NETWORK INFRASTRUCTURES

TELECOMMUNICATIONS NETWORK MODEL

5 components:
- Terminals: in/out device that transmit/receive data via telecommunication network
- Telecommunication processors: support data transmission and reception between terminals/computers
- Telecommunication channels: the medium over which data are transmitted
- Computers/Phones: endpoints of telecommunication network
- Telecom. control software: control and manage activities and functions of telecom network

There are multiple types of telecom. network according to their size:
Wide Area Network (WAN), Metropoliton Area Network (MAN), Local Area Network (LAN), Virtual Private Network (VPN, enabled by IPSec technology)

CLIENT / SERVER

Client is the end device that connects via network to the Server that provides services. and help with procesing eg. website, web services etc.

TELECOMMUNICATION MEANS

Wired (twisted pair, coaxial, fiber) or wireless (microwave, satellites, cellular system, wifi) depending on scope eg. range.

long range

medium range

Taking into account cellular system, what leads to newer generation of technology (2G → 3G etc) was (and will be) newer "killer" application
eg. 3G was needed to overcome SMS generation, 4G for video streaming (not only data anymore), 5G for IOT (and mobile in general, low latency)

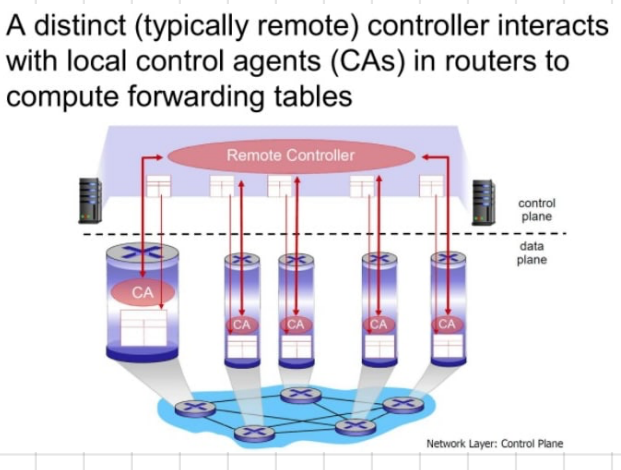
Personal Area Network: in few words the environment at home. BT, wifi, infrared etc. Included in LAN.

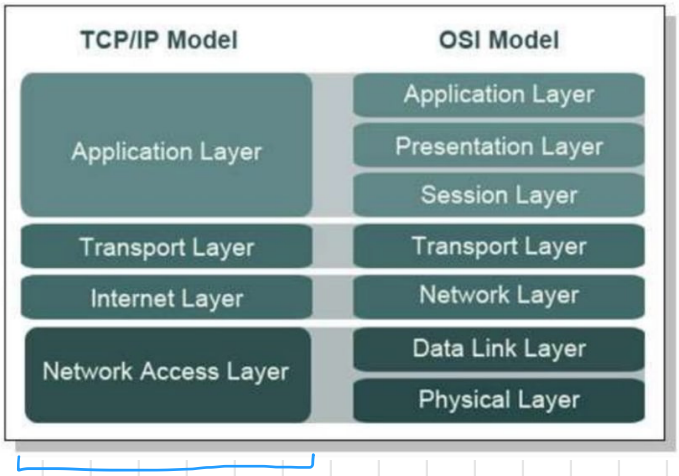
The goal of telecom network is to support the end user (client, human i.e. us). To do this it must be managed by someone (traffic, security, monitoring, capacity planning).

CONTROL PLANE & DATA PLANE

Example of control plane function is routing. Control plane works over data plane that instead implements the rules that data should follow.

Eg. Locally Centralized Control Plane

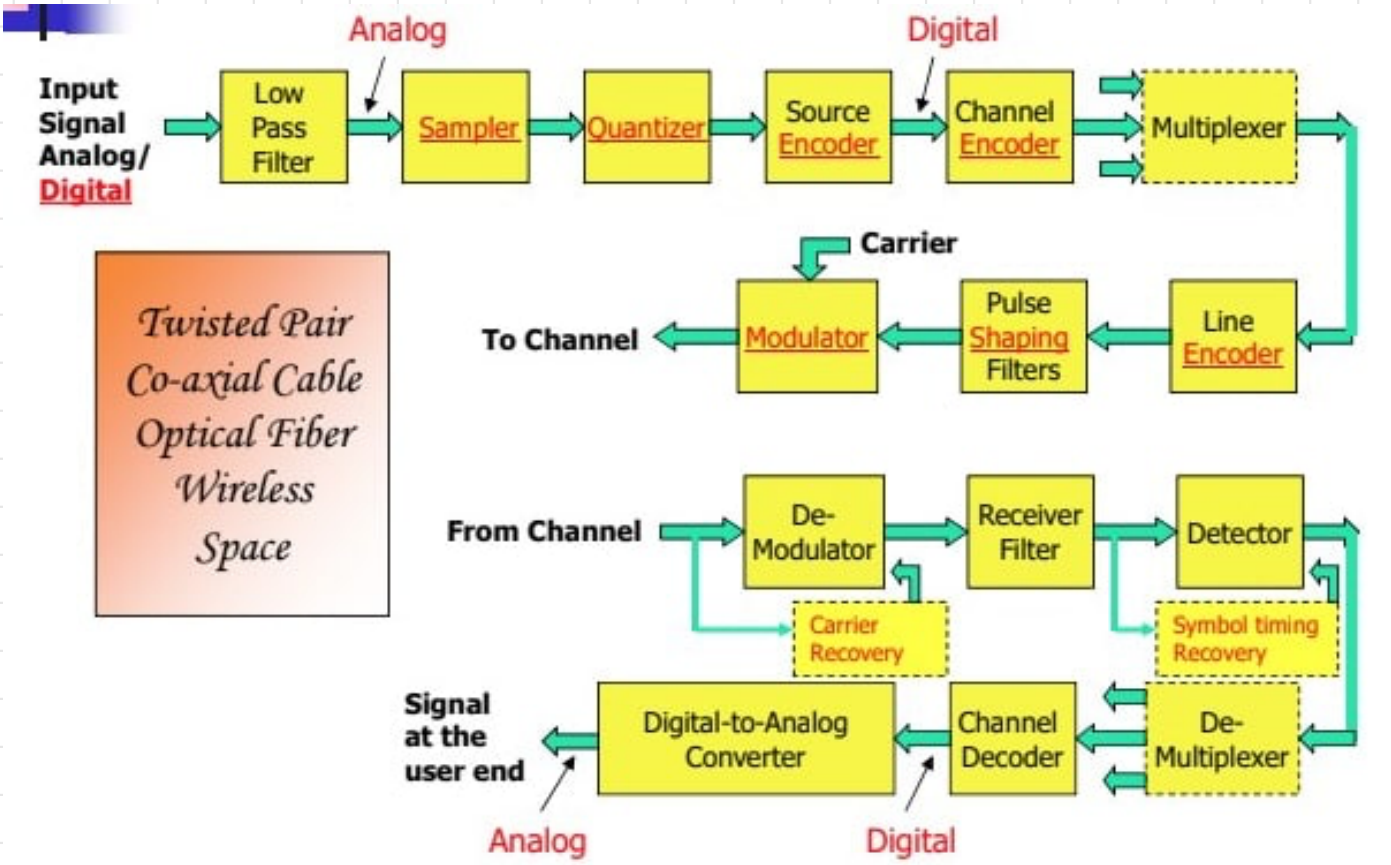




IP Devices

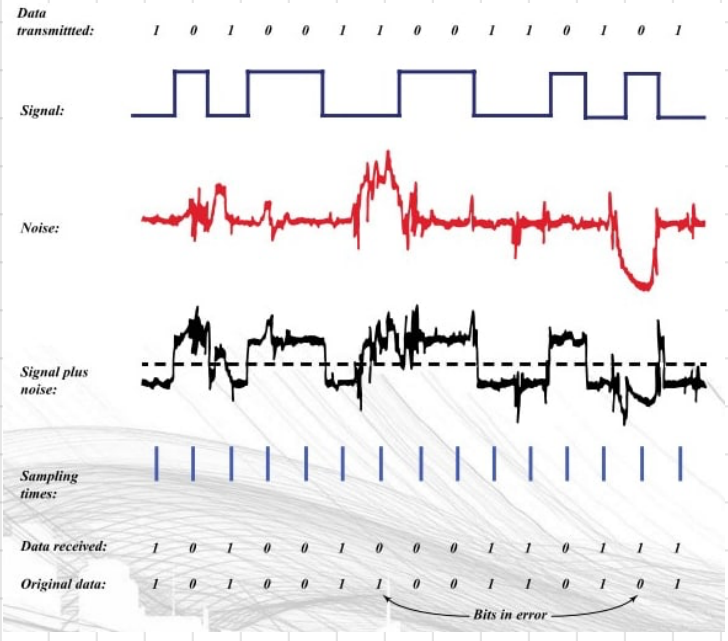
Rules that must be followed are called protocols and are organized as the architecture in picture.

DIGITAL COMMUNICATION SYSTEM



First, signal must be converted to digital signal (if not already) i.e. sequence of 1s and 0s.
Then digital stream must be passed to transmission media (modulation). Modulation adapts the digital signal to the transmision media used (e.g. light signals on fiber optic cable)
On opposite side: De-Modulation to retrieve digital stream and on other convertion to analog (if needed).
Note: in the media, the signal is analog!

Analog signals when received (in the past) were lowered (they are waves!) and was/is hard to get their original shape. On the other side, digital signal arrives as it is (or at least is easy to recover original shape)
Changes on shape are caused by noise.



Time instances (sampling times) are required to synchronize signal analysis.
Bit rate of media depends on quality of the media (BitErrorRate lower as possible) because if the noise is high we can't send big chunk of data (signals)

「Signal to Noise Ratio = $\frac{P_{signal}}{P_{noise}}$」

$$C = B \log_2 (1 + SNR) = \text{Shannon Capacity}$$

「bandwidth ↑ of the channel」

Bandwidth depends on wave spectrum. Must find tradeoff between range and bandwidth.

INTERNET OF THINGS (Seminar)

"Networked objects clear, standardized and interoperable communication protocols."

OR

"Pervasive presence around us of a variety of things or objects"

SMART OBJECT: minimal comm. functionalities, unique identifier, basic computing capabilities, interaction with environment (sensors or action triggers)

IoT can be implemented in many sectors, from smart home to smart cities or smart grids.

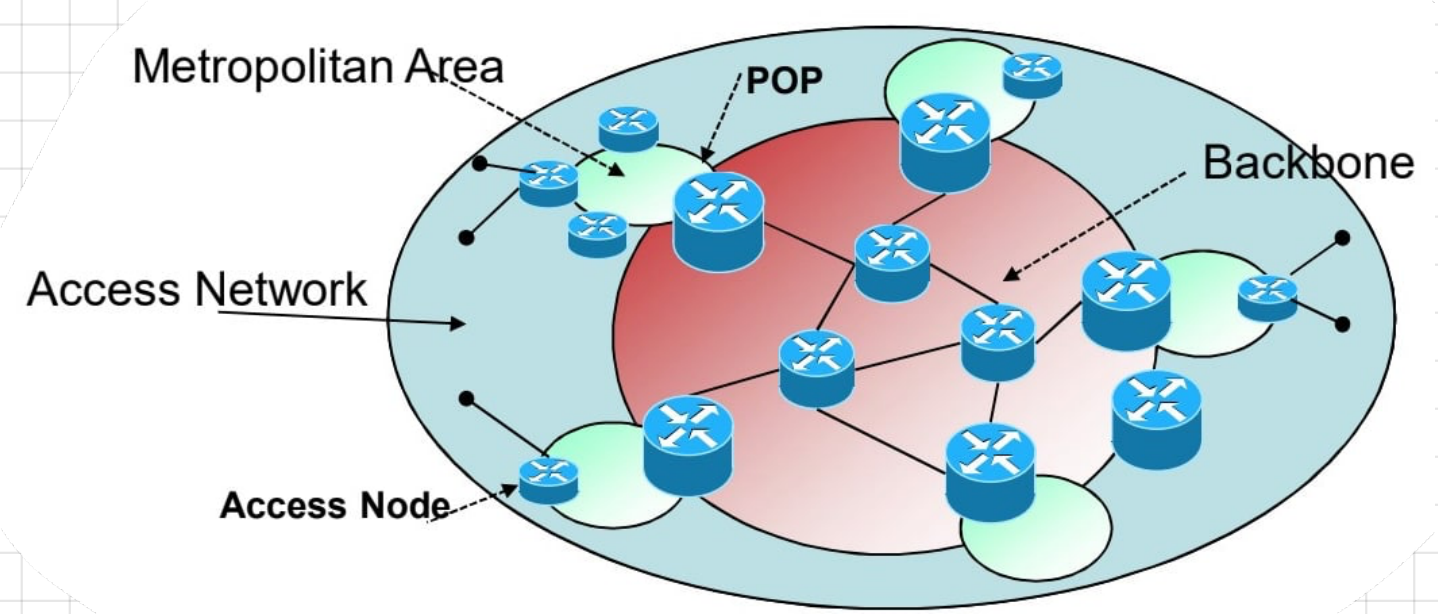
ARCHITECTURES

- CLOUD : applications as services over internet or data center hardware/software. Computing resource unlimited (in theory, pay-to-use), quick and scalable.
- EDGE: resource-rich servers close to end-users e.g. micro data-centers at wifi access point (possible also in car). Edge can contribute to form cloud and can also collect data. Main features: no time delay for decision (they are made right there!) and data can be analyzed without noise created during transfer to cloud.
- FOG: decended cloud. May include end-devices. Something like a "personal" cloud environment. Cloud is centralized while fog is usually distributed and decentralized. Offloads cloud.

