

# Guidelines for YANG module authors on using the new Network Management Datastore Architecture (NMDA)

draft-usdt-nmda-guidelines &  
draft-ietf-netmod-revised-datastores-03

Rob Wilton (Cisco), on behalf of NMDA authors

[rwilton@cisco.com](mailto:rwilton@cisco.com)

IETF 99, Prague, NETMOD WG, RTG WG

# Problem definition:

- Operator requirement for devices to clearly differentiate between:
  - What it is being **asked to do** – i.e. the **intended configuration**
  - What it is **actually doing** – i.e. **operational state, including the applied configuration.**
- Different solutions to this problem have been evaluated by IETF.
- The agreed IETF solution defines a new “operational” datastore for operational state:
  - NETCONF/RESTCONF additions to support the operational datastore.
  - Replaces the existing ‘broken’ NETCONF GET operation.
  - This also has implications on the structure of YANG models to be optimized for use with NMDA.

# Different YANG model structures

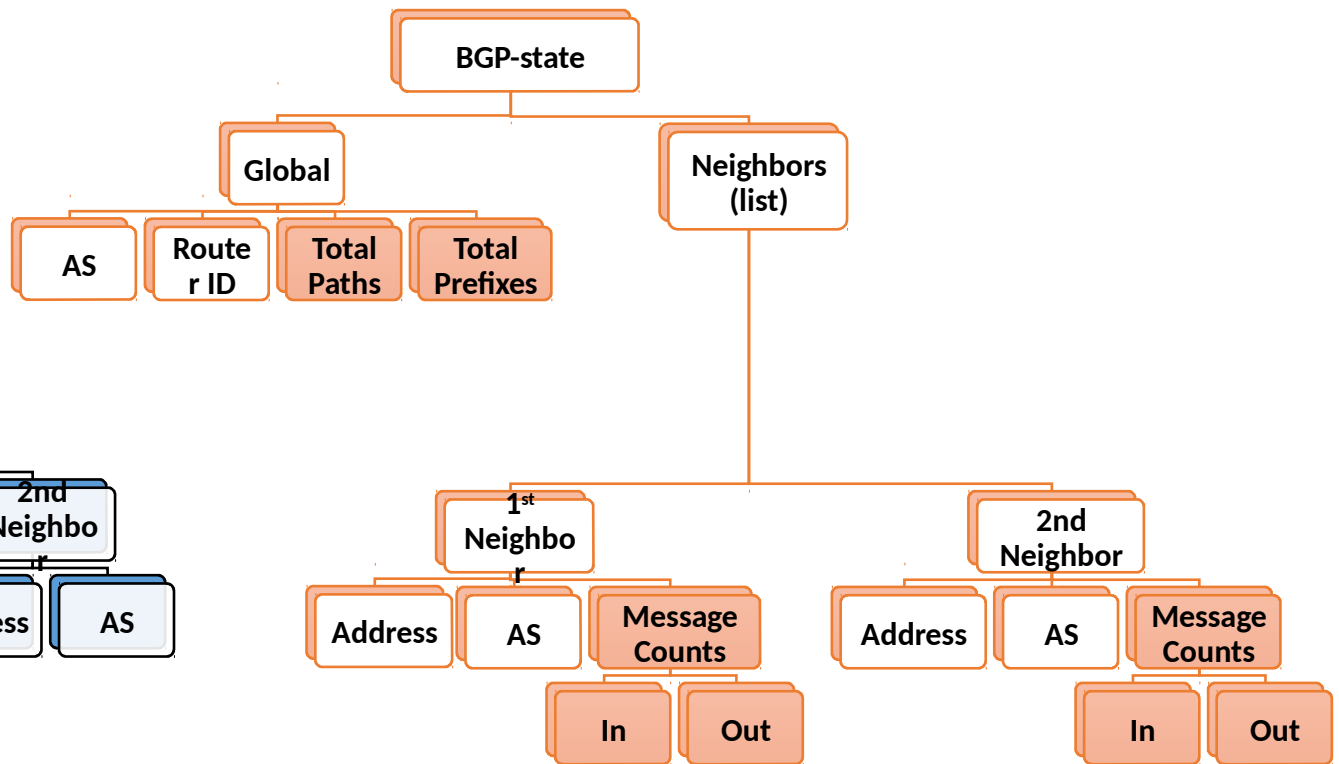
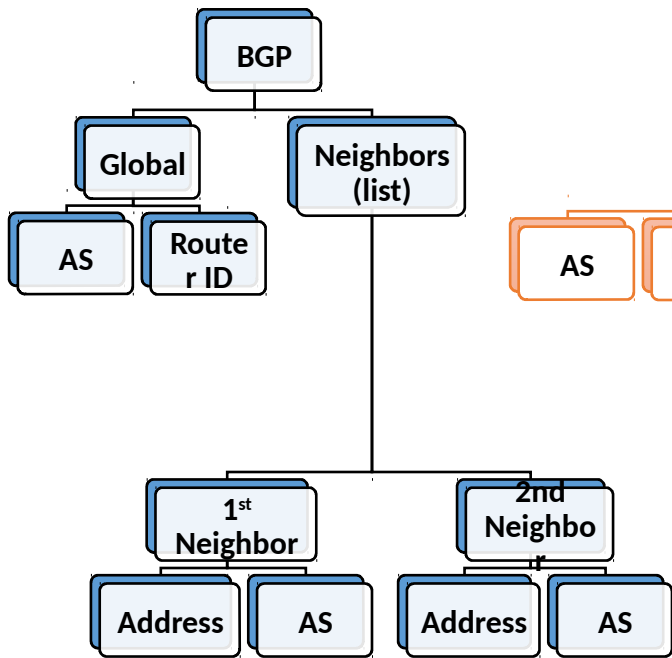
YANG models can be structured in different ways (including):

