Network Equipment and Multi-Port Server Interfaces

A *network switch* in each rack connects to each server and provides communication among the servers as well as communication to the rest of the data center and the Internet.

Data center switches use *Ethernet* technology, and the switches are sometimes called *Ethernet switches*.

The switch in each rack is usually placed near the top, giving rise to the term *Top-of-Rack switch* (*ToR switch*).

To permit rapid data transfers, the connections between the ToR switch and each server must operate at high speed.

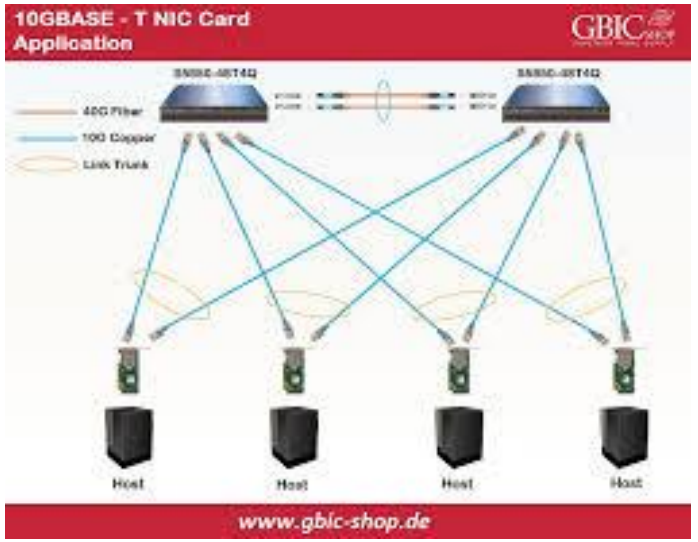
Data centers used 100 *Gigabit per second* (*Gbps*) Ethernet leading to the name *GigE*.

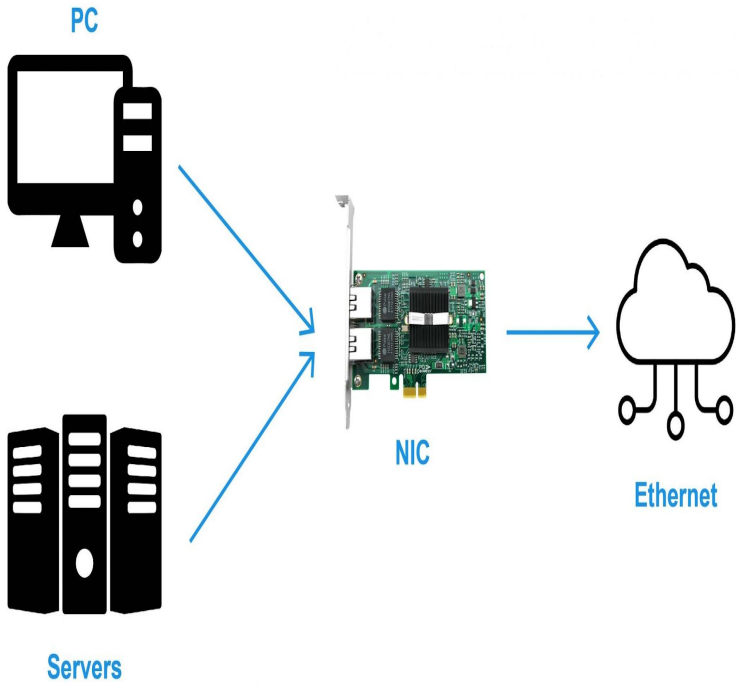
Each server can use a *multiport network interface card* (*multi-port NIC*).

