Network Technology 2 –

Lennart Franked
email:lennart.franked@miun.se
Tel:060-148683

2013-03-25

Introduction

Network Design - Local Networks

What is your network layout?

Good design?

How do you know if you have a well designed network?

- When you already know how to expand your network with for example another building, an extra WAN-link et cetera.
- When an addition to the network is done, only the directly attached devices will be affected.
- The network size should be able to get twice or three times the size without having to redesign the network.
- When troubleshooting you do not have to think about complex protocols and how they interact with eachother.

–Dr Peter Welcher

Good design?

How do you know if you have a well designed network?

- When you already know how to expand your network with for example another building, an extra WAN-link et cetera.
- When an addition to the network is done, only the directly attached devices will be affected.
- The network size should be able to get twice or three times the size without having to redesign the network.
- When troubleshooting you do not have to think about complex protocols and how they interact with eachother.

–Dr Peter Welcher

Good design?

How do you know if you have a well designed network?

- When you already know how to expand your network with for example another building, an extra WAN-link et cetera.
- When an addition to the network is done, only the directly attached devices will be affected.
- The network size should be able to get twice or three times the size without having to redesign the network.
- When troubleshooting you do not have to think about complex protocols and how they interact with eachother.

–Dr Peter Welcher

Good design?

How do you know if you have a well designed network?

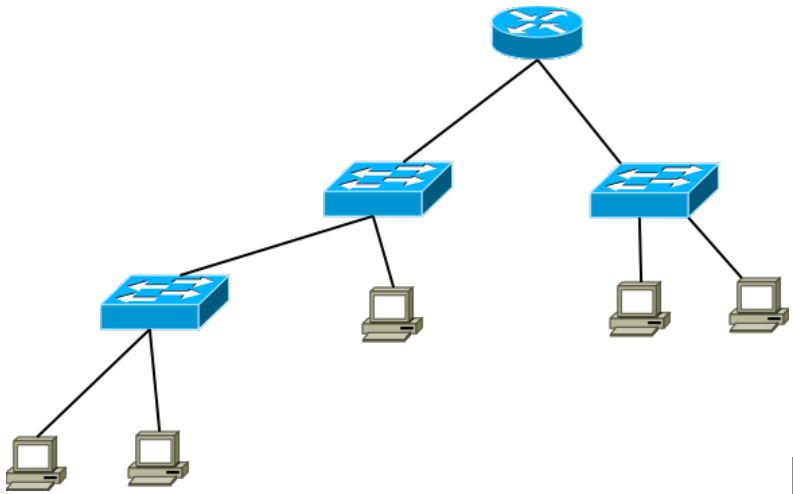
- When you already know how to expand your network with for example another building, an extra WAN-link et cetera.
- When an addition to the network is done, only the directly attached devices will be affected.
- The network size should be able to get twice or three times the size without having to redesign the network.
- When troubleshooting you do not have to think about complex protocols and how they interact with eachother.

–Dr Peter Welcher

Flat design

What characterizes a flat network topology?

- One broadcast domain.
- Every network device have the same function.
- Often no redundancy.



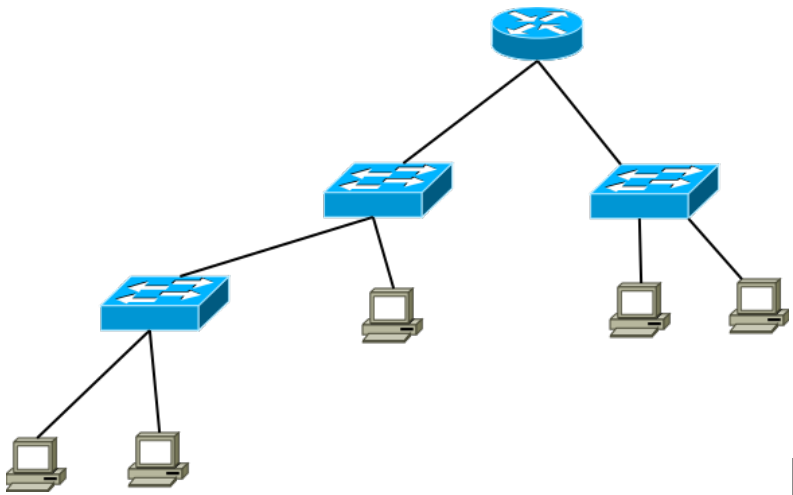
[6]

Figure 1 : Example of a flat topology

Flat design

What characterizes a flat network topology?

- One broadcast domain.
- Every network device have the same function.
- Often no redundancy.



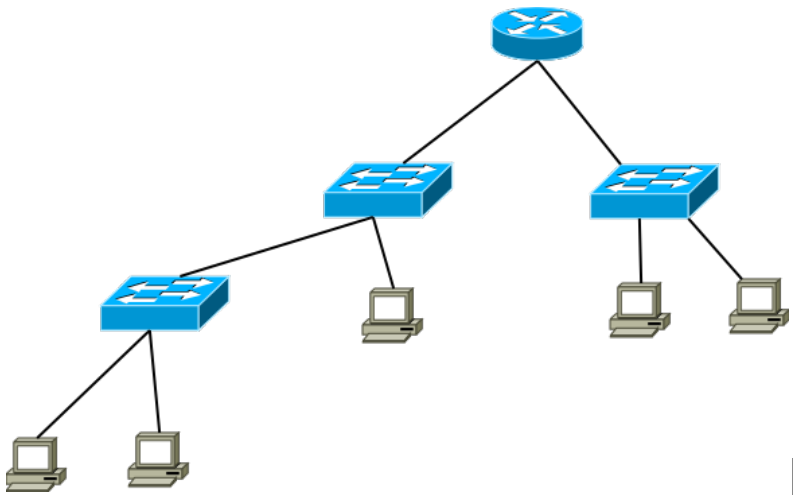
[6]

Figure 1 : Example of a flat topology

Flat design

What characterizes a flat network topology?

- One broadcast domain.
- Every network device have the same function.
- Often no redundancy.



[6]

Figure 1 : Example of a flat topology

Pro et Contra - flat design

Advantages with a flat design:

- Simple.
- Optimal for smaller networks.

Disadvantages:

- Not scalable.
- Difficult troubleshooting.

Pro et Contra - flat design

Advantages with a flat design:

- Simple.
- Optimal for smaller networks.

Disadvantages:

- Not scalable.
- Difficult troubleshooting.

Pro et Contra - flat design

Advantages with a flat design:

- Simple.
- Optimal for smaller networks.

Disadvantages:

- Not scalable.
- Difficult troubleshooting.

Pro et Contra - flat design

Advantages with a flat design:

- Simple.
- Optimal for smaller networks.

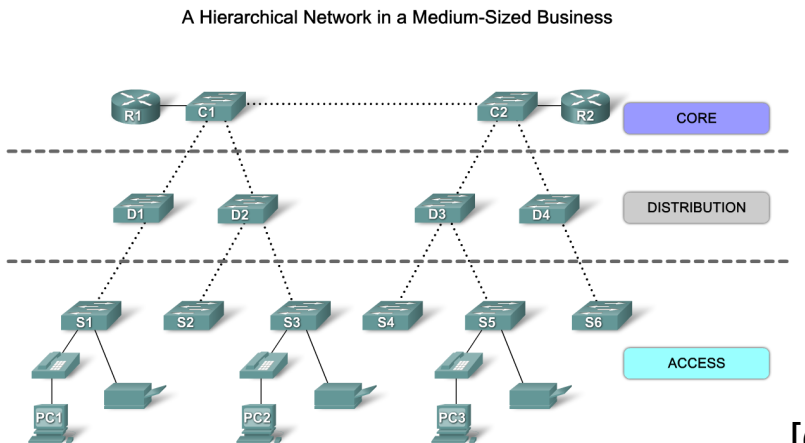
Disadvantages:

- Not scalable.
- Difficult troubleshooting.

Hierarchical design

What characterizes an hierarchical network topology?

- Multiple broadcast domains.
- Divided into layers (Core, Distribution, Access).
- A hierarchy.



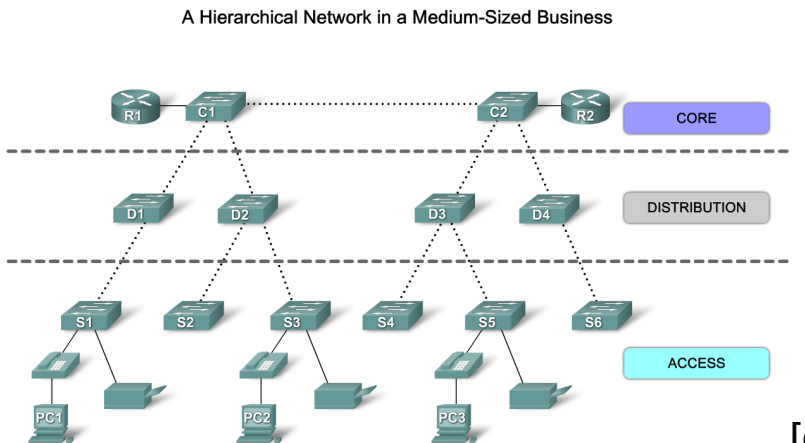
[6]

Figure 2 : Example of an hierarchical topology

Hierarchical design

What characterizes an hierarchical network topology?

- Multiple broadcast domains.
- Divided into layers (Core, Distribution, Access).
- A hierarchy.



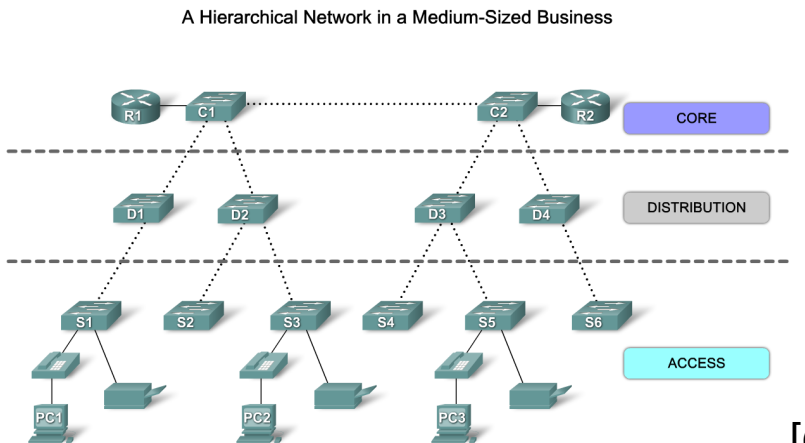
[6]

Figure 2 : Example of an hierarchical topology

Hierarchical design

What characterizes an hierarchical network topology?

- Multiple broadcast domains.
- Divided into layers (Core, Distribution, Access).
- A hierarchy.



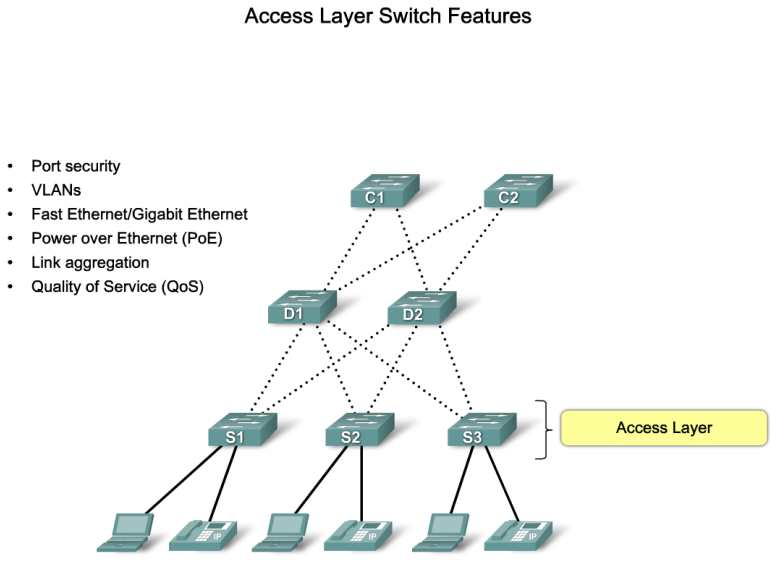
[6]

Figure 2 : Example of an hierarchical topology

Access layer

Access layers task is to provide the end nodes with a connection to the network. Due to this, the access layer must support a variety of features.

- Port Security, limit access to the network.
- VLAN, for separating traffic, for example VoIP and data traffic.
- PoE, Power over Ethernet.
- Link aggregation.
- QoS



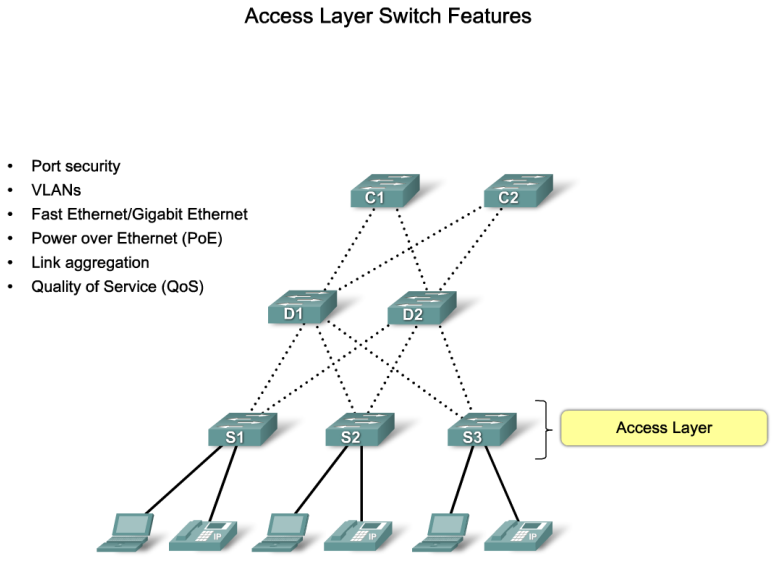
[6]

Figure 3 : Access layer

Access layer

Access layers task is to provide the end nodes with a connection to the network. Due to this, the access layer must support a variety of features.

- Port Security, limit access to the network.
- VLAN, for separating traffic, for example VoIP and data traffic.
- PoE, Power over Ethernet.
- Link aggregation.
- QoS



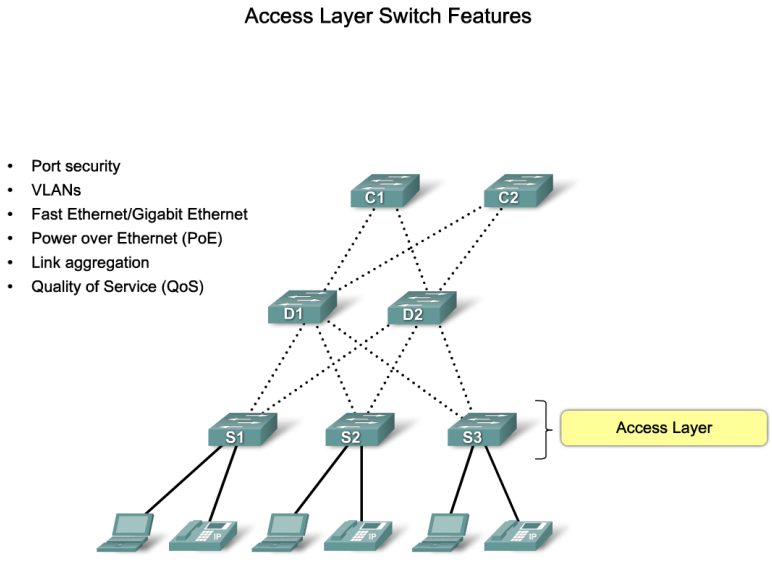
[6]

Figure 3 : Access layer

Access layer

Access layers task is to provide the end nodes with a connection to the network. Due to this, the access layer must support a variety of features.

