



Universidad Tecnológica
del Norte de Guanajuato
Organismo Público Descentralizado del Gobierno del Estado

"Educación y progreso para la vida"

Tecnologías de la Información y Comunicación

Programa educativo: TSU en Infraestructura de Redes Digitales

Área académica: Programación de Redes

Asignatura:

Unidad 3

Grupo: GIR0441

2.3 Lab - Explore YANG models using the pyang

Alumna Gómez Luna Cinthia Valeria:

Docente: Gabriel Barrón Rodríguez

Dolores Hidalgo, C.I.N., Gto., Jueves 24 de Noviembre de 2022

Introducción

En el presente documento realizado por la alumna Cinthia Valeria Gomez Luna se podrá observar unas capturas de pantalla con lo realizado hablaremos principalmente de la instalación del pyang

¿Qué es el modelo YANG? Yang es un lenguaje de modelado de datos extensible basado en estándares que se utiliza para modelar la configuración y los datos de estado operativo, las llamadas de procedimiento remoto (RPC) y las notificaciones de eventos de servidor de dispositivos de red. El grupo de trabajo NETMOD de la GTI-I diseñó originalmente YANG para modelar los datos de administración de red y proporcionar un estándar para la capa de contenido del modelo del protocolo de configuración de red (NETCONF). Sin embargo, YANG es independiente del protocolo, y los modelos de datos YANG se pueden utilizar independientemente del transporte o del protocolo RPC y se pueden convertir en cualquier formato de codificación admitido por el protocolo de configuración de red.

¿En qué casos puedo utilizarlo? Se puede utilizar para crear solicitudes de configuración de dispositivos o solicitudes de datos operativo y es muy fácil de aprender el manejo de la configuración

¿El tipo de formatos que maneja? Se dice que pueden utilizar el formato YumaPro se dice que se Puede encontrar la lista de extensiones de archivo asociada

Se deberá Instalar el módulo pyang de Python

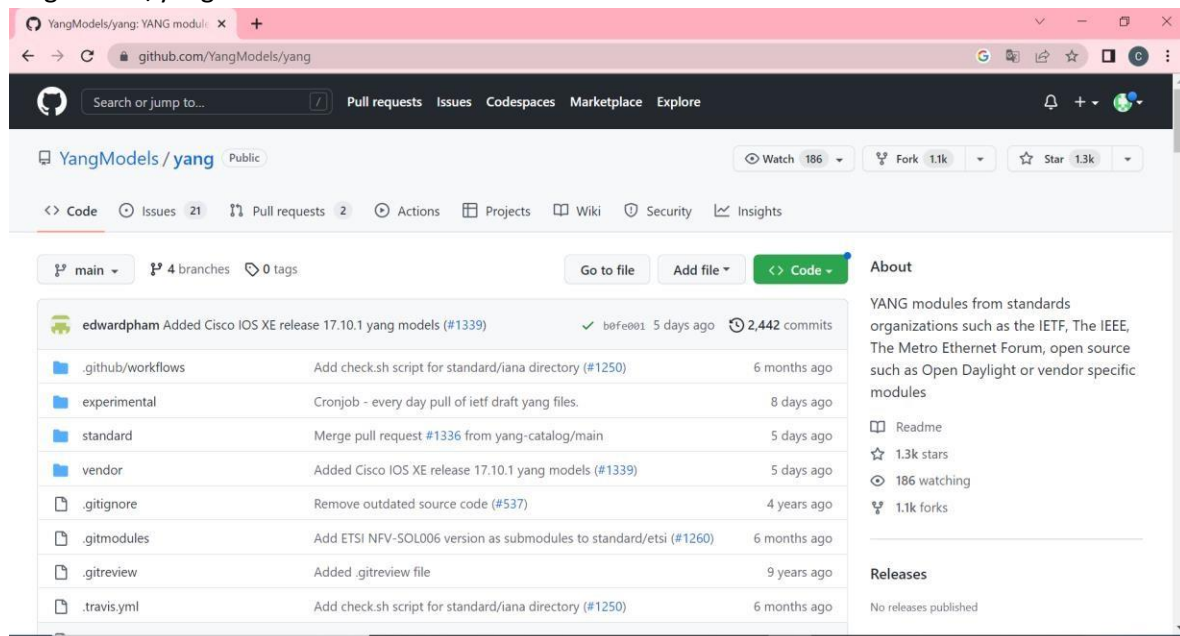
```
Símbolo del sistema
Microsoft Windows [Versión 10.0.19044.2251]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\valeria>pip install --no-binary pyang pyang
DEPRECATION: --no-binary currently disables reading from the cache of locally built wheels. In the future --no-binary will not influence the wheel cache. pip 23.1 will enforce this behaviour change. A possible replacement is to use the --no-cache-dir option. You can use the flag --use-feature=no-binary-enable-wheel-cache to test the upcoming behaviour. Discussion can be found at https://github.com/pypa/pip/issues/11453
Requirement already satisfied: pyang in c:\users\info\appdata\local\programs\python\python310\lib\site-packages (2.5.3)
Requirement already satisfied: lxml in c:\users\info\appdata\local\programs\python\python310\lib\site-packages (from pyang) (4.9.1)
```