TECHNOLOGIES

Short-range communication is fundamental. An example is Zigbee protocol. Differently from wifi it can work in P2P mode (central hub not needed). An alternative is Bluethooth Low Energy that can enable beacons functionalities (listen beacons signals from mobile devices).
Battery-less devices use elettromagnetic backscatter, inductive coupling, RFID, NFC to communicate each others or with other mobile devices.

NETWORK FUNCTIONAL AREAS



ACCESS NETWORK

Important role in a network by connecting communication couriers and service providers to subscribers. Originally based on copper-lines (DSL) now mostly on fibers. The presence of series of wires, switches etc makes it one of the most complex network in the world.

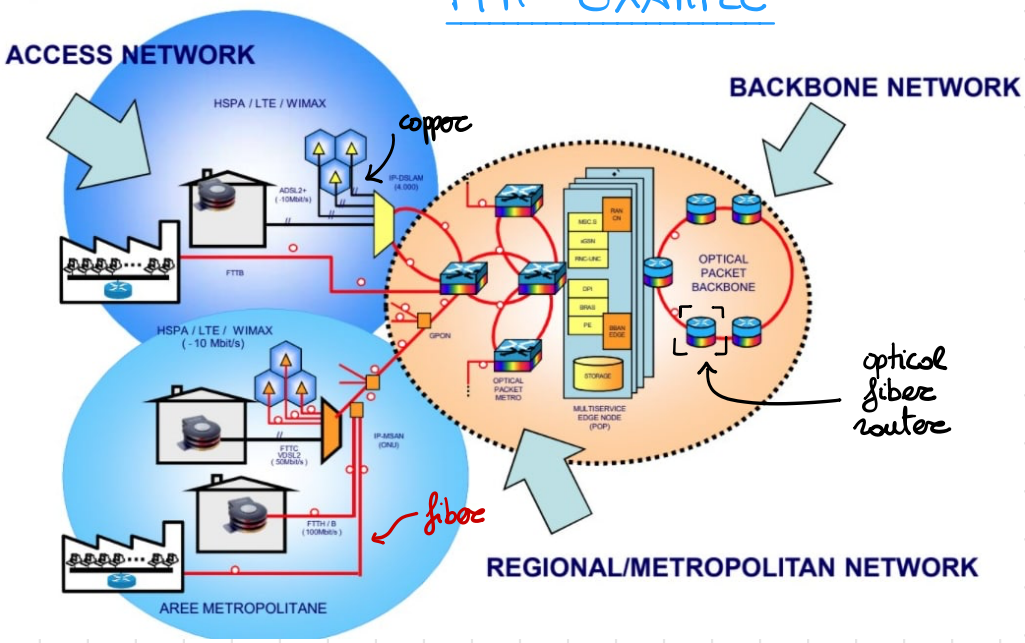
CORE NETWORK

It is the backbone network. Internet could be considered a giant core network.
Made by mostly router that connect multiple networks together with mesh topology.
Contrary to access network here robustness is fundamental!

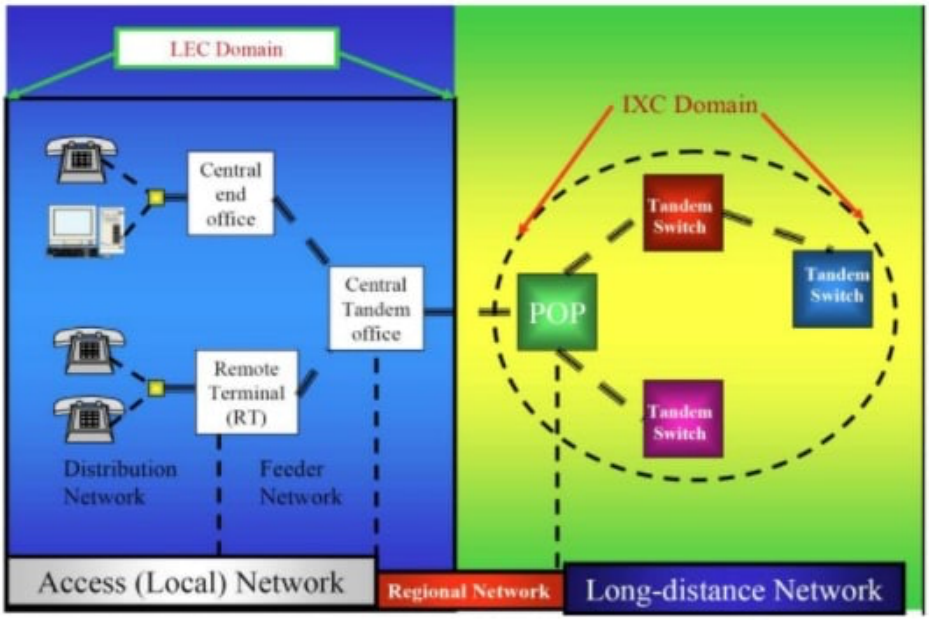
EDGE NETWORK

Allows more intelligent functions than core network (because core network manage an huge amount of data at high speed). Edge network can perform more specific packet scanning and find a more suitable route for it

TIM EXAMPLE



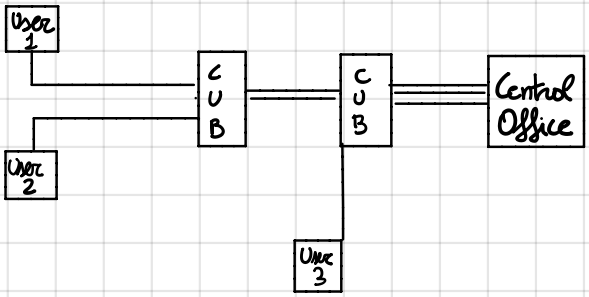
Core network uses RING and MESH topology, access network uses STAR topology and edge network uses RING topology in general cases.



NETWORK TERMS

- Exchange Area
  - Local vs long distance
- LEC – Local Exchange Carrier
- ILEC – Incumbent LEC
- CLEC - Competitive LEC
- Trunks – fiber optical
- CO - Central Office
- LATA – Local access and transport area
- IXC – Inter-exchange Carrier
  - Carry inter-LATA traffic

DISTRIBUTION NETWORK : cable aggregator, no switching. Composed by cable boxes on street (armodi = cabinet/urb/cub).

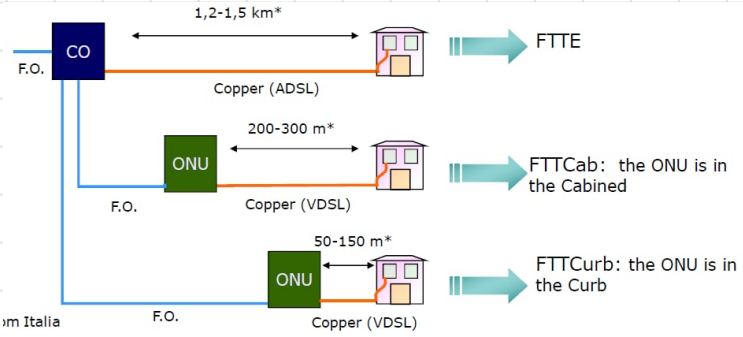


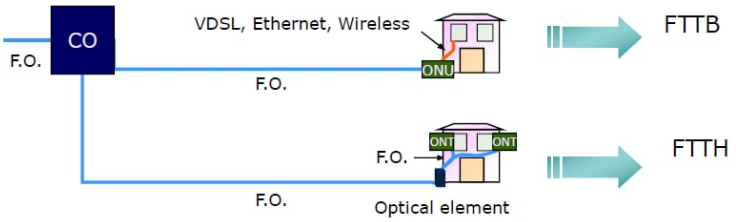
Local Loop (="last mile" network)

Unbundling : the natural monopoly (TIM in our case) is forced to rent the infrastructures to competitors (e.g. Fastweb)

- FTTE - Exchange
- FTTH - Home
- FTTC - Curb
- FTTN - Node or Neighborhood
- FTTP - Premise
- FTTB - Building or Business
- FTTU - User
- FTTZ - Zone
- FTTO - Office
- FTTD - Desk

Fiber cables are splitted using SPLITTERS (in passive optical network = PON) while in active optical network (AON) an active node (something like a switch) is used.





Optical Network Unit : connect optical lines to copper lines.
The longest the copper line part is the slowest the speed will be at final position.

Wireless access enjoys the highest expectations from the standpoint of ubiquitous networking, but is also very costly. Wireless include also cellular lines. This method requires lot of more intermediate nodes (RNC, BSC, GGSN, MSC ...) and have to manage multiple features e.g. mobility

Industrial Scientific Medical Bands : free to use network bands, usually low frequency (IOT, LORAWAN)

XDSL

Digital Subscriber Lines are used in the access part of the network, in particular is a family of technologies that provide digital data transmission over the wires of a local telephone network.

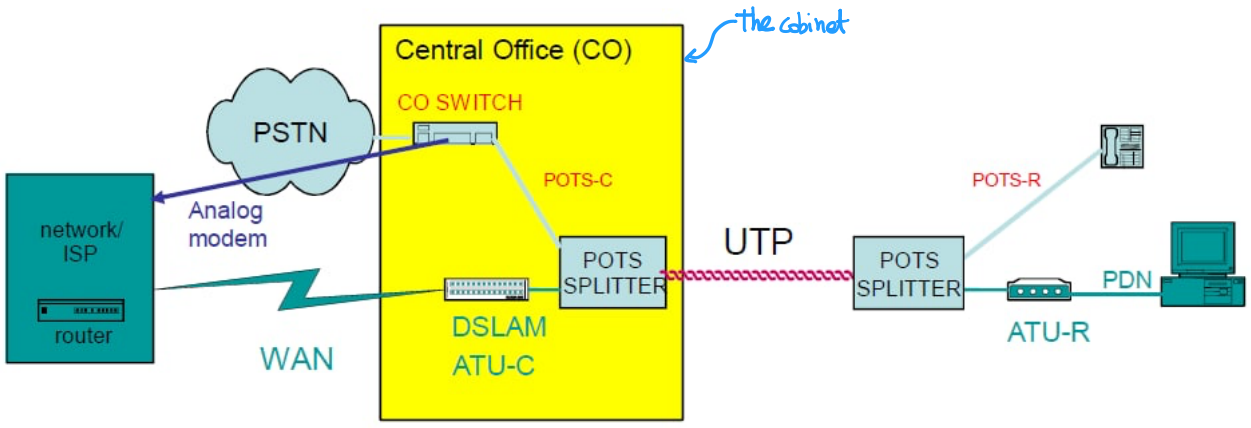
The X is replaced with different letters to identify the different technologies e.g Asymmetric DSL a DSL version with slower upload speed.

The reason of xDSL is purely because countries were/are full of copper lines.

ADSL uses two separate frequency bands, referred to as the upstream and downstream bands: the channel capacity is given by Shannon formula: $C = \text{bandwidth} \log_2(1 + SNR)$ and since every user has a "private" copper wire is possible two divide the two streams.

The problem, again, is that speed depends on distance: more distance, more attenuation → slower speed!

