

Planning App

```
checkDeviceMounted (DeviceName)
{return: connection-state}
```

Start

```
newNewDeviceInstance (DeviceName,
DeviceIPAddress,
DeviceType,
Vendor ) {return: status}
```

Step 1

Return status

End

Mounting Orchestrator

Public Functions:

```
newNetconfServer (DeviceName, Step 2
DeviceIPAddress,
DeviceType,
Vendor, NetconfPort) {return status}
```

native

Mediator

Private Functions:

```
newNetconfServerOnMediator
(DeviceName,DeviceIPAddress,DeviceType,
Vendor ) {return:MediatorVmIP, NetconfPort, status}
```

```
newNetconfServer (DeviceName,DeviceIPAddress[or
MediatorVmIP],NetconfPort ) {return status}
```

Device
TypeListRequir
eMeditor-List
Native Netconf
Port 831 TCP

Step 3

mediator

Step 6

Mediator Orchestrator

Static Config:

List-of-Mediator Host IPs, Vendor

Public Functions:

```
newMediatorInstance (DeviceName,
DeviceIPAddress,
DeviceType
Vendor ) {return:MediatorVmIP, NetconfPort, status}
```

DeviceTy
pe to
Mediator
-List

Private Functions:

```
chooseMediatorVmIP(DeviceType,
Vendor ) {return MediatorVmIP, status }
```

Return Mediator
VM with lowest
Resource
Utilization

Step 4

Step 5

```
createMediatorInstance(MediatorVmIPm,
DeviceIPAddress,DeviceType) {return NetconfPort  
status }
```

SDN Controller

Mediator VM1

Mediator Static Config

1. Mediator VM IP
2. Netconf Start Port
3. Netconf End Port
4. Trab Start Port
5. Trab End Port
6. Device Access Credentials

Mediator
Instance

Mediator Instance Manager

Public Functions:

```
newMediatorInstance(DeviceName,DCNIPAddress[,DeviceType]){  
return NetconfPort, status}  
deleteMediatorInstance(DeviceIPAddress){return NetconfPort,  
DeviceName,DeviceIPAddress,DeviceType, status}  
ListAllMediatorInstance(viod){return DeviceName,DeviceIPAddress,  
DeviceType, NetconfPort, status}
```

Private Functions:

```
chooseNetconfPort(void)  
{return NetconfPort, status }  
createMediatorInstance(DeviceName,DeviceIPAddress,DeviceType,  
NetconfPort ) {return status }
```

Next free NetConf
Port (also reuse
PortsNetconf
became free)

Optinal: E///
would benefit

Questions:

1. Where do we get the netconfPort in case of native?
Answer: Is standard SSH 831

Open Points:

1. Exception Handling (e.g for E/// TN) in case of NAT, need to be handled
2. Backup and restore. When the Mediato VM crashes, the existing netconfPort Assignment have to be correct restored.
3. It must be prevented that one Device ist configured several times as Mediator Instance on same and on different VMs
4. Structure NetconfPort Ranges, each Mediator VM gets dedicated ranges. What happens. Happens when a new VM must be created for Performance Reasons.