Una vez Instalado se deberá verificar la versión con el comando `pyang -v` la versión que se instaló es la siguiente

```
pyang 2.5.3
```

Una Vez Instalada se deberán ir a Explorar los modelos YANG en el repositorio GitHub de YangModels/yang.



Script para visualizar modelo YANG

725 lines (602 sloc) | 23.7 KB

```
1  module ietf-interfaces {
2
3      namespace "urn:ietf:params:xml:ns:yang:ietf-interfaces";
4      prefix if;
5
6      import ietf-yang-types {
7          prefix yang;
8      }
9
10     organization
11         "IETF NETMOD (NETCONF Data Modeling Language) Working Group";
12
13     contact
14         "WG Web:  <http://tools.ietf.org/wg/netmod/>
15         WG List:  <mailto:netmod@ietf.org>
16
17         WG Chair: Thomas Nadeau
18                 <mailto:tnadeau@lucidvision.com>
19
20         WG Chair: Juergen Schoenwaelder
```

Una vez Realizado todo esto nuestro objetivo será utilizar la herramienta de línea de comandos pyang para transformar los modelos YANG todo esto se debe de hacer en el cmd

```
C:\Users\valeria\Documents>pyang -f tree ietf-interfaces.yang.txt
```

```
module: ietf-interfaces
  +--rw interfaces
  |   +--rw interface* [name]
  |   |   +--rw name                string
  |   |   +--rw description?       string
  |   |   +--rw type                identityref
  |   |   +--rw enabled?           boolean
  |   |   +--rw link-up-down-trap-enable? enumeration {if-mib}?
  |   +--ro interfaces-state
  |   |   +--ro interface* [name]
  |   |   |   +--ro name                string
  |   |   |   +--ro type                identityref
  |   |   |   +--ro admin-status       enumeration {if-mib}?
  |   |   |   +--ro oper-status       enumeration
  |   |   |   +--ro last-change?      yang:date-and-time
  |   |   |   +--ro if-index          int32 {if-mib}?
  |   |   |   +--ro phys-address?     yang:phys-address
  |   |   |   +--ro higher-layer-if*  interface-state-ref
  |   |   |   +--ro lower-layer-if*  interface-state-ref
  |   |   |   +--ro speed?            yang:gauge64
  |   |   |   +--ro statistics
  |   |   |   |   +--ro discontinuity-time yang:date-and-time
  |   |   |   |   +--ro in-octets?        yang:counter64
  |   |   |   |   +--ro in-unicast-pkts?  yang:counter64
  |   |   |   |   +--ro in-broadcast-pkts? yang:counter64
  |   |   |   |   +--ro in-multicast-pkts? yang:counter64
  |   |   |   |   +--ro in-discards?     yang:counter32
  |   |   |   |   +--ro in-errors?       yang:counter32
  |   |   |   |   +--ro in-unknown-protos? yang:counter32
  |   |   |   |   +--ro out-octets?      yang:counter64
  |   |   |   |   +--ro out-unicast-pkts? yang:counter64
  |   |   |   |   +--ro out-broadcast-pkts? yang:counter64
  |   |   |   |   +--ro out-multicast-pkts? yang:counter64
  |   |   |   |   +--ro out-discards?    yang:counter32
  |   |   |   |   +--ro out-errors?      yang:counter32
```

Conclusión

Al momento de realizar la practica tuve algunos problemas al instalar el pyang ya que me marcaba demasiados errores y para lograr llegar a la solución busque más información la cual me dice que los modelos yan nos sirven para poder solicitudes de configuración lo cual se me hizo muy interesante me gusto realizar la practica a pesar de que tuve varios conflictos se puede concluir que llegamos a concluir lo que se nos pedía adjunte evidencia de como fui realizando el proceso.