



MATILDA 5G-ready Applications' Orchestration Ecosystem

Anastasios Zafeiropoulos (Ubitech) and Janez Sterle (ININ)

Join us for a live presentation of the main project
achievements at 17:30!

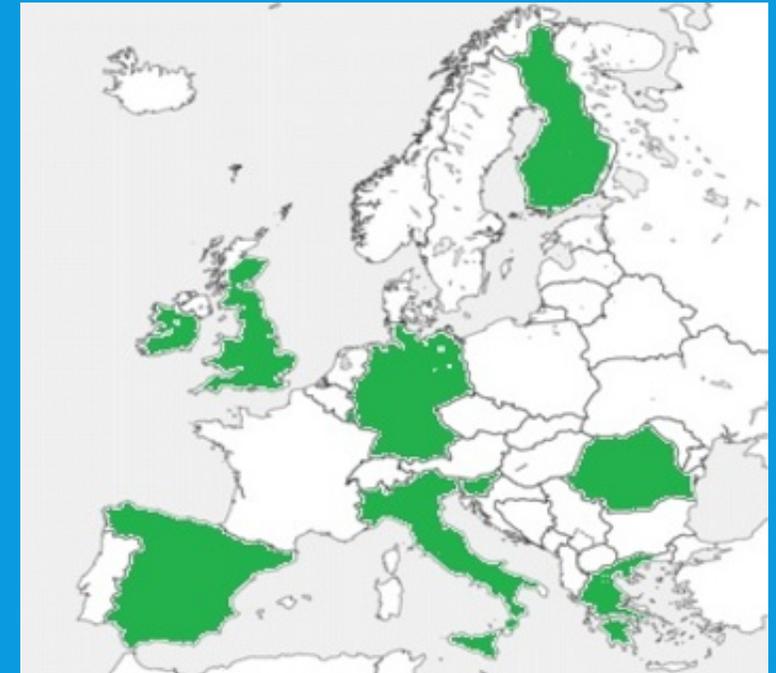
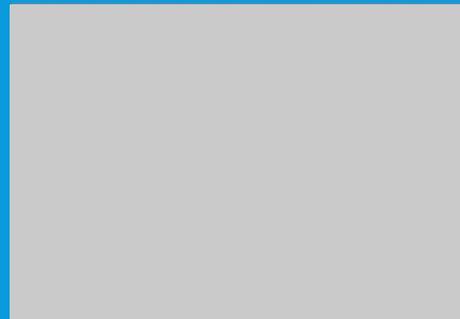
Mobile World Congress

25-28 February 2019, Barcelona

The MATILDA Consortium



MATILDA



The MATILDA Vision



MATILDA



Design and develop 5G-ready applications; applications able to take advantage of 5G programmable infrastructure.



Optimally manage and provide services/applications through a set of **intelligent orchestration** mechanisms.



Dynamically create and manage application-aware network slices able to serve 5G-ready application needs.



Separation of concerns among **vertical applications** and **network services** orchestration.

What is a 5G-ready application? Just of set of Cloud Native components?

How many levels of orchestration?

Slice aware Vertical Orchestrator?

○ Application Component

● IaaS 1 ● IaaS 2 ● IaaS 3

● Network Service ● VNF

MATILDA in a nutshell



MATILDA

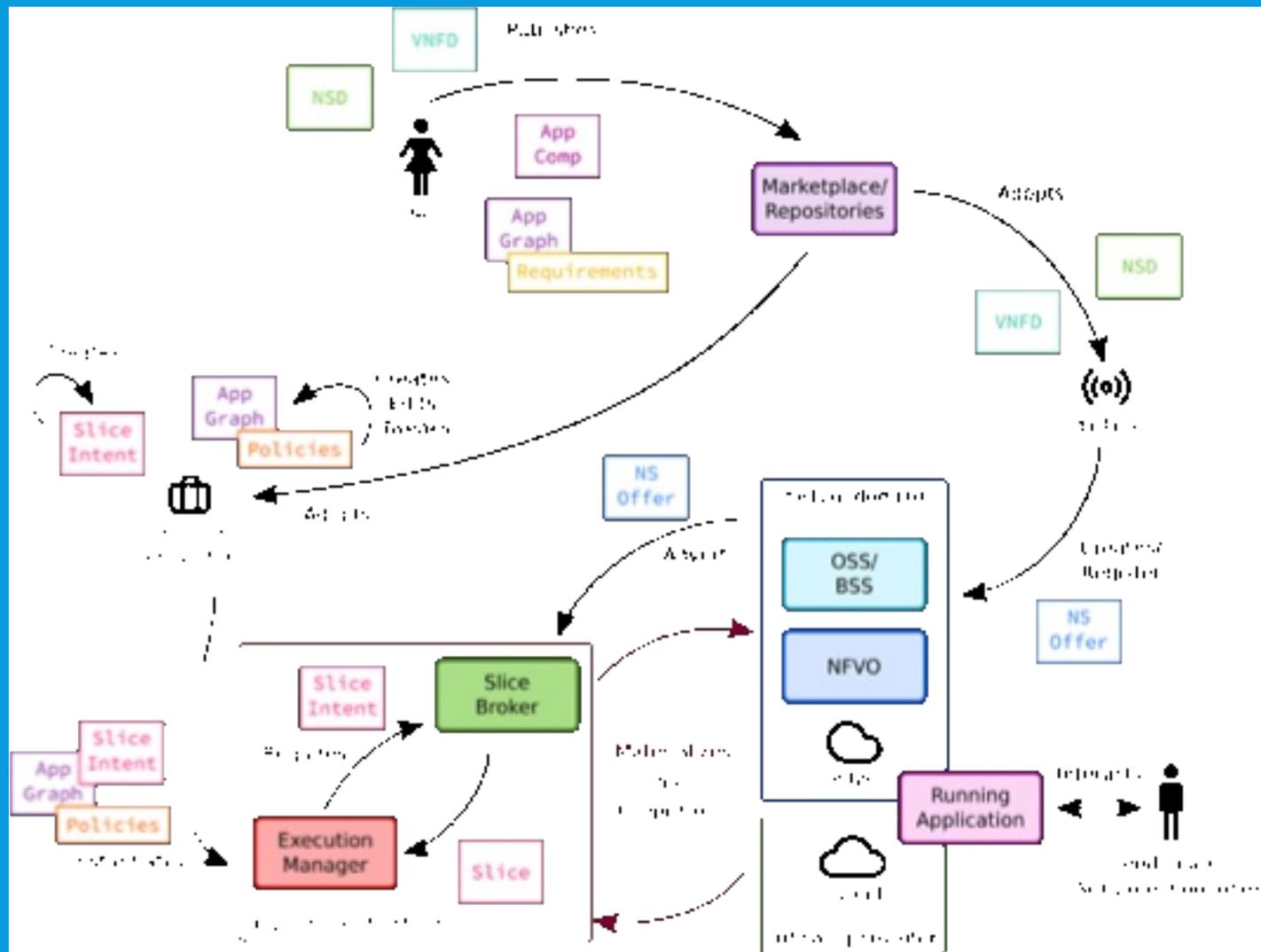
- Deployment of **cloud-native** components over **Telco Infrastructure**
- **Two main contributions:**
 - **Vertical Application Orchestration (VAO)**
 - Make use of advanced network programmability features
 - Slice Definition
 - Slice Reconfiguration
 - Policy Management
 - **Telco enhanced OSS**
 - Northbound APIs
 - Network Services Orchestration



