ARCTOS LABS

netrounds

WIND

telenor

PoC Demonstrates optimal placement in distributed (edge) cloud scenarios

Table of Contents

PoC key takeaways

- Constraint models complementing NSDs in order to capture service performance requirements
- Placement of VNF workloads based on latency requirements
- Placement decisions using real-time latency measurements
- Placement optimization using cost models to predict link and compute costs
- Placement optimization assurance to continuously re-evaluate in case of DC or link failures

Introduction

To address the increasing requirements on IT applications, communication service providers (CSPs) and enterprises need to make increasing use of edge compute.

This represents a huge automation and optimization challenge when it comes to optimally placing workloads and keeping costs under control.

Arctos Labs, Telenor, Netrounds and Wind River have jointly created a PoC to demonstrate viable concepts of how to address the challenges of automation.

The PoC uses optimization SW from Arctos Labs, live network monitoring SW from

Netrounds, and Open Source Mano (OSM) for SW lifecycle management on top of OpenStack from Wind River.

Challenge and context

Moving workloads closer to the source or destination of data (individuals, machines or companies) will generally improve the performance experienced by end users.

Doing so, however, can impose challenges and complications such as the following:

- Most authentic services contain multiple and interconnected (service SW) components and endpoints. Low latency requirements apply only to some of the components and/or links, implying that other components can be deployed in more central locations.
- Resources at edge compute locations are scarce and more costly to operate compared to resources at centralized locations.
- The data interaction between components and the users of services consumes transport capacity, which represents another constrained resource that comes with a cost.
- Placement of certain workloads may be further constrained by bandwidth, packet loss, security, privacy, integrity or reliability considerations.

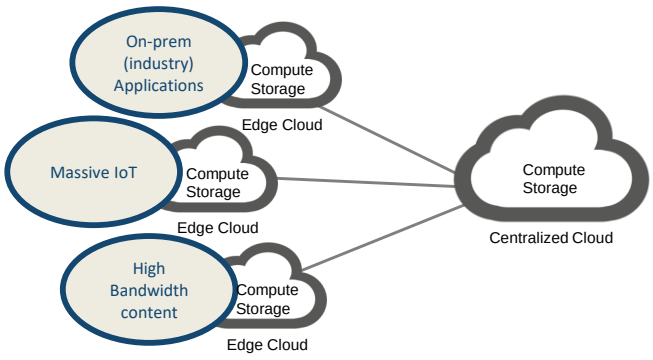

Figure 1: Introducing edge compute use cases

Note that placement challenges apply to network services, such as NFV, as well as to customer application services hosted within the network.

Introducing placement optimization

The term “fog computing” denotes a variety of edge computing which considers the possibility of placing SW components along an edge-to-central continuum, thereby enabling a more flexible placement.

The most cost-optimal placement for any SW component is most likely as central as possible, whilst still fulfilling the performance requirements of the overall service.

Placement optimization is not only about the cost and placement of compute; it also needs to take into consideration the networking interconnections available between customer premises, data sources or destinations, as well as the candidate compute PoPs in the network. The resulting application performance is a function of all of the above aspects.

The costs incurred as a result of such placement includes expenses related to allocated compute in edge or central locations, as well as the cost of transport that correlates with the components needed.

This implies that finding the optimal placement for each workload is a difficult optimization challenge that needs to take into account multiple parameters and consider all possible locations, their associated costs, and the resulting service performance.

Placement optimization is not only a one-time decision problem. As a matter of fact, it is a continuous optimization process in which the current state of the (cloud) network is constantly re-evaluated so that redeployments can be initiated when needed, for example due to infrastructure faults, capacity expansions, or new service deployments.

