

# Network Application Onboarding

Features of the 5G-IANA front-end interface



Dimitris Klonidis, Thanos Xirofotos, Stelios Tsarsitalidis



5G-IANA project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101016427.

# Webinar link



- Follow this [LINK](#) for the complete video presentation
- Duration ~ 1h

## ▪ Direct link

- <https://istitutoboella.sharepoint.com/:v:/r/sites/5G-IANA/Documenti%20condivisi/General/2022-11-22%205G-IANA%20UBI%20webinar.mp4?csf=1&web=1&e=0TrEe>

# Outline



- Role of Application Orchestrator in 5G-IANA architecture
- Features of Application Orchestrator
- NetApp onboarding workflow
  - **Application or Network Function Definition**
  - **NetApp composition and/or extension**
  - **NetApp instantiation**
  - **NetApp policy setting**
  - **NetApp monitoring**

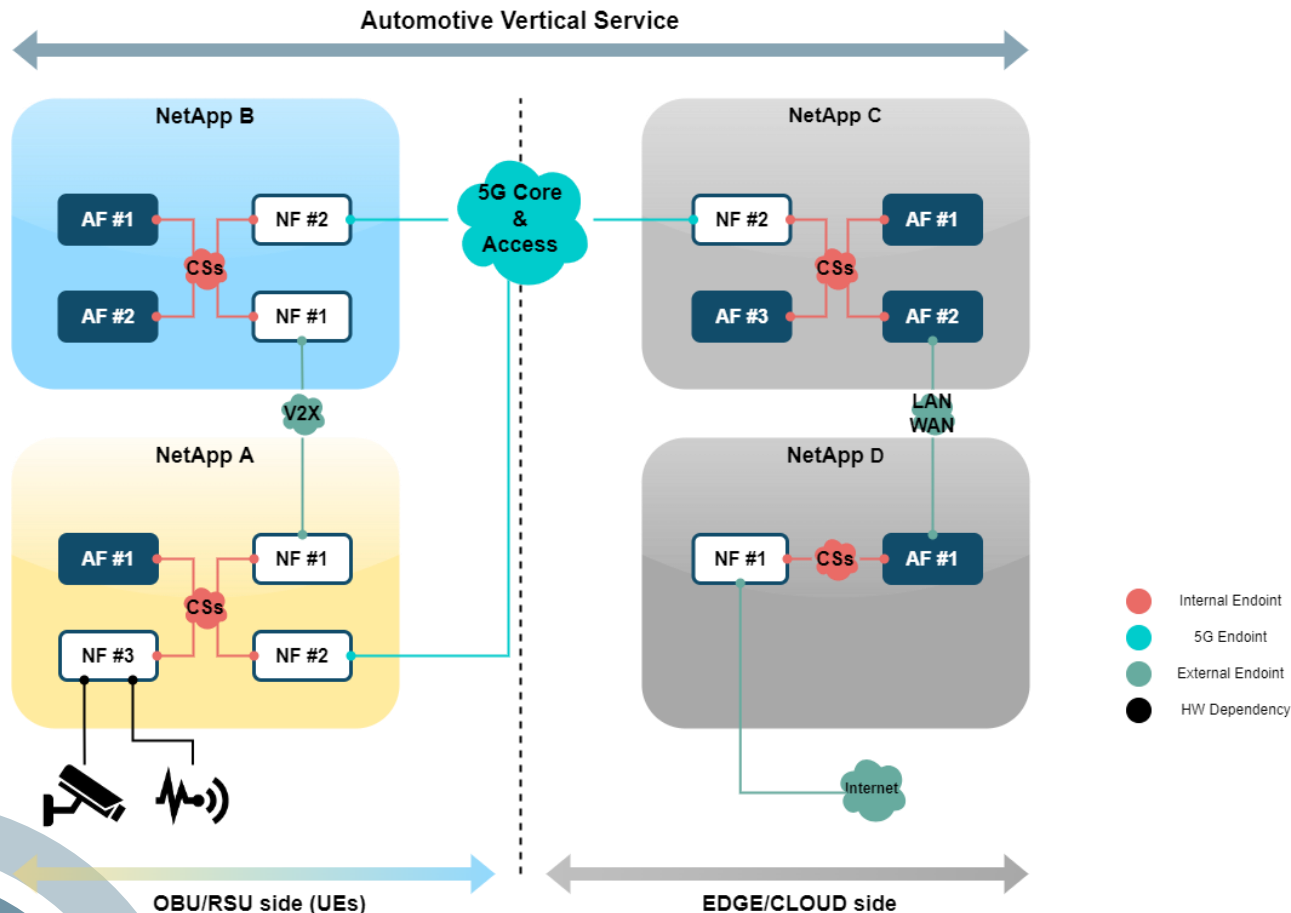


# Application Orchestrator Features



- Enables the **deployment of Network Applications (NetApps) over programmable infrastructure** (or in generic terms deployment of networked Vertical Services)
  - Vertical Service is seen as a series of linked functions forming a Network Application
  - Application/Network Functions are independent stand alone and reusable SW components
  - Upon deployment of Network Applications, the requested set of resources is identified and reserved forming a network slice
- Enables the **reconfiguration during runtime**, based on:
  - established policy criteria
  - end user dynamic requests
  - monitoring information from the network
- **Monitors the operational status** of the deployed Application Functions, and several of their metrics, and allows end users to visualize them
- **Optimizes deployments** in response to the monitoring of Vertical Service metrics

# NetApp and Functions definition



- Functions are Application specific or network specific
- A NetApp is formed as a series of linked functions able to provide an end-to-end application over the network infrastructure
- A NetApp can be just one functions if this can offer an end-to-end service
- Multiple-stand alone NetApps can be linked to form another NetApp (as a more complete type of service)
