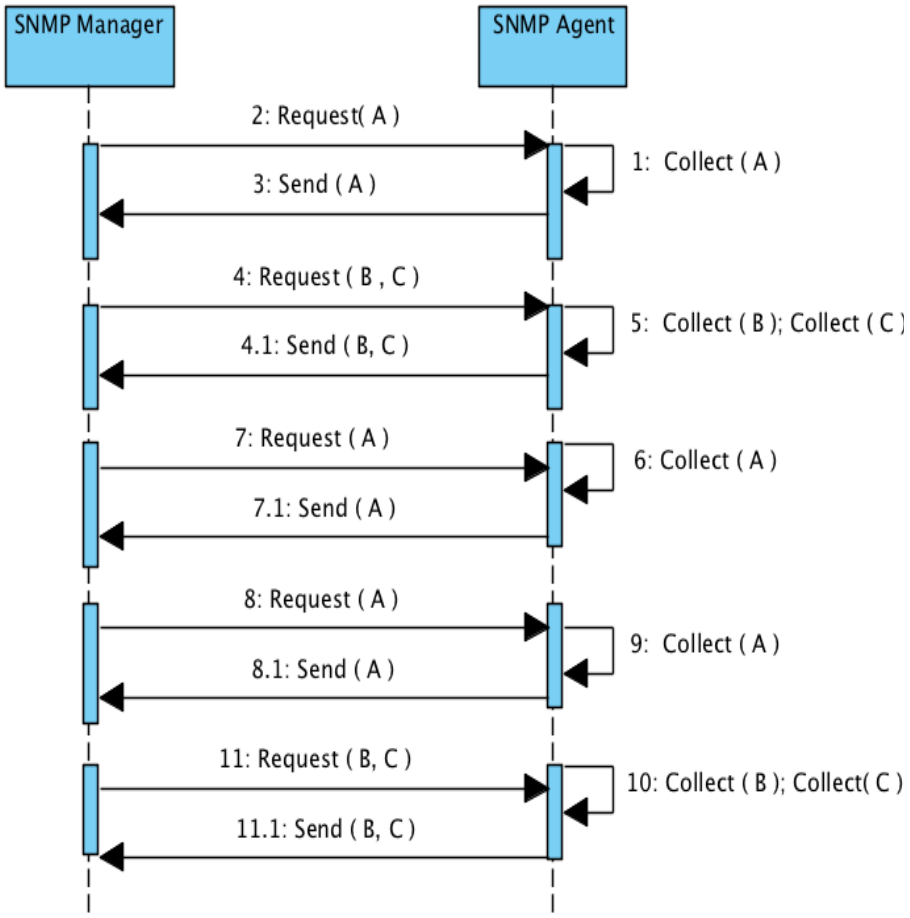
Desirable Properties

- Intelligent Information Push
  - Round-trip pull requests difficult in some deployments
- Minimize Message Size, not Node Processing
  - Smaller messages work for everybody
- Specific Data Identification
  - Do not waste transmissions exchanging synchronizing data. For example, support associative lookups, not table key dumps followed by index queries.
- Tactical Data Definition
  - Define reports with high fidelity.
- Autonomous Operation
  - Deterministic, monitor-response systems
  - Avoid reliance on mobile code which can be problematic

