Investigating Intent API for Service Chaining

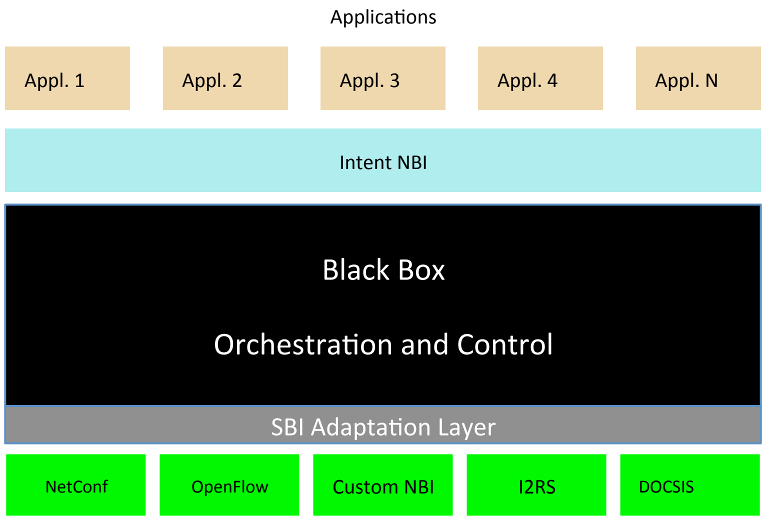
Andy Veitch NetCracker (NEC)

Goals

- Define and develop Intent NBI for service chaining
- Define for multiple underlying implementations – avoid network details
- Develop for a known use case – network access services
- Learn
  - Identify holes and areas to address
- Provide recommendations for SDO
- Advance open source Open Daylight NIC work
- Limit scope

Intent NBI

- Describes what the application wants
- No prescriptive information – nothing about how the service is to be fulfilled
- May be hierarchical – services composed of other services
- Goal also to enable portability and interoperability – deliver services irrespective of underlying technology



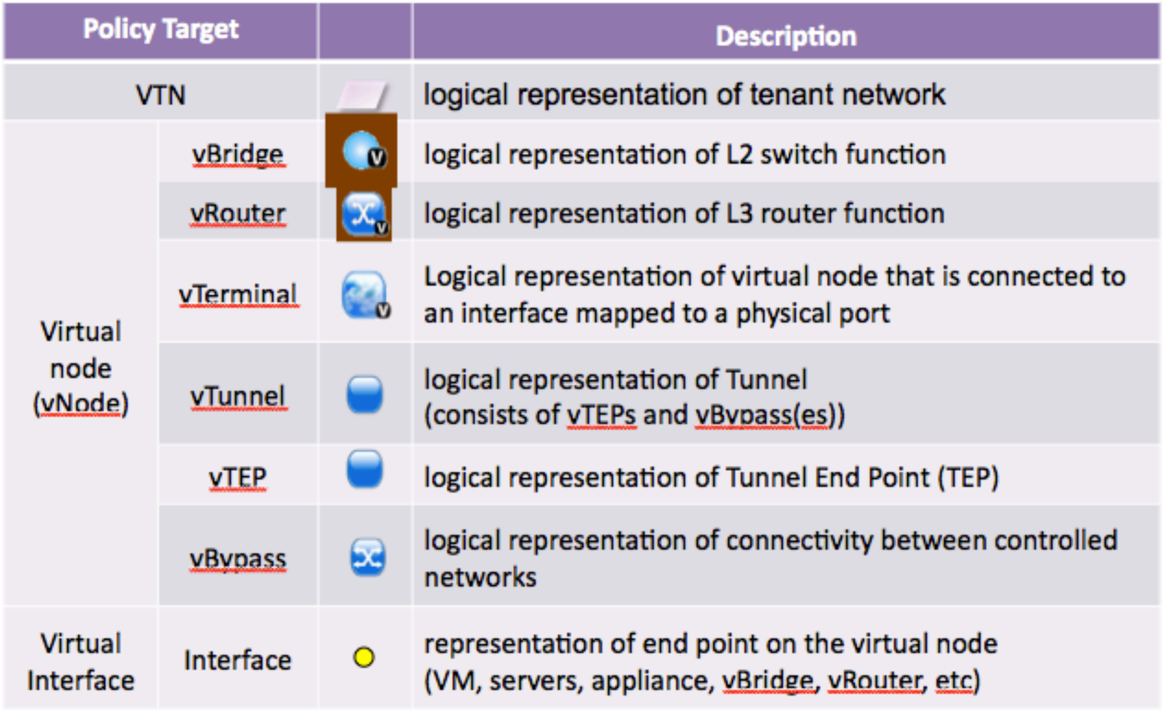
- SDOs and open source projects with active intent work
  - IETF (SUPA BoF, NeMo?)
  - ONF
  - Open Daylight (NIC, GBP)
  - ONOS
  - OpenStack (Congress, GBP)
  - OPNFV
- https://wiki.opnfv.org/_media/meetings/hares-opnfv-movie-usecases.pdf

