Vodafone Network Virtualisation Architecture for LSO, SDN and NFV

Kev Brackpool
SDN/NFV Architect
Vodafone Group
Virtualisation – Key Themes

**SDN**
- Introduction of **SDN** for Virtual, Packet and Optical networks. Leveraging existing assets and enabling **Dynamic Service Management** and optimisation

**Network Virtualisation (NFV)**
- **Virtualisation** of our network functions portfolio, worldwide (e.g., MME, HSS, S/P-GW, SGSN, GGSN, MGW, BNG, Firewalls, Gi-LAN, CGNAT)

**E2E Service Orchestration & Big-data Analytics**
- **End-to-End automated** service delivery with full collaboration of all components (e.g., SDN Controllers, CMS, VNFM (new EMSs), evolved OSS/BSS)
- Deliver **superior customer experience** through big-data and analytics based pre-emptive service assurance and fault management
Virtualisation – Key Principles

Operations-Driven

- Full collaboration with Operations Teams
- New Single E2E Target Architecture Worldwide
- End-to-End support is a fundamental part of Architecture

Openness

- Open Interfaces. Defined and published by Industry Body or Open Source project
- Partners are jointly implementing interfaces on their selected products
- Vodafone will contribute open interfaces back to communities

Multivendor

- Multivendor cooperation is essential
- Demonstrated by consortium environment in Vodafone Labs
- Collaboration on implementing open interfaces
Service Delivery Stack

A Customer can request any Vodafone product from a single portal, use a single CRM, and receive a single bill.

A product is built from service chains which can use resources anywhere across Vodafone.

Resource domains operate autonomously with local resilience, capacity management, analysis etc.

Fine grain technology management for efficient utilization and fast close loop control.

Infrastructure resources available to underpin the services.
Globalising Services

• Today interconnected autonomous local market networks (OpCos).
  – Technically aligned for consistency and reliability
  – Independent service delivery and operations stacks
• End-to-End orchestration will further improve this

1. Use standardised APIs to split the customer oriented systems from the local operational Domains

• 2. Service management instances interact directly with relevant Domains to deliver end-to-end service.
  – Service management layer manages the lifecycle of end-to-end services, coordinating local domains as necessary
  – Domains manage local resources, automated delivery and fulfilment procedures, allocating resources, resource restoration and capacity planning
Architectural Overview

Customer & End-to-end Service Management

End-to-End Service Management

Service Support Systems

Service Orchestration

Resource Management

Domain Support Systems

Domain Orchestration

Resource and Technology Management

Technology Management

Application & Transmission

Customer Management

Portal/API

CRM

Billing

SDN SBI

NfV

Vi

VNFm

CPE SDN Controller

WAN SDN Controller

VIM (Cloud Manager)

Specific/Generic

Hosting Centre

Apps./VNFs

NFVI

Physical LAN Infrastructure

Resource management

Bulk data collection (Stats, events etc.)

Standardised API
Orchestration Framework

- **Automated** management of End-to-End Customer Services
- A **single** global orchestration framework for all areas (e.g., Network, Applications, IT)
- Delivers **superior customer experience** via analytics based pre-emptive Service Assurance and fault management
- **What-if** simulation according to marketing traffic forecast

---

**E2E Service Orchestration**

- **End-to-End Service Orchestrator (SO)**
  - The SO understands the full chain of network functions and connectivity for an end-to-end service.
  - SO decomposes requests into specific requests for DO. SO/SS creates an assurance view providing SLA reports and notifications during the entire service life cycle.

- **Domain Orchestrator (DO)**
  - DO monitors resources utilization at Application (by EMS/VNFm), Infrastructure (by VIM) and connectivity level (SDN).
  - DO decomposes SO requests into specific requests for SDN controllers, Cloud Managers, Application configuration etc.
Service Use Case Example

Customer wants to purchase connectivity between two of their sites and also connection to the internet.

The E2E service comprises several distinct parts. In this example:-

- The private L3 VPN comprised of 3 separate VF network domains: Italy, UK and Group Operations
- Two CPE located on separate sites, each requiring a dedicated Access Link/Tail Circuit
- The virtualised Firewall & NAT application (which requires a VM/container on a physical server)
  - and connected to the L3 VPN instance and also the internet

The FW+NAT could be hosted in any suitable location as the customer did not specify any requirements here (assume a Group Operations hosting site since these all have direct internet access)

Possible Realisation

User Service Endpoint

CPE Tail Circuit

Site A, London-UK, 7Gbps pk, Service profile Gold, subnet 10.1/16

UK L3 IP VPN

UK Domain

GO L3 IP VPN

Group Operations Domain

IT L3 IP VPN

IT Domain

Internet

FW+NAT NFV App

ENNI

Site B, Milan-IT, 3Gbps pk, Service profile Gold, subnet 10.0.1/24

User Service Endpoint

Tail Circuit

CPE
Management Systems Roles

Customer Management
- CRM
- Billing

Resource management

Bulk data collection (Stats, events etc.)

End-to-End Service Management

Service Support Systems

Service Orchestration

Resource management

Service data collection (Stats, events etc.)

Domain Support Systems

Domain Orchestration

End-to-End Service realisation

GO L3 IP VPN

GP Ops Domain

UK Domain

IT Domain

Tail Circuit

CPE

User Service Endpoint

WAN SDN Controller

VIM (Cloud Manager)

Specific/Generic VNFM

Connectivity Manager

Bulk data collection

User Service Endpoint

UK L3 IP VPN

IT L3 IP VPN

Tail Circuit

CPE

Resource management
Service Use Case Example

Workflow Steps (simplified)

• Customers use dedicated portal to describe what they want (friendly GUI and/or API depending on needs)
• Customer Management systems pass end-to-end customer service request to Service Orchestrator.
  – Uses TMF compliant APIs to fully describe the entire customer service request.
• Service Orchestrator uses published domain capabilities to decompose into specific requests for each domain.
  – Uses TMF compliant APIs to describe the services required from each relevant domain, many customer services will continue to be delivered by a single domain.
• The Service Orchestrator needs to coordinate inter-domain connectivity service ‘stitching’.
• Domain Orchestrator uses local knowledge to decompose into specific request for each control entity (e.g. cloud manager, SDN controller, VNFM/VNF etc.)
  – APIs dependent on control capabilities e.g. MANO, YANG, TOSCA, ...
  – Domain service availability is delegated to the domain (fault handling, performance management etc.).
  – Service performance and status streamed back to the originating Service Orchestration instance (enabling it to provide a service view to customer portal)
• Cloud API compliant to ETSI NFV, MANO etc.
• SDN controller NBI API based on standardised YANG service models (‘Intents’).
• SDN SBI also uses YANG but using standardised device control models.
Thank You

Questions?
Vodafone Network Virtualisation Architecture for LSO, SDN and NFV

Kev Brackpool
SDN/NFV Architect
Vodafone Group