UNIVERSITÉ DE LORRAINE | IUT Nancy-Brabois

Licence Professionnelle AII-SARI

Automatique et Informatique Industrielle
Spécialité Systèmes Automatisés et réseaux Industriels

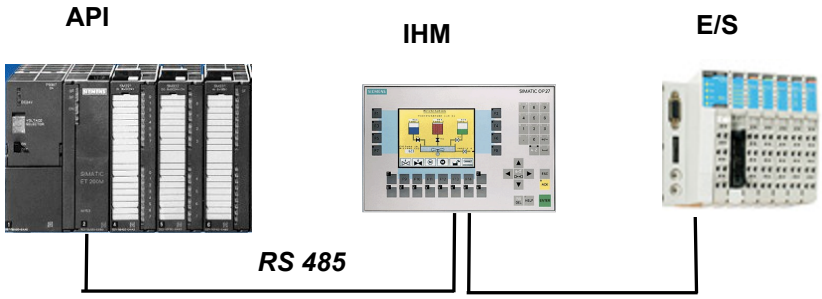
Ethernet et les ProtocolesTCP - IP
Ethernet Industriel
Modbus-TCP

Pr. Eddy BAJIC,
Nancy Université
IUT Nancy Brabois

UNIVERSITÉ DE LORRAINE | IUT Nancy-Brabois

Introduction

Depuis un Un réseau d'Automatisme “Isolé”

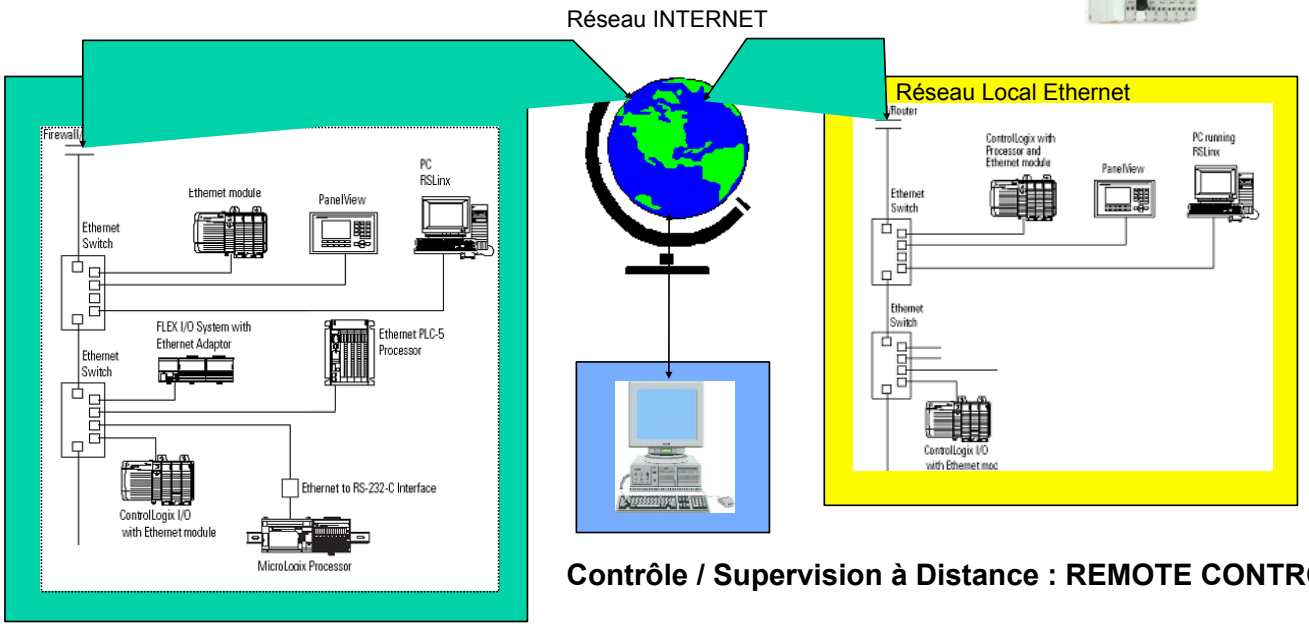


- Lire / Ecrire des informations sur BITS, MOTS vers des cibles industrielles PROCHES
- RESEAU LOCAL INDUSTRIEL , Câblage en guirlande
- Architecture traditionnelle (ancienne ?)

Vers des Automatismes sur INTERNET

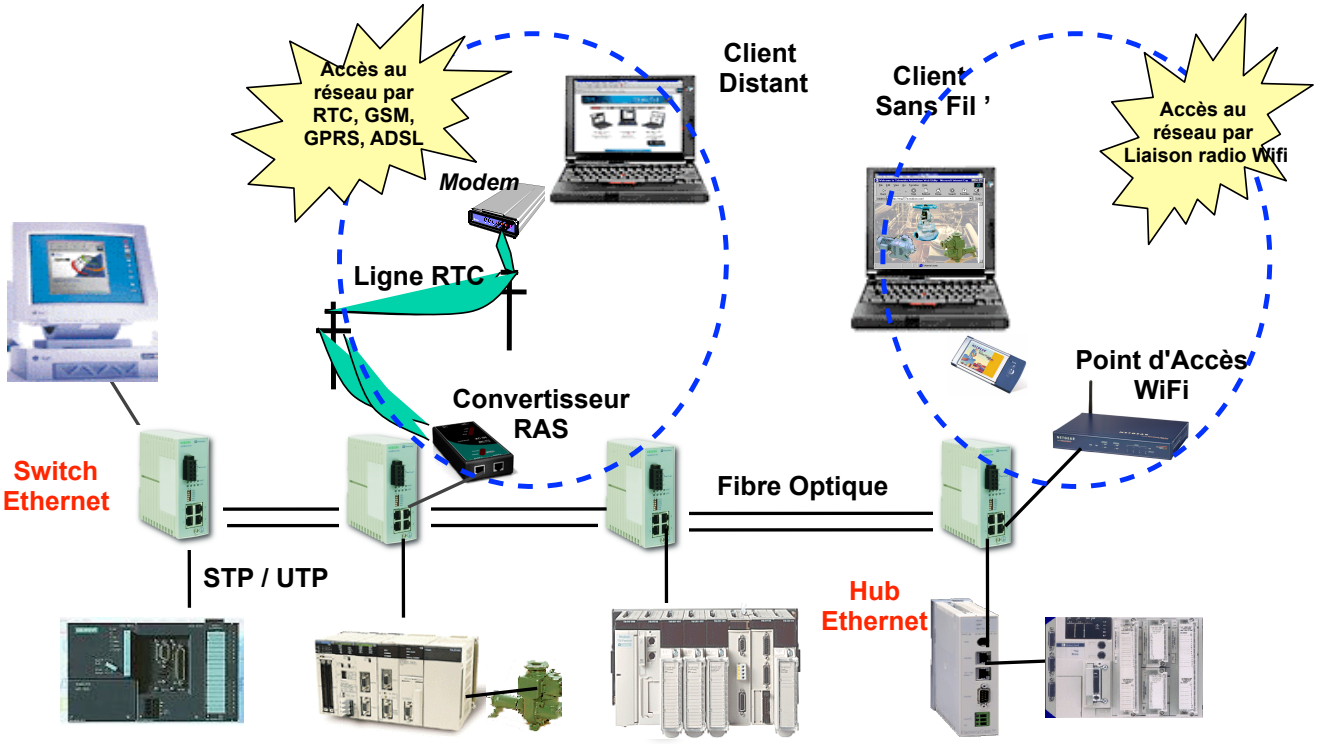
Disposer des équipements industriels (E/S, API, ….) sur Ethernet dans l'entreprise / atelier

Accéder aux équipements industriels à distance par Internet



Contrôle / Supervision à Distance : REMOTE CONTROL

Architecture Réseau Industriel Ethernet



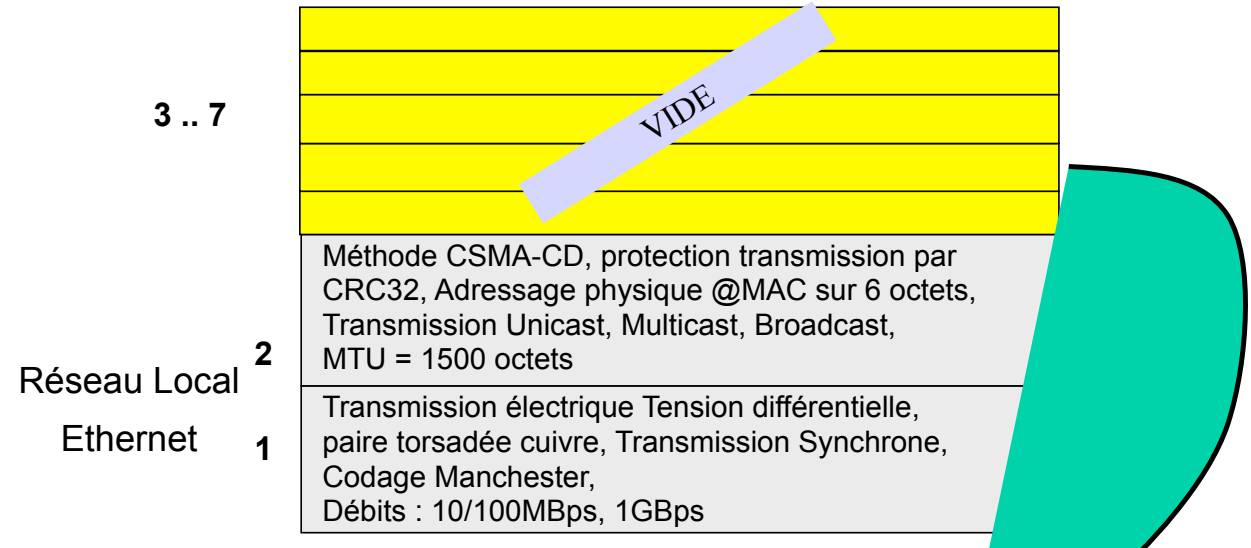
Ligne RTC : Réseau Téléphonique Commuté
Convertisseur RAS : Remote Access Serveur

Point d'Accès WiFi: Borme Ethernet Sans Fil 802.11d/g

QuickTime™ et un
décompresseur
sont requis pour visionner cette image.

Le Réseau Ethernet

Ethernet et le Modèle OSI

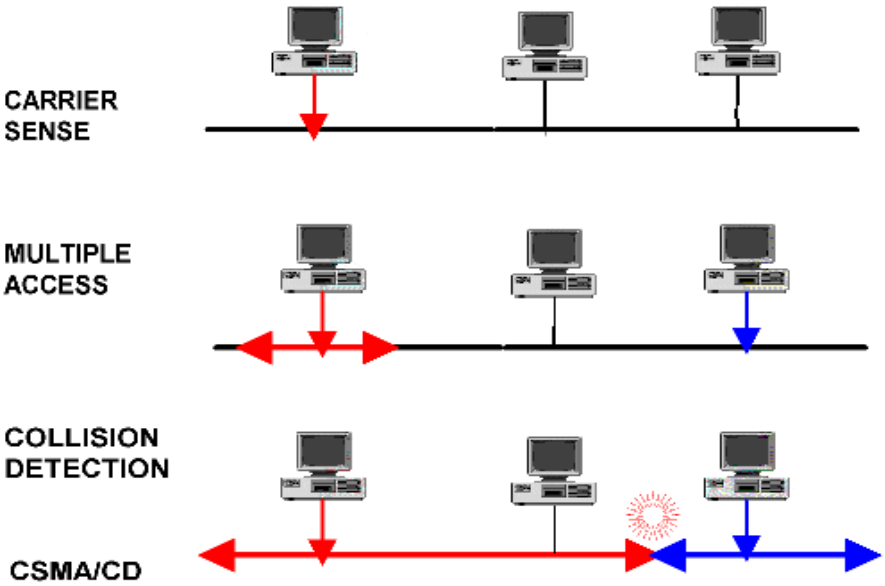


Format Trame Ethernet

	6 octets	6 octets	2 octets	1500 octets	4 octets
Préambule 64 Bits	@MAC Destination	@MAC Source	Type Protocole couche 3	Données Couche 3 1500 Octets Maxi	CRC 32

Ethernet : Méthode CSMA/CD

Standard **IEEE 802.3** : Carrier Sense and Multiple Access with Collision Detection



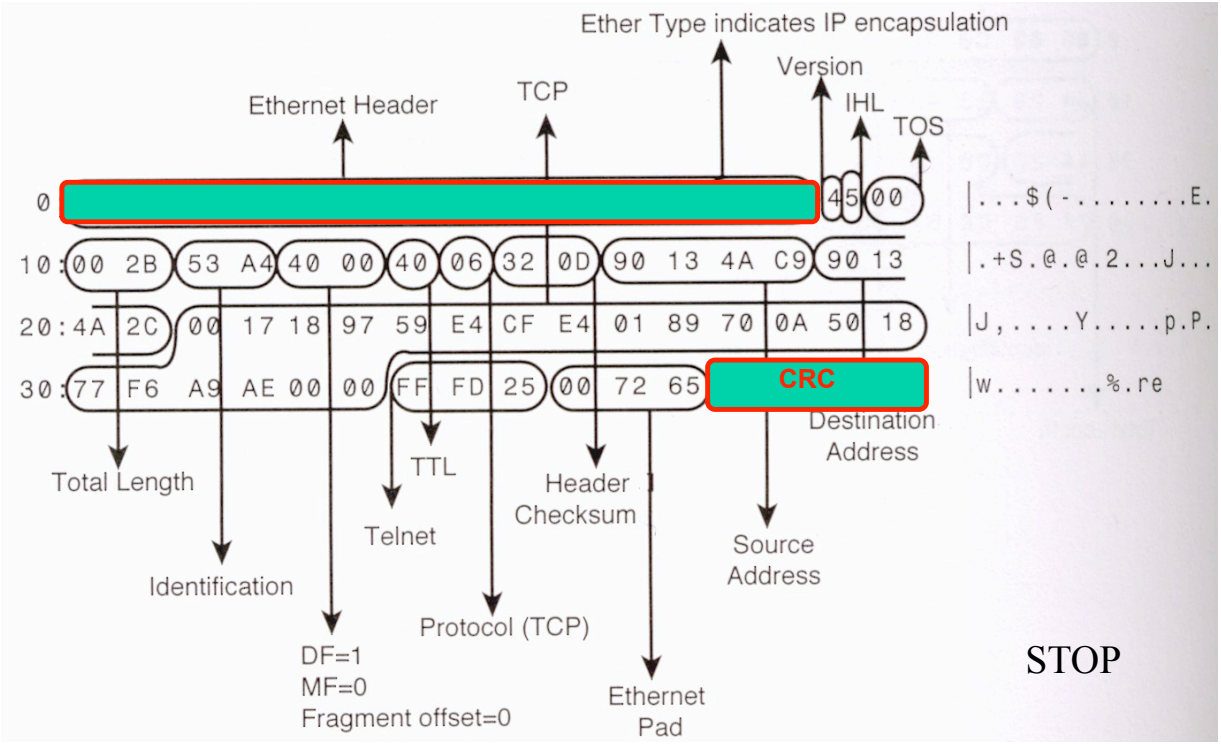
Avant d'émettre, toute station "écoute" la voie pour savoir si elle est libre :
1) Si LIBRE, elle émet sa trame
2) Si OCCUPéE, elle attend un temps tiré au hasard

Topologie apparente est un bus

Chaque station a les mêmes droits,

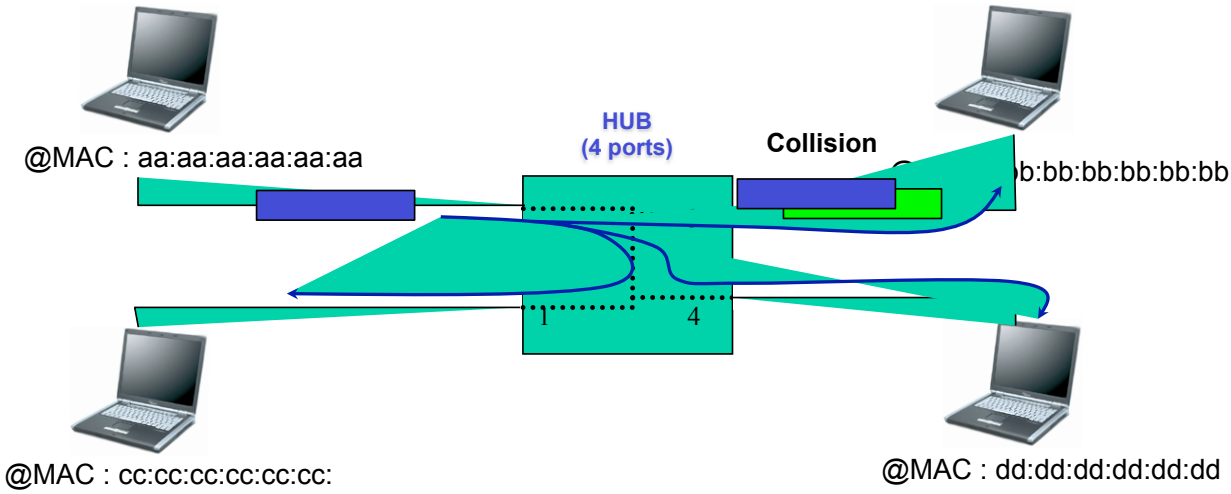
Quand une station émet elle vérifie si son message est "écrasé" par une autre station émettant en même temps : COLLISION

Détail d 'une trame Ethernet



Ethernet Partagé en Bus

Hub ou Concentrateur :
Equipement d'interconnexion de stations, agit en diffusion de trames sur tous ses ports (répéteur non intelligent : inondation des ports)



