Mobile Cloud Networking (MCN): Motivation, Vision, and Challenges

Thomas Michael Bohnert (TMB, Technical Coordinator)
Giovanni Toffetti
MCN Project

- Nov. 2012 – Oct. 2015
- 15.7 M

SAP AG, Germany
FRANCE TELECOM CA, France
TELECOM ITALIA S.p.A, Italy
BRITISH TELECOMMUNICATIONS PLC, United Kingdom
PORTUGAL TELECOM INOVAÇÃO SA, Portugal
NEC EUROPE Laboratories, Germany
INTEL PERFORMANCE LEARNING SOLUTIONS LIMITED, Ireland
ITALTEL S.P.A., Italy
CLOUDSIGMA AG, Switzerland
NEXTWORKS, Italy
Soft Telecom, Spain
ONESOURCE, CONSULTORIA INFORMÁTICA, LDA., Portugal
UNIVERSITEIT TWENTE, Netherlands
TECHNISCHE UNIVERSITÄT BERLIN, Germany
INOV INESC INOVAÇÃO, INSTITUTO DE NOVAS TECNOLOGIAS TECNOLÓGICAS, Portugal
UNIVERSITÄT BERN, Switzerland
Zürcher Hochschule für Angewandte Wissenschaften (ZHAW), Switzerland
FRAUNHOFER-GESELLSCHAFT ZUR FOERDERUNG DER ANGEWANDTEN FORSCHUNG E.V, Germany
GSMA, The Mobile Economy 2013

**Mobile Telco Industry: Status Quo**

**GLOBAL ARPU**
US$ Per Month

- 2008: 19.3
- 2009: 17.4
- 2010: 15.7
- 2011: 15.0
- 2012: 14.1

-7.6%

**Data volume/rate, CAPEX + OPEX**

**Average Revenue Per User (ARPU)**

**CAPITAL EXPENDITURE ACROSS A SELECTION OF INDUSTRIES**
Capex as a % of Revenues, 2012

- Electricity: 15.2%
- Marine: 15.2%
- Mobile: 15.0%
- Metals & Mining: 11.3%
- Oil & Gas: 11.2%
- Steel: 8.3%
- Internet: 8.0%
- Hotels: 6.6%
- Automobiles: 4.6%
- Pharma: 4.2%
- Technology: 3.7%
- Software: 3.1%
- Medical Equipment: 0.2%
Options I – Same Service, at lower CAPEX and OPEX

How low is possible?
Options II – Value Added Services

Requires new revenue stream!

Unit

Time

CAPEX & OPEX

ARPU
Optimum – Value Added Services plus lower CAPEX and OPEX

Ideal situation
Motivations

From traditional Mobile Network Operators…

… towards Mobile Cloud providers

MNOs concepts today

- Traditional connectivity & voice business
- Few value-added services only, trend towards over-the-top (OTT) provided by competitors
- Infrastructure and Platform sharing (MVNOs)
  - Infrastructures, networks, and platforms ...
    - Pre-sized, Pre-provisioned, Pre-customized, Huge CAPEX

MNO concepts tomorrow

- Adopt IaaS and PaaS for network functions
  - Resource pooling, On-demand, Elastic, Pay-as-you-go
- **Reduce costs**, move from CAPEX to OPEX
  - Exploit cloud principles for network operations
  - New approach to MVNO, new customers for traditional mobile telco business

**New Business**: Mobile Network + Computing + Storage

- End-to-end platform for novel applications
- Eco-system, developers, new revenue stream
Top-Level Directions

Scenario 1: Using the Cloud

Scenario 2: Extending the Cloud
Cloud-based mobile networks: the concept

Moving cloud computing beyond datacenters…

... towards the mobile end-users.

- On-demand and self-service
- Elastic
- Multi-tenant
- Pay-as-you-go

Mobile Connectivity
Decentralized Computing
Smart Storage
offered as a single end-to-end service
MCN Enabling Assumptions

Micro-Data Centre Deployments
- Local deployments limited resources e.g. suburban, rural areas.
- Workloads can call on additional resource from a macro-data centre or a closely micro-data centre.

