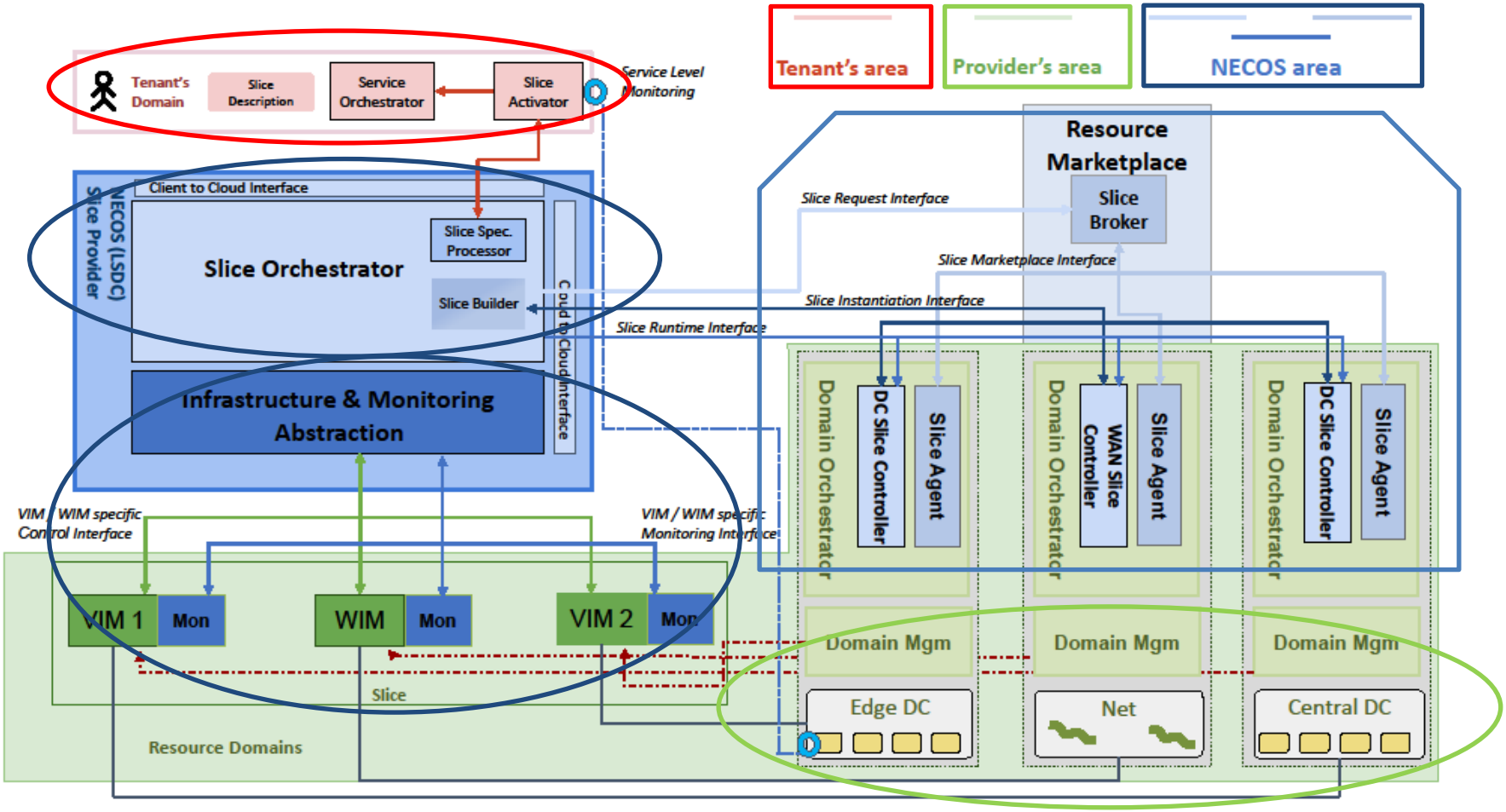
A Marketplace-based Approach to Cloud Network Slice Composition Across Multiple Domains

June 28, 2019
Polychronis Valsamas
Department of Applied Informatics University of Macedonia Thessaloniki, Greece
xvalsama@uom.edu.gr
IEEE Conference on Network Softwarization 2nd Workshop on Advances in Slicing for Softwarized Infrastructures (S4SI 2019)

Outline

- The NECOS Architecture
- Information Model
- Marketplace components and interactions
- Implementation Details
- Experimental Evaluation
- Summary & Next Steps

