

# ETSI NFV #19 SpecFest Denver 2017

## VNF Scaling with Nokia VNFM

Nokia CloudBand Application Manager (CBAM)  
Hunor Demeter CBAM, Product Owner  
[hunor.demeter@nokia.com](mailto:hunor.demeter@nokia.com)

# Agenda

1

ETSI NFV

Nokia VNF Manager

2

Nokia VNFM API

Open API documentation

3

VNF lifecycle management

Hands on exercise with VNF scaling

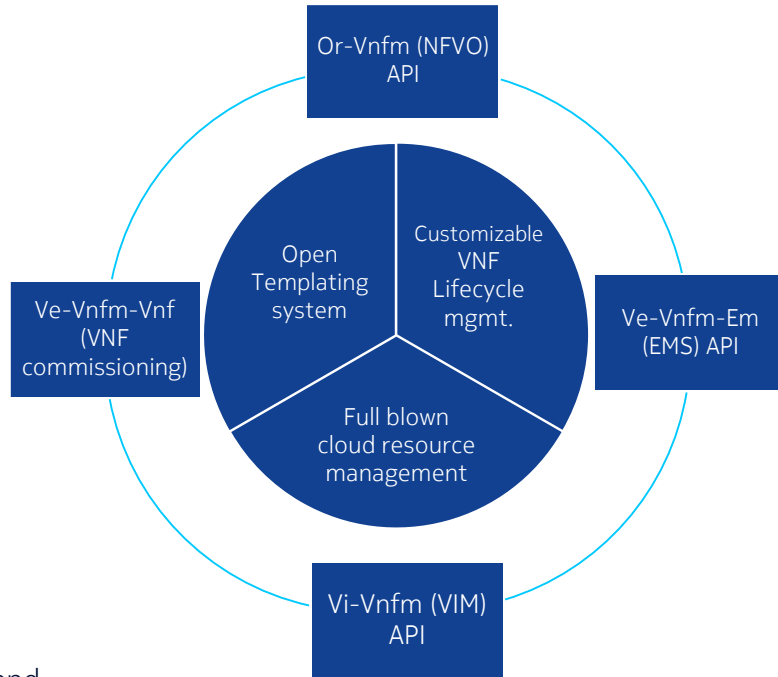
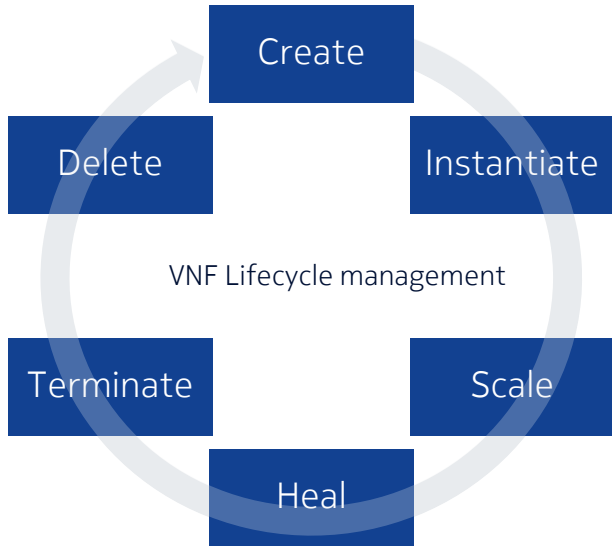
4

Conclusions

# Nokia VNF Manager

# Nokia Cloudband Application Manager

## Inside and outside



<https://networks.nokia.com/products/cloudband>

# Nokia VNFM API

# Nokia CloudBand Application Manager

## VNFM API Capabilities based on ETSI NFV IFA 007 / IFA 011

### Operation

#### **VNF lifecycle management**

Create VNF Identifier

Instantiate VNF

Scale VNF (scale in/out)