1. IETF style split 'config' and 'state' top level trees (***deprecated by NMDA***):  
E.g. this is the structure currently used by ietf-interfaces.yang (RFC 7223)
2. Open Config style 'config' and 'state' containers immediately above config true leaves:  
E.g. this is the structure consistently used by all Open Config YANG models (<https://github.com/openconfig/public>)  
The expired BGP YANG model draft also currently has this structure: draft-ietf-idr-bgp-model-02
3. IETF combined config/state tree (***NMDA style - the future of IETF YANG models***):  
Various draft modules are now following this convention:  
I2RS topology model: draft-ietf-i2rs-yang-network-topo-14  
TE topology model: draft-ietf-teas-yang-te-topo-11

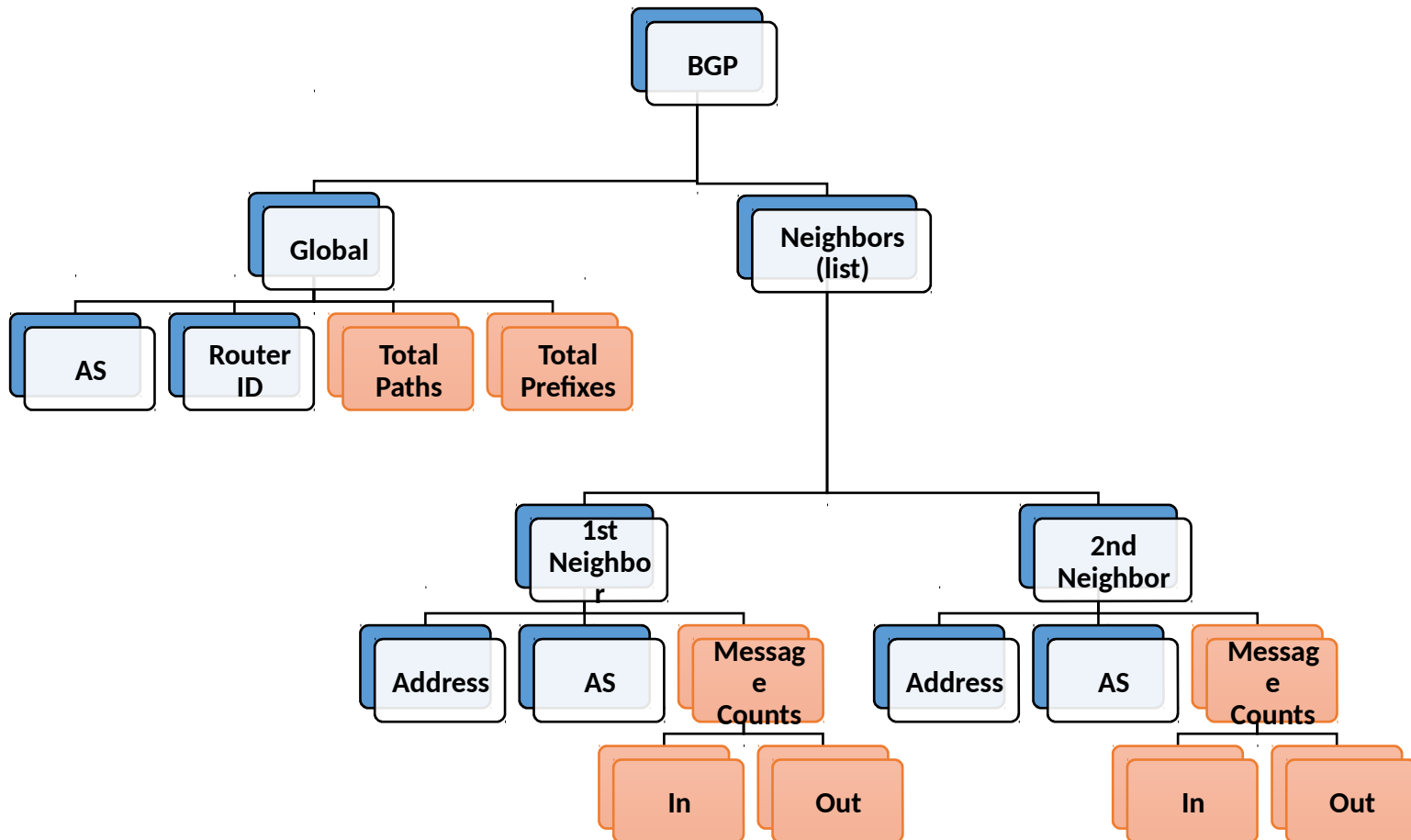
# From the BGP YANG model, consider:

- 4 Global Leaves:
  - AS number (**configurable**)
  - Router Id (**configurable**)
  - Total paths (**state only**)
  - Total prefixes (**state only**)
- 4 Per Neighbor Leaves:
  - Neighbor address (**configurable**)
  - Peer AS (**configurable**)
  - Messages: **Container with 'In' and 'Out' state leaves**