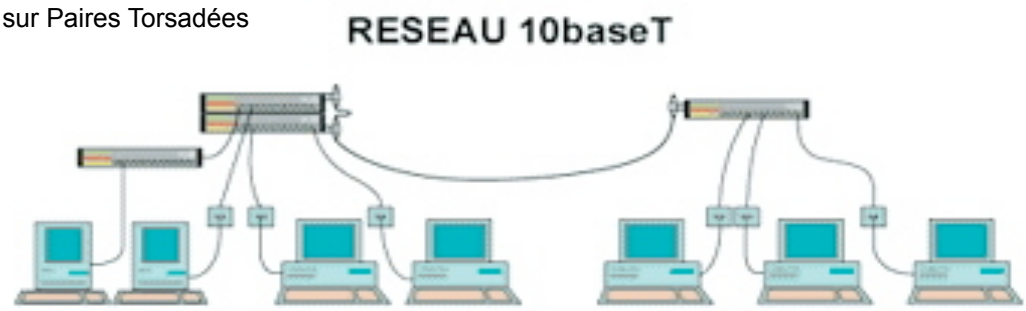
Hub Ethernet 8 voies

Topologie apparente : ETOILE

Topologie réelle : BUS

ETHERNET 10/100 Base T

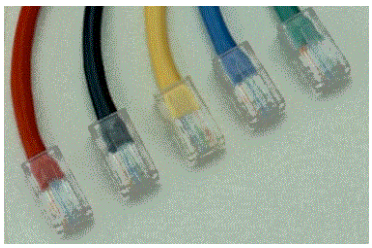
ETHERNET 10/100 Base T sur Paires Torsadées



Caractéristiques :

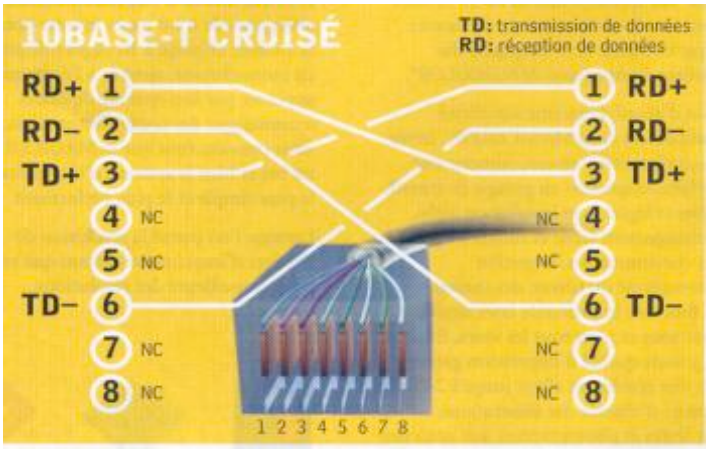
- Segment 100 m maxi STP, 40 m UTP
- Une seule Station par segment
- Segments dérivés depuis un Hub (4, 8, 16, 24 Voies)
- Connexion par connecteurs RJ45
- Câble UTP AWG 22, 24 ou 26

Connecteur RJ 45



Câblage ETHERNET 10/100 Base T

Câble Croisé entre 2 stations



Câble droit entre Station et HUB (Half Duplex ou Full Duplex)

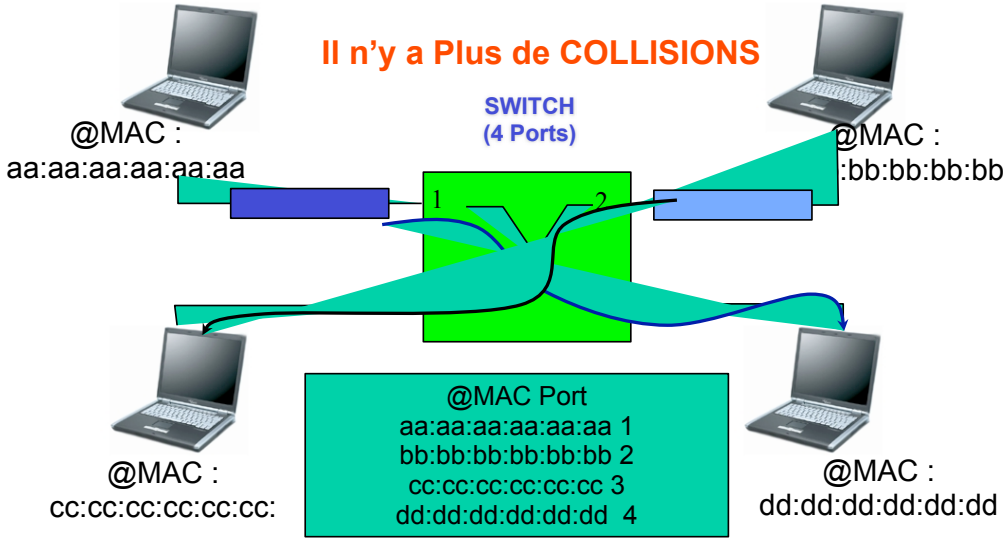
Comment Réaliser un câble Ethernet

http://www.youtube.com/watch?v=0S6cjJS5y1I

Câblage ETHERNET Facile

Ethernet commuté

Switch ou Commutateur :
Equipement d'interconnexion de stations, agit en **routage de trames** sur chaque port (équipé d'un processeur RISC qui gère le routage des trames entre les ports)

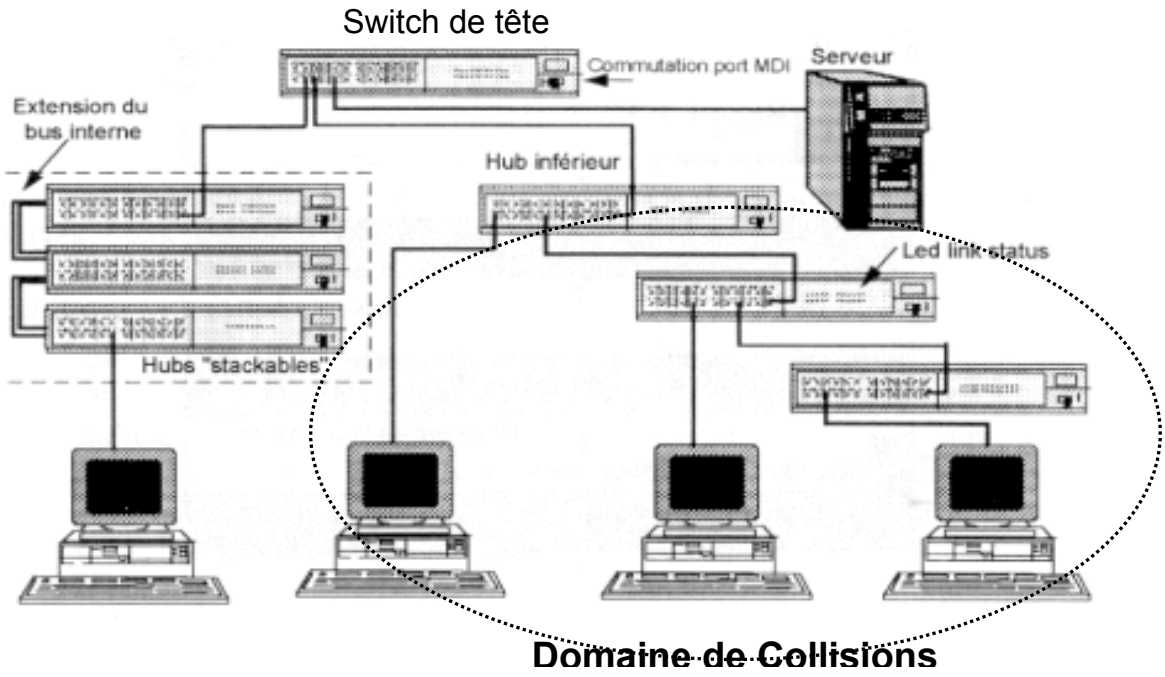


2 méthodes de gestion de flux de trames:

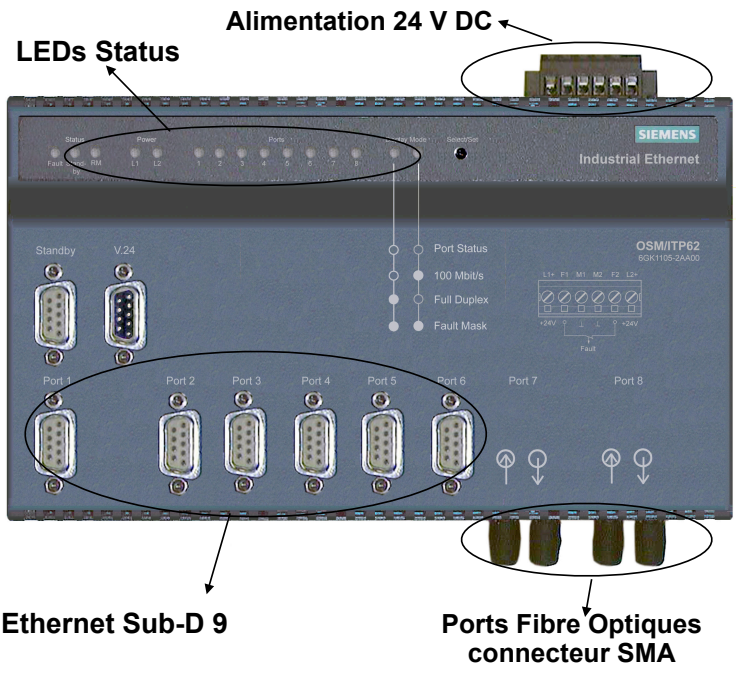
- **Cut Through** : flux tiré par l'adresse dest de la trame entrante, pas de contrôle d'erreur
- Store and Forward : trame bufferisée, analysée puis routée vers le port dest

Auto-Apprentissage des adresses MAC connectées aux différents ports

Architecture Capillaire Ethernet



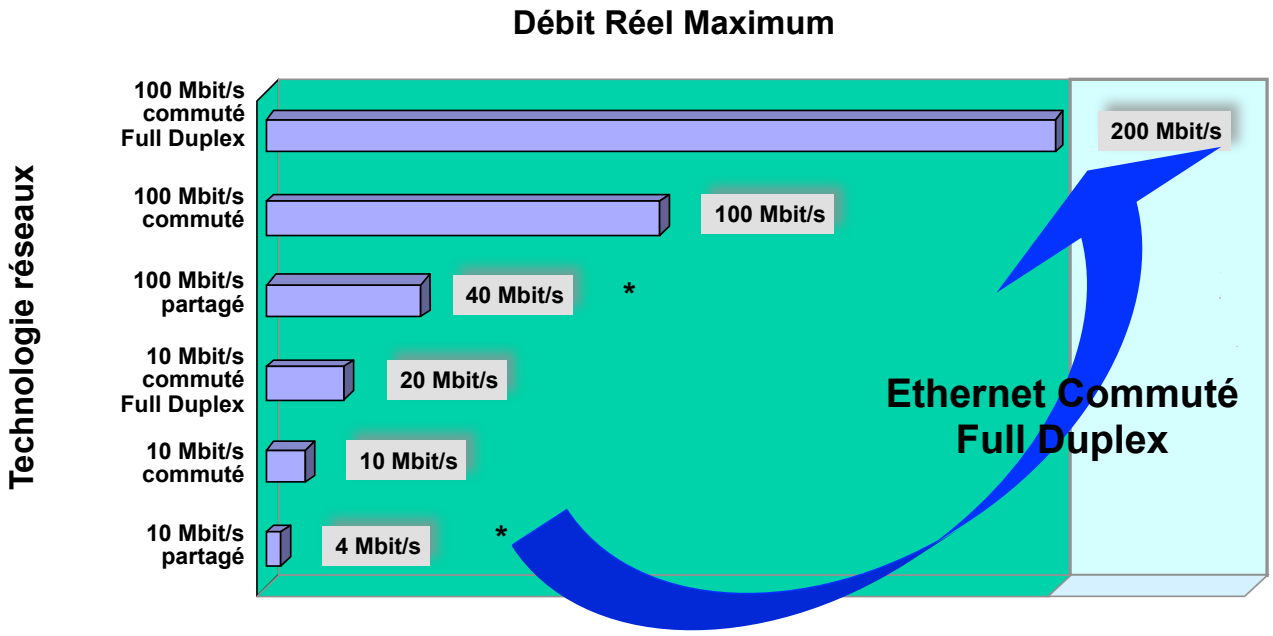
Switch Industrial Ethernet 100 Mbit/s



Caractéristiques

- Industrial Twisted Pair
- OSM : Optical Switch Module
  6 ports ITP/TP
  2 ports FO
- ESM : Electrical Switching Module
  8 ports ITP/TP
- Gestion de Redondance
- Manageable SNMP
  Simple Network Management Protocole

Performances Ethernet Partagé / Commuté



* Débit effectif Optimum à cause des collisions

Internet

Le « Réseau Internet »

☑ Ensemble de réseaux interconnectés au niveau mondial, grâce à la famille de protocoles TCP/IP.

☑ Ces réseaux sont administrés, exploités et financés dans tous pays par des d'organisations privées et publiques différentes.

☑ La coordination de l'ensemble est assurée par l' **IAB** (Internet Architecture Board), groupe de chercheurs et industriels qui coordonne et spécifie des règles de fonctionnement d'Internet.

☑Documents de standardisation Internet sont appelés Request For Comments (RFC) :

RFC xxxx (**Request For Comment** N° xxxx) (*http://ds.internic.net)*

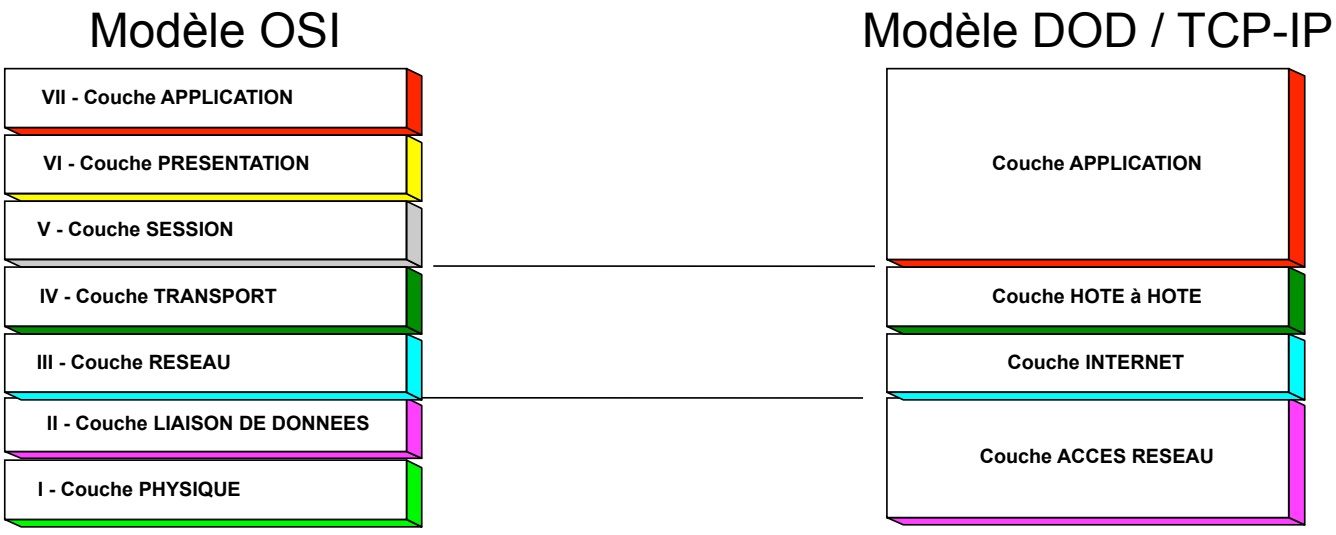
Le protocole TCP a été défini en 1981 dans la RFC 792.

Site Web de référence sur TCP - IP http://www.frameip.com/accueil/

Le Modèle DoD ou TCP-IP

Le modèle DOD (**Department Of Defense of USA**), simplification du modèle OSI.

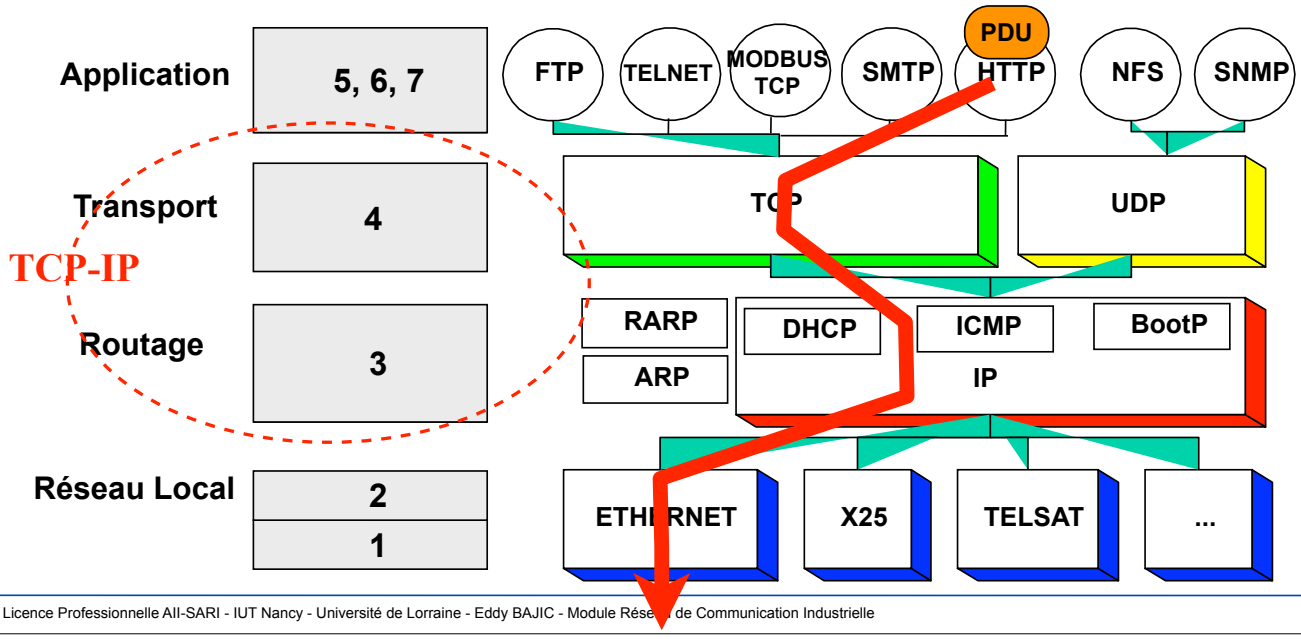
⇨ adapté aux réseaux basés sur les protocoles TCP/ IP.



Famille de Protocoles Internet

Le terme **TCP-IP** englobe un ensemble de protocoles réseaux assurant les mécanismes d'échanges d'information sur Internet (Le terme n'est pas réservé aux seuls protocoles TCP et IP)

⇨ **Modèle réseau TCP-IP / Pile de protocoles TCP-IP (Stack TCP-IP)**



Les Protocoles d'Application courants sur Internet

HTTP (Hyper Text Transfer Protocol) : Envoi de pages hypertexte à un ordinateur équipé d'un navigateur (browser) pouvant lire des documents graphiques, audio et vidéo.
Les pages web sont codées dans un langage HTML (Hyper text Markup Language).