Terminate VNF

Delete VNF Identifier

Query VNF

Heal VNF

Modify VNF Information

Get Operation Status

#### **VNF lifecycle change notification**

Subscribe

Notify

### Operation

#### **VNF lifecycle management extensions, (to be standardized)**

VNF Upgrade

Modify VIM information

#### **VNF package management**

Query

Upload

Download

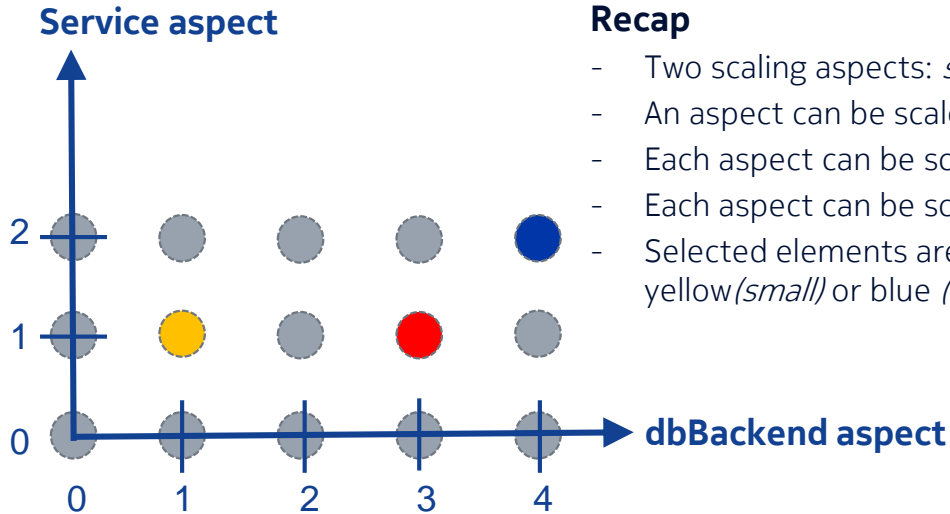
On-board / IFA 011

# VNF lifecycle management

## Hands on demo: how to scale a VNF

# VNFM lifecycle management

## Aspect based scaling in ETSI NFV SOL 003



### Recap

- Two scaling aspects: *service* and *dbBackend*.
- An aspect can be scaled by steps
- Each aspect can be scaled out to its maximum
- Each aspect can be scaled to in to 0.
- Selected elements are instantiation levels, e.g. yellow (*small*) or blue (*large*)