Build on Open Daylight NIC model for Intent

- NIC included in ODL Lithium release
- Initial intent grammar defined – designed for augmentation
- Uses Yang models to represent Intent NBI definitions
- Use Virtual Tenant Networks – current project in ODL NIC
- Target NBI to be independent of underlying technology

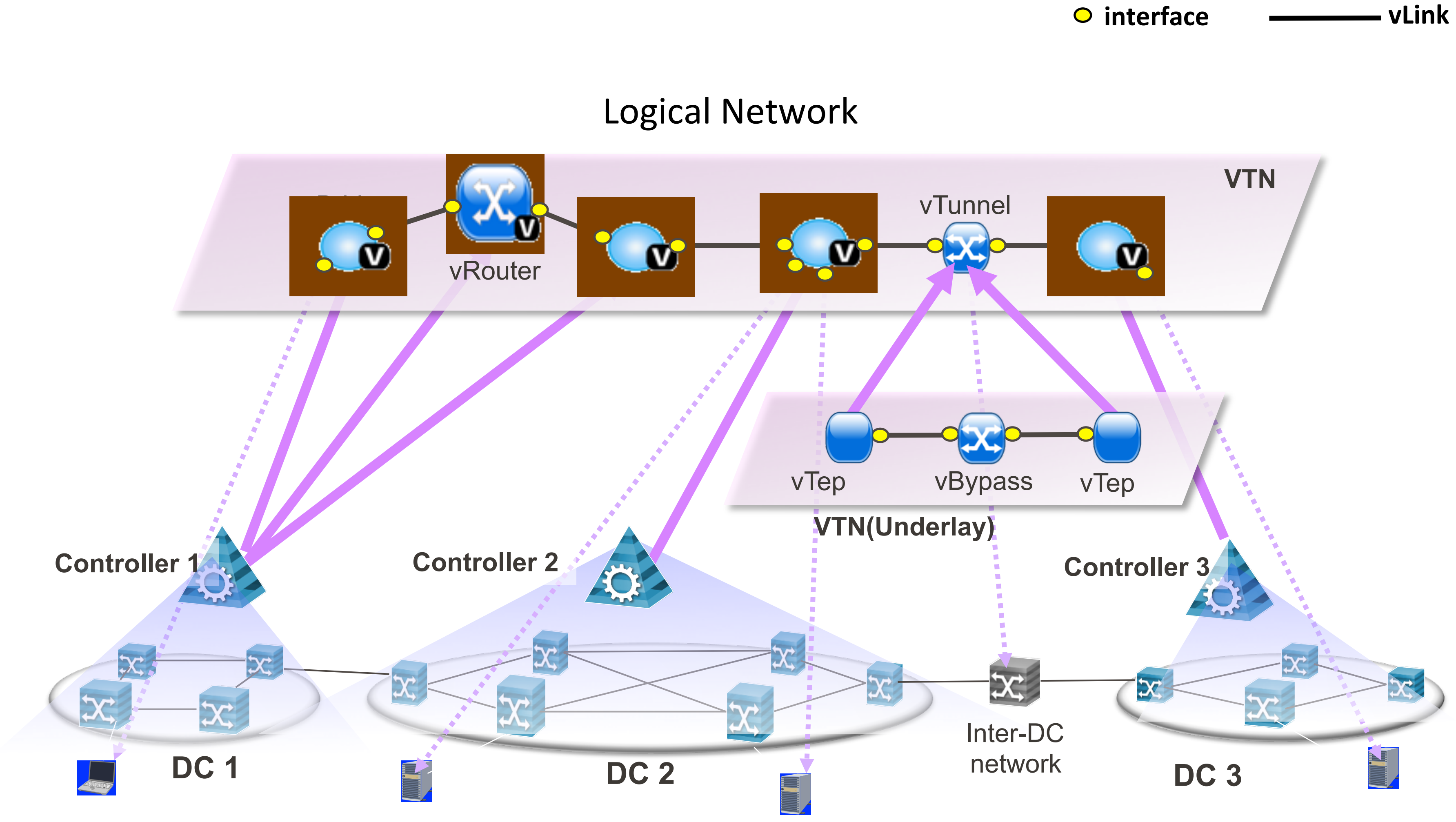
What is VTN?