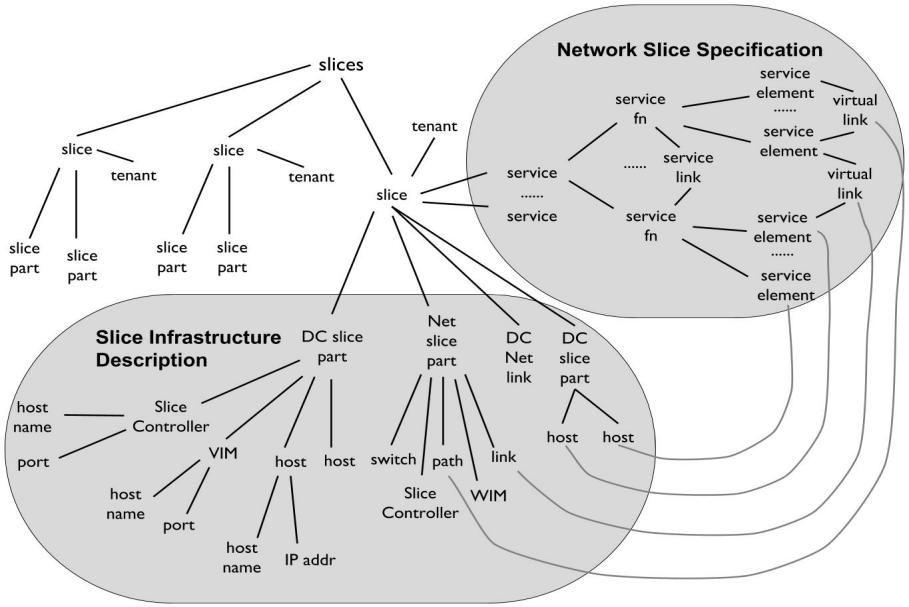
The NECOS Architecture

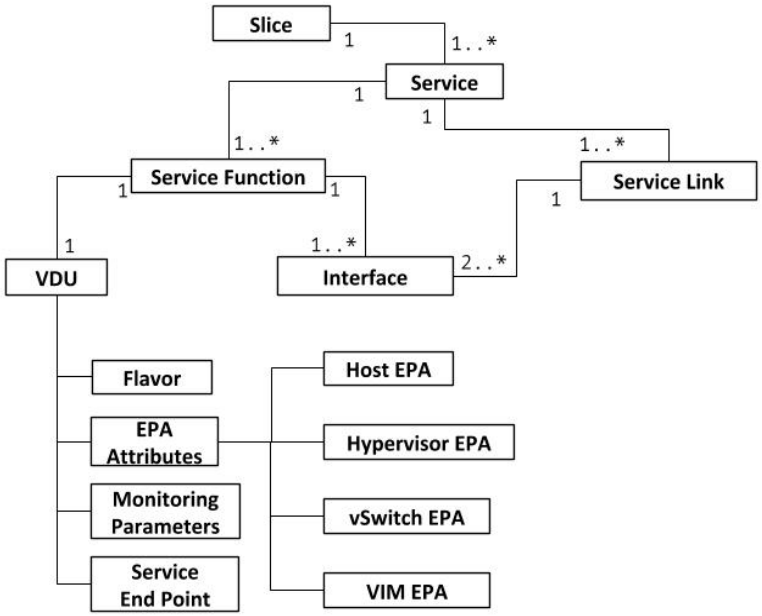


Information Model Overview

NECOS information model aims to providing a unified description of all information regarding a slice. Thus a detailed description of:

- slice parts, allocated infrastructure resources, and their properties
- services decomposed to service elements along with the necessary resource demands, deployed to these parts





YAML example

```
slice:
  id: TouristicCDN_sliced
  ...
  slice-constraints:
    geographic:
    dc-slice-parts:
    net-slice-parts:
   ...
  service:
    - service-function:
        vdu:
          id:
          ...
    - service-function:
        service-element-type: vdu
        vdu:
          id:
          epa-attributes:
            host-epa:
              cpu-model: 'single 3GHz'
              cpu-arch: X86_64
              cpu-vendor: Dell
              cpu-number: 1
              storage-gb: 256
             ....
    - service-link:
        service-element-type: link
        link:
          name:
          type: ...
```

```
slice:
    # definition of DC slice parts
    - dc-slice-part:
        name: dc-slice1
        vdus: …
    - dc-slice-part:
        name: dc-slice2
        vdus: …
        …
    - …
# definition of WAN slice parts
    - net-slice-part:
        name: external_ip_slice1-to-external_ip_slice2
        links:
          - dc-part1: dc-slice1
          - dc-part2: dc-slice2
        type: interconnect
```