# VNF Lifecycle Management

## Demo application building blocks



### **Service Node**

- Accepts SQL queries from external clients
- Stateless



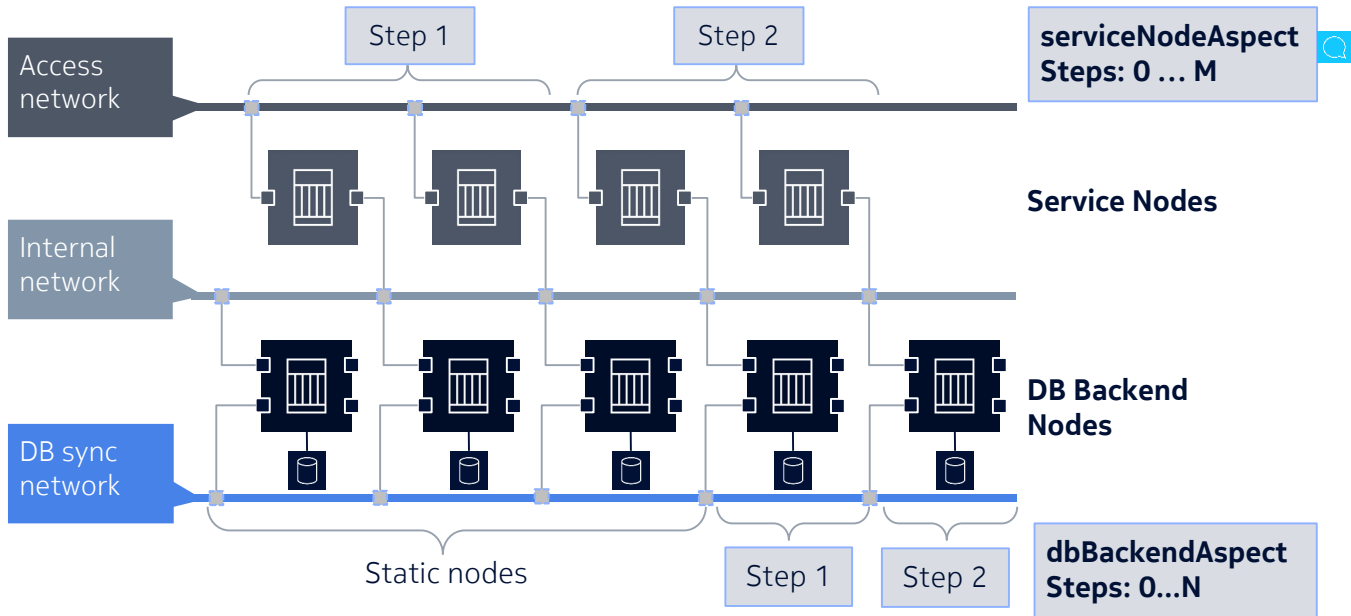
### **Database Backend**

- Stores the application data
- Responds to the service node queries
- Stateful



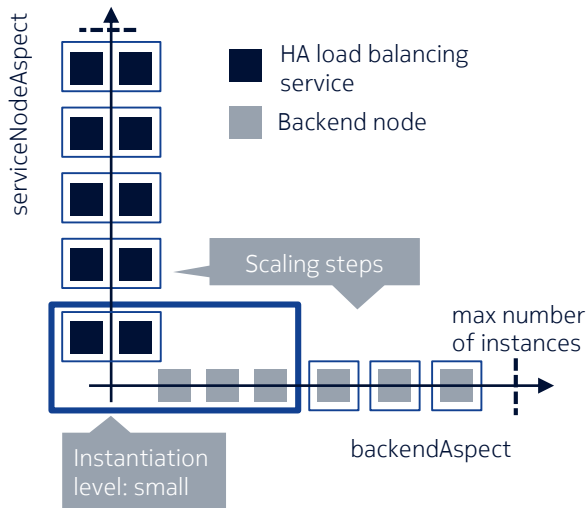
# VNFM lifecycle management

Demo application: let's make it scalable



# VNF Lifecycle Management

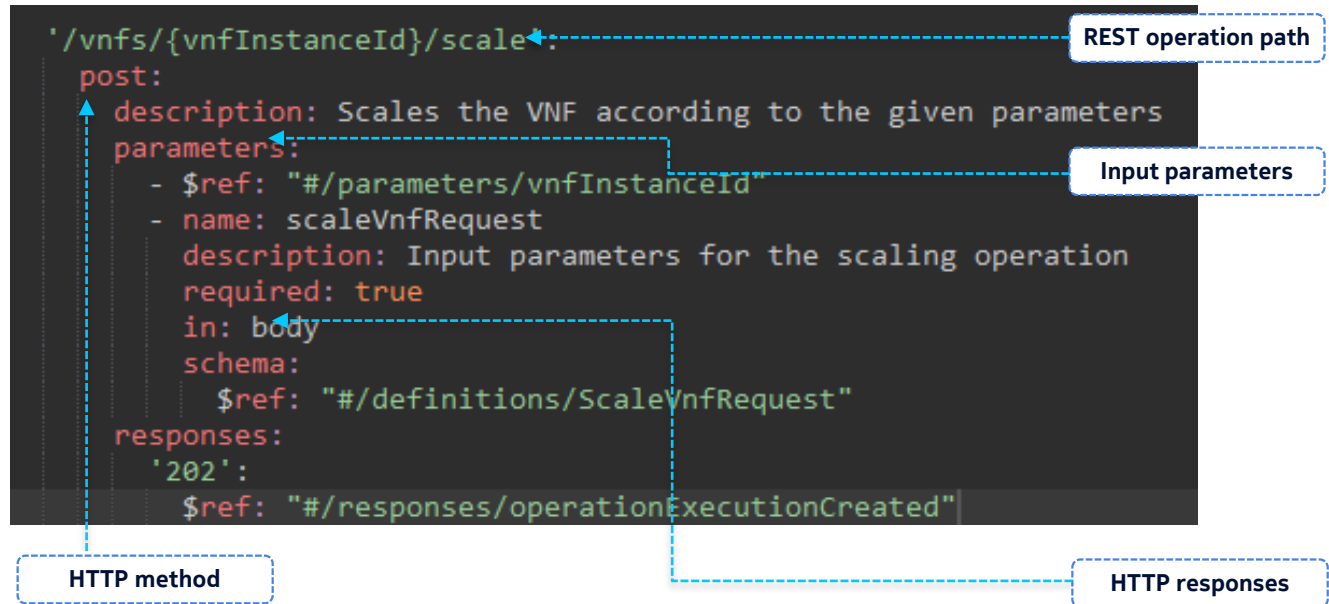
