JOHNS HOPKINS
APPLIED PHYSICS LABORATORY

Status of DTNMA Drafts

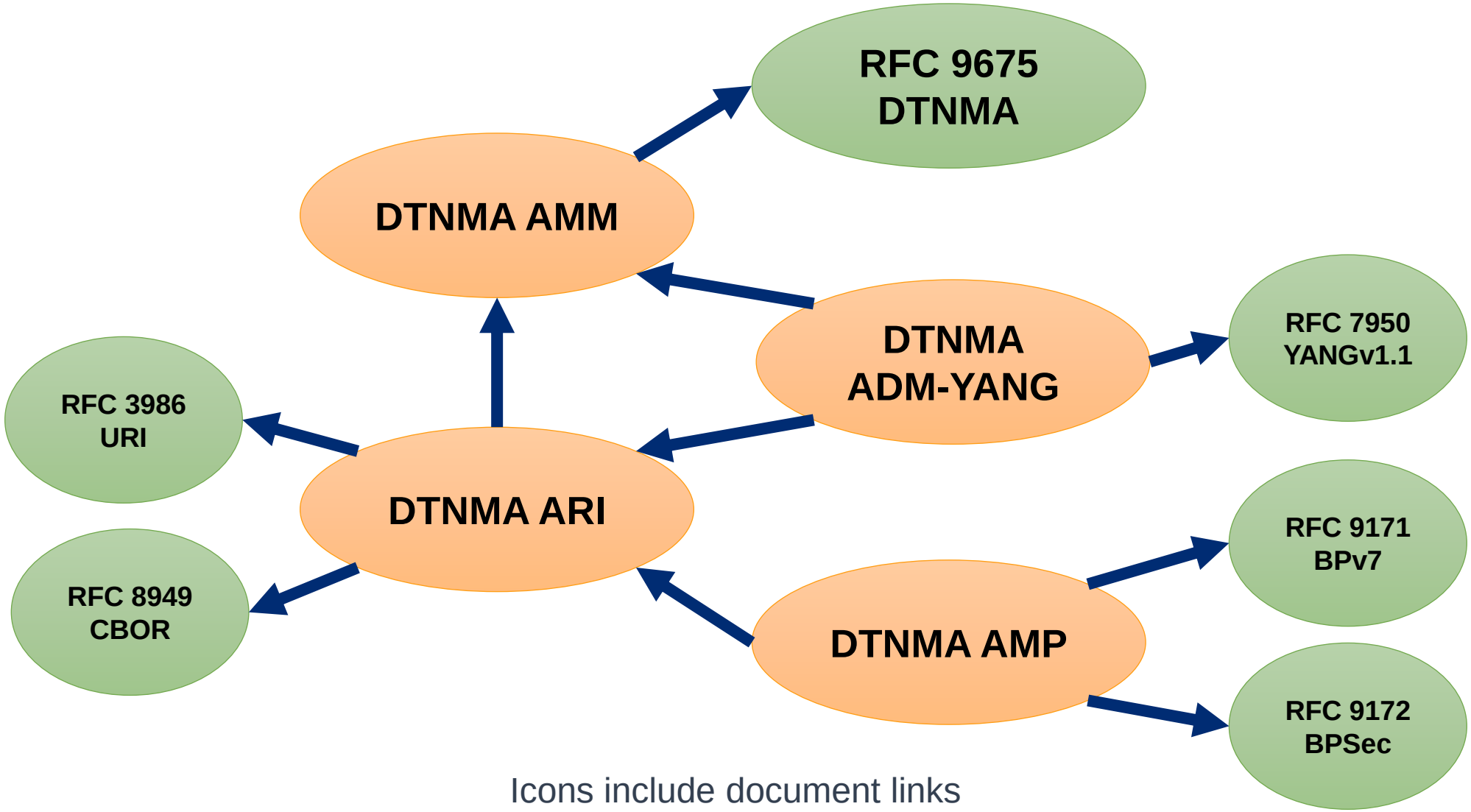
IETF 122 DTN WG

Jenny Cao and Brian Sipos
JHU/APL

Current DTNMA Drafts

- Application Management Model and Data Models (draft-ietf-dtn-amm-03)
  - Last update clarifies conditions for built-in types and some processing activities (`undefined` value handling)
  - Explains namespace breakdown into {organization ID, model ID, model revision}
  - Removes circular normative dependency on ARI draft (it is now informative)
- Application Resource Identifier (draft-ietf-dtn-ari-04)
  - Last update embodies the namespace breakdown and makes initial well-known organization registrations
  - This also clarifies the domain of identifier integer enumerations and text names
  - ARI now allows model revisions in binary form in a Manager, but not to/from and within an Agent
  - The “ari” URI scheme is now under provisional registration with IANA pending RFC publication
- Application Data Model YANG Syntax (draft-ietf-dtn-adm-yang-03)
  - Last update makes some clarifications to YANG direct reuse and updates embedded ARI namespaces
- Asynchronous Management Protocol (draft-ietf-dtn-amp-01)
  - Simple binding of ARI values onto BPv7 transport with version indication
  - Requirements on source authentication and data integrity
  - Last update was just to example ARIs and AMP messages

DTNMA Normative References



Icons include document links

DTNMA Namespace Logic

- Every data model (ADM and ODM) has a unique namespace
- Namespaces have component parts of: organization ID, model ID, model revision
- The organization ID part is ether a well-known, IANA-registered identifier or is private use
- The model ID part is either an ADM, possibly registered with IANA, or an ODM
- The model revision part is used only for ADMs and only by Managers
  - This allows disambiguating between multiple revisions over the lifecycle of an ADM
  - Not used explicitly by an Agent because it can only implement one revision at a time
  - The text form follows YANG suffix of “@YYYY-MM-DD” and binary form uses CBOR tag 1004
- All namespaces can be referenced by an ARI value
  - An well-known org. ADM namespace reference looks like `ari://example/adm-a/` or `ari://example/adm-a@2025-02-19/`