- OpenDaylight Virtual Tenant Network (VTN)
  - VTN is a logical abstraction plane – separation from physical plane
  - Provides multi-tenant virtual networks on an OpenFlow SDN controller
  - https://wiki.opendaylight.org/view/OpenDaylight_Virtual_Tenant_Network_(VTN):Main



Policy Target			Description
VTN		▱	logical representation of tenant network
Virtual node (vNode)	vBridge	●	logical representation of L2 switch function
	vRouter	●	logical representation of L3 router function
	vTerminal	●	Logical representation of virtual node that is connected to an interface mapped to a physical port
	vTunnel	●	logical representation of Tunnel (consists of vTEPs and vBypass(es))
	vTEP	●	logical representation of Tunnel End Point (TEP)
	vBypass	●	logical representation of connectivity between controlled networks
Virtual Interface	Interface	○	representation of end point on the virtual node (VM, servers, appliance, vBridge, vRouter, etc)

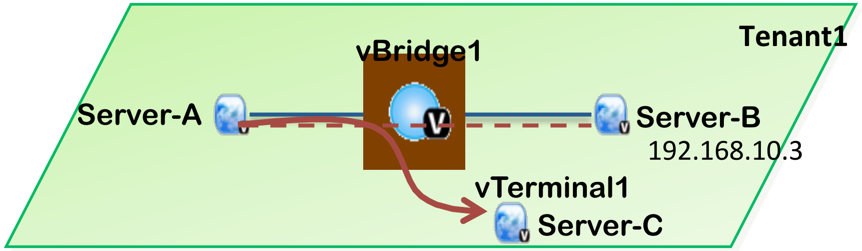
VTN Example



Physical Network

VTN Support for Flow Filtering and Traffic Redirection

- Apply flow filtering (a FEC) on any vNode – e.g. vBridge, vRouter
- Intent – redirect traffic to different node base on traffic type



Example Configuration

```
vtn Tenant1
{
 vbridge vBridge1
 {
    flow-filter in
    {
      sequence-number 1
     {
       match flow-list match-list-a
       action redirect
       redirect-destination vnode vTerminal1 interface if1
     }
 (snip)
}
```

Open Daylight NIC – Network Intent Composition

- Intent Model
  - *Subject Actions Conditions Constraints*
  - Similar (if not the same as) English grammar – Subject, verb, object, predicate
- Subject
  - Labels to identify groups of endpoints (affinity)
- Actions
  - Initial set arrived at from initial use cases
  - ALLOW, BLOCK, AUDIT, LOG. INSPECT, BANDWIDTH, REDIRECT, INJECT, LATENCY, LOSS, PROTECT, SECURE, QOS
- Constraints
  - Constrain how an "intent" is to be implemented
  - Described in abstract terms, no network or deployment specifics
  - Include-subject – intent instance must involve a given subject
  - Exclude-subject – intent instance must NOT involve a given subject
- Conditions
  - Defines when Intent is to be "active"

