Lecture 12

Network Computing II

Interconnection Networks, Grid

Ceng505 *Parallel Computing* at December 27, 2010

Dr. Cem Özdoğan
Computer Engineering Department
Çankaya University

Contents

Interconnection Networks
Ethernet
Switches
Myrinet Clos Network
The Quadrics Network
Grid Computing

Ethernet I

- Ethernet is a packet-switched LAN technology introduced by Xerox PARC in the early 1970s.

Ethernet I

- Ethernet is a packet-switched LAN technology introduced by Xerox PARC in the early 1970s.
- Ethernet was designed to be a shared bus technology where multiple hosts are connected to a shared communication medium.

Ethernet I

- Ethernet is a packet-switched LAN technology introduced by Xerox PARC in the early 1970s.
- Ethernet was designed to be a shared bus technology where multiple hosts are connected to a shared communication medium.
- All hosts connected to an Ethernet receive every transmission, making it possible to broadcast a packet to all hosts at the same time.

Ethernet I

- Ethernet is a packet-switched LAN technology introduced by Xerox PARC in the early 1970s.
- Ethernet was designed to be a shared bus technology where multiple hosts are connected to a shared communication medium.
- All hosts connected to an Ethernet receive every transmission, making it possible to broadcast a packet to all hosts at the same time.
- Ethernet uses a distributed access control scheme called Carrier Sense Multiple Access with Collision Detect (CSMA/CD).

Ethernet I

- Ethernet is a packet-switched LAN technology introduced by Xerox PARC in the early 1970s.
- Ethernet was designed to be a shared bus technology where multiple hosts are connected to a shared communication medium.
- All hosts connected to an Ethernet receive every transmission, making it possible to broadcast a packet to all hosts at the same time.
- Ethernet uses a distributed access control scheme called Carrier Sense Multiple Access with Collision Detect (CSMA/CD).
- Multiple machines can access an Ethernet at the same time.

Ethernet I

- Ethernet is a packet-switched LAN technology introduced by Xerox PARC in the early 1970s.
- Ethernet was designed to be a shared bus technology where multiple hosts are connected to a shared communication medium.
- All hosts connected to an Ethernet receive every transmission, making it possible to broadcast a packet to all hosts at the same time.
- Ethernet uses a distributed access control scheme called Carrier Sense Multiple Access with Collision Detect (CSMA/CD).
- Multiple machines can access an Ethernet at the same time.
- Each machine senses whether a carrier wave is present to determine whether the network is idle before it sends a packet.

Ethernet I

- Ethernet is a packet-switched LAN technology introduced by Xerox PARC in the early 1970s.
- Ethernet was designed to be a shared bus technology where multiple hosts are connected to a shared communication medium.
- All hosts connected to an Ethernet receive every transmission, making it possible to broadcast a packet to all hosts at the same time.
- Ethernet uses a distributed access control scheme called Carrier Sense Multiple Access with Collision Detect (CSMA/CD).
- Multiple machines can access an Ethernet at the same time.
- Each machine senses whether a carrier wave is present to determine whether the network is idle before it sends a packet.
- Only when the network is not busy sending another message can transmission start.

Ethernet II

- Each transmission is limited in duration and there is a minimum idle time between two consecutive transmissions by the same sender.

Ethernet II

- Each transmission is limited in duration and there is a minimum idle time between two consecutive transmissions by the same sender.
- In order to achieve an acceptable level of performance and to eliminate any potential bottleneck,

Ethernet II

- Each transmission is limited in duration and there is a minimum idle time between two consecutive transmissions by the same sender.
- In order to achieve an acceptable level of performance and to eliminate any potential bottleneck,
- there must be some **balance between the Ethernet and the processor speeds**.

Ethernet II

- Each transmission is limited in duration and there is a minimum idle time between two consecutive transmissions by the same sender.
- In order to achieve an acceptable level of performance and to eliminate any potential bottleneck,
- there must be some **balance between the Ethernet and the processor speeds**.
- The initial Beowulf prototype cluster in 1994 was built with DX4 processors and 10 Mbit/s Ethernet. The processors were too fast for this kind of Ethernet.

Ethernet II

- Each transmission is limited in duration and there is a minimum idle time between two consecutive transmissions by the same sender.
- In order to achieve an acceptable level of performance and to eliminate any potential bottleneck,
- there must be some **balance between the Ethernet and the processor speeds**.
- The initial Beowulf prototype cluster in 1994 was built with DX4 processors and 10 Mbit/s Ethernet. The processors were too fast for this kind of Ethernet.
- In late 1997, a good choice for a cluster system was sixteen 200 MHz P6 processors connected by Fast Ethernet.

Ethernet II

- Each transmission is limited in duration and there is a minimum idle time between two consecutive transmissions by the same sender.
- In order to achieve an acceptable level of performance and to eliminate any potential bottleneck,
- there must be some **balance between the Ethernet and the processor speeds**.
- The initial Beowulf prototype cluster in 1994 was built with DX4 processors and 10 Mbit/s Ethernet. The processors were too fast for this kind of Ethernet.
- In late 1997, a good choice for a cluster system was sixteen 200 MHz P6 processors connected by Fast Ethernet.
- The network configuration of a high-performance cluster is dependent on