Macro-Data Centre Deployments
- Centralised deployment with access to cheap resources (power) e.g. metropolitan areas.
- Workloads can be migrated near to user on macro-data centre.
From ....

- System is contained to local resources
- Scaling is limited by local resources
  - Difficult beyond - requires rearchitecting
- Many *existing* systems are built like this
... to a cloud-native design

- System is not contained to local resources
- Scaling is adding as many resources/nodes that are available
- Elasticity enabled grow and shrink as needed
- Existing systems are **not built for this**
- Requires **additional** orchestration and management
Goals of MCN Architecture

- Modularity, reusability
- Creation of composed (end-to-end) services
- Adhere to the NIST cloud computing definition
- Enable cloudification of services e.g. EPC
  - keep functional arch, adapt software arch
- Common framework and lifecycle to design services that accommodates all identified scenarios
- No technology specific dependencies
- Leverage & influence suitable/relevant standards to ensure interoperability and integration
Terminology

- **Service**
  - *E.g. CDNaaS*

- **Service Instance**
  - *E.g. CDN service instance for customer X*

- **Service Instance Components (SIC)**
  - *E.g. MME or DSS cache*

- **Resources** (Physical/Virtual) build services
<table>
<thead>
<tr>
<th>Service Category</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Atomic</td>
<td>An indivisible service that executes a particular singular business or technical function and generally implemented using a service provider’s resources.</td>
</tr>
<tr>
<td>Composed</td>
<td>A service that is created by combining two of more services, including atomic or even other composed services.</td>
</tr>
<tr>
<td>Support</td>
<td>Platform(^{13}) services of MCN that provide targeted, specific functionality for use by any service.</td>
</tr>
<tr>
<td>MCN</td>
<td>A service offering implemented within MCN whose implementation consists of a service manager, one or more service orchestrators and it’s the service functionality, including other deployables such as VMs, code bundles etc.</td>
</tr>
</tbody>
</table>
Lifecycle of a MCN Service

MCN Services
- RANaaS
- EPCaaS
- IMSaaS
- DSSaaS
- CDN/ICNaaS

Support Services
- LBaaS
- MaaS
- RCBaaS
- DNSaaS
- MOBaaS
- SLaaS
- AaaS
- DBaaS
- AAAaaS

Atomic Services
- IaaS
  - Compute
  - Storage
  - Network

Business
- Design
- Agreement

Technical
- Design
- Implementation
- Deployment
- Provisioning
- Runtime & Mgt
- Termination
MCN Services

- **RANaaS**, Wireless-as-a-Service, enabled by RAN virtualisation, that is Remote Radio Head (HW) / Base Band Unit (SW) separation with Base Band Units deployed on-demand on elastic IaaS running on top of micro data centres close to antennas.

- **EPCaaS**, Evolved Packet Core as a Service (EPCaaS) that is on-demand deployment of distributed EPC instances on top of elastic IaaS on micro and/or macro data centres based on individual needs.

- **IMSaaS**, that is on-demand deployment of IMS (IP-Multimedia-Subsystem) instances for complementing voice/video services on top of elastic IaaS on micro- and macro-data centres and based on individual needs.

- On-demand and elastic content / storage / application distribution services, on top of IaaS on micro and macro data centres exploiting cloud-storage services (Follow-Me cloud).

- **End-to-End MCN Service Orchestration** (infrastructure, platform, services).

- Mobile Cloud Networking AAA, SLA, Monitoring, Rating, and Charging compliant with XaaS.
MCN Key Arch Elements

**Service Manager**
- Provides an external interface to the user
- Business dimension: encodes agreements
- Technical dimension: Management Service Orchestrators of a particular tenant

**Service Orchestrator**
- Oversees E2E orchestration of a service instance
- Domain specific component
- Manages service instance
- 'Runtime & Management' step of the Service Lifecycle
- One SO is instantiated per each tenant within the domain
- SO is associated with a Service Manager
- Monitors application specific metrics and scales (SOE/SOD)

**CloudController**
- Supports the deployment, provisioning, and disposal of services
- Access to atomic services
- Access to support services
- Configures atomic services (IaaS)
Service Manager Internals

- Main entry point so service management for EEU
- Maintains list of services offered SM
- Overall management of SM’s SO’s
Service Orchestrator Internals

• enforces decisions towards the CC
• interacts with CC entities

• Graph of required services and resources for service instance
Orchestration graphs
MCN Key Arch Elements Overview

All are used throughout MCN
How is a MCN service instance deployed?