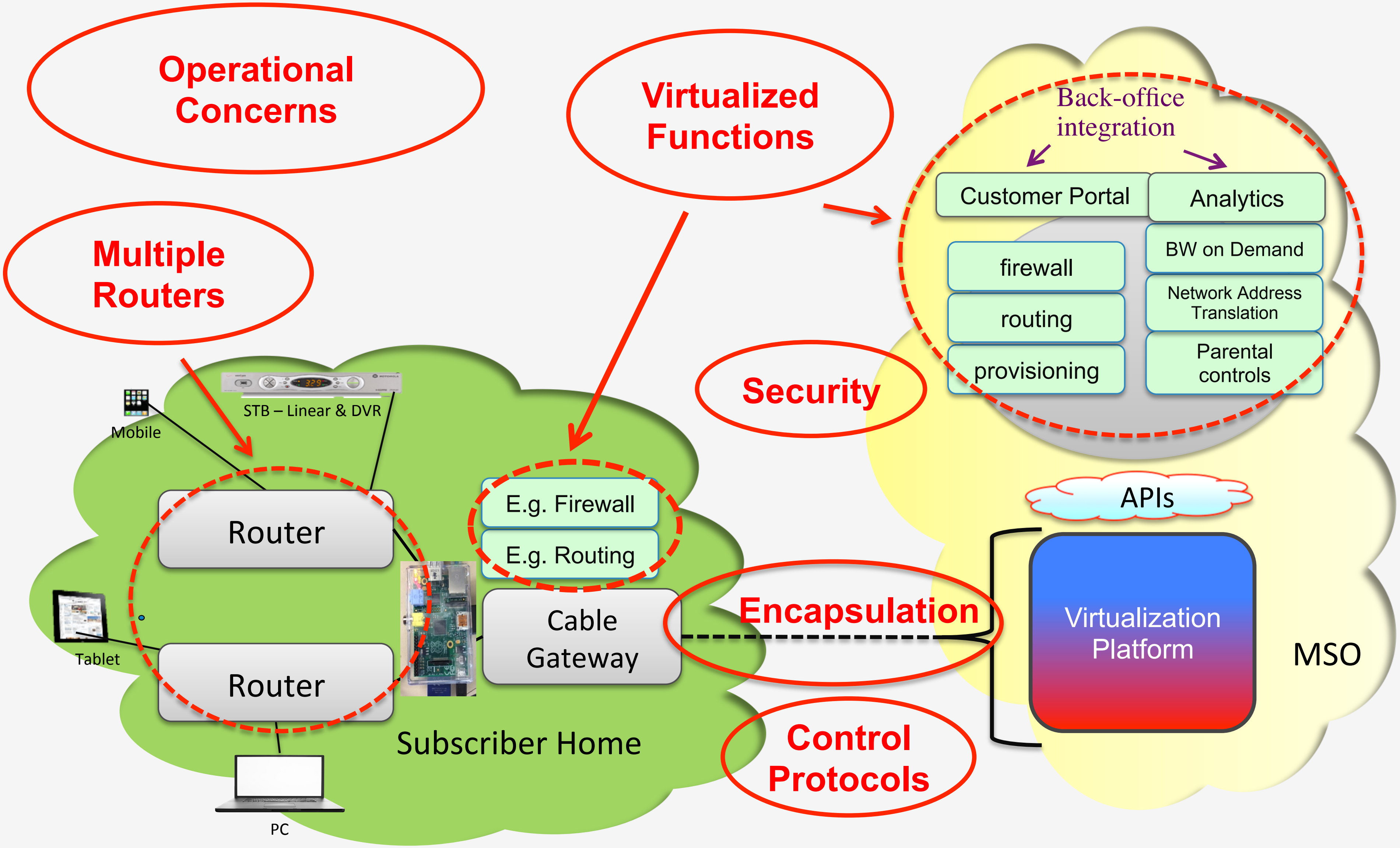
See presentation at
http://www.slideshare.net/opendaylight/whats-the-intent
NIC project wiki page a
https://wiki.opendaylight.org/view/Network_Intent_Composition:Main

Open Daylight NIC – Network Intent Composition

- Designed to be augmented
  - Modules can augment Subjects, Actions, Constraints, Conditions
  - Modules can implement renderers – translate intents to network model changes