- Functions and generic NetApps are reusable, but NetApps for deployment over a certain infrastructure have target specific features.

# NetApp Onboarding Workflow

Steps taken to onboard and manage Functions and NetApps through the Application Orchestrator

# NetApp Onboarding and Operation steps



1. Definition of software to constitute an Application/Network Function
  - **Registration of the function as software component with specific parameters, OR**
  - **Pulling of software component from external catalogue (direct use/reuse)**
2. Formation and update of NetApps (using NetApp catalogue) by
  - **Functions are linked to form a NetApp graph, by connecting exposed interfaces with required ones, OR**
  - **Existing NetApp is reused, altering parameters, or wiring together new function with existing NetApp**
3. Instantiation of a NetApp, by selecting available (target specific) resources, and adding deployment parameters and policies
4. Selection of monitoring parameters to be visualized
5. Live monitoring of NetApp instances

# *Application or Network Function Definition*



- Docker software image to be used, via Distribution Parameters
- Architecture and Elasticity Controller selection
- Execution / resource requirements selection
- Health check, container execution and environment variables definition
- Exposed and consumed interfaces definition
- Volumes, devices and labels definition
- Default advanced parameters definition

# Application or Network Function Definition – Through UI



- DASHBOARD
- INSTANCES
- APPLICATIONS
- COMPONENTS
- RESOURCES
- DOMAIN NAME
- PLUGINS
- WORKSPACE

## Components

Create new

Name

Filter Reset

Identifier	Name ^	Organization	Visibility	Date Created	
v2c68vit9w	WordPress	Admin_Organization	Public	12/11/2022 - 19:09	⋮
twDvum9KKi	Traefik	Admin_Organization	Public	12/11/2022 - 19:09	⋮
kex4t7zx3c	SumFunction	Admin_Organization	Public	12/11/2022 - 19:09	⋮
1mux7czqfb	PPDRSamba	Admin_Organization	Public	12/11/2022 - 19:09	⋮
jnfsdwc4vc	PPDRPhpDashboard	Admin_Organization	Public	12/11/2022 - 19:09	⋮