SMTP (Simple Mail Transfer Protocol) : Envoi de messages (E-Mail) depuis un client SMTP vers un serveur SMTP. Le compte destinataire du message sur le serveur est identifié par son adresse électronique.

FTP (File Transfer Protocol) : Transfert de fichiers entre deux machines. L'utilisateur (client FTP) se connecte à un serveur FTP et peut visualiser et transférer les fichiers et répertoires de l'hôte.

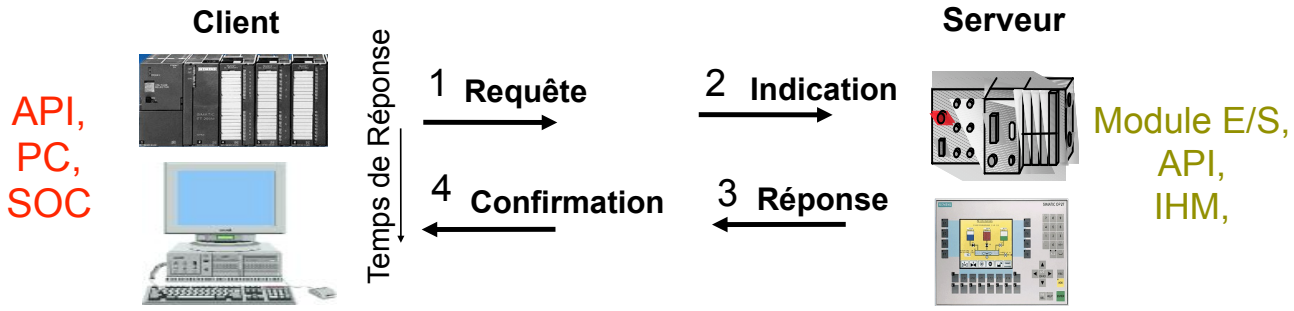
SNMP (Simple Network Management Protocol) Gestion à distance des équipements d'interconnexion réseau tels que routeur, hub , switch
Un programme (agent SNMP) doit être installé sur ces périphériques.

NFS (Network File System) : Equivalent à FTP pour des systèmes Unix.

TELNET : Connexion à distance à une station. Permet de travailler comme en connexion locale avec l'hôte (clavier/écran déportée par le réseau)

Modèle Client – Serveur TCP/IP

L'architecture Internet est basée sur un modèle de communication Client-Serveur



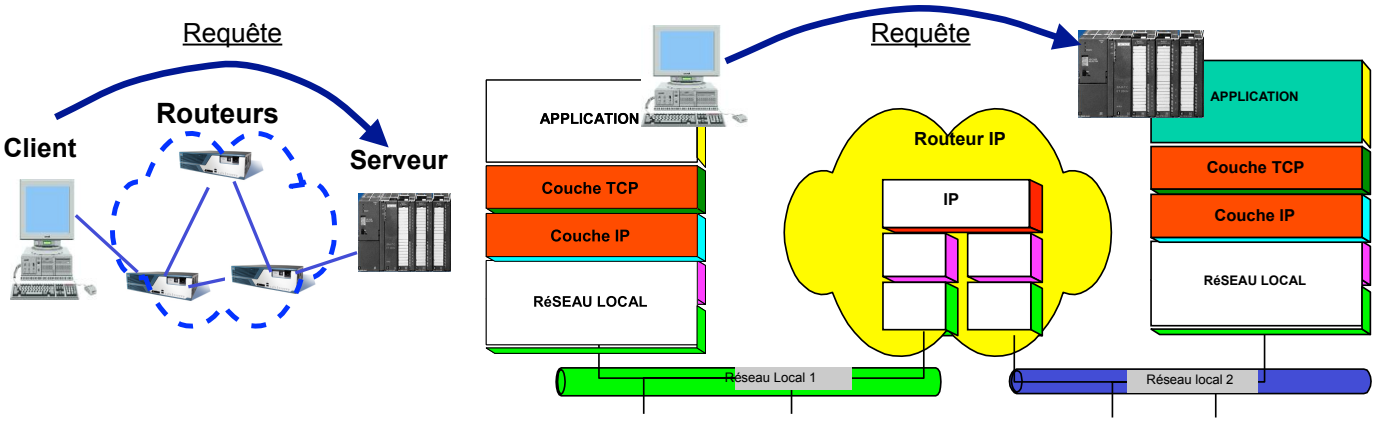
Client : Sollicite un service du Serveur par une Requête

Serveur :

- Est à l'écoute des clients qui le sollicitent (écoute sur N° de port TCP/UDP donné)
- Ne répond qu'aux requêtes des clients

Temps de Réponse : Non garanti, indéterminé, il dépend de l'architecture réseau traversée

Interconnexion par Protocoles TCP-IP



4 Mécanismes de base nécessaires aux réseaux TCP-IP

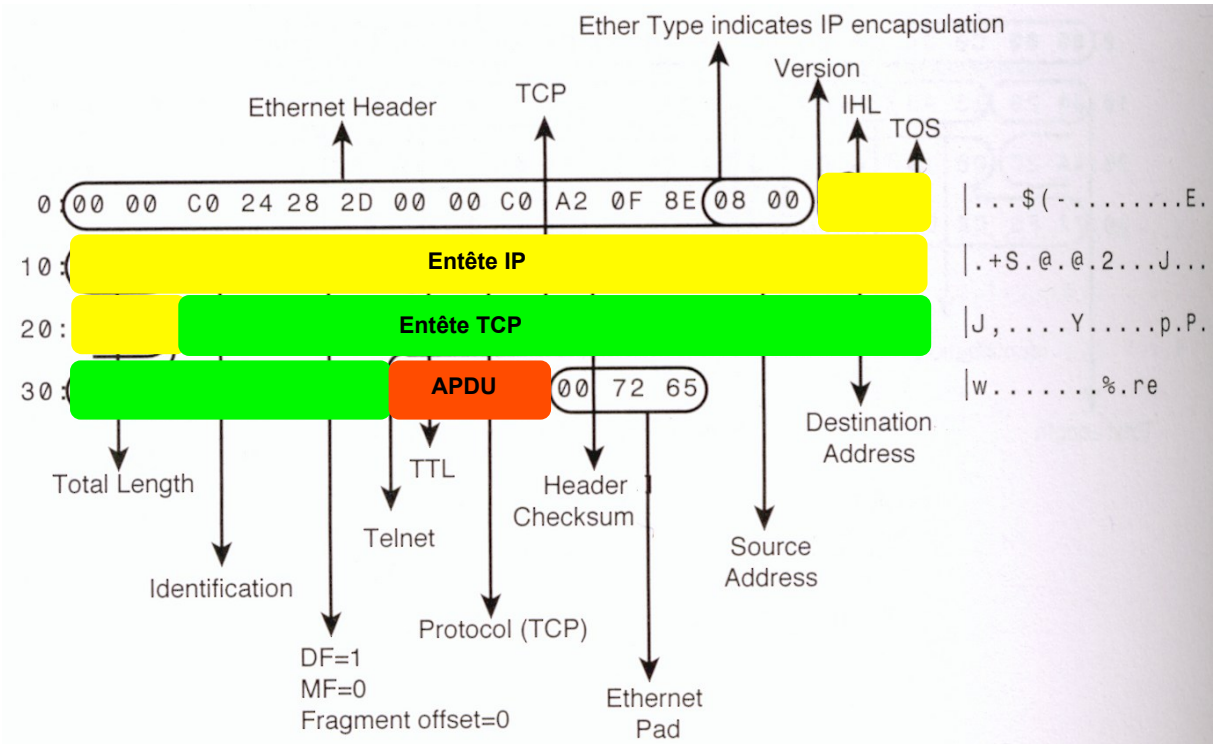
⇨ **ADRESSE LOGIQUE** d'une machine sur le réseau Global (adressage mondial)

⇨ **ROUTAGE** d'une trame à travers les réseaux successifs pour atteindre le destinataire

⇨ **CONTRôLE DE FLUX** des paquets pour garantir leur acheminement au destinataire (Acquittement / Réémission)

⇨ **FRAGMENTATION / ASSEMBLAGE** des paquets dans les réseaux traversés pour correspondre aux tailles maximales des trames autorisées

Encapsulation d'une Trame Ethernet



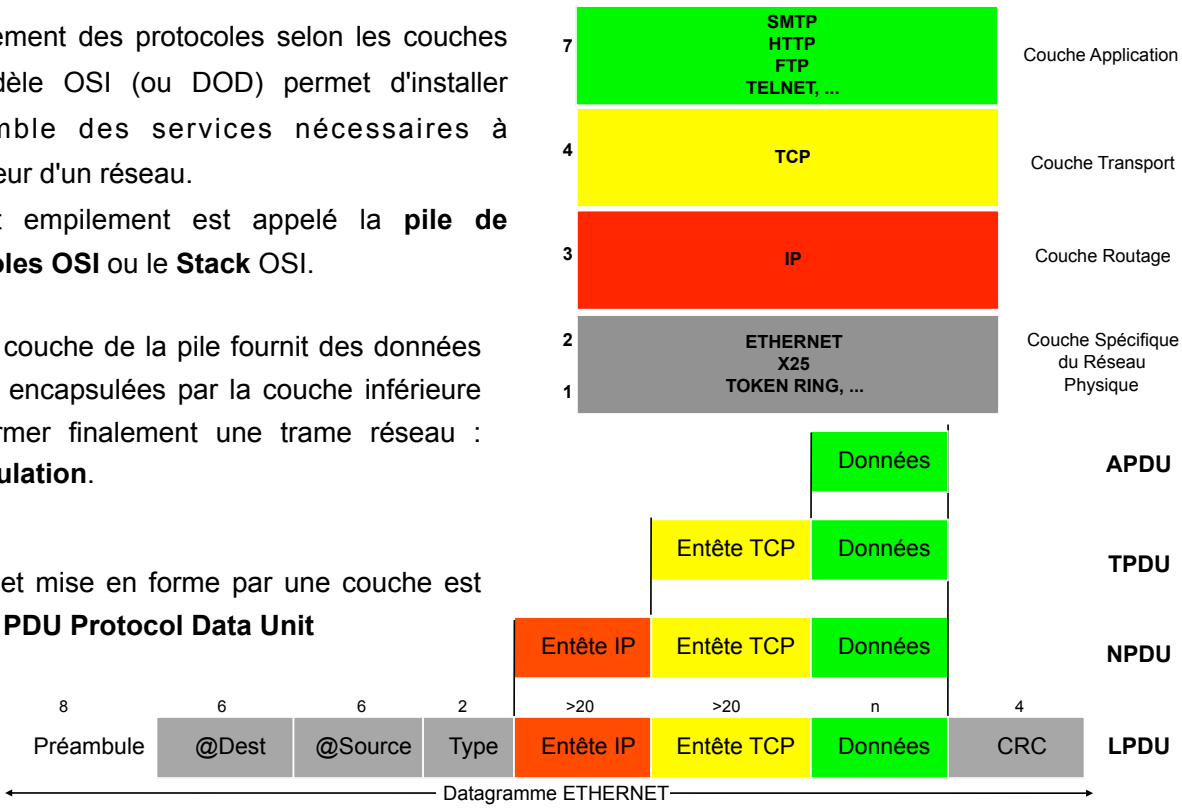
Mécanisme d 'encapsulation

L'empilement des protocoles selon les couches du modèle OSI (ou DOD) permet d'installer l'ensemble des services nécessaires à l'utilisateur d'un réseau.

⌦ Cet empilement est appelé la **pile de protocoles OSI** ou le **Stack** OSI.

Chaque couche de la pile fournit des données qui sont encapsulées par la couche inférieure pour former finalement une trame réseau : **Encapsulation**.

Le paquet mise en forme par une couche est appelé : **PDU Protocol Data Unit**



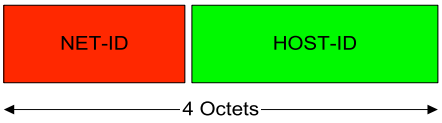
Adressage IP

Adressage IP et Classes de Réseau Internet

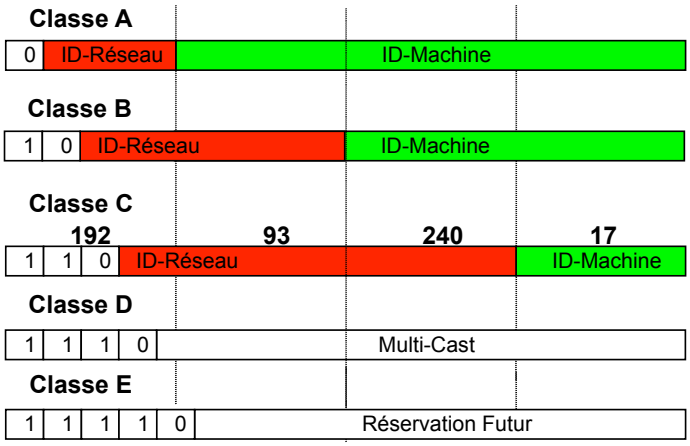
Une @ IP tient sur **4 octets** soit **32 bits**.

192	93	240	17

Une @ IP permet d'**identifier le réseau IP** et la **machine hôte**



3 classes de réseaux Internet basés sur la taille maximale du réseau (nb maxi de stations adressables)



Classe	Plage d'Adressage	Nbre de Réseaux	Nbre de Stations
A	0.xxx.xxx.xxx à 127.xxx.xxx.xxx	127	16 777 214
B	128.0.xxx.xxx à 191. 255.xxx.xxx	16 383	65 534
C	192.0.0.xxx à 223.255.255.xxx	2 097 151	254

<u>Adressage IPv4 (32 bits)</u>
permet de définir **2,1 Millions de réseaux pour un total de 3,72 Milliards d'hôtes**.

<u>Adressage IPv6 (128 bits)</u>
permet d'adresser au minimum **1 Milliard de réseaux**

Masque de Sous - Réseau IP

Une station peut déterminer à quel réseau IP elle appartient ainsi que son numéro de station grâce à son masque de sous – réseau

Station A : IP : **192 . 44 . 01 . 48** Masque de sous réseau : 255.255.255.0

1) Déterminer le réseau IP auquel appartient la Station A : (Net ID)

@IP	192 . 044 . 001 . 48	
		ET LOGIQUE
MASQUE	255 . 255 . 255 . 000	
	192 . 044 . 001 . 000	

2) Déterminer le numéro de la Station A sur son réseau : (Host ID)

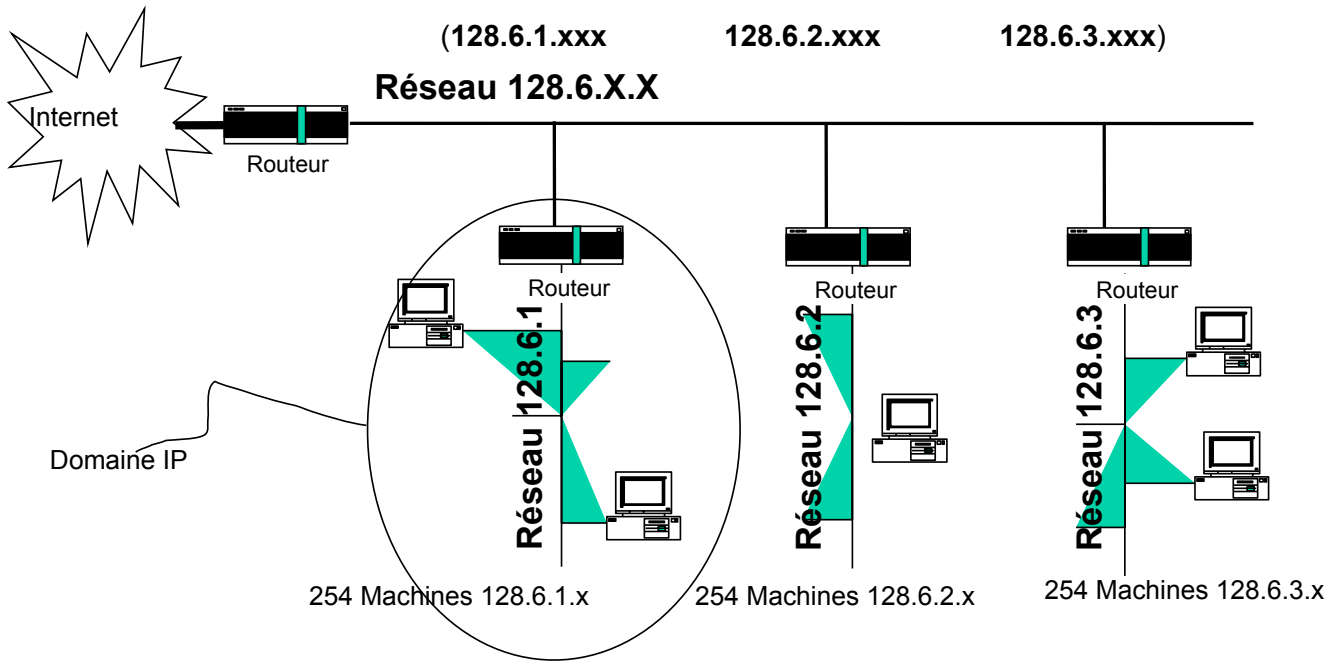
@IP	192 . 044 . 001 . 48	
		ET LOGIQUE
NOT MASQUE	000 . 000 . 000 . 255	
	000 . 000 . 000 . 48	

➪ **La Station A est la Machine N° 48 sur le Réseau IP : 192.44.001 de Classe C**

Segmentation de Réseau Internet

Réseau local d'une entreprise de classe B : **128.6.xxx.xxx.**

Segmenté en sous-réseaux codés avec le 3ème octet

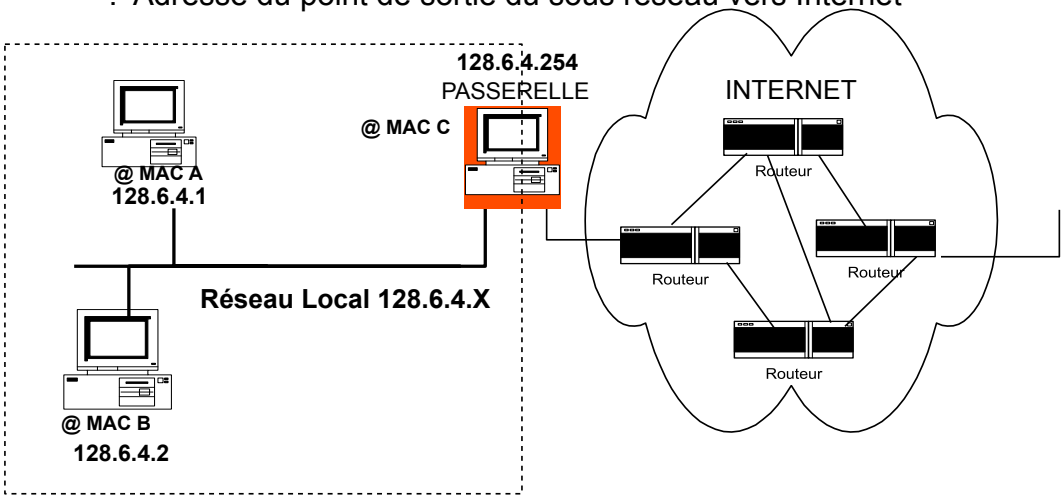


Paramètres d'une Machine Connectée à Internet

Une machine (PC, API) connectée à INTERNET est d'abord connectée à un réseau LOCAL ETHERNET :

4 paramètres sont nécessaires pour communiquer sur Internet

@ MAC	: Adresse Locale Ethernet 48 Bits (souvent inconnue du programmeur)
@ IP Host	: Adresse Mondiale 32 Bits (adresse toujours utilisée)
Masque sous-réseau	: connaître le sous réseau IP de la machine
@ IP Passerelle	: Adresse du point de sortie du sous réseau vers Internet



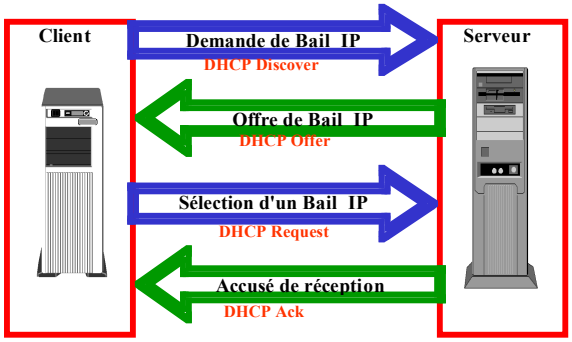
Protocoles DHCP / BootP

Serveur DHCP (Dynamic Host Configuration Protocol)

C'est un logiciel installé sur une machien qui permet d'attribuer des adresses IP dynamiques à chaque poste de travail, et ainsi éviter de configurer manuellement les adresses IP.

