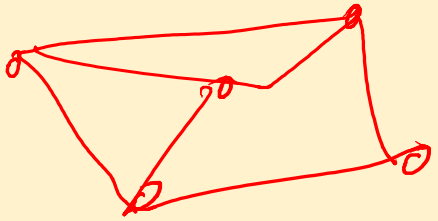
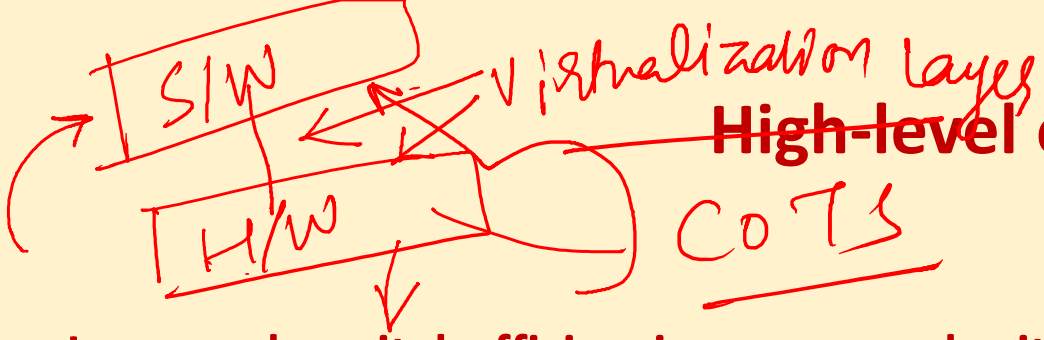
Unit 4
NFV
(Network Functions Virtualisation)

Ramakant Soni, Assistant Professor, CS Dept., B K Birla Institute of Engineering & Technology, Pilani, Rajasthan



NFV Objective

- Current networks are comprised of **diverse network functions**. These network functions are connected in a certain way, in order to achieve the desired overall functionality/ service that the network is designed to provide.
- Most current network services are defined by **statically combining network functions** in a way that can be expressed using an **NF Forwarding Graph** or **NF Set construct**. NF-FG shows the graph of logical links connecting NF nodes for the purpose of describing the traffic flow between them.
- A major change brought by NFV is that **virtualisation enables additional dynamic methods** rather than just static ones to construct and manage the network function graphs or sets combining these network functions.
- A major focus of NFV is to **enable the dynamic construction and management of network function graphs or sets**, and their relationships regarding their associated data, control, management, dependencies and other attributes.



High-level objectives of NFV are:

- **Improved capital efficiencies compared with dedicated hardware implementations.** This is achieved by using **commercial-off-the-shelf (COTS) hardware** (i.e. general purpose servers and storage devices) to provide Network Functions (NFs) through software virtualisation techniques.

- **Improved flexibility in assigning VNFs to hardware.** This aids both scalability and largely decouples functionality from location, which allows software to be located at the most appropriate places, referred to as NFVI-PoPs, e.g. at customers' premises, at network exchange points, in central offices, data centers, etc.

- **Rapid service innovation** through software-based service deployment.

- **Improved operational efficiencies** resulting from common automation and operating procedures.

- **Reduced power** usage achieved by migrating workloads and powering down unused hardware.

- **Standardized and open interfaces** between virtualised network functions and the infrastructure and associated management entities so that such decoupled elements can be provided by different vendors.

Virtualisation

Virtualisation means that a network function and part of the infrastructure are implemented in software.

It uses the technologies of IT virtualization to virtualize entire classes of network node functions into building blocks that may connect, or chain together, to create communication services.

Implementing Virtualisation

In non-virtualised networks, NFs(Network Functions) are implemented as a combination of vendor specific software and hardware (network nodes or elements).

NFV (Network Functions Virtualisation) introduces a number of differences in current networks as listed below:

- **Decoupling software from hardware**: As the network element is no longer a collection of integrated hardware and software entities, the evolution of both are independent of each other. This **enables the software to progress separately from the hardware**, and vice versa.

- **Flexible network function deployment:** The detachment of software from hardware helps reassign and share the infrastructure resources, thus together, **hardware and software, can perform different functions at various times.** Assuming that the pool of hardware or physical resources is already in place and installed at some NFVI-PoPs, the actual network function software instantiation can become more automated. **This helps network operators deploy new network services faster over the same physical platform**.

