

# NMDA discussion

Yan Zhuang

Huawei Technologies

2017 August 28<sup>th</sup>

# NMDA changes in IETF

- A New datastore architecture
  - <https://tools.ietf.org/html/draft-ietf-netmod-revised-datastores>
- Changes to Netconf/restconf
  - Transport protocols should access proper datastores to configure/get data
  - <https://tools.ietf.org/html/draft-dsdt-netconf-restconf-nmda>
- Changes to YANG modules: Combine configuration/operational state tree into a single tree
  - Guidelines to design YANG modules following NMDA
  - <https://tools.ietf.org/html/draft-dsdt-nmda-guidelines>
  - Ongoing rfc6087bis: <https://tools.ietf.org/html/draft-ietf-netmod-rfc6087bis-13>

# Status of this work in IETF

- The NMDA Revised data store
  - WG Last Call
- The NMDA guidelines
  - WG document
- rfc6087bis:Guidelines for Authors and Reviewers of YANG Data Model Documents
  - WG document
- Revised ietf-interface module: rfc7223bis
  - Submitted as individual document, will be adopted by WG soon
- A WG document takes at least 1 year to be published as a RFC in IETF.

# NMDA module Example

## A new NMDA module:

Example: Create a New Module

Create an NMDA-compliant module, using combined configuration and state subtrees, whenever possible.

```
module example-foo {
  namespace "urn:example.com:params:xml:ns:yang:example-foo";
  prefix "foo-";

  container foo {
    // configuration data child nodes
    // operational value in operational datastore only
    // may contain config=false nodes as needed
  }
}
```

Combine configuration/operational state tree  
into a single tree

## Example: Convert an old Non-NMDA Module

Old Module:

```
module example-foo {
  namespace "urn:example.com:params:xml:ns:yang:example-foo";
  prefix "foo";

  container foo {
    // configuration data child nodes
  }

  container foo-state {
    config false;
    // operational state child nodes
  }
}
```

Converted NMDA Module:

```
module example-foo {
  namespace "urn:example.com:params:xml:ns:yang:example-foo";
  prefix "foo-";

  container foo {
    // configuration data child nodes
    // operational value in operational datastore only
    // may contain config=false nodes as needed
    // will contain any data nodes from old foo-state
  }

  // keep original foo-state but change status to deprecated
  container foo-state {
    config false;
    status deprecated;
    // operational state child nodes
  }
}
```

# Progress of groups

- BBF has started a team to deal with NMDA for their models
- For IEEE 802, a yangsters group is proposed and set by Glenn
  - Webpage: <http://ieee802dot1.wpengine.com/yangsters/>
  - Mailing list: [STDS-802-YANG@listserv.ieee.org](mailto:STDS-802-YANG@listserv.ieee.org)
  - One role of the group will be to discuss guidelines for developing YANG models in 802, including create a document that contains the high-level categories where guidelines are needed. Initial thoughts for guidelines are:
    - Structure
      - Example: How the IEEE YANG modules are organized
      - Example: Use of IETF NMDA
    - Coding
      - Example: Use of IEEE comment resolution process
      - Example: How revision dates are used
    - Naming
      - Example: naming conventions
    - Tooling
      - Example: repository usage, yang validation tools
- For 802.1, given the ietf-interface module move in IETF, they are starting to consider adoption/change of the 802.1 modules.

# Option 1

- Change modules to NMDA-style in D1.x.
- Update it if any changes during D2.x/D3.x.
  
- Advantage:
  - Less work for future maintenance.
- Disadvantage
  - Delay TF work progress and postpone draft move to WG ballot.
  - Changes due to the changes of IETF drafts.

# Option 2

- Keep the existing module style and move forward during Task force review.
- Review the NMDA work during WG ballot.
- Advantage:
  - Move our work forward.
  - Wait IETF work to be stable a bit.
- Disadvantage
  - Need model structure change if we decide to follow NMDA during WG review.