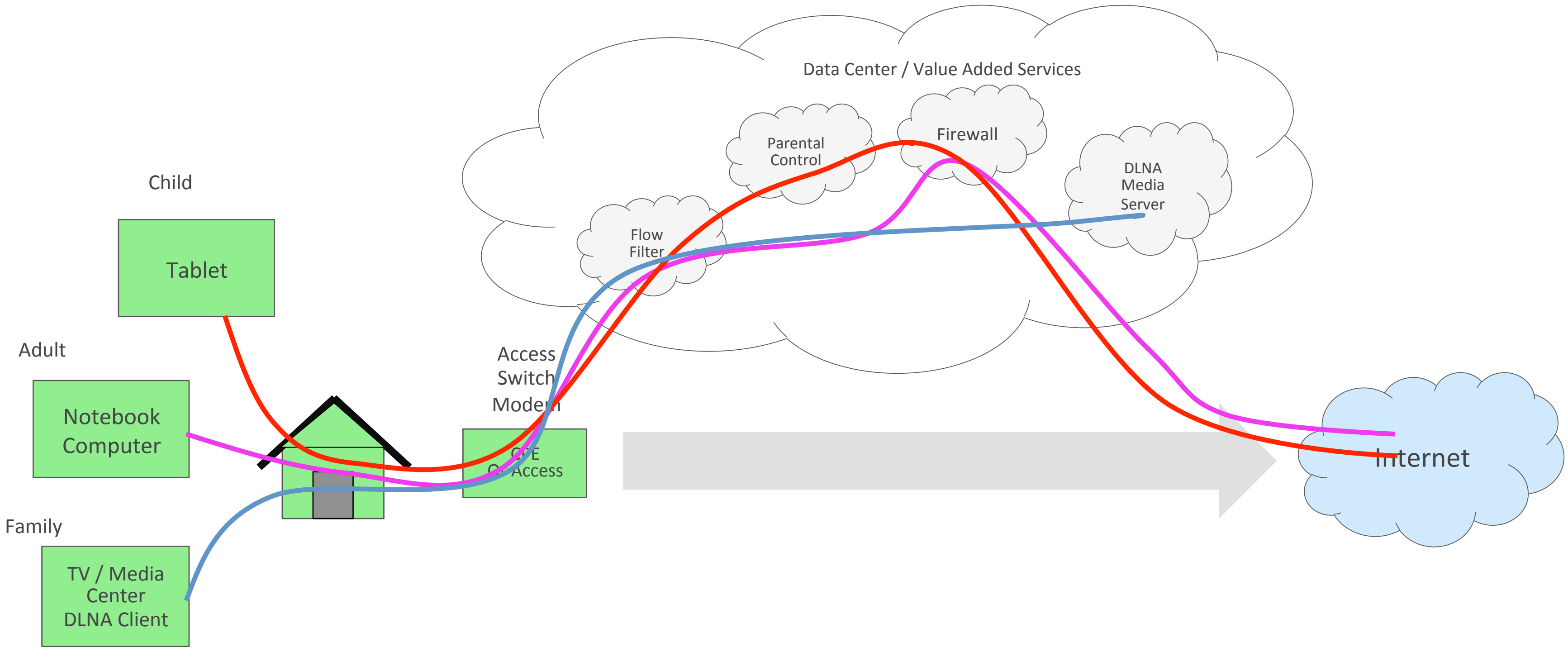
Virtual Home Networks - Challenges

CableLabs



Forwarding of Network Traffic to Deliver Services150

- VNFs are in the operator’s data centers – “the Cloud”
- The specific type and configuration of each VNF is based on the intent, context and relevant policies