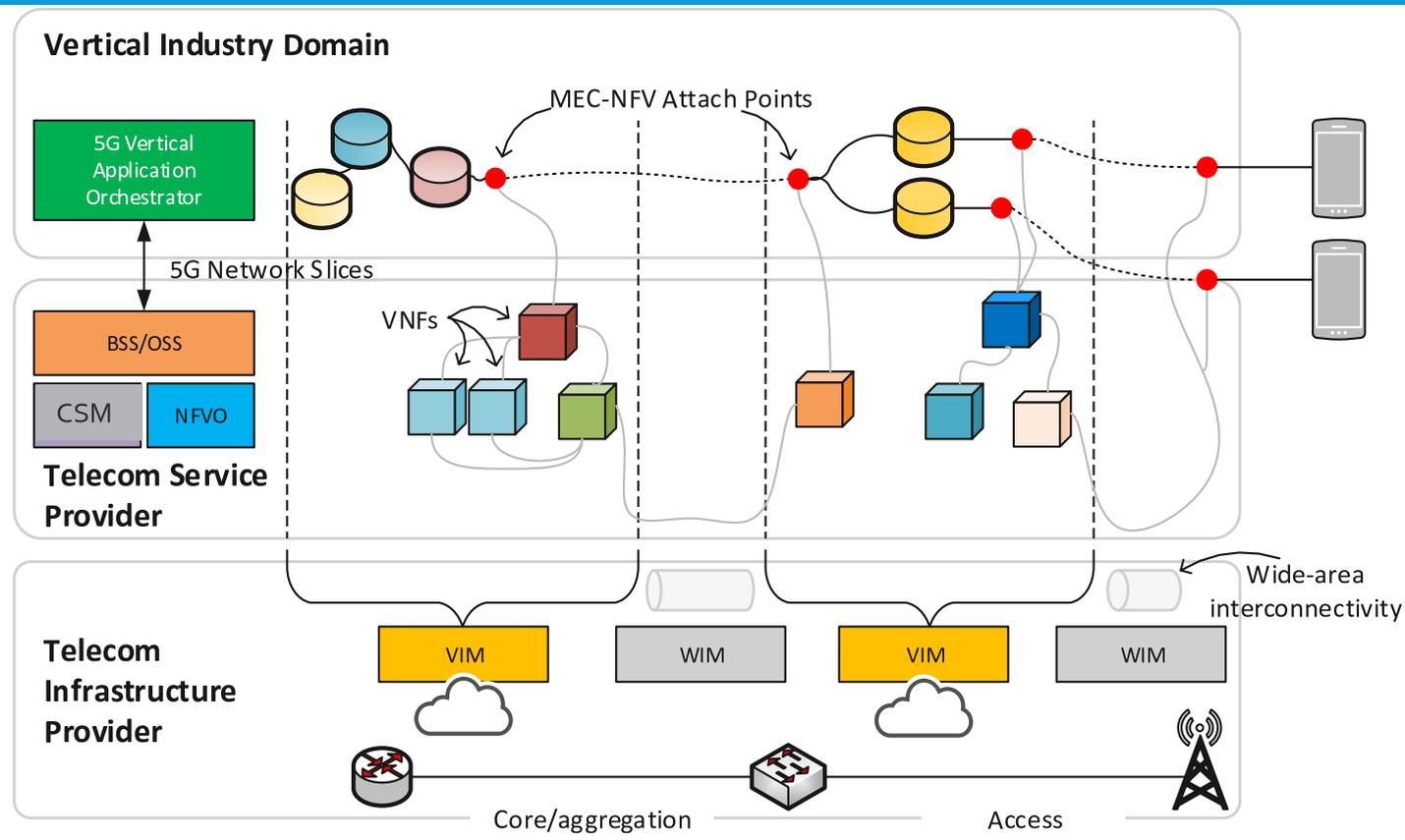
MATILDA end to end story



MATILDA



MATILDA architectural approach



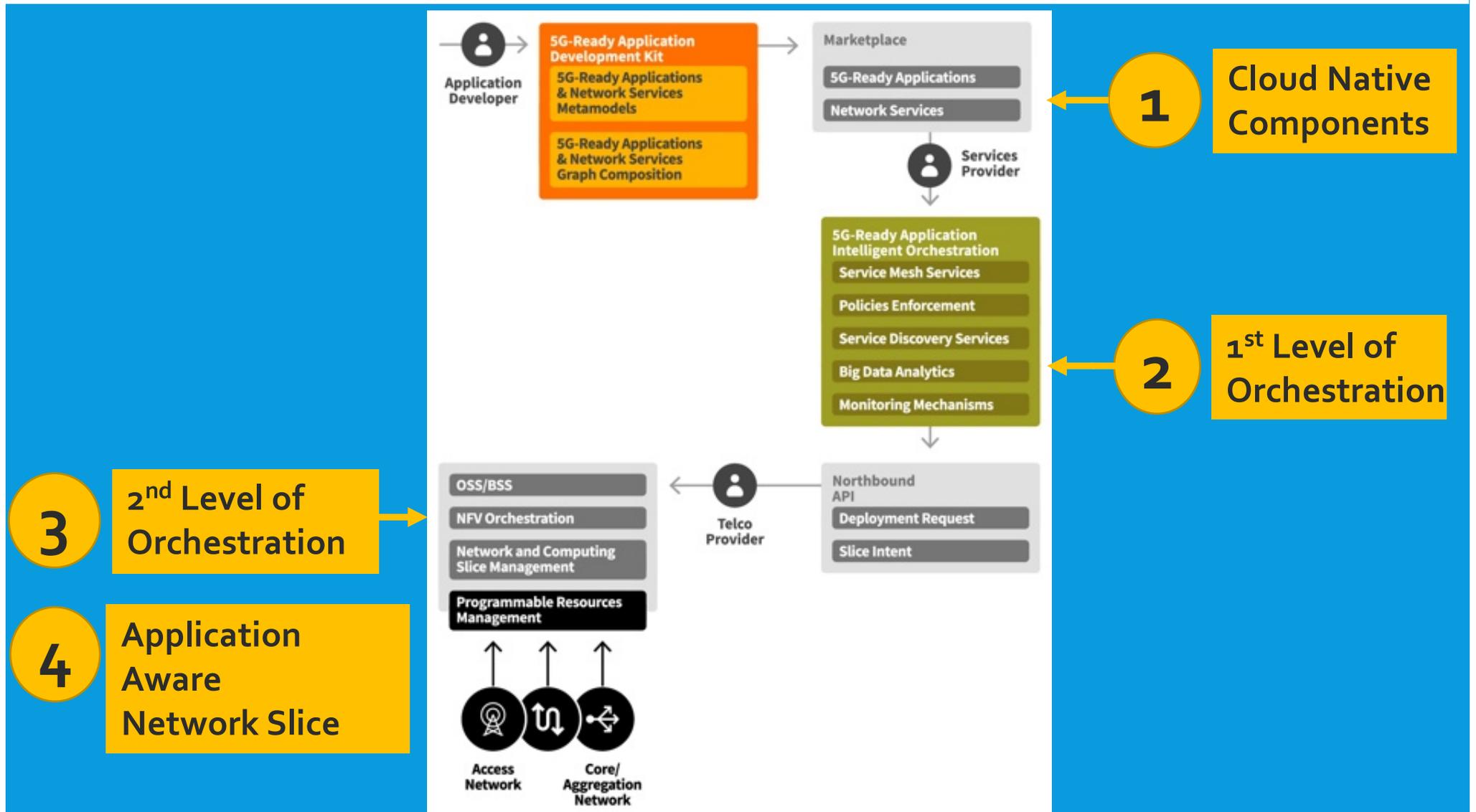
Legend

- CSM:** Computing Slice Manager
- VIM:** Virtual Infrastructure Manager
- VNF:** Virtual Network Function
- NFVO:** Network Functions Virtualization Orchestrator
- WIM:** Wide-area Infrastructure Manager
- MEC:** Multi-access Edge Computing
- BSS / OSS:** Business Support System / Operational Support System

