Network Management

YANG as a motivation for open source?

Glenn Parsons, IEEE 802.1 WG chair
December 2017
Network Management protocol soup

- **Network Management** is the process of administering and managing the networks of one or many organizations:
  - fault analysis
  - performance management
  - provisioning of networks
  - maintaining the quality of service

- Several SDOs have defined an architecture:
  - ISO – FCAPS
  - TMF – FAB

- ...and protocols:
What is network management for?

- **YANG**
  - YANG is a data modeling language used to model configuration data, state data, Remote Procedure Calls, and notifications for network management protocols (e.g., NETCONF, RESTCONF, etc.)

- **NETCONF**
  - The Network Configuration Protocol (NETCONF) provides mechanisms to install, manipulate, and delete the configuration of network devices.
  - It is an example network configuration protocol
Motivation in development was to satisfy Network Operator requirements (RFC 3535)
- Ease of use, clear distinction between configuration data and operation state & stats, ability to fetch separately configuration/operational data from the device, etc.

YANG is
- Human readable and easy to learn
- Hierarchical configuration
- Reusable types and groupings
- Extensibility through augmentation
- Formal constraints for configuration validation
- Data modularity through modules and sub-modules
Approved YANG model PARs in IEEE 802

- P802.1Xck – Port authentication
- P802.1Qcp – VLAN Bridges
- P802.1Qcw – TSN
- P802.1Qcx – CFM
- P802.1ABcu – LLDP
- P802.3.2 – Ethernet
- …
GitHub – YANG repository

- **YANG Model directory** for IETF, IEEE, MEF, BBF, ...

- **IEEE License:**
  - All files contained within this sub-directory are considered to be intended as IEEE Contributions.
  - All issues entered into the trouble ticket system for this directory are considered to be intended as IEEE Contributions.
  - All pull requests submitted for this directory are considered to be intended as IEEE Contributions.
  - All contributions to IEEE standards development (whether for an individual or entity standard) shall meet the requirements outlined in the [IEEE-SA Copyright Policy](#)
  - Copyright release for YANG modules: Users may freely reproduce the YANG modules contained under /experimental/ieee/ so that they can be used for their intended purpose.
A **YANG model catalog** and registry that allows users to find models relevant to their use cases from the large and growing number of YANG modules being published.

- A **YANG Validator**, a web frontend that allows for validation of YANG modules.
- A **YANG Search**, a web frontend that allows for searches over the content of the module catalog.
- A **YANG impact analysis** tool.
- View a module's **metadata details**.
- An interactive **YANG exploration** tool that includes a YANG browser, RPC builder, and a YDK script generator to experiment with YANG modules
- A **YANG Regex Validator**, a YANG regular expression validator to experiment with W3C YANG "pattern" statements
Source Code in IEEE standards

- Tooling from the Open Source community has provided a significant improvement in code development

- Current process has been enhanced to support this:
  - Development of code in GitHub with IEEE license indication – all code contributions are considered contributions to the standard
  - Publication of code in IEEE standard – pasted inline and attached to PDF as text code files
  - Publication of code on website or in GitHub
  - Copyright release to freely reproduce the YANG modules so that they can be used for their intended purpose.
What can full open source offer?

- Faster and continuous (aka agile) updating of YANG models
  - Errors in code, Additional enumerations in lists, Augmentations
- Flexibility on release cadence
- Much larger contributor / developer pool
- Early adoption by developers and network operators
- Added complexity for contributors?
  - Open Source Contribution License Agreement vs Standards Contribution Copyright Policy & Patent Policy
- Uncertainty on how to control direction of the open source specification
  - Consensus by balloting vs benevolent dictator
- A different revenue model