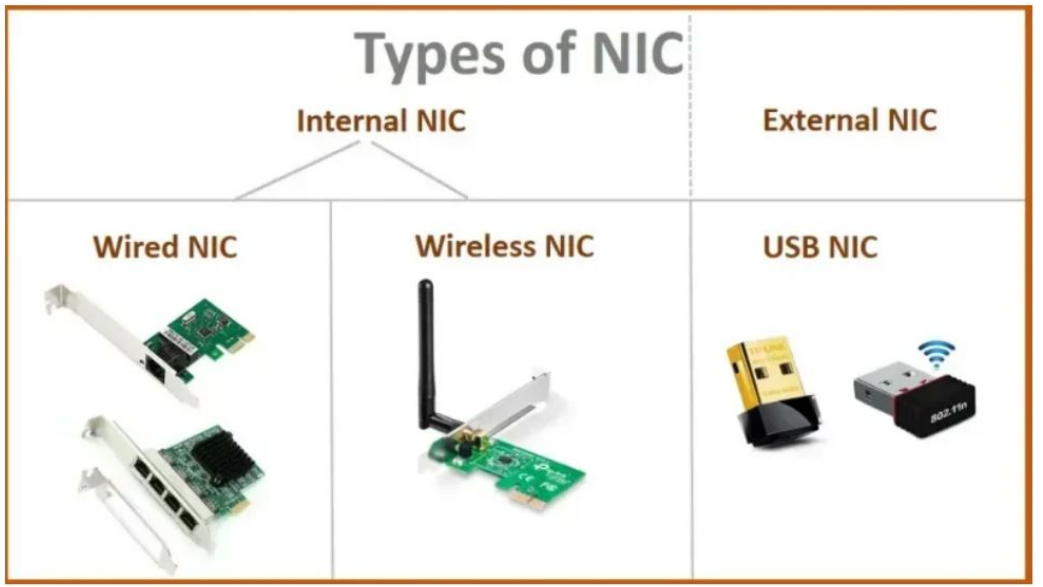
Each of the ports connects to the ToR switch, and each operates independently and in parallel.

Thus, a NIC with *K* ports means the server has *K* network interfaces and allowing it to send and receive *K* times as much data each second as a single network interface.

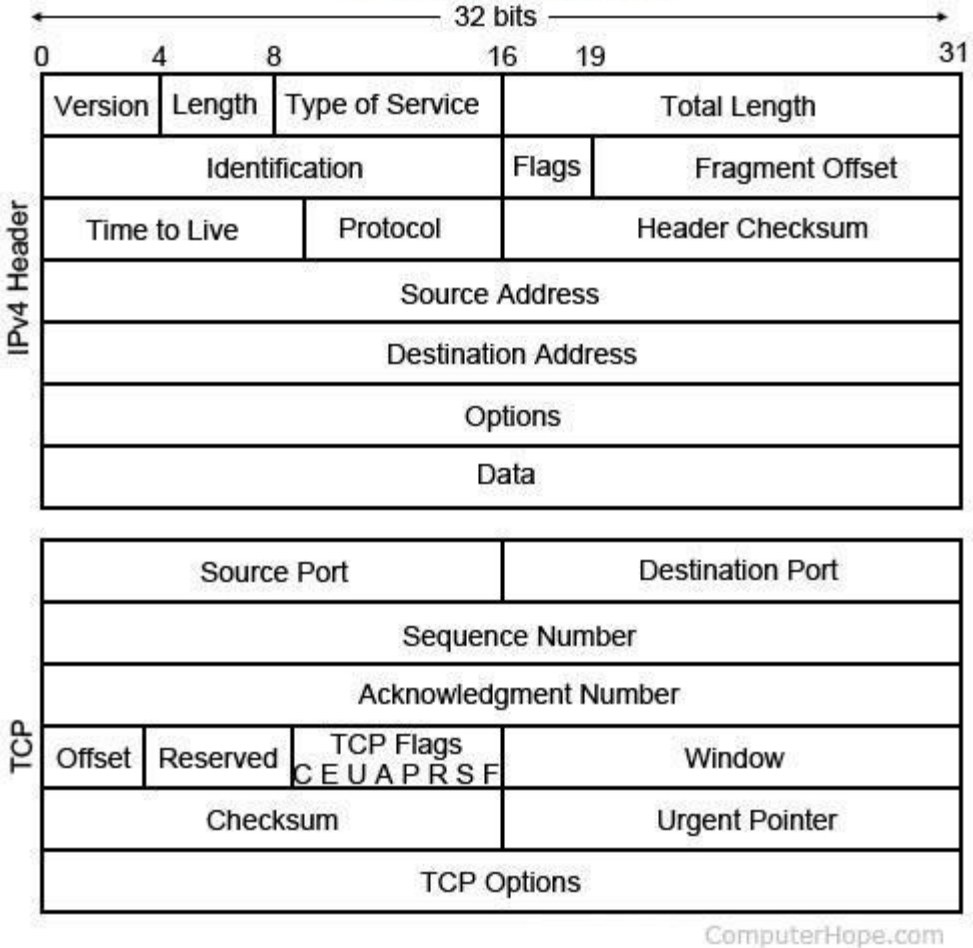
A multi-port NIC works well with a multi-core server because it allows the server to send and receive more data.