DHCP est une extension du protocole BOOTP qui permet à un client sans disque dur (terminal X, imprimante, etc.) de démarrer et de configurer automatiquement TCP/IP.

L'objectif du protocole DHCP est de permettre à un client d'obtenir une adresse IP (et d'autres paramètres éventuellement) auprès d'un serveur DHCP.



En Résumé

Protocole DHCP : Dynamic Host Control Protocol (2)

En-têtes de trames.

No.	Time	Source	Destination	Protocol	Info
1	0.000000	0.0.0.0	255.255.255.255	DHCP	DHCP Discover - Transaction ID 0x6719436e
2	0.001182	192.168.0.253	192.168.0.9	ICMP	Echo (ping) request
3	0.342454	192.168.0.253	192.168.0.9	DHCP	DHCP Offer - Transaction ID 0x6719436e
4	0.344405	0.0.0.0	255.255.255.255	DHCP	DHCP Request - Transaction ID 0x6719436e
5	0.348264	192.168.0.253	192.168.0.9	DHCP	DHCP ACK - Transaction ID 0x6719436e
6	0.353014	CIS_b9:49:37	Broadcast	ARP	Who has 192.168.0.9? Tell 192.168.0.9
7	0.571241	CIS_b9:49:37	Broadcast	ARP	Who has 192.168.0.9? Tell 192.168.0.9
8	1.571441	CIS_b9:49:37	Broadcast	ARP	Who has 192.168.0.9? Tell 192.168.0.9

1. Le client DHCP démarre, il n'a pas d'IP et utilise 0.0.0.0 pour faire un "broadcast général (255.255.255.255)". C'est le DHCP Discover.
2. Notre serveur DHCP, qui a l'intention d'offrir à ce client l'IP 192.168.0.9, fait un ping sur cette adresse, histoire de voir si elle est réellement disponible sur le réseau.
3. Comme il ne reçoit pas de réponse à son ping, il offre cette adresse au client.
4. Le client fait alors un DHCP Request
5. Le serveur accepte
6. Le client fait un broadcast ARP pour vérifier de son côté que l'IP 192.168.0.9 n'est pas dupliquée sur le réseau.
7. idem
8. idem

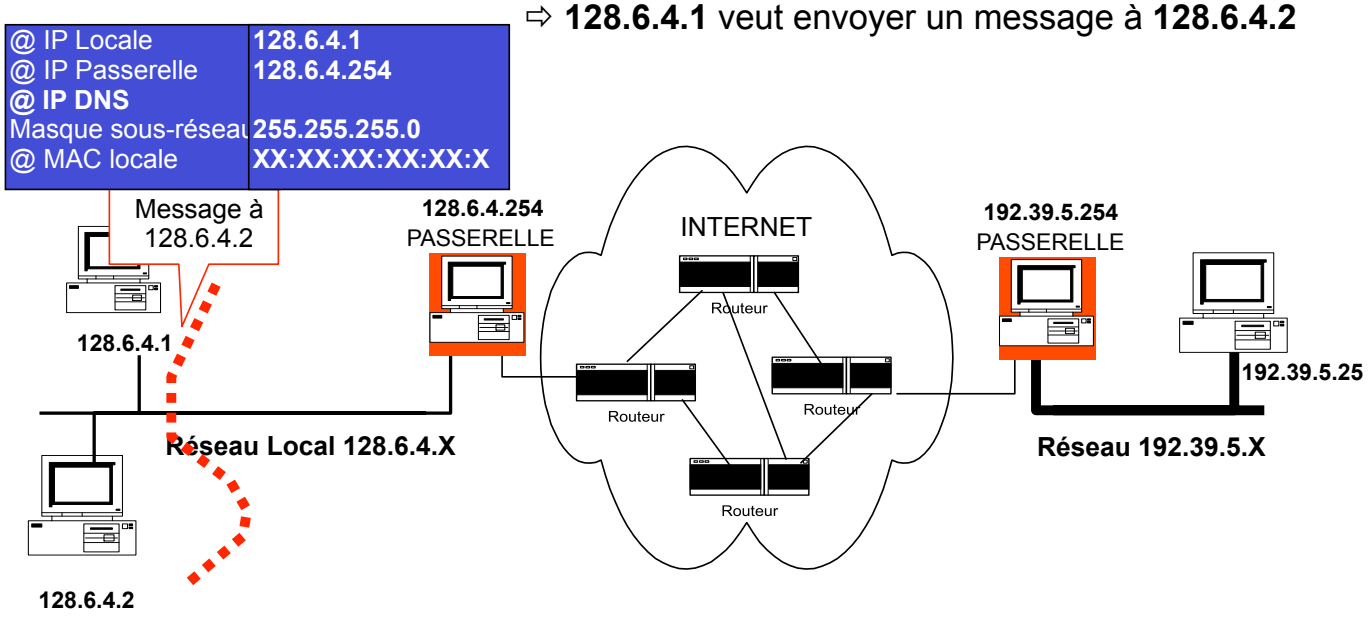
"Routage" d'un Paquet IP

STOP

Réseau Local IP

Les machines source et destination sont sur le même réseau local

⇨ **128.6.4.1** veut envoyer un message à **128.6.4.2**



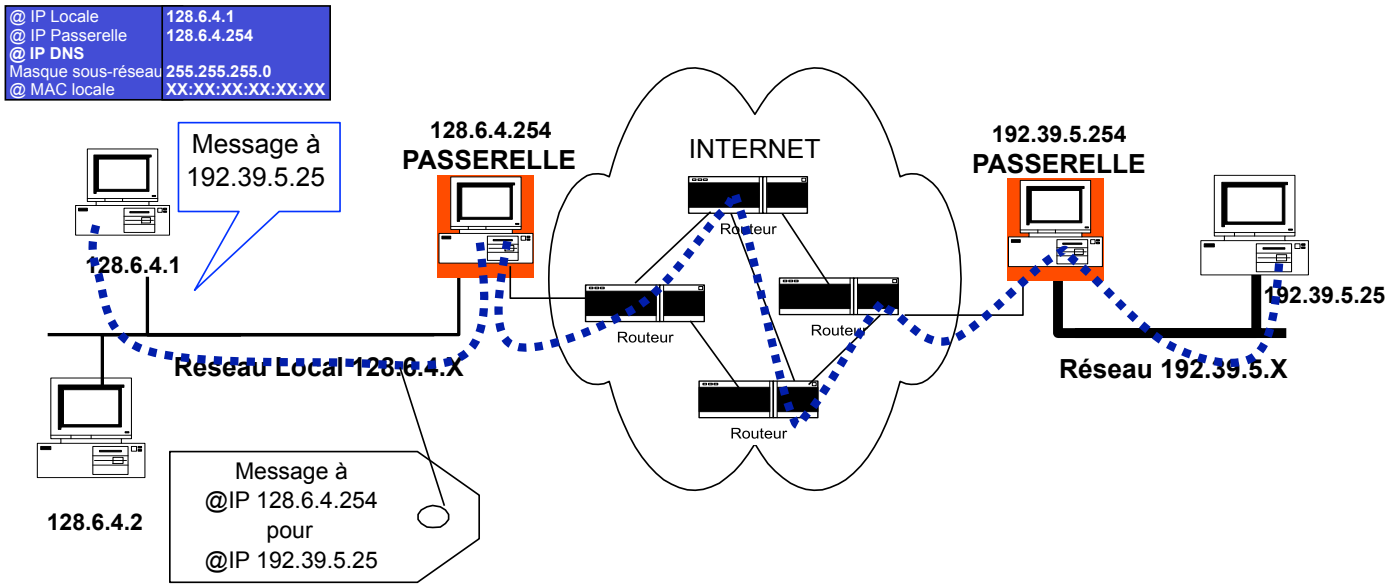
⇨La machine source doit connaître l'adresse physique (@MAC) de la machine dest. pour lui envoyer le message

⇨Pas de routage

Routage d'un Datagramme IP

Les machines source et destination ne sont pas sur le même réseau

⇨ **128.6.4.1** veut envoyer un message à **192.39.5.25**



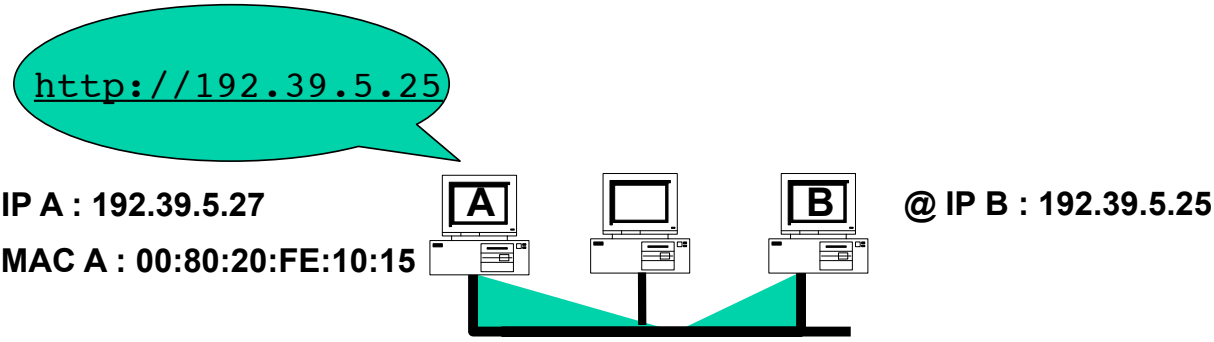
⇨La machine source doit utiliser **une passerelle pour sortir** du réseau local

⇨La passerelle se charge de **router le message** vers le destinataire (Le datagramme est routé sur Internet)

Protocole ARP

Mécanisme de Résolution d'Adresse ARP

La station A veut envoyer un message à la station B
Mais A ne connaît que l'adresse IP de la station B (@IP 192.39.5.25)

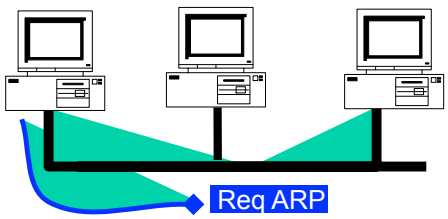


Or Ethernet a besoin de l'adresse MAC du destinataire pour lui envoyer une trame

➜ **Protocole de Résolution d'adresse MAC : Protocole ARP**

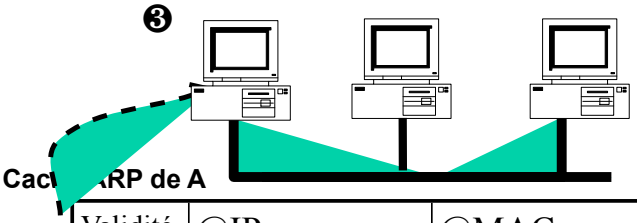
Mécanisme de Résolution d'Adresse ARP (suite)

❶



Trame Ethernet Diffusion : "Qui est @IP=192.39.5.25 ?"

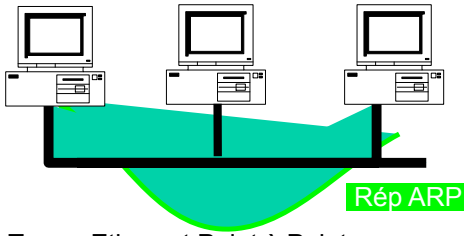
❸



Cache ARP de A

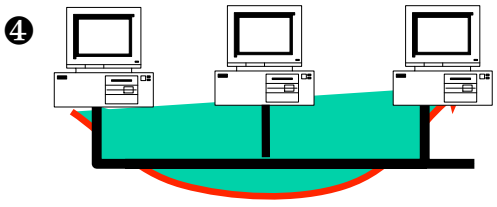
Validité	@IP	@MAC
13:45	192.39.5.25	**00:80:20:FE:50:64**

❷



Trame Ethernet Point à Point :
"@IP=192.39.5.25 :=: **00:80:20:FE:50:64**"

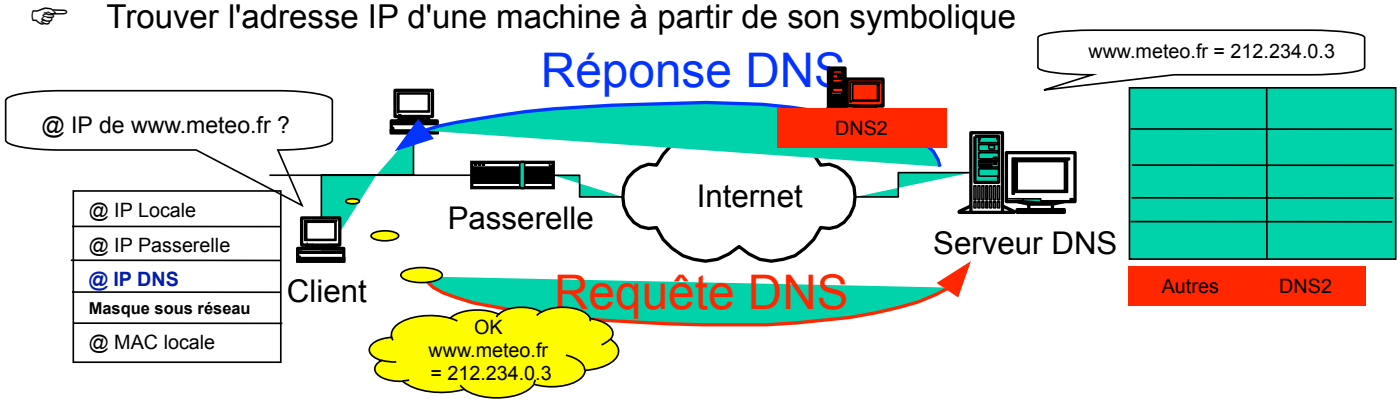
❹



Serveur DNS

Principe d 'un Service DNS

☞ Trouver l'adresse IP d'une machine à partir de son symbolique



❶ **la résolution de nom** : Le client interroge son serveur DNS pour avoir l '@IP du nom symbolique : *www.meteo.fr*

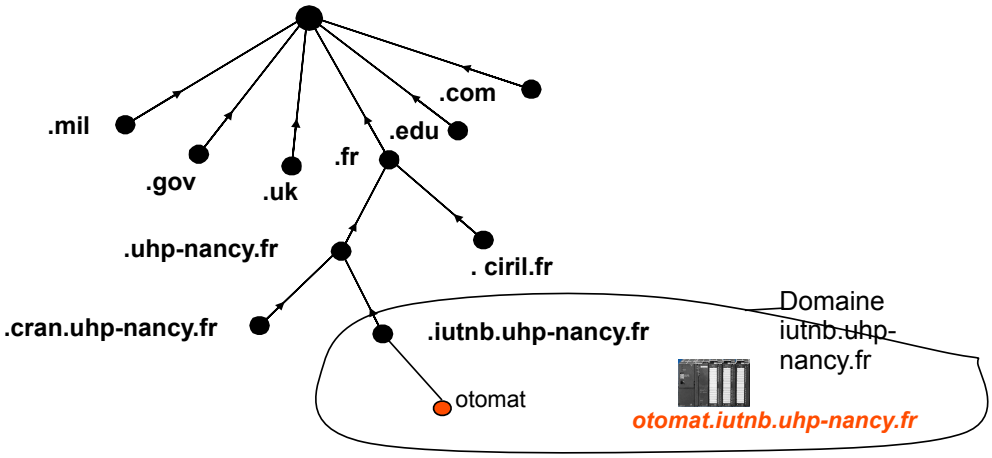
❷ le **serveur DNS interroge d'autres DNS** jusqu'à trouver l'association nom de domaine ⇔ @IP

❸ un **serveur DNS retourne l'adresse IP** du nom demandé : 212.234.0.3

❹ le **client peut maintenant contacter le destinataire** par son adresse IP : *212.234.0.3*

Arborescence des domaines et Nommage des machines

Le nom d'une machine est : ***Machine . Sous-Domaine . Domaine***



Lorsqu'un client a besoin d'un nom ou d'une adresse (Nom ⇔ Adresse IP), il s'adresse au serveur DNS qui gère son sous-domaine.