- Port Security, limit access to the network.
- VLAN, for separating traffic, for example VoIP and data traffic.
- PoE, Power over Ethernet.
- Link aggregation.
- QoS



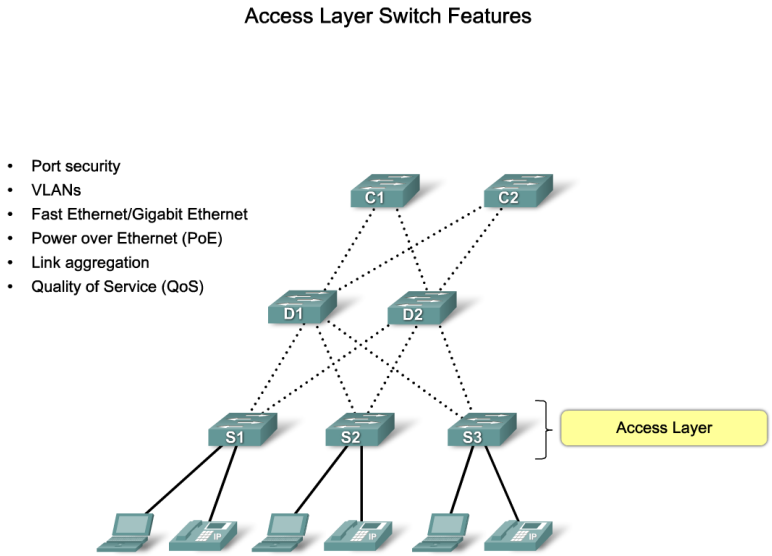
[6]

Figure 3 : Access layer

Access layer

Access layers task is to provide the end nodes with a connection to the network. Due to this, the access layer must support a variety of features.

- Port Security, limit access to the network.
- VLAN, for separating traffic, for example VoIP and data traffic.
- PoE, Power over Ethernet.
- Link aggregation.
- QoS



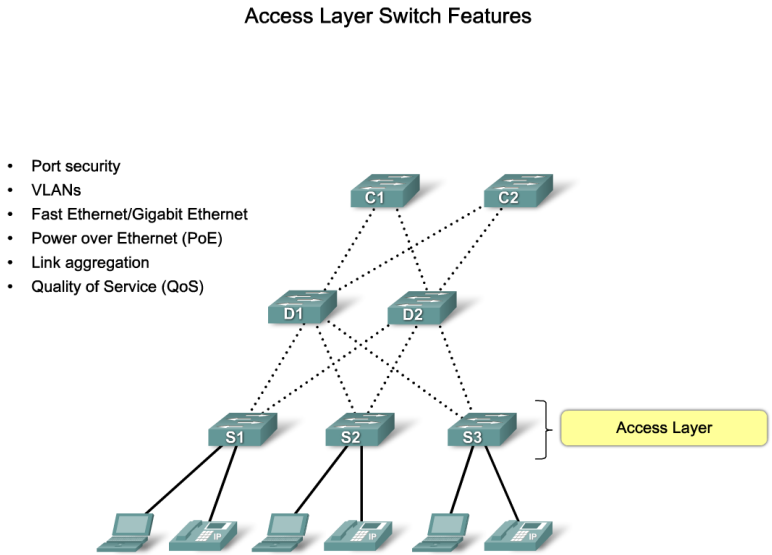
[6]

Figure 3 : Access layer

Access layer

Access layers task is to provide the end nodes with a connection to the network. Due to this, the access layer must support a variety of features.

- Port Security, limit access to the network.
- VLAN, for separating traffic, for example VoIP and data traffic.
- PoE, Power over Ethernet.
- Link aggregation.
- QoS



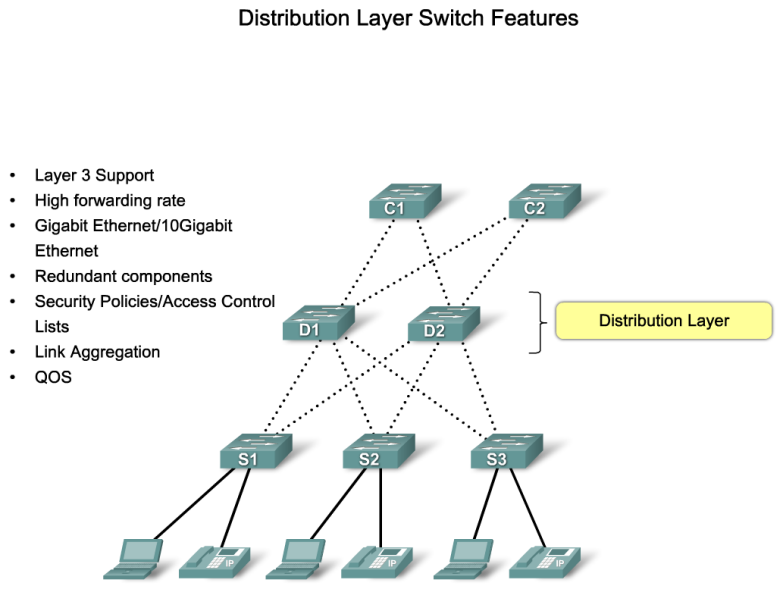
[6]

Figure 3 : Access layer

Distribution layer

Distributions layers function is to aggregate traffic from the access layer, and if needed forward the data up to the core layer.
The following features should be supported in the distribution layer:

- VLAN.
- Layer 3 switching.
  - Inter-VLAN routing.
  - Access Control Lists.
- High speed interfaces.
- Redundancy.
- link aggregation towards both access layer and core layer.
- QoS.



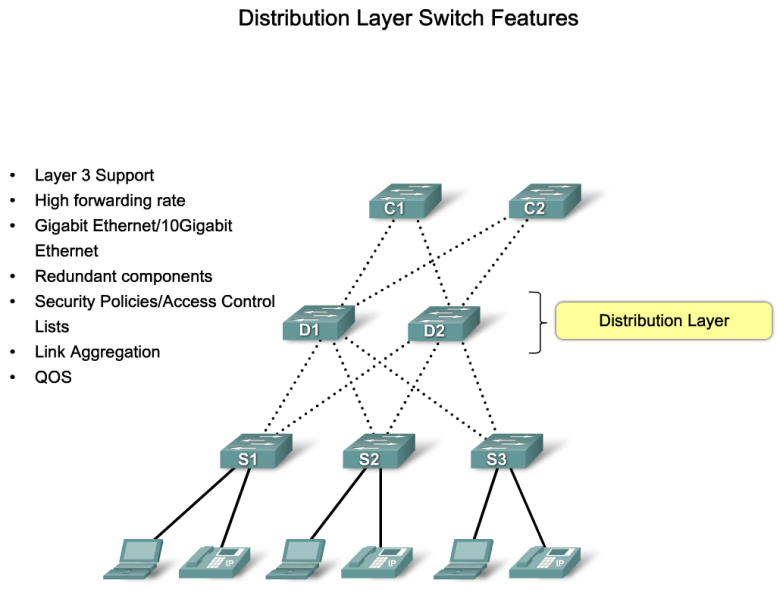
[6]

Figure 4 : Distribution layer

Distribution layer

Distributions layers function is to aggregate traffic from the access layer, and if needed forward the data up to the core layer.
The following features should be supported in the distribution layer:

- VLAN.
- Layer 3 switching.
  - Inter-VLAN routing.
  - Access Control Lists.
- High speed interfaces.
- Redundancy.
- link aggregation towards both access layer and core layer.
- QoS.



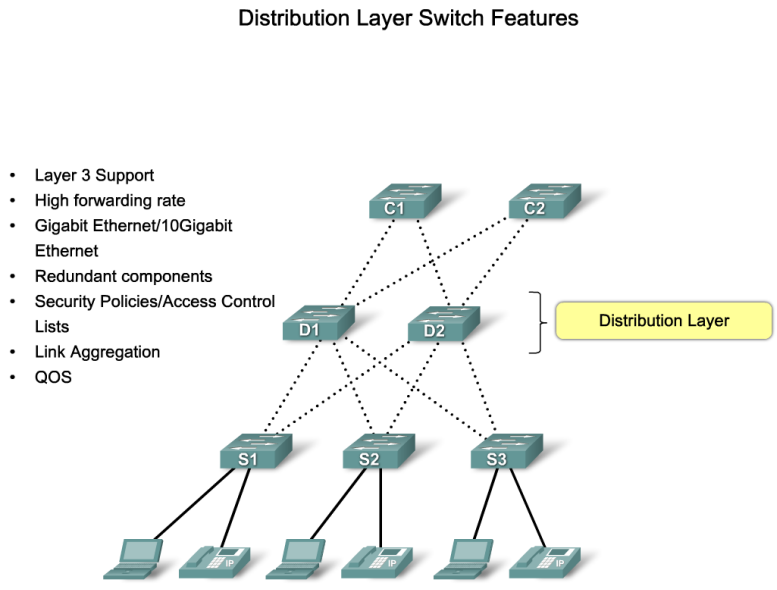
[6]

Figure 4 : Distribution layer

Distribution layer

Distributions layers function is to aggregate traffic from the access layer, and if needed forward the data up to the core layer.
The following features should be supported in the distribution layer:

- VLAN.
- Layer 3 switching.
  - Inter-VLAN routing.
  - Access Control Lists.
- High speed interfaces.
- Redundancy.
- link aggregation towards both access layer and core layer.
- QoS.



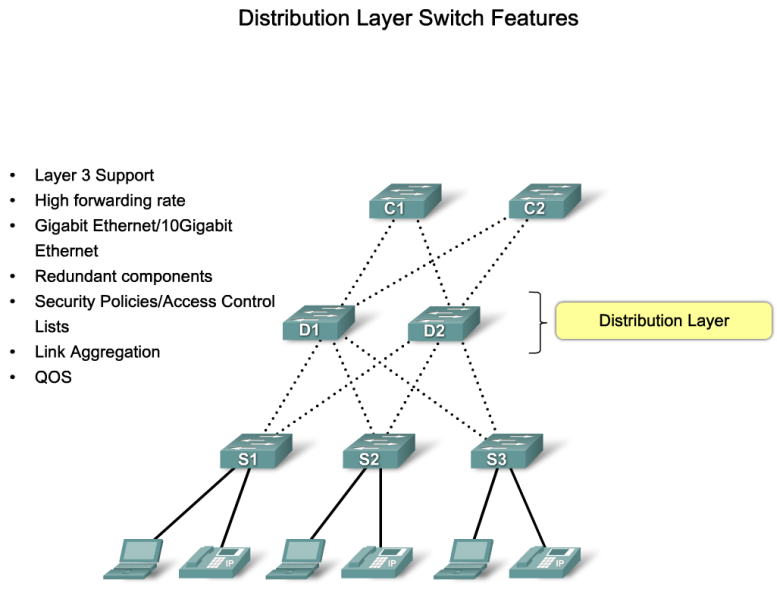
[6]

Figure 4 : Distribution layer

Distribution layer

Distributions layers function is to aggregate traffic from the access layer, and if needed forward the data up to the core layer.
The following features should be supported in the distribution layer:

- VLAN.
- Layer 3 switching.
  - Inter-VLAN routing.
  - Access Control Lists.
- High speed interfaces.
- Redundancy.
- link aggregation towards both access layer and core layer.
- QoS.



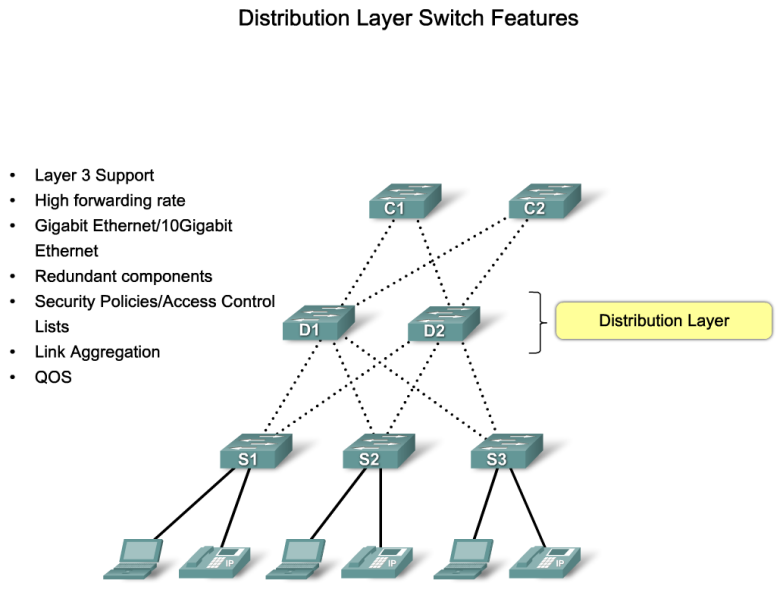
[6]

Figure 4 : Distribution layer

Distribution layer

Distributions layers function is to aggregate traffic from the access layer, and if needed forward the data up to the core layer.
The following features should be supported in the distribution layer:

- VLAN.
- Layer 3 switching.
  - Inter-VLAN routing.
  - Access Control Lists.
- High speed interfaces.
- Redundancy.
- link aggregation towards both access layer and core layer.
- QoS.



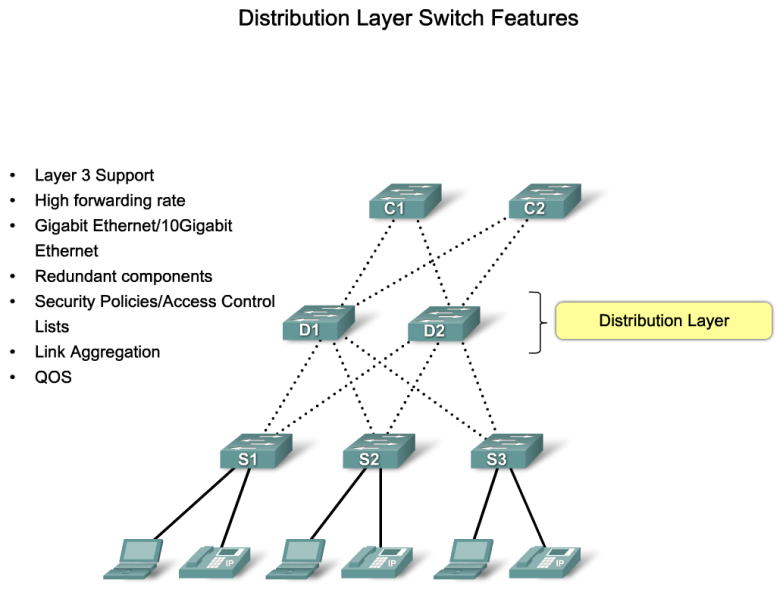
[6]

Figure 4 : Distribution layer

Distribution layer

Distributions layers function is to aggregate traffic from the access layer, and if needed forward the data up to the core layer.
The following features should be supported in the distribution layer:

- VLAN.
- Layer 3 switching.
  - Inter-VLAN routing.
  - Access Control Lists.
- High speed interfaces.
- Redundancy.
- link aggregation towards both access layer and core layer.
- QoS.



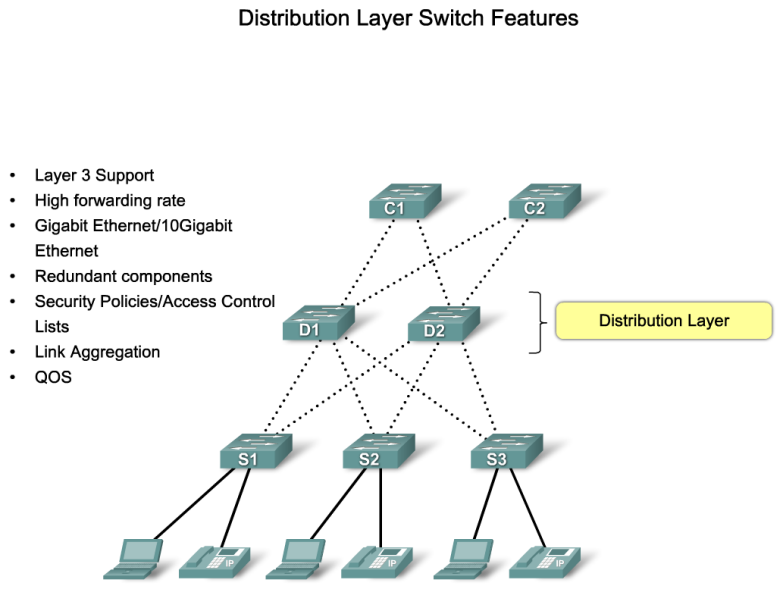
[6]

Figure 4 : Distribution layer

Distribution layer

Distributions layers function is to aggregate traffic from the access layer, and if needed forward the data up to the core layer.
The following features should be supported in the distribution layer:

- VLAN.
- Layer 3 switching.
  - Inter-VLAN routing.
  - Access Control Lists.
- High speed interfaces.
- Redundancy.
- link aggregation towards both access layer and core layer.
- QoS.