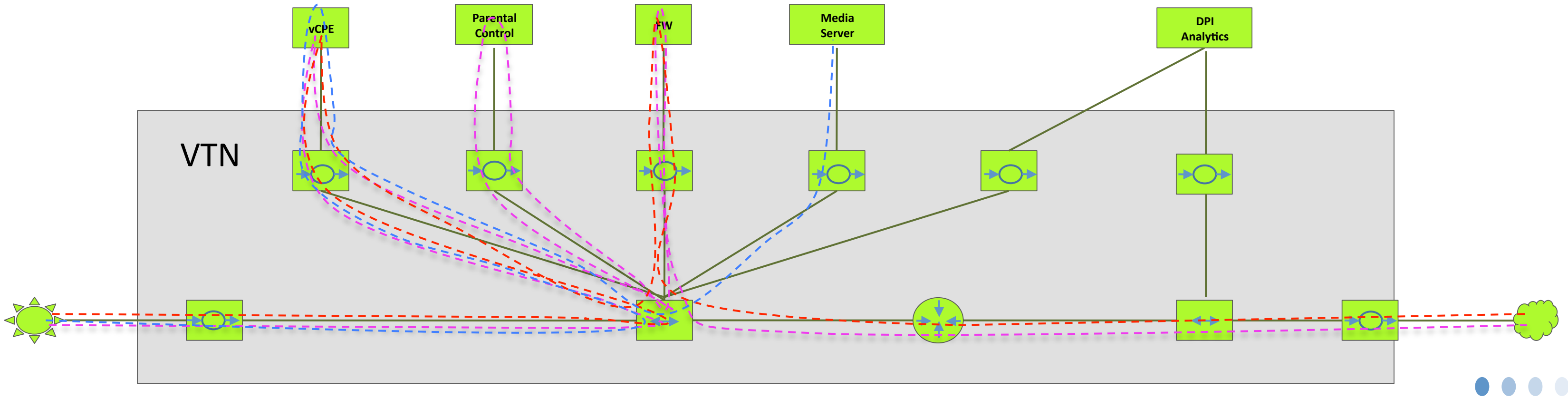


Intent NBI for Service Chaining

Top Level Intent (GUI)	Provide user xyz the Family Gold service and add DDoS service
Policies and <user xyz> context	Family Gold composed of services – children access, adult access, media access, broadband access (100Mbps)
	Intent NBI for each component service – "Provide user xyz children access", etc.
	Children access service composed of "Parental Control" and Firewall service -
	Identify affinity group of child endpoints (MAC addresses) (Match)
	Identify home media center endpoint (group) ((Match)
Set of NIC Intent input	Redirect child group to parental-control and and firewall to internet
	Redirect adult group to firewall and to internet
	Redirect media group to media server
	Create 100Mbs connection to xyz

- Layers of abstraction and models
- Top level NBI to lowest layer implementation

VTN setup – configure filters, VTN virtual infrastructure



ETSI Network Services and Virtual Network Function Forwarding Graphs (VNFFG)

- Difficult to map layers of intent – no support today for composition of services – single layer of abstraction
- Networks services can include multiple VNFFGs – choose option based on conditions and policies – must have all possible VNFFGs defined
- Could model NS, VNFFG, etc. as Yang

Not Intent

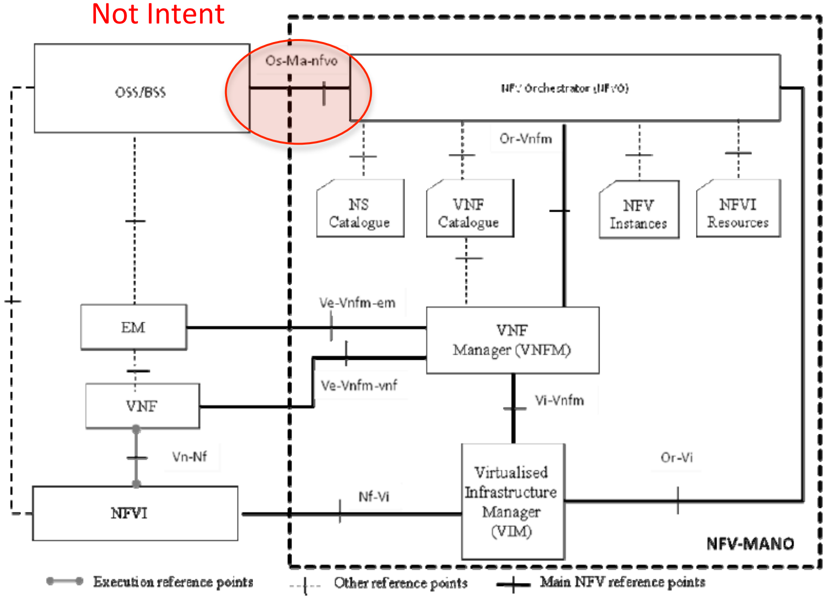


Figure 5.1: The NFV-MANO architectural framework with reference points

Network Functions Virtualisation (NFV); Management and Orchestration

ETSI GS NFV-MAN 001 V1.1.1 (2014-12)

