

Le protocole

Protocole

"A protocol is a set of rules that define how communication occurs in a network"

un protocole est un ensemble de règles qui définissent comment se produit une communication dans un réseau.

Un genre de pilote

Un pilote permet au matériel de communiquer avec le système. En d'autres termes, un pilote c'est le protocole de communication entre le matériel et le système.

Sans un pilote, votre souris ne peut pas fonctionner, elle ne peut pas communiquer avec le système. Vous comprenez donc que le pilote est l'interface de communication entre le système et le matériel, il en est de même pour le protocole.

Un genre de langue

Faire communiquer 2 personnes =
- dialoguer VIA même langue
- dialoguer VIA interprète (L1 <-> L2 <-> L2)

En réseau, c'est la même chose. La langue que les humains parlent, c'est un protocole pour les hôtes dans un réseau. Pas n'importe quel protocole car il en existe plusieurs. Mais celui qui nous concerne est appelé « protocole de communication ».

Quand à l'interprète de notre exemple, dans un réseau, ce sera la passerelle (appliquée) qui permettra de faire communiquer deux réseaux basés sur des protocoles différents en assurant plusieurs fonctions telles que la traduction des protocoles et des signaux, l'isolation d'erreurs, l'adaptation d'impédances, etc.

Exemple de fonctionnement d'un protocole

Pierre veut transmettre un message à Jean

Langage humain

Langage informatique

Pierre souhaite communiquer avec Jean.
1. Pierre compose le numéro, attend que Jean décroche.
A. Numéro composé incorrect n'existe pas.
B. Numéro correct mais indisponible.
C. Numéro correct et Jean décroche et dit Allô
D. Numéro correct, Jean ne décroche pas
2. Pierre dialogue et délivre son message
3BIS Pierre parle dans le vide car Jean fait autre chose que l'écouter.
4. Au revoir Jean. Pierre raccroche.

1. Pierre envoie un paquet d'initialisation de session à Jean.
A. ICMP packet (Internet Control Message Protocol) qui enverra une erreur de type 3 (destination unreachable, destination inaccessible) et de code 7 (Destination host unknown, destination inconnu).
B. erreur de type 3 (destination unreachable, destination inaccessible).
C. Initialisation de session.
D. erreur ICMP de type 3 (destination inaccessible) et de code 1 (destination inaccessible)

2. Session en cours, communication établie

3. l'hôte Jean va envoyer un paquet à Pierre disant « je n'ai pas reçu le dernier paquet, renvoie-le stp ». Pierre va alors renvoyer le dernier paquet.

3BIS Pour prévenir ce genre de chose, Pierre peut vérifier la présence de Jean en demandant toutes les x minutes « Tu me suis ? Tu es là ? ». En réseau, avec TCP il s'agit d'une vérification périodique de l'état de la session de communication. Cela signifie qu'il faut enverra un paquet de vérification de session pour savoir si l'hôte Jean est toujours connecté. Si Jean ne répond pas après un certain laps de temps, la communication est terminée (la session se termine là).

4. Fin de session.

ICMP est un protocole dans la suite protocolaire TCP/IP utilisé pour envoyer des messages d'erreur entre deux hôtes dans un réseau. Il travaille en partenariat avec le protocole IP.
protocole TCP
l'hôte Pierre demande un message d'accusé de réception à chaque envoi de paquet l'hôte Jean devra répondre « oui j'ai reçu le paquet » tout le long de la communication
protocole UDP
l'hôte Pierre ne demande pas un message d'accusé de réception à chaque envoi de paquet, mais délivre plus vite le message

Attention : la vérification périodique et l'accusé de réception ne sont pas des paquets mais des informations qui sont dans l'en-tête des paquets.

Exigences d'un protocole : c'est à dire ses fonctions essentielles

Il faudrait que ledit protocole soit en mesure d'assurer des fonctions vitales au bon déroulement d'une communication. Il existe plusieurs « fonctions vitales » (comprendre exigences) qu'un protocole de communication doit être capable de remplir.