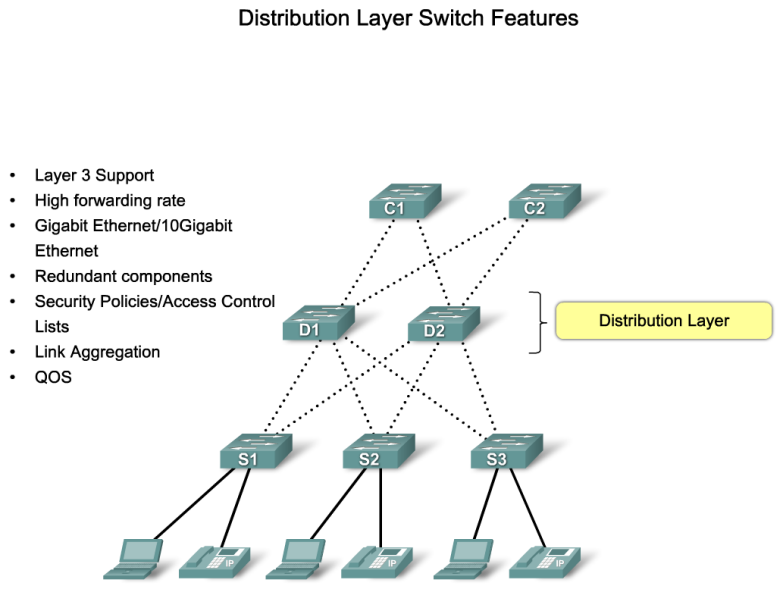
[6]

Figure 4 : Distribution layer

Distribution layer

Distributions layers function is to aggregate traffic from the access layer, and if needed forward the data up to the core layer.
The following features should be supported in the distribution layer:

- VLAN.
- Layer 3 switching.
  - Inter-VLAN routing.
  - Access Control Lists.
- High speed interfaces.
- Redundancy.
- link aggregation towards both access layer and core layer.
- QoS.



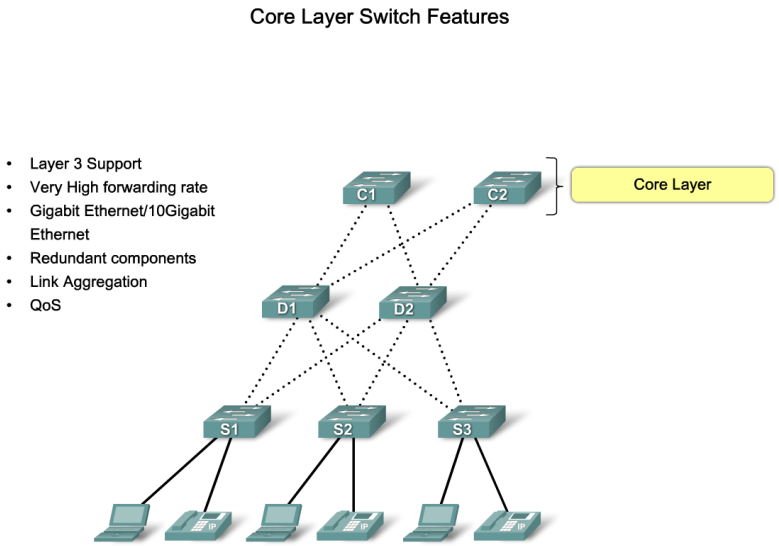
[6]

Figure 4 : Distribution layer

Core layer

The core layers function is to be a high-speed backbone and only forward traffic between different distribution layer switches, or towards the gateway.
The Following features are needed at the core layer.

- Lager 3 switching.
- VLAN.
- High bandwidth.
- Redundancy
- Aggregation.
- QoS.



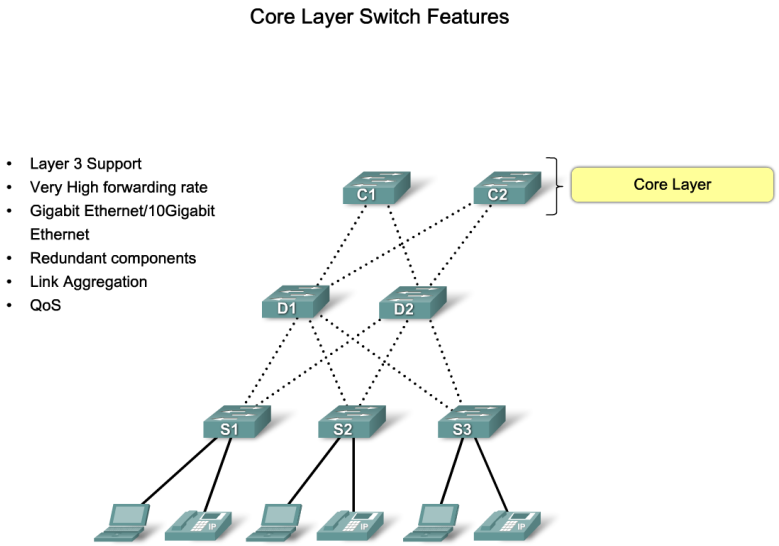
[6]

Figure 5 : Core layer

Core layer

The core layers function is to be a high-speed backbone and only forward traffic between different distribution layer switches, or towards the gateway.
The Following features are needed at the core layer.

- Lager 3 switching.
- VLAN.
- High bandwidth.
- Redundancy
- Aggregation.
- QoS.



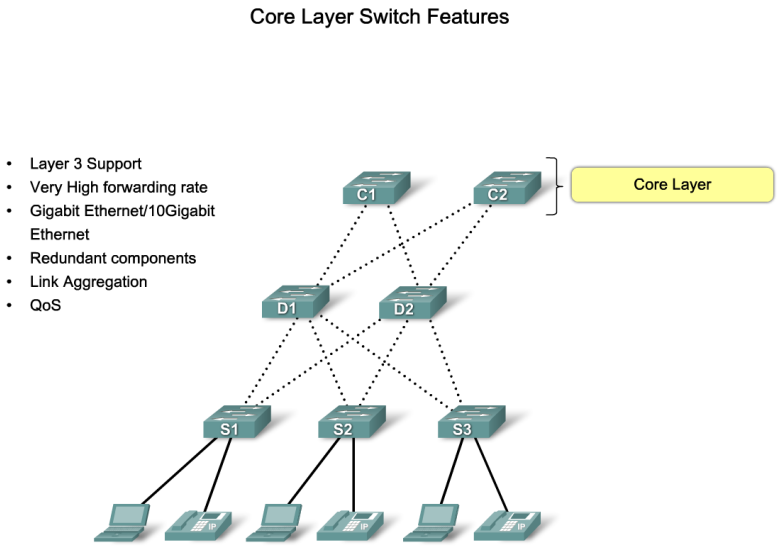
[6]

Figure 5 : Core layer

Core layer

The core layers function is to be a high-speed backbone and only forward traffic between different distribution layer switches, or towards the gateway.
The Following features are needed at the core layer.

- Lager 3 switching.
- VLAN.
- High bandwidth.
- Redundancy
- Aggregation.
- QoS.



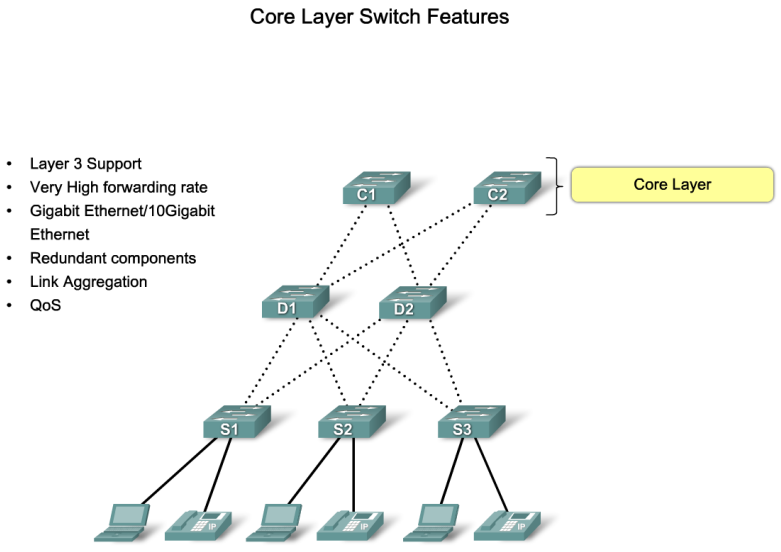
[6]

Figure 5 : Core layer

Core layer

The core layers function is to be a high-speed backbone and only forward traffic between different distribution layer switches, or towards the gateway.
The Following features are needed at the core layer.

- Lager 3 switching.
- VLAN.
- High bandwidth.
- Redundancy
- Aggregation.
- QoS.



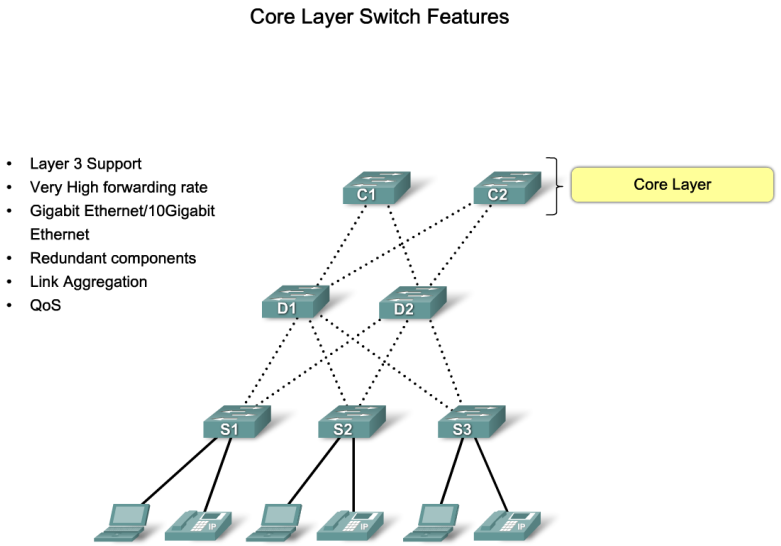
[6]

Figure 5 : Core layer

Core layer

The core layers function is to be a high-speed backbone and only forward traffic between different distribution layer switches, or towards the gateway.
The Following features are needed at the core layer.

- Lager 3 switching.
- VLAN.
- High bandwidth.
- Redundancy
- Aggregation.
- QoS.



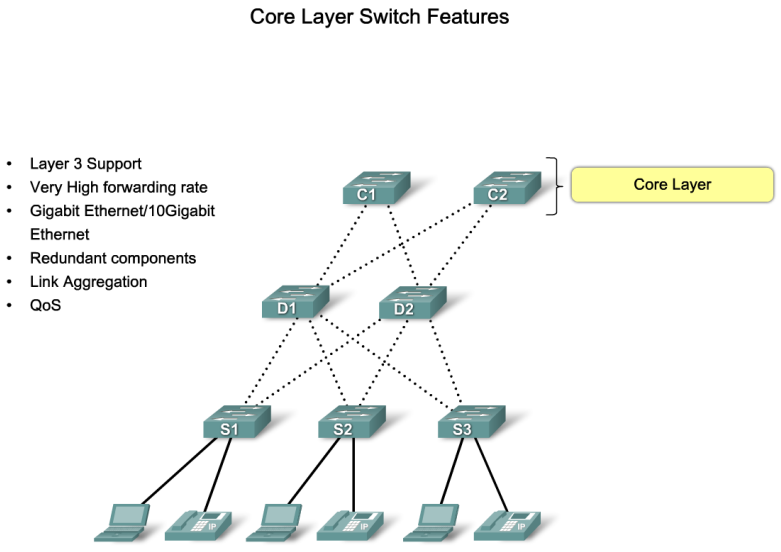
[6]

Figure 5 : Core layer

Core layer

The core layers function is to be a high-speed backbone and only forward traffic between different distribution layer switches, or towards the gateway.
The Following features are needed at the core layer.

- Lager 3 switching.
- VLAN.
- High bandwidth.
- Redundancy
- Aggregation.
- QoS.



[6]

Figure 5 : Core layer

Advantages

Advantages with a hierarchical network.

- Scalable.
- Redundant.
- Performance.
- Security, easy to control access at the access layer.
- Manageable. Since all the switches in the same layer has the same function.
- Maintenance.

Advantages

Advantages with a hierarchical network.

- Scalable.
- Redundant.
- Performance.
- Security, easy to control access at the access layer.
- Manageable. Since all the switches in the same layer has the same function.
- Maintenance.

Advantages

Advantages with a hierarchical network.

- Scalable.
- Redundant.
- Performance.
- Security, easy to control access at the access layer.
- Manageable. Since all the switches in the same layer has the same function.
- Maintenance.

Advantages

Advantages with a hierarchical network.

- Scalable.
- Redundant.
- Performance.
- Security, easy to control access at the access layer.
- Manageable. Since all the switches in the same layer has the same function.
- Maintenance.

Advantages

Advantages with a hierarchical network.

- Scalable.
- Redundant.
- Performance.
- Security, easy to control access at the access layer.
- Manageable. Since all the switches in the same layer has the same function.
- Maintenance.

Advantages

Advantages with a hierarchical network.

- Scalable.
- Redundant.
- Performance.
- Security, easy to control access at the access layer.
- Manageable. Since all the switches in the same layer has the same function.
- Maintenance.

Disadvantages

Disadvantages with a hierarchical topology:

- Redundancy require some version of STP to protect against layer 2 loops.
- Due to this, there will exist unused links in the network.

Disadvantages

Disadvantages with a hierarchical topology:

- Redundancy require some version of STP to protect against layer 2 loops.
- Due to this, there will exist unused links in the network.

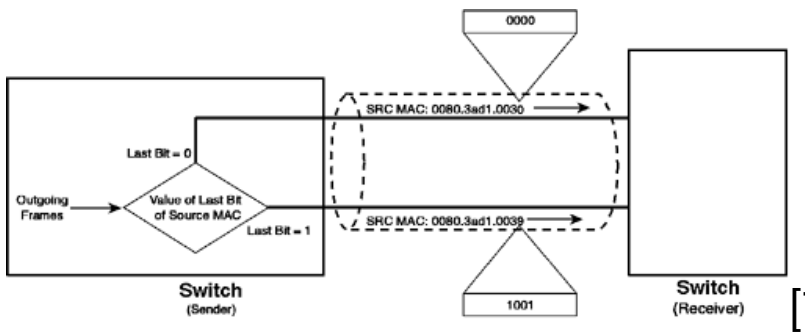
Aggregation with EtherChannel

Bandwidth aggregation or port aggregation allows us to use multiple physical interfaces to increase bandwidth and add redundancy.

- Links must have the same speed.
- Which link that will be used is based on the layer 2,3 or 4 address (And IOS version)
- XORing the addresses decides which link will be used.

# Links	Distribution
8	1:1:1:1:1:1:1:1
7	2:1:1:1:1:1:1
6	2:2:1:1:1:1
5	2:2:2:1:1
4	2:2:2:2
3	3:3:2
2	4:4

Table 1 : EtherChannel - Allocating the data traffic



[7]

Figure 6 : Deciding link use

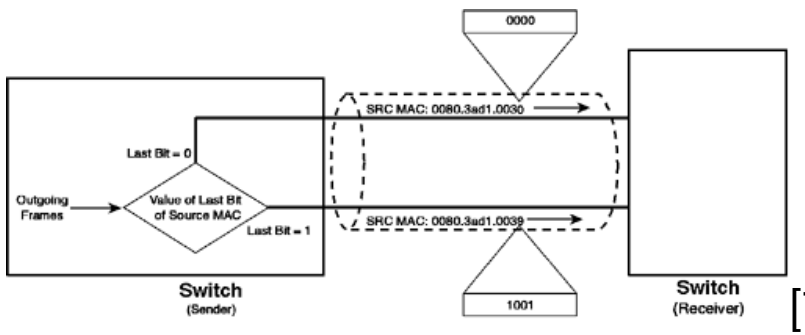
Aggregation with EtherChannel

Bandwidth aggregation or port aggregation allows us to use multiple physical interfaces to increase bandwidth and add redundancy.

- Links must have the same speed.
- Which link that will be used is based on the layer 2,3 or 4 address (And IOS version)
- XORing the addresses decides which link will be used.

# Links	Distribution
8	1:1:1:1:1:1:1:1
7	2:1:1:1:1:1:1
6	2:2:1:1:1:1
5	2:2:2:1:1
4	2:2:2:2
3	3:3:2
2	4:4

Table 1 : EtherChannel - Allocating the data traffic



[7]

Figure 6 : Deciding link use

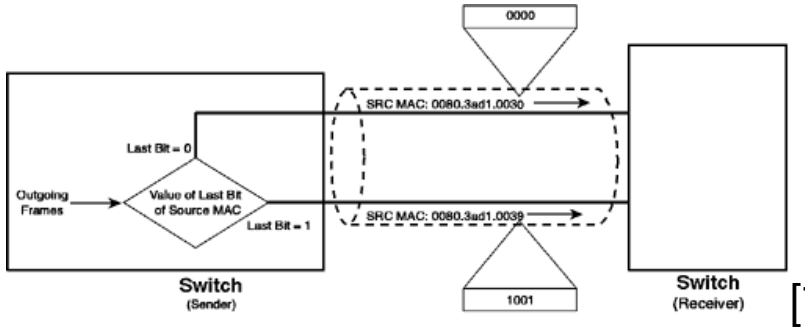
Aggregation with EtherChannel

Bandwidth aggregation or port aggregation allows us to use multiple physical interfaces to increase bandwidth and add redundancy.