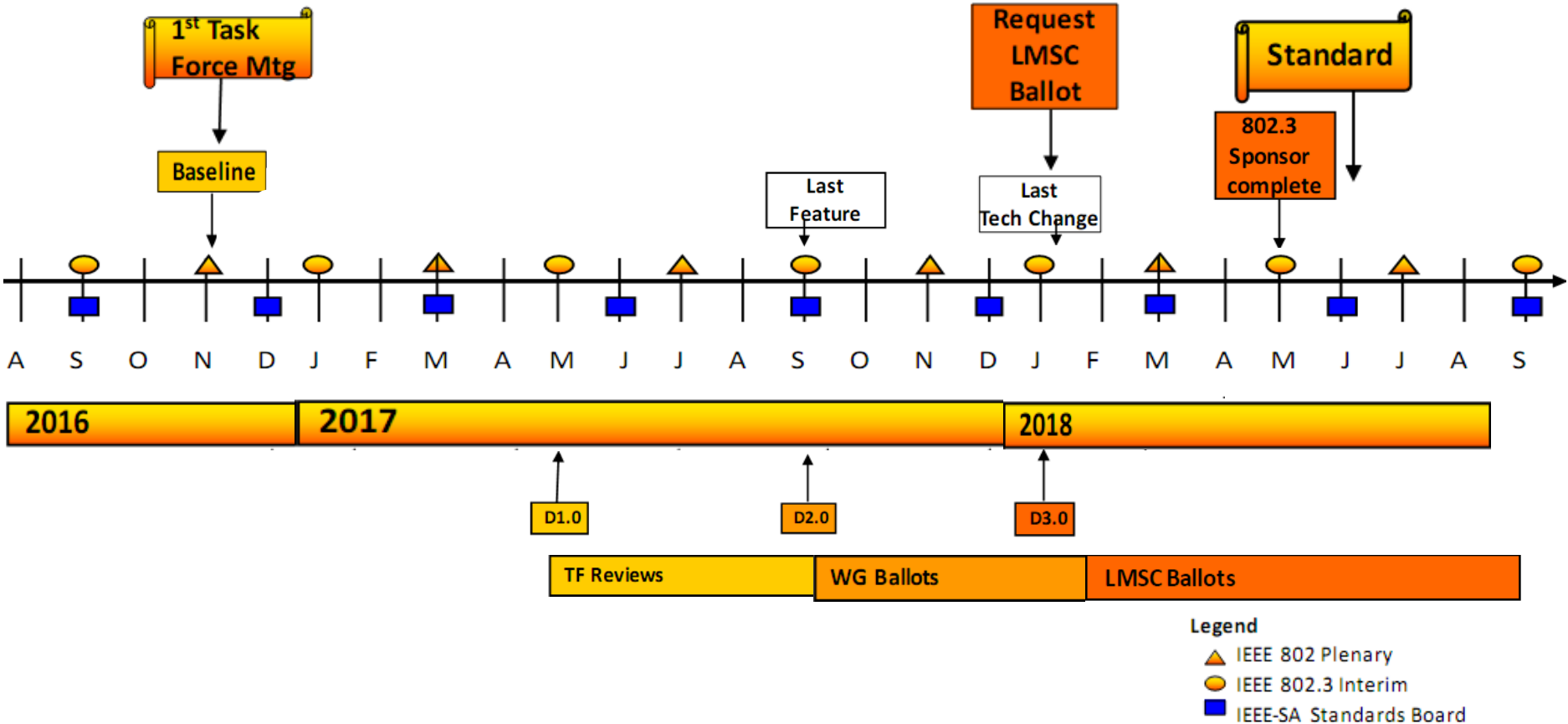
# Option 3

- .3cf Task Force will follow the non-NMDA style.
- Left NMDA work to future maintenance.
- Advantage:
  - Move our work forward.
- Disadvantage
  - More work for maintenance.
  - May not keep pace with other SDOs work( or 802.1YANG).



Thank you!

# Draft Timeline



Draft proposal

# NMDA Guidelines

- The specific approach to be taken for models being developed now and during the NMDA transition period should be based on both the expected usage and the maturity of the data model.
  - New models and models that are not concerned with the operational state of configuration information SHOULD immediately be structured to be NMDA-compatible.
  - Models that require immediate support for "in use" and "system created" information SHOULD be structured for NMDA. A non-NMDA version of these models SHOULD also be published, using either an existing model or a model created either by hand or with suitable tools that support current modeling strategies. Both the NMDA and the non-NMDA modules SHOULD be published in the same document, with NMDA modules in the document main body and the non-NMDA modules in a non-normative appendix. The use of the non-NMDA model will allow temporary bridging of the time period until NMDA implementations are available.