Anyway ADSL2+ (newer version) optimized the problem a bit (from 8 Mbit/s of ADSL to 24 Mbit/s)



POTS Splitter: separates POTS from DSL signals → separates telephone and data

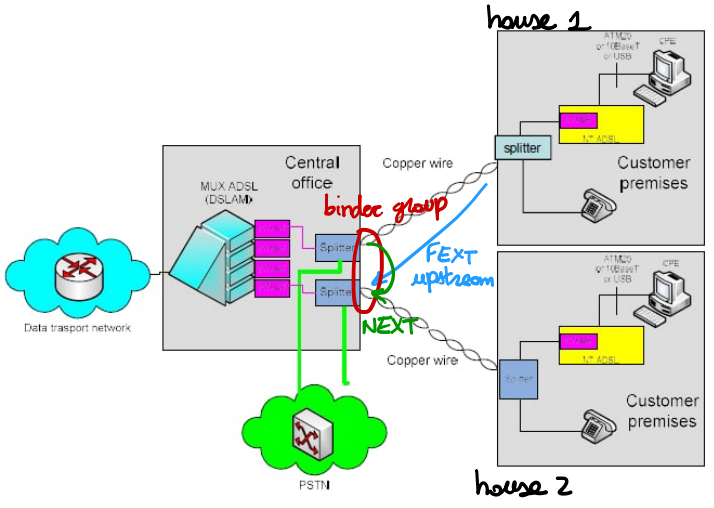
ATU-R: ADSL Termination Unit – Remote side

ATU-C: ADSL Termination Unit Central office side

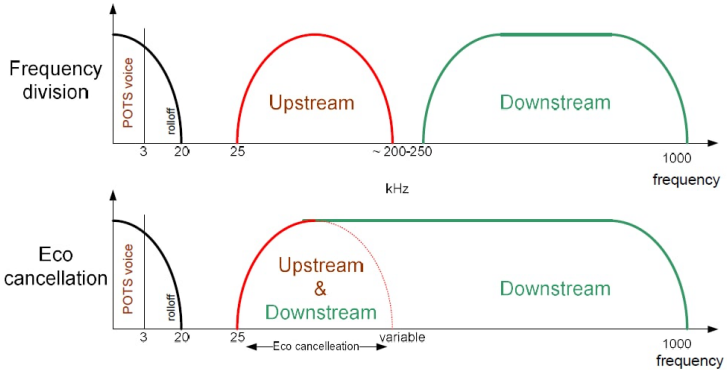
"twins routers"

DSLAM: DSL Access Multiplexer

Adsl copper wires that share some cable (binder group) interfere each others (some space, time and frequency) = cross-talks. FEXT (far end crosstalk) is a cross-talk between a trasmitter and a receiver placed on opposite sides of cable. The fact that is far lead to an attenuated interference. NEXT instead is "near end" between transmitter and receiver on same side of cable (higher interference)



Not possible to have FEXT downstream because the two receiver (houses) are not close enough (different binder groups)

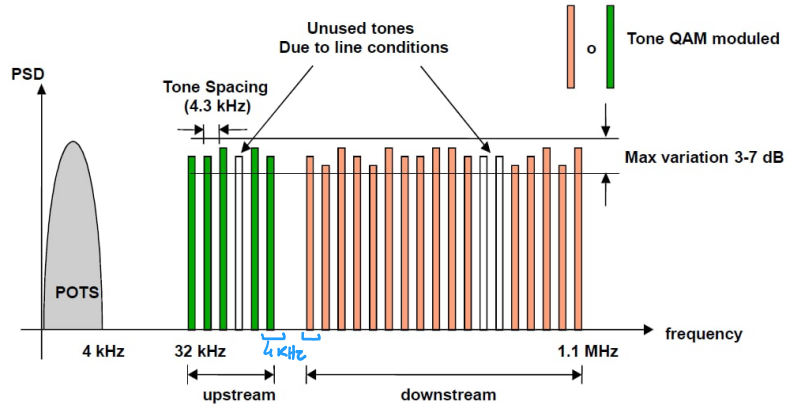


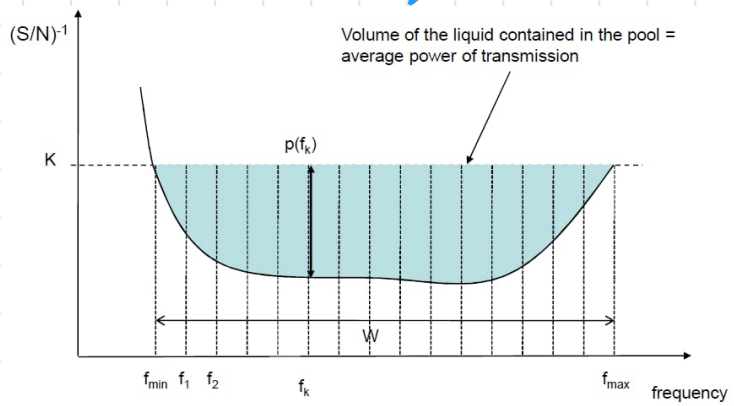
In the first configuration NEXT is not possible since the interferrence happens only in some time, space AND FREQUENCY.

MODULATION

Divide ADSL bandwidth is small subchannel (carrier / tones), so small that the behaviour inside them can be considered flat.

Water Filling

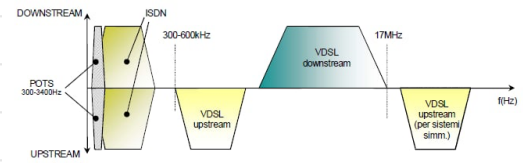




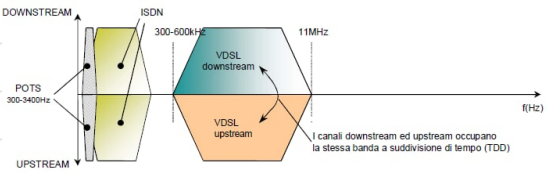
The signal power of each subcarrier is determined as the depth of the liquid in a pool. Knowing the discrete values $p(f_k)$ of each subcarrier f_k, one may deduce the number of bit per simbol to associate to the QAM costellation used in each subchannel (Discrete MultiTone modulation).
If a tone experiences some interferences it may not be used in favor of other subchannels.
Important to notice is the fact that ADSL protocol architecture is composed by multiple layers of different protocols. One example is Point to Point Protocol: used with ADSL for its ability to connect the user with the central office (authorization, automatic interface configuration and DHCP support).
In the same way PPP is composed by multiple layers: Link Control Protocol that establish, control and terminate links and Network Control Protocol, a family of protocols used to configure the network layers.

VDSL

Fiber optic from central office to a middle component called Optical Network Unit and from ONU copper wires to the edge. Practically speaking is like FTTC.
As ADSL two bandwidth solutions are possible.



FDD solution



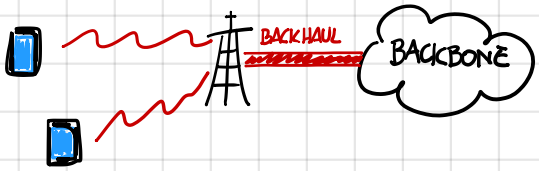
TDD solution

VDSL allows higher down/upload speed also on longer distances
The high frequencies used in VDSL generate interferences (crosstalk)
To avoid this VDSL Vectoring is applied:
1. using specific messages between DSLAccessMultiplexer and modem, for each line the noise is calculated.
2. DSLAM calculate the crosstalk between each line → high CPU usage

③ "anti-noise" signal generated to compensate crosstalk, almost completely.

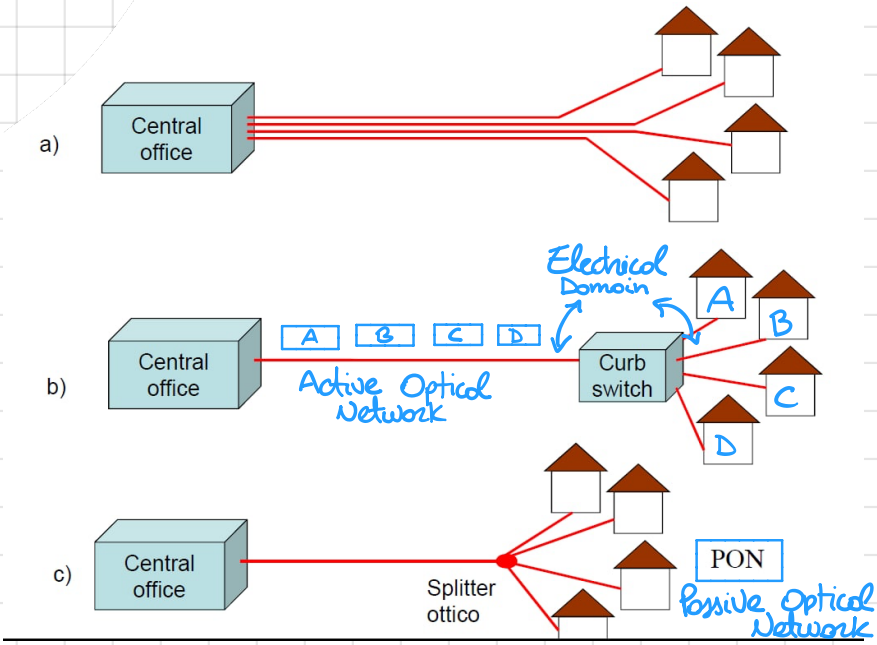
PASSIVE OPTICAL NETWORK

BACKHAUL: portion of network between backbone network and edge network
Backhaul requires high level performance in terms of bitrate because aggregate traffic from/to several users.



For this reason fiber cables are used. Different "architecture" are possible: FTTC, FTTH etc.

How? Possible ways:



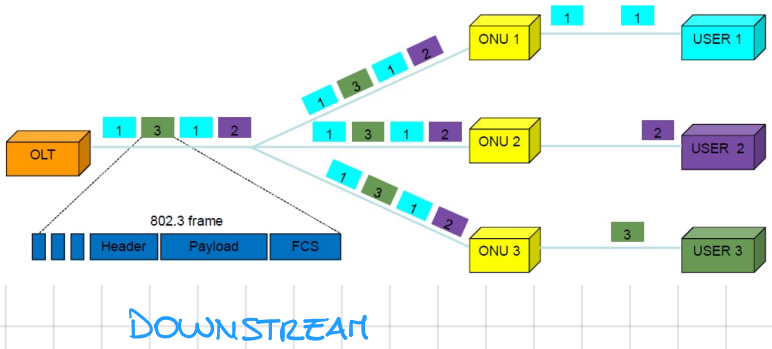
PRO: High bitrate for each user, robustness
CONS: cost, lot of cables

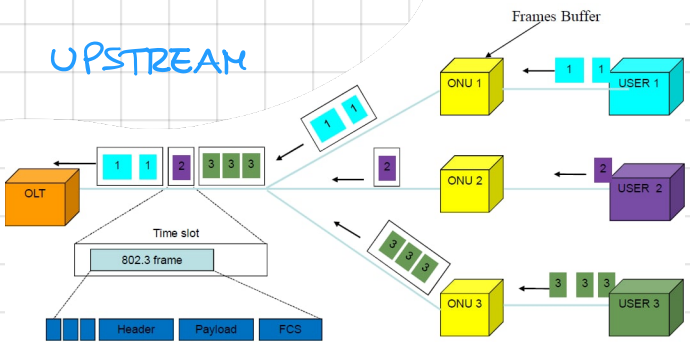
PRO: cheaper than (a) in terms of cable placement
CONS: curb needs power and manteinance.

Splitter used to "fuse" together multiple cables. Not needing any active switch it is way cheaper than (b)

In PON each user share the same bandwidth but this is irrelevant due to the very high bandwidth of fiber cables (respect to copper wires)

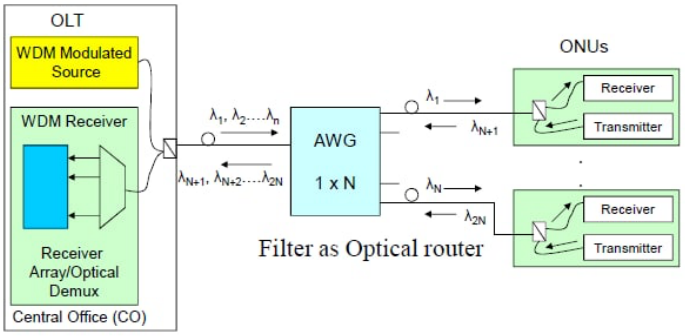
PON then become Ethernet PON when all packets are encapsulated in ethernet frames.





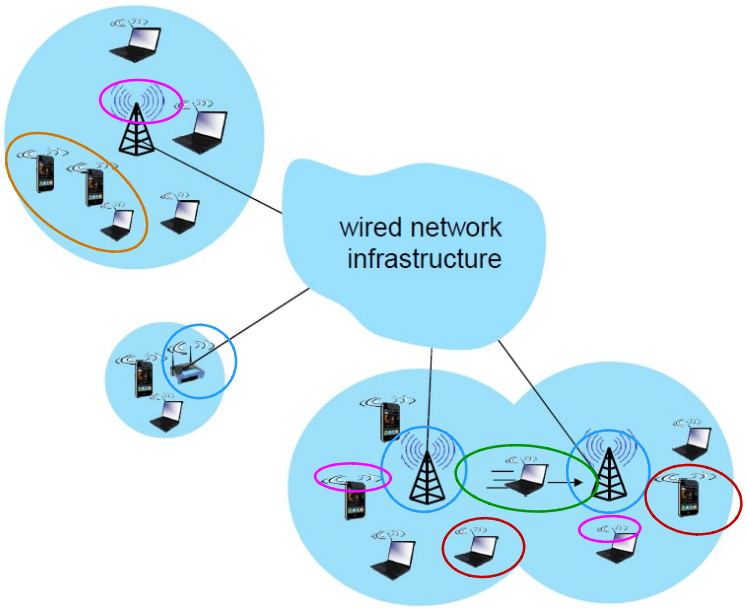
To avoid collision on upstream packets, PON standards propose Time Division Multiplexing Access assigning different timeslots to each ONU and "modulating" each frames to the same power level.
PON family evolved over the years from A-PON, B-PON, E-PON until Gigabit-capable-PON

WDM - PON



The idea is to have multiple wavelengths on some fiber cable (upstream and downstream)
This remove the TDM not requiring anymore the time slot coordination.
In this way a single fiber cable can carry multiple wavelength without the need of any active switch.