- Links must have the same speed.
- Which link that will be used is based on the layer 2,3 or 4 address (And IOS version)
- XORing the addresses decides which link will be used.

# Links	Distribution
8	1:1:1:1:1:1:1:1
7	2:1:1:1:1:1:1
6	2:2:1:1:1:1
5	2:2:2:1:1
4	2:2:2:2
3	3:3:2
2	4:4

Table 1 : EtherChannel - Allocating the data traffic



[7]

Figure 6 : Deciding link use

Ethernet

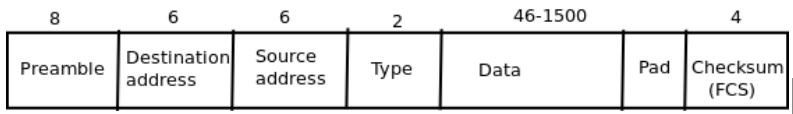
Two different versions of Ethernet. IEEE 802.2 and IEEE 802.3.
DIX Ethernet.
What is what?

DIX Ethernet

DIX Ethernet

DIX
Digital Intel Xerox

- Developed by Robert Metcalf mid 70s.
- Xerox PARC (Paolo Alto Research Center).
- Inspired by ALOHA that Metcalf worked with before he started at PARC.
- This protocol quickly gained popularity that Xerox together with Intel and Digital developed the DIX Ethernet standard.
- Published 1980.
- Preamble, Destination, Source, Type, Data, Pad, Checksum.
- Information about length is retrieved by looking into the data portion of the frame.
- This standard is the de-facto standard in today's NICs noder.

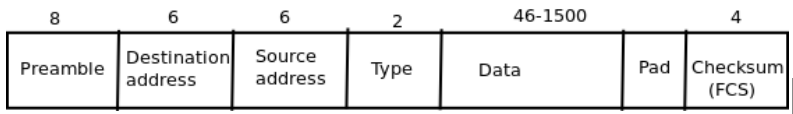


[2]

Figure 7 : Ethernet DIX - header

DIX
Digital Intel Xerox

- Developed by Robert Metcalf mid 70s.
- Xerox PARC (Paolo Alto Research Center).
- Inspired by ALOHA that Metcalf worked with before he started at PARC.
- This protocol quickly gained popularity that Xerox together with Intel and Digital developed the DIX Ethernet standard.
- Published 1980.
- Preamble, Destination, Source, Type, Data, Pad, Checksum.
- Information about length is retrieved by looking into the data portion of the frame.
- This standard is the de-facto standard in today's NICs noder.

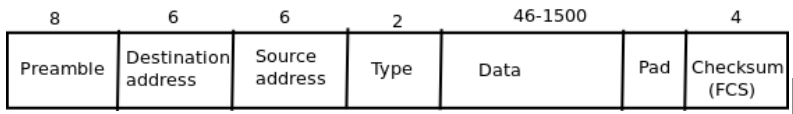


[2]

Figure 7 : Ethernet DIX - header

DIX
Digital Intel Xerox

- Developed by Robert Metcalf mid 70s.
- Xerox PARC (Paolo Alto Research Center).
- Inspired by ALOHA that Metcalf worked with before he started at PARC.
- This protocol quickly gained popularity that Xerox together with Intel and Digital developed the DIX Ethernet standard.
- Published 1980.
- Preamble, Destination, Source, Type, Data, Pad, Checksum.
- Information about length is retrieved by looking into the data portion of the frame.
- This standard is the de-facto standard in today's NICs noder.

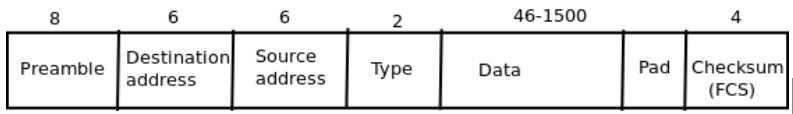


[2]

Figure 7 : Ethernet DIX - header

DIX
Digital Intel Xerox

- Developed by Robert Metcalf mid 70s.
- Xerox PARC (Paolo Alto Research Center).
- Inspired by ALOHA that Metcalf worked with before he started at PARC.
- This protocol quickly gained popularity that Xerox together with Intel and Digital developed the DIX Ethernet standard.
- Published 1980.
- Preamble, Destination, Source, Type, Data, Pad, Checksum.
- Information about length is retrieved by looking into the data portion of the frame.
- This standard is the de-facto standard in today's NICs noder.

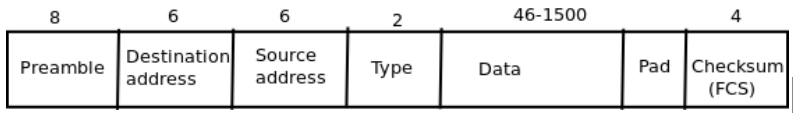


[2]

Figure 7 : Ethernet DIX - header

DIX
Digital Intel Xerox

- Developed by Robert Metcalf mid 70s.
- Xerox PARC (Paolo Alto Research Center).
- Inspired by ALOHA that Metcalf worked with before he started at PARC.
- This protocol quickly gained popularity that Xerox together with Intel and Digital developed the DIX Ethernet standard.
- Published 1980.
- Preamble, Destination, Source, Type, Data, Pad, Checksum.
- Information about length is retrieved by looking into the data portion of the frame.
- This standard is the de-facto standard in today's NICs noder.

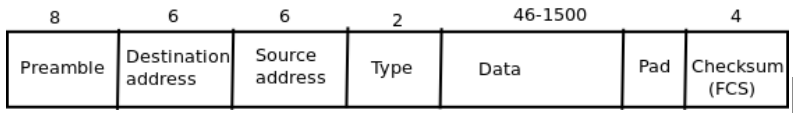


[2]

Figure 7 : Ethernet DIX - header

DIX
Digital Intel Xerox

- Developed by Robert Metcalf mid 70s.
- Xerox PARC (Paolo Alto Research Center).
- Inspired by ALOHA that Metcalf worked with before he started at PARC.
- This protocol quickly gained popularity that Xerox together with Intel and Digital developed the DIX Ethernet standard.
- Published 1980.
- Preamble, Destination, Source, Type, Data, Pad, Checksum.
- Information about length is retrieved by looking into the data portion of the frame.
- This standard is the de-facto standard in today's NICs noder.

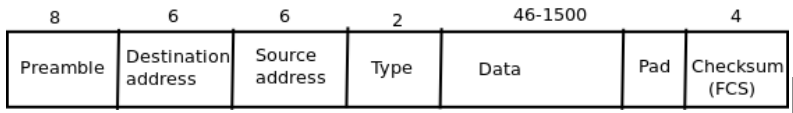


[2]

Figure 7 : Ethernet DIX - header

DIX
Digital Intel Xerox

- Developed by Robert Metcalf mid 70s.
- Xerox PARC (Paolo Alto Research Center).
- Inspired by ALOHA that Metcalf worked with before he started at PARC.
- This protocol quickly gained popularity that Xerox together with Intel and Digital developed the DIX Ethernet standard.
- Published 1980.
- Preamble, Destination, Source, Type, Data, Pad, Checksum.
- Information about length is retrieved by looking into the data portion of the frame.
- This standard is the de-facto standard in today's NICs noder.

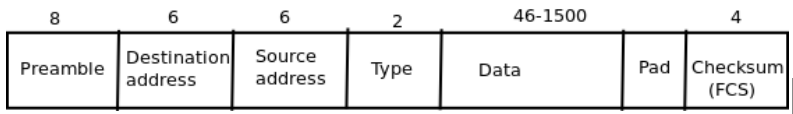


[2]

Figure 7 : Ethernet DIX - header

DIX
Digital Intel Xerox

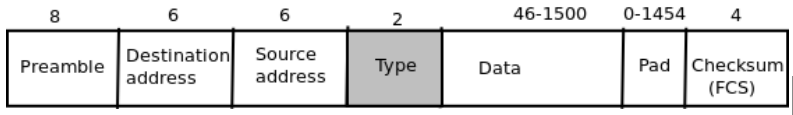
- Developed by Robert Metcalf mid 70s.
- Xerox PARC (Paolo Alto Research Center).
- Inspired by ALOHA that Metcalf worked with before he started at PARC.
- This protocol quickly gained popularity that Xerox together with Intel and Digital developed the DIX Ethernet standard.
- Published 1980.
- Preamble, Destination, Source, Type, Data, Pad, Checksum.
- Information about length is retrieved by looking into the data portion of the frame.
- This standard is the de-facto standard in today's NICs noder.



[2]

Figure 7 : Ethernet DIX - header

Ethertype



[2]

Figure 8 : Ethernet Type field

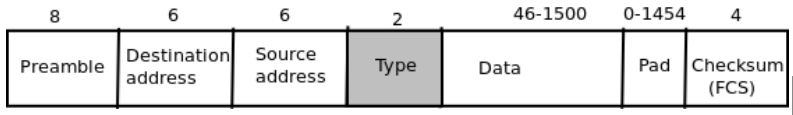
- 2 byte field
- Identifies the layer 3 protocol.

Ethertype #	Protocol
0800	IPv4
0806	ARP
86DD	IPv6

Table 2 : EtherType - Examples

[3]

Ethertype



[2]

Figure 8 : Ethernet Type field

- 2 byte field
- Identifies the layer 3 protocol.

Ethertype #	Protocol
0800	IPv4
0806	ARP
86DD	IPv6

Table 2 : EtherType - Examples

[3]

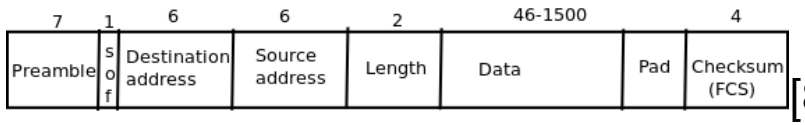
IEEE Ethernet

IEEE 802.3 and 802.2

IEEE 802.3

Published 1983 and is based on the DIX standard, with some few exceptions.

- Start of Frame - Informs that the frame will now start.
- Length instead of type
- More true to the layer model than DIX.
- Did not include information about the higher layer protocol.
- "Solved" with IEEE 802.2



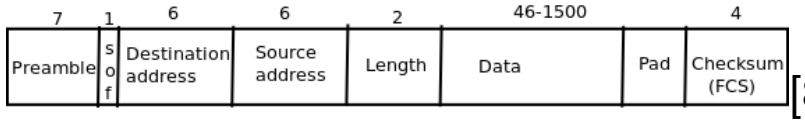
[8]

Figure 9 : Ethernet 802.3

IEEE 802.3

Published 1983 and is based on the DIX standard, with some few exceptions.

- Start of Frame - Informs that the frame will now start.
- Length instead of type
- More true to the layer model than DIX.
- Did not include information about the higher layer protocol.
- "Solved" with IEEE 802.2



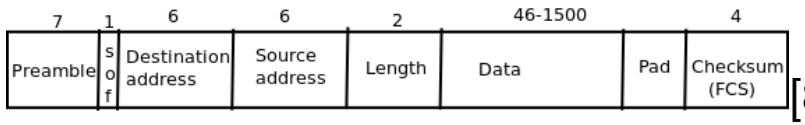
[8]

Figure 9 : Ethernet 802.3

IEEE 802.3

Published 1983 and is based on the DIX standard, with some few exceptions.

- Start of Frame - Informs that the frame will now start.
- Length instead of type
- More true to the layer model than DIX.
- Did not include information about the higher layer protocol.
- "Solved" with IEEE 802.2



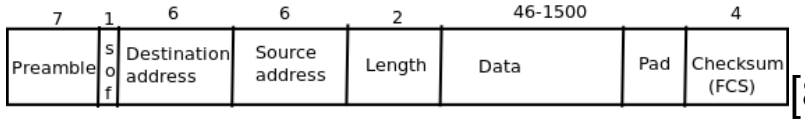
[8]

Figure 9 : Ethernet 802.3

IEEE 802.3

Published 1983 and is based on the DIX standard, with some few exceptions.

- Start of Frame - Informs that the frame will now start.
- Length instead of type
- More true to the layer model than DIX.
- Did not include information about the higher layer protocol.
- "Solved" with IEEE 802.2



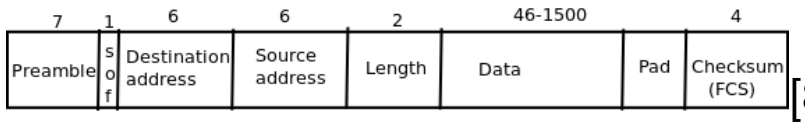
[8]

Figure 9 : Ethernet 802.3

IEEE 802.3

Published 1983 and is based on the DIX standard, with some few exceptions.

- Start of Frame - Informs that the frame will now start.
- Length instead of type
- More true to the layer model than DIX.
- Did not include information about the higher layer protocol.
- "Solved" with IEEE 802.2

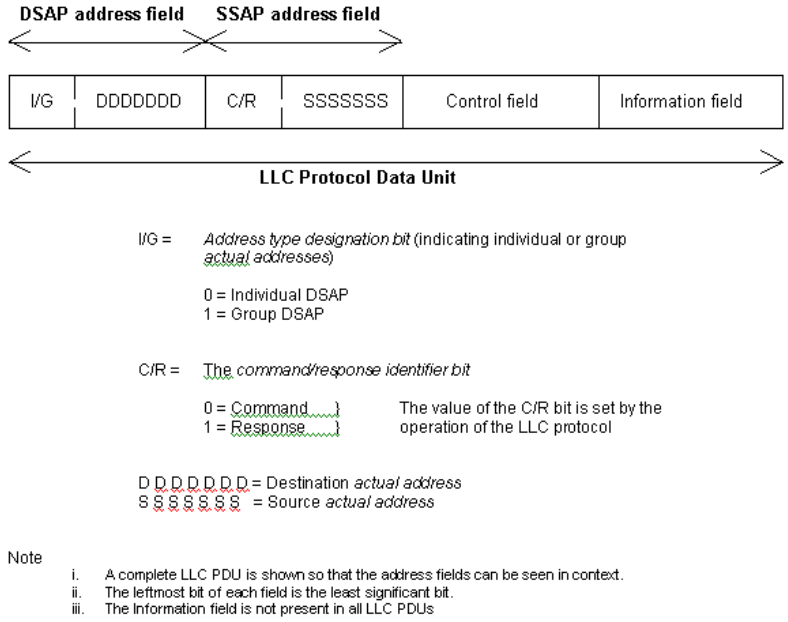


[8]

Figure 9 : Ethernet 802.3

IEEE 802.2
Logical Link Control (LLC)

- Identifies the higher layer protocol.
- DSAP (Destination Service Access Point) 6 bits.
- SSAP (Source Service Access Point)
- IP was given a SAP-number, ARP wasn't.
- I/G Individual/Group, deliver to multiple layer three protocols.
- Command/Response - Indicates whether the package is a request or response.
- Control Field - Indicates whether the connection is connection-less or connection-oriented, reliable or unreliable.
- Information field, Used for extra information, used by SNAP.

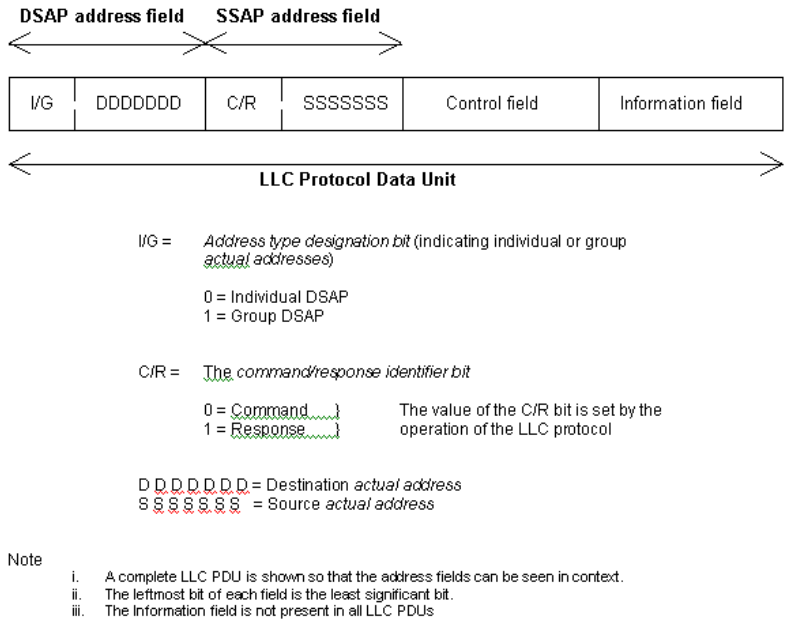


[4]

Figure 10 : LLC header

IEEE 802.2
Logical Link Control (LLC)

- Identifies the higher layer protocol.
- DSAP (Destination Service Access Point) 6 bits.
- SSAP (Source Service Access Point)
- IP was given a SAP-number, ARP wasn't.
- I/G Individual/Group, deliver to multiple layer three protocols.
- Command/Response - Indicates whether the package is a request or response.
- Control Field - Indicates whether the connection is connection-less or connection-oriented, reliable or unreliable.
- Information field, Used for extra information, used by SNAP.