Chaque sous domaine (noeud du graphe) est géré par un ou plusieurs serveurs DNS

Le serveur DNS relaie la demande au niveau supérieur s'il ne connaît pas le nom recherché (arborescence)

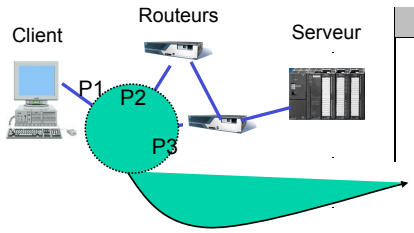
Communication Industrielle

Protocole IP

Internetwork Protocol RFC 791

Principe de Routage IP

- Le routage est le processus permettant à une **Trame** d'être acheminé vers son destinataire quand celui-ci n'est pas sur le même réseau physique que l'émetteur.
- Le chemin parcouru par une trame est le résultat du **processus de routage** qui effectue les choix nécessaires afin d'acheminer le datagramme.
- Un **routeur ne connaît jamais le chemin complet** pour atteindre la destination.
- **Les routeurs forment une structure coopérative** : un datagramme transite de routeur en routeur, jusqu'à ce que l'un d'entre eux le délivre à son destinataire.
- Chaque routeur dispose d'une **table de routage IP**, indiquant la manière d'atteindre les destinations



Destination	Prochain Routeur	Interface
183.27.0.0	Connexion directe	P1
172.9.0.0	Connexion directe	P2
137.5.0.0	183.27.2.5	P1
140.4.0.0	172.9.1.2	P3
0.0.0.0	183.27.2.5	P1 (par défaut)

Routages Statique / Dynamique

On distingue 2 types de routage IP :

1. Routage statique : On entre manuellement la table de routage dans le routeur : Configuration statique.

2. Routage Dynamique : Le routeur gère dynamiquement et construit automatiquement sa table de routage selon des protocoles d'échanges entre routeurs : **protocole de routage**.

Un Protocole de routage permet à des routeurs d'échanger des informations pour construire la meilleure route jusqu'à une destination sur Internet.

Un protocole de routage s'exécute au-dessus d'IP :

➔ **RIP** : Routing Information Protocol

➔ **OSPF** : Open Shortest Path Time

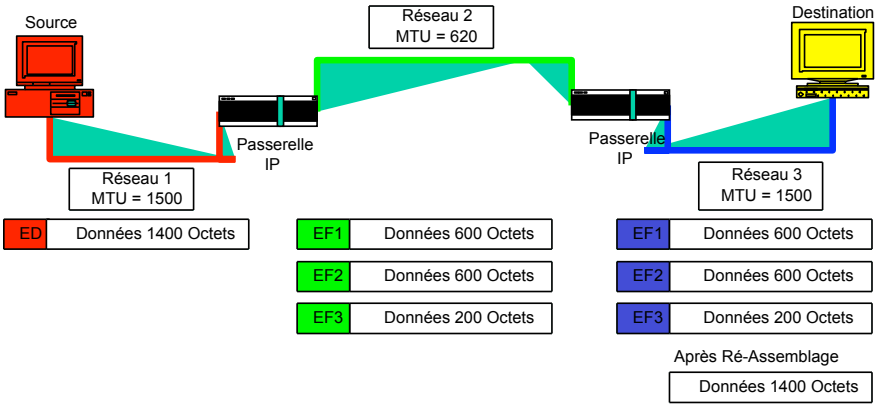
Protocole IP : Internetwork Protocol

Le protocole **IP** (RFC 792) a été conçu pour s'adapter à des types de réseaux différents.
La taille maxi. des données admissibles par un réseau (MTU) varie selon la nature du réseau

⌦ IP s'adapte par un mécanisme de **fragmentation** de paquets.
Maximum Transfert Unit (**MTU**) : Taille maximale en octets des données transportées par un réseau

Ethernet : MTU = 1500 octets **X25** : MTU = 1007 octets **Token Ring** : MTU = 4440 à 17940 octets f (temps conservation jeton) **ATM** : MTU = 32

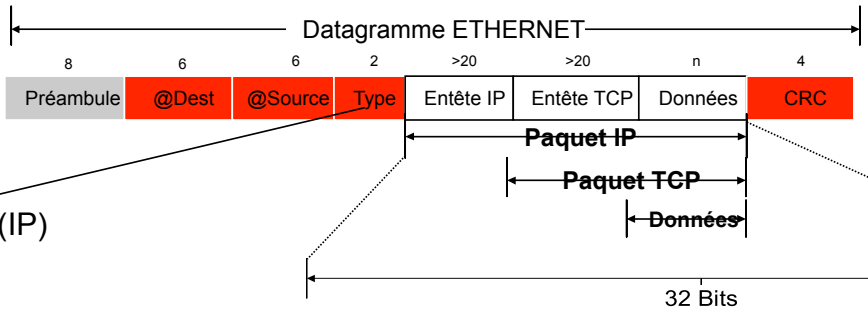
IP adapte la taille des paquets de données aux MTU des différents réseaux traversés, en découpant le paquet en paquets plus petits : c'est un rôle majeur des routeurs sur Internet appelé **fragmentation**



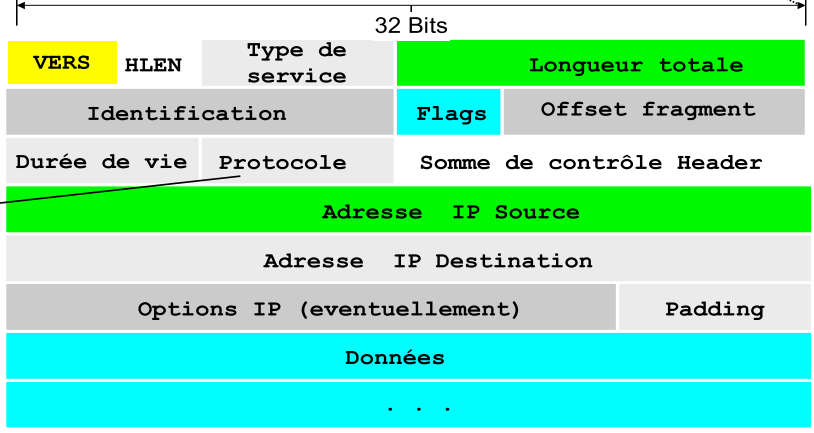
Le **ré-assemblage** est effectué par la couche IP du destinataire **jamais par les routeurs**.

Format de l'Entête IP

Le protocole IP encapsule les données générées par les couches 7 à 4 pour former **un paquet IP**.



Type = 0800 (IP)



Un **paquet IP** est constitué d'une entête de taille minimum 20 octets suivie par les données IP

Protocole : Indique le protocole de niveau supérieur encapsulé

(6 : TCP, 17 : UDP, ...)

Format de l ’Entête IP

VERS : Version IP (4 : IPV4, 6 : IPV6 IPng)
HLEN : Longueur de l'entête IP en mots de 32 bits
Longueur Totale : Long. de l'entête et des données IP en octets.
Identification : Numéro du datagramme IP.

Type de Service : TOS : Qualité de service demandée

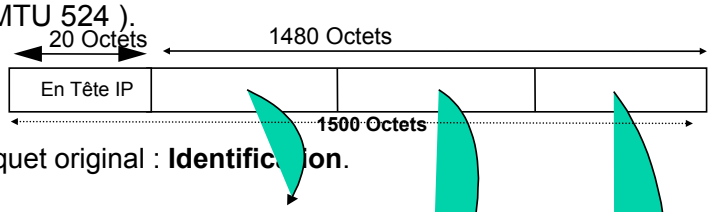
Bits 0-2	1	1	1	1	1
Préséance	Délai	Débit	Fiabilité	Coût	MBZ
000 Routine 001 Priority 010 Immediate 011 Flash 101 Critical 110 Internetwork Control 11 Network Control	0 Normal 1 Faible	0 Normal 1 Elevé	0 Normal 1 Elevé	0 Normal 1 Faible	

Flags : DF = 1 : (Do not Fragment) Le Datagramme ne doit pas être fragmenté
MF = 1 : (More Fragments) Le destinataire est informé que d'autres fragments vont arriver
Offset Fragment : Offset sur 13 bits des données contenues dans le datagramme IP depuis le 1er fragment
Durée de Vie : Compteur de sauts (**hops**) décrémenté à la traversée de chaque routeur. Si =0 le routeur stoppe le paquet et renvoi un message ICMP. Limite la durée de vie des datagrammes sur Internet.
Protocole : Indique le protocole de couche supérieure (6 : TCP, 17 : UDP, 1 : ICMP,….)
Somme de contrôle header : Checksum sur 16 bits de l'entête sans les données IP. Recalculé à chaque routeur car TTL change.
Adresses IP Source et Destination
Options : Indique le niveau de sécurité, information de routage, d'horodatage : rarement utilisées.
Padding : Sert à compléter l'entête IP en nombre de mots de 32 pleins.
Données : Données encapsulées à destination du protocole de niveau supérieur.

Fragmentation d'un datagramme IP

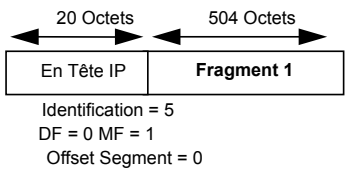
Dans un routeur, la couche IP se charge de fragmenter un datagramme afin de s'adapter aux MTUs des réseaux qu'elle dessert (MTU = 1500 vers MTU 524).

Processus de fragmentation

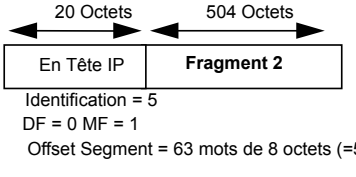


⇨ Les fragments appartiennent au même N° de paquet original : **Identification**.

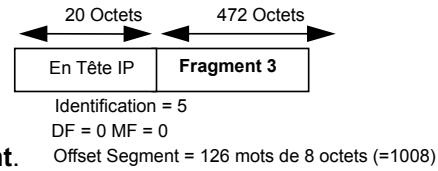
⇨ Premier Fragment : **DF** = 0 et **MF** = 1



⇨ Deuxième Fragment : **DF** = 0 et **MF** = 1



⇨ Dernier Fragment : **MF** = 0 (No More Fragment).



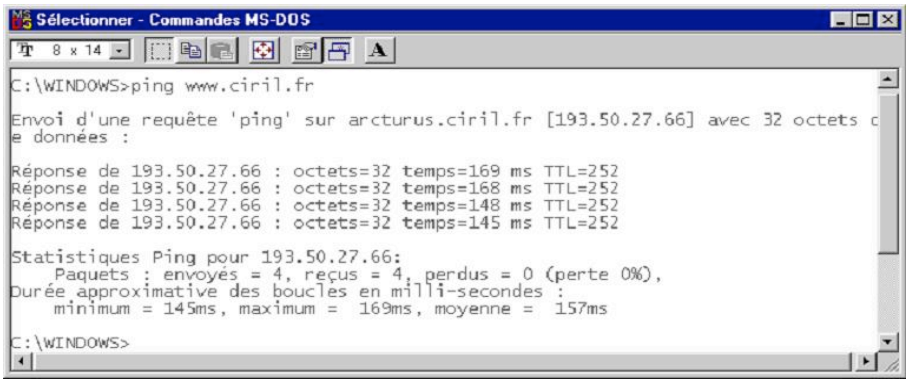
⇨Le réassemblage dans l’ordre sera effectué sur la base des **offset de fragment**.

La commande PING

La commande **ping** (Packet Internet Groper) permet de vérifier qu'une machine est connectée et accessible sur le réseau IP (Internet)
C'est une commande interne au protocole IP implémentée dans le système d'exploitation de la machine

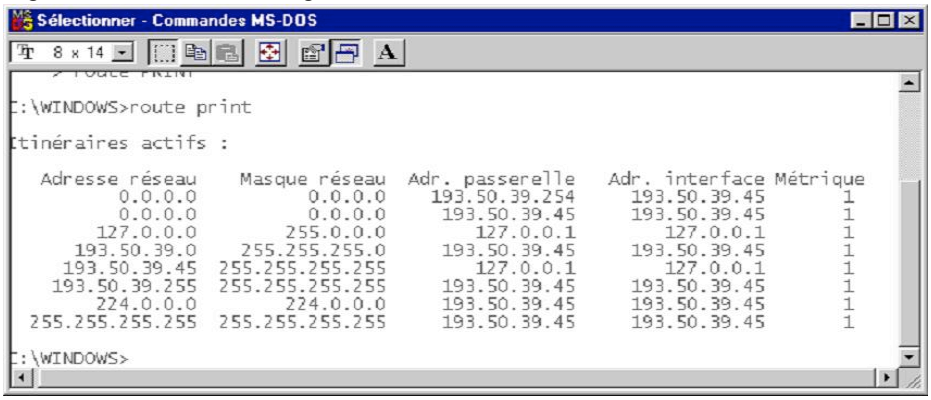
- Envoi des paquets (32 octets de données ASCII 'a' à 'z' …) vers l '@ cible
  : Commande ECHO du protocole **ICMP** (Internet Control Message Protocol)
- Attente des réponses, jusqu'à 1 seconde pour chaque paquet.
- Vérification des paquets reçus (32 octets de données ASCII 'a' à 'z' …) , et mesure du temps d'attente.

STOP



La commande Route

Cette commande gère les tables de routage de la couche réseau IP de l'ordinateur hôte.



1. Adr réseau : 0.0.0.0 : C'est la route par défaut que les paquets vont prendre lorsqu'ils n'on pas trouvé un meilleur chemin. C'est la ligne la plus intéressante, parce qu'elle fait intervenir une adresse de passerelle (193.50.39.254) et une adresse d'interface (193.50.39.45) différentes.

3. Adr réseau 127.0.0.0 : C'est la boucle interne, celle qui permet à l'hôte de se parler à lui même.

4. Adr réseau 193.50.39.0 : c'est le réseau local

5. Adr réseau 193.50.39.45 : Pour atteindre 193.50.39.45 , c'est à dire moi-même, il faudra utiliser 127.0.0.1 (adresse interne toujours la même sur tous les hôtes quelque soit l'OS).

6. Pour réaliser un broadcast sur mon réseau, il faudra utiliser 193.50.39.45

7. Si l'on souhaite faire du multicast, même chose

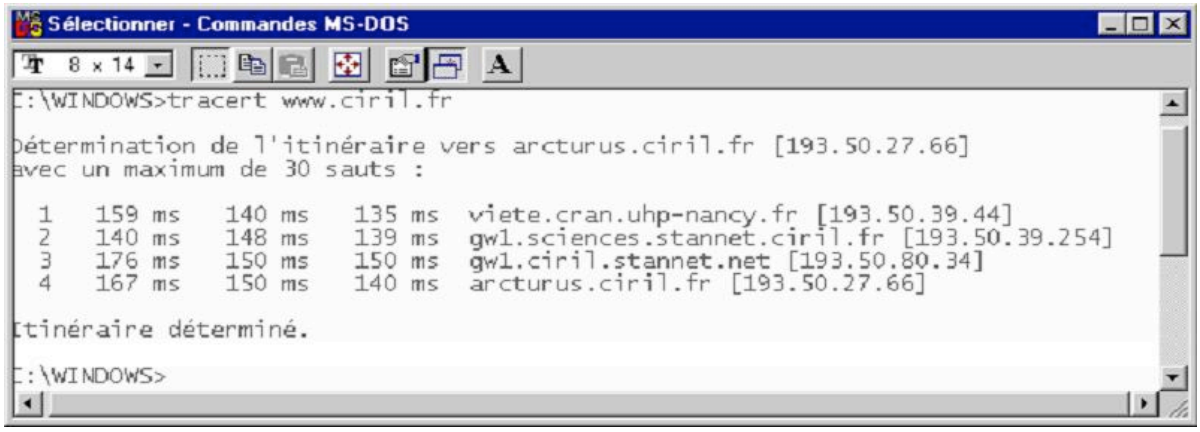
8. Si l'on souhaite faire du broadcast étendu, encore la même chose.

La commande Tracert

La commande **tracert** (TRACE RouTer) permet de connaître tous les routeurs traversés pour accéder à une station d '@IP donnée.

• Exécute un ping vers une station distante avec un paramètre interprété par chaque routeur traversé lui signifiant de répondre par son adresse.

• Attente des réponses et mesure du temps d'attente.

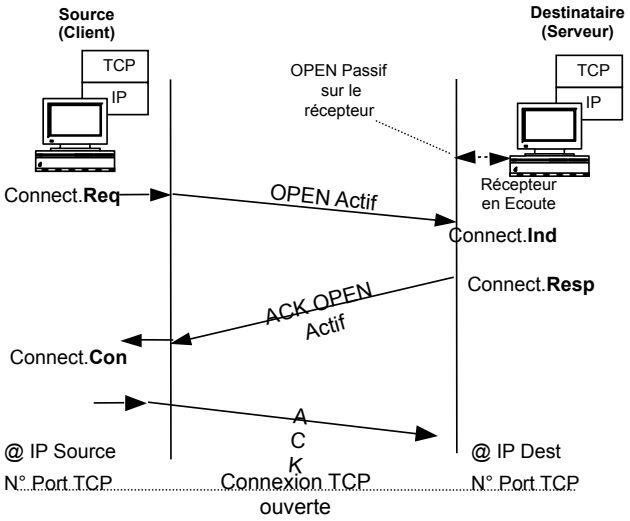


Protocole TCP

Transmission Control Protocol RFC 792

Ouverture d'une connexion TCP

Une connexion TCP est établie en 3 temps de manière à assurer la synchronisation entre Client et Serveur



Emetteur	Destinataire
(@ IP, port TCP) **(124.32.12.1 , 1512)**	(@ IP, port TCP) **(19.24.67.2 , 1538)**

Le serveur se met en écoute : OPEN passif.
La couche TCP serveur se met en attente de connexions.

Un client fait une demande de connexion : OPEN actif.
La couche TCP cliente spécifie les ***sockets*** source et destination liées à cette connexion.
Le Serveur Accepte ou Refuse la connexion (Nb limité)

Une connexion est établie entre l'émetteur et le destinataire caractérisés chacun par un couple **(@ IP, N° port TCP**), appelé une **SOCKET**

Gestion d'une connexion TCP

L 'envoi de trame réseau Internet est réalisé après **ouverture préalable d 'une connexion** entre l'émetteur et le destinataire.