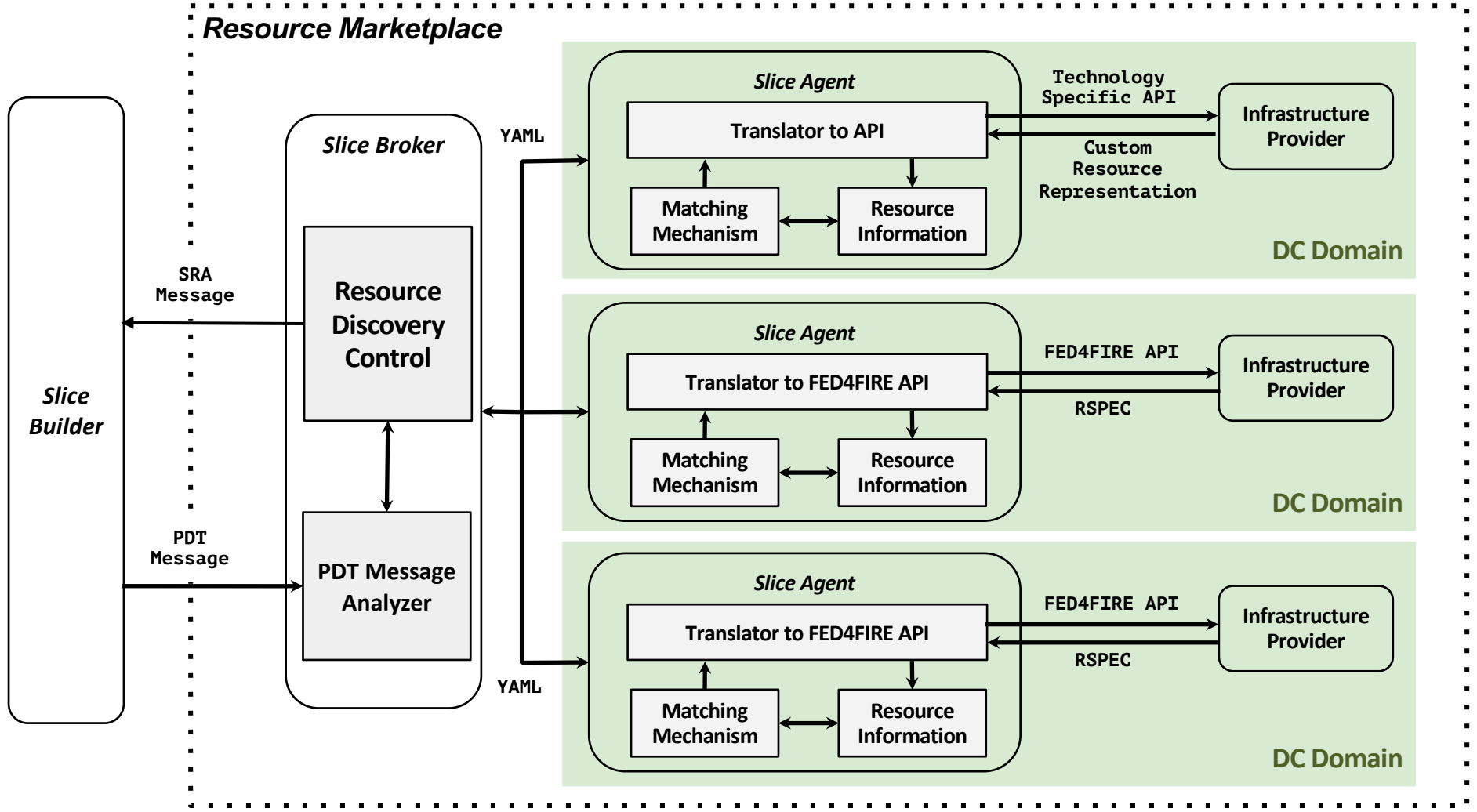
Slice Geographic Constraints

Service Descriptions

Service Specific Resource Flavor Constraints

Extended Attributes

Marketplace components and interactions (1/2)