Ethernet II

- Each transmission is limited in duration and there is a minimum idle time between two consecutive transmissions by the same sender.
- In order to achieve an acceptable level of performance and to eliminate any potential bottleneck,
- there must be some **balance between the Ethernet and the processor speeds**.
- The initial Beowulf prototype cluster in 1994 was built with DX4 processors and 10 Mbit/s Ethernet. The processors were too fast for this kind of Ethernet.
- In late 1997, a good choice for a cluster system was sixteen 200 MHz P6 processors connected by Fast Ethernet.
- The network configuration of a high-performance cluster is dependent on
  - the size of the cluster,

Ethernet II

- Each transmission is limited in duration and there is a minimum idle time between two consecutive transmissions by the same sender.
- In order to achieve an acceptable level of performance and to eliminate any potential bottleneck,
- there must be some **balance between the Ethernet and the processor speeds**.
- The initial Beowulf prototype cluster in 1994 was built with DX4 processors and 10 Mbit/s Ethernet. The processors were too fast for this kind of Ethernet.
- In late 1997, a good choice for a cluster system was sixteen 200 MHz P6 processors connected by Fast Ethernet.
- The network configuration of a high-performance cluster is dependent on
  - the size of the cluster,
  - the relationship between processor speed and network bandwidth

Ethernet II

- Each transmission is limited in duration and there is a minimum idle time between two consecutive transmissions by the same sender.
- In order to achieve an acceptable level of performance and to eliminate any potential bottleneck,
- there must be some **balance between the Ethernet and the processor speeds**.
- The initial Beowulf prototype cluster in 1994 was built with DX4 processors and 10 Mbit/s Ethernet. The processors were too fast for this kind of Ethernet.
- In late 1997, a good choice for a cluster system was sixteen 200 MHz P6 processors connected by Fast Ethernet.
- The network configuration of a high-performance cluster is dependent on
  - the size of the cluster,
  - the relationship between processor speed and network bandwidth
  - the price for each of the components.

Switches I

- An $n_1 \times n_2$ switch consists of n_1 input ports, n_2 output ports,

Switches I

- An $n_1 \times n_2$ switch consists of $\underline{n_1 \text{ input}}$ ports, $\underline{n_2 \text{ output}}$ ports,
- links connecting each input to every output, control logic to select a specific connection, and internal buffers.

Switches I

- An $n_1 \times n_2$ switch consists of n_1 input ports, n_2 output ports,
- links connecting each input to every output, control logic to select a specific connection, and internal buffers.
- Although n_1 and n_2 do not have to be equal, in practice and in most cases they have the same value, which is usually power of two.

Switches I

- An $n_1 \times n_2$ switch consists of n_1 input ports, n_2 output ports,
- links connecting each input to every output, control logic to select a specific connection, and internal buffers.
- Although n_1 and n_2 do not have to be equal, in practice and in most cases they have the same value, which is usually power of two.
- A switch is used to establish connections from the input ports to the output ports.

Switches I

- An $n_1 \times n_2$ switch consists of n_1 input ports, n_2 output ports,
- links connecting each input to every output, control logic to select a specific connection, and internal buffers.
- Although n_1 and n_2 do not have to be equal, in practice and in most cases they have the same value, which is usually power of two.
- A switch is used to establish connections from the input ports to the output ports.
- These connections may be

Switches I

- An $n_1 \times n_2$ switch consists of n_1 input ports, n_2 output ports,
- links connecting each input to every output, control logic to select a specific connection, and internal buffers.
- Although n_1 and n_2 do not have to be equal, in practice and in most cases they have the same value, which is usually power of two.
- A switch is used to establish connections from the input ports to the output ports.
- These connections may be
  - one-to-one, which represent point-to-point connections, or

Switches I

- An $n_1 \times n_2$ switch consists of n_1 input ports, n_2 output ports,
- links connecting each input to every output, control logic to select a specific connection, and internal buffers.
- Although n_1 and n_2 do not have to be equal, in practice and in most cases they have the same value, which is usually power of two.
- A switch is used to establish connections from the input ports to the output ports.
- These connections may be
  - one-to-one, which represent point-to-point connections, or
  - one-to-many, which represent multicast or broadcast.

Switches I

- An $n_1 \times n_2$ switch consists of n_1 input ports, n_2 output ports,
- links connecting each input to every output, control logic to select a specific connection, and internal buffers.
- Although n_1 and n_2 do not have to be equal, in practice and in most cases they have the same value, which is usually power of two.
- A switch is used to establish connections from the input ports to the output ports.
- These connections may be
  - one-to-one, which represent point-to-point connections, or
  - one-to-many, which represent multicast or broadcast.
- The case of many-to-one should cause conflicts at the output ports and therefore needs arbitration to resolve conflicts if allowed.

Switches I

- An $n_1 \times n_2$ switch consists of n_1 input ports, n_2 output ports,
- links connecting each input to every output, control logic to select a specific connection, and internal buffers.
- Although n_1 and n_2 do not have to be equal, in practice and in most cases they have the same value, which is usually power of two.
- A switch is used to establish connections from the input ports to the output ports.
- These connections may be
  - one-to-one, which represent point-to-point connections, or
  - one-to-many, which represent multicast or broadcast.
- The case of many-to-one should cause conflicts at the output ports and therefore needs arbitration to resolve conflicts if allowed.
- When only one-to-one connections are allowed, the switch is called *crossbar*.

Switches II

- An $n \times n$ crossbar switch can establish $n!$ connections (to allow only one-to-one connections,

Switches II

- An $n \times n$ crossbar switch can establish $n!$ connections (to allow only one-to-one connections,
  - the first input port should have n choices of output ports,

Switches II

- An $n \times n$ crossbar switch can establish $n!$ connections (to allow only one-to-one connections,
  - the first input port should have n choices of output ports,
  - the second input port will have $(n - 1)$ choices,

Switches II

- An $n \times n$ crossbar switch can establish $n!$ connections (to allow only one-to-one connections,
  - the first input port should have n choices of output ports,
  - the second input port will have $(n-1)$ choices,
  - the third input port will have $(n-2)$ choices, and so on.