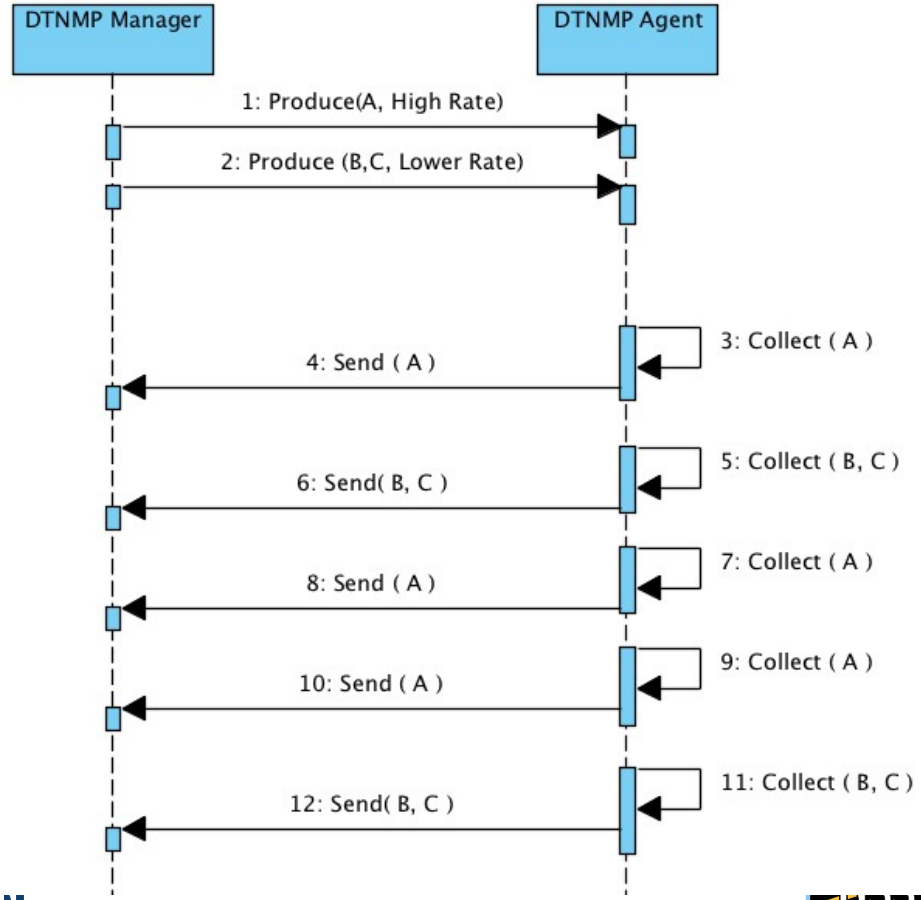
Push, don't Pull.

(extreme) Example: Collect A at high rate, Collect B,C at lower rate.

SNMP (PULL)



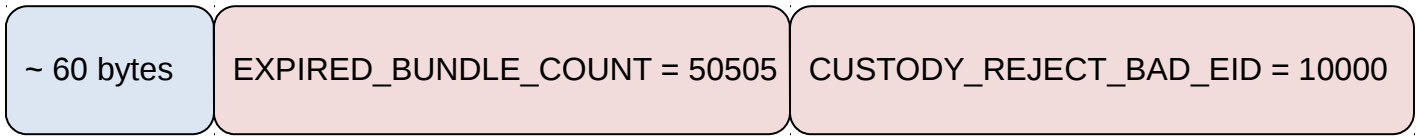
AMP (PUSH)



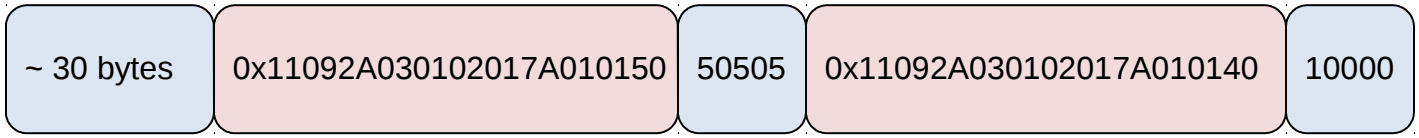
Keep Message Sizes As Small As Practical

Currently recommend pre-shared schemes and binary encoding.

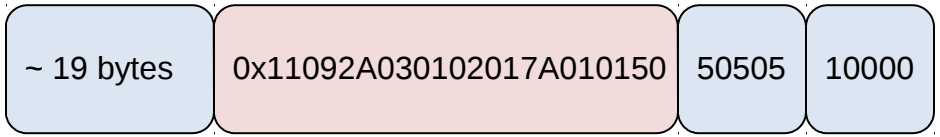
Fully Named ASCII Data (Good)