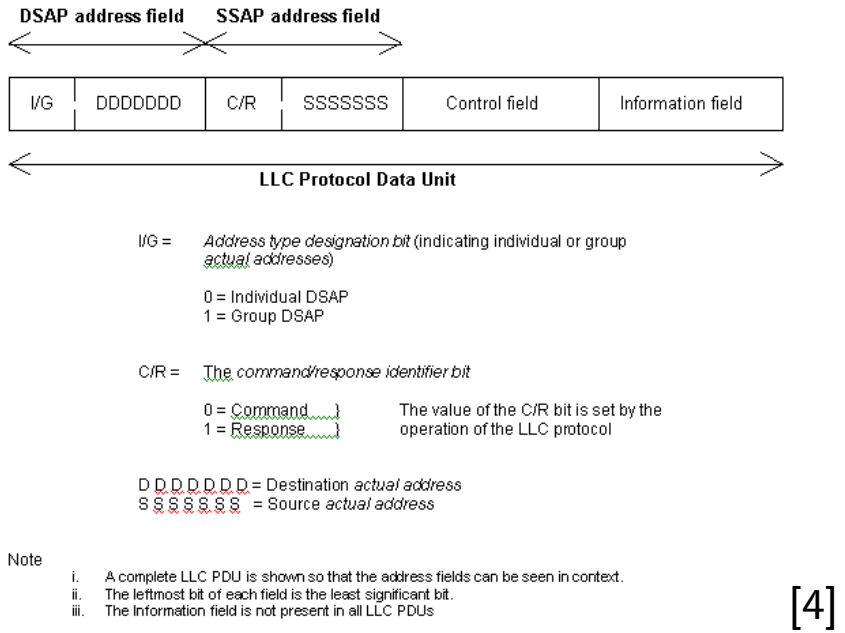


[4]

Figure 10 : LLC header

IEEE 802.2
Logical Link Control (LLC)

- Identifies the higher layer protocol.
- DSAP (Destination Service Access Point) 6 bits.
- SSAP (Source Service Access Point)
- IP was given a SAP-number, ARP wasn't.
- I/G Individual/Group, deliver to multiple layer three protocols.
- Command/Response - Indicates whether the package is a request or response.
- Control Field - Indicates whether the connection is connection-less or connection-oriented, reliable or unreliable.
- Information field, Used for extra information, used by SNAP.

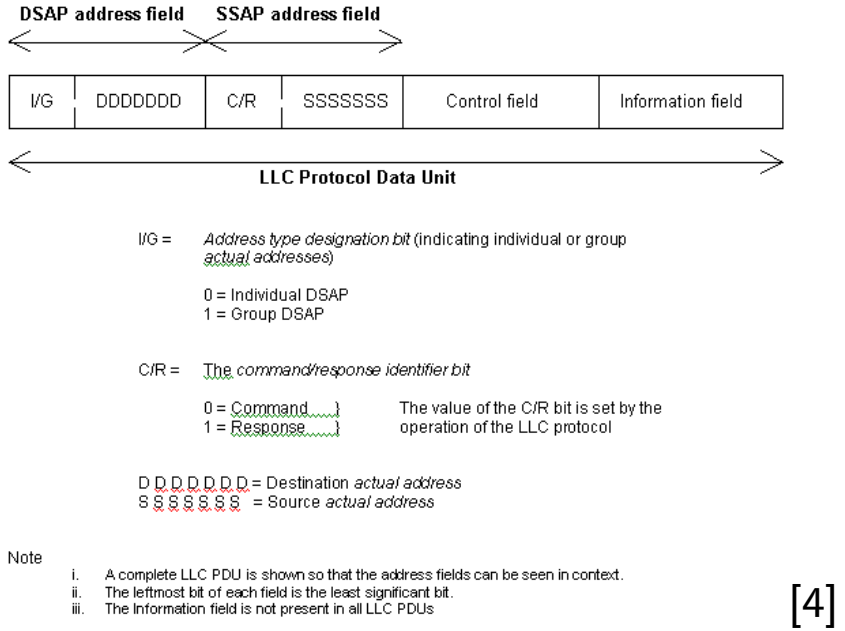


[4]

Figure 10 : LLC header

IEEE 802.2
Logical Link Control (LLC)

- Identifies the higher layer protocol.
- DSAP (Destination Service Access Point) 6 bits.
- SSAP (Source Service Access Point)
- IP was given a SAP-number, ARP wasn't.
- I/G Individual/Group, deliver to multiple layer three protocols.
- Command/Response - Indicates whether the package is a request or response.
- Control Field - Indicates whether the connection is connection-less or connection-oriented, reliable or unreliable.
- Information field, Used for extra information, used by SNAP.

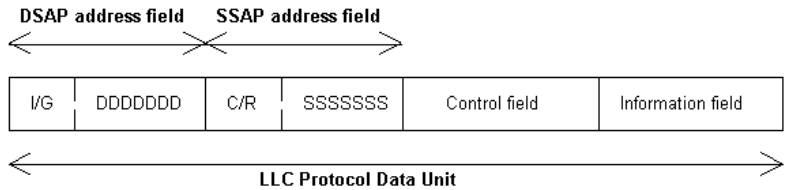


[4]

Figure 10 : LLC header

IEEE 802.2
Logical Link Control (LLC)

- Identifies the higher layer protocol.
- DSAP (Destination Service Access Point) 6 bits.
- SSAP (Source Service Access Point)
- IP was given a SAP-number, ARP wasn't.
- I/G Individual/Group, deliver to multiple layer three protocols.
- Command/Response - Indicates whether the package is a request or response.
- Control Field - Indicates whether the connection is connection-less or connection-oriented, reliable or unreliable.
- Information field, Used for extra information, used by SNAP.



I/G = *Address type designation bit* (indicating individual or group *actual addresses*)

0 = Individual DSAP
1 = Group DSAP

C/R = The *command/response identifier bit*

0 = Command } The value of the C/R bit is set by the
1 = Response } operation of the LLC protocol

D D D D D D D = Destination *actual address*
S S S S S S S = Source *actual address*

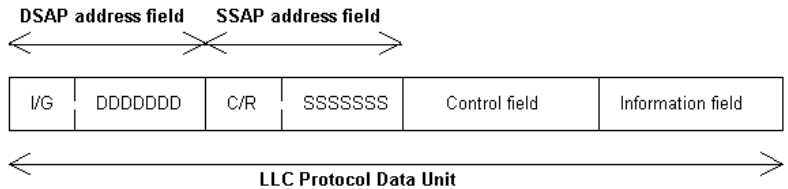
Note
i. A complete LLC PDU is shown so that the address fields can be seen in context.
ii. The leftmost bit of each field is the least significant bit.
iii. The Information field is not present in all LLC PDUs

[4]

Figure 10 : LLC header

IEEE 802.2
Logical Link Control (LLC)

- Identifies the higher layer protocol.
- DSAP (Destination Service Access Point) 6 bits.
- SSAP (Source Service Access Point)
- IP was given a SAP-number, ARP wasn't.
- I/G Individual/Group, deliver to multiple layer three protocols.
- Command/Response - Indicates whether the package is a request or response.
- Control Field - Indicates whether the connection is connection-less or connection-oriented, reliable or unreliable.
- Information field, Used for extra information, used by SNAP.



I/G = *Address type designation bit* (indicating individual or group *actual addresses*)

0 = Individual DSAP
1 = Group DSAP

C/R = The *command/response identifier bit*

0 = Command } The value of the C/R bit is set by the
1 = Response } operation of the LLC protocol

D D D D D D D = Destination *actual address*
S S S S S S S = Source *actual address*

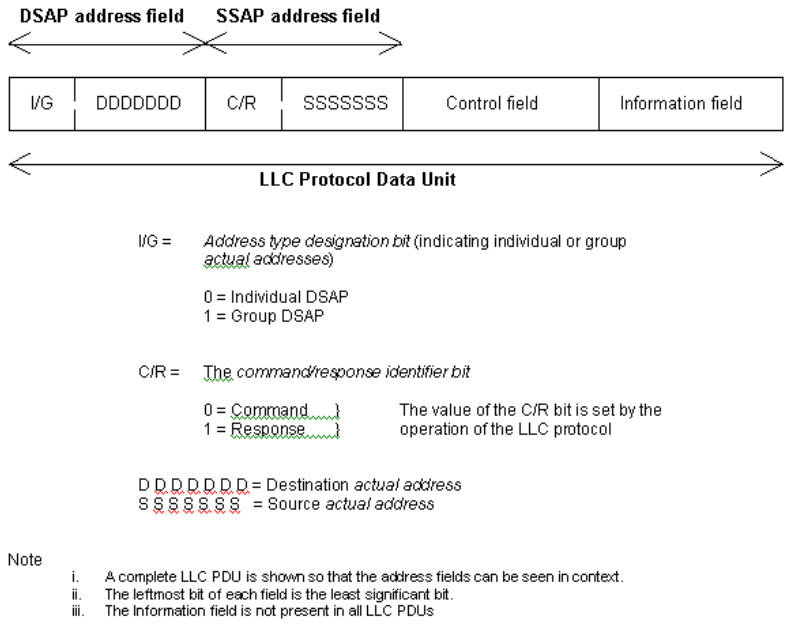
Note
i. A complete LLC PDU is shown so that the address fields can be seen in context.
ii. The leftmost bit of each field is the least significant bit.
iii. The Information field is not present in all LLC PDUs

[4]

Figure 10 : LLC header

IEEE 802.2
Logical Link Control (LLC)

- Identifies the higher layer protocol.
- DSAP (Destination Service Access Point) 6 bits.
- SSAP (Source Service Access Point)
- IP was given a SAP-number, ARP wasn't.
- I/G Individual/Group, deliver to multiple layer three protocols.
- Command/Response - Indicates whether the package is a request or response.
- Control Field - Indicates whether the connection is connection-less or connection-oriented, reliable or unreliable.
- Information field, Used for extra information, used by SNAP.

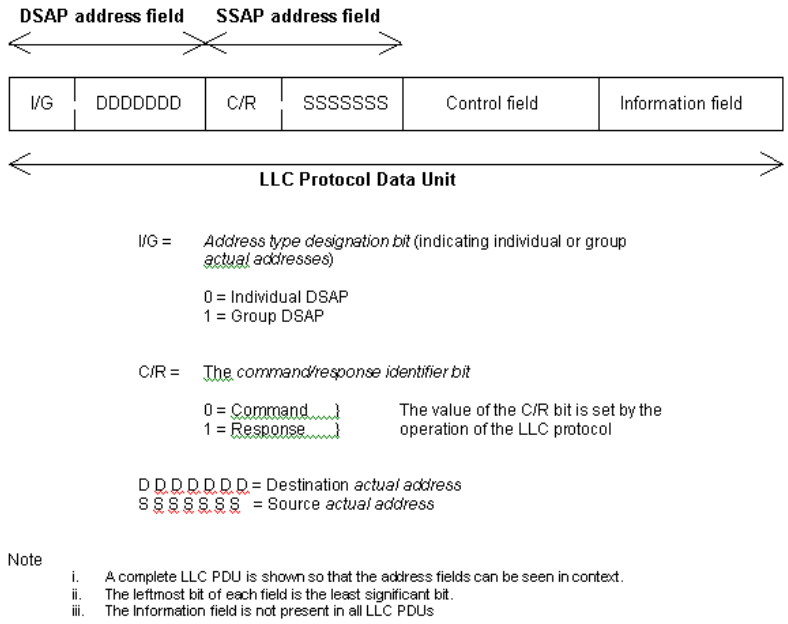


[4]

Figure 10 : LLC header

IEEE 802.2
Logical Link Control (LLC)

- Identifies the higher layer protocol.
- DSAP (Destination Service Access Point) 6 bits.
- SSAP (Source Service Access Point)
- IP was given a SAP-number, ARP wasn't.
- I/G Individual/Group, deliver to multiple layer three protocols.
- Command/Response - Indicates whether the package is a request or response.
- Control Field - Indicates whether the connection is connection-less or connection-oriented, reliable or unreliable.
- Information field, Used for extra information, used by SNAP.

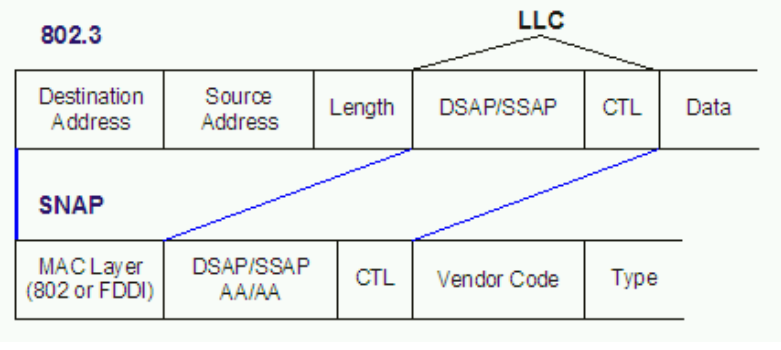


[4]

Figure 10 : LLC header

SNAP
Sub Network Access Point

- Supplement to 802.2
- Increases the number of protocols that 802.2 can support.
- Vendor code identifies the organization responsible for the protocol.
- Type (Ethertype)

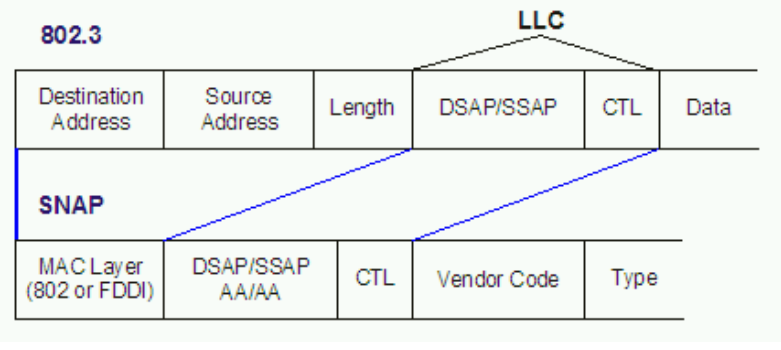


[1]

Figure 11 : SNAP header

SNAP
Sub Network Access Point

- Supplement to 802.2
- Increases the number of protocols that 802.2 can support.
- Vendor code identifies the organization responsible for the protocol.
- Type (Ethertype)

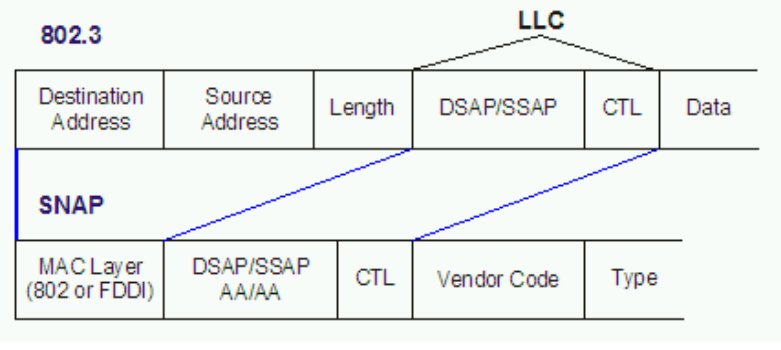


[1]

Figure 11 : SNAP header

SNAP
Sub Network Access Point

- Supplement to 802.2
- Increases the number of protocols that 802.2 can support.
- Vendor code identifies the organization responsible for the protocol.
- Type (Ethertype)

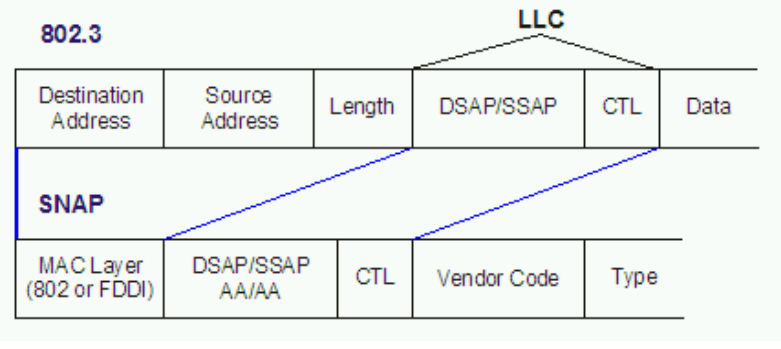


[1]

Figure 11 : SNAP header

SNAP
Sub Network Access Point

- Supplement to 802.2
- Increases the number of protocols that 802.2 can support.
- Vendor code identifies the organization responsible for the protocol.
- Type (Ethertype)



[1]

Figure 11 : SNAP header

IEEE 802.3 compromise

- Gave up 1997.
- Type is now used as both length and type, hence Type/Length.
- Since all the Ethertype values used in 1997 had values above 1500.
- If Length/Type field is less than or equal to 1500 the field is interpreted as length.
- If Length/Type field is larger than 1500, it will be interpreted as Type.