Switches II

- An $n \times n$ crossbar switch can establish $n!$ connections (to allow only one-to-one connections,
  - the first input port should have n choices of output ports,
  - the second input port will have $(n-1)$ choices,
  - the third input port will have $(n-2)$ choices, and so on.
  - Thus, the number of one-to-one connections is $n * (n-1) * (n-2) \ldots * 2 * 1 = n!$

Switches II

- An $n \times n$ crossbar switch can establish $n!$ connections (to allow only one-to-one connections,
  - the first input port should have n choices of output ports,
  - the second input port will have $(n-1)$ choices,
  - the third input port will have $(n-2)$ choices, and so on.
  - Thus, the number of one-to-one connections is $n * (n-1) * (n-2) \ldots * 2 * 1 = n!$
- For example, a binary switch has two input(output) ports.

Switches II

- An $n \times n$ crossbar switch can establish $n!$ connections (to allow only one-to-one connections,
  - the first input port should have n choices of output ports,
  - the second input port will have $(n-1)$ choices,
  - the third input port will have $(n-2)$ choices, and so on.
  - Thus, the number of one-to-one connections is $n * (n-1) * (n-2) \ldots * 2 * 1 = n!$
- For example, a binary switch has two input(output) ports.
- The number of one-to-one connections in a binary switch is two (straight and crossed),

Switches II

- An $n \times n$ crossbar switch can establish $n!$ connections (to allow only one-to-one connections,
  - the first input port should have n choices of output ports,
  - the second input port will have $(n-1)$ choices,
  - the third input port will have $(n-2)$ choices, and so on.
  - Thus, the number of one-to-one connections is $n * (n-1) * (n-2) \ldots * 2 * 1 = n!$
- For example, a binary switch has two input(output) ports.
- The number of one-to-one connections in a binary switch is two (straight and crossed),
- while the number of all allowed connections is four (straight, crosses, lower broadcast, and upper broadcast).

Switches III

- Routing can be achieved using two mechanisms:

Switches III

- Routing can be achieved using two mechanisms:

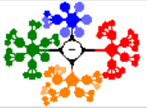
1. **Source-path**, the entire path to the destination is stored in the packet header at the source location.

Switches III

- Routing can be achieved using two mechanisms:

1 **Source-path**, the entire path to the destination is stored in the packet header at the source location.

- When a packet enters the switch, the outgoing port is determined from the header.



Switches III

- Routing can be achieved using two mechanisms:

1 **Source-path**, the entire path to the destination is stored in the packet header at the source location.

- When a packet enters the switch, the outgoing port is determined from the header.
- Used routing data is stripped from the header and routing information for the next switch is now in the front.

Switches III

- Routing can be achieved using two mechanisms:

1 **Source-path**, the entire path to the destination is stored in the packet header at the source location.

- When a packet enters the switch, the outgoing port is determined from the header.
- Used routing data is stripped from the header and routing information for the next switch is now in the front.

2 **Table-based routing**, the switch must have a complete routing table that determines the corresponding port for each destination.

Switches III

- Routing can be achieved using two mechanisms:

1. **Source-path**, the entire path to the destination is stored in the packet header at the source location.

- When a packet enters the switch, the outgoing port is determined from the header.
- Used routing data is stripped from the header and routing information for the next switch is now in the front.

2. **Table-based routing**, the switch must have a complete routing table that determines the corresponding port for each destination.

- When a packet enters the switch, a table lookup will determine the outgoing port.

Switches IV

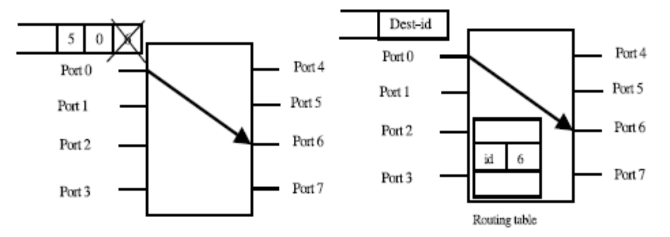


Figure: Source-path routing vs. table-based routing.

- Figure 1 illustrates the difference between source-path routing and table-based routing in the case when a packet enters an 8-port switch at port 0.

Switches IV

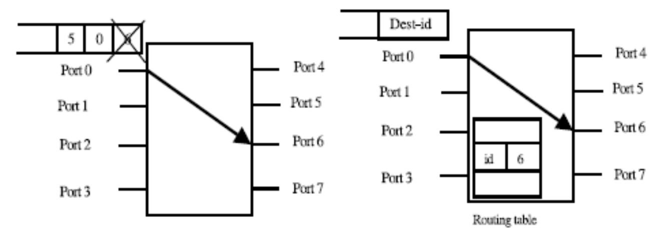


Figure: Source-path routing vs. table-based routing.

- Figure 1 illustrates the difference between source-path routing and table-based routing in the case when a packet enters an 8-port switch at port 0.
- In the source-path case, the header contains the entire path and the next port is port 6.