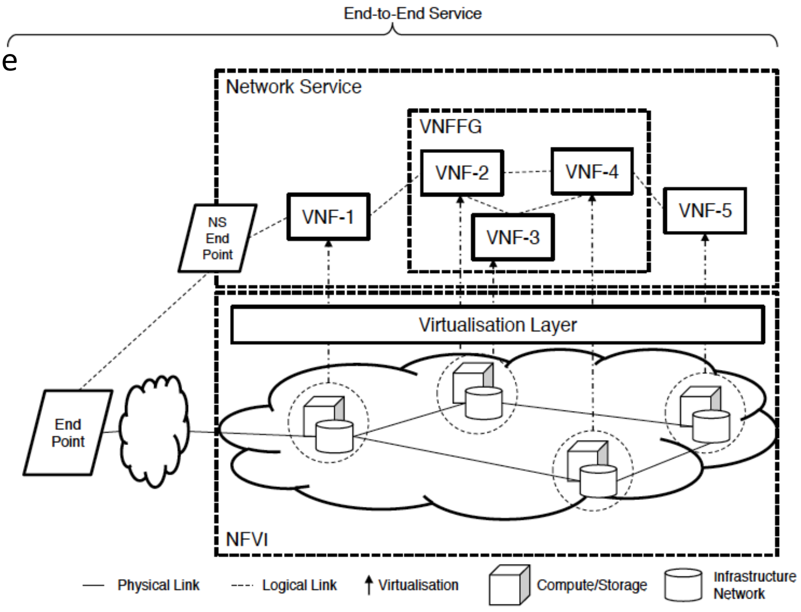
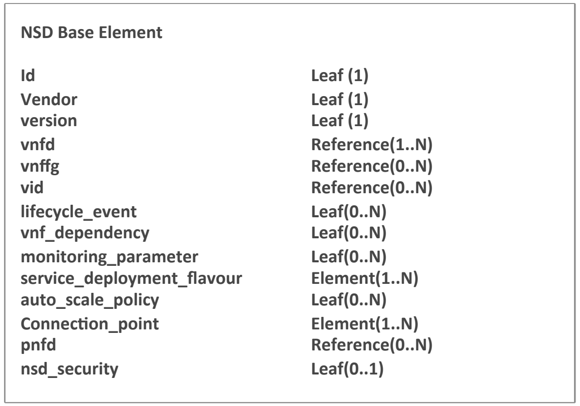


