JOHNS HOPKINS
APPLIED PHYSICS LABORATORY

Status of DTNMA Drafts

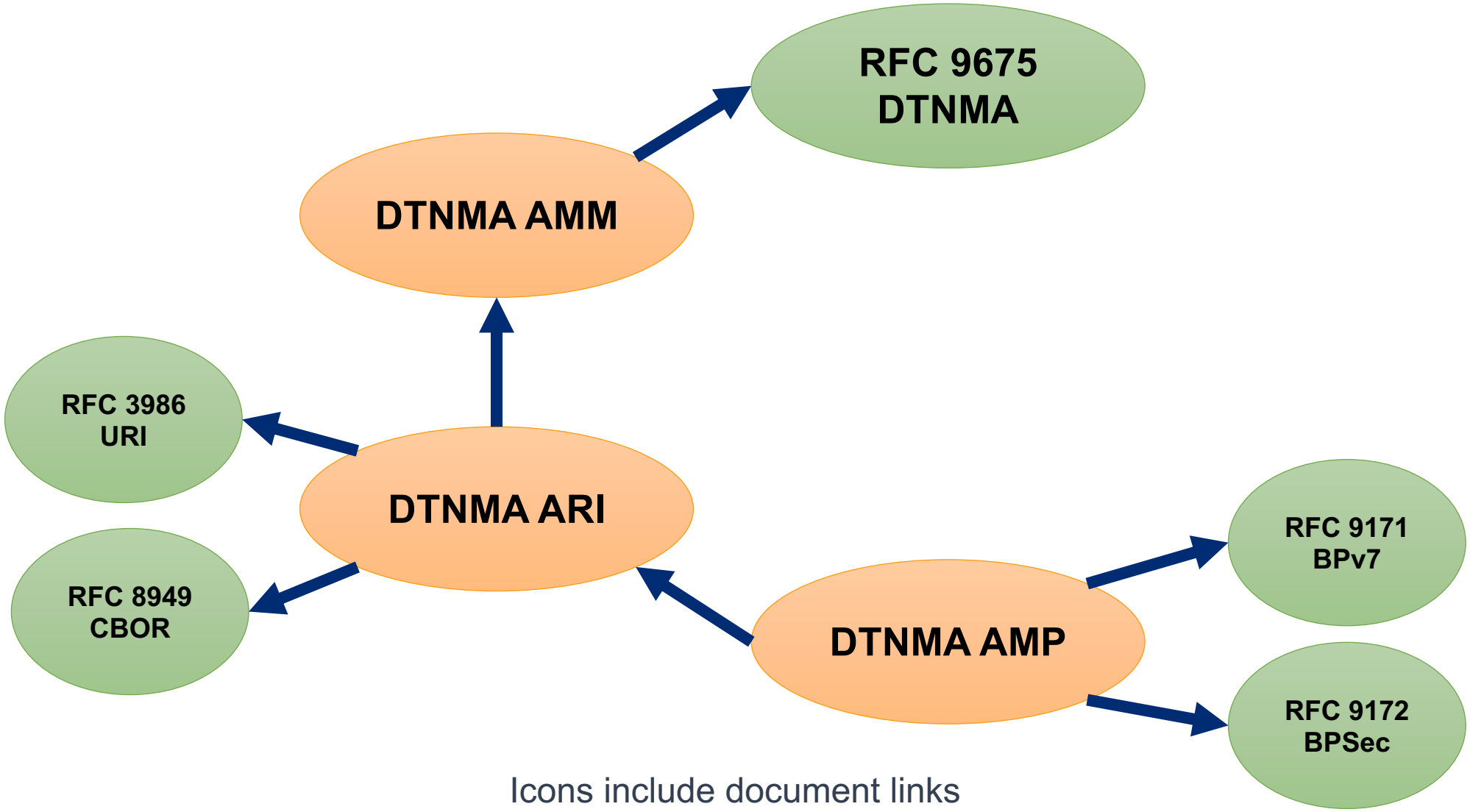
IETF 123 DTN WG

Justin Ethier and Brian Sipos
JHU/APL

Current DTNMA Drafts

- Application Management Model and Data Models (draft-ietf-dtn-amm-05)
  - Includes “namespace reference” as a full member of the built-in AMM type system
  - Latest update allows creation of IDENT object hierarchies
  - Allows introspection of semantic types within the AMM via a base IDENT object
  - AMM values may be compacted by eliding type information inferred by context of use
  - Extends list of ADM author considerations
- Application Resource Identifier (draft-ietf-dtn-ari-06)
  - ARI allows structuring of object reference parameters as either a list or a map
  - Adds recommendations for interoperability using existing, well-known media types
  - Expand IANA DTNMA object type registries to include range of reserved type values for experimental use
  - Includes EXECSET and RPTSET examples with comprehensive set of ARI variations
- Asynchronous Management Protocol (draft-ietf-dtn-amp-02)
  - Supports proxy transports for cases where the ultimate destination endpoint is not available
  - Requires an AMP message to contain one or more binary form ARI values

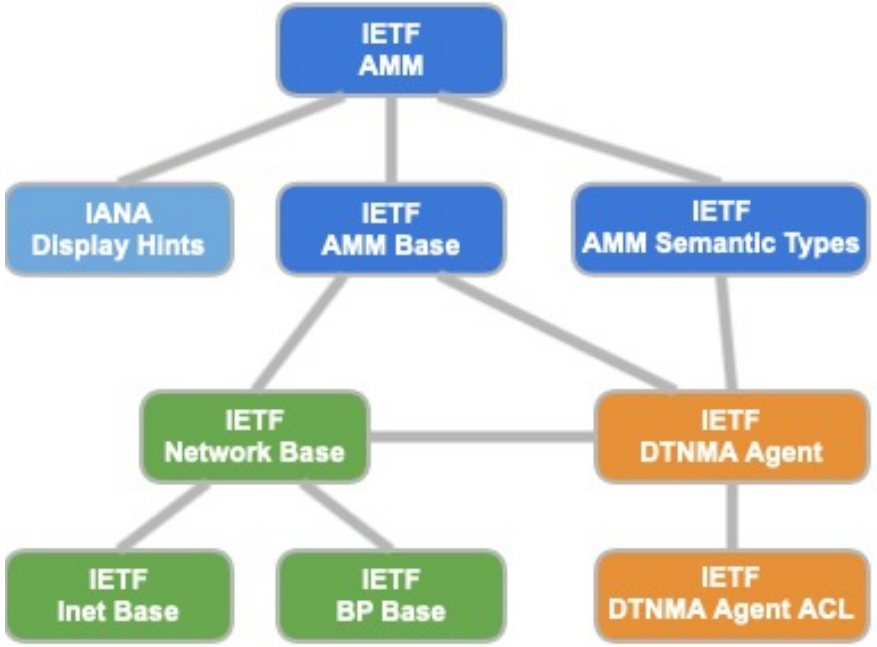
DTNMA Normative References



Icons include document links

Updates to Application Data Models

- Added new ADM's for:
  - AMM base types and semantic type introspection
  - AMM semantic types and unions
  - Base networking objects