Switches IV

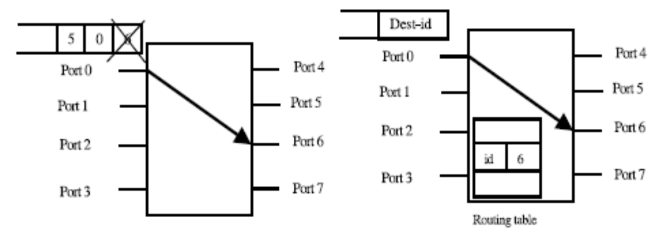


Figure: Source-path routing vs. table-based routing.

- Figure 1 illustrates the difference between source-path routing and table-based routing in the case when a packet enters an 8-port switch at port 0.
- In the source-path case, the header contains the entire path and the next port is port 6.
- In the table-based case, the destination address dest-id is looked up in the routing table and port 6 is followed.

Myrinet Clos Network I

- Myrinet is a high-performance, packet-communication and switching technology.

Myrinet Clos Network I

- Myrinet is a high-performance, packet-communication and switching technology.
- It was produced by Myricom as a high-performance alternative to conventional Ethernet networks.

Myrinet Clos Network I

- Myrinet is a high-performance, packet-communication and switching technology.
- It was produced by Myricom as a high-performance alternative to conventional Ethernet networks.
- Myrinet switches are multiple-port (4, 8, 12, 16) components that route a packet entering on an input channel of a port to the output channel of the port selected by the packet.

Myrinet Clos Network I

- Myrinet is a high-performance, packet-communication and switching technology.
- It was produced by Myricom as a high-performance alternative to conventional Ethernet networks.
- Myrinet switches are multiple-port (4, 8, 12, 16) components that route a packet entering on an input channel of a port to the output channel of the port selected by the packet.
- For an *n*-port switch, the ports are addressed $0, 1, 2, \ldots, n-1$.

Myrinet Clos Network I

- Myrinet is a high-performance, packet-communication and switching technology.
- It was produced by Myricom as a high-performance alternative to conventional Ethernet networks.
- Myrinet switches are multiple-port (4, 8, 12, 16) components that route a packet entering on an input channel of a port to the output channel of the port selected by the packet.
- For an *n*-port switch, the ports are addressed $0, 1, 2, \ldots, n - 1$.
- For any switching permutation, there may be as many packets traversing a switch concurrently as the switch has ports.

Myrinet Clos Network II

- These switches are implemented using two types of chips: crossbar-switch chips and the Myrinet-interface chip.

Myrinet Clos Network II

- These switches are implemented using two types of chips: crossbar-switch chips and the Myrinet-interface chip.
- The basic building block of the Myrinet-2000 network is a 16-port Myrinet crossbar switch, implemented on a single chip designated as Xbar16.

Myrinet Clos Network II

- These switches are implemented using two types of chips: crossbar-switch chips and the Myrinet-interface chip.
- The basic building block of the Myrinet-2000 network is a 16-port Myrinet crossbar switch, implemented on a single chip designated as Xbar16.
- It can be interconnected to build various topologies of varying sizes.

Myrinet Clos Network II

- These switches are implemented using two types of chips: crossbar-switch chips and the Myrinet-interface chip.
- The basic building block of the Myrinet-2000 network is a 16-port Myrinet crossbar switch, implemented on a single chip designated as Xbar16.
- It can be interconnected to build various topologies of varying sizes.
- The most common topology is the Clos network.

Myrinet Clos Network III

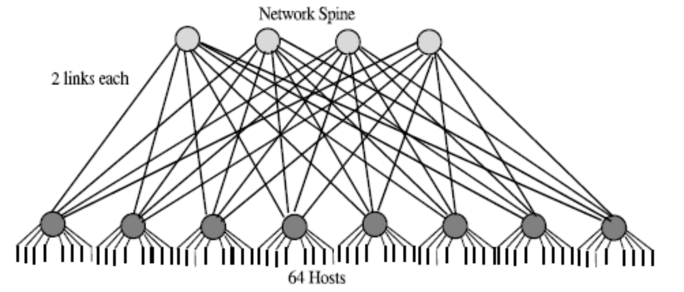


Figure: A 64-host Clos network using 16-port Myrinet switch (each line represents two links).

- A network of 64 hosts or fewer would require eight-port switches for the spine.

Myrinet Clos Network III

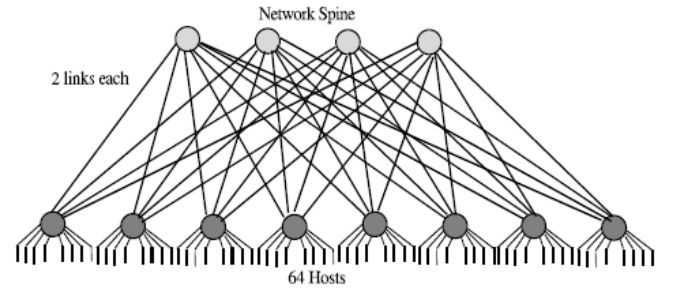


Figure: A 64-host Clos network using 16-port Myrinet switch (each line represents two links).

- A network of 64 hosts or fewer would require eight-port switches for the spine.
- In the figure, an Xbar16 switch can serve the purpose of two 8-port switches.