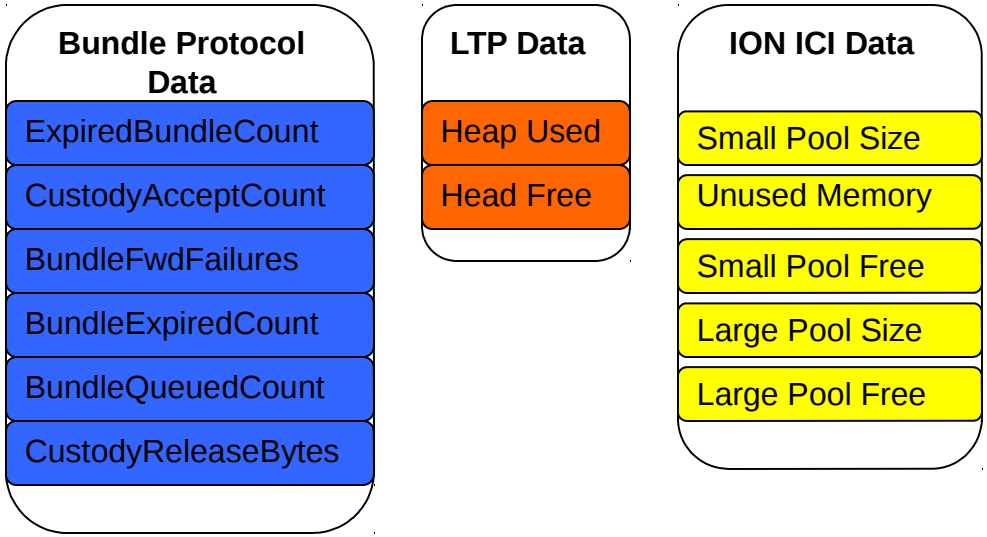
Fully Named Binary Data (Better)



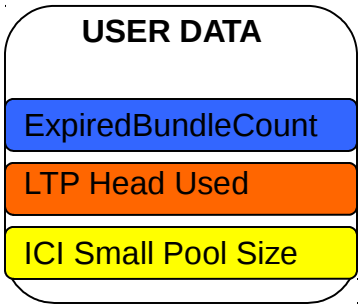
Summary Named Binary Data (Best)



Specific, Tactical Data Definition



- Pre-defined sets of data
  - BP, LTP, ICI
  - Query individual items
- Pre-defined collects per set
  - All BP Disposition
  - All LTP Stats
  - All ICI SDR Stats



- How to mix/match across data sets?
  - ExpiredBundleCount + Head Used + Small Pool Size
  - Could make 3 queries (3 sets of NAME=VALUE)
    - This is wasteful from previous slide)
  - Define new report to represent 3 values
    - 1 NAME, 3 VALUES
    - More bandwidth efficient

Application Data Model

ADMs are, notionally, a superset of MIBs. Recommend they be specified in YANG.

- Atomic Data and Controls.
  - Well-defined data definitions.
    - *A = <firmware-sampled value>*
  - Well-defined, parameterized command opcodes.
    - *SetNewTemperature(float NewTemp, uint Deadline)*
- Literals and Operators.
  - Custom constants can be defined per-ADM.
    - *PI = 3.14159*
  - Special operators (unary, binary, and more) can be defined.
    - *Pow(x,y), Avg(A,B,C,D), etc…*
- Computed Data.
  - Runtime-calculated, typed data: CD = Data OP Data
    - *CD1 = A + B; CD2 = pow(A,C1);*
- Collections.
  - What pre-defined collections of data values (reports) and control sequences (macros) have been created?