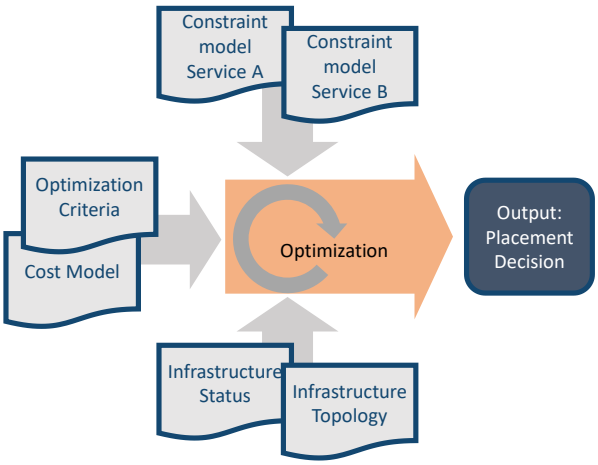

Figure 2: Overview of placement optimization

As outlined in Figure 2, optimization takes as input four categories of information:

- Service requirements – referred to here as constraints.
- Information about the underlying infrastructure – topology and status.
- A cost model that captures the costs related to utilizing certain infrastructure.
- An optimization criterion that formulates what is considered as optimal (e.g. lowest possible cost).

Potential scenarios for placement optimization

The above placement optimization is foreseen to be applicable in numerous scenarios, as outlined below in Figure 3.

Planning

When planning the underlying infrastructure and its topology, it is useful to be able to analyze the consequent service performance in "what-if" scenarios.

Network optimization

In scenarios where deployments are more static, it is still relevant to revisit the placement decisions made and consider further optimization possibilities by moving some of the components that have been deployed.

Service orchestration

In order to take automation further, a self-service approach is often considered where customers can launch their services in a zero-touch manner. Obviously, such an approach requires a fully automated placement optimization.

Assurance

Networks may be likened to living organisms in that parts of them may be temporarily faulty. This means that placement needs to be continuously re-evaluated in order to restore components that become unavailable due to faults.

All of the above applies for network services, such as NFV, and their placement – regardless of whether they are more statically planned or dynamically deployed – as well as for hosted service components that are deployed as a result of customer requests.

Furthermore, the edge compute infrastructure can be defined as a set of PoPs that are an integral part of a CSP network (such as central offices), but it can also extended to encompass public cloud locations and their corresponding costs.

Outlining the architecture of the PoC

The PoC architecture is built from three basic parts:

- A placement optimization solution provided by Arctos Labs.
- An active testing and monitoring solution that measures data plane performance and provides metrics such as available bandwidth, one-way delay, packet reordering and loss, as well as voice and video quality. Provided by Netrounds.
- These two parts sit alongside an open source MANO that provides lifecycle management of the service components.

The underlying network infrastructure is composed of four different data centers, each of which is managed by individual OpenStack instances.