Myrinet Clos Network III

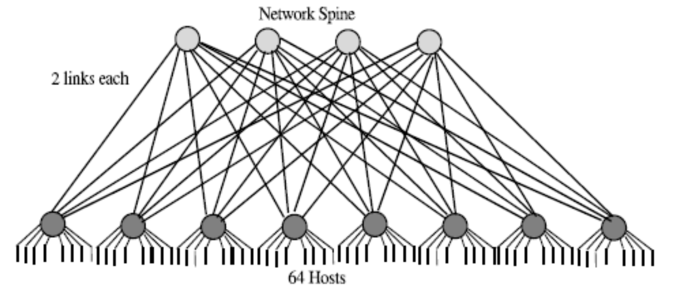


Figure: A 64-host Clos network using 16-port Myrinet switch (each line represents two links).

- A network of 64 hosts or fewer would require eight-port switches for the spine.
- In the figure, an Xbar16 switch can serve the purpose of two 8-port switches.
- The thick line connecting a spine switch to a leaf switch represents two links.

Myrinet Clos Network IV

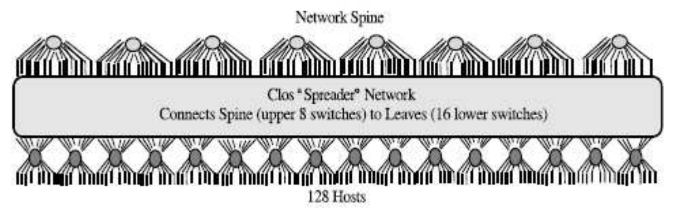


Figure: A 128-host Clos network using 16-port Myrinet switch, which includes 24 Xbar16s.

- Each Xbar16 switch is represented using a circle.

Myrinet Clos Network IV

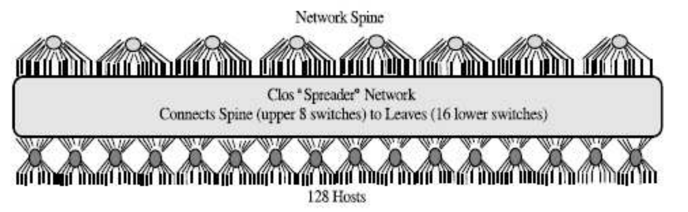


Figure: A 128-host Clos network using 16-port Myrinet switch, which includes 24 Xbar16s.

- Each Xbar16 switch is represented using a circle.
- The eight switches forming the upper row is the Clos network *spine*, which is connected through a Clos spreader network to the 16 *leaf* switches forming the lower row.

Myrinet Clos Network IV

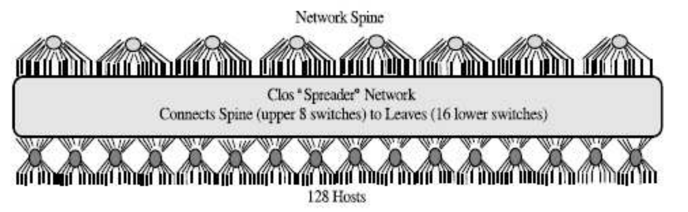


Figure: A 128-host Clos network using 16-port Myrinet switch, which includes 24 Xbar16s.

- Each Xbar16 switch is represented using a circle.
- The eight switches forming the upper row is the Clos network *spine*, which is connected through a Clos spreader network to the 16 *leaf* switches forming the lower row.
- The Clos network provides routes from any host to any other host.

Myrinet Clos Network IV

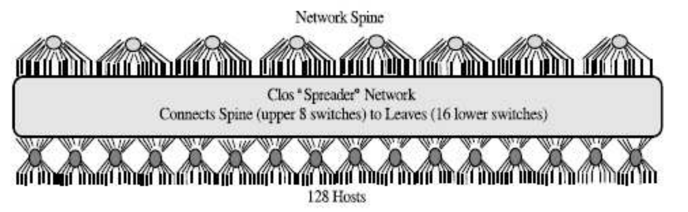


Figure: A 128-host Clos network using 16-port Myrinet switch, which includes 24 Xbar16s.

- Each Xbar16 switch is represented using a circle.
- The eight switches forming the upper row is the Clos network *spine*, which is connected through a Clos spreader network to the 16 *leaf* switches forming the lower row.
- The Clos network provides routes from any host to any other host.
- Routes between hosts connected to different Xbar16s traverse three Xbar16 switches.

Myrinet Clos Network V

- The routing of Myrinet packets is based on the source routing approach.

Myrinet Clos Network V

- The routing of Myrinet packets is based on the source routing approach.
- Each Myrinet packet has a variable length header with complete routing information.

Myrinet Clos Network V

- The routing of Myrinet packets is based on the source routing approach.
- Each Myrinet packet has a variable length header with complete routing information.
- When a packet enters a switch, the leading byte of the header determines the outgoing port before being stripped off the packet header.

Myrinet Clos Network V

- The routing of Myrinet packets is based on the source routing approach.
- Each Myrinet packet has a variable length header with complete routing information.
- When a packet enters a switch, the leading byte of the header determines the outgoing port before being stripped off the packet header.
- At the host interface, a control program is executed to perform source-route translation.

The Quadrics Network I

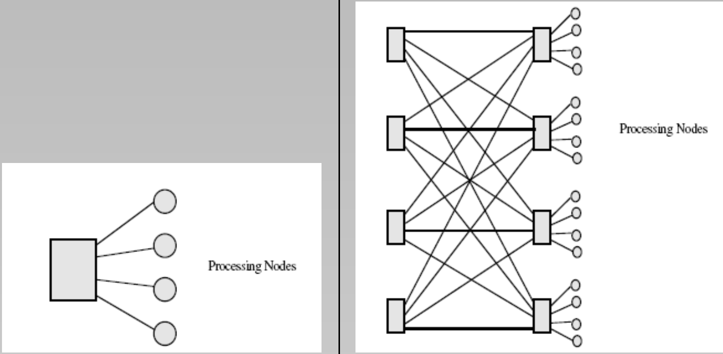


Figure: Quaternary fat tree of dimension 1 (left) and Elite switch of Quadrics networks (right).