10GBASE - T NIC Card
Application
GBIC
Host
Host
Host
Host
www.gbic-shop.de


PC
NIC
Ethernet
Servers


Types of NIC
Internal NIC
External NIC
Wired NIC
Wireless NIC
USB NIC

TCP/IP Packet



Smart Network Interfaces and Offload

Each time a packet arrives at a server, fields in the packet headers must be examined to determine whether the packet has been formed correctly and whether the packet is destined to the server.

Similarly, each time a packet is sent, headers must be added to the data.

In a conventional computer, software in the operating system performs all packet processing tasks, which means the processor spends time handling each packet.

The processing required for network packets can be significant, especially if the data must be encrypted before being sent and decrypted when received.

To send and receive data at high speed, the network interfaces used in many data centers contain special hardware that handles packet processing tasks.

A *smart NIC*, such an interface card can assemble an outgoing packet, compute a checksum, and even encrypt the data.

Similarly, a smart NIC can check the headers on each incoming packet, validate the checksum, extract and decrypt the data, and deliver the data to the operating system.

- A variety of network architectures have been used in data centers. Although many variations exist, we can group them into two broad categories based on the type of traffic they are intended to handle:
- North-south traffic
- East-west traffic

North-south traffic. Industry uses the term *north-south traffic* to describe traffic sent between arbitrary computers on the Internet and servers in a data center.

Generally, data centers focused on large-scale web sites. Web traffic falls into the category of north-south traffic.

- A single load balancer dividing incoming requests among all servers in the data center.
- A network device that performs load balancing has a limited number of connections. Therefore, to accommodate large scale, a network must be designed as a hierarchy with an initial load balancer dividing requests among a second level of load balancers.

- *East-west traffic:*
- As data centers moved from large-scale web service to cloud computing, network traffic patterns changed.
- When the company fills an order, software may need to access both a catalog of products as well as a customer database.
- Similarly, when a manager approves time off, software may need to access an employee's record, payroll data, and the company's accounting system.
- Communication within the company means network traffic will travel among the servers the company has leased (i.e., from a server in one rack to a server in another). In terms of Figure , communication proceeds left and right, which leads to the name east-west traffic.