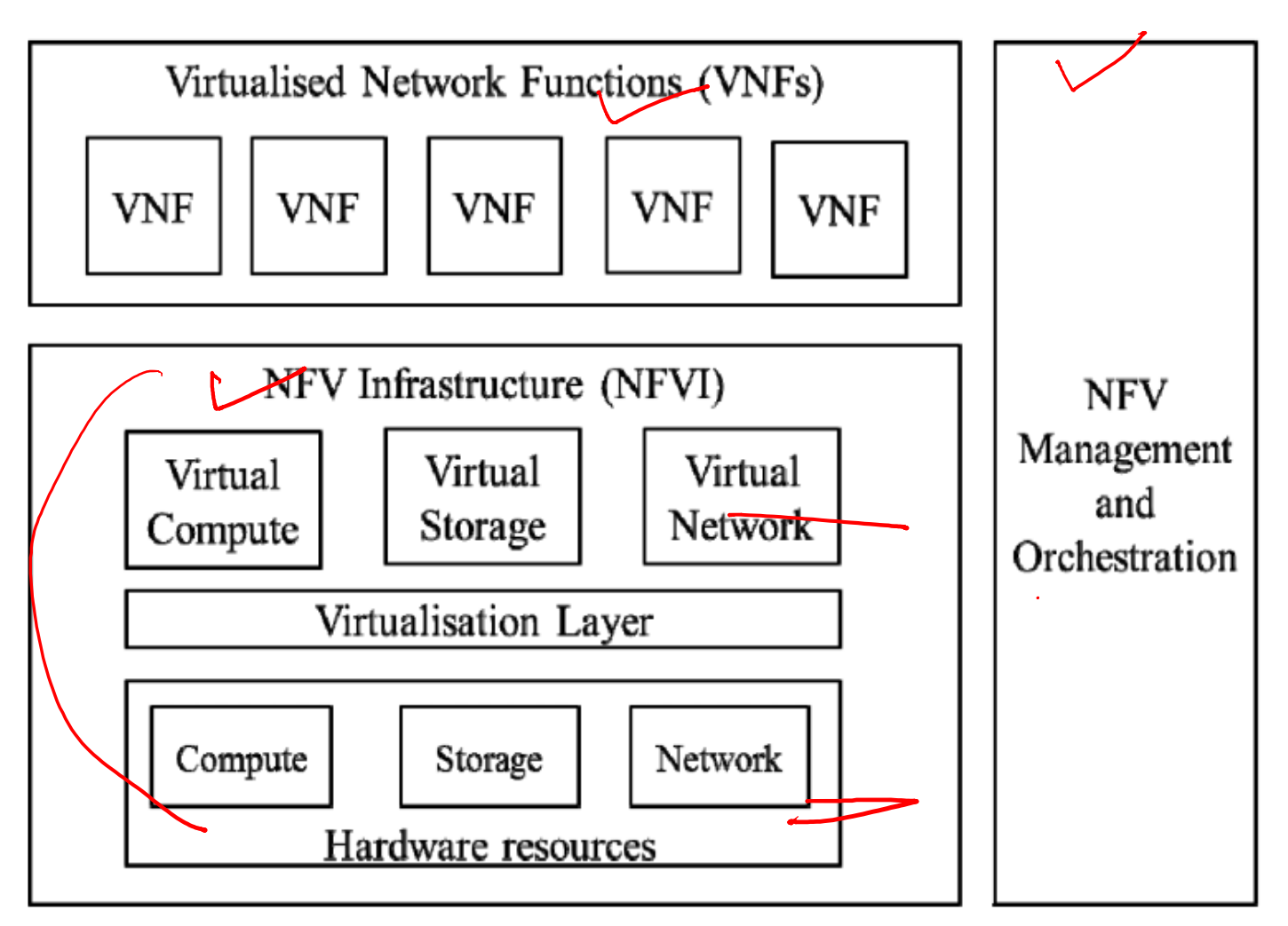
- **Dynamic operation:** The decoupling of the functionality of the network function into instantiable software components provides **greater flexibility to scale the actual VNF performance in a more dynamic way** .