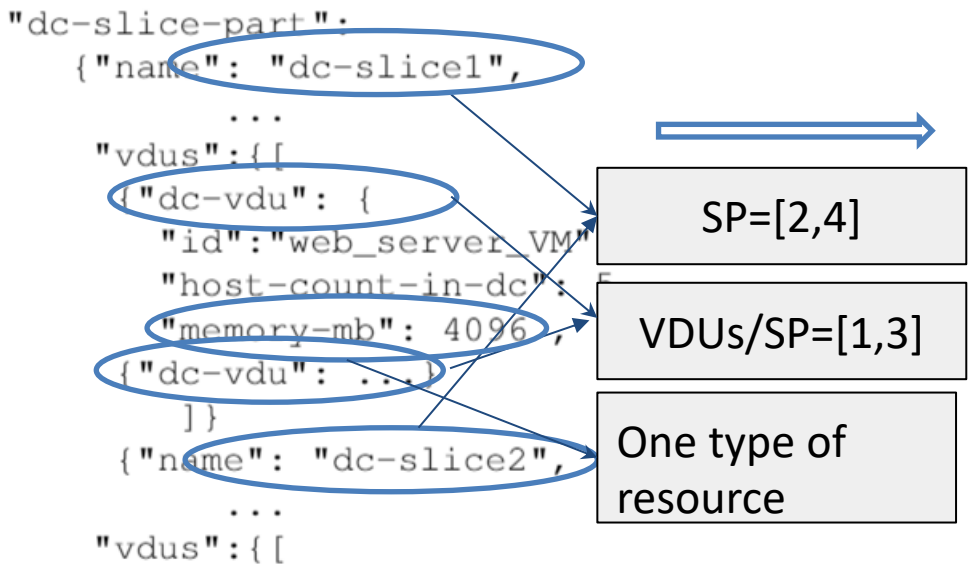
Marketplace components and interactions (2/2)

Marketplace includes the following main architectural components:

- Slice broker
  - PDT Message Analyser collect all the necessary information in the request message and translate into a suitable form for the slice agents
  - Resource Discovery Control is responsible to query all in the marketplace for each slice part and collect all the responses
- Slice agent
  - Translator API retrieves in real-time resources status directly with each test-bed and translating the response message in a more coherent format, in compliance to the NECOS information model
  - Matching Mechanism component translate the request originating form the slice broker to a set of resource availability constraints

Experimental Evaluation: Quantitative Results

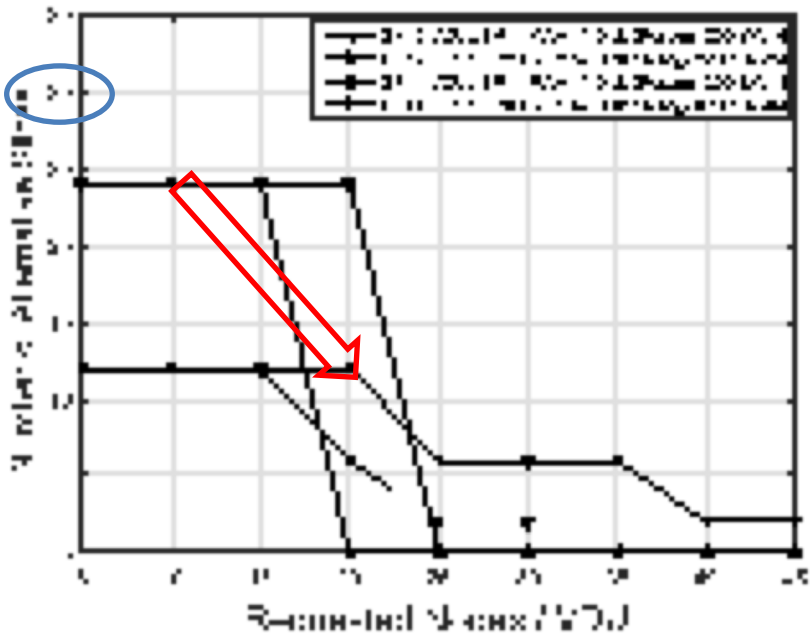
- Our experimental setting consists of six different test-beds (i.e., w-iLab2, Virtual Wall 1 & 2, Grid5000, Cloudlab in Utah and Wisconsin)
- Assumptions:
  - One DC provider is comprised of a set of node clusters
  - One DC slice part per DC provider
  - Each *dc-vdu* assigned to a specific node cluster



Experimental Evaluation: Quantitative Results

```
"dc_slice_part":
   {"name": "dc_slice1",
          ...
    "vdus":{[
     {"dc_vdu": {
       "id":"web_server_VM",
       "host_count_in_dc": 5,
       "memory_mb": {greater_or_equal: 4096}
       "storage_gb: {greater_or_equal: 200},
       "nics_bw": {greater_or_equal: 4}
       ...},
     {"dc_vdu": ...}
        ]}
     {"name": "dc_slice2",
          ...
    "vdus":{[
        }}
```

Three types of resources



- In both graphs the number of alternative solutions decreases as the demand for requested nodes increases
- The number of alternative solutions decreases, as the complexity of the resource requirements increases