## Multi-dimensional scaling



- Two scaling aspects are used to test multi-dimensional scaling
- The default instantiation level is “small”, which contains
  - 3 backend nodes as static resources
  - 1 scaling step for serviceNodeAspect having 2 Service nodes
- Scaling is performed by adding / subtracting one or more steps
- Scaling is limited by caps in both aspects
- Host level Anti-Affinity rules applied for the for both static and scaled out VMs

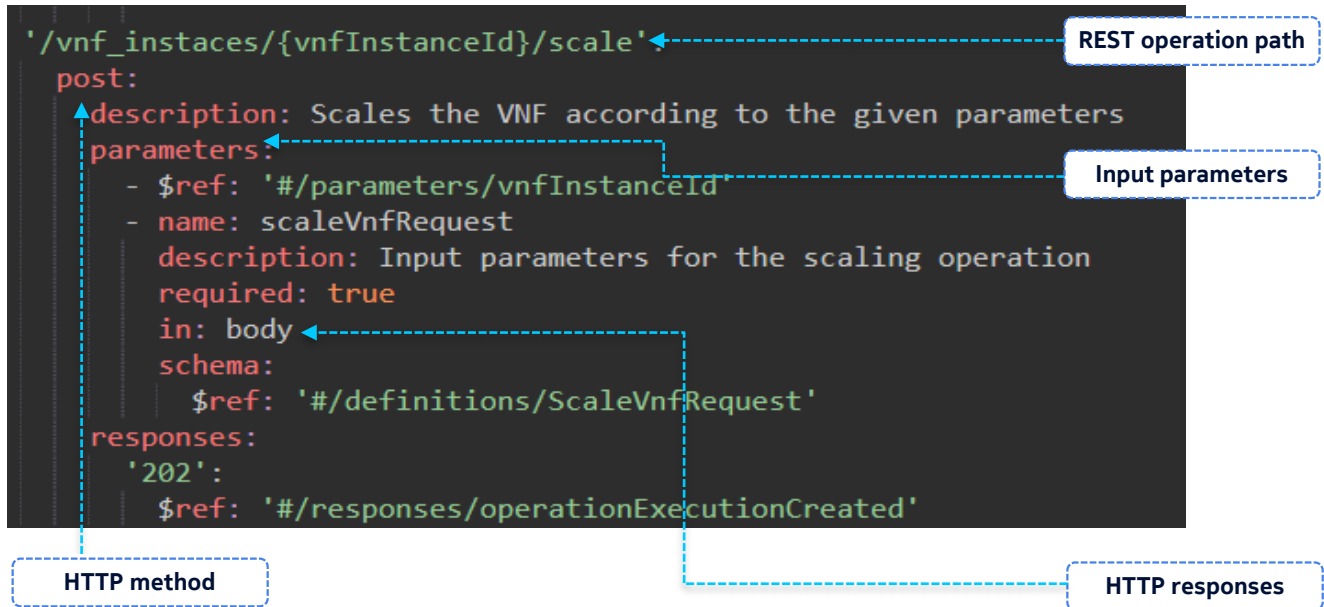
# VNF Lifecycle Management – Open API Schema