WIRELESS INFRASTRUCTURE



- Base station: connected to wired network, sends packets between wired network and wireless hosts
- Wireless host: stationary or mobile, where applications run
- Wireless link: connects mobile to base station also used as backbone link. Multiple access protocol, various transmission rates, distances and frequency bands
- Infrastructure mode: mobiles can change base station in mobility (handoff)

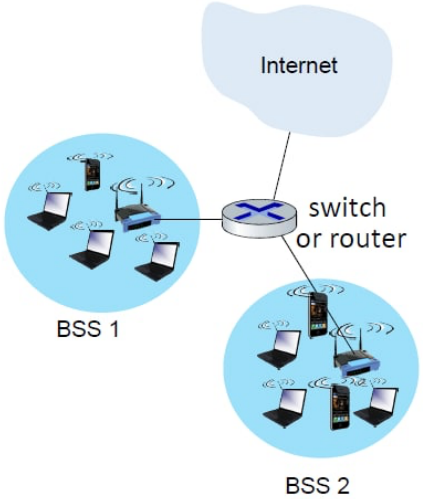
- Ad Hoc mode allows nodes to organize themselves as network without base station and connect with eachother eg. Bluetooth

Wireless links differ from wired ones for:
- lower signal strength → attenuation during propagation
- presence of interferences i.e. shared frequencies (e.g. 2.4 Ghz), hidden terminal problem
- multipath propagation → radio signals arrive at destination at different times due to reflection on objects.

This make communication over wireless link much more difficult, but
- larger SNR make easier to extract signal from noise.

Code Division Multiple Access: each user has a "code". Sharing the same frequency, interferences are minimized encoding their own data with their own code (chipping sequence)
ENCODING → inner product: original data X chipping sequence
DECODING → summed inner product: encoded data X chipping sequence

802.11 WIRELESS LAN

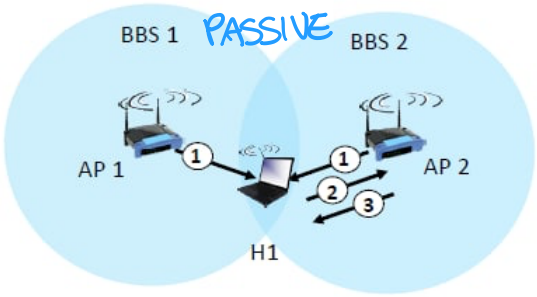


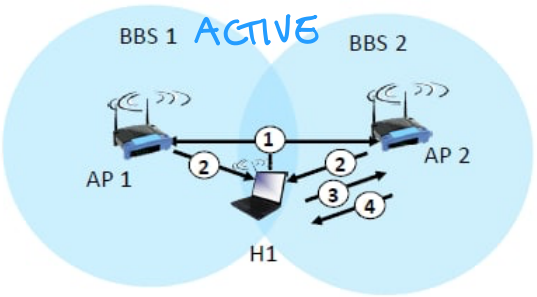
BSS = Basic Service Set aka "cell"

Spectrum divided into channels at different frequencies, AP frequency chosen by AP admin

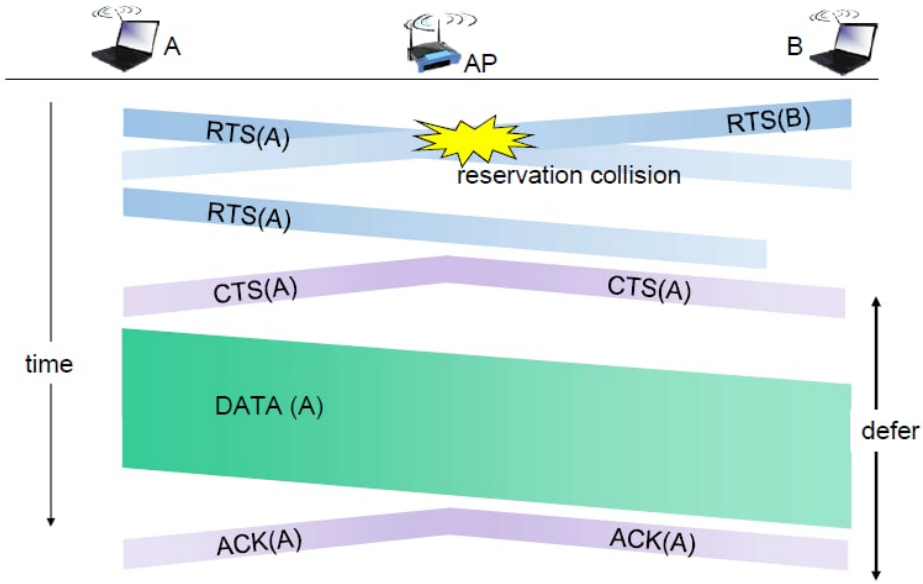
Arriving host must associate with an AP: scan for AP's name (SSID) and MAC, select AP, authentication, DHCP and so on.

Scanning can be passive (started by APs) or active (started by the device)





COLLISION AVOIDANCE RTS-CTS exchange



RTS = request to send
Short and can collide without problems
CTS = clear to send
When received by all nodes, other stations defer transmissions.

This solve also the hidden terminal problem.

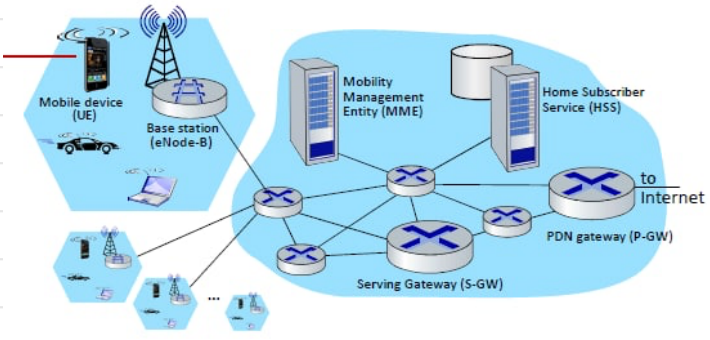
4G/5G CELLULAR NETWORK

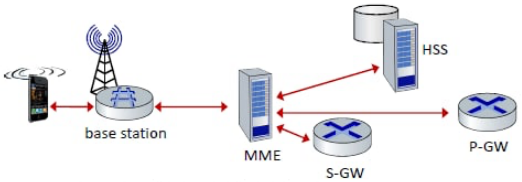
Both technologies are standardized by 3GPP. Differently from wired internet ofcourse we have wireless link layer, mobility as 1st class service, user identity via SIM card and worldwide access with authentication infrastructure

Base station in 4G LTE architecture il called eNodeB = enhanced node and aims to manage wireless radio resources and mobile devices (called User Equipment) in its coverage area ("cell")

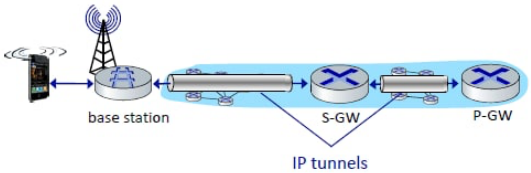
Home Subscriber Service stores info about UE for which HSS's network is their "home network".

Mobility Management Entity manage device authentication





control plane
- new protocols for mobility management, security, authentication (later)



data plane
- new protocols at link, physical layers
- extensive use of tunneling to facilitate mobility

1. BS broadcast synch signal every 5 ms on all frequency
2. device find the primary synch signal(s) and locates a 2nd synch signal on some frequency
3. Choose the BS to which connect to, home carrier preferred
4. authentication, establish state etc.

NAMEX SEMINAR (IXP)

INTERNET EXCHANGE POINT

Essential infrastructure to exchange internet traffic between autonomous systems.
IXP allows to ISPs' networks to be interconnected. NAMEX is one of the few IXP in Italy
As a network of networks, the internet critically depends on adequate interconnection between the different participants in the internet ecosystem.
We can have two types of relationships:

- TRANSIT: basically allow internet traffic of AS to pass through / connect to other(s), paying.
- PEERING: physical / logical connection between ASs that permit to mutually exchange of traffic without charge.

Peering for sure is the best of the two, with multiple advantages.

PEERING PLATFORM: the switching technology used by the IXP. IP fabric used by NAMEX.
Spine nodes (main) are connected to every leaf node. The use of IP as "fabric" protocol, paired with BGP for dynamically controlling data paths, ensures stability and semplicity, overcoming L2 meshed networks issues.

ROUTE SERVER: every participants in exchange node just connect to the route server to establish a connection with each other instead of establish a "private" connection.
With a single BGP connection, an AS can connect to every other ASs connected to the IXP. Faster.

Peer connections are more secure than passing through multiple ASs

OPTICAL NETWORKS

A public network is divided into:
- Access network: used by endusers generally. ADSL, wifi etc.
- Interoffice network: connected to access networks via Central Office node. This is the so called WAN. Here starts the transport network, where generally we cant find source of data.
- Interexchange network: connect networks over very long distances eg. US↔EU. Mesh.

Amount of data increases going from access network to interexchange network, because multiple networks add up to a single one.
Paths from an access network to an other may change for many reason eg. congestion.
Routing is the path creation function.

SERVICES

- connection oriented: sender and receiver connects each other before communication
- connectionless: send data whenever i want.

Core devices work according to one of these paradigm:
- circuit switching: static multiplexing
- packet switching: statistical multiplexing

Circuit switching creates a temporary connection between hosts with dedicated bandwidth guaranting the quality and latency but wasting unused bandwidth. Packet switching instead allow users to equally use bandwidth but makes no promises concerning quality and latecy.

Optical networks (ONs), basically fiber, can deliver bandwidth in flexible manner.
To enhance transmision capacity on a fiber there are two ways:
- Time Division Multiplexing
- Wavelength Division Multiplexing

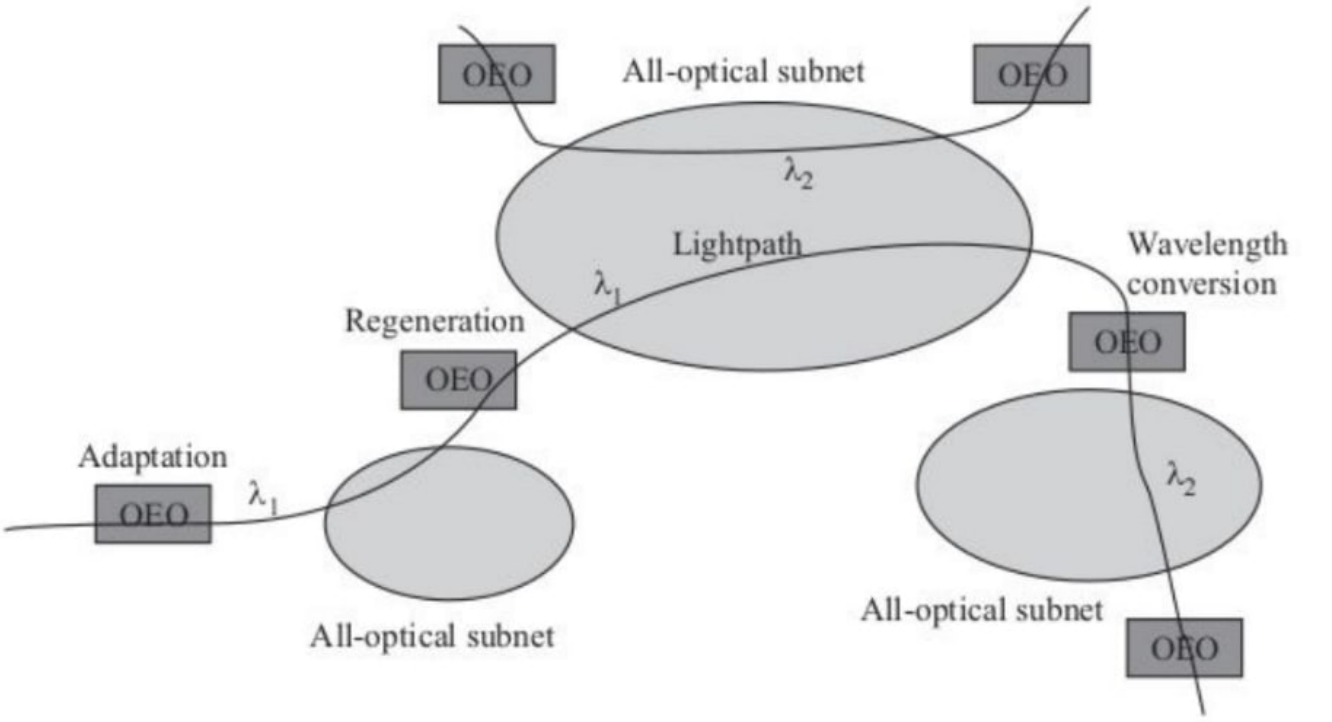
The first one divides time slot to network users while the second one assigns to each of them a different wavelength.

Second generation ON use the second one: also known as wavelength routed network.
The main idea is to incorporate some of the switching and routing functions into optical domain of the network. Over the same fiber cable is possible to have multiple lightpaths with different wavelength (i.e. something like frequency)
Electrical layer is used to convert frequencies only!
Optical network is the first layer of TCP communication (transport layer).

Network elements that enable optical network are: OLTs, OADMs, OXCs.
Optical Add/Drop Multiplexer is mostly used in ring topology where selectively drop some of incoming wavelengths locally while letting other pass through. The selectively add wavelength to the outbound signal.