High-Level NFV Framework

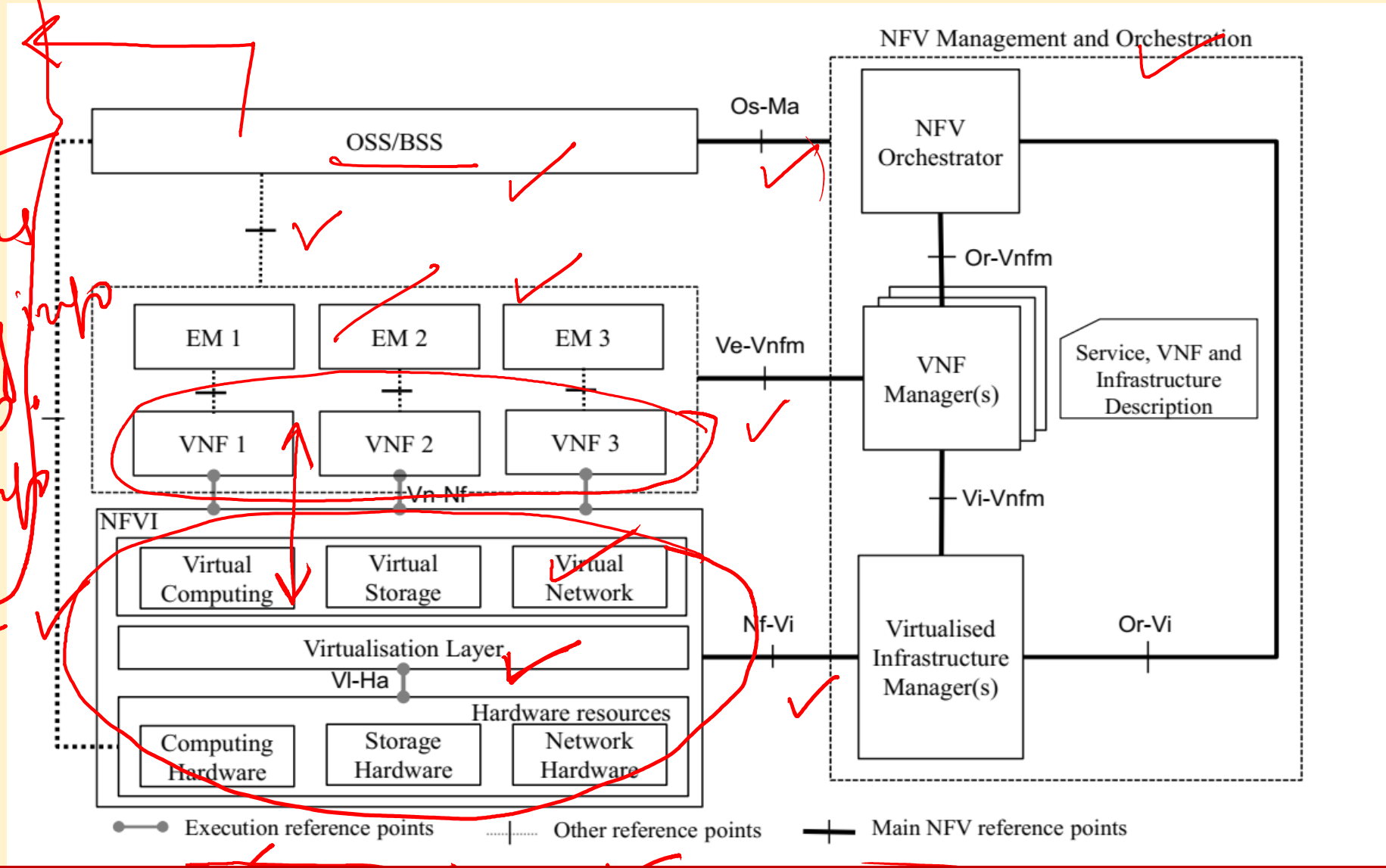
Network Functions Virtualisation visualizes the implementation of NFs as software-only entities that run over the NFV Infrastructure (NFVI).

The NFV framework consists of three main components:

- **Virtualised Network Function (VNFs),** as the software implementation of a network function which is capable of running over the NFVI.
- **NFV Infrastructure (NFVI),** including the diversity of physical resources and how these can be virtualised. NFVI supports the execution of the VNFs.
- **NFV Management and Orchestration (NFV-MANO),** which covers the orchestration and lifecycle management of physical and/or software resources that support the infrastructure virtualisation, and the lifecycle management of VNFs. NFV-MANO focuses on all virtualisation-specific management tasks necessary in the NFV framework