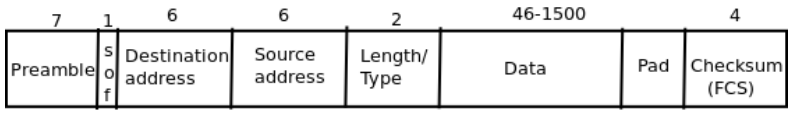


Figure 12 : Ethernets compromise

IEEE 802.3 compromise

- Gave up 1997.
- Type is now used as both length and type, hence Type/Length.
- Since all the Ethertype values used in 1997 had values above 1500.
- If Length/Type field is less than or equal to 1500 the field is interpreted as length.
- If Length/Type field is larger than 1500, it will be interpreted as Type.

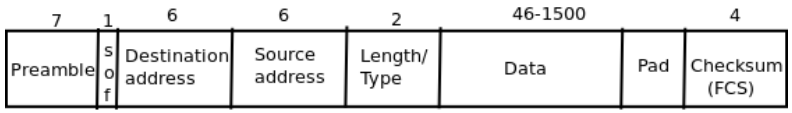


Figure 12 : Ethernets compromise

IEEE 802.3 compromise

- Gave up 1997.
- Type is now used as both length and type, hence Type/Length.
- Since all the Ethertype values used in 1997 had values above 1500.
- If Length/Type field is less than or equal to 1500 the field is interpreted as length.
- If Length/Type field is larger than 1500, it will be interpreted as Type.

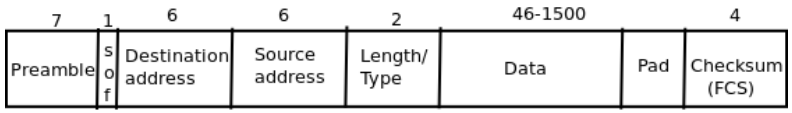


Figure 12 : Ethernets compromise

IEEE 802.3 compromise

- Gave up 1997.
- Type is now used as both length and type, hence Type/Length.
- Since all the Ethertype values used in 1997 had values above 1500.
- If Length/Type field is less than or equal to 1500 the field is interpreted as length.
- If Length/Type field is larger than 1500, it will be interpreted as Type.

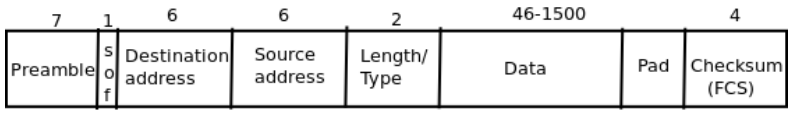


Figure 12 : Ethernets compromise

IEEE 802.3 compromise

- Gave up 1997.
- Type is now used as both length and type, hence Type/Length.
- Since all the Ethertype values used in 1997 had values above 1500.
- If Length/Type field is less than or equal to 1500 the field is interpreted as length.
- If Length/Type field is larger than 1500, it will be interpreted as Type.

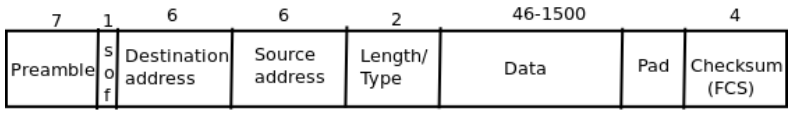


Figure 12 : Ethernets compromise

Back to CCNA3

Back to CCNA3 - LAN Switching and Wireless

Switch MAC address table

- MAC address table contains which MAC addresses that can be reached on a certain port.
- Inspects source MAC on the receiving frames and add this to its table.
- If the destination address is not in the table, the frame will be broadcasted.
- When the destination frame responds, its MAC address will be added to the table.
- The switch already knows since step 2 which port the destination address can be reached on.

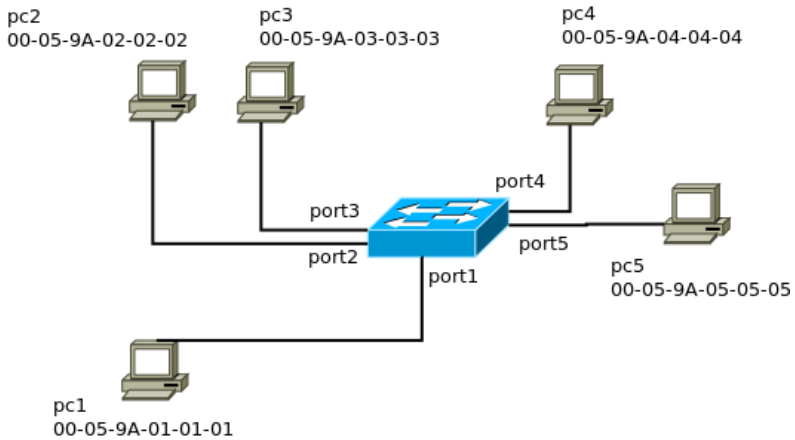


Figure 13 : Switch topology

Port #	MAC
1	
2	
3	
4	
5	

Table 3 : Switch MAC Address Table

Switch MAC address table

- MAC address table contains which MAC addresses that can be reached on a certain port.
- Inspects source MAC on the receiving frames and add this to its table.
- If the destination address is not in the table, the frame will be broadcasted.
- When the destination frame responds, its MAC address will be added to the table.
- The switch already knows since step 2 which port the destination address can be reached on.

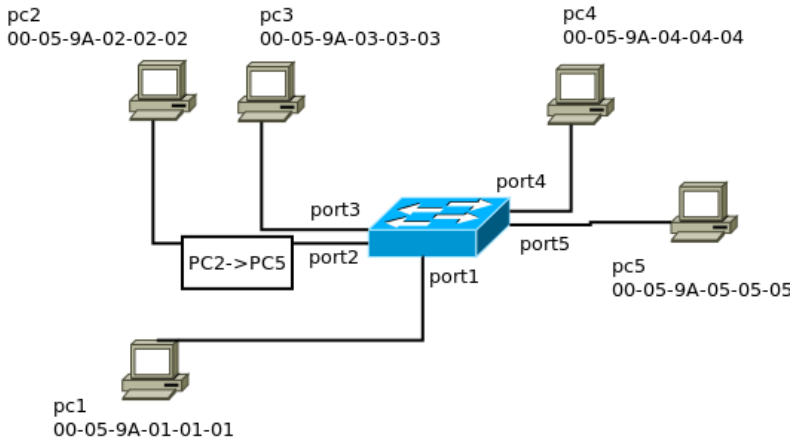


Figure 14 : Switch topology

Port #	MAC
1	
2	
3	
4	
5	

Table 4 : Switch MAC Address Table

Switch MAC address table

- MAC address table contains which MAC addresses that can be reached on a certain port.
- Inspects source MAC on the receiving frames and add this to its table.
- If the destination address is not in the table, the frame will be broadcasted.
- When the destination frame responds, its MAC address will be added to the table.
- The switch already knows since step 2 which port the destination address can be reached on.

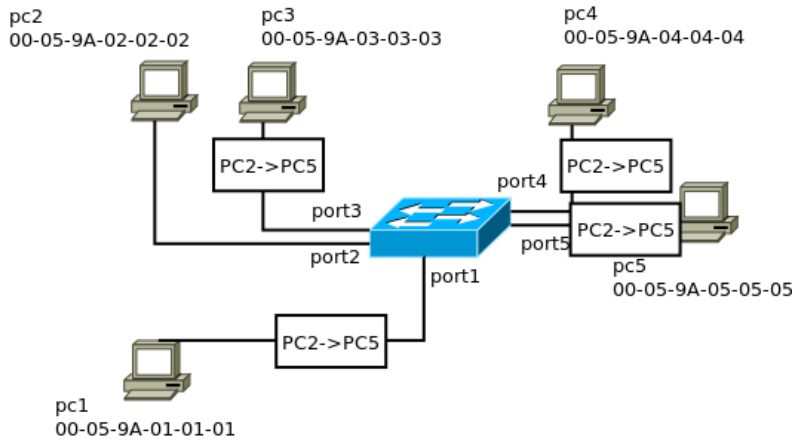


Figure 15 : Switch topology

Port #	MAC
1 2 3 4 5	00-05-9A-02-02-02

Table 5 : Switch MAC Address Table

Switch MAC address table

- MAC address table contains which MAC addresses that can be reached on a certain port.
- Inspects source MAC on the receiving frames and add this to its table.
- If the destination address is not in the table, the frame will be broadcasted.
- When the destination frame responds, its MAC address will be added to the table.
- The switch already knows since step 2 which port the destination address can be reached on.

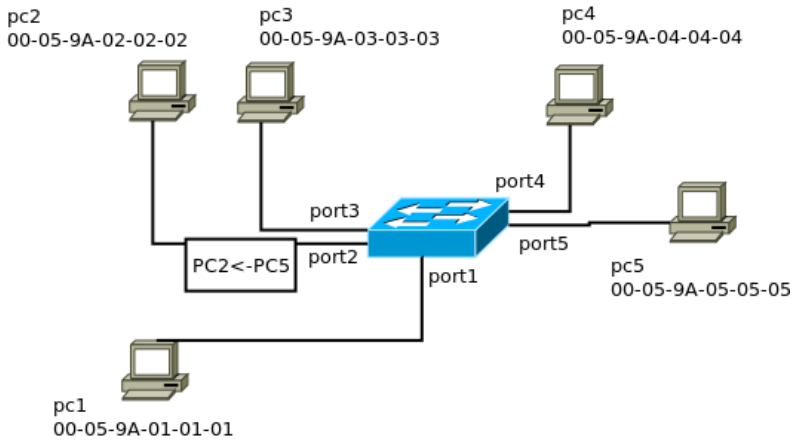


Figure 16 : Switch topology

Port #	MAC
1	
2	00-05-9A-02-02-02
3	
4	
5	00-05-9A-05-05-05

Table 6 : Switch MAC Address Table

Switch MAC address table

- MAC address table contains which MAC addresses that can be reached on a certain port.
- Inspects source MAC on the receiving frames and add this to its table.
- If the destination address is not in the table, the frame will be broadcasted.
- When the destination frame responds, its MAC address will be added to the table.
- The switch already knows since step 2 which port the destination address can be reached on.

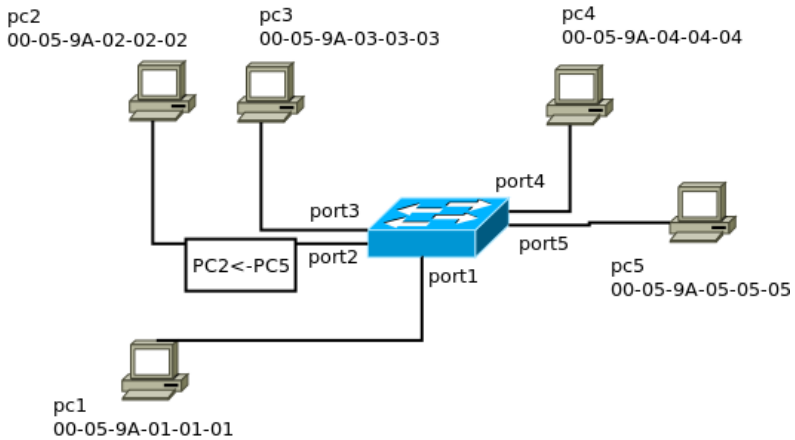


Figure 16 : Switch topology

Port #	MAC
1	
2	00-05-9A-02-02-02
3	
4	
5	00-05-9A-05-05-05

Table 6 : Switch MAC Address Table

Switching fabrics

- Switching fabric is the mechanism that will take the frame from the incoming queue and places it in the correct outgoing queue.
- Switching via memory.
- Switching via a bus.
- Switching via a crossbar.

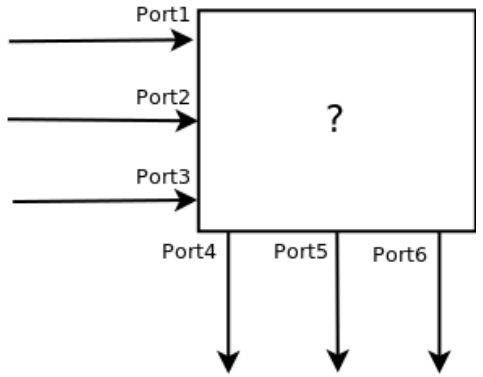


Figure 17 : Switching fabric

Switching fabrics

- Switching fabric is the mechanism that will take the frame from the incoming queue and places it in the correct outgoing queue.
- Switching via memory.
- Switching via a bus.
- Switching via a crossbar.

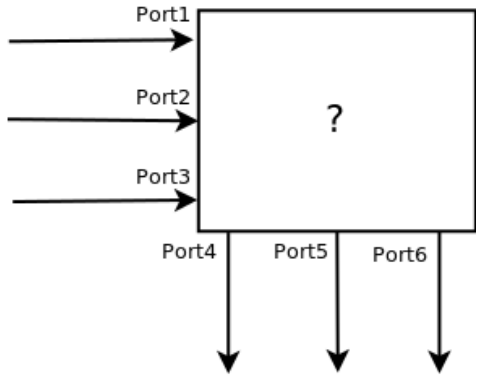


Figure 17 : Switching fabric

Switching fabrics

- Switching fabric is the mechanism that will take the frame from the incoming queue and places it in the correct outgoing queue.
- Switching via memory.
- Switching via a bus.
- Switching via a crossbar.

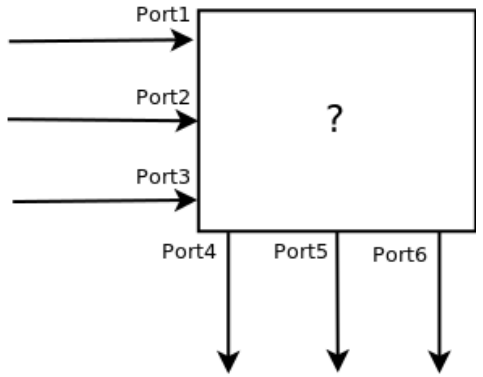


Figure 17 : Switching fabric

Switching fabrics

- Switching fabric is the mechanism that will take the frame from the incoming queue and places it in the correct outgoing queue.
- Switching via memory.
- Switching via a bus.
- Switching via a crossbar.