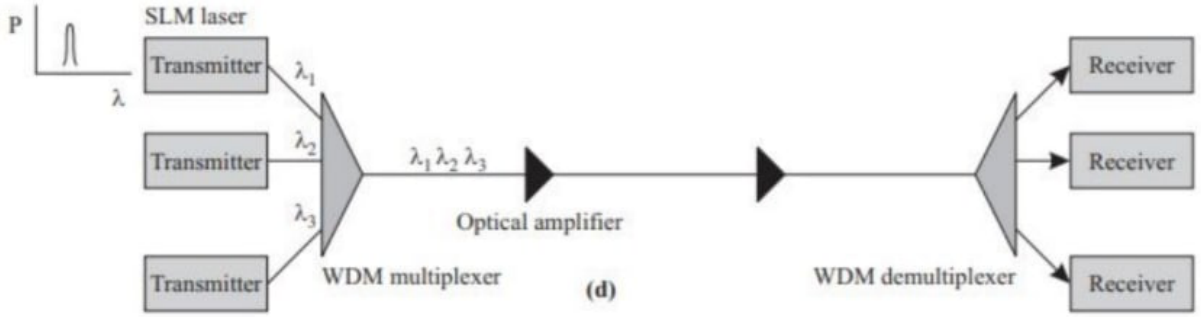
SERVICE TRANSPARENCY

The ligthpath can accomodate different types of services once set up.
Having an all-optical network anyway is very difficult because some electrical layer will be always present. Electronics play a crucial role in performing the intelligent control and monagement functions.
ADAPTATION: adopt the signals entering the optical domain (electrical-to-optical conversion)
REGENERATION: enhance the signals between two optical (sub)net and convert wavelengths (core)



Regeneration can be of 3 types:
- 1R (optical amplifier): increase the signal power, support analog signals but has poor performance
- 2R regen. + reshaping: reshape the signal reducing noise restoring original wave.
- 3R regen + reshaping + retiming: produce a "fresh" copy of the signal with correct timing and shape but loosing transparency.

Each client technology has its own shape so every client technology needs its own regenerator reducing transparency. In this case transparency means that regenerated data cant be distinguished from original one.

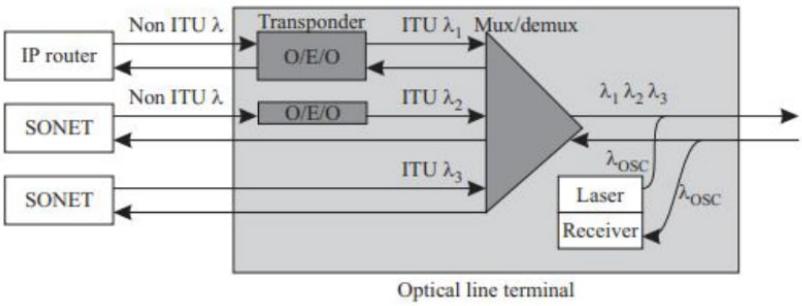


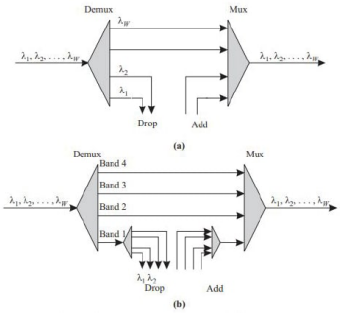
Current topology.

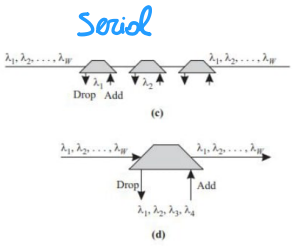
WDM NETWORK ELEMENTS

OPTICAL LINE TERMINAL

Used at either end of a point2point link to multiplex and demultiplex wavelengths.







Serial

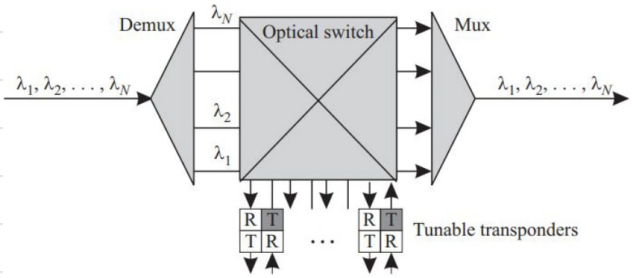
Parallel

OADM

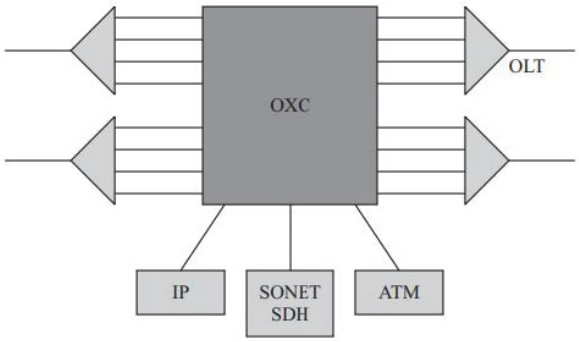
Cost-effective means for handling passthrough traffic in both metro and long-haul network.

RECONFIGURABLE OADM

Like OADM but reconfigurable.
Reconfigurability refers to the ability to select the desired wavelengths to be dropped and added on the fly.



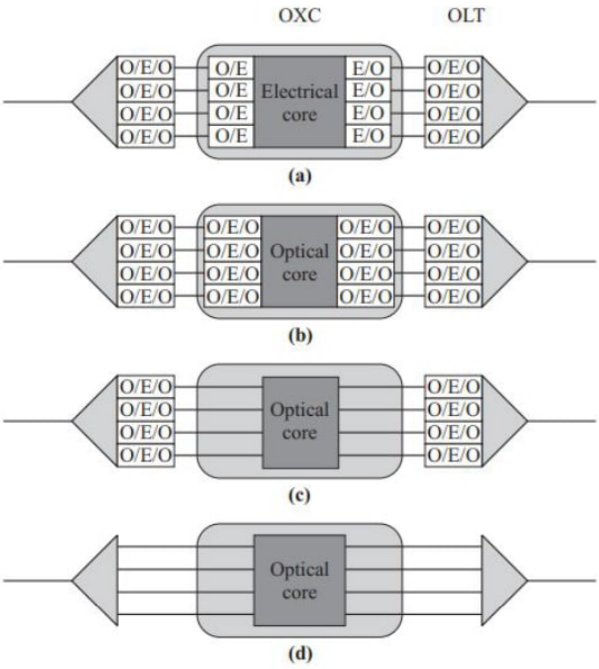
OPTICAL CROSS CONNECTS (OXC)



OXC enables reconfigurable optical networks, where lightpaths can be set up and taken down as needed. Allows to handle complex topologies and large number of wavelengths.

Service provisioning, protection, bit rate transparency, performance monitoring, fault localization, wavelength conversion, multiplexing and grooming are the features available thanks to OXCs.

Different configurations are possible: OEO or OE/EO switching, Electrical or Optical core.



Attribute	Opaque Electrical Fig. 7.11(a)	Opaque Optical with O/E/Os Fig. 7.11(b)	Opaque Optical Fig. 7.11(c)	All-Optical Fig. 7.11(d)
Low-speed grooming	Yes	No	No	No
Switch capacity	Low	High	High	Highest
Wavelength conversion	Yes	Yes	Yes	No
Switching triggers	BER	BER	Optical power	Optical power
Interface on OLT	SR/VSR	SR/VSR	IR/serial VSR	Proprietary
Cost per port	Medium	High	Medium	Low
Power consumption	High	High	Medium	Low
Footprint	High	High	Medium	Low

These are some examples of posible configurations of OXCs.

CONTROL AND MANAGEMENT

Different functions are available:

- Performance monitoring: monitor current status of network. For example check if Bit Error Rate is lower or greater of an allowed threshold. SNR and BER are the parameters under control
- Fault management: detect failures, isolate faulty components and find new path available (lightpath)
- Configuration management:
  - connection management: setup, take down and keep track connections
  - adaptation management: convert external signals for the optical layer.
- Security and accounting management

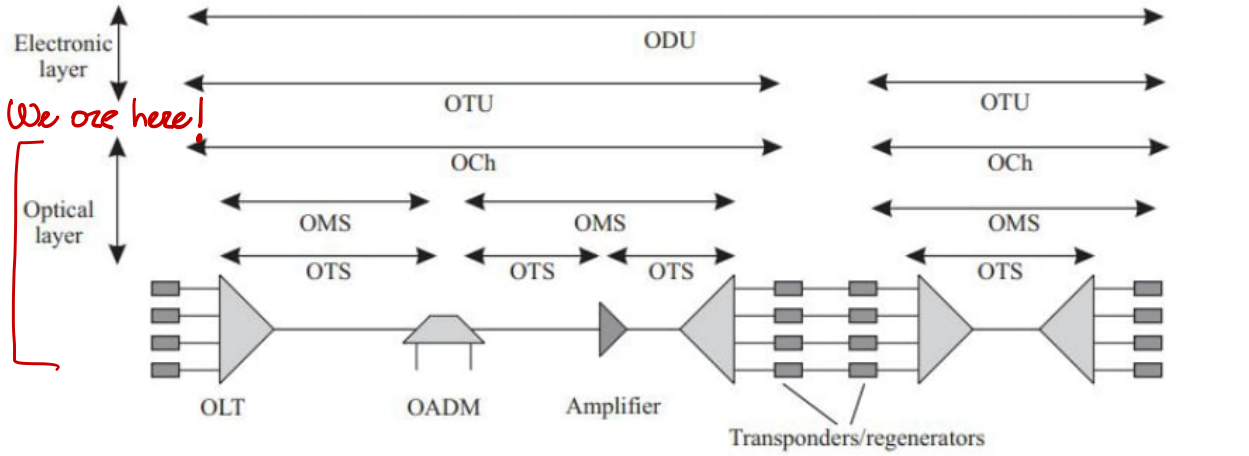


Figure 8.2 Layers within OTN. The optical layers are the optical channel layer (OCh), optical multiplex section (OMS) layer, and the optical transmission section (OTS) layer. The electronic layers are the optical channel data unit (ODU) layer and the optical channel transport unit (OTU).

All of this provides on demand lightpath setup, bandwidth negotiation, adaptation, guaranted BER, multiple levels of protection, unidirectional and bidirectional lightpaths, multicast support and requirements on jitter and maximum delay.

The available approaches are client-server and peer model for centralized or distributed control.

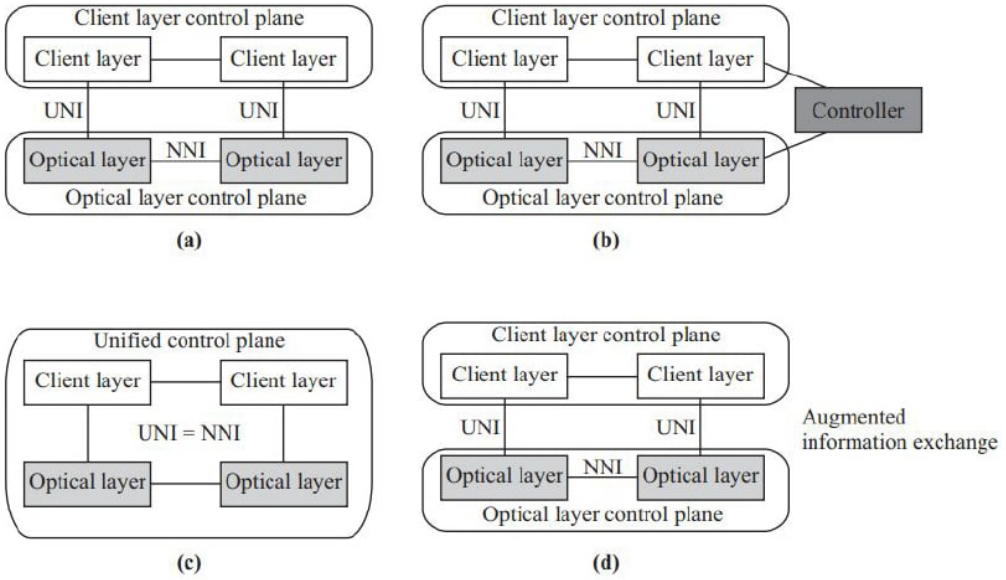


Figure 8.9 Different control plane models for interconnecting client layers with the optical layer. (a) Overlay model, (b) overlay+ model, (c) peer model, and (d) augmented model.

About distributed connection each node monitor the status of link sending "hello" message periodically, and routing algorithms are used on topology database stored in each node.

MULTI PROTOCOL LABEL SWITCHING (MPLS)

DATAGRAM VS VIRTUAL CIRCUIT PACKET NETWORKS

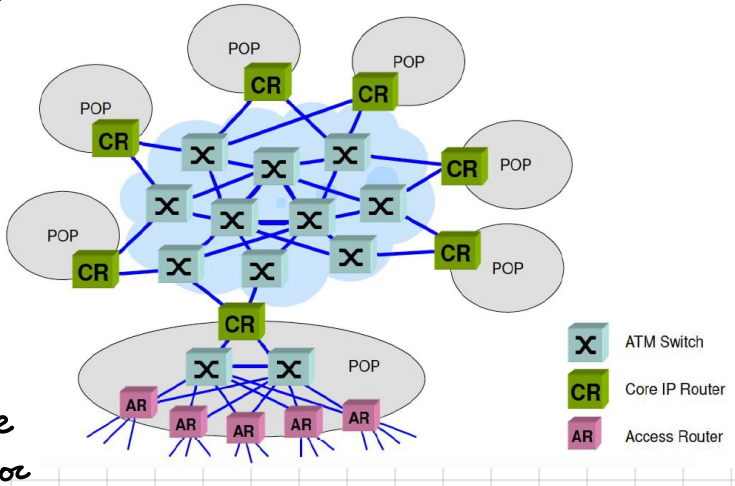
A network provider needs Quality of Service for its customers (Service Level Agreement), wants to run its backbone in a cost-effective way with convergence of services (e.g. voice and data on same infrastructure) and having good support for VPN, traffic engineering and protection/restoration mechanisms.
→ Connection Oriented packet switching (virtual circuit)

In datagram packet networks (connectionless) two consecutive packets of the same traffic can follow different path while in virtual circuit (connection oriented) they follow the same path (routing is chosen and fixed during connection setup phase).