NFV Reference Architectural Framework

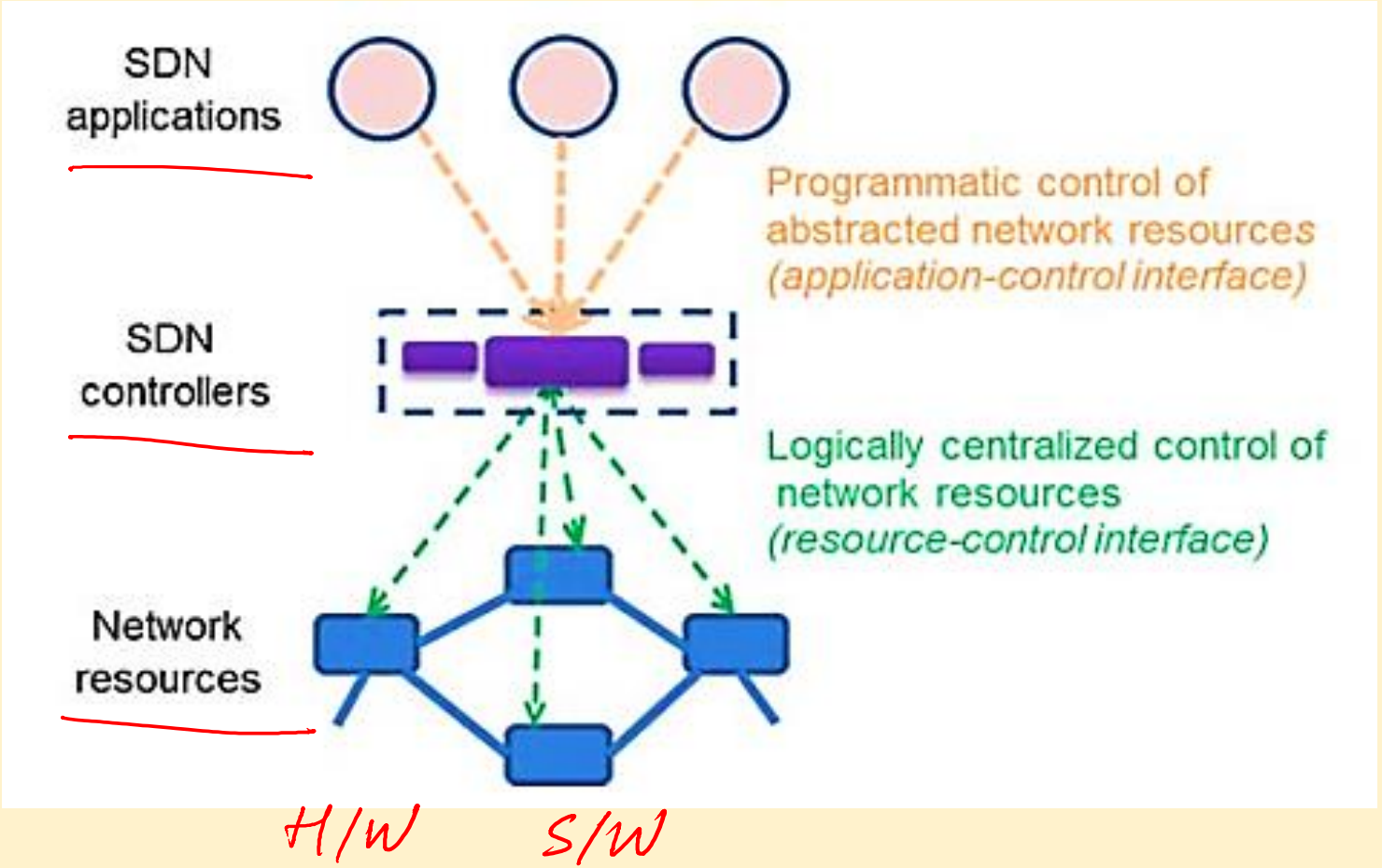


SDN in the NFV architectural framework

SDN

SDN is a set of techniques that enables to directly program, orchestrate, control and manage network resources, which facilitates the design, delivery and operation of network services in a dynamic and scalable manner.

SDN refers to software control of physical or virtual network resources that use standard interfaces (open APIs) to facilitate interoperability and evolution in a multi - vendor environment.



