**TOSCA To CloudShell transformation**

**SHELL FILE**

**metadata**:  
 **name**: NXOS Shell // Details.Name  
 **author**: Meni Besso // Details. Author  
 **version**: 1.0.0 // Details.Version  
  
**description**: > // Details.Description  
 This is an example of a shell implementation  
 1) It declares following the 'NetworkDevice' standard  
 2) It associates the NXOS shell to the standard 'Switch' family  
 3) It expands the standard capabilities with additional property and additional function  
  
**imports**:  
 - **cloudshell\_standard**: cloudshell\_networking\_standard.yaml  
  
**node\_types**:

// All leaf node types will translate to Root Models with the following definitions:  
  
 **cloudshell.switch.NXOS**: suffix should go to Components.AllModels.Details.Name

**derived\_from**: cloudshell.nodes.Switch

**properties**: // SEE BELOW #1  
 **SNMP\_version**:  
 **type**: string  
 **default**: ''  
 **tags**: [configuration, setting, search\_filter, abstract, insight, admin\_only, read\_only]  
 **description**: Possible values - v1, v2c, v3  
 **constraints**:  
 - **valid\_values**: [ v1, v2c, v3]

**interfaces**:   
 **notify\_device\_owner**: {}

**artifacts**:  
 **icon**:  
 **file**: NXOS.png   
 **type**: tosca.artifacts.File

#1

Property Translation to CloudShell attributes

Attributes Bank For Models

|  |  |
| --- | --- |
| JSON  "Attributes":[  {  "Name":"My Attribute",  "Description":"",  "DefaultValue":"a",  "AttributeType":"String",  "IsReadOnly":false,  "IsCommand":false,  "Tags":["Setting", "Configuration"],  "LookupValues":[]  “DefaultValueWasSet”:”false”  } | TOSCA  **properties**: // SEE BELOW #1  **SNMP\_version**:  **type**: string  **default**: ''  **tags**: [configuration, setting, search\_filter, abstract, insight, admin\_only, read\_only, user\_input]  **description**: Possible values - v1, v2c, v3  **constraints**:  - **valid\_values**: [ v1, v2c, v3] |
| My Attribute | **SNMP\_version** |
| Description | Possible values - v1, v2c, v3 |
| DefaultValue | Empty |
| AttributeType | * T:string -> C:String * T:integer -> C: Numeric * T:float -> C: Numeric * T:boolean -> C:Boolean * T:timestamp -> C:String (converted) |
| IsReadOnly | If tags contain read\_only set to True otherwise set to False |
| IsCommand | Always false |
| LookupValues | No support for lookup data type. |
| Tags | * T:configuration -> C:Configuration * T:setting -> C:Setting * T: search\_filter -> C:Display In Search Filters * T: abstract -> C:Available For Abstract Resource * T: insight -> C:Display In Insight * T: admin\_only -> C:Admin Only * T: read\_only -> (see above: IsReadOnly = True) * T: user\_input -> C: |
| Default value was set | False |

Attached Attributes Per Model

|  |  |
| --- | --- |
| JSON  "AttachedAttributes":[  {  "AttributeName":"My Attribute",  "IsLocal":true,  "IsOverridable":false,  "IsRequired":false,  "AllowedValues":[],  "Value":"b"  “ValueWasSet”:”true”  } ] | TOSCA  **properties**: // SEE BELOW #1  **SNMP\_version**:  **type**: string  **default**: ''  **tags**: [configuration, setting, search\_filter, abstract, insight, admin\_only, read\_only, user\_input]  **description**: Possible values - v1, v2c, v3  **constraints**:  - **valid\_values**: [ v1, v2c, v3] |
| My Attribute | **SNMP\_version** |
| IsLocal | True |
| IsOverridable | False |
| IsRequired (UserInput) | Only for services (will be part of the tags) |
| AllowedValues | **If tosca contains constraints + valid values – this collection includes the valid values.**  **constraints**:  - **valid\_values**: [ v1, v2c, v3] |
| Value | **default**: '' |
| Value was set | **If there is default -> True**  **Otherwise -> False** |