- Data Plane (User Plane) = forwarding of data packets
- Control Plane = interaction with nearby devices to setup routing/forwarding tables, setup/release connections
- Management Plane = interaction with device's owner for initial config. (or permanent) and monitoring

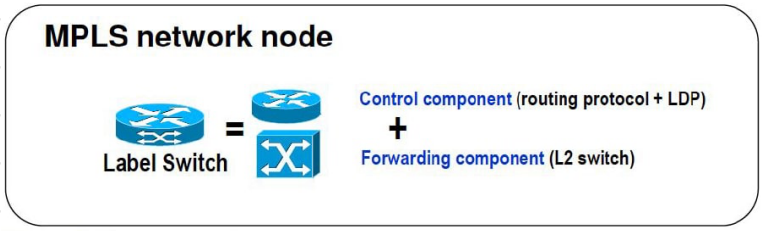
ASYNCHRONOUS TRANSFER MODE NETWORKS

In the past, backbones were built with IP routers connected with dedicated circuits (max 45/34 Mbps)
The increased traffic amount then required the implementation of ATM interfaces (155 Mbps)
Having now multiple kinds of devices, introduces more complexity in the configuration.
The structure anyway is seen by all of them as a full mesh topology
The main idea is: IP routers communicate over a set of ATM Permanent Virtual Circuits that work as logical circuits providing connection for edge routers. PVC are point2point links for routers.
Main problems: double management (ATM infrastructure and overlay IP network), N^2 PVC required, stress for routing, ATM interfaces not available for high speed connections (2.5 Gbps).



MPLS

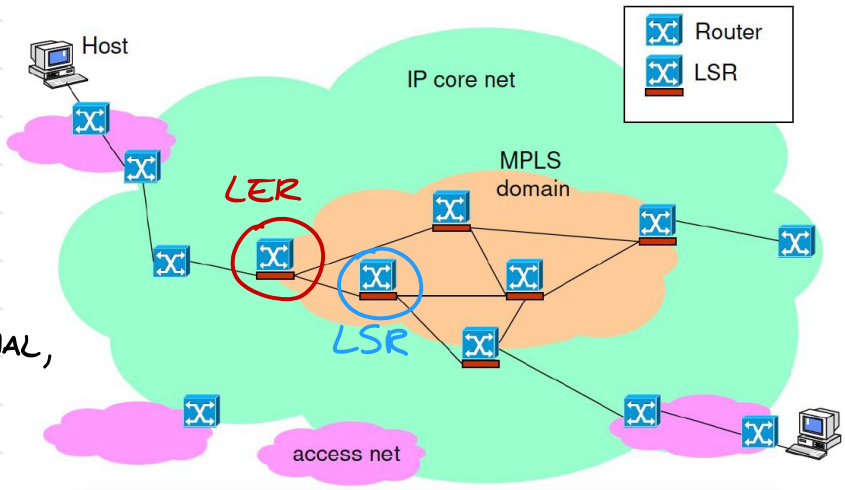
Add a fixed length identifier (Label) that can be used by internetworking device.
MPLS is independent both from underlying technology and from upper networking protocols.



The advantage respect to ATM is that we no longer have 2 network overlapped (ATM + IP)
The MPLS label is carried inside MPLS header called shim header, between layer 2 and 3 header. Next step is to remove lower level limitations (SDH) and pass to ethernet.
MPLS will be removed too in the future →

IP/SRv6
Eth
Optical

- **Label Edge Router (LER):** border node for a MPLS «domain»
  - it forwards packets to and from the MPLS domain, inserting and removing the Label to packets entering and going out from the domain
- **Label Switching Router (LSR):** a node that perform the Label switching within the MPLS domain
- **Label Distribution Protocol (LDP):** together with traditional IP routing protocols, a LDP is used to distribute the Labels among MPLS devices
- **Forwarding Equivalence Class (FEC):** a set of IP packets that are forwarded in the same way and receive the same treatment
- **Label Switched Path (LSP):** the path across one or more LSRs (followed by packet belonging to a FEC)
  - it corresponds to the "Virtual Circuit" concept

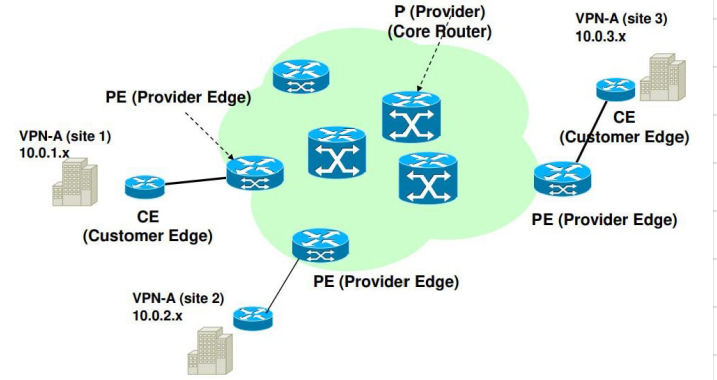


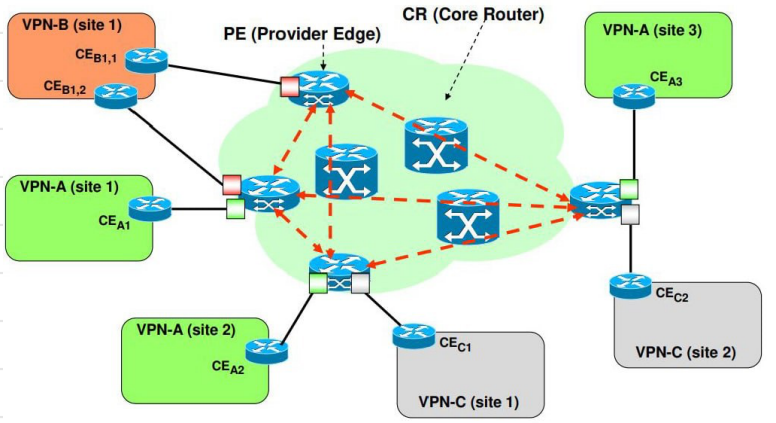
MPLS Label Switched Paths are UNIDIRECTIONAL, can be configured statically or dynamically and compared to ATM provide easier management and no cell-tax (overhead on each cell in ATM). In particular MPLS allows implementation of following features.

VPN

Main features of VPN are security/privacy and support of private IP addressing, two things that MPLS can provide since forwarding is based on the label and not on IP destination.

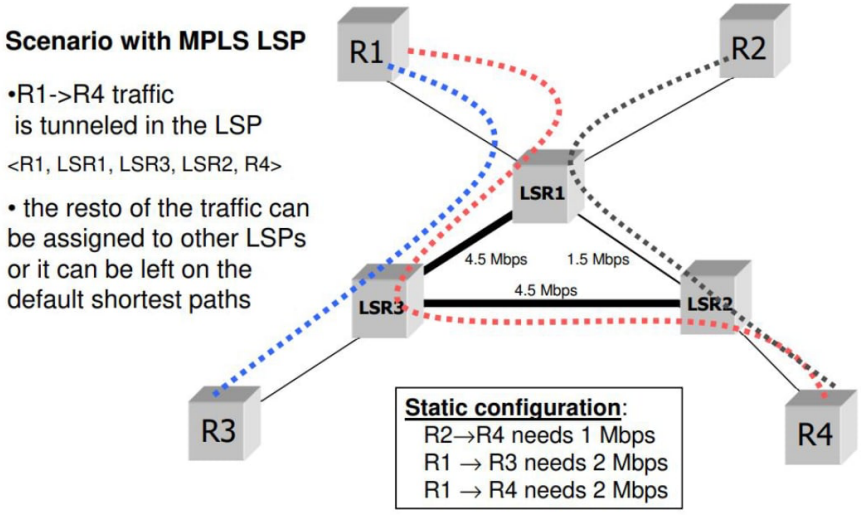
BGP/MPLS model





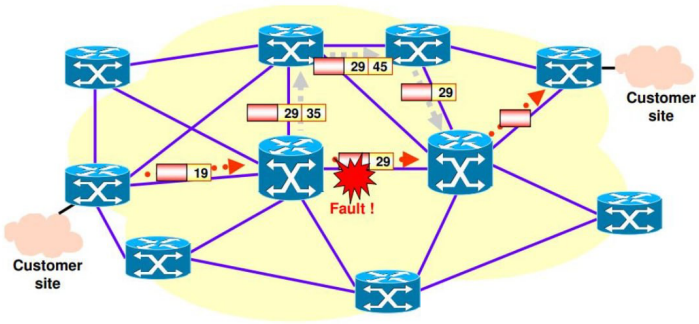
TRAFFIC ENGINEERING WITH MPLS LSP

In presence of OSPF/RIP or other shortest path routing scenarios there can be problems of congestions. This because routing doesn't take into account link bandwidth.
Even OSPF Equal Cost Multipaths is not suitable.

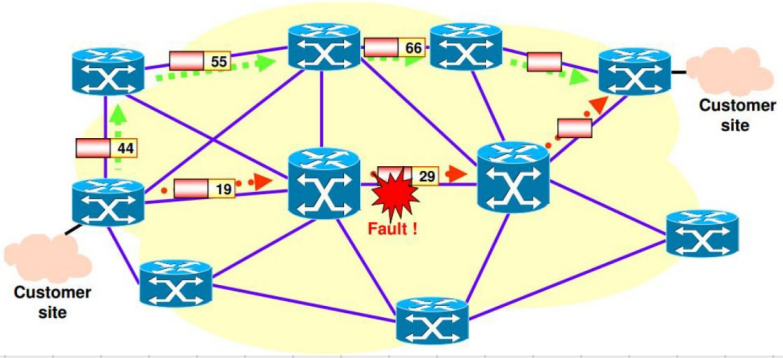


The path is choosen by an external entity and then the routers are configured (centralized) or on demand by routers (decentralized)
How to react to link failure?

- Rerouting (fast rerouting): a "protection tunnel" is used as alternative route from the point of failure
- Backup LSP: a secondary totally new path is used in case of failure.



Protection Tunnel



Backup LSP

SEGMENT ROUTING (SR)

Segment Routing leverages the source routing paradigm. The concept is to move state information from routers to packets (more overhead but also more speed). These information are called segments and are assigned to packets by the node

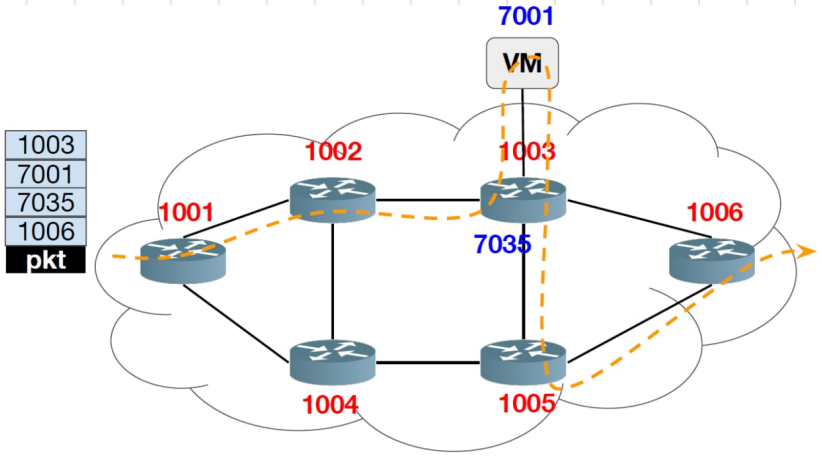
Control plane can be:

Distributed: segments allocated using OSPF and node individually computes the source-routed policy

Centralized: segments allocated and instantiated by a SR controller that computes also the source-routed policy. Most used.

Hybrid: a mix of the two above e.g. if destination outside IGP domain, SR controller may compute source-routed policy on behalf of IGP node.

SR brings flexibility at level 3 (from level 2)



From 1001 to 1003 routing just uses shortest path, then the packet start following SR (not necessarily the shortest path)

Constraints Based Routing: find route satisfying the channels bandwidth constraints.