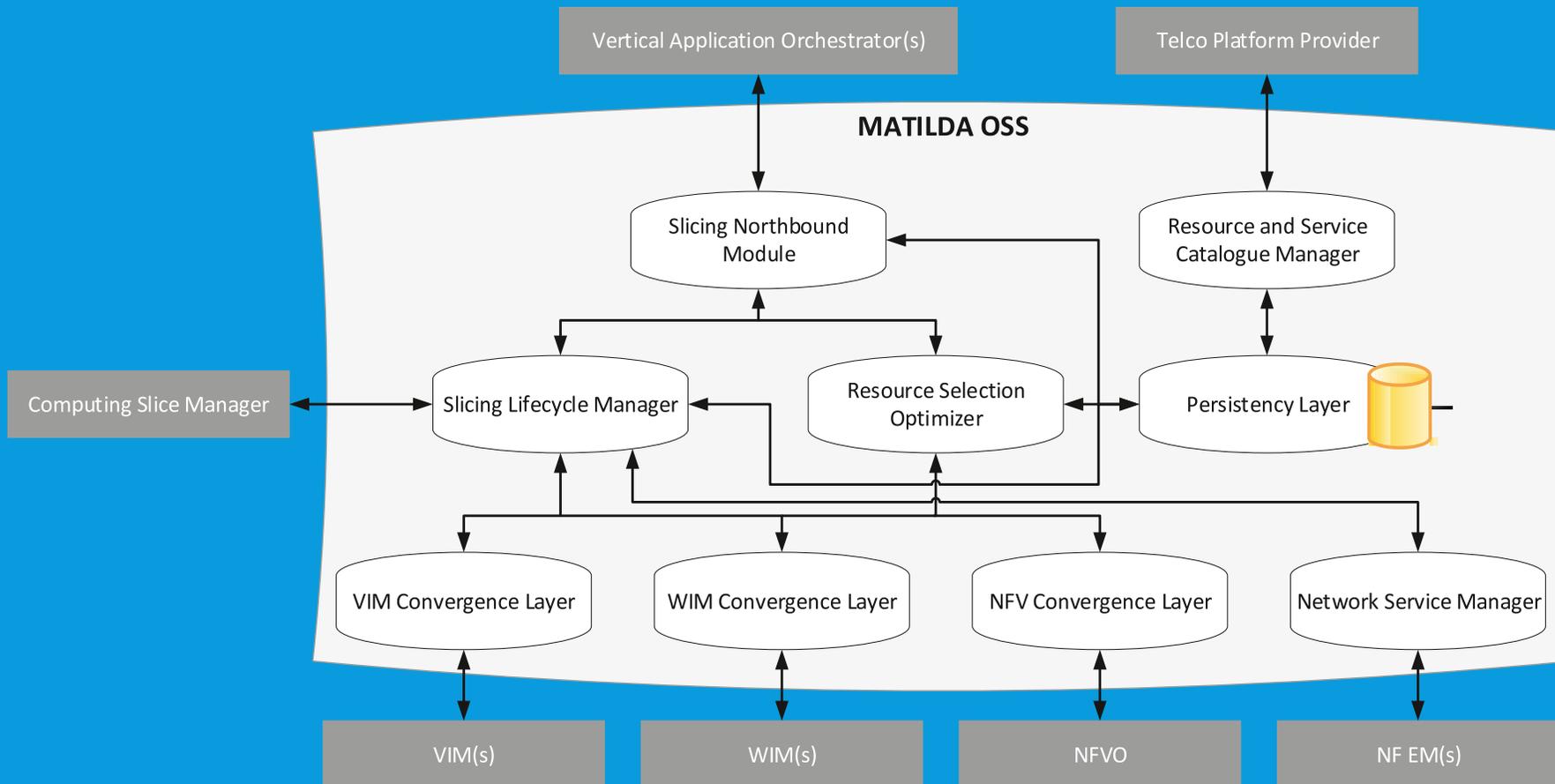
MEO: Multi-access Edge Orchestrator
 VIM: Virtual Infrastructure Manager
 VNF: Virtual Network Function
 BSS/OSS: Business Support System / Operational Support System

NFVO: Network Functions Virtualization Orchestrator
 WIM: Wide-area Infrastructure Manager
 MEC: Multi-access Edge Computing

MATILDA architectural flow



MATILDA OSS



MATILDA Dashboard



- DASHBOARD
- INSTANCES
- APPLICATIONS
- COMPONENTS
- SSH KEYS
- RESOURCES
- DETECTION RULES
- PLUGINS
- CONFIGURATION >

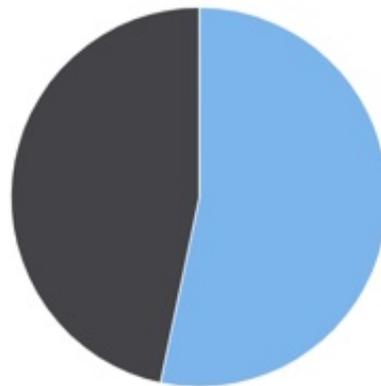
Dashboard

⚡ Application Instances
2

★ Applications
13

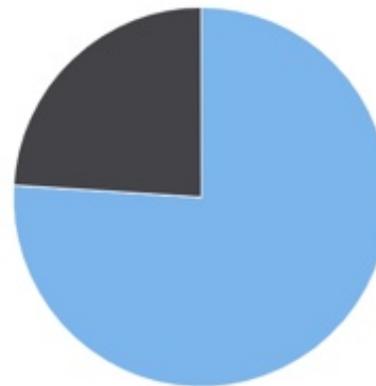
🔧 Components
31

CPU (Total: 30)



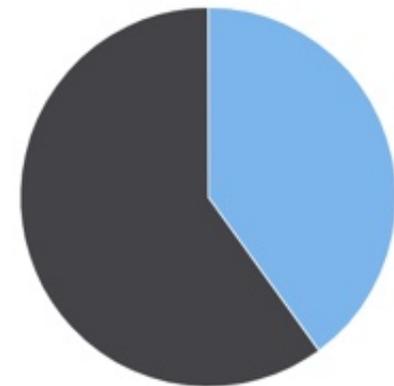
● Used vCPUs ● Remaining vCPUs

Memory (Total: 51200 MB)



● Used Memory ● Remaining Memory

Instances (Total: 20)



● Used Instances ● Remaining Instances

MATILDA Repositories



COMPONENTS

SSH KEYS

RESOURCES

DETECTION RULES

PLUGINS

CONFIGURATION >

Name

Filter

Reset

| Identifier | Name | Visibility | Date Created | |
|------------|-----------------|------------|---------------------------------------|---|
| nxZ6RyPgS5 | CustomMetricApp | Public | Wednesday, November 21, 2018 11:11 AM | Edit Delete ⋮ |
| 6fmnzF4kKd | DivFunc | Public | Wednesday, November 21, 2018 11:11 AM | ⋮ |
| JgT2p82zY3 | FaceDetector | Public | Wednesday, November 21, 2018 11:11 AM | ⋮ |
| 6eSZUugge7 | httpserver | Public | Wednesday, November 21, 2018 11:11 AM | ⋮ |
| FE1bWwiSSF | HttpStressApp | Public | Wednesday, November 21, 2018 11:11 AM | ⋮ |