Scenario
- 4 service providers (C1-C4)
- 3 services orchestrated - **RAN**, **Core**, **CDN**
- 1 value added E2E service offered to the enterprise end user
- Both public and private cloud resources

Scenario Assumption
- Service designed and implemented
How is a MCN service instance deployed (1)?

EEU requests a service instance

Providers, Services and CloudControllers
How is a MCN service instance deployed (2)?

**Deployment phase**

Service managers inside each service provider
How is a MCN service instance deployed (3)?

**Deployment phase**
Service Orchestrator created to oversee instance creation
How is a MCN service instance deployed (4)?

Deployment phase
Service Orchestrator requests necessary services creation
How is a MCN service instance deployed (5)?

Deployment phase
Each required service provider’s service manager creates a service orchestrator
How is a MCN service instance deployed (6)?

Deployment phase
Service orchestrators that require services from the CloudController requests them.
How is a MCN service instance deployed and provisioned?

Where are we?
- Deployment phase is completed
- Eventually all services are created
- Not configured however

- Provisioning phase begins…
How is a MCN service instance provisioned?

**Provision phase**
The SO has access to all other service instance management endpoints. Configuration information is supplied to these.
How is a MCN service instance provisioned?

Provision phase
Service orchestrators *may* pass on configuration to CloudController.
How is a MCN service instance created?

**Where are we?**

- Ready for service
- Deployment & provisioning phase completed
- Service instance management interfaces are available to the EEU
  - degree of configurability is dependent on service provider
- EUU can use & further customise the service instance
- SO of all service instances manage runtime
  - SOD & SOE
Short demo

- Orchestration Video
#!/usr/bin/python
import mcn.sdk
[[logic for SO]]

SO logic:
- uses MCN CC SDK to manage VNFs
- uses Heat templates for the deployment on Openstack
Deploying a service instance with MCN

1. **End User** interacts with **Service Manager**
2. **End User** provides **Occi**
3. **Service Manager** processes **Occi**
4. **Service Manager** requests from **Cloud Controller**
5. **Cloud Controller** deploys a **SO bundle**
6. **SO bundle** is deployed to **OpenShift**
7. **OpenShift** deploys the **SO bundle**

**SO bundle** includes:
- **Python Runtime Engine**
- **MCN CC SDK**
- **Container**
- **OpenStack**
- **Heat**
Deploying a service instance with MCN

Dev team
Vendor

End User

**SO bundle**

1. Dev team or Vendor submits a service instance request.
2. Service Manager receives the request and checks the availability of the service instance.
3. Service Manager triggers the deployment process.
4. OCCI (Open Cloud Interface) is used to manage the service instance.
5. The service instance is deployed in a container.
6. The container is then deployed on the Cloud Controller.
7. The Cloud Controller manages the deployment of the service instance, which includes DSSaaS, DNSaaS, and MaaS.
8. The deployment process is monitored and controlled using tools like OpenStack and Heat.

© 2012-2015 MCN. All rights reserved. / Page 42
Architecture
Auto-scaling

User > SM > SO
- deploy
- create
- deploy svc
- deploy VM
- create user
- credentials

CC
- register alarm on 1min_avg(RT)>X
- set agent credentials
- publish data
- check alarms
- decision
- scaling update
- ...

Openstack

VM

Monasca
MCN Experience

• Microservices principles:
  – loose coupled services with clear boundaries defined by interfaces
  – microservice independence:
    • performance and failure isolation
    • delegation to a single team
    • own release cycle
    • best technology for the task
    • decentralized data management
  – infrastructural automation
  – design for failure

• MCN is more about service composition

• Cloud-native services vs.
  – services depending on specific physical resources (sw replication not enough)
  – services with established communication channels

• Performance (latency) issues in RANaaS
THANK YOU!