IETF YANG Versioning Packages Update

draft-ietf-netmod-yang-packages-04

NETMOD WG IETF 121, November 2024

Presenting (on behalf of authors): Rob Wilton

Solution Overview

Solution Overview draft: draft-ietf-netmod-yang-versioning-reqs-09

Proposed full solution is made up of 5 drafts (at different levels of maturity):

1. Updated YANG Module Revision Handling (draft-ietf-netmod-yang-module-versioning-11)
2. Module semantic version number scheme (draft-ietf-netmod-yang-semver-15)
3. **Versioned YANG packages (draft-ietf-netmod-yang-packages-03)**
4. Protocol operations for package version selection (draft-ietf-netmod-yang-ver-selection-00)
5. YANG schema comparison tooling (draft-ietf-netmod-yang-schema-comparison-02)

Current Status

YANG Packages work was put on hold:

- WG LCs of Module Versioning & Semver
- Ratholed in complexity of API vs implementation packages & optionality

We started work again (approx 4 weeks ago):

- Also, discussions and reset on goals/approach
- -04 posted: Updated/fixed YANG errors, mounted schema prototype
- I asked the other contributors a few questions

Q. What does a package define?

- Is it just a hierarchical equivalent to YANG Library?
- Should we define:
  - **API packages** (e.g., a collection of related IETF modules), vs
  - **Implementation packages** (i.e., what a device implements)

Contributors' consensus:

- Keep it simple
- Packages just define what is there (like YANG library)

Q. Should packages allow for including optional modules/packages?

- Flexible API packages might want to include optional functionality. E.g., an IETF routing package making some protocols optional, or mounted schema

Contributors' consensus:

- Keep it simple
- Packages just define what is there
- No optionality (beyond YANG's standard feature statements/deviations)

Q: What should packages indicate about deviations?

E.g.,

a. Nothing
b. A flag to indicate that the package contains deviations
c. Further indications as to which module files contain deviations (this is more complex).

Contributors' consensus:

- Not clear yet
- Probably a, possibly b, not c.

Q: Should packages support schema mount?

E.g.,

a. No
b. Yes (indicating minimum set of packages that would be found)

Contributors' consensus:

- Not clear yet
- Probably b, needed for some IETF modules, but … more complex

We are also discussing whether to allow including specific mount instances (e.g., instance data path with optional keys)

Q: Should packages include checksums?

E.g.,

a. No, not needed
b. No, we don’t want to specify a canonical format for YANG modules
c. Yes, this is useful, also need to specify a canonical format for YANG modules
d. Yes, this is useful, just use the SHA hash over the raw text contents

Contributors’ consensus:

- Leaning towards “Not now”. I.e., keep simple and defer to a future version

TL;DR Summary

Current approach is to keep YANG Packages as simple as possible:

- Packages are an alternative hierarchical version of YANG library
- Not considering ‘API packages’ now. Hence, probably smaller packages
- Cannot take modules/packages out of a package
- Schema mount support is probably needed
- Checksums deferred for now

Does the WG agree with this direction, any comments?