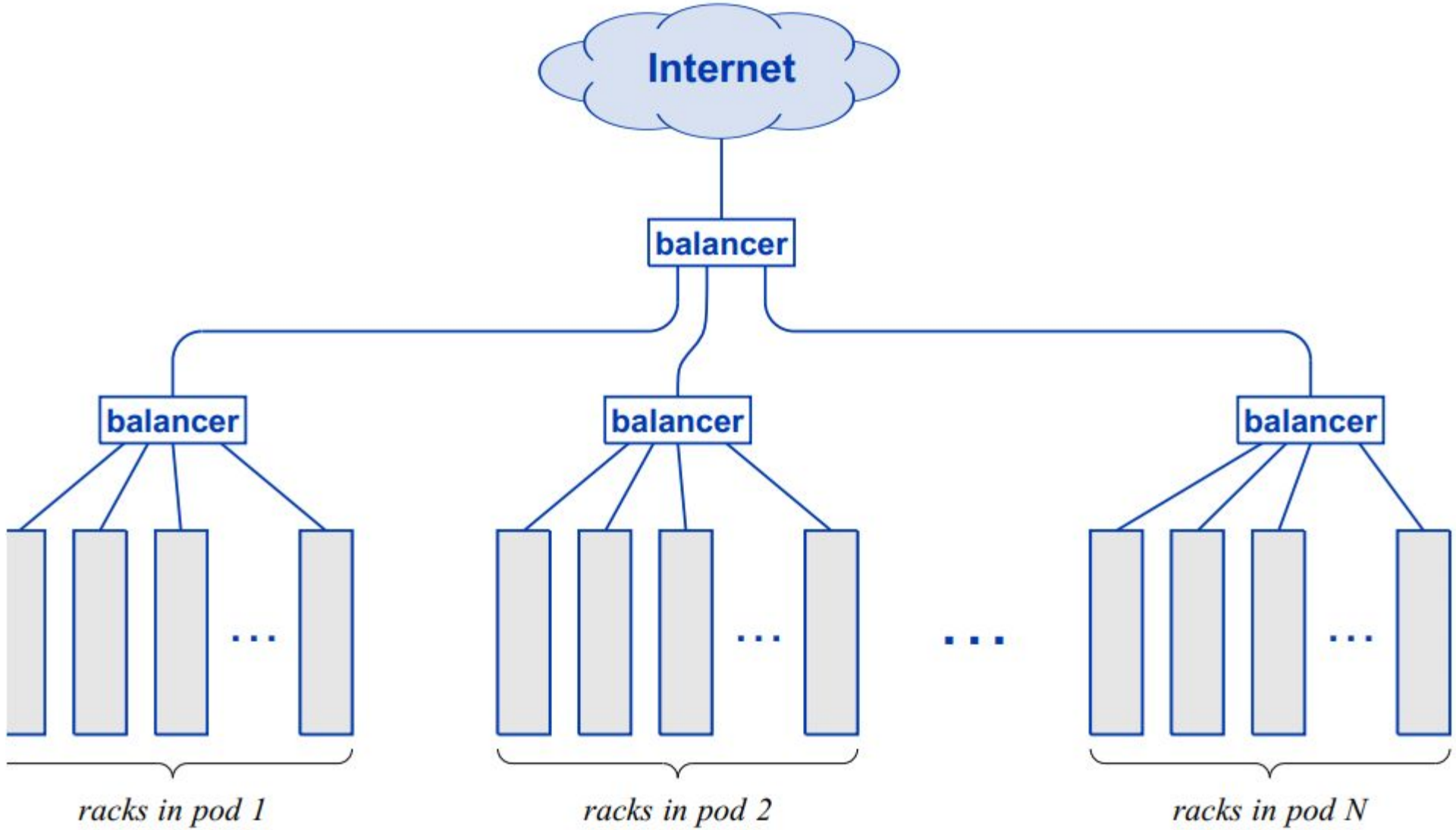


Figure 4.2 Illustration of a simplified hierarchy of load balancers used to achieve a large-scale web service.

Network Hierarchies, Capacity, and Fat Tree Designs

Arranging a data center network as a hierarchy has a disadvantage: links near the top of the hierarchy carry more traffic than links farther away.

The link between the Internet and the first load balancer carries 100% of the traffic that enters and leaves the data center.

At the next level down, however, the traffic is divided among the pods. If the data center contains *P* pods, each of the *P* links that connects the first load balancer and the load balancer for a pod only needs to carry *1/P* of the data.

If a pod contains *R* racks, a link between a rack and the load balancer for the pod only needs to carry *1 /RP* of the data.