  - An private-org. ADM namespace reference looks like `ari://!private/adm-a/` or `ari://!private/adm-a@2025-02-19/`
  - An ODM namespace reference looks like `ari://example/!odm-a/`

Organization Registration

- The current ARI draft defines an initial table of well-known organizations based on current ADM drafts
- The registration procedure is a split between:
  - Negative values reserved for private uses
  - Expert review for low values and
  - First come first served for high values
- This is similar in concept and function to, but distinct from, the IPN allocator registry
- The expert review procedure *does not* require any specific external specification
  - Expect to have allocations for CCSDS / SANA
- Initial orgs are used for ADMs:
  - “ietf” for ADMs with behavioral objects
  - “iana” for ADMs with code point definitions (IDENT and TYPEDEF objects)
  - “example” for documentation

Enumeration Range	Registration Procedure
-2147483648 to 0	Reserved
1 to 65535	Expert Review
65536 to 2147483647	First Come First Served

Table 4: Namespace Organizations Registration Procedures

Enumeration	Name	Contact	Notes
-2147483648 to -2147483777			Reserved for Experimental Use
-2147483776 to -1			Reserved for Private Use
0			Reserved
1	ietf	IETF Chair <chair@ietf.org>	All IETF standardized models will use this organization
2	iana	IETF Chair <chair@ietf.org>	All IANA registry models will use this organization
3 to 65534			*Unassigned*
65535	example	IETF Chair <chair@ietf.org>	Reserved for example ADMs
65536 to 2147483647			*Unassigned*

Table 5: Namespace Organizations

Open Issues and Work

- Making a “namespace reference” a full member of the built-in AMM type system
- Relocating TYPEDEFs from the `ietf-amm` ADM
  - This would keep the AMM ADM purely as a definition of extension keywords
  - Basic types can be relocated to an `ietf-amm-types` ADM
  - IP-related types can be relocated to an `ietf-inet-types` ADM to separate the topic
  - BP-related types can be relocated to an `ietf-bp-types` ADM to separate the topic
- The module-import directed graph of ADMs needs more examination and design work
  - This is not a matter of *what* to implement regarding IDENT and TYPEDEF, just *where* they reside
  - Preference needs to be toward making derived IDENT/TYPEDEF modules optional
    - If you don’t implement TCP/IP or UDP/IP or DNS names, then don’t use the “inet-types” module