Experimental Evaluation: Qualitative Results

Slice consists of Core and Edge Cloud Servers

Core Servers have higher resource demands	Edge Servers low resource demands

```
dc_part(dc-slice-1,location:'undefined',
 ['dc-vdu'(vdu11,[
   'available-nodes'>=2,'memory-gb'>64,
   'min-storage-gb'> 250,'nics-bw'> 2]),
  'dc-vdu'(vdu12,[
  'available-nodes'>=1,'memory-gb'>=128,
  'min-storage-gb'>=500,'nics-bw'>=5])
])
```

```
dc_part(dc-slice-2,location:'undefined',
 ['dc-vdu'(vdu21,[
   'available-nodes'>=1,'memory-gb'>=8,
   'min-storage-gb'>=80,'nics-bw'>=1]),
  'dc-vdu'(vdu22,[
   'available-nodes'>=1,'memory-gb'>=8,
   'min-storage-gb'>=80,'nics-bw'>=1])
]),
```

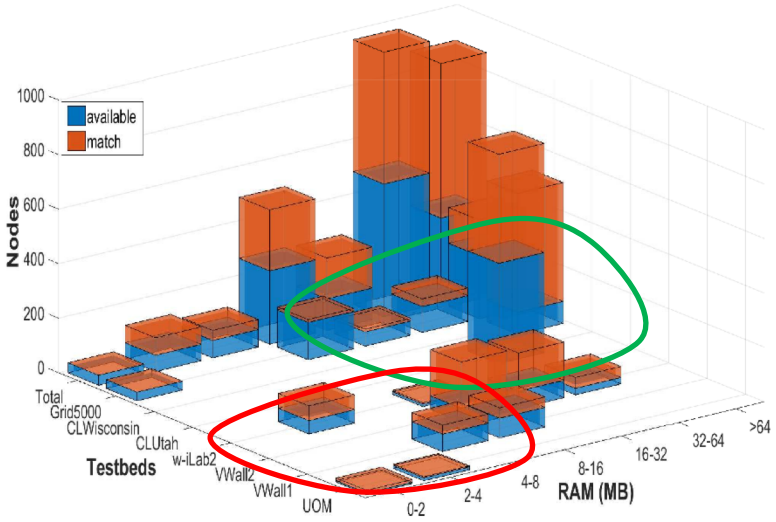
Undefined geographic Constraints

Resource demands in terms of CPU, memory, storage, bandwidth per dc-vdu

Experimental Evaluation

Slice Part 1	*Slice Part 2*	*Slice Part 3*	*Slice Part 4*
Grid5000	CLabUtah	CLabWisconsin	VWall2
Grid5000	CLabUtah	VWall1	VWall2
Grid5000	CLabUtah	w-iLab2	VWall2
Grid5000	CLabWisconsin	CLabUtah	VWall2
Grid5000	CLabWisconsin	VWall1	VWall2
Grid5000	CLabWisconsin	w-iLab2	VWall2
Grid5000	VWall1	CLabUtah	VWall2
Grid5000	VWall1	CLabWisconsin	VWall2
Grid5000	VWall1	w-iLab2	VWall2

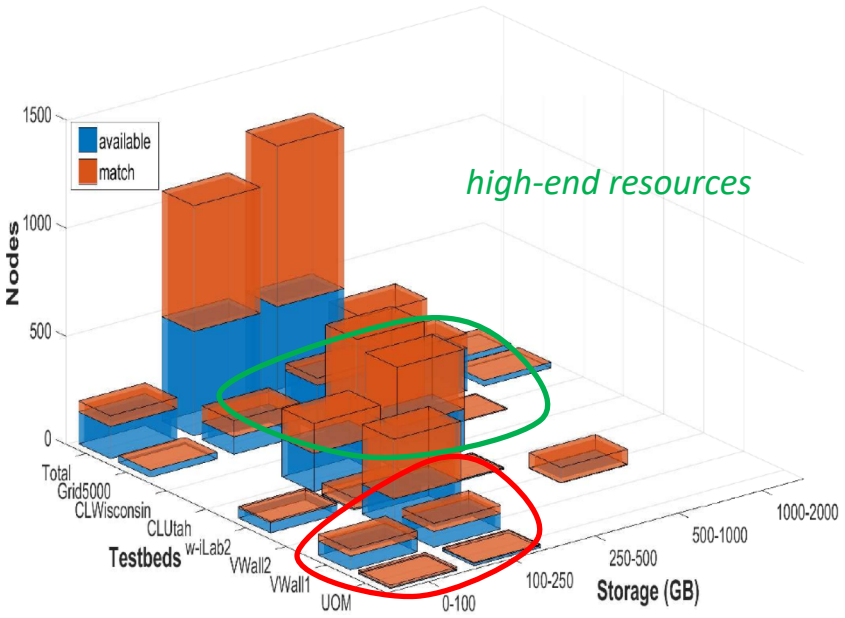
Resource Availability Investigation



Resources satisfying memory criteria

high-end resources

low-end resources



high-end resources

Resources satisfying disk storage criteria

low-end resources

Summary

- We highlighted the NECOS Marketplace approach handling slice requests over multiple infrastructure providers
- We described in detail the main Marketplace components
- We carried out real experiments utilizing real measurements on the resource availability of variety of open-access test-beds.