SR infrastructure can be instantiated on various dataplanes (process the data requests):

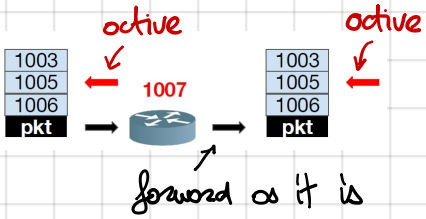
- SR-MPLS: no change on forwarding plane
- SR-IPv6: new type of header called SR Header (SRH)

SR FORWARDING

PUSH: insert segment header (routing instructions) over packet

NEXT: check next instruction and follow it

CONTINUE: current instruction not possible → keep it and go on ⇒



Segment can be Global or Local. In first case, the instruction associated to the segment is defined at SR domain level (e.g. shortest path), while in second case it is defined at node level.

POLICY

An SR Policy is a tuple <headend, color, endpoint>

- Headend: node where policy is instantiated
- color: 32-bit numerical value that associates policy with an intent (eg low-latency), unique per node
- endpoint: destination of policy

Note: headend and endpoint are IPv4/6 addresses

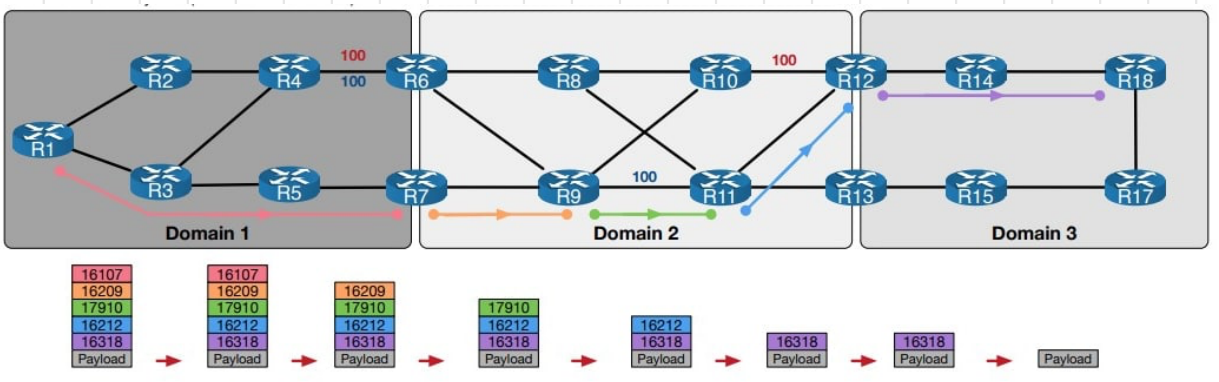
Each policy is associated with one or more candidate path (possible path that satisfy the policy)

Each path is associated with some Segment-List (SID-List) that represent the route from headend

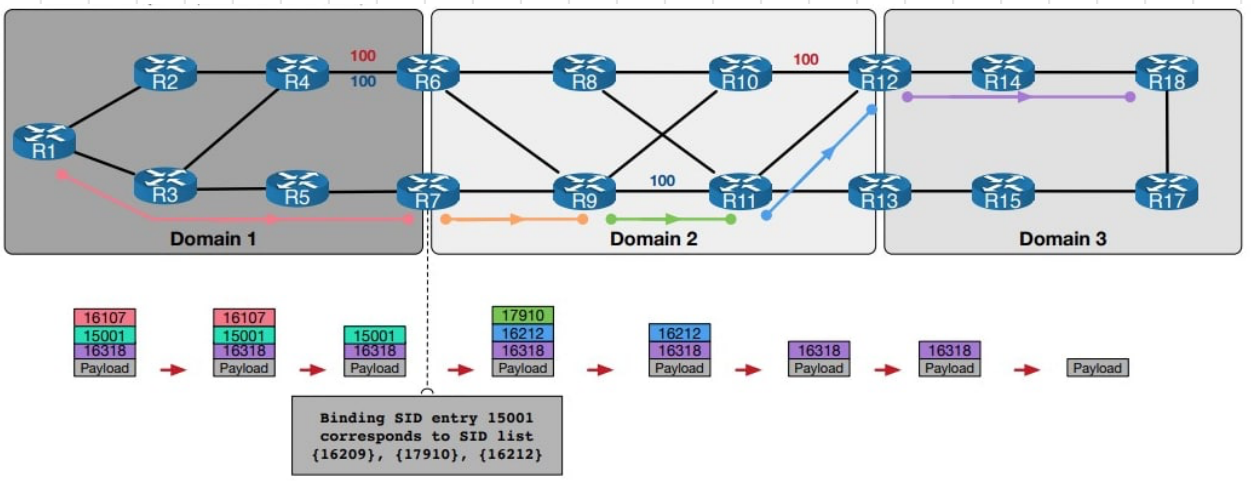
to endpoint.

BINDING SID

A BSID is a local label that is associated to a SR Policy, and so to its path.
The advantage of using BSID is that a router no longer needs to know the entire SID-List but can use a single entry (the BSID) to summarize them in the label stack.
This also eliminates the need for routers to update SID-List in case of failures.
Old behaviour:



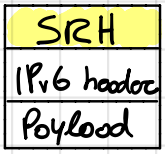
With BSID:



When arriving on node that "imposes" the policy, BSID entry is switched with SID-List corresponding to it.

SRH

SRH is added to the packet by its source:
- at node originating the packet (host, server)
- at ingress node of a SR domain.



← new "header layer"

} IPv6 packet

Only the routers whose address is in the DA field of the packet header MUST inspect the SRH
DA = destination address.

[SRH field is read backforward (A2::0, A3::0 ...)]

Each SRv6-capable node N maintains a MyLocalSID Table containing all the local SRv6 segments explicitly instantiated at node N, which is their parent node. Each SID has a specific instruction bounded to it.

→ Segment ID

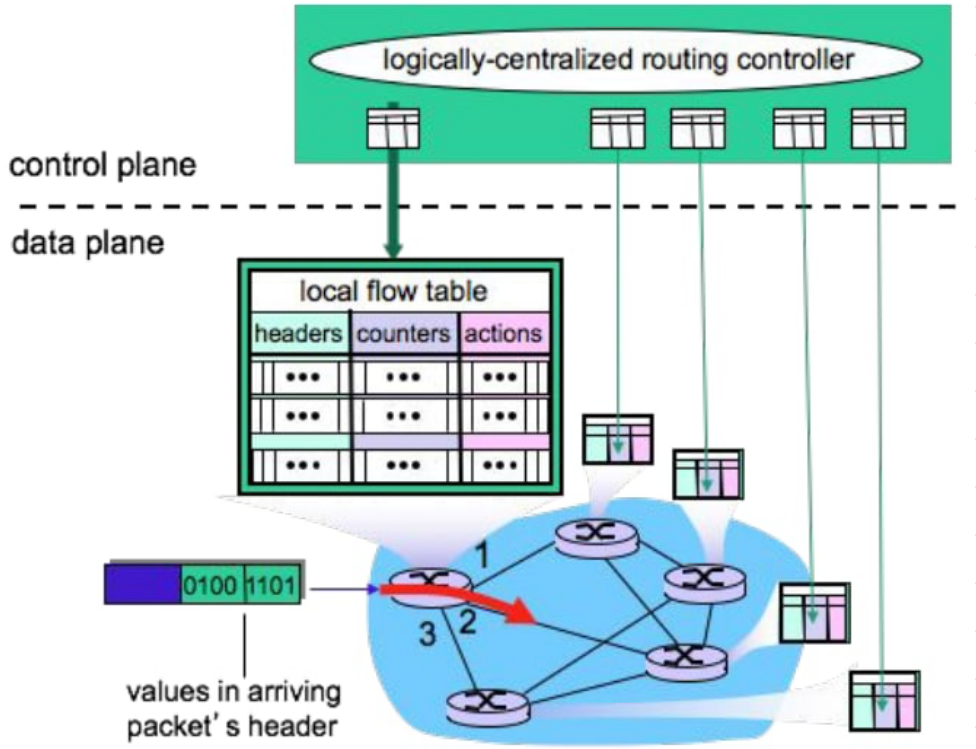
END: update DA with next segment and forward packet accordingly
END.X: forward to layer-3 adjacency bound to the SID

SOFTWARE DEFINED NETWORKING

Two key functions of network layer:
- forwarding: move packets from router's input to appropriate output (data plane)
- routing: determine the route from source to destination (control plane)

Control plane can be:
- Per-router: individual routing algorithms in each and every router interact in the control plane
- Logically centralized: a distinct (typically remote) controller interacts with local control agents (CAs)

The second type is used in SDN: each router contains a flow table that is computed and distributed by a logically centralized routing controller (similar to IP tables)



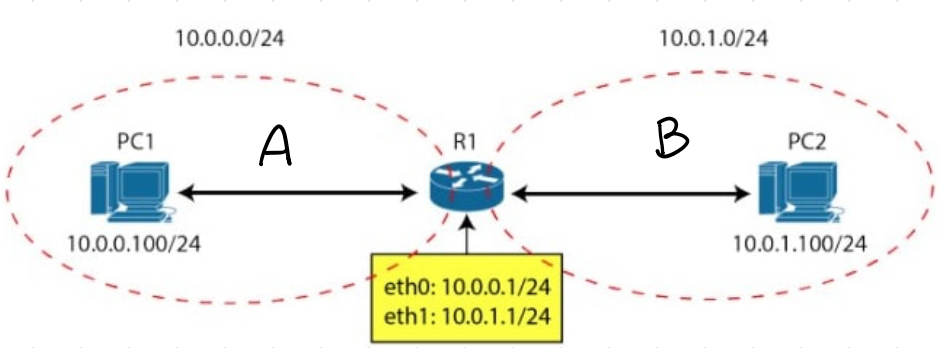
Why a logically centralized control plane?
- easier network management
- centralized programming is easier than distributed programming (OpenFlow API)
- open implementation

SDN CONTROLLER

It's the "network OS" that maintains network state information, interact with control applications and switches.
When a router experiences a failure it notify the controller, using OpenFlow "port status" message, that will update "link status" info. The application in charge of routing accesses graph info, computes new route and interact with controller which will compute new flow tables. Then flow tables are installed in switches that need update.

KATHARA' LAB

lab.conf



pc1[0] = A → eth0 of pc1 to collision domain A

r1[0] = A → eth0 of r1 to collision domain A

r1[1] = B → eth1 of r1 to collision domain B

pc2[0] = B → eth0 of pc2 to collision domain B

⇒ "kathara lstart" to start the lab (turn on docker first)
"kathara lclean" to close everything

ip link : show device interfaces (eth0/1 on r1)

ROUTING (iproute2)

Command suite used for network configuration.

ip address show : show assigned addresses
add A.B.C.D/M dev ethX : assign address to interfaces (dev)

To let pc1 and pc2 communicate we have to:
- define default route (gateway)
- specify how to reach pc2 subnet from pc1 and viceversa

ip route show
add A.B.C.D/M via NEXT_HOP_IP

NB "default" as A.B.C.D/M specify the default gateway (0.0.0.0/0)

NOTE: Create a "node".startup file (eg pc1.startup) to define commands for that node avoiding to enter them manually on CLI (eg routing)

NETCAT

Allow to send TCP/UDP packets. Define protocol, port and if we are sending or receiving.

nc -l -p 8080 : listening on port 8080
nc IP_add Port : open connection
add -u for UDP (TCP default)

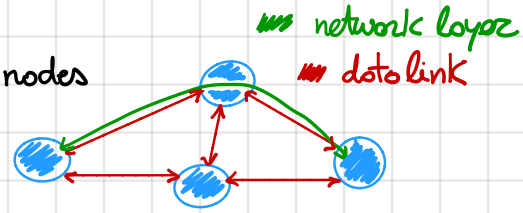
TCPDUMP

Listen to all packets on an interface (or log them)

tcpdump -i int_name -w filename.pcap
→ can be opened with Wireshark

ROUTING, IP SUBNETTING & ARP

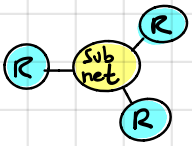
Data Link Layer (L2) addres specific bytes to connect adjacent nodes
Network Layer (L3): addressing between FAR nodes.
Transport Layer: introduce port concept for multiplexing.



All of this to explain that data are encapsulated multiple time each time it cross a layer (eg MAC(IP(TCP(data))))

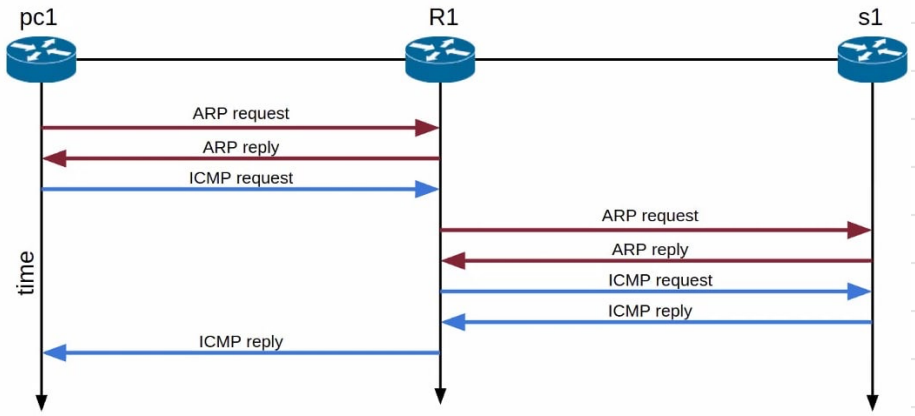
MAC address: unique identifier of device in L2 for every network interface

Address Resolution Protocol: used to discover link layer MAC from a given IP
How? Ask MAC of that IP flooding a broadcast request (ARP request). The one associated with that IP will respond with its MAC when receiving the request (with ARP reply)
ARP only works on directly connected nodes (i.e only 1 hop) → works on subnet



MAC collected by ARP are stored in ARP table (IP-MAC pairs) always checked when we want to contact a specific IP (ARP request if not in table).