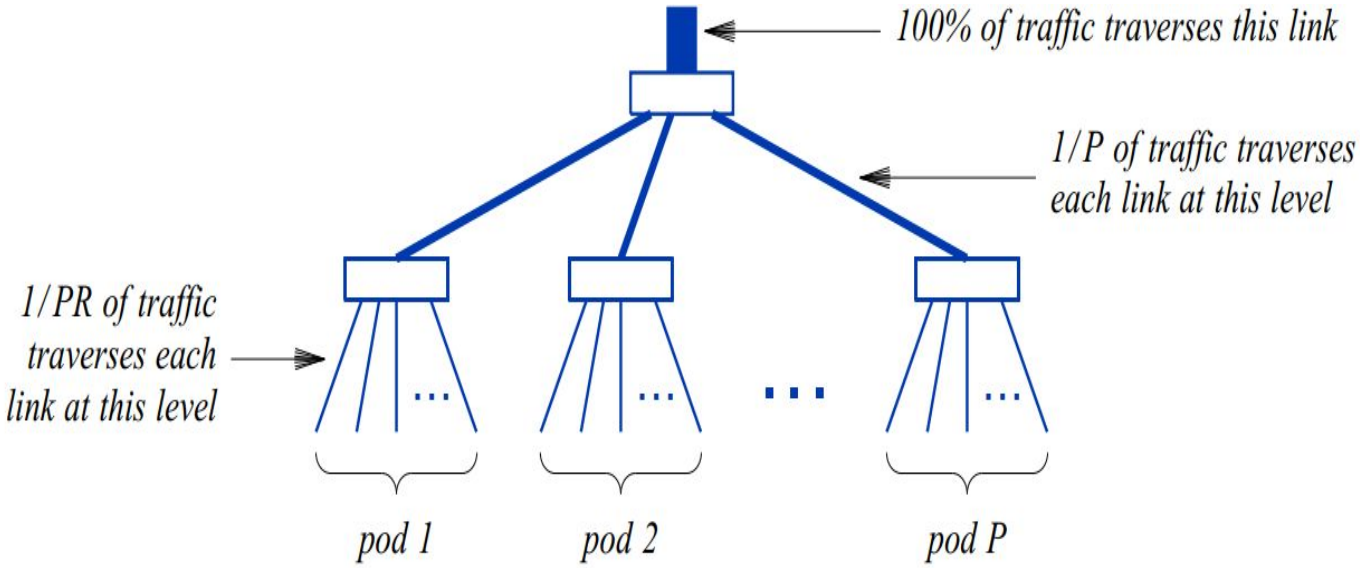


Figure 4.3 Internet traffic in a conceptual network hierarchy for a data center with *P* pods and *R* racks per pod.

High Capacity And Link Aggregation

For a data center that handles high volumes of traffic, links near the top of the hierarchy can require extremely high capacity.

A data center owner must face two constraints:

- Capacities available commercially
- Cost of high-capacity network hardware

Capacities available commercially: Network hardware is not available in arbitrary capacities. Only specific capacities have been standardized.

Ethernet hardware is available for 1, 10, 40, 100, and 400 Gigabits per second (*Gbps*), but not 6 Gbps.

Thus, a hierarchy must be designed carefully to use combinations of capacities that match commercially available hardware.

Cost of high-capacity network hardware.

A second factor that must be considered is the cost of network hardware. The cost of high-capacity network hardware is significantly higher than the cost of hardware with lower capacity, and the cost disparity is especially high for networks that cover long distances.

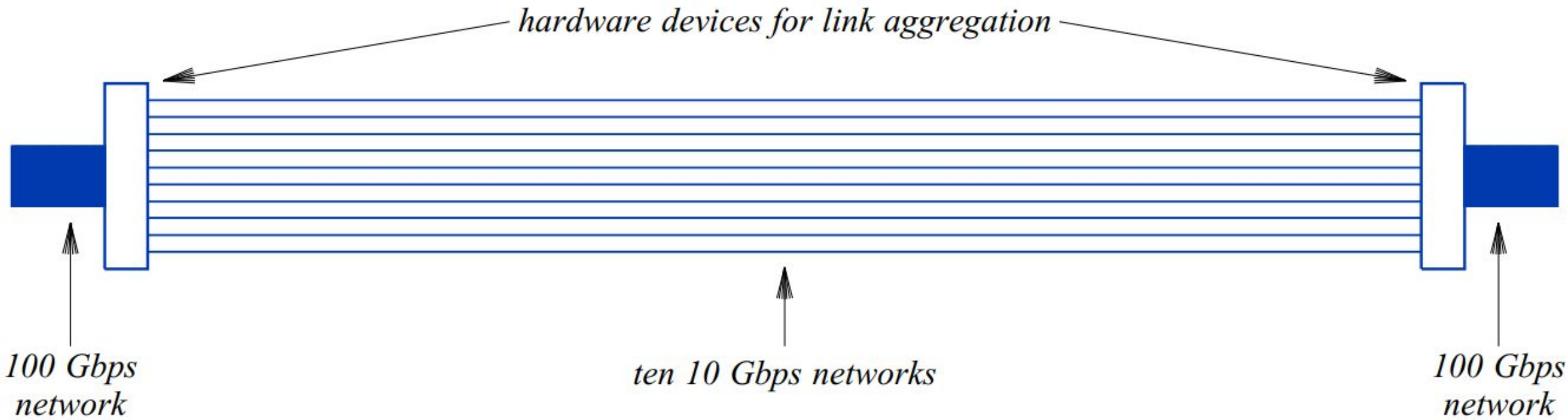


Figure 4.4 Illustration of two hardware devices aggregating ten networks operating at 10 Gbps to provide a 100 Gbps network.