Next Steps

The Marketplace approach faces interesting challenging issues, including:

- **Scalability**:
  - The Marketplace concept can scale up to a plethora of providers,
  - Resource requests handled can involve a large number of parallel slices/resources
- **Performance:**
  - Number of messages exchanged, trade-offs, time to respond, etc.
- **Elasticity:**
  - Extend and experiment with slice resource discovery workflows for slice elasticity
- **Heterogeneity**:
  - Resource discovery coping with a diverse range of server specifications.
- **Cost efficiency (Business models):**
  - Selection of the minimum cost slice among alternatives.

Q&A

Relate Work

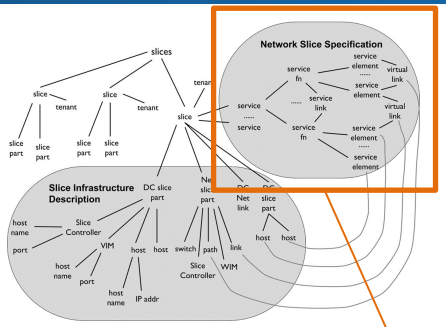
[1] I. Afolabi, A. Ksentini, M. Bagaa, T. Taleb, M. Corici, and A. Nakao, "*Towards 5G network slicing over multiple domains*," IEICE Transactions on Communications, Special section on Network Virtualization, Network Softwarisation, and Fusion Platform of Computing and Networking. Vol 100B, N11, November 2017 , 11 2017. [Online]. Available: http://www.eurecom.fr/publication/5375

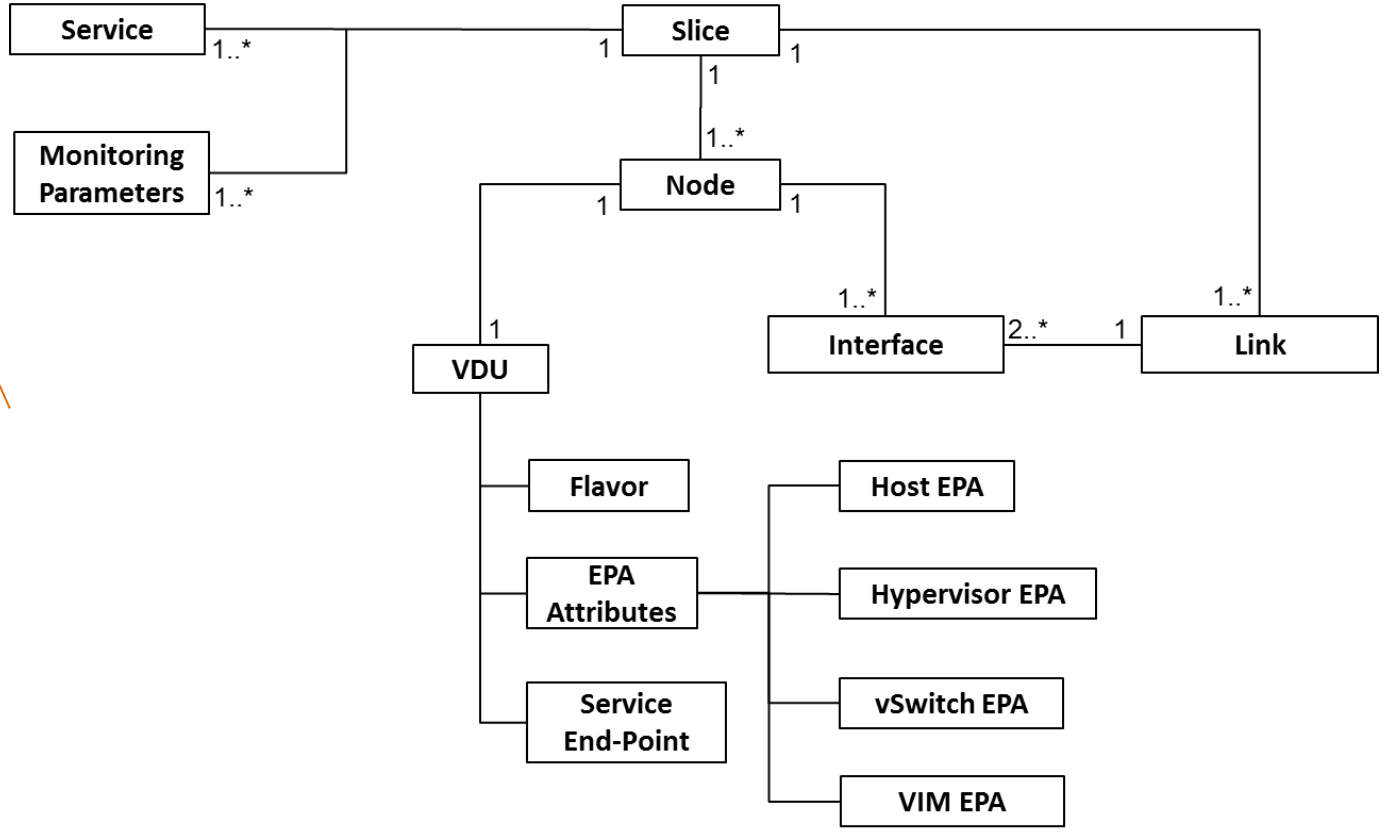
[2] K. Katsalis, N. Nikaein, E. Schiller, A. Ksentini, and T. Braun, "*Network slices toward 5g communications: Slicing the lte network*," IEEE Communications Magazine , vol. 55, no. 8, pp. 146–154, 2017.