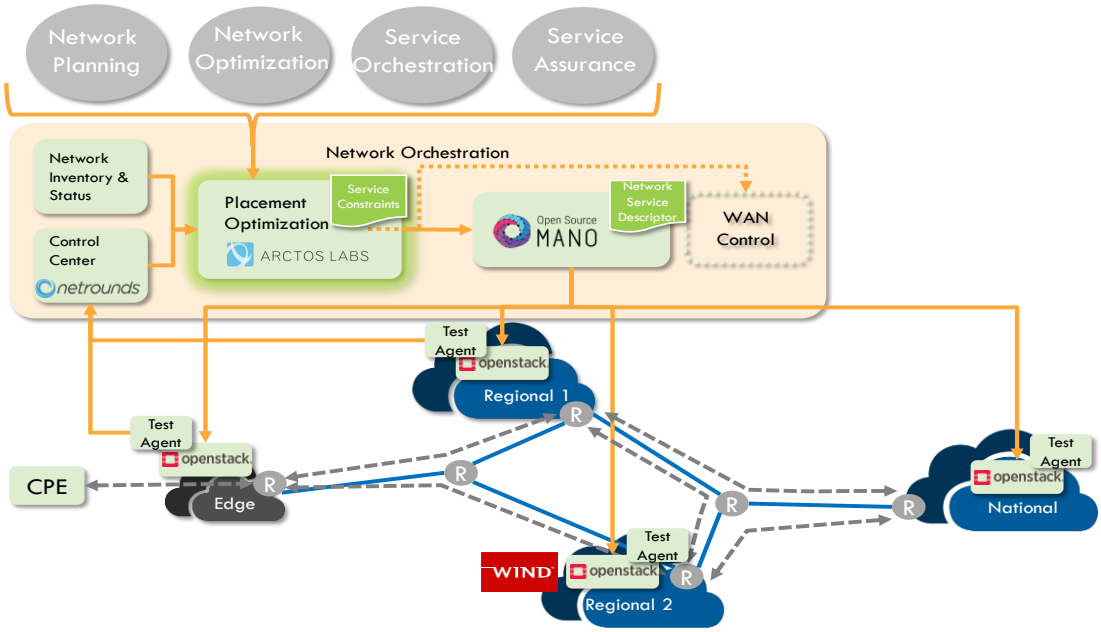

Figure 3: Outline of the PoC

PoC services and their constraints

The PoC demonstrates the placement of three different services. All of them have been imaginatively designed with respect to the constraints (requirements) they impose. The PoC uses the services as placeholders for different constraints that will drive different placement decisions.

All services consist of three VNFs (for demo visualization purposes), and the NSDs in OSM are therefore similar.

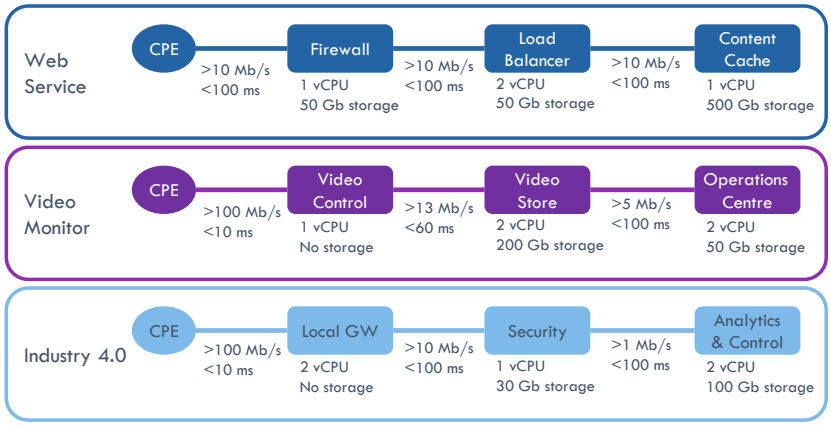

Figure 4: Three different services and their constraints

The PoC demo dashboard

The demo dashboard is used to control the demo as well as to visualize the resulting placement.

The dashboard contains four panels, each with a specific purpose.

Map

The map shows the four data centers and CPE locations and visualizes the resulting VNF deployments.

Inventory & cost model

This panel outlines the available data centers and their corresponding compute costs. The panel also includes live latency metrics for all DC-to-DC links collected by Netrounds Control Center by means of Test Agents deployed in each data center. Links are also attached with a cost model.

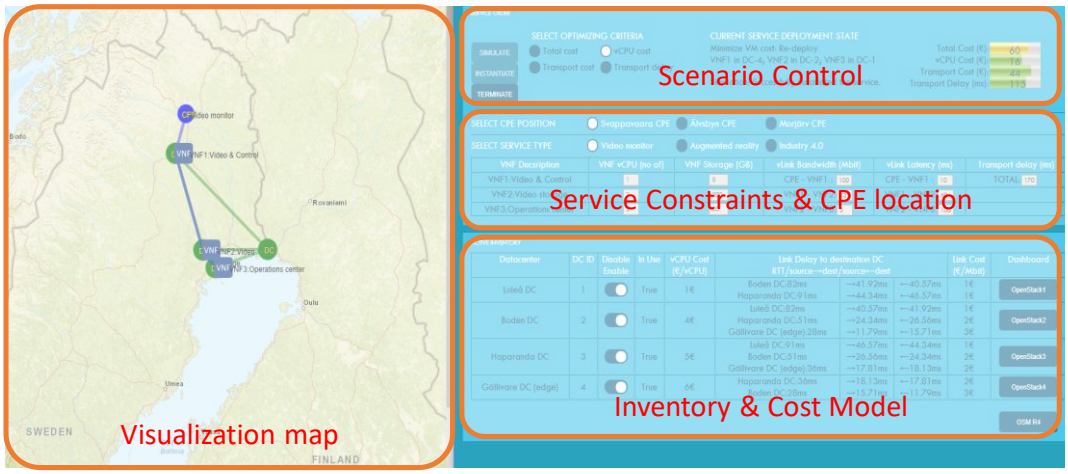

Figure 5: Overview of PoC demo dashboard

Service constraints and CPE location

Here the three onboarded services are displayed for selection. It is also possible to select one out of three predefined customer locations.