- Types relocated out of ietf-amm, which remains purely as a definition of extension keywords
- Basic types relocated to `ietf-amm-base` ADM
- IP-related types relocated to `ietf-inet-base` ADM
- BP-related types relocated to `ietf-bp-base` ADM
- New definitions added to `ietf-dtnma-agent` for ODM introspection and maintenance of autonomy rules



ADM Author Considerations

- An “Author Considerations” section in the AMM document is meant to assist ADM authors
- These are not related to requirements of the AMM or ADM syntax, but best practices for usability and maintainability of models
- Highlights are summarized to the right

- Object life cycle recommendations: Current, deprecated, and obsolete
- EDD, OPER definitions need nilpotency
- CTRL definitions needs idempotency
- Choosing between CONST, VAR, or EDD
- Use Tables for Related Data
- Use of parameters for IDENT (extensible, data structures), CONST/VAR (avoid redundancy), EDD (associative lookups)
- VAR definitions need to consider state changes from the manager
- Choosing between VAR (in-agent) or application-managed state
- Application state visibility: When there is a CTRL changing application state, provide an EDD to inspect resulting state

Open Issues and Work

- Provide a full definition for ARI object reference patterns
- Adjust access control list (ACL) specifications to be fully implementable and extensible
- Ongoing work implementing remainder of existing ADM objects
- Adding model “features” to allow more lightweight agents to operate
- Follow-on ADM work: alarm state management, log management, host OS telemetry, etc.