MATILDA Application Composer



MATILDA

PPDRDemo

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

Search a component

ppdr

Generic info:
ID: iHLcgCz2sQ
Name: PPDRPhpDashboard

PPDRPhpDashboard262

Required Interfaces:

- Interface: SambaInterface1
- Interface: sqlInterface

Exposed Interfaces:

- Interface: PhpDashboardInterface

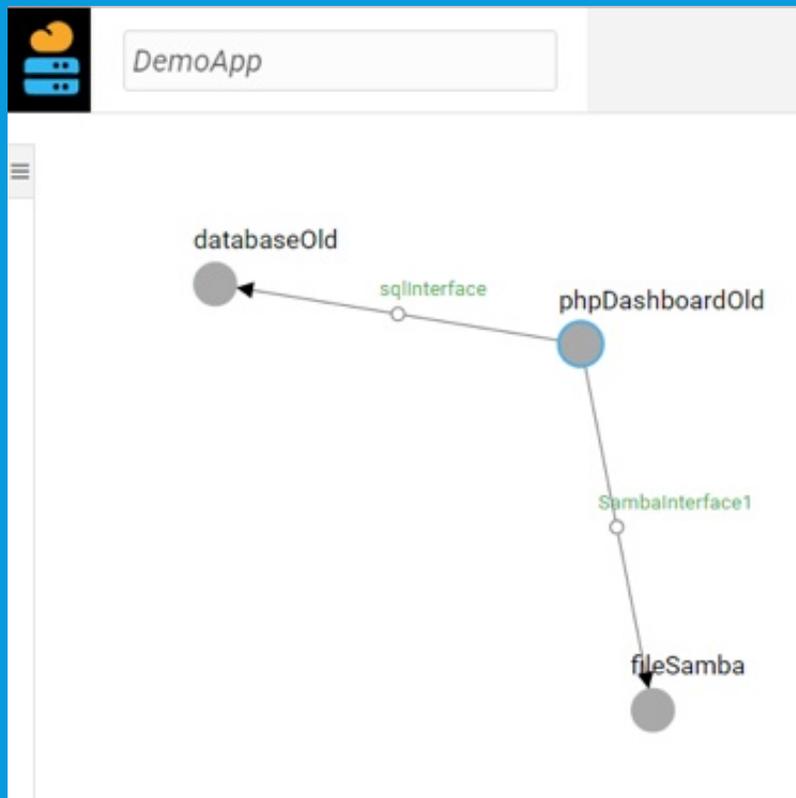
Save

```
graph TD; A((PPDRPhpDashboard262)) ---|sqlInterface| B((PPDRDatabase241)); A ---|SambaInterface1| C((PPDRSamba253));
```

Constraints Definition



MATILDA



Set the Constraints of "SambaInterface1" graph link

Maximum Delay (ms)

Value:

Constraint Type: Soft Hard

Maximum Jitter (ms)

Value:

Constraint Type: Soft Hard

Maximum Packet Loss (%)

Value:

Constraint Type: Soft Hard

Minimum Throughput (Kbps)

Throughput:

Constraint Type: Soft Hard

Slice Intent Creation



MATILDA

Maximum Delay (ms)

Value:

Constraint Type: Soft Hard

Maximum Jitter (ms)

Value:

Constraint Type: Soft Hard

Maximum Packet Loss (%)

Value:

Constraint Type: Soft Hard

Minimum Throughput (Kbps)

Throughput:

Constraint Type: Soft Hard

{

```
"applicationInstanceID": "580",
"name": "OSSScenario",
"callbackURL": "http://localhost:8080/api/v1/callback/slice/580",
"authenticationDetails": {
  "clientToken": "!telcoprovider!",
  "clientKey": "telcoprovider"
},
"componentNodeInstances": [{
  "componentNodeInstanceID": "581",
  "componentNodeInstanceName": "TestCaseMariaDB"
}, {
  "componentNodeInstanceID": "587",
  "componentNodeInstanceName": "TestCasePhpMyAdmin"
}],
"constraints": [{
  "constraintID": "591",
  "interfaceInstanceID": "590",
  "qi": "10",
  "radioServiceType": "1",
  "resourceType": "DELAY_CRITICAL_GBR",
  "allocationRetentionPriorityProfile": 1,
  "minimumGuaranteedBandwidth": 120.0,
  "maximumRequiredBandwidth": 200.0,
  "constraintUnit": "kbps",
  "category": "ACCESS",
  "type": "HARD"
}]
```

Automated Deployment



MATILDA

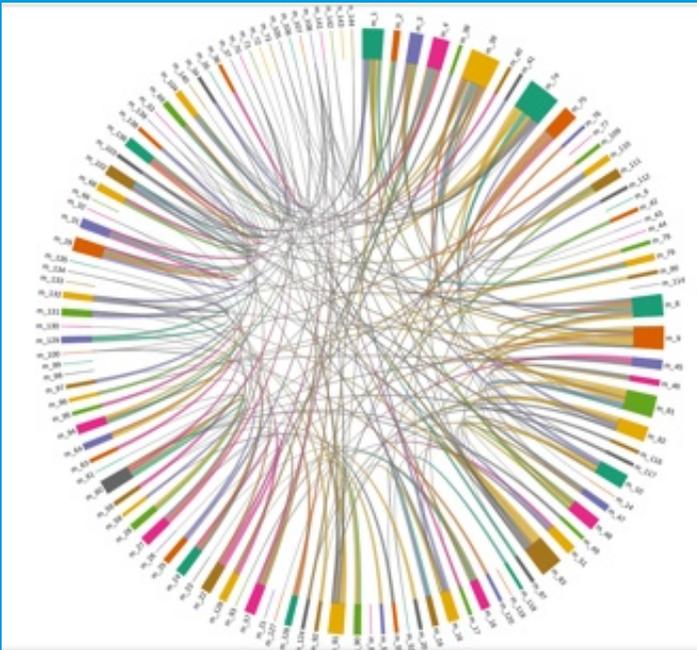
The screenshot displays the MATILDA web interface for a component named 'phpDashboard'. The interface is divided into several sections:

