

- 1.) Bauen Sie über myExperTeach die Verbindung zur VM auf.
- 2.) Vorbereitende Maßnahme für die praktischen Übungen (falls nicht bereits geschehen oder im Vorfeld vom Trainer erledigt): Starten Sie eine Linux Shell (Desktop Icon MATE-Terminal). Für die Übungen werden einige Dateien benötigt, die aus einem Repository bei GitHub geclonet werden:

```
git clone https://github.com/Wotan242/NETY.git
```

Die Dateien sollten sich nun in dem Ordner /NETY befinden.

- 3.) Starten sie eine Shell. Wechseln Sie in das Verzeichnis YANG. Dort finden sie zwei YANG-Modelle A1.yang und B1.yang. Öffnen sie die beiden Dateien mit dem Sublime-Editor:

```
et@ubuntu:~/YANG$ subl A1.yang          et@ubuntu:~/YANG$ subl B1.yang
```

Untersuchen Sie die Statements in beiden Modellen. Welche Gestalt hat der Schema Tree?

- 4.) Analysieren sie beide YANG-Modelle mit pyang (Die Option --tree-print-groupings ist erforderlich, weil pyang Groupings per Default nicht anzeigt):

```
et@ubuntu:~/YANG$ pyang -f tree --tree-print-groupings A1.yang
et@ubuntu:~/YANG$ pyang -f tree --tree-print-groupings B1.yang
```

- 5.) Importieren Sie Modul B1 in das Modul A1 mit dem Prefix B1:

```
import B1 {prefix B1;}
```

Welches Resultat bekommt man mit pyang? Ist dieser Import sinnvoll?

```
et@ubuntu:~/YANG$ pyang -f tree --tree-print-groupings A1.yang B1.yang
```

- 6.) Binden Sie in Modul A1 das aus Modul B1 importierte Grouping Group_B1 mit

```
uses B1:Group_B1;
```

in den Container Cont_A1 ein. Was ergibt die Auswertung mit pyang? Warum wird im uses Statement ein Qualifier B1: benötigt und warum taucht dieser Qualifier für die aus dem Grouping stammenden Statements im Output von pyang nicht mehr auf?

- 7.) Augmentieren Sie in Modul A1 den importierten Container Cont_B1 mit einem zusätzlichen Leaf Node:

```
augment "/B1:Cont_B1" {
    leaf Leaf_A2 {
        type empty;
    }
}
```

Beachten Sie in der Pfadangabe des augment Statements die korrekte Qualifizierung der importierten Node Names! Wie wird das in pyang dargestellt?

- 8.) Augmentieren sie in Modul A1 den in Cont_A1 enthaltenen Container Cont_B2 (aus dem Grouping) mit einem zusätzlichen Leaf Node:

```
augment "/Cont_A1/Cont_B2" {
    leaf Leaf_A3 {
        type empty;
    }
}
```

Warum wird in der Pfadangabe des augment-Statements für die aus dem importierten Grouping stammenden Schema Nodes kein Qualifier B1: benötigt?

- 9.) Sie finden die finalen Versionen der beiden YANG-Module auf dem Desktop der VM unter YANG_A1_final.yang und YANG_B1_final.yang.