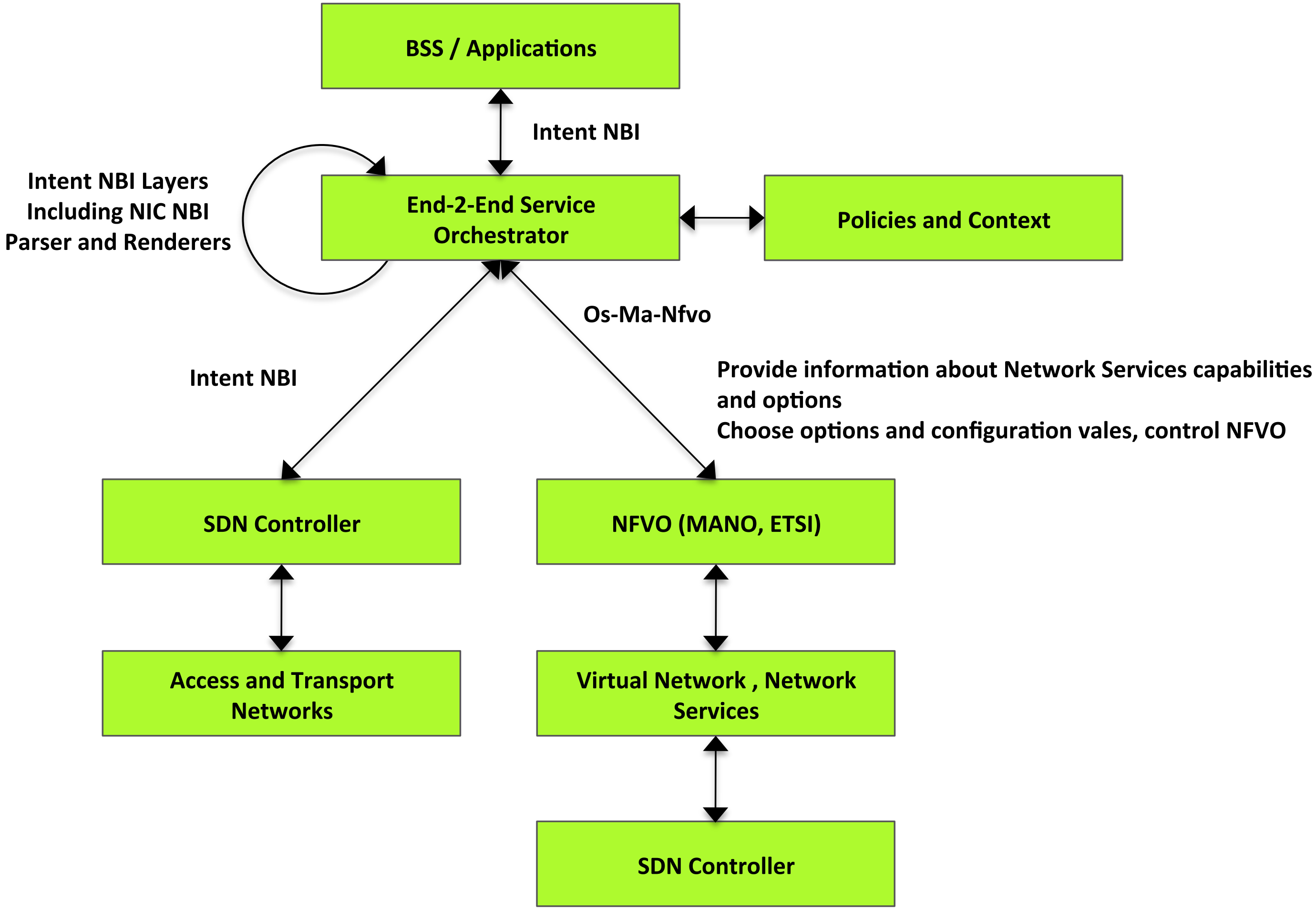
Figure 6.6: Example of a Network Service as part of an end-to-end service



NSD Base Element

Id	Leaf (1)
Vendor	Leaf (1)
version	Leaf (1)
vnfd	Reference(1..N)
vnffg	Reference(0..N)
vid	Reference(0..N)
lifecycle_event	Leaf(0..N)
vnf_dependency	Leaf(0..N)
monitoring_parameter	Leaf(0..N)
service_deployment_flavour	Element(1..N)
auto_scale_policy	Leaf(0..N)
Connection_point	Element(1..N)
pnfd	Reference(0..N)
nsd_security	Leaf(0..1)

Architectural View


BSS / Applications
Intent NBI
Intent NBI Layers
Including NIC NBI
Parser and Renderers
End-2-End Service Orchestrator
Policies and Context
Os-Ma-Nfvo
Intent NBI
Provide information about Network Services capabilities and options
Choose options and configuration vales, control NFVO
SDN Controller
NFVO (MANO, ETSI)
Access and Transport Networks
Virtual Network , Network Services
SDN Controller

Conclusions / Future work

- Vocabulary and semantics must be aligned among cooperating software agents
  - Mandatory to support portability / interoperability
  - It can be difficult to define across domains
  - IETF should continue work to define modeling – build commercial experience and open source
    - Should SUPA be moved to WG, same comment regarding policies
- Composition of services is very important to support – “code reuse” – layers of NBI and models
- ETSI should consider these requirements and need for flexibility
- Continue work to develop chaining NBI
- FWIW – Open Daylight summit next week – presentation on chaining on ODL VTN