<< < > >>



# Application or Network Function Definition - Parameters



```
{
  "id": 0,
  "name": "string",
  "hexID": "string",
  "architecture": "string",
  "elasticityControllerMode": "string",
  "iconBase64": "string",
  "dockerImage": "string",
  "dockerRegistry": "string",
  "dockerCredentialsUsing": true,
  "dockerCustomRegistry": true,
  "dockerUsername": "string",
  "dockerPassword": "string",
  "exposedInterfaces": [{
    "interfaceID": 0,
    "component": "string",
    "name": "string",
    "port": "string",
    "vna": "string",
    "interfaceType": "string",
    "transmissionProtocol": "string",
    "dateCreated": "2022-11-16T11:30:55.236Z",
    "lastModified": "2022-11-16T11:30:55.236Z"
  }],
}
```

Basics

Docker Image to be used

Exposed Interface definition

# Application or Network Function Definition - Parameters

```
"requirement": {  
  "requirementID": 0,  
  "component": "string",  
  "getvCPUs": 0,  
  "ram": 0,  
  "storage": 0,  
  "hypervisorType": "string",  
  "gpuRequired": true,  
  "dateCreated": "2022-11-16T11:30:55.236Z",  
  "lastModified": "2022-11-16T11:30:55.236Z"  
},  
"healthCheck": {  
  "healthCheckID": 0,  
  "component": "string",  
  "name": "string",  
  "httpURL": "string",  
  "args": "string",  
  "interval": 0,  
  "dateCreated": "2022-11-16T11:30:55.236Z",  
  "lastModified": "2022-11-16T11:30:55.236Z"  
}
```

Execution Requirements

Health Check

# NetApps



- Name and create new NetApp
- Pick Functions from the ones available in the list
  - **Application or Network functions will appear in this list**
- Pick NetApps to include or extend
  - **There will exist a list of NetApps to be reused, within the NetApp composer**
  - **Reusable NetApps will be reside in a NetApp catalogue**
  - **The list will be similar to the Functions/Components picker**
- Make connections between interfaces of the functions
  - **Drag and drop connections from required interfaces to exposed ones in the graph**



Create new

## Applications

Name

[Filter](#) [Reset](#)

Identifier	Name	Organization	Visibility	Date Created	
qmepvg98xp	CMSApp	Admin_Organization	Public	12/11/2022 - 19:09	⋮
4k52dc5trs	DBMS	Admin_Organization	Public	12/11/2022 - 19:09	⋮
em70qywuqu	FunctionPilot	Admin_Organization	Public	12/11/2022 - 19:09	⋮
xv8ynsgbh0	olistic	Admin_Organization	Public	12/11/2022 - 19:09	⋮
0shxqwajz3	PPDR	Admin_Organization	Public	12/11/2022 - 19:09	⋮

- DASHBOARD
- INSTANCES
- APPLICATIONS
- COMPONENTS
- RESOURCES
- DOMAIN NAME
- PLUGINS
- WORKSPACE



# NetApp creation (1) – Pick Components



Application name



MyTestNetApp



MyTestNetApp

(Public) If this option is checked, anyone could see this Application

Search a component

Search...

- + Traefik
- + LambdaProxy
- + MariaDB
- + MongoDB
- + phpMyAdmin
- + WordPress
- + HttpEcho

Save



(Public) If this option is checked, anyone could see this Application

Search a component

Search...

- + Traefik
- + LambdaProxy
- + MariaDB
- + MongoDB
- + phpMyAdmin
- + WordPress
- + HttpEcho