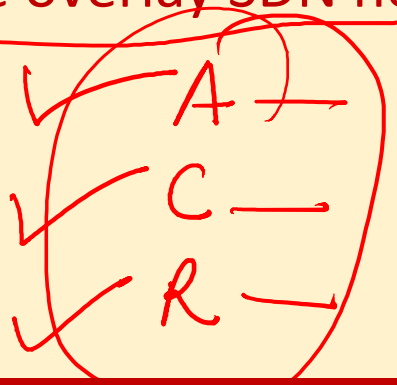
SDN in NFV Framework

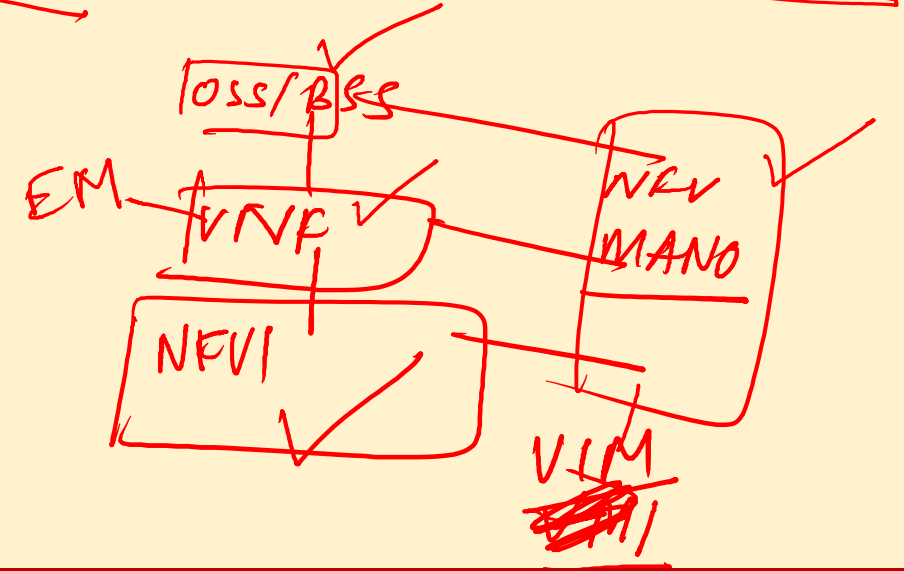
- Within the NFV architectural framework, SDN solutions might be used in the infrastructure domain, in the tenant domain or both.
- When used in the infrastructure domain, the SDN controller acts as a Network Controller
- The SDN controller is not necessarily a stand-alone physical entity, e.g. a software component(s) of the VIM.
- Ideally, when the SDN technology is used in the infrastructure domain, the VIM, the SDN controller, and the Network Resources (physical or virtual) form a hierarchy for delivering connectivity services.
- In some cases, multiple SDN controllers form a hierarchy across management and resources, depending on the placement of the functionality.
- The SDN controller responsibility includes very specific control functions, interfacing with management agents responsible for control and management functions.

SDN in NFV Framework

Many technical and non-technical issues need to be formulated and answered regarding all the functional entities that constitute this integrated architectural framework, such as:

- The position of the SDN resources.
- The position of the SDN controller.
- The softwarization & virtualisation of the various SDN entities.
- The interaction between the Element Managers, VNF Managers, SDN controllers and SDN applications that become enabled VNFs.
- The hierarchy of SDN networks.
- The position of the overlay SDN networks.
- Others.