- Header:** Shows the component name 'MWCDemo.PPDR' and the MATILDA logo.
- Left Panel:**
 - phpDashboard:** Contains links for 'Generic info', 'Go to Prometheus', 'Detection Rules', and 'Amend Rulesets'. There is a 'Select...' dropdown and a 'Change Rules' button.
 - Workers:** Lists a single worker: '(ID: VYTafaGm4Q) - phpDashboardWorker1'. It provides details: 'Provider: UBIDELLMatilda', 'IP: 10.6.0.12 (matildaNet)', and 'status: Component is healthy, up and running'. A 'View Graphs' link is also present.
- Center Panel:** A dependency graph showing 'phpDashboard (1 WRKs)' at the top, connected to 'database', 'phpDashboardLB', and 'Samba' below.
- Bottom Right Panel:** A 'Logs' window showing the deployment process:
 - 19-02-2019 13:59:09 [LOADING] : phpDashboardWorker1 - A new worker is going to be created!
 - 19-02-2019 13:59:59 [INFO] : phpDashboardWorker1 - Agent dependencies fulfilled
 - 19-02-2019 14:00:57 [INFO] : phpDashboardWorker1 - The component is waiting for it's dependencies
 - 19-02-2019 14:01:19 [SUCCESS] : phpDashboardWorker1 - Component is healthy, up and running

Runtime Policies and Profiling



MATILDA



Elasticity Policy | Create

Elasticity Policy Management for "MWC" Application Instance

Name *

Scaling Policy

Expression

Select Function: Average

Select Metric: _ipv4_packets_packets_persec_average (packets/s)