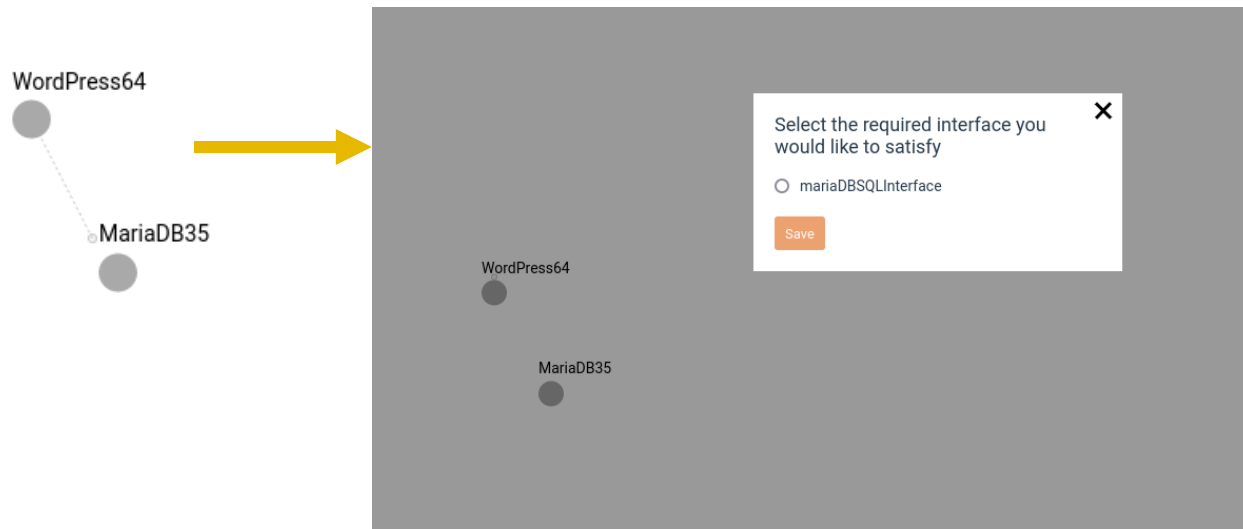
Save

MongoDB41



22/11/2022

# NetApp creation (3) – Connected interfaces



**MAESTRO** applications development and orchestration

MyTestNetApp



(Public) If this option is checked, anyone could see this Application

Search a component

Search...

---

**Generic info:**  
ID: t3tsy1ve7a  
Name: MariaDB

**MariaDB35**  



**Required Interfaces:**  
1. N/A

**Exposed Interfaces:**  
1. Interface: mariaDBSQLInterface

**Save**



**Generic info:**  
ID: v2c68vit9w  
Name: WordPress

**WordPress64**  

**Required Interfaces:**  
1. Interface: mariaDBSQLInterface

**Exposed Interfaces:**  
1. Interface: WordPressAccessInterface



# NetApp instantiation



## Applications

Create new

Name  
Search by Name

Filter Reset

Identifier	Name	Organization	Visibility	Date Created	
qmepvg98xp	CMSApp	Admin_Organization	Public	12/11/2022 - 19:09	Create Instance Delete

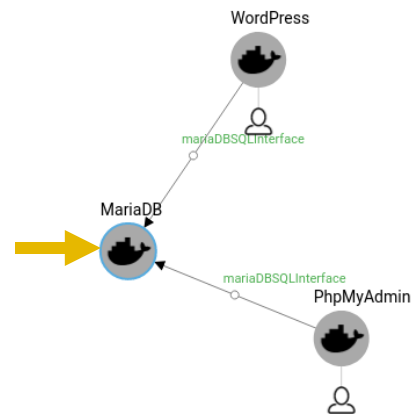


TestNetAppInstance

Select Provider  
UBIDELL

Enable End-To-End Encrypted IPv6 Communication  
 Enable Soc on each component Node Instance

Proceed



### Configure "MariaDB" Component

(NetApp creation (1) - Pick Co...)