[3]A. Boubendir, F. Guillemin, C. Le Toquin, M.-L. Alberi-Morel, F. Faucheux, S. Kerboeuf, J.-L. Lafragette, and B. Orlandi, "*Federation of cross-domain edge resources: a brokering architecture for network slicing*," in 2018 4th IEEE Conference on Network Softwarization and Workshops (NetSoft) . IEEE, 2018, pp. 415–423.

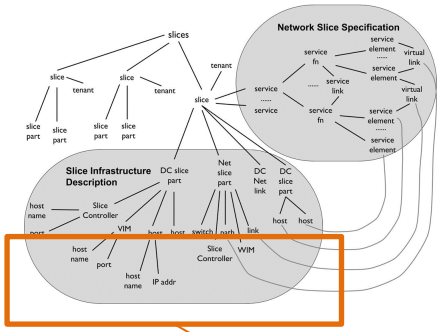
[4] P. Twamley, M. Muller, P.-B. Bok, G. K. Xilouris, C. Sakkas, M. A. Kourtis, M. Peuster, S. Schneider, P. Stavrianos, and D. Kyriazis, "*5gtango: An approach for testing nfv deployments*," in 2018 European Conference on Networks and Communications (EuCNC) . IEEE, 2018, pp. 1–218.

Information Model: Slice View





Information Model: Infrastructure View



Infrastructure
- Infrastructure ID
- Domains

Router
- Router ID
- Availability
- Role (edge/core)
- Processing (IP, MPLS)
- Monitoring parameters

Domain
- Domain ID
- Provider
- Type (DC, MEC, WAN)
- Hosts
- Network Elements
- Links

CPU
- Cycles
- Cores
- Model
- Architecture
- Instruction set

Host
- Host ID
- Hostname
- Availability
- Location
- CPU
- Memory
- Storage
- Ports
- Monitoring parameters

Switch
- Switch ID
- Availability
- Role (ToR, Agg., Root)
- Monitoring parameters

Network Element
- Network element ID
- Availability
- Type (Router, Switch, AP)
- Ports
- FIB size