Scenario control

In this panel specific placement scenarios can be initialized, and the resulting costs are indicated.

Netrounds Control Center UI

This panel shows the network service quality between the different locations and is equipped with drill-down capabilities. In addition, it is possible to edit the testing and assurance templates that are used for the various services.

The PoC demo scenarios

Legend:

The green lines represent links between data centers. They are viewed as overlays on the real transport capacity; that is, they do not illustrate the actual topology of the transport network.

The blue lines illustrate the service links resulting from placement, including the CPE link.

1 - Deploying a no-latency-demanding service

The first PoC scenario is to deploy a traditional service with no specific latency requirements. This will result in the VNFs being deployed in the DC that has the lowest cost of compute.

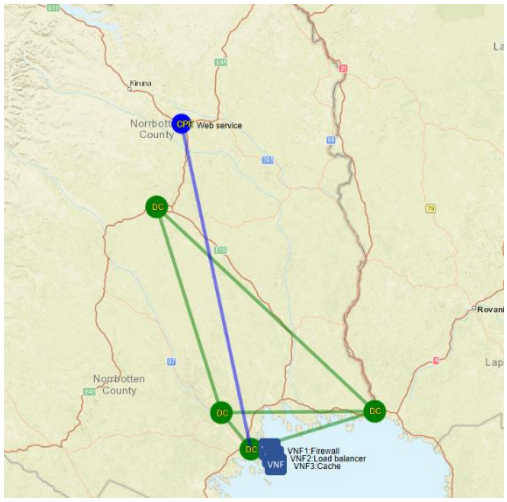

Figure 6: Placement of a non-demanding service

2 - Deploy a service with latency requirements

If the service contains specific latency requirements, some of the VNFs must be deployed closer to the customer. In this case, only the VNFs that unconditionally require low latency will be moved closer to the customer, whilst others can be placed in more cost-efficient locations, taking all cost parameters into account.

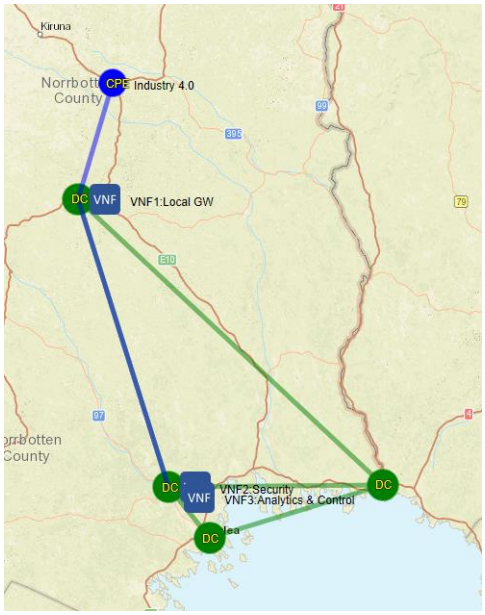

Figure 7: Placement of a latency demanding service

3 - Failover on DC failure

In this scenario, one of the DCs used for deployment fails. As the placement algorithm is active and constantly evaluates best placement, the service will be redeployed using the remaining DCs. This implies that the cost will be increased, but the service availability will be maintained.

