U-4 (SDN).

DOMS | Page No.
Date / /

* Northbound Application Interface -

→ Interface / set of protocols that allows appln or s/w sys to interact with a higher-level component in h/w architecture.
→ Interface through which appln communicate with SDN controller.
→ SDN controller → central brain of h/w → controls how h/w traffic flows
→ makes decisions on how packets are forwarded.
→ Northbound API → defines methods, data structures & protocols use to exchange info with SDN controller
→ allows appear to programming retreive h/w info, set policies, configure h/w paths.
→ Northbound API → provides standarized way for appl to send commands.
→ By using Northbound API → appln can leverage capabilities & flexibility of SDN.
→ to implement h/w services.
→ optimises traffic flows..
→ monitor h/w performance
→ performs other h/w related tasks.
→ API abstracts → underlying complexities of SDN controller.
→ provides a simplified & standarized interface for appl"

→ specific design & features of N API can vary depending on SDN controller implementation.

→ common examples of Northbound APIs used in SDN include. OpenFlow, NETCONF, RestFull APIs.

DOMS | Page No.
Date / /

✱ Current Language & Tools -

① Frenetic -

- domain specific language for programming OpenFlow networks.
- allows new operators to program n/w as a whole than manually configuring configuring each connected n/w device.
- introduces set of purely functional abstractions enable module program.
- defines high level, programming centric, packet processing operators & eliminates many of difficulties of 2 tier prog model.
- embedded in python → 2 level abstraction.
  - 1) High level abstraction.
  - 2) Modular constructs.
  - 3) portability
  - 4) Rigorous semantic foundation.

② Procera -

- High level n/w control language that allows new operators to express reactive n/w control policies, without having to resort.
- networks. devices n/w intelligence solns based on deep packet inspection technology.

③ Ryu -

- highly modular, small SDN controller → Python.
- core of Ryu is smaller than controllers, every feature is implement.
- supports multiple OpenFlow versions, along with related protocols.
- Has no form of governance or corporate sponsors.

DOMS | Page No.
Date / /

④ Open Daylight -
- modular controller written in Java.
- supports large no. of networking protocols OpenFlow, BGP, LISP.
- has a large amt of backing from diff companies.
- uses Apache Karaf framework from diff. companies
- config & logging handled by Karaf.
- Karaf has no. of components of available for use, most important a web server used by northbound interface of Opendaylight.

⑤ NetKAT -
- n/w language for SDN programming has mathematical & semantic foundation.
- applies its complete equational theory & provides techniques.
- appln in h/w include → checking reachability, isolating traffic b/w programs.

⑥ Mininet (Joel) -
- network emulator capable of creating virtual networks with 100s of host & switches on single computer.
- Based on Linux process virtualization to run nodes in OS kernel.
- mininet can create a realistic virtual n/w on any type of machine.
- provide an inexpensive scalable & streamlined development running in live.
- Basic commands -
  → sudo mn -c (after logging into error, start mininet reset to initate.
  → sudo mn (start mininet w/o cleaning it up)
  → mininet > net (to list typology type)
  → mininet > help (display mn CLI commands).
  → mininet > nodes (display nodes)

⑦ Floodlight -
- Java based OpenFlow controller
- developed by Big Switch Network & used in commercial switches
- compatible with h/w & virtual switches as OVS.

Composition of SDN -

① Flow table -

→ fundamental data structure in SDN devices

→ FTs allow device evaluate incoming packets → takes apt action based on contents of packet.

→ Resides on h/w device + consists of series of flow entries + action to perform when a packet matching flow arrives at device

→ When SDN device packet, consults in flow tables in search of match

→ If doesn't find match, switch can either drop packet or pass it to controller.

→ Each flow table entry contains -

(a) Match fields -

(b) Priority -

(c) Counters -

(d) Instructions -

(e) Timeouts -

(f) cookie -

② SDN s/w switches -

① SDN s/w device implementations found in s/w based h/w devices, such as hypervisors of a virtualization system.

② Implementation of SDN devices in s/w is simplest means of creating an SDN device - flow tables, flow entries & match fields involved are easily mapped to s/w data structures such as sorted arrays & hash tables.

DOMS | Page No.
Date / /

(3) Hardware SDN devices -.

