

NETMOD Versioning Solution Update

NETMOD WG

Mar 2022

Presenting on behalf of the weekly versioning call attendees:
Joe Clarke

IETF 113

Agenda

This presentation:

Solution Overview – Joe Clarke

Following presentations on individual drafts:

YANG Packages – Rob Wilton

See backup slides for solution overview & summary of Module Versioning and YANG Semver drafts.

Recap - YANG Versioning Solution Overview

Complete solution consists of five drafts:

1. Updated YANG Module Revision Handling:
Notify nbc changes between module revisions, allows branched revision history, revision-labels
2. Module semantic version number scheme:
YANG semver for module revision-labels and package versioning
3. Versioned YANG packages:
Versioning at the schema level rather than individual modules
4. Protocol operations for package version selection:
Devices can support multiple schema versions, clients can select for session
5. YANG schema comparison tooling:
Tooling to algorithmically compare module or schema revisions

Latest working drafts can all be found here: <https://github.com/netmod-wg/yang-ver-dt>

General update on weekly Versioning calls

- Authors + interested parties continue to meet on a weekly call to progress this work
 - Meetings are open to all
 - Regular participation from people across 5+ different companies (mostly equip. vendors, participation from more operators/users would be beneficial)
 - Key issues are brought back to WG mailing list
 - Weekly meeting is currently on Tues @2pm UK time / 9am Eastern.
Thanks to the authors and contributors for their regular attendance

General update on weekly Versioning calls

Focus of weekly meetings since IETF112:

Architectural discussions and closing issues against the Packages draft

- Brainstorming on effective structure, types of packages, and use cases for metadata
- owners assigned to issues to produce updated text
- many still to work through

Module Versioning Updates Since IETF112

[draft-ietf-netmod-yang-module-versioning-05](#)

- WG LC just closed
- Issues raised by Jürgen, Italo and Andy
 - Will be analyzed/discussed on the weekly calls, with responses back on the main NETMOD mailing list for each item
- Issues raised include (amongst others):
 - contradictions in filename guidelines
 - clarify how to understand the history of an individual module revision, vs the entire branched tree of a module
 - clarify how to calculate if revision Y of a module is derived from another specific revision X
 - consider tags for individual changed elements
 - issue of YANG revision dates needing to be unique in branched modules
 - general complexity of the proposed solution

YANG Semver Updates Since IETF112

[draft-ietf-netmod-yang-semver-06](#)

Completed WG LC

- Issues raised by Jürgen and Italo
- Some of Jürgen's smaller suggestions added to git
- Issues raised for:
 - Clarify that the YANG Semver scheme is not SemVer
 - Clarify where SemVer is supported and what is unique to YANG Semver
 - Difference between "bug fix" vs. "editorial update"
 - When to bump the YANG Semver revision lab for a draft?

YANG Semver Updates Since IETF112

[draft-ietf-netmod-yang-semver-06](#)

Italo asked what to do about file naming in <CODE BEGINS>/<sourcecode>

Example:

```
<CODE BEGINS> file "ietf-yang-revisions@2021-11-04.yang"
<sourcecode name="ietf-yang-semver@2021-11-04.yang"
```

- Use revision date as it is now?
- Support revision-labels (e.g., `ietf-yang-revisions#1.0.0`)
- **SUGGESTION:** Keep it as it is now and have tooling extract both a module file with revision date and revision-label; or remove revision altogether and have tooling extract both a module file with a revision date and a revision-label

Alternative to YANG Semver

- Jürgen raised a thread proposing an alternative to the versioning work on which Andy chimed in
- Provide annotations at the node level to signal NBC changes

```
leaf foo {  
    nbc-change "2022-03-01" {  
        description "changed type from int32 to string";  
    }  
    // ...  
}
```

- Approach has been considered previously and is part of "tooling" draft
- Needs further discussion - will be a topic on weekly versioning calls after IETF 113

Next Steps

- Issues tracked in github, discussed on WG mail list
<https://github.com/netmod-wg/yang-ver-dt>
- Phase work on the documents:
 - Address WG LC comments on module versioning and YANG Semver
 - Focus on Packages draft:
 - many open issues
 - various people contributing text updates
 - Version Selection and Schema Comparison to follow

BACKUP SLIDES

Module Revision Handling + YANG Semver Drafts

Summary of main requirements & recommendations

[draft-ietf-netmod-yang-module-versioning - Updated YANG Module Revision Handling](#)

[draft-ietf-netmod-yang-semver - YANG Semantic Versioning](#)

A) Mark a revision as non-backwards-compatible

```
module example-module {  
  
    namespace "urn:example:module";  
    prefix "prefix-name";  
  
    import ietf-yang-revisions { prefix "rev"; }  
    import ietf-yang-semver { prefix "ysver"; }  
  
    rev:revision-label-scheme "ysver:yang-semver";  
  
    revision 2019-06-01 {  
        rev:revision-label 3.1.0;  
        description "Add new functionality.";  
    }  
  
    revision 2019-03-01 {  
        rev:revision-label 3.0.0;  
        rev:non-backwards-compatible;  
        description  
            "Add new functionality. Remove some deprecated nodes.";  
    }  
  
    revision 2019-02-01 {  
        rev:revision-label 2.0.0;  
        rev:non-backwards-compatible;  
        description "Apply bugfix to pattern statement";  
    }  
}
```

Module Revision Handling + YANG Semver Drafts

Summary of main requirements & recommendations

B) Import by revision-or-derived

Specify a **minimum** revision for import, but any descendant is also conformant.

Use the latest descendant available.

```
import example-module {  
    rev:revision-or-derived 3.0.0  
}
```

Module Revision Handling + YANG Semver Drafts

Summary of main requirements & recommendations

C) YANG Semver revision-label

X.Y.Z = Major.Minor.Editorial

Major -> change for NBC changes

Minor -> additional leafs, etc

Editorial -> fix a description

Example YANG module with branched revision history.

Module revision date	Revision label
2019-01-01	<- 1.0.0
2019-02-01	<- 2.0.0
2019-03-01	<- 3.0.0
2019-04-01	<- 2.1.0
2019-05-01	<- 2.2.0
2019-06-01	<- 3.1.0

```
module example-module {  
  
    namespace "urn:example:module";  
    prefix "prefix-name";  
  
    import ietf-yang-revisions { prefix "rev"; }  
    import ietf-yang-semver { prefix "ysver"; }  
  
    rev:revision-label-scheme "ysver:yang-semver";  
  
    revision 2019-06-01 {  
        rev:revision-label 3.1.0;  
        description "Add new functionality.";  
    }  
  
    revision 2019-03-01 {  
        rev:revision-label 3.0.0;  
        rev:non-backwards-compatible;  
        description  
            "Add new functionality. Remove some deprecated nodes.";  
    }  
}
```

Optional compatibility extension:

rev:revision-label 3.0.2 non_compatible

Module Revision Handling + YANG Semver Drafts

Summary of main requirements & recommendations

D) Filename recommendation

```
my-module#3.2.0.yang
```