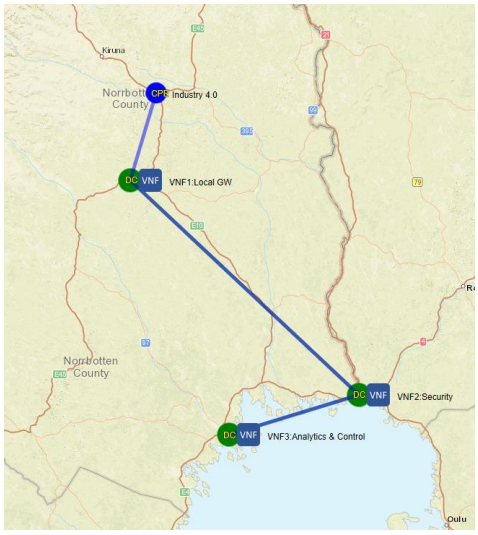

Figure 8: Redeploying on DC failure

Capturing network properties

A crucial feature of the solution is the ability to measure actual latency on links in the topology.

This is achieved by using Netrounds virtual Test Agents and Netrounds Control Center. Specific tests can be defined which actively measures the desired metrics on the data plane, such as latency, jitter, loss and throughput. The output from such tests can be used as input to placement decisions.

Figure 9: Netrounds Control Center displaying inter-DC latency

The Control Center can furthermore be used for troubleshooting and other actions in the event of failures, by drilling down into specific monitors and tests and the information they produce.

Figure 10: Detailed view of link properties

OpenStack & OSM dashboards

The four OpenStack dashboards as well as the OSM dashboard are available in the PoC to show actual deployments being made in the four DCs.

Figure 11 illustrates two NSDs. The first contains the initial set of Test Agents that are present as part of DC readiness to capture latency metrics which will be used for service placement. Secondly, the resulting service is illustrated with the VNFs being deployed in the selected DCs.

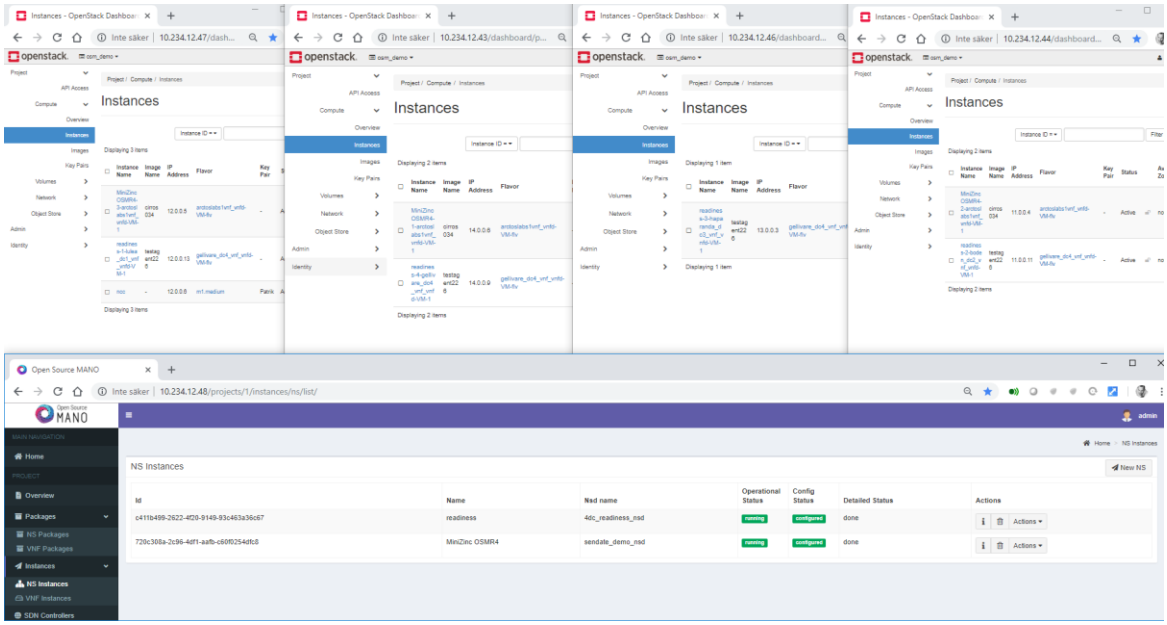

Figure 11: OpenStack and OSM dashboards