Les services fournis par la couche TCP sont :

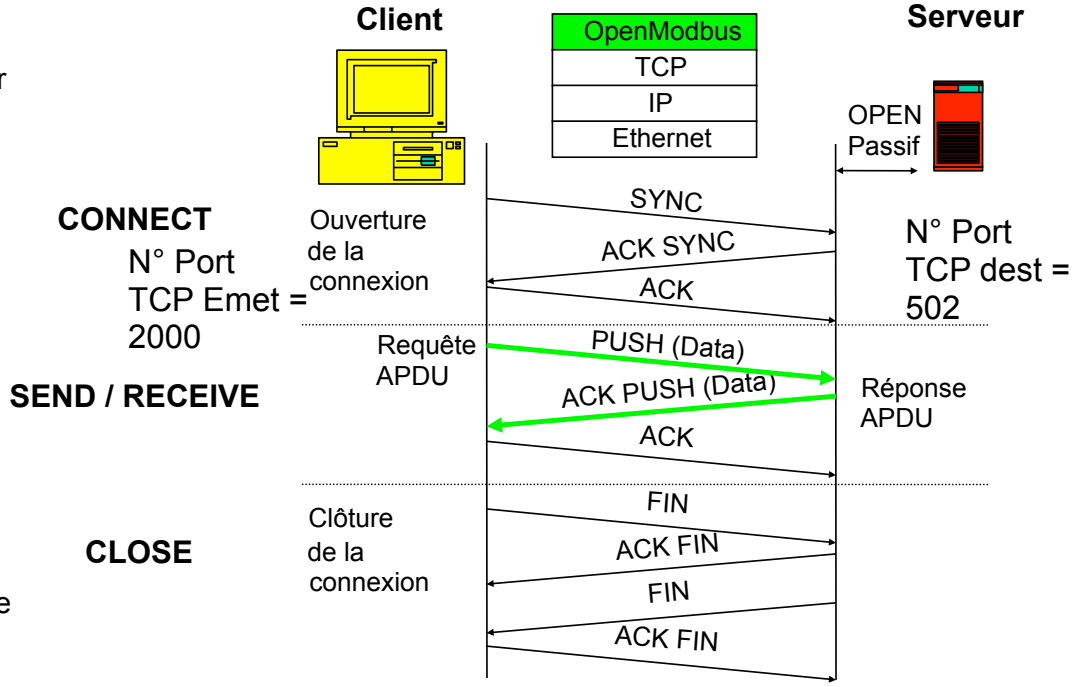
CONNECT ouverture d'une connexion

CLOSE : fermeture d'une connexion

SEND : envoyer les données sur une connexion ouverte

RECV : recevoir les données sur une connexion ouverte

STATUS : fournir des renseignements sur une connexion ouverte

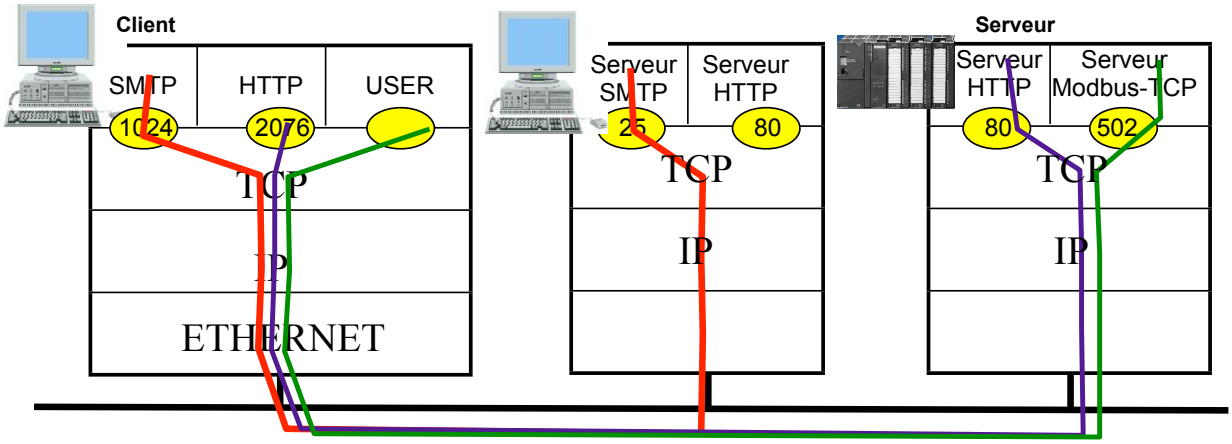


Multiplexage des connexions TCP

☞ TCP peut servir plusieurs applications en même temps sur la même machine source :

⇨ Principe de multiplexage.

☞ TCP associe un numéro de port à chaque application qui utilise ses services. STOP

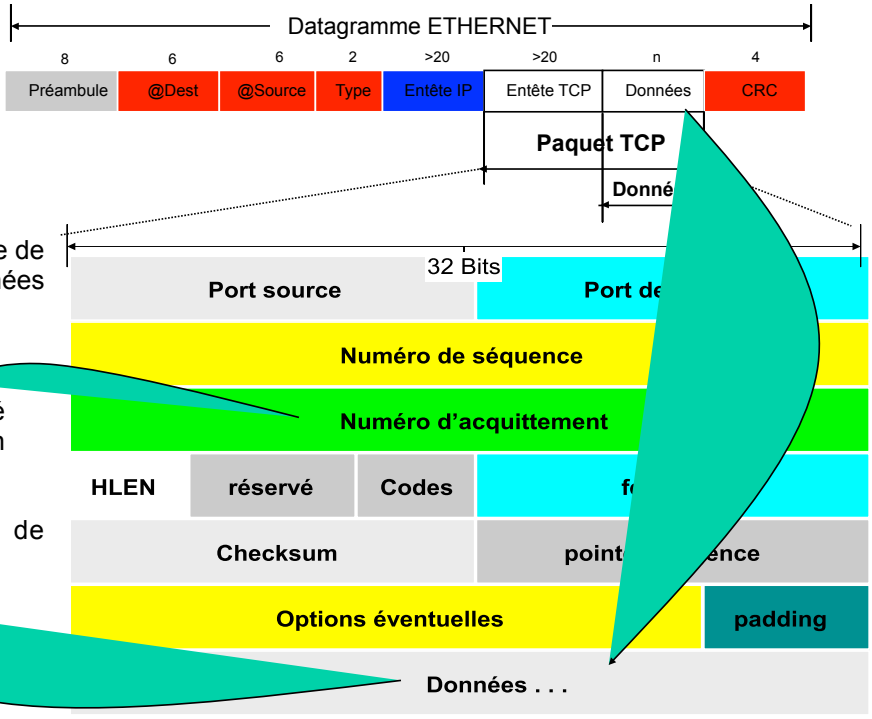


Port TCP réservés : 20, 21 : FTP, 23 : TELNET, 25 : SMTP, 80 : HTTP, 110 : POP3, 512 : MODBUS-TCP…

N° Port libres utilisateur > 1023

Format de l 'entête TCP

Le protocole TCP encapsule les données générées par les couches 7 à 5 pour former **un paquet TCP**



Un paquet TCP est constituée d'une entête de 20 octets minimum suivie par des données TCP

Chaque paquet TCP sera acquitté assurant un transport fiable de bout en bout

Les Données TCP sont constituées de l'APDU de la couche supérieure

Format d ’entête TCP

L'entête TCP à une longueur minimale de 20 Octets, qui peut augmenter avec les options.

Port Source, Port Destination : ports en connexion des applications origine et destination.

Numéro de séquence : Le numéro du premier octet est le numéro de séquence.

Numéro d'acquittement : prochain numéro de séquence attendu par l’émetteur de cet acquittement.
Acquitte implicitement les datagrammes reçus avec les octets précédents.

HLEN : Code sur 4 bits la longueur de l'entête TCP en mots de 32 bits

Codes : 6 bits de drapeau d'acquittement

URG	: Indique l'envoi des données en urgence sans attente
ACK	: Indique que le champ "Numéro d'acquittement est valide"
PSH	: Indique la remise immédiate des données à la couche supérieure
RST	: Demande de ré-initialisation de la connexion
SYN	: Indique l'ouverture d'une connexion de circuit virtuel
FIN	: Indique la fermeture de la connexion

Fenêtre : Taille de la fenêtre d'anticipation.

Checksum : Somme complémentée à 1 des compléments à 1 de tous les mots de 16 bits du paquet TCP

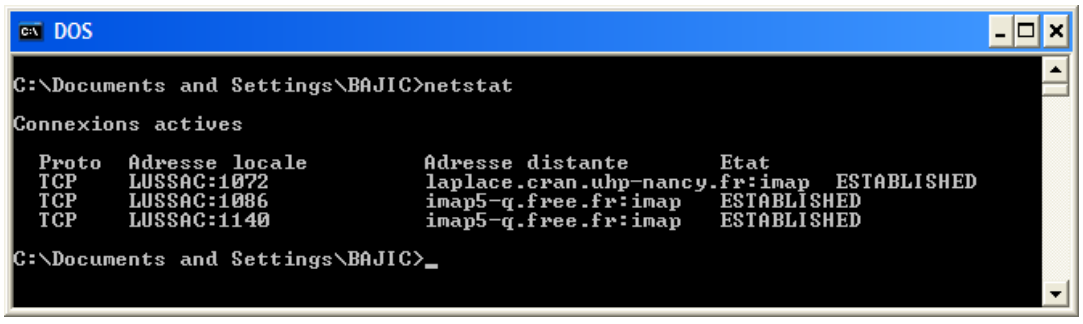
Pointeur d'urgence
Options : Permet de négocier la taille maximale des segments échangés. TCP calcule une taille maximale de segment de manière à ce que le datagramme IP résultant corresponde au MTU du réseau. La recommandation est de 536 octets.

Padding : Sert à complété en nombre de mots de 32 pleins l'entête TCP.

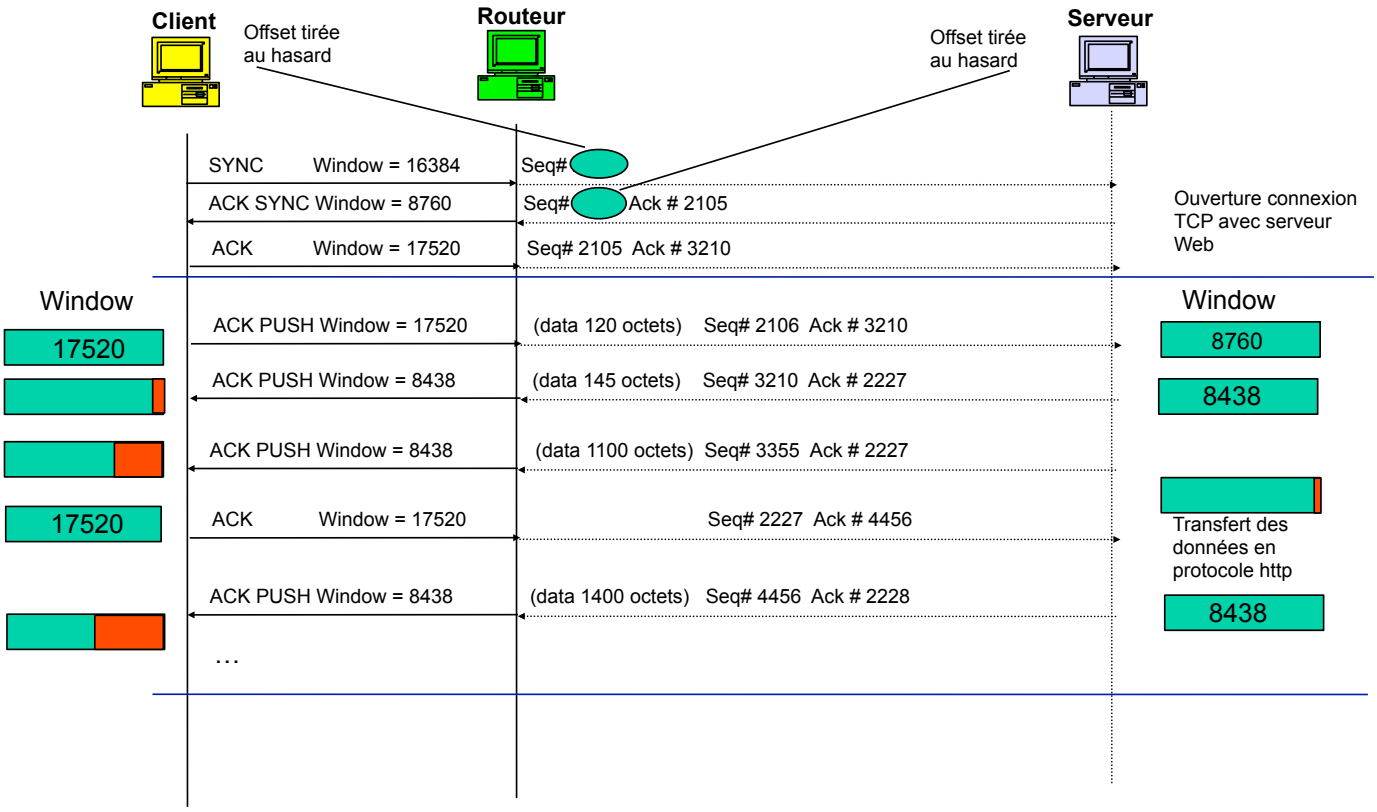
Données : Données encapsulées à destination du protocole de niveau supérieur.

Ports TCP ouverts

La commande NETSTAT.EXE permet de connaître les connexions ouvertes et les ports utilisés par la couche TCP



Acquittement TCP par Numérotation des Octets



Protocole UDP

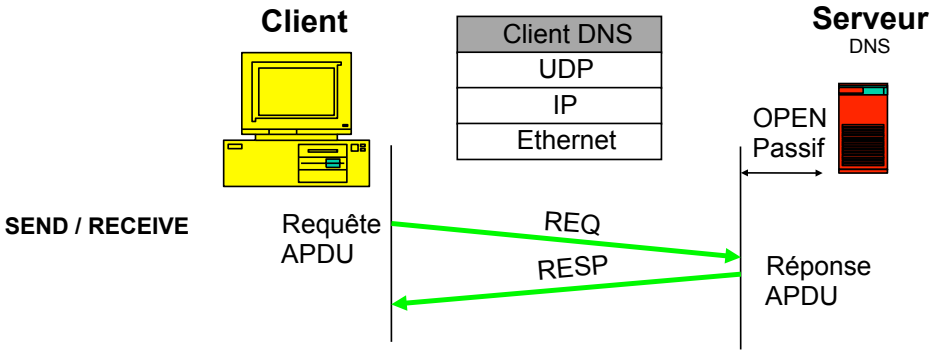
User Datagram Protocol RFC 768

Requête APDU en Mode non connecté UDP

L ’échange avec UDP est de type Question/Réponse sans connexion préalable.

⇨ Le transport de bout en bout n’est pas fiable car la réception d’une trame n ’est jamais acquittée.

⇨ Seule la réponse transmise permet de savoir si la requete est bien arrivée (Time Out)



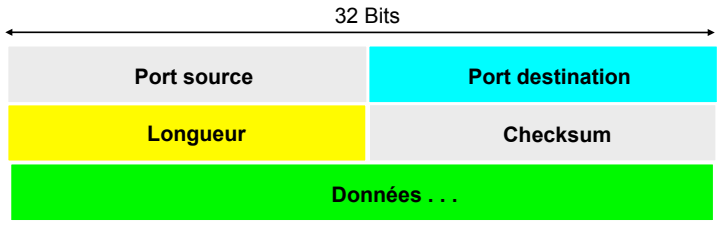
Les échanges UDP sont très rapides comparés à TCP :

⇨ **Pas de connexion**

⇨ **Pas de fragmentation de paquets**

Protocole UDP

L'entête UDP, très simple, est de taille fixe de 8 octets



Port Source, Port Destination : Indiquent les N° port des applications émetteur et destinataire sur 16 bits .

Longueur : Longueur totale du paquet UDP, comprenant l'entête de huit octets plus les données UDP.

Checksum : Somme complémentée à 1 des compléments à 1 de tous les mots de 16 bits du paquet UDP, avec un remplissage de zéros pour obtenir une longueur multiple de deux octets, auquel est ajoutée une pseudo-entête telle que ci-dessous

Ethernet Industriel

Modbus-TCP

Selon Spécification Schneider Mars 1999

(www.modbus.org)

Pr. Eddy BAJIC
IUT Nancy Brabois
Nancy Université

Pourquoi un Ethernet Industriel ?

Mais Pourquoi Utiliser Ethernet en Automatisation Industrielle ?