- The Quadrics network (QsNet) consists of two hardware building blocks:

The Quadrics Network I

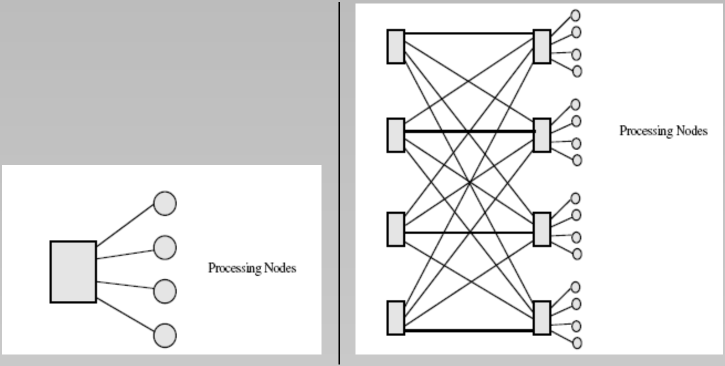
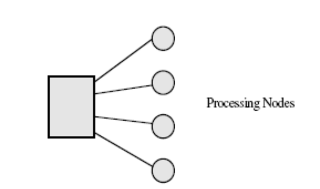


Figure: Quaternary fat tree of dimension 1 (left) and Elite switch of Quadrics networks (right).

- The Quadrics network (QsNet) consists of two hardware building blocks:
  1. a programmable network interface called Elan and

The Quadrics Network I



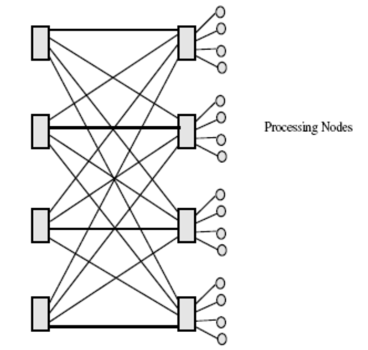


Figure: Quaternary fat tree of dimension 1 (left) and Elite switch of Quadrics networks (right).

- The Quadrics network (QsNet) consists of two hardware building blocks:
  1. a programmable network interface called Elan and
  2. a high-bandwidth, low-latency communication switch called Elite.

The Quadrics Network II

- The **Elan network interface** connects the Quadrics network to a processing node containing one or more CPUs.

The Quadrics Network II

- The **Elan network interface** connects the Quadrics network to a processing node containing one or more CPUs.
- QsNet connects Elite switches in a quaternary fat-tree topology.

The Quadrics Network II

- The **Elan network interface** connects the Quadrics network to a processing node containing one or more CPUs.
- QsNet connects Elite switches in a quaternary fat-tree topology.
- A quaternary fat tree of dimension n is composed of 4^n processing nodes and $n \times 4^{n-1}$ switches interconnected as a delta network.

The Quadrics Network II

- The **Elan network interface** connects the Quadrics network to a processing node containing one or more CPUs.
- QsNet connects Elite switches in a quaternary fat-tree topology.
- A quaternary fat tree of dimension n is composed of 4^n processing nodes and $n \times 4^{n-1}$ switches interconnected as a delta network.
- It can be recursively built by connecting four quaternary fat trees of dimension $n - 1$.

The Quadrics Network II

- The **Elan network interface** connects the Quadrics network to a processing node containing one or more CPUs.
- QsNet connects Elite switches in a quaternary fat-tree topology.
- A quaternary fat tree of dimension n is composed of 4^n processing nodes and $n \times 4^{n-1}$ switches interconnected as a delta network.
- It can be recursively built by connecting four quaternary fat trees of dimension $n - 1$.
- Figure 3left and -right shows quaternary fat trees of dimensions 1 and 2, respectively.

The Quadrics Network II

- The **Elan network interface** connects the Quadrics network to a processing node containing one or more CPUs.
- QsNet connects Elite switches in a quaternary fat-tree topology.
- A quaternary fat tree of dimension n is composed of 4^n processing nodes and $n \times 4^{n-1}$ switches interconnected as a delta network.
- It can be recursively built by connecting four quaternary fat trees of dimension $n - 1$.
- Figure 3left and -right shows quaternary fat trees of dimensions 1 and 2, respectively.
- When $n = 1$, the network consists of one switch and four processing nodes.

The Quadrics Network II

- The **Elan network interface** connects the Quadrics network to a processing node containing one or more CPUs.
- QsNet connects Elite switches in a quaternary fat-tree topology.
- A quaternary fat tree of dimension n is composed of 4^n processing nodes and $n \times 4^{n-1}$ switches interconnected as a delta network.
- It can be recursively built by connecting four quaternary fat trees of dimension $n - 1$.
- Figure 3left and -right shows quaternary fat trees of dimensions 1 and 2, respectively.
- When $n = 1$, the network consists of one switch and four processing nodes.
- When $n = 2$, the network consists of eight switches and 16 processing nodes.

The Quadrics Network III

- **Elite network**s are source routed.

The Quadrics Network III

- **Elite network**s are source routed.
- The Elan network interface attaches route information to the packet header before transmitting the packet into the network.

The Quadrics Network III

- **Elite network**s are source routed.
- The Elan network interface attaches route information to the packet header before transmitting the packet into the network.
- The route information is a sequence of Elite link tags.