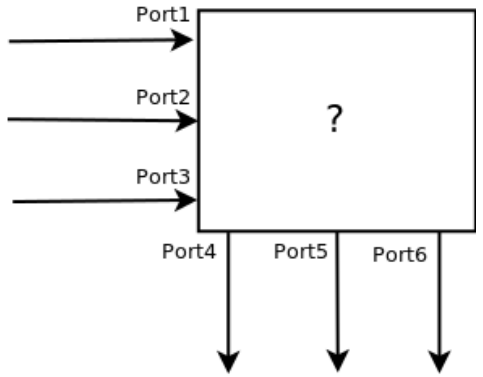
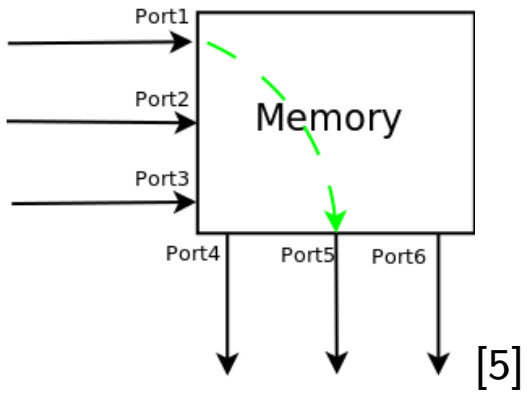


Figure 17 : Switching fabric

Switching via memory

- Most common type of switching, used primarily in regular computers and in some Cisco switches.
- The processor will handle the switching as a common I/O operation.
- When a frame is places in an incoming interfaces buffer, an interrupt will be sent.
- The frame will be copied into the memory and the destination will be checked, after which it will be places in the correct outgoing queue buffer.
- Can only handle one frame at a time.
- Speed is limited by the memory's bandwidth

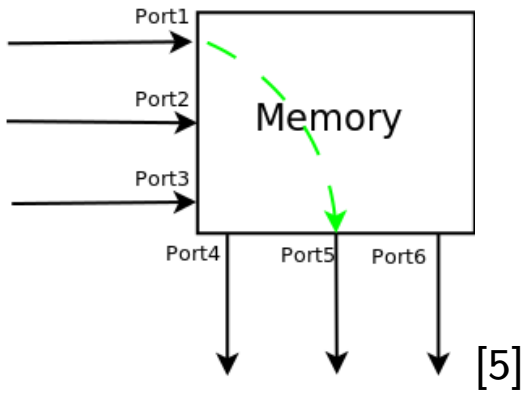


[5]

Figure 18 : Switching fabric - Memory

Switching via memory

- Most common type of switching, used primarily in regular computers and in some Cisco switches.
- The processor will handle the switching as a common I/O operation.
- When a frame is places in an incoming interfaces buffer, an interrupt will be sent.
- The frame will be copied into the memory and the destination will be checked, after which it will be places in the correct outgoing queue buffer.
- Can only handle one frame at a time.
- Speed is limited by the memory's bandwidth

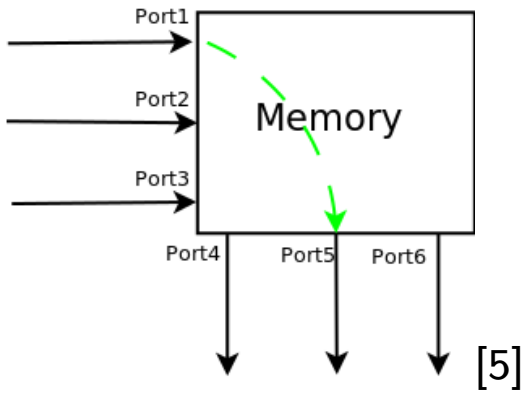


[5]

Figure 18 : Switching fabric - Memory

Switching via memory

- Most common type of switching, used primarily in regular computers and in some Cisco switches.
- The processor will handle the switching as a common I/O operation.
- When a frame is places in an incoming interfaces buffer, an interrupt will be sent.
- The frame will be copied into the memory and the destination will be checked, after which it will be places in the correct outgoing queue buffer.
- Can only handle one frame at a time.
- Speed is limited by the memory's bandwidth

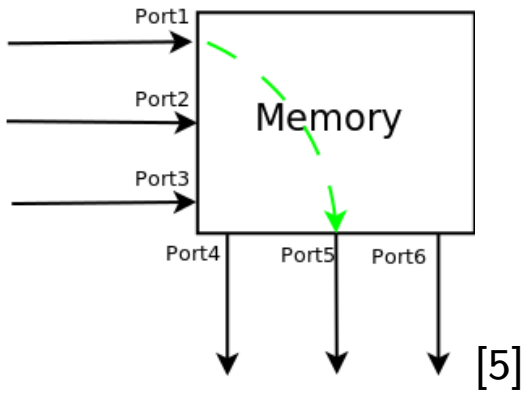


[5]

Figure 18 : Switching fabric - Memory

Switching via memory

- Most common type of switching, used primarily in regular computers and in some Cisco switches.
- The processor will handle the switching as a common I/O operation.
- When a frame is places in an incoming interfaces buffer, an interrupt will be sent.
- The frame will be copied into the memory and the destination will be checked, after which it will be places in the correct outgoing queue buffer.
- Can only handle one frame at a time.
- Speed is limited by the memory's bandwidth



[5]

Figure 18 : Switching fabric - Memory

Switching via memory

- Most common type of switching, used primarily in regular computers and in some Cisco switches.
- The processor will handle the switching as a common I/O operation.
- When a frame is places in an incoming interfaces buffer, an interrupt will be sent.
- The frame will be copied into the memory and the destination will be checked, after which it will be places in the correct outgoing queue buffer.
- Can only handle one frame at a time.
- Speed is limited by the memory's bandwidth

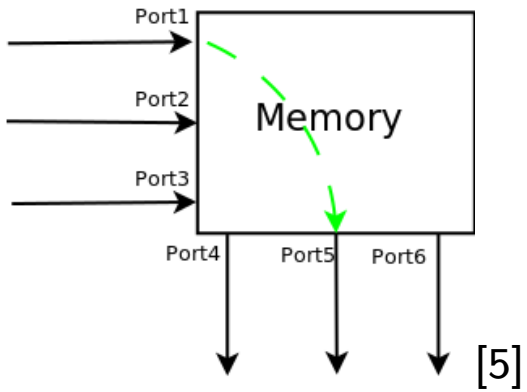


Figure 18 : Switching fabric - Memory

Switching via memory

- Most common type of switching, used primarily in regular computers and in some Cisco switches.
- The processor will handle the switching as a common I/O operation.
- When a frame is places in an incoming interfaces buffer, an interrupt will be sent.
- The frame will be copied into the memory and the destination will be checked, after which it will be places in the correct outgoing queue buffer.
- Can only handle one frame at a time.
- Speed is limited by the memory's bandwidth

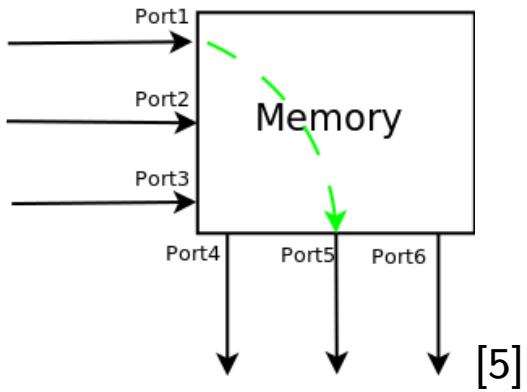
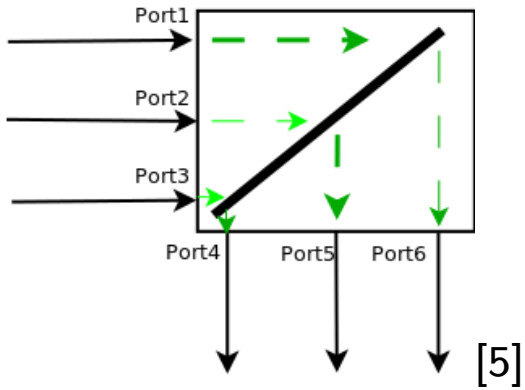


Figure 18 : Switching fabric - Memory

Switching via bus

- Incoming frames will be places in the incoming queue buffer on the interface.
- The destination will be inspected and an internal header will be added to the frame.
- The frame will be sent to all outgoing interfaces using a shared bus.
- Only the interface that has been given in the internal header will accept the frame.
- Only one frame can be handled at one time.
- Usually a high speed bus are used. (Cisco 5600 have a bus speed of 32Gbps)
- Forwarding speed is limited by the speed of the bus.

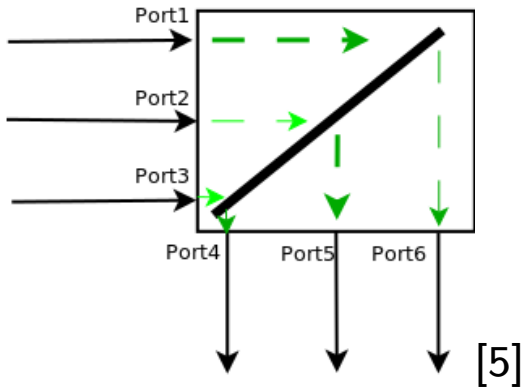


[5]

Figure 19 : Switching fabric - bus

Switching via bus

- Incoming frames will be places in the incoming queue buffer on the interface.
- The destination will be inspected and an internal header will be added to the frame.
- The frame will be sent to all outgoing interfaces using a shared bus.
- Only the interface that has been given in the internal header will accept the frame.
- Only one frame can be handled at one time.
- Usually a high speed bus are used. (Cisco 5600 have a bus speed of 32Gbps)
- Forwarding speed is limited by the speed of the bus.

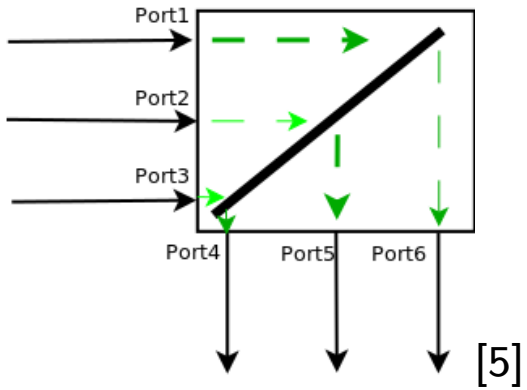


[5]

Figure 19 : Switching fabric - bus

Switching via bus

- Incoming frames will be places in the incoming queue buffer on the interface.
- The destination will be inspected and an internal header will be added to the frame.
- The frame will be sent to all outgoing interfaces using a shared bus.
- Only the interface that has been given in the internal header will accept the frame.
- Only one frame can be handled at one time.
- Usually a high speed bus are used. (Cisco 5600 have a bus speed of 32Gbps)
- Forwarding speed is limited by the speed of the bus.

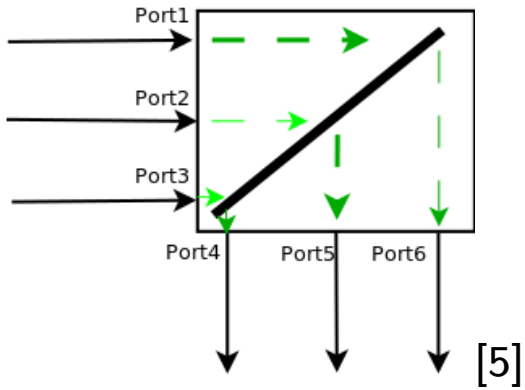


[5]

Figure 19 : Switching fabric - bus

Switching via bus

- Incoming frames will be places in the incoming queue buffer on the interface.
- The destination will be inspected and an internal header will be added to the frame.
- The frame will be sent to all outgoing interfaces using a shared bus.
- Only the interface that has been given in the internal header will accept the frame.
- Only one frame can be handled at one time.
- Usually a high speed bus are used. (Cisco 5600 have a bus speed of 32Gbps)
- Forwarding speed is limited by the speed of the bus.

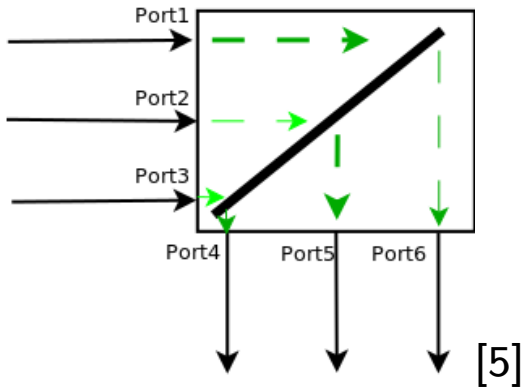


[5]

Figure 19 : Switching fabric - bus

Switching via bus

- Incoming frames will be places in the incoming queue buffer on the interface.
- The destination will be inspected and an internal header will be added to the frame.
- The frame will be sent to all outgoing interfaces using a shared bus.
- Only the interface that has been given in the internal header will accept the frame.
- Only one frame can be handled at one time.
- Usually a high speed bus are used. (Cisco 5600 have a bus speed of 32Gbps)
- Forwarding speed is limited by the speed of the bus.

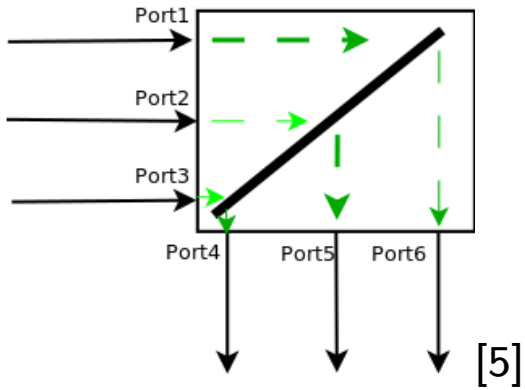


[5]

Figure 19 : Switching fabric - bus

Switching via bus

- Incoming frames will be places in the incoming queue buffer on the interface.
- The destination will be inspected and an internal header will be added to the frame.
- The frame will be sent to all outgoing interfaces using a shared bus.
- Only the interface that has been given in the internal header will accept the frame.
- Only one frame can be handled at one time.
- Usually a high speed bus are used. (Cisco 5600 have a bus speed of 32Gbps)
- Forwarding speed is limited by the speed of the bus.

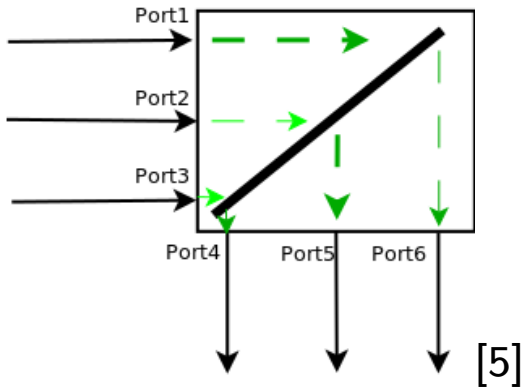


[5]

Figure 19 : Switching fabric - bus

Switching via bus

- Incoming frames will be places in the incoming queue buffer on the interface.
- The destination will be inspected and an internal header will be added to the frame.
- The frame will be sent to all outgoing interfaces using a shared bus.
- Only the interface that has been given in the internal header will accept the frame.
- Only one frame can be handled at one time.
- Usually a high speed bus are used. (Cisco 5600 have a bus speed of 32Gbps)
- Forwarding speed is limited by the speed of the bus.

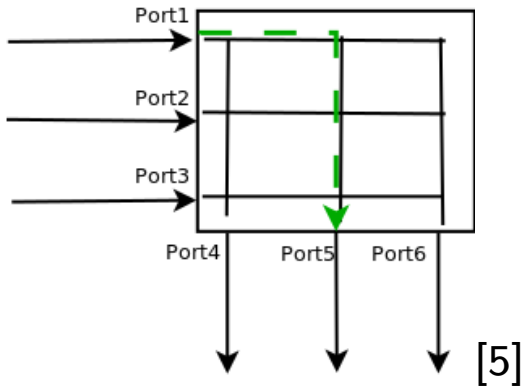


[5]

Figure 19 : Switching fabric - bus

Switching via crossbar

- A network of N incoming ports and N outgoing ports.
- Every port has its own bus, that can be interconnected when needed, that is 2N busses.
- An incoming frame is analysed and once the destination port has been found, the incoming bus will be connected to the outgoing bus.
- Allows multiple frames to be sent in parallel
- If two frames are destined for the same outgoing interface, one must be queued.
- This technique is used in the more advanced switches (Catalyst 12000)

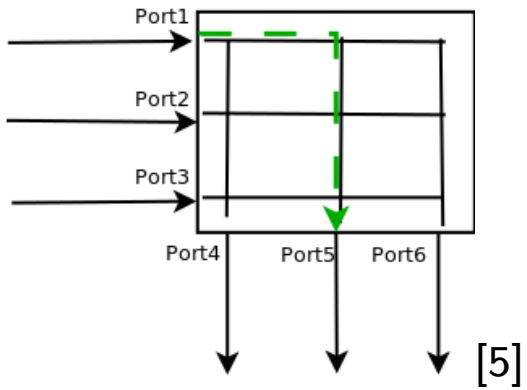


[5]

Figure 20 : Switching fabric - crossbar

Switching via crossbar

- A network of N incoming ports and N outgoing ports.
- Every port has its own bus, that can be interconnected when needed, that is 2N busses.
- An incoming frame is analysed and once the destination port has been found, the incoming bus will be connected to the outgoing bus.
- Allows multiple frames to be sent in parallel
- If two frames are destined for the same outgoing interface, one must be queued.
- This technique is used in the more advanced switches (Catalyst 12000)