Avantages *Standard Mondial, Banalisé,*
Peu coûteux, Multi-usages
Rapide,
Protocoles TCP-IP et au dessus : HTTP (Web), SMTP (Mail), FTP (Fichiers) …

Inconvénients *Non déterministe (CSMA-CD, Domaine de Collision)*
Non temps réel (Accès Aléatoire, Probabiliste)

Besoin de Solutions "Ethernet Industriel" Adaptée aux Contextes Industriels

Fin des Domaines de Collision : ***Commutation, Full Duplex*** *(Switch, ASIC switch)*

Des Protocoles d'application dédiés pour l'Industrie : ***Bataille pour un standard***

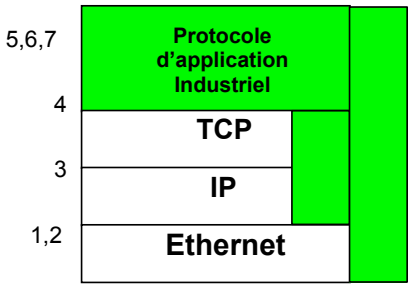
La quête du déterminisme : ***Temps réel, Isochronisme***

Sites Internet Intéressants :

http://ethernet.industrial-networking.com

www.iona.org Assoc promotion Ethernet dans l'automatisation

Solutions Ethernet Industriel



Ethernet Industriel : http://ethernet.industrial-networking.com/

1) Ethernet TCP-IP Standard

2) Court circuiter TCP-IP : Gain de Temps

3) Circuit Ethernet spécifique : Switch priorité intégrée

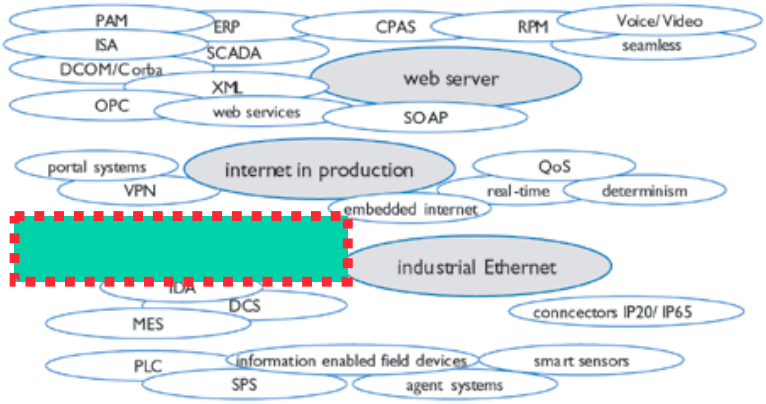
SIEMENS : PROFINET

ROCKWELL : ETHERNET / IP

SCHNEIDER : MODBUS-TCP

BECKHOFF : ETHERCAT

Web Technologies

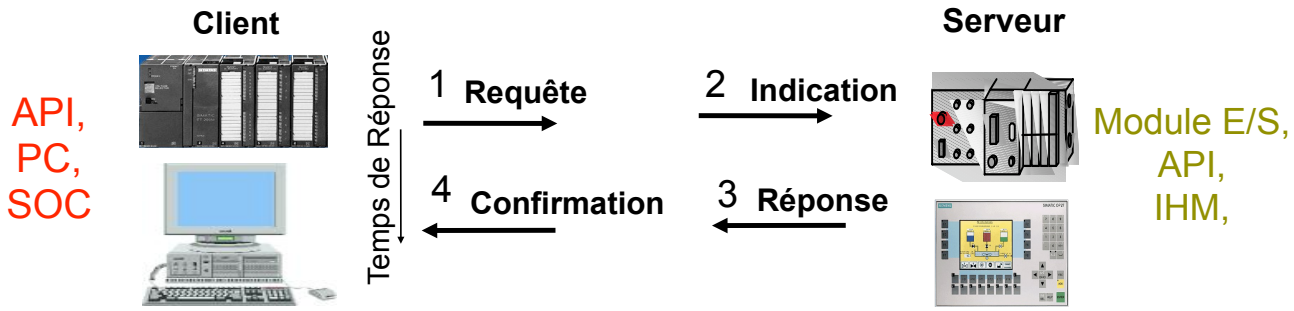


Modbus –IDA Architecture for Distributed Automation : http://www.modbus-ida.org

Modbus TCP conformance test laboratory : http://www.eecs.umich.edu/~modbus/

Modèle Client – Serveur TCP/IP

L'architecture Internet est basée sur un modèle de communication Client-Serveur



Client : Sollicite un service du Serveur par une Requête

Serveur :

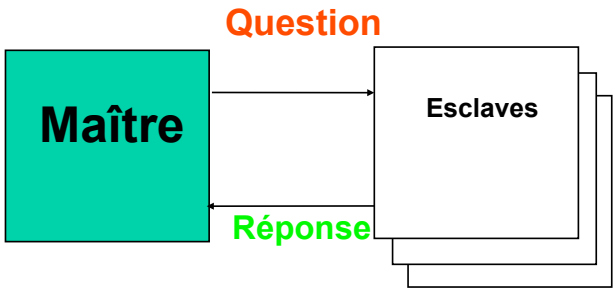
- Est à l'écoute des clients qui le sollicitent (écoute sur N° de port TCP/UDP donné)
- Ne répond qu'aux requêtes des clients

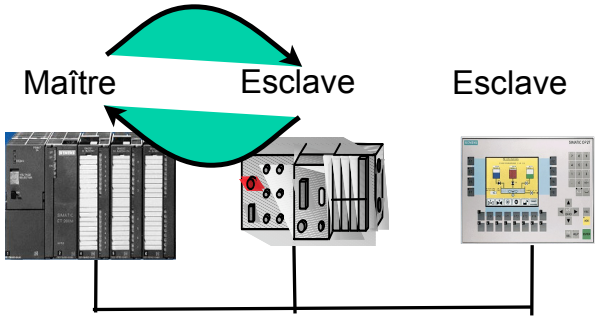
Temps de Réponse : Non garanti, indéterminé, il dépend de l'architecture réseau traversée

Modèle Maître / Esclave Vs Client / Serveur

Modèle Maître / Esclave (MODBUS-RTU)

Principe de Communication par Question/Réponse





Bus RS 485

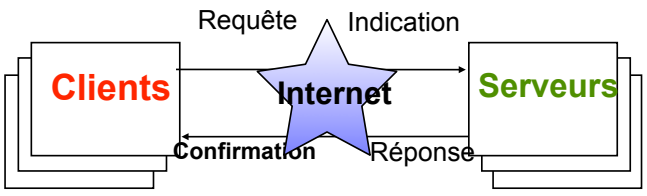
- 1 Maître unique
- Plusieurs Esclaves (@1 à @63)
- Transactions séquentielles

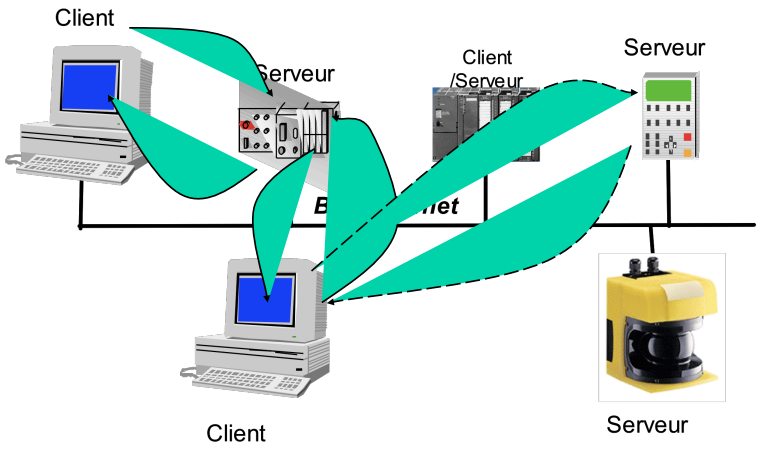
à 19 200 Bits/s
⇨ 1 transaction : Lecture ou Ecriture de 30 Mots de 16 Bits
≈ 100 ms

Modèle Maître / Esclave Vs Client / Serveur

Modèle Client / Serveur (MODBUS-TCP)

Principe de Communication par Requêtes Multiples





- Clients Multiples (@IP clients)
- Serveurs Multiples (@IP serveurs)
- Transactions Simultanées

à 10 MBps
⇨ 1 transaction : Lecture ou Ecriture de 100 Mots de 16 Bits
≈ 100 ns

Comparatif Performances Modbus-RTU / Modbus-TCP

Les performances temporelles dépendent du réseau, du trafic et du matériel :

MODBUS RTU liaison série : Asynchrone, 11 Bits / Caractère , à 19200 Bits/s, 0,6 ms / Caractère
Lecture 10 Mots # 48 ms (transaction réseau)
⇨ Soit **Lecture 208 Mots par seconde** (Supervision, commande processus lent)

MODBUS-TCP en Intranet : Ethernet 10 Base T offre un débit réel # 1.25 Mbytes/sec
Lecture 10 Mots, rendement encapsulation de # 25 % : 12/66 (Requête) et 24/78 (Confirmation)
⇨Soit 1.25M / 2 * 25% = 156 250
⇨**Lecture 156 250 Mots par seconde** (Supervision, commande processus)

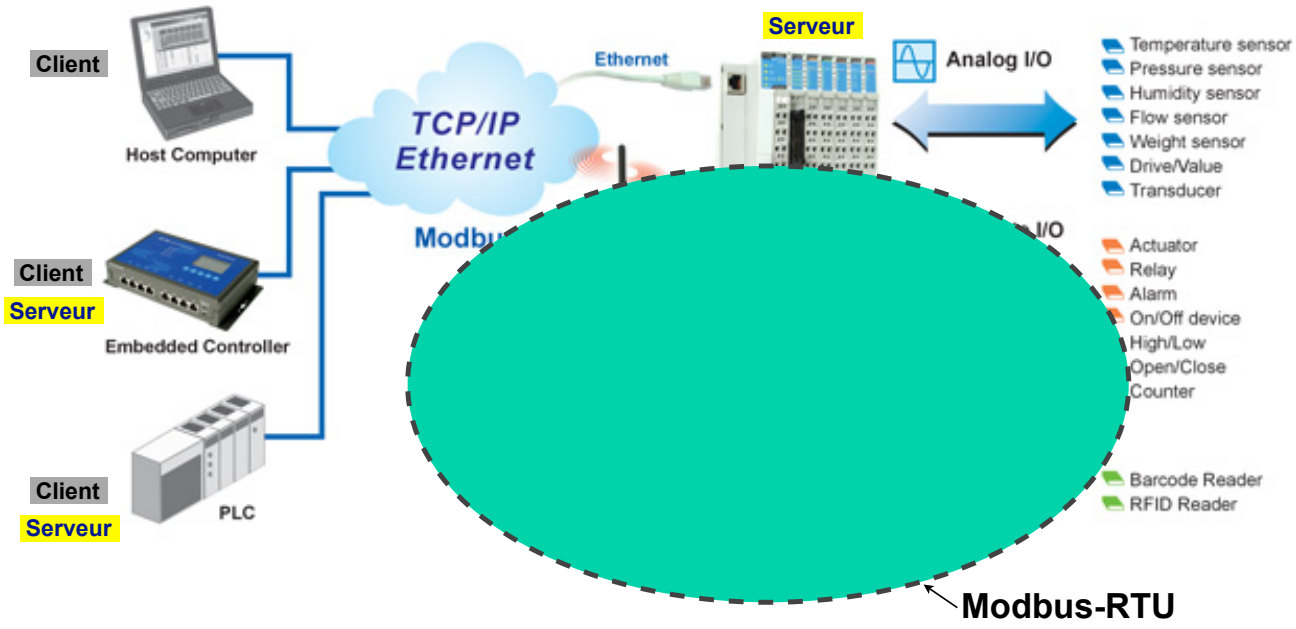
MODBUS-TCP Via Internet : Temps de réponse lié à la charge du réseau Internet
Dizaine de ms à plusieurs centaines de ms (supervision, consultation, … à distance)
⇨**Temps variable non garanti** (supervision, consultation, … à distance)

REMARQUE : Ce sont des performances idéales au maximum des bandes passantes : ⇨ Il fait tenir compte des temps de cycle des équipements API, …

Test effectué par Schneider Automation, MOMENTUM Ethernet PLC, 4000 nœuds interrogés par secondes avec chacun 16 voies analogiques 12-bit ou 32 entrées TOR : 4 nœuds scannés / milliseconde

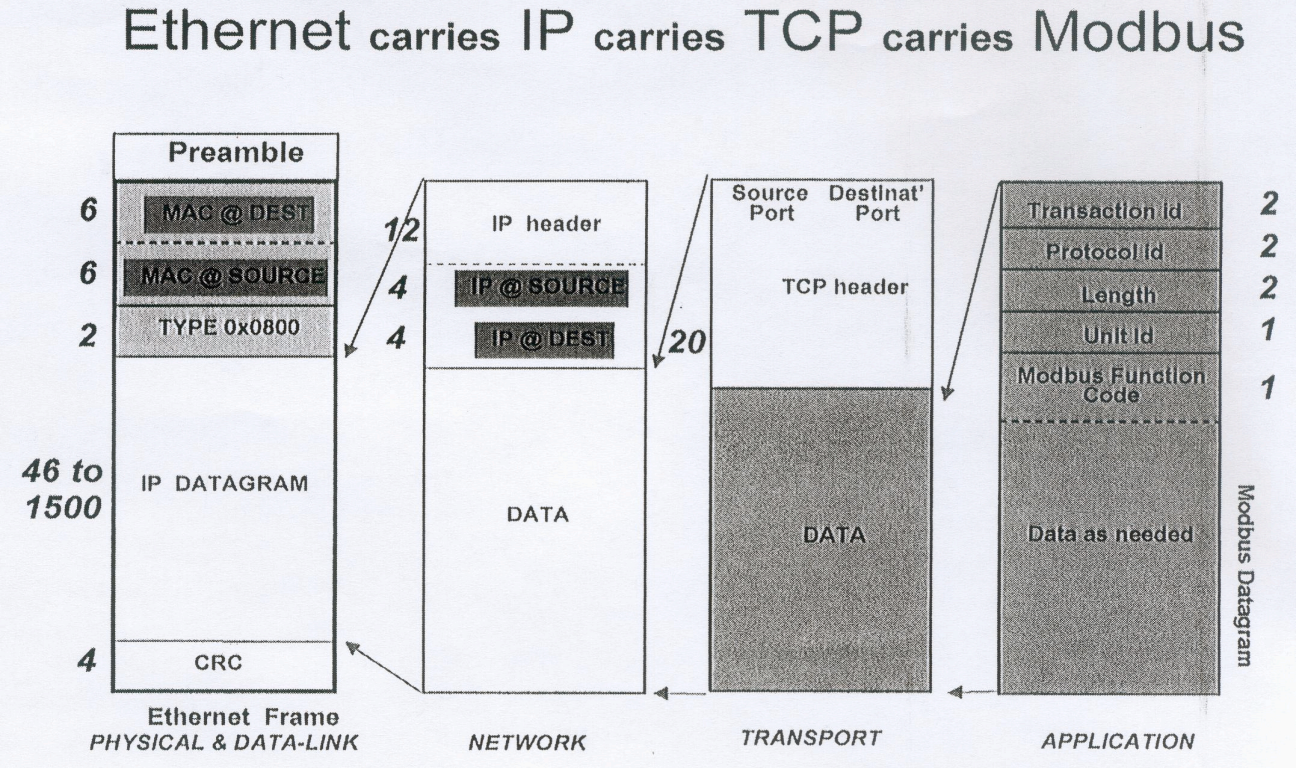
Les E/S de terrain sur Modbus-TCP

Accéder aux E/S d'un processus à Distance en Mode Client / Serveur



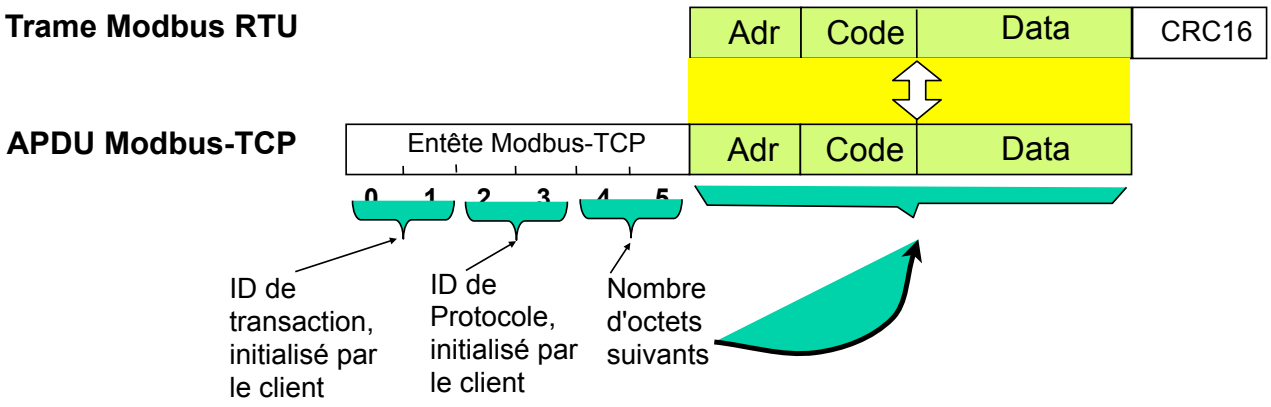
http://www.moxa.com/product/iologik-4000.htm#top

Encapsulation Modbus-TCP/TCP/IP/Ethernet



Format APDU MODBUS-TCP

★ La trame bien connue du protocole Modbus est **réutilisée**
★ La Trame Modbus est **complétée d'une entête spécifique**.
★ Le **CRC16 Modbus est supprimé** (car TCP et IP intègre un checksul et EThernet un CRC32 !!!)



ID de transaction : Valeur positionnée par le client afin de repérer les réponses retournées par le serveur qui recopie la valeur

ID de Protocole : Valeur positionnée par le client (Modbus = 0)

Scénario d'une requête MODBUS-TCP routée sur Internet

❶ L'APDU Modbus-TCP est encapsulée dans une trame Ethernet-TCP-IP.

Trame Ethernet

6	6	2	>20	>20	n	4
@Dest	@Source	0800	Entête IP	Entête TCP	APDU Modbus TCP	CRC

La passerelle (Serveur) Modbus-TCP extrait l'APDU

Entête Modbus-TCP	Adr	Code	Data

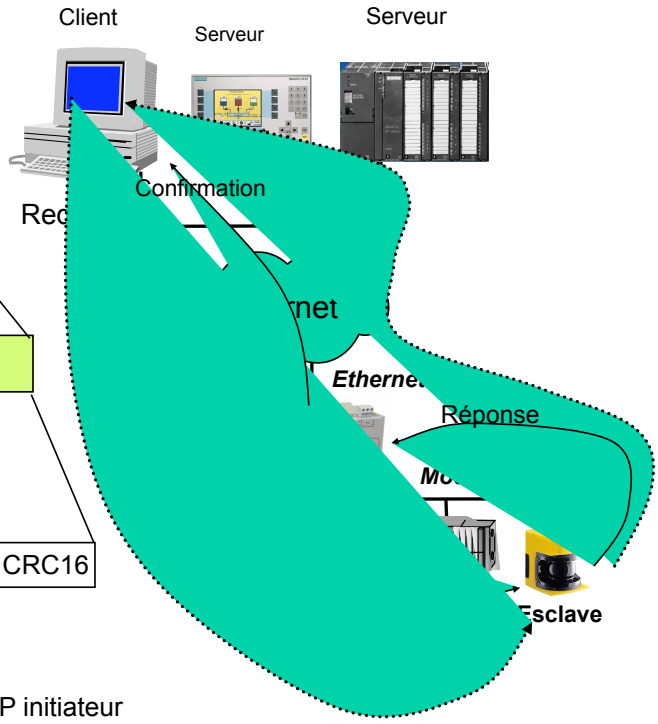
❷ La passerelle Modbus-TCP génère une question Modbus-RTU à l'esclave adressé.

