6.1 Virtualization adn Virtual Machines

6.1.1 How can Virtualization improve resource utilization and flexibility?

6.1.2 How can Layer 2 Switching make the migration process of virtual machines in a network seamless?

6.2 How can SDN and NFV can reduce costs and improve DCN management?

6.3 What are the advantages and disadvantages of using butterfly, clos (and folded clos), and flattened butterflies topology?

6.4 How does data travel from Processor 2 to Processor 5 in the given butterfly network?

Assuming that the network depicts a *wrapped* butterfly network, such that rank 0 gets merged with rank 3. The packet transmitted over the links has the following form:

Header = 101 (because we want to the send the payload to processor 5)	message M	trailer like checksum etc.

- The packet is sent out from processor 2 and arrives at N(0,2). The leftmost bit of the header is removed to decide the direction. Since it is 1, the right link which connects to N(1,6) is selected.
- After arriving at N(1,6) the 0 of the header is "dequeued" which lead to the selection of the left link of the node transferring the packet to node N(2,4).
- Since the header is now 1 the right link is selected to send the packet to node N(3,5).
- The header is now empty and processor 5 receives the packet including the payload and the trailer.

6.5 What are the advantages of high-radix over low-radix?

A good example for an high-radix topology would be the flattened butterfly topology, because all of the routers in the lowest dimension are fully connected to each other. Therefore the routers can exploit the advantage of packaging locality where each package can be packaged locally with short cables, which can significantly reduce the network's cost. Furthermore by reducing the diameter of the network, high-radix networks are more advantageous both for latency and power - ultimately affecting the costs more.

6.6 How can Traffic Engineering be applied in Data Centres and what are the opportunities and challenges?

Because Data Centres consist of 10's and 100's of thousand hosts which are often closely coupled the load must be managed and balanced to avoid bottlenecks. Also there is a huge communication traffic between each host which can easily lead to performance issues if some nodes are overloaded. In this scenario introducing Traffic Enginnering can help, because its goal is to avoid congestions by keeping the links from being overloaded. Therefore improving the performance of the traditional switching protocols can lead to an huge performance increase of the whole data centre.

The opportunities of using Traffic Engineering in Data Centres are that network gets more efficient because of low propagation delay and high capacity and the ability of using specialized topologies like Fat Tree or Clos Networks which enable the use of hierarchical addressing. Also TE enables the flexible movement of workload by replicating the services on multiple servers and data centres. Finally TE has the ability to control both network and hosts which can be used to optimize routing and server placement as well as moving the network functionality to an end host.

The most common challenges for Traffic Engineering are that in data centres many switches, hosts, and VMs must work together which affects the need of scalability. Also in data centres the churn rate, failures of components or VM migration, can be really high which must be considered when implementing Traffic Engineering. Furthermore the Traffic itself can be very volatile and most likely occur in unpredictable patterns. There are also special performance requirements because the applications can be delay-sensitive or the resources are isolated between each tenant.