The Quadrics Network III

- **Elite network**s are source routed.
- The Elan network interface attaches route information to the packet header before transmitting the packet into the network.
- The route information is a sequence of Elite link tags.
- As the packet moves inside the network, each switch removes the first route tag from the header and forwards the packet to the next switch in the route or the final destination.

The Quadrics Network III

- **Elite network**s are source routed.
- The Elan network interface attaches route information to the packet header before transmitting the packet into the network.
- The route information is a sequence of Elite link tags.
- As the packet moves inside the network, each switch removes the first route tag from the header and forwards the packet to the next switch in the route or the final destination.
- Packets are routed using wormhole routing flow control (each packet is divided into flow control digits (flits)).

The Quadrics Network III

- **Elite networks** are source routed.
- The Elan network interface attaches route information to the packet header before transmitting the packet into the network.
- The route information is a sequence of Elite link tags.
- As the packet moves inside the network, each switch removes the first route tag from the header and forwards the packet to the next switch in the route or the final destination.
- Packets are routed using wormhole routing flow control (each packet is divided into flow control digits (flits)).
- In QsNet, the size of each flit is 16 bits. Network nodes can send packets to multiple destinations using the network's broadcast capability.

Interconnection Networks I

Table: List of local area networks device bandwidths.

Interconnection Technology	Data Rate (bit/s)	Data Rate (byte/s)	Year
Ethernet (10BASE-X)	10 Mbit/s	1.25 MB/s	1990
Fast Ethernet (100BASE-X)	100 Mbit/s	12.5 MB/s	1995
FDDI	100 Mbit/s	12.5 MB/s	
Token Ring IEEE 802.5v	1 Gbit/s	125 MB/s	2001
Gigabit Ethernet (1000BASE-X)	1 Gbit/s	125 MB/s	1998
Myrinet 2000	2 Gbit/s	250 MB/s	
Infiniband SDR 1X[24]	2 Gbit/s	250 MB/s	
Quadrics QsNetI	3.6 Gbit/s	450 MB/s	
Infiniband DDR 1X[24]	4 Gbit/s	500 MB/s	
Infiniband QDR 1X[24]	8 Gbit/s	1 GB/s	
Infiniband SDR 4X[24]	8 Gbit/s	1 GB/s	

Interconnection Networks II

Table: List of local area networks device bandwidths, Contnd.

Interconnection Technology	Data Rate (bit/s)	Data Rate (byte/s)	Year
Quadrics QsNetII	8 Gbit/s	1 GB/s	
10 Gigabit Ethernet (10GBASE-X)	10 Gbit/s	1.25 GB/s	
Myri 10G	10 Gbit/s	1.25 GB/s	
Infiniband DDR 4X[24]	16 Gbit/s	2 GB/s	
Scalable Coherent Interface (SCI) Dual Channel SCI, x8 PCIe	20 Gbit/s	2.5 GB/s	
Infiniband SDR 12X[24]	24 Gbit/s	3 GB/s	
Infiniband QDR 4X[24]	32 Gbit/s	4 GB/s	
40 Gigabit Ethernet (40GBASE-X)	40 Gbit/s	5 GB/s	
Infiniband DDR 12X[24]	48 Gbit/s	6 GB/s	
Infiniband QDR 12X[24]	96 Gbit/s	12 GB/s	
100 Gigabit Ethernet (100GBASE-X)	100 Gbit/s	12.5 GB/s	

Grid Computing I

- While clusters are collections of computers tied together as a single system,

Grid Computing I

- While clusters are collections of computers tied together as a single system,
- grids consist of multiple systems that work together while maintaining their distinct identities.

Grid Computing I

- While clusters are collections of computers tied together as a single system,
- grids consist of multiple systems that work together while maintaining their distinct identities.
- Owing to the decentralized and heterogeneous nature of the grid,

Grid Computing I

- While clusters are collections of computers tied together as a single system,
- grids consist of multiple systems that work together while maintaining their distinct identities.
- Owing to the decentralized and heterogeneous nature of the grid,
- the middleware that glues the different components is more complicated compared with that of clusters.

Grid Computing I

- While clusters are collections of computers tied together as a single system,
- grids consist of multiple systems that work together while maintaining their distinct identities.
- Owing to the decentralized and heterogeneous nature of the grid,
- the middleware that glues the different components is more complicated compared with that of clusters.
- Resembling an electric power grid, the computing grid is expected to become a pervasive (spread throughout) computing infrastructure that supports large-scale and resource-intensive applications.

Grid Computing I

- While clusters are collections of computers tied together as a single system,
- grids consist of multiple systems that work together while maintaining their distinct identities.
- Owing to the decentralized and heterogeneous nature of the grid,
- the middleware that glues the different components is more complicated compared with that of clusters.
- Resembling an electric power grid, the computing grid is expected to become a pervasive (spread throughout) computing infrastructure that supports large-scale and resource-intensive applications.
- The significant increase in application complexity and the need for collaboration have made grids an attractive computing infrastructure.

Grid Computing II

- Thus, the need for the distributed grid infrastructure will continue to be an important resource.

Grid Computing II

- Thus, the need for the distributed grid infrastructure will continue to be an important resource.
- A user signing on at one location would view computers at other remote locations **as if they were part of the local system**.