General System Security Workers Flavor Health Check Container Execution Interfaces Environmental Variables Plugin

**General**

Select SSH Key  
adminKey

Select Provider  
UBIDELL

Select Region \*

# NetApp instances



## Application Instances

Name: Search by Name      Status: -- Select --      Application: -- Select --

[Filter](#) [Reset](#)

Identifier	Name	Application Name (Hex ID)	Status	Date Created
ebhqr7a4q	billy	mbtrst (mt33au5i55)	<a href="#">View Graph</a> <a href="#">Profiling</a> <a href="#">Elasticity Policies</a> <a href="#">Security Dashboard</a> <a href="#">Soc-Policies</a> <a href="#">Analytics</a> <a href="#">SLO</a> <a href="#">Un-deploy</a>	
exfy8mivab	DS	DBMS (atnwnbjjj6)	DEPLOYED	17/11/2022 - 16:45
c58826jsjt	ubitest	UWSv2 (p0sg4x1gqn)	DEPLOYED	29/09/2022 - 12:03



# NetApp monitoring



- Gathering of monitoring data, regarding performance, through netdata
- Storage of monitoring data in Prometheus
- Visualization of monitoring data through time is included

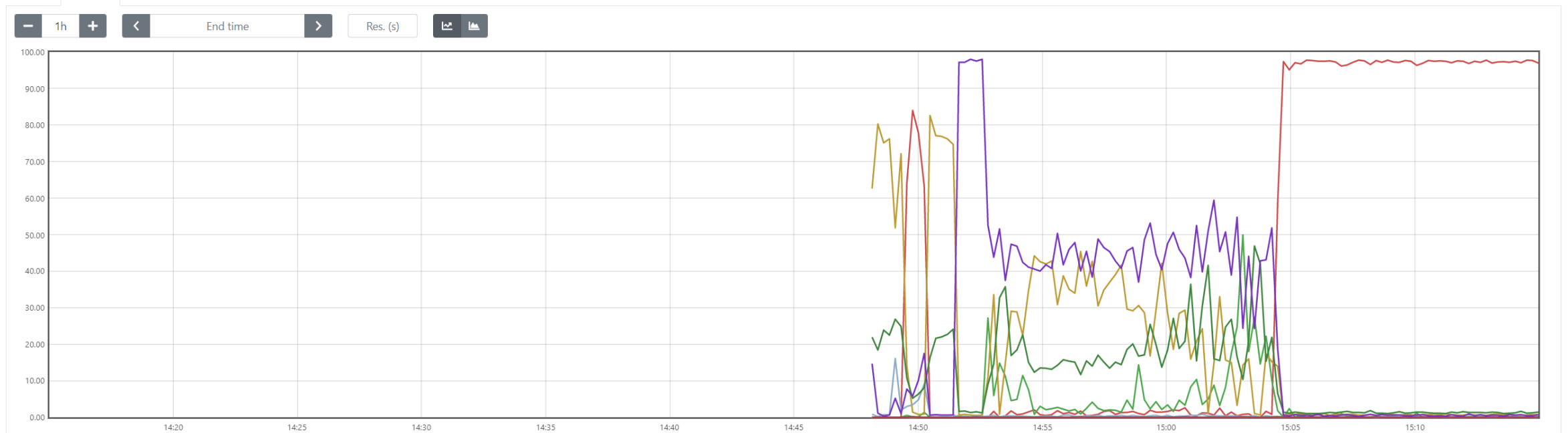
# NetApp monitoring



Enable query history  Use local time  Enable autocomplete

netdata:atnwnbjjj6:exfy8mivab:3vwdig5sez\_cpu\_cpu\_percentage\_average Execute

Table Graph Load time: 25ms Resolution: 14s Result series: 10



- netdata:atnwnbjjj6:exfy8mivab:3vwdig5sez\_cpu\_cpu\_percentage\_average (chart="cpu.cpu0", dimension="guest", family="utilization", instance="[fc21:5874:8e84:455b:6b4c:18a6:6d66:f691]:19999", job="netdata")
- netdata:atnwnbjjj6:exfy8mivab:3vwdig5sez\_cpu\_cpu\_percentage\_average (chart="cpu.cpu0", dimension="guest\_nice", family="utilization", instance="[fc21:5874:8e84:455b:6b4c:18a6:6d66:f691]:19999", job="netdata")
- netdata:atnwnbjjj6:exfy8mivab:3vwdig5sez\_cpu\_cpu\_percentage\_average (chart="cpu.cpu0", dimension="idle", family="utilization", instance="[fc21:5874:8e84:455b:6b4c:18a6:6d66:f691]:19999", job="netdata")
- netdata:atnwnbjjj6:exfy8mivab:3vwdig5sez\_cpu\_cpu\_percentage\_average (chart="cpu.cpu0", dimension="iowait", family="utilization", instance="[fc21:5874:8e84:455b:6b4c:18a6:6d66:f691]:19999", job="netdata")
- netdata:atnwnbjjj6:exfy8mivab:3vwdig5sez\_cpu\_cpu\_percentage\_average (chart="cpu.cpu0", dimension="irq", family="utilization", instance="[fc21:5874:8e84:455b:6b4c:18a6:6d66:f691]:19999", job="netdata")
- netdata:atnwnbjjj6:exfy8mivab:3vwdig5sez\_cpu\_cpu\_percentage\_average (chart="cpu.cpu0", dimension="nice", family="utilization", instance="[fc21:5874:8e84:455b:6b4c:18a6:6d66:f691]:19999", job="netdata")