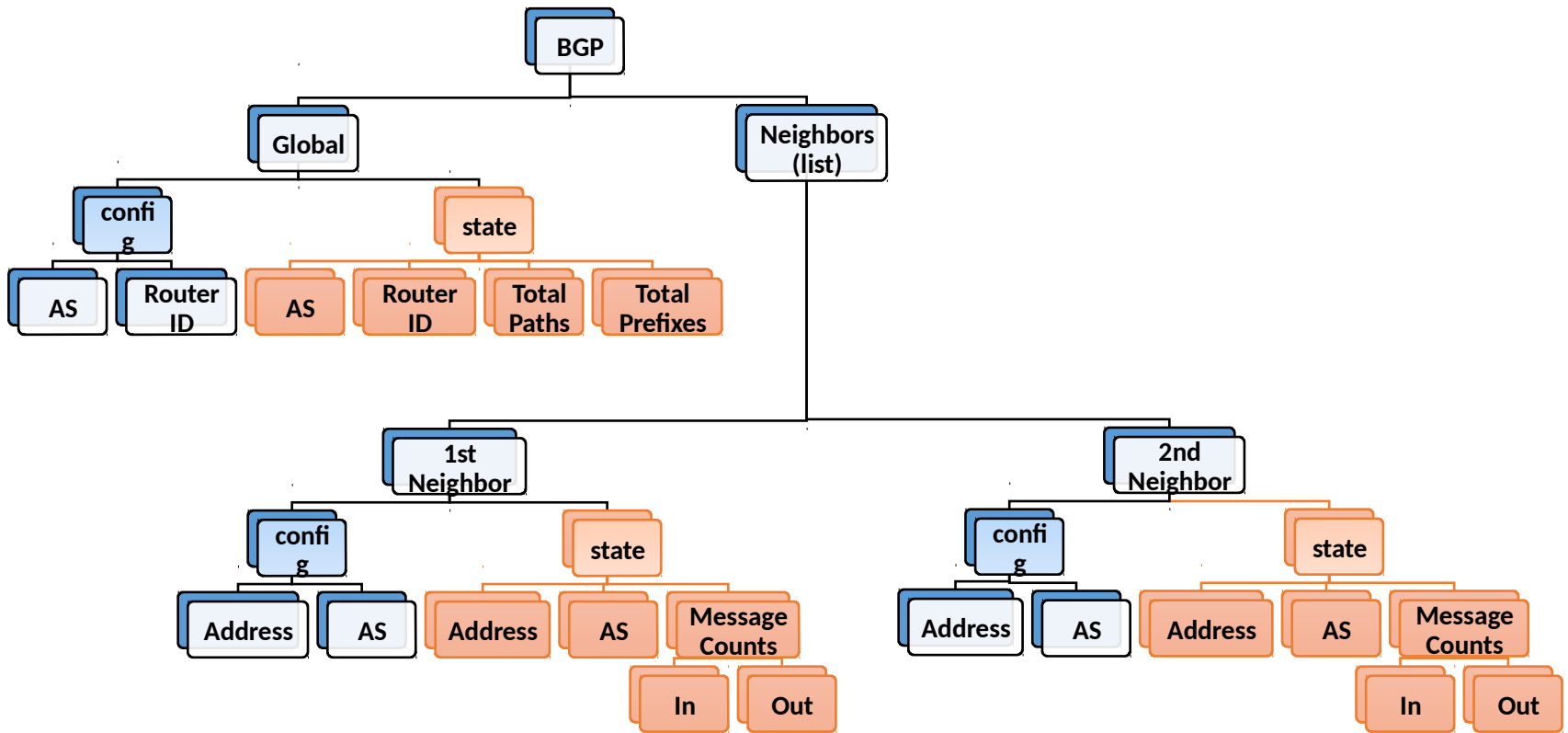
# μBGP model as IETF split config/state module: (*deprecated*)



# μBGP model as IETF combined config/state YANG Module: (NMDA – the future for IETF YANG models)



# μBGP model as OpenConfig style YANG Module: One tree with config/state containers (for reference - not for IETF YANG models)



# Advantages of combined (NMDA) YANG module structure

- The module is simpler and shorter to write, and doesn't require extensive use of groupings.
- It is impossible for the configuration and state trees to become inconsistent in either path or value space.
- The modules are fully consistent with existing YANG semantics and all language constructs.
- The proposed structure takes into consideration support for other IETF work such as the I2RS WG.
- Deduplication of leaves means other model styles could be generated by tooling, if required.



# How to migrate to NMDA style

- All YANG modules produced by IETF SHOULD conform to the NMDA architecture ...
- All YANG modules already published by IETF should be revised to conform to NMDA:
  - All nodes in any <foo>-state trees are copied into the <foo> (config true) tree, creating it if necessary.
  - The existing state tree is marked as deprecated.
  - Update descriptions as required for semantic consistency.
- Please also update WG draft YANG models to NMDA style
  - As above, but <foo>-state tree is also deleted.

# Can we use NMDA style modules on existing NETCONF/RESTCONF?

- Yes, but:
  1. There is no way of reporting the “applied configuration value”.
  2. It cannot report system created configurable objects  
(e.g. an interface that always exists even without configuration)
- For most modules/implementations these limitations should not be problem.
- In a small number of cases, where this limitation is a problem:
  - A temporary “config false” “<foo module>-state” module MAY be constructed and put in the draft appendix, for use until NMDA compliant implementations become available.
  - Expected to be obsoleted over time.

# Summary

- All unpublished IETF YANG modules SHOULD follow the NMDA style.
- Extra generated “<foo>-state” modules may be added into the draft appendix, when there is a genuine requirement to do so.
- Please email NMDA draft authors, or NETMOD WG alias, if you have questions on how to migrate your drafts:
  - [draft-ietf-netmod-revised-datastores@ietf.org](mailto:draft-ietf-netmod-revised-datastores@ietf.org)
  - [netmod@ietf.org](mailto:netmod@ietf.org)
- NMDA work is progressing in both NETCONF and NETMOD WGs.