## Nokia CBAM VNF Implemented Scale Operation



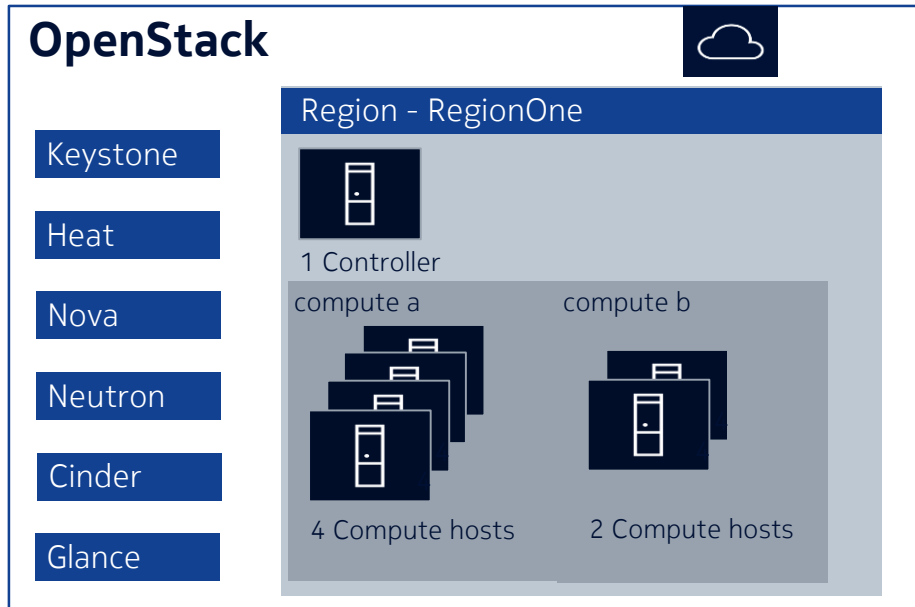
# VNF Lifecycle Management – Open API Schema

## Nokia CBAM Planned VNF Scale Operation



# VNFM lifecycle management

## Demo Environment : VIM type and structure




**VIM: Red Hat  
OpenStack  
Platform 9 (*Mitaka*)**

**VNFM: Nokia CBAM**



# VNF Lifecycle Management

## Scale VNF scenario one

- HTTP Request: 
  - Missing HTTP header: valid Oauth2 bearer token
- HTTP Response:
  - HTTP 401 Unauthorized

# VNF Lifecycle Management





## Scale VNF scenario two

- Pre-check: 
  - Query VNF information:  
dbBackendAspect : maxScaleLevel = 8
- Input:
  - Scale out dbBackend aspect with 100 steps
- Output: 
  - HTTP 422 – Unprocessable entity



# VNF Lifecycle Management

## Scale VNF scenario three

- Pre Check: 
  - VNF Query returns vnf\_id and database backend node names: <vnf\_id>-database\*
  - OpenStack Horizon contain three database backend VMs
- HTTP Request:
  - Scale dbBackend aspect with 2 steps
- HTTP Response
  - HTTP 202 Accepted 
  - Asynchronous operation started, takes 7 minutes to complete
  - VNF LCM operation occurrence link in header 
- Post-check
  - dbBackend aspect increased with two additional nodes 
  - OpenStack Horizon contains the five database backend VMs
  - Operation execution finishes successfully.

# VNF Lifecycle Management

## Scale VNF scenario four

- HTTP Request 
  - Scale out while a scale out operation is ongoing
- HTTP Response
  - 409 Conflict

# Conclusion

# Conclusions

- ETSI NFV IFA 007, IFA 008 and IFA 0011 provide a solid basis for the information model.
- ETSI NFV SOL 002 and 003 are great achievements, providing the required data models and protocol descriptions.
- And it works!
  
- Open API initiative is useful to ensure interoperability between the components

**NOKIA**