SDN Control and Management Plane

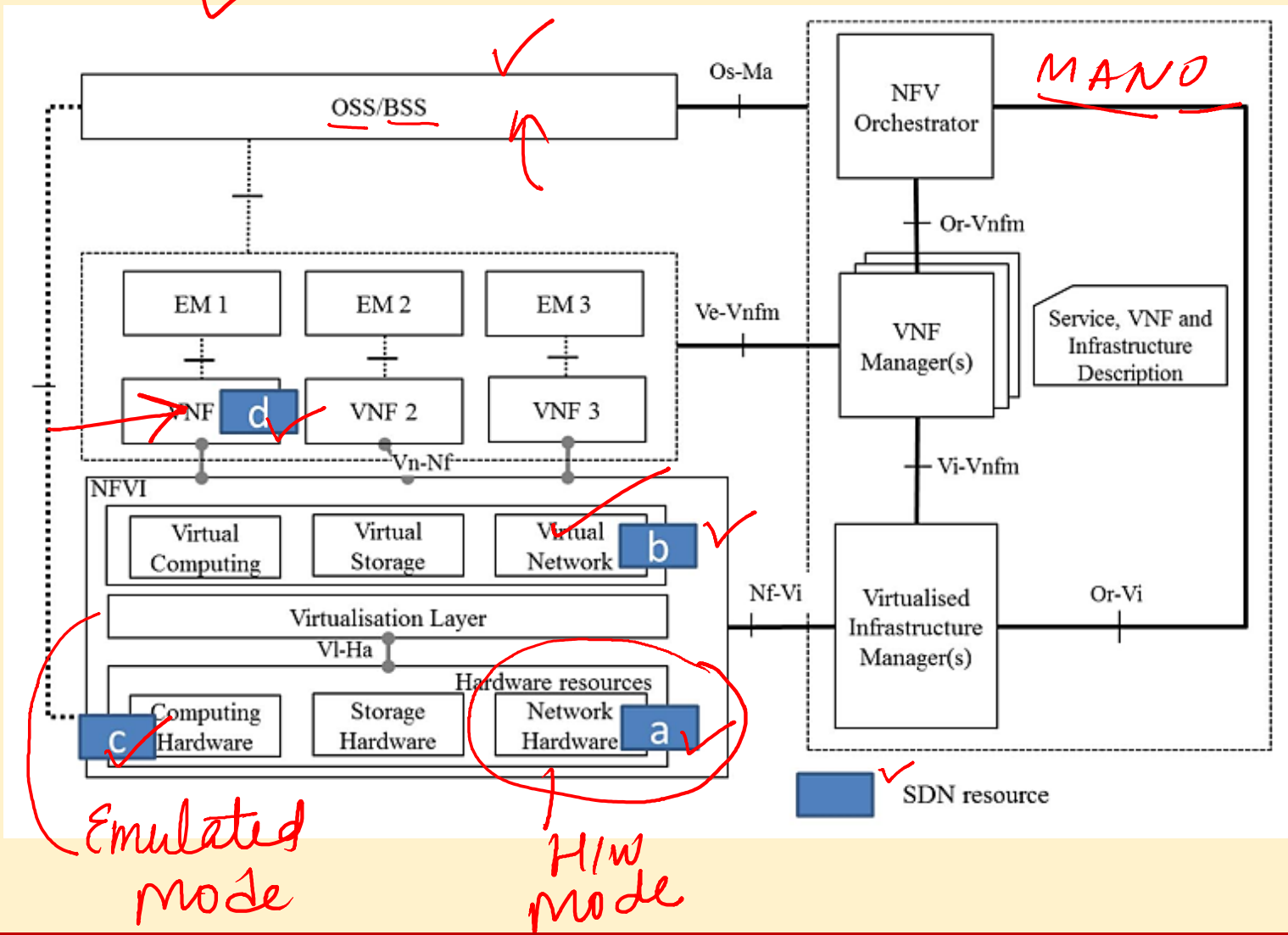
SDN Control Plane	SDN Management Plane
Is responsible for making decisions on how packets are forwarded by one or more network resources and pushing such decisions down to the network resources for execution.	Is responsible for monitoring, configuring, and maintaining network devices, e.g. making decisions regarding the state of a network resources.

Position of SDN resources in an NFV architectural framework

The first entities to be considered are the SDN resources. Multiple scenarios might be envisaged for their actual location or for their images:

- **Case a:** physical switch or router
- **Case b:** virtual switch or router
- **Case c:** e-switch, software based SDN enabled switch in a server NIC
- **Case d:** switch or router as a VNF

SDN resource might be logically part of the NFVI or belong to an independent tenant's domain.



Position of SDN Controller in an NFV architectural framework

SDN controller interfaces with SDN network resources via the Resource Control Interface. One SDN controller might interface with multiple SDN network resources.

The possible locations of an SDN Controller in the context of an NFV framework:

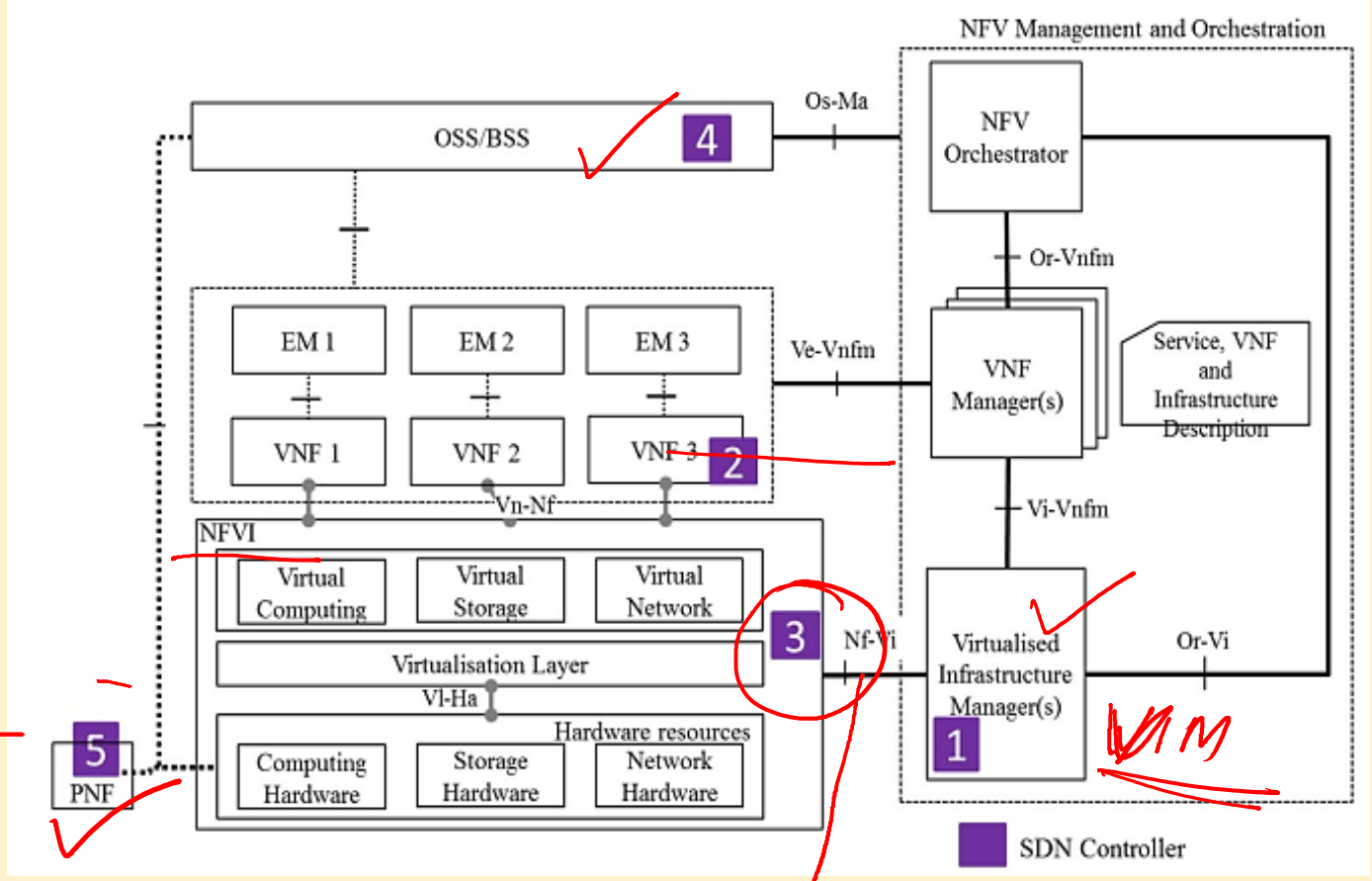
Case 1: SDN controller functionality merged with the VIM functionality, in such case the two functions are not distinguishable.

Case 2: SDN controller as a VNF is typically the case of an SDN controller Virtualised as a VNF itself, or being part of a VNF. This VNF might be logically part of the NFVI and therefore belong to a special infrastructure tenant or belong to an independent tenant.

Case 3: SDN controller in the NFVI is a classic case of SDN controller for the network connectivity in the NFVI, where the SDN controller is not implemented as a VNF.

Case 4: SDN controller part of the OSS

Case 5: SDN controller as a PNF



Position of SDN Application in an NFV architectural framework

SDN application interfaces with the SDN controller. An SDN application might interface with multiple SDN controllers.

The positions of the SDN applications:

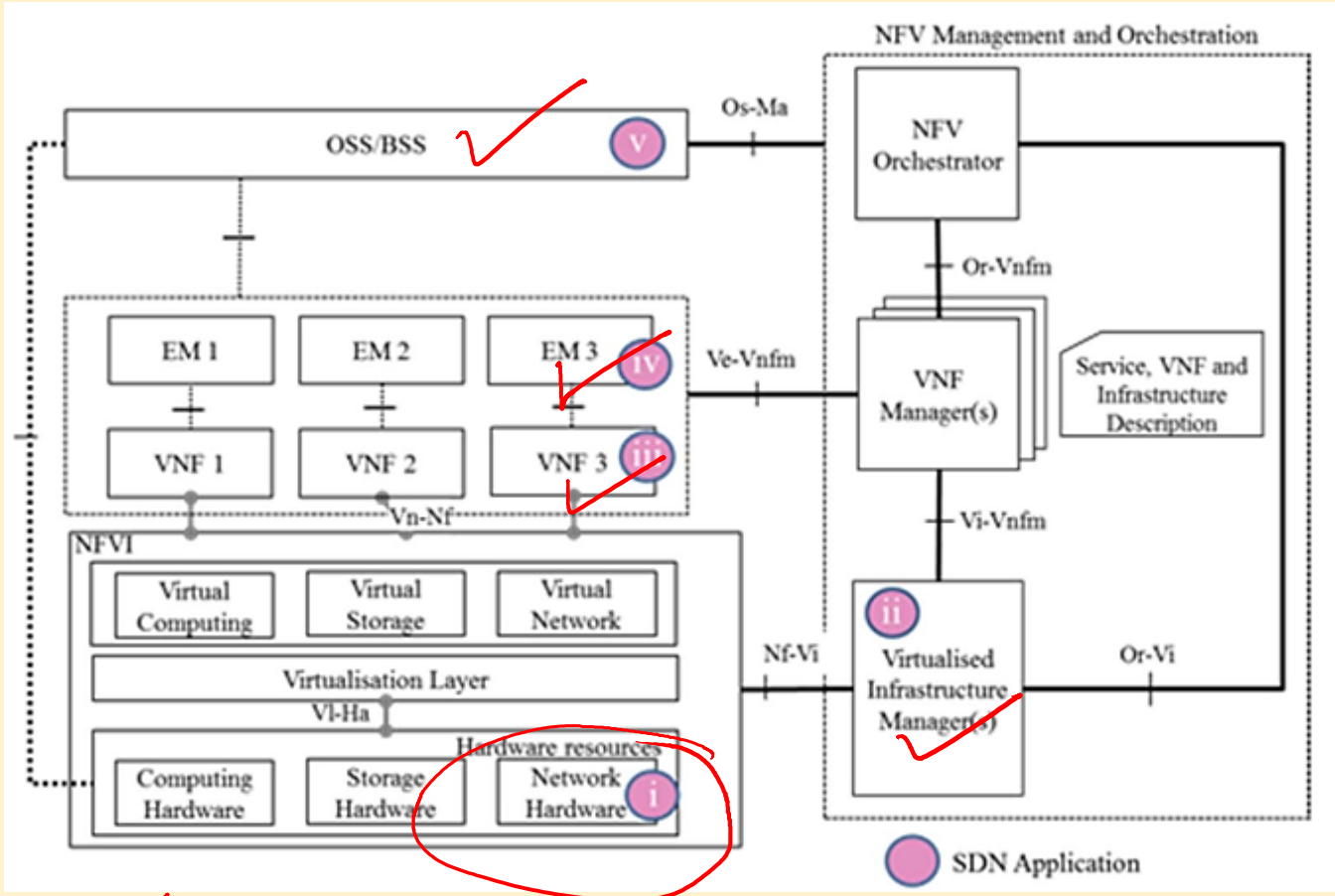
Case i - as part of a PNF: the network hardware might be a physical appliance talking to an SDN controller, or a complete solution including multiple SDN components, such as SDN controller + SDN application for instance.

Case ii - as part of the VIM: the VIM might be an application interfacing with an SDN controller in the NFVI .

Case iii - Virtualised as a VNF: the SDN application might be a VNF talking to an SDN controller, being Virtualised or not.

Case iv - as part of an EM: the SDN application might be an element manager interfacing with an SDN controller to collect some metrics or configure some parameters.

Case v - as part of the OSS/BSS: the SDN application might be an application interfacing with an SDN controller for instance in the OSS-BSS for tenant SDN service definitions



SDN controller interfaces in the NFV architectural framework

SDN controller comprises the following elements:

- **Application Control Interface : interface between an SDN controller and an SDN application**
  - it provides an application programmatic control of abstracted network resources.

- **Resource Control Interface : Interface between an SDN controller and SDN resources**
  - it is used to control network resources.

- **Orchestration Interface - interface between an SDN controller and an NFV Orchestrator:**
  - It might need to pass information between the 2 entities, such as topology information in both directions.
  - The interface might be an indirect interface.
  - The same interface might also be used between an SDN application and an NFV Orchestrator.

- **Controller-Controller Interface - interface between SDN controllers**
  - It might need to pass information between SDN controllers either in the same hierarchy or in different hierarchies