By the way ARP done is not secure and network admin should implement defenses against ARP attacks.



When routing a packet, subnet is checked on routing table:
- 1 match → send there
- multiple match → choose longest prefix i.e. longest mask
- no match: default entry?
  - yes → send there
  - no → "host unreachable" error

DHCP & OSPF

Note: we can organize files of nodes in foldees. Folder "pc1" containing "foo.txt" will make foo.txt accesible from /etc/foo.txt. In this way we can create the file /etc/networks/interfaces to configure interfaces at startup.

e.g. auto eth0
iface eth0 inet static
address x.x.x.x

! "/etc/init.d/networking restart" inside startup script of every node to automatically mount

DHCP: automatically assign IPs and default gateway and DNS inside a LAN
It follow the client-server architecture where client ask for IP and DHCP server answer.

CLIENT: "iface eth* inet dhcp" ↝ instead of static

SERVER: create file dhcpd.conf in /etc/dhcp/

```
default-lease-time 3600;

subnet 10.0.0.0 netmask 255.255.255.0 {
        range 10.0.0.100 10.0.0.254;
        option routers 10.0.0.1;
}

subnet 10.0.1.0 netmask 255.255.255.0 {

        range 10.0.1.100 10.0.1.254;
        option routers 10.0.1.1;
}
```

? Just put it

from — to

default gateway

Subnet 1

Subnet 2

Then: "/etc/init.d/isc-dhcp-server start" in startup script

OSPF: dynamic routing using "flooding" messages, notifying neighbors of node to wich nodes it is connected/know info about. (in routers)

① create file "daemons" in /etc/quagga/

```
zebra=yes
bgpd=no
ospfd=yes
ospf6d=no
ripd=no
ripngd=no
isisd=no
ldpd=no
~
```

② create "zebra.conf" in /etc/quagga/

```
! -*- zebra -*-
!
! zebra sample configuration file
!
! $Id: zebra.conf.sample,v 1.1.1.1 2002/12/13 20:15:30 paul Exp $
!
hostname r1
password zebra
enable password zebra

interface eth0
ip address 1.0.10.2/31
link-detect

interface eth1
ip address 1.0.10.9/31
link-detect

interface eth2
ip address 1.0.10.13/31
link-detect

interface eth3
ip address 1.0.10.19/31
link-detect
```

One for each network interface

③ create file "ospfd.conf" in /etc/quagga/

```
hostname r1
password zebra

interface eth0
ospf hello-interval 2
ospf cost X
interface eth1
ospf hello-interval 2

interface eth2
ospf hello-interval 2

interface eth3
ospf hello-interval 2

router ospf
network 1.0.10.18/31 area 0.0.0.0
network 1.0.10.2/31 area 0.0.0.0
network 1.0.10.8/31 area 0.0.0.0
network 1.0.10.12/31 area 0.0.0.0
```

Is ponible to specify cost for each link

"traceroute dest_add" = like ping, but also show all the steps of the packets.

Areas are used to subdivide the network

0.0.0.0 ≠ 1.1.1.1 ≠ 2.2.2.2 etc

Backbone area

Routers connected to different areas are called Area Border Routers, the ones with at least one interface on the backbone area called Backbone Routers.

SSH

Physical access is not always possible. Secure Shell allows remote access as client-server configuration on a secure channel. SSH provides authentication, encryption and integrity (in both way)

Encrypted connection setup

1. The client opens a TCP on (by default) port 22. They both disclose the SSH protocol versions they support
2. They both provide a list of accepted cipher methods
3. The client selects a cipher method and negotiate a *session key*
   a. From now on the rest of the communication are encrypted symmetrically using the *session key*. It provides encryption.

Server authentication

4. The server identifies itself to the client: it sends to the client its *host key*
5. The client sends a challenge to the server
6. The server replies with an authenticator
7. The client checks the authenticator. If all went well the server is authenticated to the client.

Client authentication

8. The client now proceeds to authenticate to the server

Two methods are available for CLIENT AUTH.

- Password → sends the password as it is to server (connection is encrypted but server may be malicious/infected)
- Public Key → simply private-public keys pair authentication

"ssh user@host -p port" → connect to host as user

```
/etc/init.d/networking restart
/etc/init.d/ssh restart
mkdir /home/ssh_user
useradd ssh_user -d /home/ssh_user
chown ssh_user:ssh_user /home/ssh_user
echo -e 'ilovessh\nilovessh\n' | passwd ssh_user
```

restart networking and ssh daemon

Create ssh_user folder and add an user in it, giving access

set password for the user

"echo 'IP alternative_name' >> /etc/hosts" → use a string instead of IP numbers

If we want to use public-key auth, we first create the key pairs on client: "ssh-keygen"
The command will generate 3 files containing private and public keys and known hosts (trusted fingerprints) inside .ssh folder.
Now we have to copy public key to server: "scp local_path user@host:/remote_path"
↖ invert for download!

Now on server we have to authorize the new public key

The "database" is the file `/home/ssh_user/.ssh/authorized_keys` which we have to create

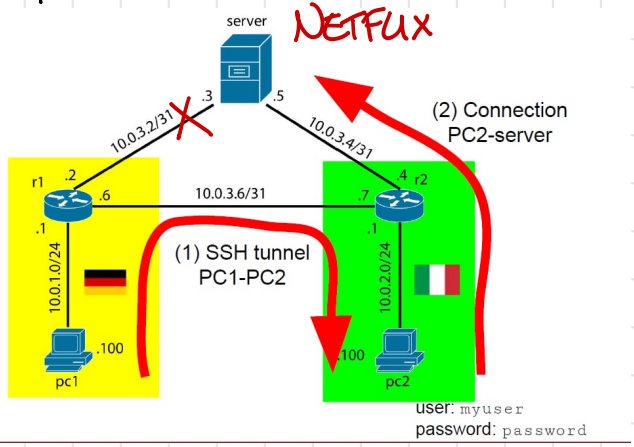
```
$ mkdir .ssh                        #create .ssh folder
$ touch .ssh/authorized_keys        #create authorized_keys file
# Set correct permissions
$ chown ssh_user:ssh_user /home/ssh_user/.ssh
$ chmod 700 .ssh
$ chown ssh_user:ssh_user /home/ssh_user/.ssh/authorized_keys
$ chmod 600 .ssh/authorized_keys
```

Then: "cat pub_key >> .ssh/authorized_keys"

/etc/ssh/sshd_config → PasswordAuthentication no : disable pwd auth. after enabling public key one.

PORT FORWARDING

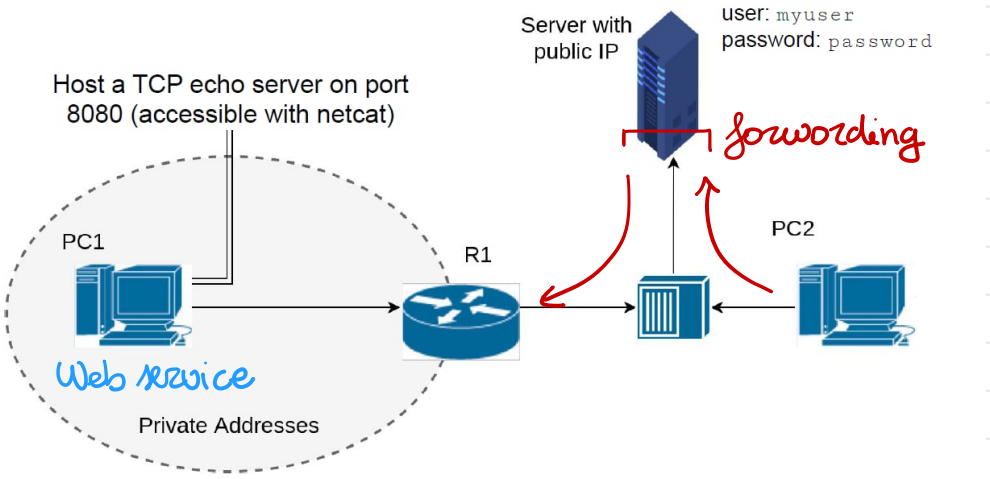
SSH allows port forwarding: redirect packets inside an existing SSH connection
A possible scenario:



We can't access netflix from PC1 so we pass through PC2. This is called local port forwarding
"ssh -NL 8080 (local port) :10.0.3.5:80 (dest), user@10.0.2.100 (tunnel IP)"
The above command redirects all traffic directed to port 8080 of server through an SSH tunnel over port 9000 from PC1 to PC2

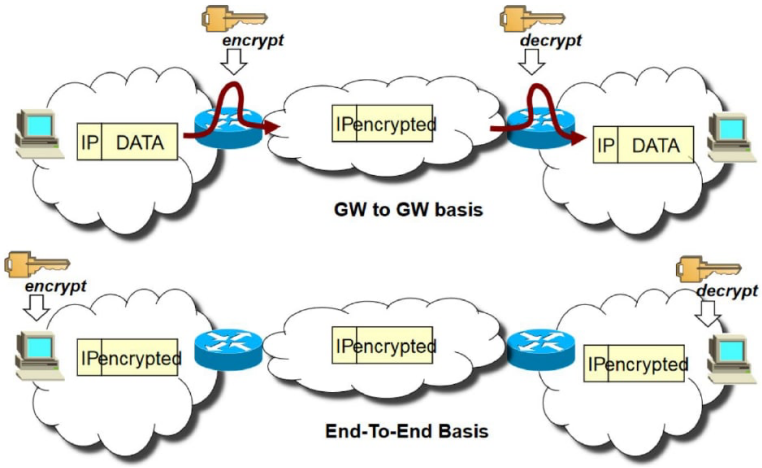
On contrary Remote port forwarding allows to redirect packets coming to a port on a remote host to one on your local port
For example if we have a local web service we can allow user outside the LAN

to access it passing through a remote server (that has a public IP)



VPN

Generally implemented to use public internet to connect private network. VPN provides encryption, authentication, packet tunnelling and firewalls.
Some issues remain: our connection pass through third parties network layer/device



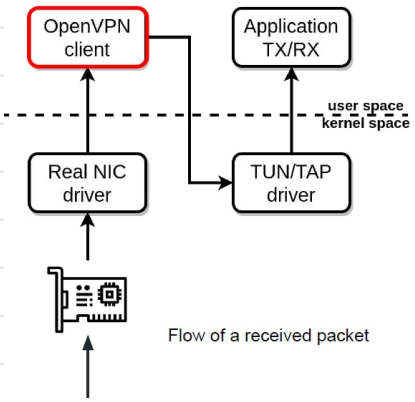
VPN can be implemented directly on pcs (end-to-end) or just on gateways (routers) depending on the use case. Also a mixture of the two is possible.
We will use OpenVPN, a Linux VPN implementation open source.

Public Key Certificate: data structure that binds public (and private) key to the identity of the owner, made by a certification authority using its private key. Using authority public key is possible to check validity of certificate. PKI infrastructure is the structure that sustain PKC.
Chain of Trust certificates the validity of CA. Biggest CA certificates for big ones that certificates for small ones and so on like a pyramid.
First step for an OpenVPN conf. is to establish a PKI:
- master CA certificate and key
- a certificate for server and each client → authentication: check if certificate signed by master CA

"cd /usr/share/easy-rsa"
"cp openssl-1.0.0.cnf openssl.cnf" → openssl config
"source ./vars" → load easy-rsa variables
"./clean-all" → clean any existing CA
"./build-ca" → build CA

CLIENT/SERVER

"./build-key-server SERVER_NAME" → generate certificate and private key for server
"./build-key CLIENT_NAME" → certificate and keys for client



OpenVPN creates two types of interfaces:
TUN: IP tunnelling (layer 3), easier and less overhead
TAP: Ethernet tunnelling (layer 2), the host behaves as it was directly connected via Eth to the LAN
We will cover only TUN.

FIREWALLS

Software/hardware that block/allow in/out connections following some custom rules (IPTABLES)
IPTables are the frontend of NETFILTER a framework that allow hook handling on connections.

- **Accept** packets of **TCP port 22 coming from interface eth1**
  - `iptables -A INPUT -p tcp --dport 22 -i eth1 -j ACCEPT` (`-p tcp --dport 22 -i eth1`: matching rule; `ACCEPT`: action)
- **Drop** packets **routed to subnet 192.168.1.0/24**
  - `iptables -A FORWARD -d 192.168.1.0/24 -j DROP`
- **Redirect** packets of **TCP directed to local port 80 to port 8080**
  - `iptables -t nat -A PREROUTING -p tcp --dport 80 -j DNAT --to-destination 127.0.0.1:8080`

Check CNS notes on Github for detailed schema on how to build rules

We will use "filter" table (INPUT, FORWARD, OUTPUT chains) and "nat" table (PREROUTING, OUTPUT, POSTROUTING chains)