Select Operand: >

Select Component: PPDRPhpDashboard262

Select Dimension: received

Threshold: 400

Period (Seconds) *: 20

Inertia Time (Minutes) *: 1

Actions

Select Component: PPDRPhpDashboard262

Type: Scale Out

Workers: 1

+ -



MATILDA

5G PPDR Use Case

CURRENT PPDR ECOSYSTEM



PPDR OPERATIONS

Every day
operations



Extreme
situations



NEXTGEN (5G) PPDR SYSTEMS



Flexibility in deployment and system use

- Strategic level
- Tactical level
- User/mission level

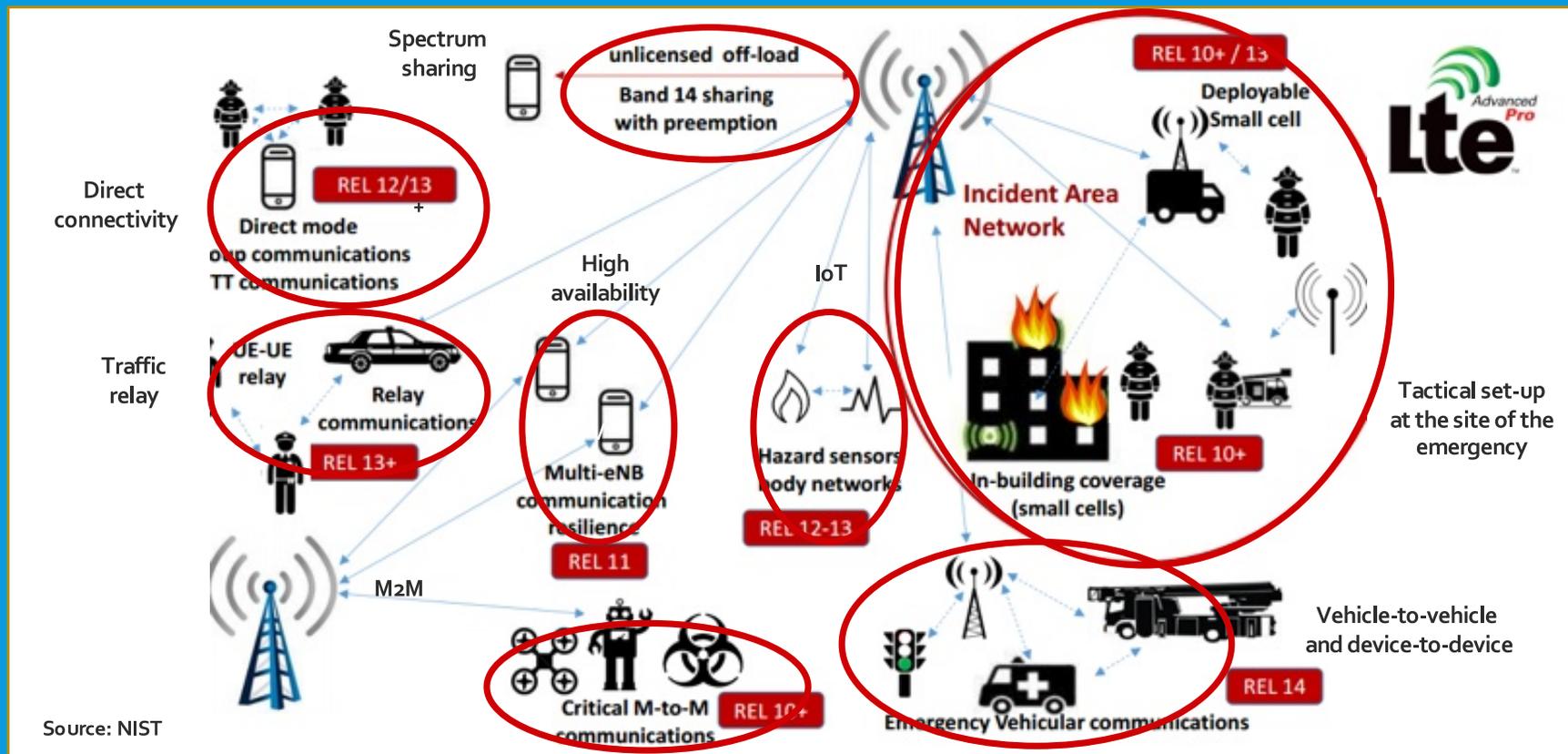


User/mission level

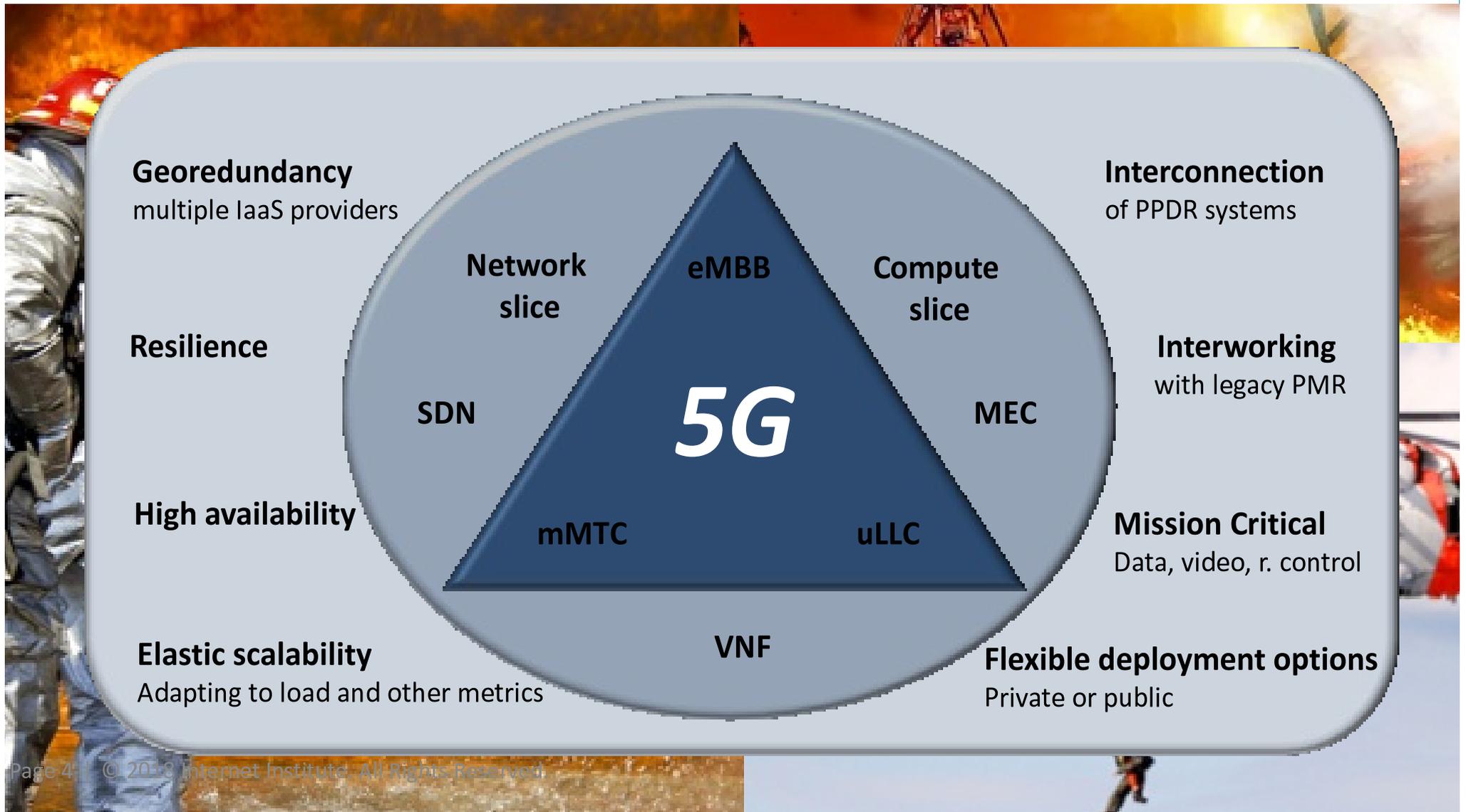


Vertical and horizontal scaling of the system and services components is essential!