A Leaf-Spine Network Design For East-West Traffic

- How can a data center network be designed that handles large volumes of east-west traffic without using a hierarchical design? The answer lies in parallelism and a form of load balancing. The specific approach used in data centers is known as a *leaf-spine network architecture*.
- In leaf-spine terminology, each Top-of-Rack switch is called a *leaf*.
- The data center owner adds an additional set of *spine* switches and connects each leaf switch to each spine switch.

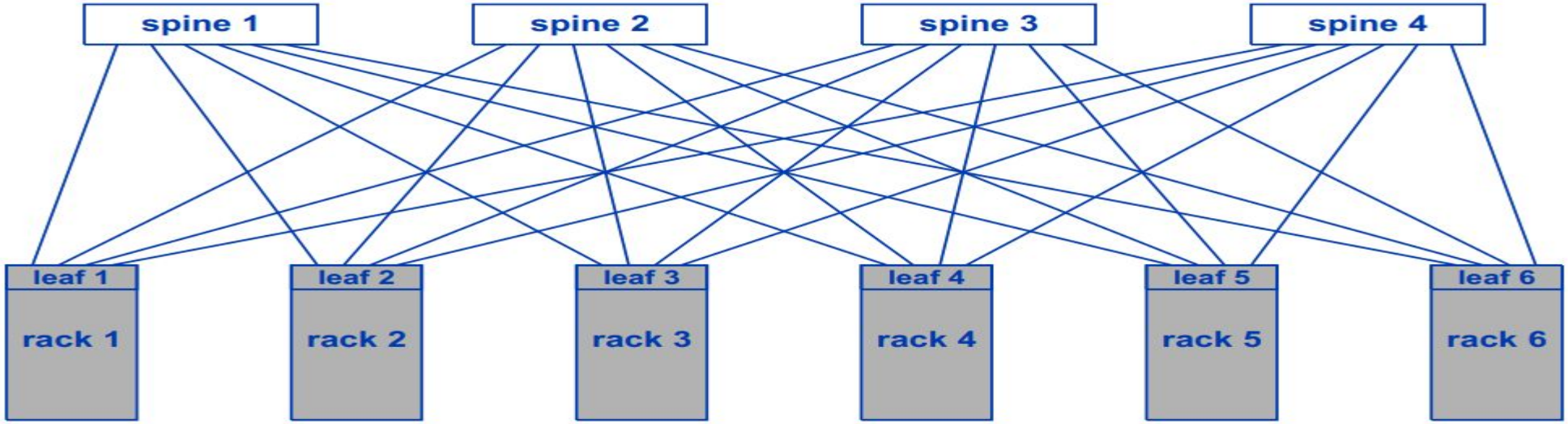


Figure 4.5 An example leaf-spine network with a leaf (i.e., a ToR switch) in each rack connecting to four spine switches.

The leaf-spine architecture offers two main advantages over a hierarchical design:

- Higher capacity for east-west traffic
- Redundant paths to handle failures

Higher capacity for east-west traffic: To understand the capacity, consider traffic traveling east-west from one rack to another. Because both the source and destination racks connect to all four spine switches, four independent paths exist between each pair of racks, one path through each spine switch.

Furthermore, a leaf switch equipped with *Equal Cost Multipath Routing* (*ECMP*) technology can be configured to divide traffic equally among the paths.

Consider how ECMP can be used in the network in Figure 4.5. As an example, suppose servers in rack 1 are sending data to servers in rack 6. ECMP means onefourth of the data will travel through spine 1, another fourth of the data will travel through spine 2, and so on. The ability to divide data among paths leads to an important property of the leaf-spine architecture: capacity can be increased incrementally.

Redundant paths to handle failures. To understand how leaf-spine accommodates failure, consider Figure 4.6 which illustrates the leaf-spine configuration in Figure 4.5 after spine switch 3 has failed.

As the figure shows, removing a single spine switch reduces capacity, but does not disrupt communication because three paths still remain between any pair of racks. Ofcourse, packet forwarding in the switches must be changed to accommodate the failure by dividing traffic among the spines that remain functional. Once the hardware in a leaf switch detects that a spine is no longer available, network management software performs the reconfiguration automatically, without requiring a human to handle the problem. Similarly, when the spine switch has been replaced and links become active, network management software will automatically reconfigure forwarding to use the spine again.