ADM Example

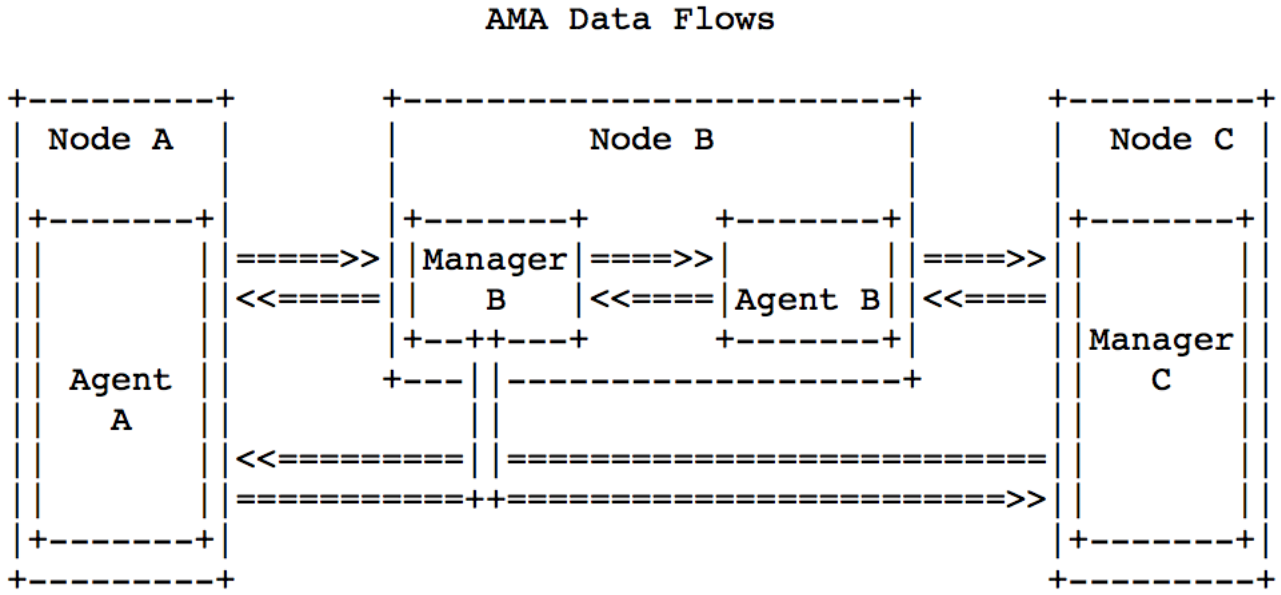
A sample ADM for an application implementing a stack.

"STACK" Application Data Model

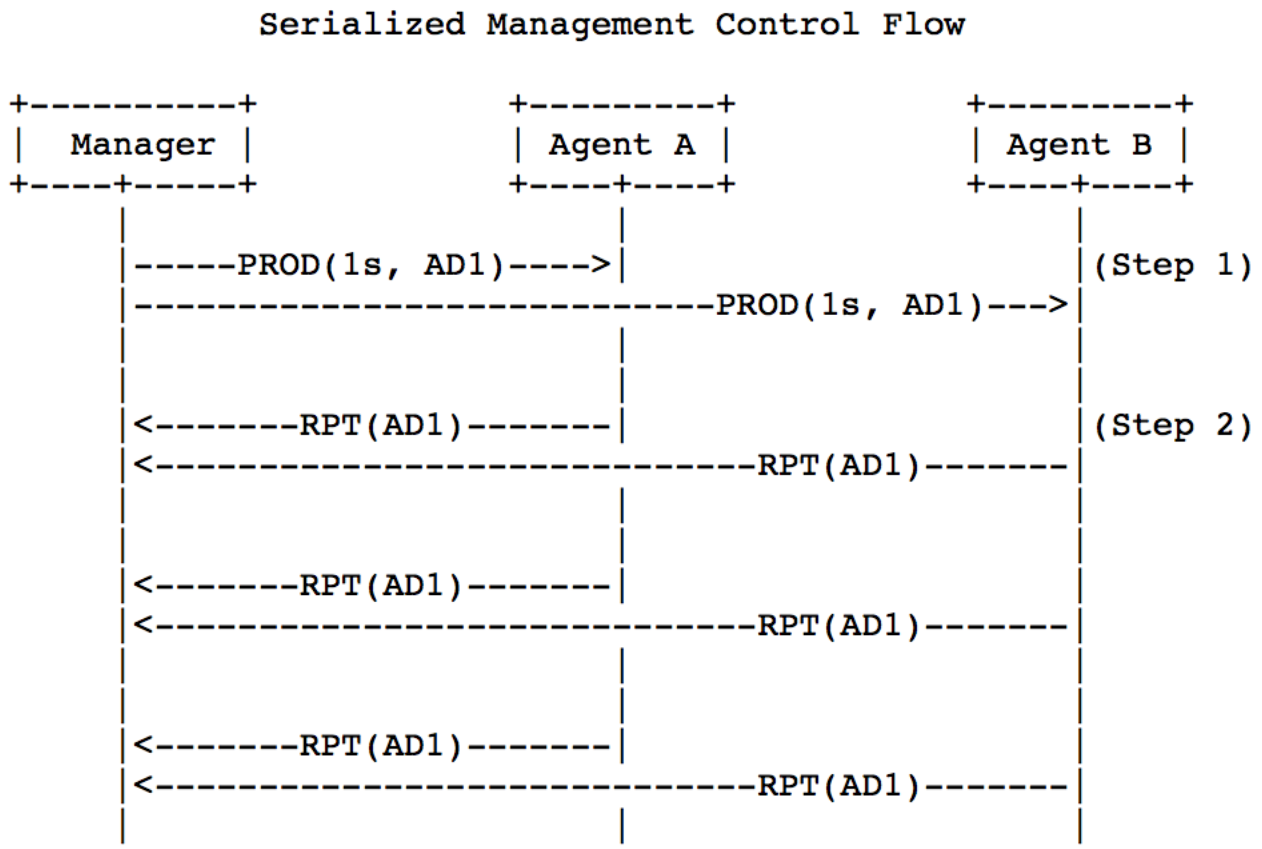
Atomic Controls	Computed Data	Atomic Data
- PUSH(X) - POP(X)	- Average POPs	- Stack Depth - Total Items - Total # POPs