Grid Computing II

- Thus, the need for the distributed grid infrastructure will continue to be an important resource.
- A user signing on at one location would view computers at other remote locations **as if they were part of the local system**.
- Grid computing works by polling the resources available,

Grid Computing II

- Thus, the need for the distributed grid infrastructure will continue to be an important resource.
- A user signing on at one location would view computers at other remote locations **as if they were part of the local system**.
- Grid computing works by polling the resources available,
- and then allocating them to individual tasks as the need arise.

Grid Computing II

- Thus, the need for the distributed grid infrastructure will continue to be an important resource.
- A user signing on at one location would view computers at other remote locations **as if they were part of the local system**.
- Grid computing works by polling the resources available,
- and then allocating them to individual tasks as the need arise.
- Resources are returned to the pool upon completion of the task.

Grid Computing II

- Thus, the need for the distributed grid infrastructure will continue to be an important resource.
- A user signing on at one location would view computers at other remote locations **as if they were part of the local system**.
- Grid computing works by polling the resources available,
- and then allocating them to individual tasks as the need arise.
- Resources are returned to the pool upon completion of the task.
- Grid gives an illusion of a big virtual computer capable of carrying out enormous tasks.

Grid Computing III

- Support of grids requires innovative solutions to a number of challenging issues including:

Grid Computing III

- Support of grids requires innovative solutions to a number of challenging issues including:
  - resource management,

Grid Computing III

- Support of grids requires innovative solutions to a number of challenging issues including:
  - resource management,
  - resource monitoring,

Grid Computing III

- Support of grids requires innovative solutions to a number of challenging issues including:
  - resource management,
  - resource monitoring,
  - interoperability,

Grid Computing III

- Support of grids requires innovative solutions to a number of challenging issues including:
  - resource management,
  - resource monitoring,
  - interoperability,
  - security,

Grid Computing III

- Support of grids requires innovative solutions to a number of challenging issues including:
  - resource management,
  - resource monitoring,
  - interoperability,
  - security,
  - billing and accounting,

Grid Computing III

- Support of grids requires innovative solutions to a number of challenging issues including:
  - resource management,
  - resource monitoring,
  - interoperability,
  - security,
  - billing and accounting,
  - communication, and

Grid Computing III

- Support of grids requires innovative solutions to a number of challenging issues including:
  - resource management,
  - resource monitoring,
  - interoperability,
  - security,
  - billing and accounting,
  - communication, and
  - performance.

Grid Computing IV

- There are several examples of grid platforms and tools such as Globus and TeraGrid.

Grid Computing IV

- There are several examples of grid platforms and tools such as Globus and TeraGrid.
  - The **Globus Toolkit** is an enabling technology for the grid.

Grid Computing IV

- There are several examples of grid platforms and tools such as Globus and TeraGrid.
  - The **Globus Toolkit** is an enabling technology for the grid.
  - The toolkit includes software services and libraries for resource monitoring, discovery, and management, plus security and file management.

Grid Computing IV

- There are several examples of grid platforms and tools such as Globus and TeraGrid.
  - The **Globus Toolkit** is an enabling technology for the grid.
  - The toolkit includes software services and libraries for resource monitoring, discovery, and management, plus security and file management.
  - It also includes software for communication, fault detection, and portability.

Grid Computing IV

- There are several examples of grid platforms and tools such as Globus and TeraGrid.
  - The **Globus Toolkit** is an enabling technology for the grid.
  - The toolkit includes software services and libraries for resource monitoring, discovery, and management, plus security and file management.
  - It also includes software for communication, fault detection, and portability.
  - The Globus Toolkit has grown through an open-source strategy. Version 1.0 was introduced in 1998 followed by the 2.0 release in 2002. The latest 3.0 version is based on new open-standard Grid services.

Grid Computing V

- **TeraGrid** is a large high-performance computing project headed by the National Center for Supercomputing Applications at the University of Illinois at Urbana-Champaign.

Grid Computing V

- **TeraGrid** is a large high-performance computing project headed by the National Center for Supercomputing Applications at the University of Illinois at Urbana-Champaign.
- The TeraGrid uses thousands of Intel Itanium 2 processors located at four sites in the United States.

Grid Computing V

- **TeraGrid** is a large high-performance computing project headed by the National Center for Supercomputing Applications at the University of Illinois at Urbana-Champaign.
- The TeraGrid uses thousands of Intel Itanium 2 processors located at four sites in the United States.
- The TeraGrid is an effort to build and deploy the world's largest, fastest distributed infrastructure for open scientific research.

Grid Computing V

- **TeraGrid** is a large high-performance computing project headed by the National Center for Supercomputing Applications at the University of Illinois at Urbana-Champaign.
- The TeraGrid uses thousands of Intel Itanium 2 processors located at four sites in the United States.
- The TeraGrid is an effort to build and deploy the world's largest, fastest distributed infrastructure for open scientific research.
- The TeraGrid is expected to include 20 teraflops of computing power, facilities capable of managing and storing nearly 1 petabyte of data, high-resolution visualization environments, and toolkits for grid computing.

Grid Computing V

- **TeraGrid** is a large high-performance computing project headed by the National Center for Supercomputing Applications at the University of Illinois at Urbana-Champaign.
- The TeraGrid uses thousands of Intel Itanium 2 processors located at four sites in the United States.
- The TeraGrid is an effort to build and deploy the world's largest, fastest distributed infrastructure for open scientific research.
- The TeraGrid is expected to include 20 teraflops of computing power, facilities capable of managing and storing nearly 1 petabyte of data, high-resolution visualization environments, and toolkits for grid computing.
- These components will be tightly integrated and connected through a network that will operate at 40 gigabits per second.