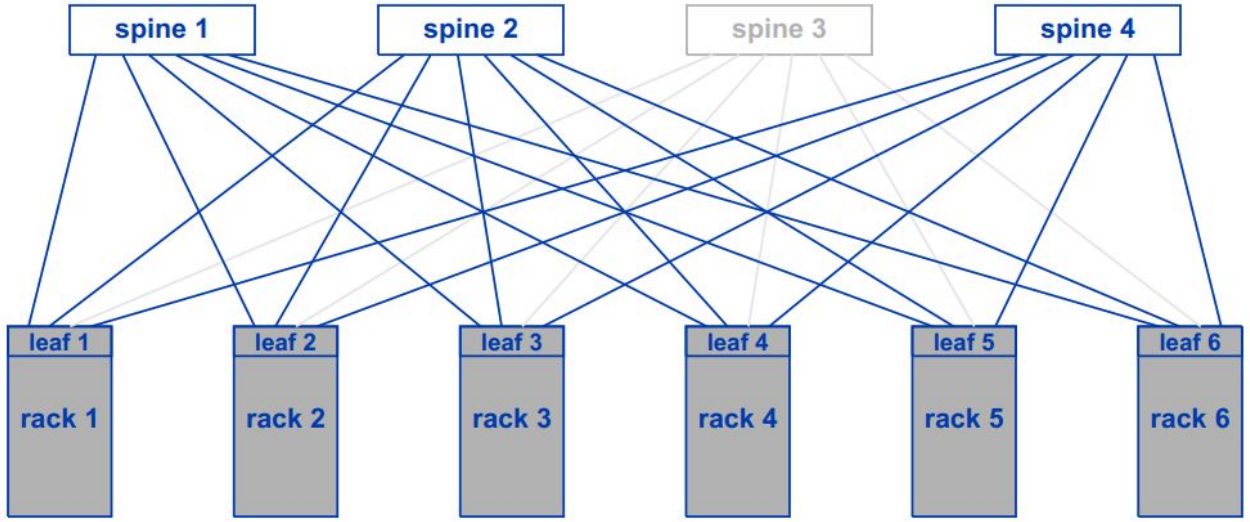


Figure 4.6 An example leaf-spine network in which the third spine switch has failed.

Scaling A Leaf-Spine Architecture With A Super Spine

Although it works for small numbers of racks, connecting all racks to each spine switch does not scale to tens of thousands of racks because the largest switches do not have tens of thousands of ports. To handle scaling, a data center uses a separate leafspine network to connect the racks in each pod. An additional level of switches known as a *super spine* is added to connect the spine switches to each pod. Each super spine switch connects to every spine switch. Figure 4.7 illustrates the arrangement.