① H/w devices utilise special hardware designed to facilitate the inspection of incoming packets & subsequent decisions that follow based on packet matching operation.

② H/w includes layer 2 & layer 3 forwarding tables usually implements using -
CAMS - content addressable memories.
TCAMS - Ternary content addressable memories.

* NFV (Network Functions Virtualization) -

- way to virtualization h/w services, such as routers, firewalls load balancers (traditionally h/w or proprietry h/w).
- services are packaged as VMs on commodity h/w it allows service providers to run their n/w on standard servers.
- NFV architecture consists of.

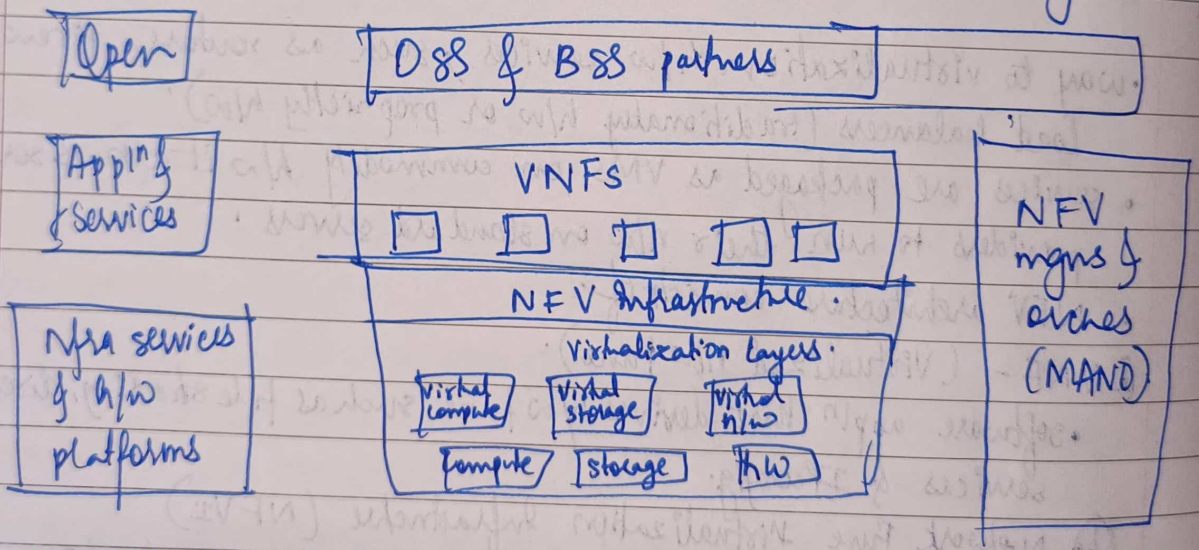
(a) VNF - (Virtualized h/w func).
- software appln that devices h/w func such as file sharing, directory services & IPconfig.

(b) Network Func Virtualization Infrastructure (NFVI).
- consists of infra components, compute, storage, networking on a platform to support s/w such as supervisor svc KVM of. a container mgmt platform needed to run new apps.

(c) Management & Automation & N/w Orchestration (MANO) -
- provides framework for managing NFV infrastructure & provising new VNFs.

(4) Critical goals of NFV arch -

① Decomposition of phys n/w elements into virtualized h/w fns to allow operators to choose best implementation from range of vendors
② Portability of VNFs to diff h/w platforms & hypervisors.
③ Rapid s/w based service delivery.
④ standard, open interfaces for improved interoperability in multivendor NFV soln.
⑤ use of low-cost, commercial off shelf (COTS) h/w
⑥ MANO provides overarching mgmt & orchestration of VNFs in NFV arch
⑦ MANO instantiates n/w services through automation, provisioning & coordination of work flows to VIM & VNF managers.



SDN Implementation -

1) Define a usecase 2) Assemble a cross fuctional team
3) Test in less critical network area 4) Review your test case
5) Gain maturity before expanding deployment.

- No. of prog lang for SDN & majority can be applied to open flow.
- Open flow - 1st & only standardized interface b/w control & infrastructure layer.

Appln → ① Data Centers ② Enterprise Networks.

③ optical h/w ④ Bandwidth mgmt
⑤ content availability.