Path
- Path ID
- Availability
- Links
- Capacity
- Delay
- Jitter

Link
- Link ID
- Availability
- Type
- Capacity
- Delay
- Jitter

Queue
- Queue ID
- Availability
- Capacity

Port
- Port ID
- Availability
- Capacity
- Queues
- IP address
- MAC address

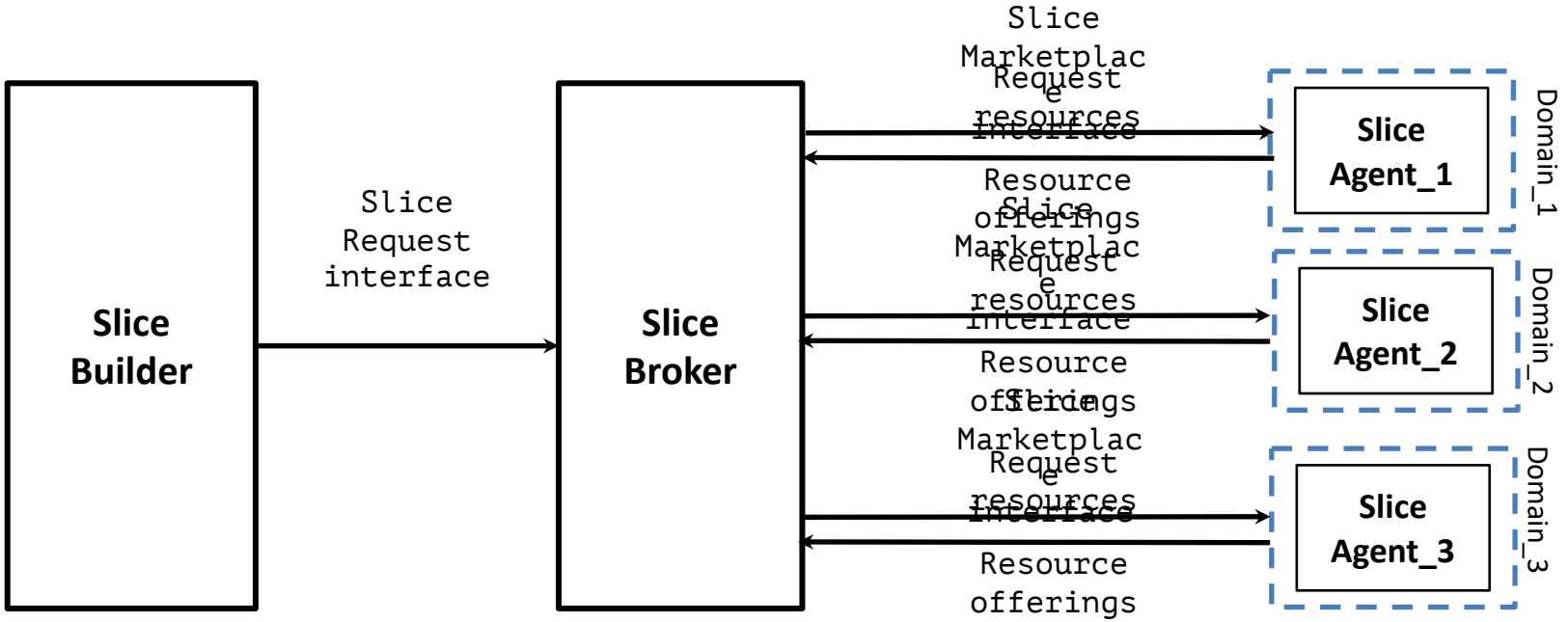
Controller
- Controller ID
- Role (DC, Network)
- Configuration protocol
- Configuration interface

DC Provider Resource Description

- Resource Description comes from FED4FIRE testbeds.
  - Real Time data from Experimental testbeds
  - Experimentation: *Snapshots* of resource descriptions and utilizations.


```
"testbed": {
  "id": "wilab2",
  "location": "Europe",
  "nodes": [
   {   "node_cluster": {
       "available_nodes": 31,
       "cpu_cores": 2,
       "cpu_ghz": 1.8,
       "cpu_number": 2,
       "cpu_vendor": "Intel ",
       "memory_mb": 4096,
       "min_storage_gb": 160,
       "name": "ZOTAC ",
       "storage_description": "160",
       "total_nodes": 45 }   },
   {
       "node_cluster": {
       "available_nodes": 34,
       "name": "APU 1d4",
       …
       "total_nodes": 43 }  },
```

An abstract view of the resource discovery workflow



- For each slice request submitted by the **Slice Builder**:
  - The **Slice Broker** requests resources from each Slice Agent
  - The **Slice Agent** matches the requested to the available resources and returns a set of resource offerings
  - The **Slice Broker** filters among the resource offerings for the slice creation all those that form feasible resource alternatives

Introduction

- Novel solutions in search of flexibility, agility, cost efficiency: (Cloud) Network Slicing
- Services & Vertical industries may bring diverging use cases and application scenarios
- Development of an integrated system for enabling cloud networking slicing capabilities in multidomain scenarios (native integration of cloud computing and advanced networking)
- The **NECOS** platform implements the Slice-as-a-Service model, enabling the dynamic creation of end-to-end (E2E) slices from a set of constituent slice parts contributed from multiple domains
- Optimal allocation of resources to slices in the cloud and networking infrastructure, to respond to the dynamic requests of the various service demands
- A **Marketplace** approach to the composition of Cloud Network Slices across multiple domains