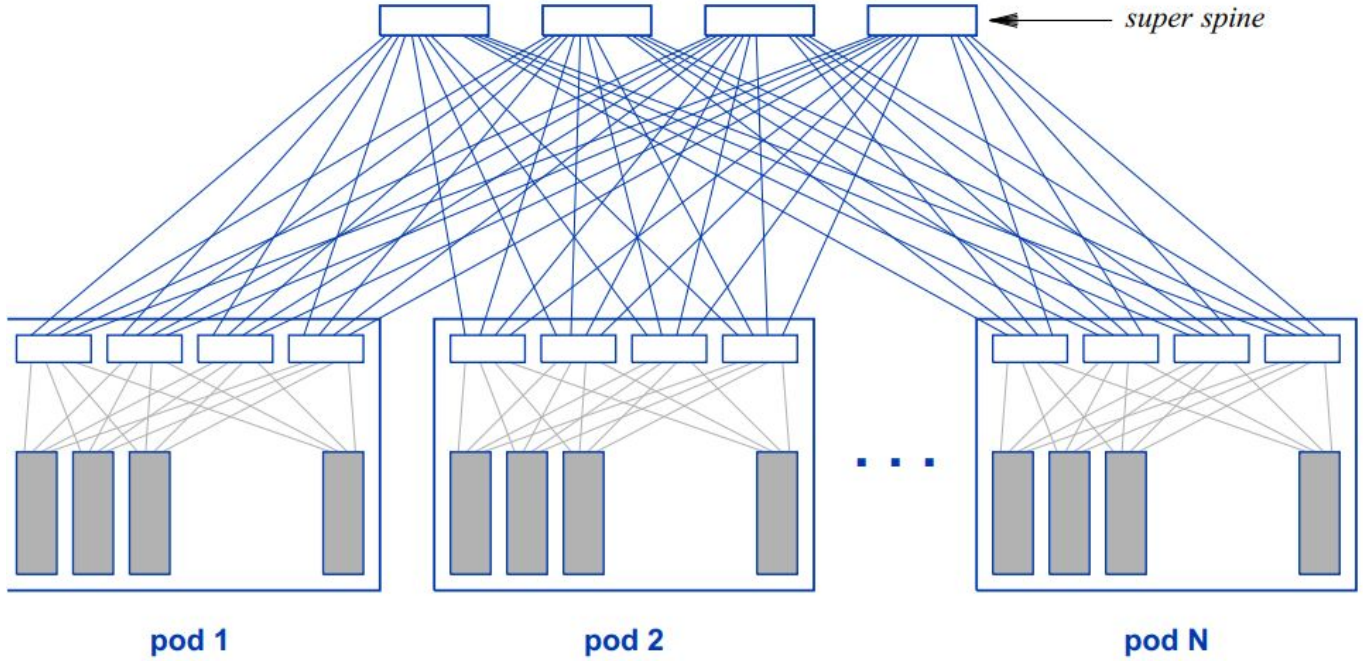


Figure 4.7 Illustration of a super spine configuration in which each pod has a leaf-spine network, and each spine switch connects to each of the super spine switches.

External Internet Connections

- How does a super spine architecture connect to the Internet? The incoming Internet connection may pass through a router, a hardware firewall, or other equipment, and eventually reaches a special switch. In fact, most data centers dedicate at least two switches to each connection in case one switch fails. The two external connection switches each connect to all the super spine switches as if they are spine switches (i.e., the two act like a miniature pod).
  The advantage of the super spine architecture should be clear: short paths across the data center for both internal and external traffic without requiring high-capacity links. When two servers in a rack communicate, packets flow from one to the other through the Top-of-Rack (i.e., leaf) switch. When two servers in racks in the same pod communicate, packets flow through the sender's leaf switch to a spine switch, and through the receiver's leaf switch to the receiver. Finally, when a server communicates with a distant server or an external site, traffic flows across the super spine. For example, when a server in one pod communicates with a server in another pod, packets flow from the sender through the sender's leaf switch to a spine switch in the sender's pod, to a super spine, to a spine switch in the receiver's pod, and through the receiver's leaf switch to the receiver.

Storage In A Data Center

Data center providers follow the same basic approach for storage facilities as they do for computational facilities: parallelism. That is, instead of a single, large disk storage mechanism, the physical storage facilities used in data centers consist of many inexpensive, commodity disks. Early data centers used electromechanical disks, sometimes called *spinning disks*. Modern data centers use *Solid State Disks* (*SSDs*).

Although the shift to solid state disk technology has increased reliability, the most significant change in data center storage arises from a change in the placement of storage equipment. To understand the change, recall that many early data centers were designed to support large-scale web service. Data centers used conventional PCs that each had their own disk. Data for the web site was replicated, and each server had a copy on the local disk. Only transactions, such as placing an order, required access to a central database.

The introduction of multi-tenant cloud systems made local storage on servers problematic for two reasons. First, a given server ran virtualized servers from multiple customers at the same time, and software was needed to limit the amount of storage each customer could use. Second, because disks were spread across the entire data center, replacing a failed disk required sending a staff member to the correct rack†.

To overcome the problems, data centers employ an approach for storage analogous to the approach used for computation: virtualized disks. In a small data center, the owner places all physical storage devices in a centralized location; in a larger data center, multiple locations are used. Software then creates virtualized disks for customers. When a virtualized server is created, the virtualized server is granted access to a corresponding virtualized disk. As software on a virtualized server accesses or stores data on its disk, requests travel across the data center network to the storage facility, and replies travel back over the network. Industry uses the term *block storage* to refer to virtualized disks because traditional disks provide an interface that transfers a fixedsize *block* of data.

From the point of view of software running on a server, *block storage* behaves exactly like a local disk except that transfers must pass across a network. Network communication introduces latency, which means that accessing block storage over a network can take longer than accessing a local disk. To avoid long delays, a data center owner can create multiple storage facilities and place each facility near a group of racks. For example, some data centers place a storage facility with each pod. The higher reliability of solid state disks has lowered failure rates, making replacement much less frequent.