- netdata:atnwnbjjj6:exfy8mivab:3vwdig5sez\_cpu\_cpu\_percentage\_average (chart="cpu.cpu0", dimension="softirq", family="utilization", instance="[fc21:5874:8e84:455b:6b4c:18a6:6d66:f691]:19999", job="netdata")
- netdata:atnwnbjjj6:exfy8mivab:3vwdig5sez\_cpu\_cpu\_percentage\_average (chart="cpu.cpu0", dimension="steal", family="utilization", instance="[fc21:5874:8e84:455b:6b4c:18a6:6d66:f691]:19999", job="netdata")
- netdata:atnwnbjjj6:exfy8mivab:3vwdig5sez\_cpu\_cpu\_percentage\_average (chart="cpu.cpu0", dimension="system", family="utilization", instance="[fc21:5874:8e84:455b:6b4c:18a6:6d66:f691]:19999", job="netdata")
- netdata:atnwnbjjj6:exfy8mivab:3vwdig5sez\_cpu\_cpu\_percentage\_average (chart="cpu.cpu0", dimension="user", family="utilization", instance="[fc21:5874:8e84:455b:6b4c:18a6:6d66:f691]:19999", job="netdata")

Click: select series, CTRL + click: toggle multiple series

# NetApp policy setting



- Service-related optimization policies are set for each instance
- CRUD operations can be done on these policies
- This operation is done in tandem with the NetApp toolkit
- Two types of policies:
  - **Operational policies**
  - **Application-specific policies**
- **Examples:**
  - If a vCPU exceeds 80% for more than 80 seconds then increase the number of vCPUs at the specific component by 1, else if a vCPU drops below 10% for more than 120 seconds then decrease the number of vCPUs at the specific component by 1, by retaining as minimum the default number of vCPUs previously defined (*Operational*)
  - If total latency (including network latency and processing latency at Function A) exceeds 250ms, then bypass Function A and send traffic directly from Function B to Function C through direct Web Sockets (*Application-specific*)

# Policy Definition through UI



Name \*

Expression

Select Function	Average	Select Component	phpDashboard
Select Metric	_apps_cpu_cpu_time__average (cpu time %)	Select Dimension	-- Select --
Select Operand	>	Threshold	0.7

Period (Seconds) \*

Inertia Time (Minutes) \*

Actions

Select Component	phpDashboard	Type	Scale Out
Workers	1		

+ -

# Slice Intent Creation



- Create Slice Intent upon NetApp instance creation
- Slice Intent is sent to the Slice Management Layer
- A slice response is sent back to the Application Orchestrator
  - **Successful slice creation, and resource allocation**
  - **Errors in resource allocation**
- The instantiation proceeds upon successful slice creation

[www.5g-iana.eu](http://www.5g-iana.eu)

**Thank you for your attention!**

**Any questions?**



<Presenter> - UBITECH



5G-IANA project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101016427.