

YANG Semantic Versioning

[draft-ietf-netmod-yang-semver](#)

NETMOD WG

March 2021

**Presenting on behalf of the authors:
Joe Clarke**

Current Status

- Added text to address version gaps (though more text and clarity coming post-IETF 110)
- Addressed GitHub issue #45 by adding text that IETF modules **MUST** have YANG semver revision-labels
- Added guidance text for IANA-maintained modules

Addressing Version Gaps

- Allowing gaps could break “import revision-or-derived”
 - If a version lineage goes 1.0, 1.1, 1.3, how would an import of 1.2 work?
 - The revision-label 1.2 is not tied to any entry in the history
 - Likewise, can you import by a revision date that does not exist?
- New text added to state that version gaps are *allowed* but in doing so, this can impact importing by revision-label
- More text to clarify this and address issue #78 (allowing one to change MAJOR version for backwards-compatible changes) coming after IETF 110 (see <https://github.com/netmod-wg/yang-ver-dt/pull/79>)

YANG Semver mandatory for IETF modules

GitHub Issue #45: Should all newly published IETF YANG modules include a yang-semver revision-label?

Authors' proposal: yes

- helps readers understand “at a glance” if two revisions of a module are backwards compatible
- New text:

All publish[ed] IETF modules MUST use YANG semantic versions for their revision-labels. For IETF YANG modules that have already been published, revision labels MUST be retrospectively applied to all existing revisions when the next new revision is created, starting at version "1.0.0" for the initial published revision, and then incrementing according to the YANG Semver version rules specified in [Section 3.3](#) .

Guidelines for IANA-Maintained Modules

- Text has been added to Section 9.2 to guide IANA on how to apply YANG semver revision labels to existing modules upon next update
- Highlights:
 - IANA maintained modules MUST also include a YANG semver revision-label
 - Revision-labels MUST be retroactively applied to existing modules when the next revision is published
 - IANA modules are not expected to ever use the `_compatible` or `_non_compatible` modifiers

Next Steps

- We received feedback from a semver contributor in GitHub issue #78 on a number of points; we need to audit those to see what text changes are warranted
- One point to call out: the semver.org (Semver 2.0.0) spec changes without changing the version number (ironically)
 - In essence, this means there have been changes (some normative) between semver 2.0.0 five years ago and 2.0.0 today
 - Change xref to point to static GitHub semver revision
 - Add text to document to indicate why the reference to the specific GitHub version