Literals	Data Collections	Control Collections
- MAX_DEPTH = 10	Report 1: - Cur. Stack Depth - Total Items - Average POPs	EMPTY: Stack Depth > 0 POP(X)

AMA Roles and Responsibilities

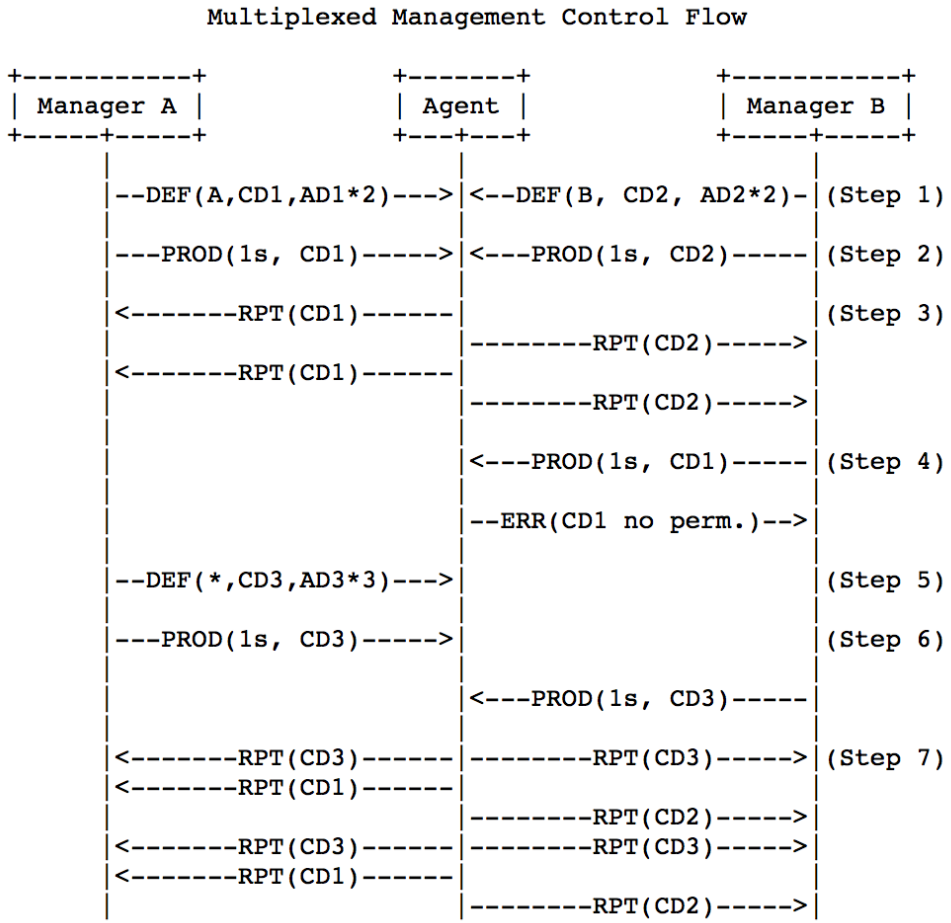


AMA Basic Data Flow

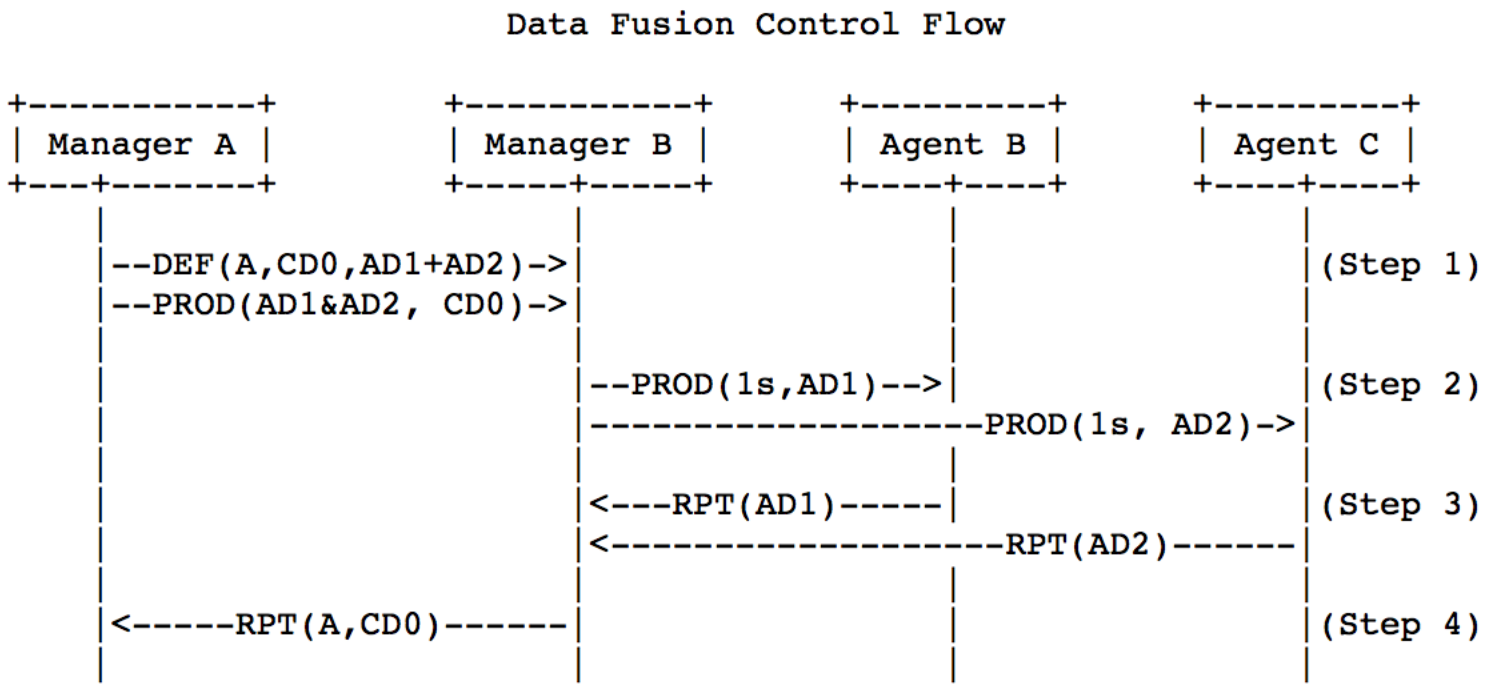


In a simple network, a Manager interacts with multiple Agents.

AMA Multi-Manager Flow



AMA Data Fusion Flow



Data fusion occurs amongst Managers in the network.

Current Status

- Related internet drafts
  - Asynchronous Management Protocol (AMP)
    - *Binary encoding*
    - *Uses any transport layer*
    - *https://tools.ietf.org/html/draft-birrane-dtn-amp-02*
  - ADMs
    - *https://tools.ietf.org/html/draft-birrane-dtn-adm-agent-01*
    - *https://tools.ietf.org/html/draft-birrane-dtn-adm-bp-00*
    - *https://tools.ietf.org/html/draft-bsipos-dtn-amp-yang-00*
- Reference Implementations (AMP)
  - Reference implementation in ION open source, this summer

Thank you!

Questions?