Trame Modbus

Adr	Code	Data	CRC16

❸ L'esclave répond au Maître Modbus-RTU

❹ La passerelle Modbus-TCP confirme au client Modbus-TCP initiateur



Analyse de Trafic Ethernet

Une Connexion et requête Modbus TCP

Logiciel ETHEREAL (www.ethereal.com)

capture échange Modbus TCP WAgo.eth - Ethereal

File Edit View Go Capture Analyze Statistics Help

Filter: Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Info
1	0.000000	192.168.2.106	193.49.35.63	TCP	3186 > 502 [SYN] Seq=0 Ack=0 Win=25200 Len=0 MSS=1460
2	0.062895	193.49.35.63	192.168.2.106	TCP	502 > 3186 [SYN, ACK] Seq=0 Ack=1 Win=1500 Len=0 MSS=512
3	0.062938	192.168.2.106	193.49.35.63	TCP	3186 > 502 [ACK] Seq=1 Ack=1 Win=25200 Len=0
4	2.452694	192.168.2.106	193.49.35.63	Modbus	query [1 pkt(s)]: trans: 0; unit: 1, func: 6: Write single register.
5	2.515688	193.49.35.63	192.168.2.106	Modbus	response [1 pkt(s)]: trans: 0; unit: 1, func: 6: Write single register.
6	2.667830	192.168.2.106	193.49.35.63	TCP	3186 > 502 [ACK] Seq=13 Ack=13 Win=25188 Len=0
7	84.415068	192.168.2.106	193.49.35.63	Modbus	query [1 pkt(s)]: trans: 0; unit: 1, func: 3: Read multiple registers.
8	84.478560	193.49.35.63	192.168.2.106	Modbus	response [1 pkt(s)]: trans: 0; unit: 1, func: 3: Read multiple registers.
9	84.585667	192.168.2.106	193.49.35.63	TCP	3186 > 502 [ACK] Seq=25 Ack=30 Win=25171 Len=0
10	94.749318	192.168.2.106	193.49.35.63	Modbus	query [1 pkt(s)]: trans: 0; unit: 1, func: 3: Read multiple registers.
11	94.813640	193.49.35.63	192.168.2.106	Modbus	response [1 pkt(s)]: trans: 0; unit: 1, func: 3: Read multiple registers.
12	95.000656	192.168.2.106	193.49.35.63	TCP	3186 > 502 [ACK] Seq=37 Ack=289 Win=24912 Len=0
13	102.58838	192.168.2.106	193.49.35.63	TCP	3186 > 502 [FIN, ACK] Seq=37 Ack=289 Win=24912 Len=0
14	102.64770	193.49.35.63	192.168.2.106	TCP	502 > 3186 [ACK] Seq=289 Ack=38 Win=1500 Len=0
15	102.64887	193.49.35.63	192.168.2.106	TCP	502 > 3186 [FIN, ACK] Seq=289 Ack=38 Win=1500 Len=0
16	102.64890	192.168.2.106	193.49.35.63	TCP	3186 > 502 [ACK] Seq=38 Ack=290 Win=24912 Len=0

```
▷ Frame 7 (66 bytes on wire, 66 bytes captured)
▷ Ethernet II, Src: 00:80:c8:02:27:5e, Dst: 00:04:e2:7c:6f:e8
▷ Internet Protocol, Src Addr: 192.168.2.106 (192.168.2.106), Dst Addr: 193.49.35.63 (193.49.35.63)
▷ Transmission Control Protocol, Src Port: 3186 (3186), Dst Port: 502 (502), Seq: 13, Ack: 13, Len: 12
▷ Modbus/TCP
```

```
0000  00 04 e2 7c 6f e8 00 80  c8 02 27 5e 08 00 45 00   ...|o... ..'^..E.
0010  00 34 73 6b 40 00 80 06  df d5 c0 a8 02 6a c1 31   .4sk@... .....j.1
0020  23 3f 0c 72 01 f6 83 82  28 b6 bd 95 fe 0e 50 18   #?.r.... (.....P.
0030  62 64 2c 87 00 00 00 00  00 00 00 06 01 03 02 00   bd,..... ........
0040  00 04                                             ..
```

File: capture P: 16 D: 16 M: 0

Passerelle ModBus série RTU / Modbus Ethernet TCP, alimentation interne

- **Compatible Ethernet 10Base-T (802.3)**
- **Fonction serveur TCP/IP (HTTP, TFTP, SMTP, SNMP, etc...)**
- **Supporte les protocoles Modbus RTU-ASCII**
- **E/S série V24, V11 ou RS485 avec isolation optique**
- **Logiciel re-routage port COM vers Ethernet en option**

Le serveur de périphériques SP1xxxRMB ou passerelle Modbus/TCP permet la gestion de périphériques asynchrones Modbus, via un réseau Ethernet , à partir du protocole TCP/IP.
Le logiciel applicatif pilotant l'équipement asynchrone peut utiliser soit un port COM traditionnel avec un logiciel de re-routage soit un port Ethernet 10Base-T (Modbus/TCP).
Le protocole de communication ModBus, basé sur le principe Maître - Esclaves, est supporté de différentes façons par le serveur de périphérique SP1xxxRMB:
- Connexion d'équipements Modbus esclaves vers un poste maître Modbus/TCP
- Connexion d'un poste maître Modbus-RTU vers des équipements esclaves Modbus/TCP
- Connexion traditionnelle Modbus maître / esclaves via un support Ethernet TCP/IP

Le serveur de périphérique ou passerelle Modbus SP1xxxRMB supporte ces différentes configurations, qui permettent la migration progressive vers Modbus/TCP. Différents paramètres doivent être sélectionnés lors de la phase de configuration :
- Adresse "slave" unique ou multiple (pour une adresse IP)
- Time-out de réception caractères
- Time-out de réception réponse de l'eclave

Le serveur de périphériques ou passerelle Modbus SP1xxxRMB comporte une E/S série asynchrone de type V24, V11 ou RS485 avec isolation optique et un débit maximal de 115,2 Kbps avec/sans gestion de flux.
L'accès Ethernet 10 Mbps (IEEE802.3) de type 10Base-T (RJ45) permet le raccordement sur le réseau local Ethernet via un port de hub Ethernet ou par l'intermédiaire d'un convertisseur de média cuivre / fibre optique.
Le serveur de périphériques SP1 est proposé sous forme de boîtier plastique autonome ou de boîtier compact pour fixation sur rail Din.
De nombreuses applications sont envisageables grâce au serveur de périphériques, de la simple gestion de capteurs à des processus industriels, en passant par des applications de télécontrôle et de téléalarme, etc...
Le serveur de périphériques peut également supporter localement des applicatifs spécifiques sur demande.
La passerelle Modbus/TCP inclut de nombreux protocoles en plus de TCP/IP, UDP, Telnet, DHCP, SNMP, TFTP, HTTP, etc...
Les applications autorisées par ces différents services sont à prendre en compte par l'utilisateur.
GMI-DATABOX propose plusieurs types de prestations de service, du transfert de compétences à la réalisation complète de vos applications spécifiques.
De nombreux autres modèles sont disponibles pour des environnements ModBus et industriels avec :
- entrées et/ou sorties numériques (T.O.R)
- entrées et sorties analogiques, etc...

Contactez notre service commercial.

CARACTERISTIQUES

: Fonction : Convertisseur Modbus / Ethernet TCP-IP
E/S série : V24, V11 ou RS485
Mode de transmission : Asynchrone
Code de transmission : transparent
Protocole de transmission : Modbus RTU ou ASCII
Débit : jusqu'à 115,2 Kbps
Gestion de flux : XON/XOFF, CTS/DTR ou sans.
Isolation : par coupleurs optiques
Accès Ethernet :
Conforme IEEE 802.3 (CSMA/CD)
Interface : RJ45 (10 Base-T)
Adresse IP : programmable
Protocoles : TCP, IP, ICMP, ARP
Services TCP : SNMP, DHCP, TFTP, HTTP, SMTP, etc...
Dimensions :
- Boîtier plastique : 110 x 50 x 170 mm (l x h x p)
- Boîtier rail Din : 75 x 100 x 110 mm

Connecteurs :
- LAN : RJ45
- E/S de contrôle : DB9 femelle
- E/S utilisateur : DB9 mâle

Voyants :
- Lan : Activité, TD, RD et Collision
- E/S : TD, RD, ON

Alimentation : 230 VAC 50/60 Hz
Option alimentation : 24 ou 48VDC
Environnement :
- Température d'utilisation : 0° à 50°C
- Température de stockage : -10° à +70°C
- Humidité : 10 % à 90 % sans condensation.

Normes : CEM 89/336
EN60950
Fabrication : France

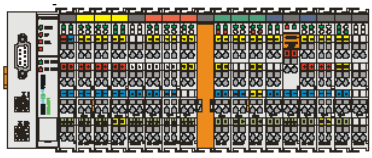
http://www.gmidatabox.fr/gmidatabox/site/produit/produit_fiche.php?id_produit=345&type=reference&let=S

Module Wago Ethernet

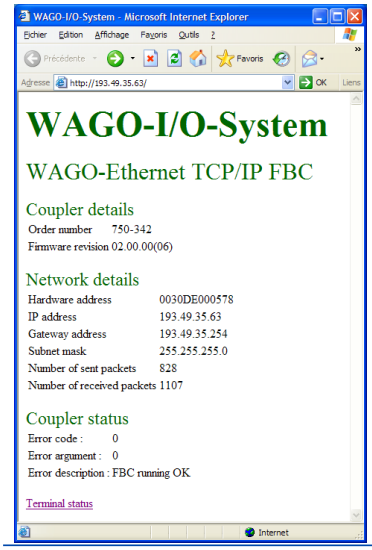
Pr. Eddy BAJIC
IUT Nancy Brabois
Nancy Université

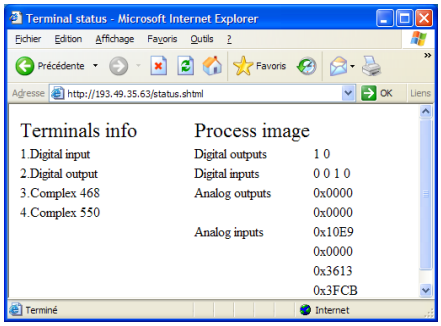
Module E/S WAGO sur Modbus TCP

- Module d'Entrées /Sorties déportées sur Ethernet
- Autoconfiguration de paramètre IP par BootP (serveur BootP Wago freeware)
- Serveur Modbus TCP (3 connexions maximum simultanées)
- Serveur Web Statique intégré

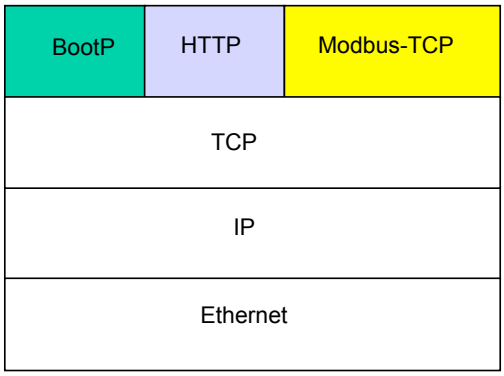


Wago 750

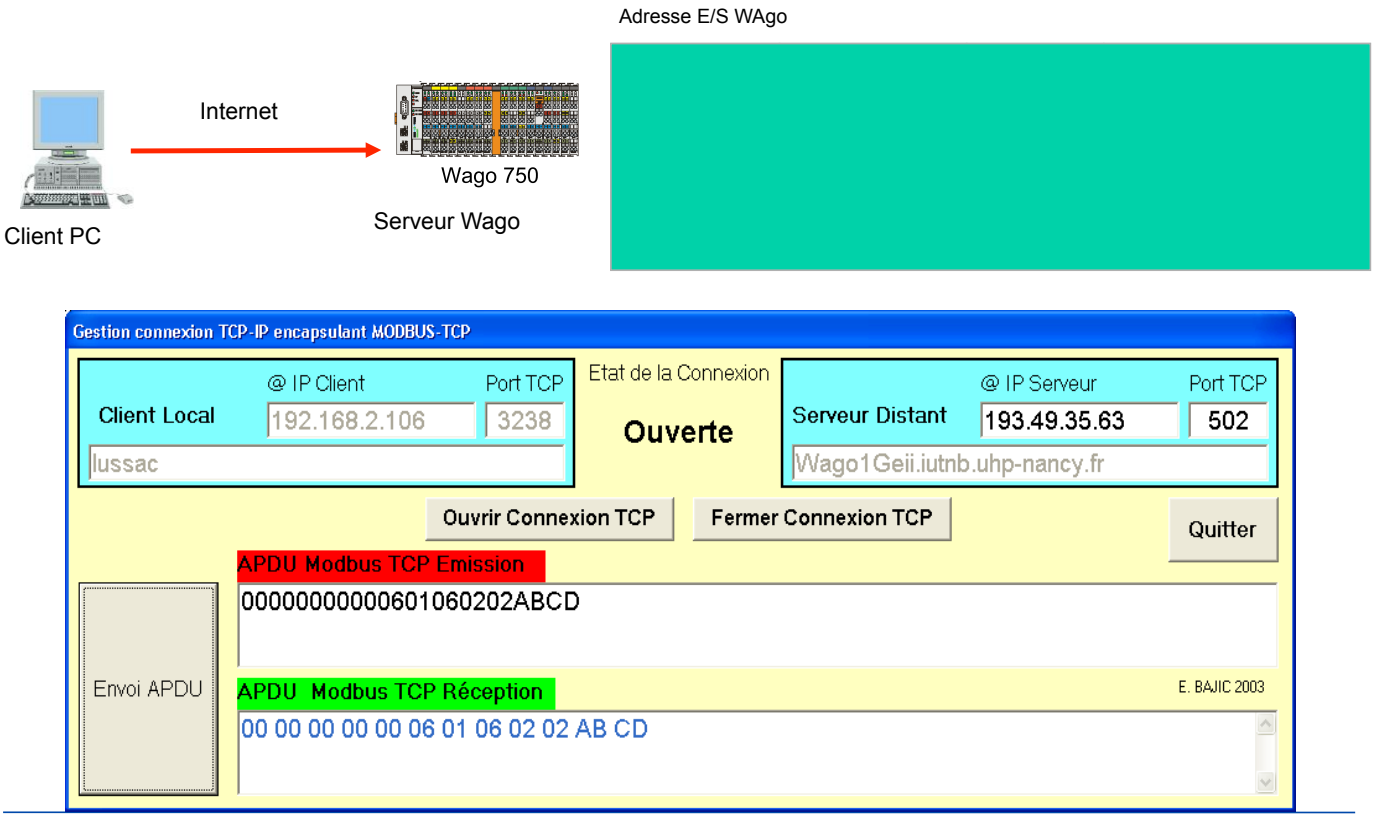




Stack TCP-IP du module WAGO 750 -342



Application VB – Modbus TCP



Programmation VB – Modbus TCP

```
Private Sub BtnClose_Click()
Winsock.Close ' puis on ferme la connection
lblConnexion.Caption = "Fermée"
btnSendData.Enabled = False
End Sub

Private Sub BtnConnect_Click()
 Winsock.RemoteHost = txtIPServeur.Text ' Paramétrage de la connexion
Winsock.RemotePort = txtPortTCP.Text
lblConnexion.Caption = "en cours..."
Winsock.connect ' puis on tente une connexion
'Attente d'acquittement d'ouverture de la cnx
Timer1.Enabled = True: timeout = 0
While (timeout <> 1)
    DoEvents ' on laisse le PC travailler ...
    If Winsock.State = sckConnected Then
        lblConnexion.Caption = "Ouverte"
        txtServeurHostName.Text = GetHostNameFromIP (txtIPServeur.Text)
        txtPortClient = Winsock.LocalPort
        txtIPClient = Winsock.LocalIP
        txtClientHostName = Winsock.LocalHostName
        Timer1.Enabled = False: timeout = 1 ' on sort de la boucle dès qu'on est connecté
        End If
Wend
' test si cnx OK
If Not Winsock.State = sckConnected Then
    lblConnexion.Caption = "Refusée"
End If
End Sub

Private Sub btnQuit_Click()
Winsock.Close ' on ferme la connexion
End
End Sub

Private Sub btnSendData_Click()
'Bouton envoi de données sur la cnx
Dim trame As String

If Not Winsock.State = sckConnected Then
    MsgBox ("Communication impossible, Pas de Connexion ouverte !")
    Else
    trame = Str2hex(txtEnvoiDonnees)
    Winsock.SendData trame
    End If
End Sub

Private Sub Form_Load()
txtIPClient = Winsock.LocalIP
txtClientHostName = Winsock.LocalHostName
End Sub

Private Sub Timer1_Timer()
timeout = 1: Timer1.Enabled = False
End Sub

Private Sub Winsock_Connect()
' Activé lorsque la connexion est effective

lblConnexion.Caption = "Ouverte"
btnSendData.Enabled = True
End Sub

Private Sub WinSock_DataArrival(ByVal bytesTotal As Long)
Dim Mess, msg As String

Winsock.GetData Mess, vbString ' on capture ce qui est arrivé sur la cnx ..
msg = Hex2str(Mess)
txtAPDUReception = msg
End Sub

Private Sub Winsock2_Error(ByVal number As Integer, Description As String, ByVal Scode As Long, ByVal Source As String, ByVal HelpFile As String, ByVal HelpContext As Long, CancelDisplay As Boolean)
End Sub
```


APPLICATIONS INDUSTRIELLES de MODBUS - TCP

Pr. Eddy BAJIC
IUT Nancy Brabois
Nancy Université

Gestion de l' Eclairage Public

Enjeux : Optimiser la gestion de l'éclairage public

- ✓Importance du budget de maintenance
- ✓Temps d'indisponibilité des installations
- ✓Gestion du bon fonctionnement des centaines, milliers de lampes
- ✓Économies d'énergie
- ✓Futures charte de l'environnement,

Infrastructures à gérer

- ✓**Lampes à décharge sur réseau électrique :**
  - Sodium Haute Pression
  - Iodure métallique
  - Ballon fluo
  - ...
- ✓**Ballast et condensateur**

Principes

- ✓Un logiciel de supervision avec géo localisation sur un PC serveur : Plug & View
- ✓ Un concentrateur CPL dans l'armoire de distribution
- ✓ Un équipement d'interface CPL par mât
- ✓ Une valise de paramétrage

Avantages

- ✓ Performance
- ✓ Modularité
- ✓ Bas coûts
- ✓ Simplicité d'installation
- ✓ Facilité à utiliser

Architecture de gestion de l'éclairage