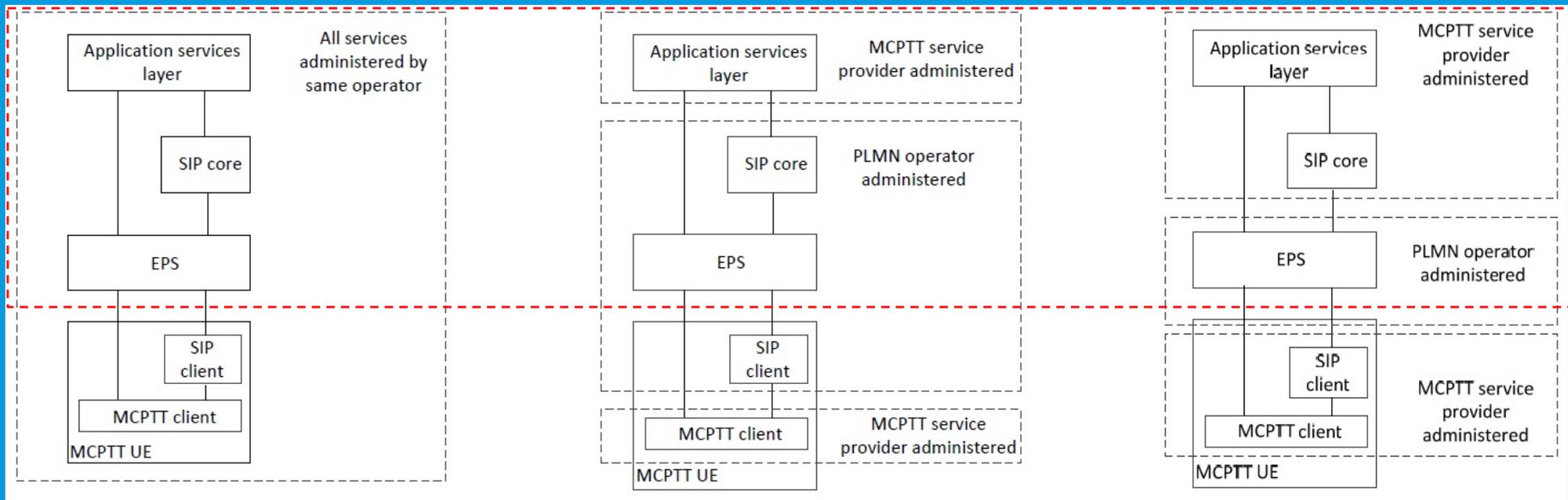
STANDARDIZATION – 3GPP R14



5G FOR PPDR

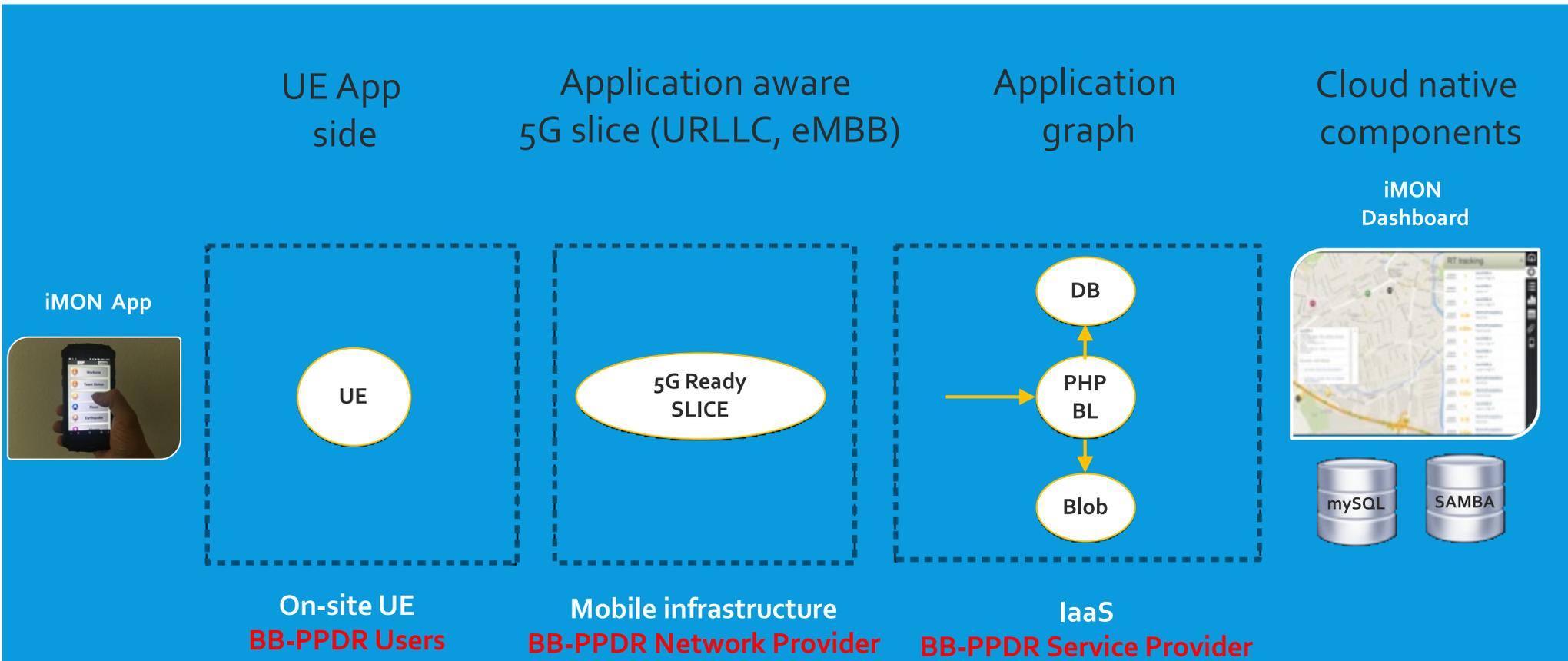


3GPP MC-PTT SERVICES DEPLOYMENT MODELS



5G PPDR challenge targeted with MATILDA: **High availability, reliability and scalability of network and services infrastructure ("MC PPT", "MC Video" in "MC Data") in extreme situations!**

IMON – 5G-READY APPLICATION



Automated deployment, slice creation, vertical and horizontal components scaling,...



MATILDA

Thank you!