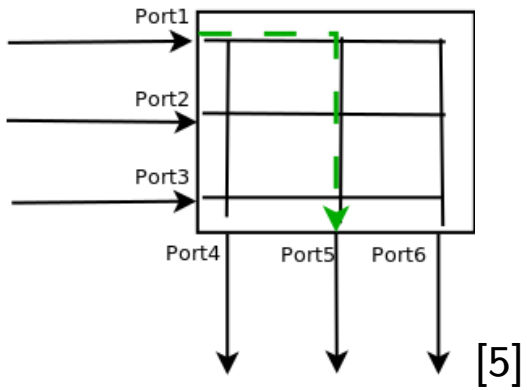


[5]

Figure 20 : Switching fabric - crossbar

Switching via crossbar

- A network of N incoming ports and N outgoing ports.
- Every port has its own bus, that can be interconnected when needed, that is 2N busses.
- An incoming frame is analysed and once the destination port has been found, the incoming bus will be connected to the outgoing bus.
- Allows multiple frames to be sent in parallel
- If two frames are destined for the same outgoing interface, one must be queued.
- This technique is used in the more advanced switches (Catalyst 12000)

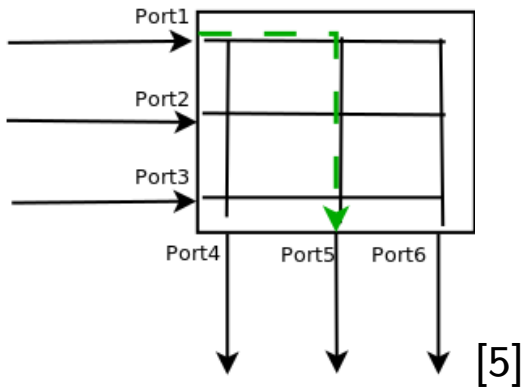


[5]

Figure 20 : Switching fabric - crossbar

Switching via crossbar

- A network of N incoming ports and N outgoing ports.
- Every port has its own bus, that can be interconnected when needed, that is 2N busses.
- An incoming frame is analysed and once the destination port has been found, the incoming bus will be connected to the outgoing bus.
- Allows multiple frames to be sent in parallel
- If two frames are destined for the same outgoing interface, one must be queued.
- This technique is used in the more advanced switches (Catalyst 12000)

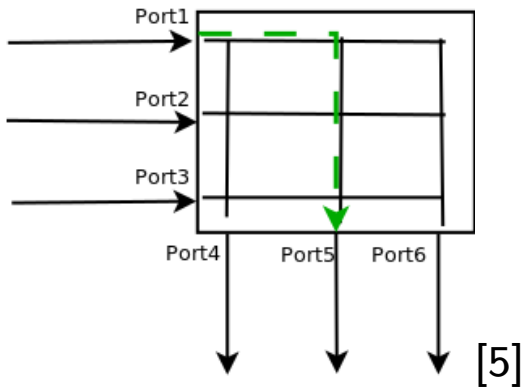


[5]

Figure 20 : Switching fabric - crossbar

Switching via crossbar

- A network of N incoming ports and N outgoing ports.
- Every port has its own bus, that can be interconnected when needed, that is 2N busses.
- An incoming frame is analysed and once the destination port has been found, the incoming bus will be connected to the outgoing bus.
- Allows multiple frames to be sent in parallel
- If two frames are destined for the same outgoing interface, one must be queued.
- This technique is used in the more advanced switches (Catalyst 12000)

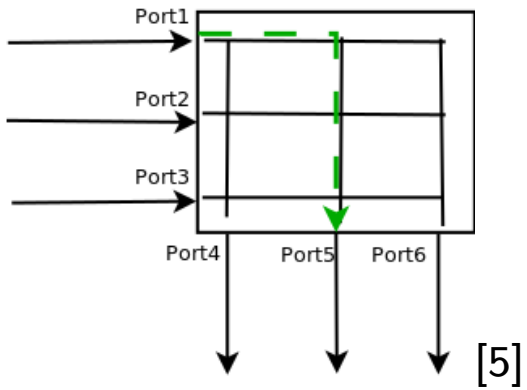


[5]

Figure 20 : Switching fabric - crossbar

Switching via crossbar

- A network of N incoming ports and N outgoing ports.
- Every port has its own bus, that can be interconnected when needed, that is 2N busses.
- An incoming frame is analysed and once the destination port has been found, the incoming bus will be connected to the outgoing bus.
- Allows multiple frames to be sent in parallel
- If two frames are destined for the same outgoing interface, one must be queued.
- This technique is used in the more advanced switches (Catalyst 12000)



[5]

Figure 20 : Switching fabric - crossbar

Switch forwarding methods

- Two types of switch forwarding methods.
- Store-and-forward.
- Cut-Through Switching.

Store-and-forward

- Stores the entire frame in a buffer.
- Calculates the frame checksum.
- If the frame is corrupted it will discard the frame.
- Decrease the number of corrupt frames sent on the network.
- Most commonly used method.
- Must be used when QoS is run.

Store-and-forward

- Stores the entire frame in a buffer.
- Calculates the frame checksum.
- If the frame is corrupted it will discard the frame.
- Decrease the number of corrupt frames sent on the network.
- Most commonly used method.
- Must be used when QoS is run.

Store-and-forward

- Stores the entire frame in a buffer.
- Calculates the frame checksum.
- If the frame is corrupted it will discard the frame.
- Decrease the number of corrupt frames sent on the network.
- Most commonly used method.
- Must be used when QoS is run.

Store-and-forward

- Stores the entire frame in a buffer.
- Calculates the frame checksum.
- If the frame is corrupted it will discard the frame.
- Decrease the number of corrupt frames sent on the network.
- Most commonly used method.
- Must be used when QoS is run.

Store-and-forward

- Stores the entire frame in a buffer.
- Calculates the frame checksum.
- If the frame is corrupted it will discard the frame.
- Decrease the number of corrupt frames sent on the network.
- Most commonly used method.
- Must be used when QoS is run.

Store-and-forward

- Stores the entire frame in a buffer.
- Calculates the frame checksum.
- If the frame is corrupted it will discard the frame.
- Decrease the number of corrupt frames sent on the network.
- Most commonly used method.
- Must be used when QoS is run.

Cut-Through Switching

Two variants

- Fast Forward
  - Only the first 14 bytes will be read (To find out the destination)
  - Once the destination has been known, the frame will be forwarded.
  - Low processing time.
  - Will send corrupted frames as well.
- Fragment-free
  - Will store the first 64 bytes of a frame.
  - Most of the collisions happen within the time it takes to send the first 64 bytes of the frame.
  - Decreases the number of corrupt frames sent.

Cut-Through Switching

Two variants

- Fast Forward
  - Only the first 14 bytes will be read (To find out the destination)
  - Once the destination has been known, the frame will be forwarded.
  - Low processing time.
  - Will send corrupted frames as well.
- Fragment-free
  - Will store the first 64 bytes of a frame.
  - Most of the collisions happen within the time it takes to send the first 64 bytes of the frame.
  - Decreases the number of corrupt frames sent.

Cut-Through Switching

Two variants

- Fast Forward
  - Only the first 14 bytes will be read (To find out the destination)
  - Once the destination has been known, the frame will be forwarded.
  - Low processing time.
  - Will send corrupted frames as well.
- Fragment-free
  - Will store the first 64 bytes of a frame.
  - Most of the collisions happen within the time it takes to send the first 64 bytes of the frame.
  - Decreases the number of corrupt frames sent.

Cut-Through Switching

Two variants

- Fast Forward
  - Only the first 14 bytes will be read (To find out the destination)
  - Once the destination has been known, the frame will be forwarded.
  - Low processing time.
  - Will send corrupted frames as well.
- Fragment-free
  - Will store the first 64 bytes of a frame.
  - Most of the collisions happen within the time it takes to send the first 64 bytes of the frame.
  - Decreases the number of corrupt frames sent.

Cut-Through Switching

Two variants

- Fast Forward
  - Only the first 14 bytes will be read (To find out the destination)
  - Once the destination has been known, the frame will be forwarded.
  - Low processing time.
  - Will send corrupted frames as well.
- Fragment-free
  - Will store the first 64 bytes of a frame.
  - Most of the collisions happen within the time it takes to send the first 64 bytes of the frame.
  - Decreases the number of corrupt frames sent.

Cut-Through Switching

Two variants

- Fast Forward
  - Only the first 14 bytes will be read (To find out the destination)
  - Once the destination has been known, the frame will be forwarded.
  - Low processing time.
  - Will send corrupted frames as well.
- Fragment-free
  - Will store the first 64 bytes of a frame.
  - Most of the collisions happen within the time it takes to send the first 64 bytes of the frame.
  - Decreases the number of corrupt frames sent.

Cut-Through Switching

Two variants

- Fast Forward
  - Only the first 14 bytes will be read (To find out the destination)
  - Once the destination has been known, the frame will be forwarded.
  - Low processing time.
  - Will send corrupted frames as well.
- Fragment-free
  - Will store the first 64 bytes of a frame.
  - Most of the collisions happen within the time it takes to send the first 64 bytes of the frame.
  - Decreases the number of corrupt frames sent.

Cut-Through Switching

Two variants

- Fast Forward
  - Only the first 14 bytes will be read (To find out the destination)
  - Once the destination has been known, the frame will be forwarded.
  - Low processing time.
  - Will send corrupted frames as well.
- Fragment-free
  - Will store the first 64 bytes of a frame.
  - Most of the collisions happen within the time it takes to send the first 64 bytes of the frame.
  - Decreases the number of corrupt frames sent.

Cut-Through Switching

Two variants

- Fast Forward
  - Only the first 14 bytes will be read (To find out the destination)
  - Once the destination has been known, the frame will be forwarded.
  - Low processing time.
  - Will send corrupted frames as well.
- Fragment-free
  - Will store the first 64 bytes of a frame.
  - Most of the collisions happen within the time it takes to send the first 64 bytes of the frame.
  - Decreases the number of corrupt frames sent.

Buffers

Two techniques are used for storing frames in a switch/router.

- Memory based
  - All ports will share the same memory.
  - Allows for larger frames, since the memory is dynamically allocated to each port.
  - Allows asymmetric switching, where frames arrive faster than they can be sent.
  - One connection might utilize the entire memory.
- Port based
  - Every port have its own buffer.
  - One frame can block an incoming interface, if the outgoing interface is busy.

Buffers

Two techniques are used for storing frames in a switch/router.

- Memory based
  - All ports will share the same memory.
  - Allows for larger frames, since the memory is dynamically allocated to each port.
  - Allows asymmetric switching, where frames arrive faster than they can be sent.
  - One connection might utilize the entire memory.
- Port based
  - Every port have its own buffer.
  - One frame can block an incoming interface, if the outgoing interface is busy.

Buffers

Two techniques are used for storing frames in a switch/router.

- Memory based
  - All ports will share the same memory.
  - Allows for larger frames, since the memory is dynamically allocated to each port.
  - Allows asymmetric switching, where frames arrive faster than they can be sent.
  - One connection might utilize the entire memory.
- Port based
  - Every port have its own buffer.
  - One frame can block an incoming interface, if the outgoing interface is busy.

Buffers

Two techniques are used for storing frames in a switch/router.

- Memory based
  - All ports will share the same memory.
  - Allows for larger frames, since the memory is dynamically allocated to each port.
  - Allows asymmetric switching, where frames arrive faster than they can be sent.
  - One connection might utilize the entire memory.
- Port based
  - Every port have its own buffer.
  - One frame can block an incoming interface, if the outgoing interface is busy.

Buffers

Two techniques are used for storing frames in a switch/router.

- Memory based
  - All ports will share the same memory.
  - Allows for larger frames, since the memory is dynamically allocated to each port.
  - Allows asymmetric switching, where frames arrive faster than they can be sent.
  - One connection might utilize the entire memory.
- Port based
  - Every port have its own buffer.
  - One frame can block an incoming interface, if the outgoing interface is busy.

Buffers

Two techniques are used for storing frames in a switch/router.

- Memory based
  - All ports will share the same memory.
  - Allows for larger frames, since the memory is dynamically allocated to each port.
  - Allows asymmetric switching, where frames arrive faster than they can be sent.
  - One connection might utilize the entire memory.
- Port based
  - Every port have its own buffer.
  - One frame can block an incoming interface, if the outgoing interface is busy.

Buffers

Two techniques are used for storing frames in a switch/router.

- Memory based
  - All ports will share the same memory.
  - Allows for larger frames, since the memory is dynamically allocated to each port.
  - Allows asymmetric switching, where frames arrive faster than they can be sent.
  - One connection might utilize the entire memory.
- Port based
  - Every port have its own buffer.
  - One frame can block an incoming interface, if the outgoing interface is busy.

Buffers

Two techniques are used for storing frames in a switch/router.

- Memory based
  - All ports will share the same memory.
  - Allows for larger frames, since the memory is dynamically allocated to each port.
  - Allows asymmetric switching, where frames arrive faster than they can be sent.
  - One connection might utilize the entire memory.
- Port based
  - Every port have its own buffer.
  - One frame can block an incoming interface, if the outgoing interface is busy.

Layer 2 and Layer 3 switching

- Lager 2 switches will only forward based on MAC address.
- Lager 3 switches will also use the IP-address.
- Difference between a layer 3 switch and a router?

Layer 2 and Layer 3 switching

- Lager 2 switches will only forward based on MAC address.
- Lager 3 switches will also use the IP-address.
- Difference between a layer 3 switch and a router?

Layer 2 and Layer 3 switching

- Lager 2 switches will only forward based on MAC address.
- Lager 3 switches will also use the IP-address.
- Difference between a layer 3 switch and a router?

Layer 2 and Layer 3 switching

- Lager 2 switches will only forward based on MAC address.
- Lager 3 switches will also use the IP-address.
- Difference between a layer 3 switch and a router?
  - Layer 3 switch often only supports Ethernet.

Layer 2 and Layer 3 switching

- Lager 2 switches will only forward based on MAC address.
- Lager 3 switches will also use the IP-address.
- Difference between a layer 3 switch and a router?
  - Layer 3 switch often only supports Ethernet.
  - Router supports other layer 1 standards (PPP, Frame Relay)

Layer 2 and Layer 3 switching

- Lager 2 switches will only forward based on MAC address.
- Lager 3 switches will also use the IP-address.
- Difference between a layer 3 switch and a router?
  - Layer 3 switch often only supports Ethernet.
  - Router supports other layer 1 standards (PPP, Frame Relay)
  - Both L3 and router can run a routing protocol (RIP, OSPF, EIGRP et cetera.)

Layer 2 and Layer 3 switching

- Lager 2 switches will only forward based on MAC address.
- Lager 3 switches will also use the IP-address.
- Difference between a layer 3 switch and a router?
  - Layer 3 switch often only supports Ethernet.
  - Router supports other layer 1 standards (PPP, Frame Relay)
  - Both L3 and router can run a routing protocol (RIP, OSPF, EIGRP et cetera.)
  - Router supports both NAT/PAT, can establish connections.

Layer 2 and Layer 3 switching

- Lager 2 switches will only forward based on MAC address.
- Lager 3 switches will also use the IP-address.
- Difference between a layer 3 switch and a router?
  - Layer 3 switch often only supports Ethernet.
  - Router supports other layer 1 standards (PPP, Frame Relay)
  - Both L3 and router can run a routing protocol (RIP, OSPF, EIGRP et cetera.)
  - Router supports both NAT/PAT, can establish connections.
  - This functionality has started to show up on L3 switches as well.

Questions?

Epost:lennart.franked@miun.se
Tel: 060-148683

References

[1] RAD Data Communication. Ethernet frame format and ieee 802.3, 2010. URL `http://www3.rad.com/networks/infrastructure/lans/etherform.htm`.

[2] DIX. The ethernet, September 1980.

[3] IEEE. Ieee standard ethertypes. URL `http://standards.ieee.org/develop/regauth/ethertype/eth.txt`.

[4] IEEE. The structure and coding of logical link control (llc) addresses: A tutorial guide, 2010. URL `http://standards.ieee.org/develop/regauth/tut/llc.pdf`.

[5] James F. Kurose and Keith W. Ross. *Computer networking : a top-down approach*. Pearson Education, Boston, [Mass.], 6. ed., international ed. edition, 2013. ISBN 0-273-76896-4.

[6] Wayne Lewis. *LAN switching and wireless : CCNA exploration companion guide*. Cisco, Indianapolis, Ind., 2008. ISBN 1-58713-207-9 (hardcover w/cd).

[7] Justin Menga. *CCNP, Practical Studies: Switching*. Cisco Press, 800 East 96th Street, 3rd Floor, IN, 2004. ISBN 978-1-58-720060-1.

[8] Andrew S. Tanenbaum and D. Wetherall. *Computer networks*. Pearson, Boston, 5th ed. edition, 2011. ISBN 978-0-13-255317-9 (hft.) (International ed.).