Gestion du format de données

Une donnée = ENTETE + CONTENU

package

Gestion du format d'adresses

Dans une lettre on met EMETTEUR + DESTINATAIRE > sur l'enveloppe EGALLEMENT

Correspondance d'adresses

Adresse écrite sur Enveloppe = ADRESSE LOGIQUE mais adresse physique = maison de la personne.

Le Facteur fait une correspondance (ADDRESS MAPPING) entre :
- adresse LOGIQUE IP
- adresse physique MAC

Detection des erreurs de transmission

Détecter les erreurs via un CRC (Cyclic Redundancy Check) ajouté à la fin des paquets

Accusé de réception

informer un hôte émetteur qu'il a reçu le paquet envoyé pour empêcher ce dernier de renvoyer les mêmes choses

Gestion de perte d'informations

un paquet met trop de temps à arriver à son destinataire, "il se perd"

Si l'hôte-récepteur B répond dans un intervalle de x secondes à l'hôte-émetteur A, ce dernier saura alors que B a bien reçu les données, et n'essaiera plus de les renvoyer

Direction du flux d'information

FULL DUPLEX : A et B peuvent communiquer en même temps
HALF DUPLEX : A puis B puis A puis B...

Contrôle de séquences

SEGMENTATION de l'information en séquences, découpées, numérotées
CONGESTION du réseau et des routes
ENVOI en désordre des données pour accélérer le transfert
NUMÉROTATION = suppression des doublons et remise en ordre des morceaux

Gestion de flux

Gestion de la VOLATILITÉ = donner à celui qui "écoute" le temps de comprendre ce qui est dit, puisqu'il se peut que l'émetteur parle plus vite que le récepteur.

LE DEBIT de parole

Attention : TCP/IP n'est pas un protocole

TCP/IP est un protocole stack : une suite protocolaire, c'est l'association de eux protocoles distincts :

- le protocole TCP

+ le protocole IP

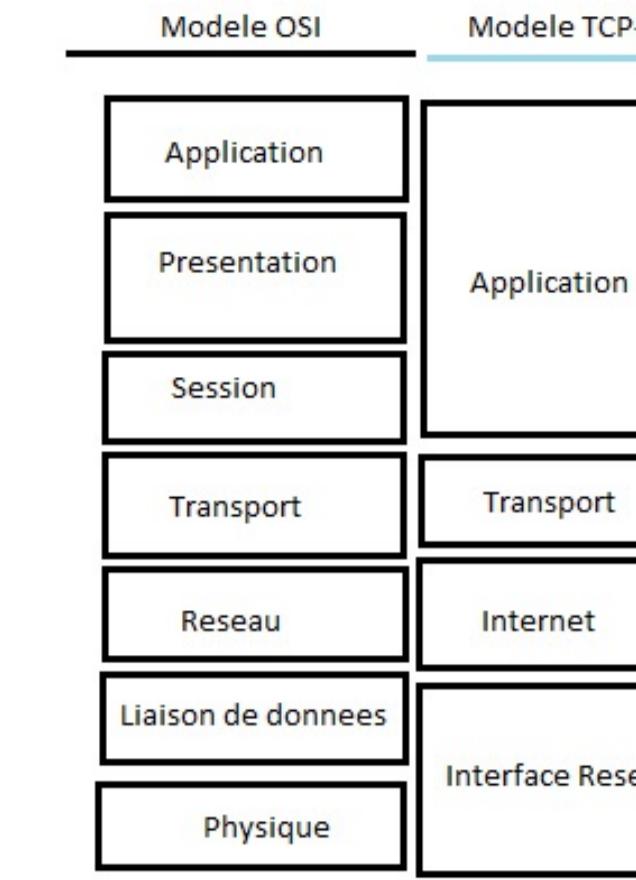
cet ensemble de protocoles fonctionnant en harmonie et cohésion pour le bon déroulement de la communication.

Objectif Normatif → Modéliser la communication entre 2 hôtes

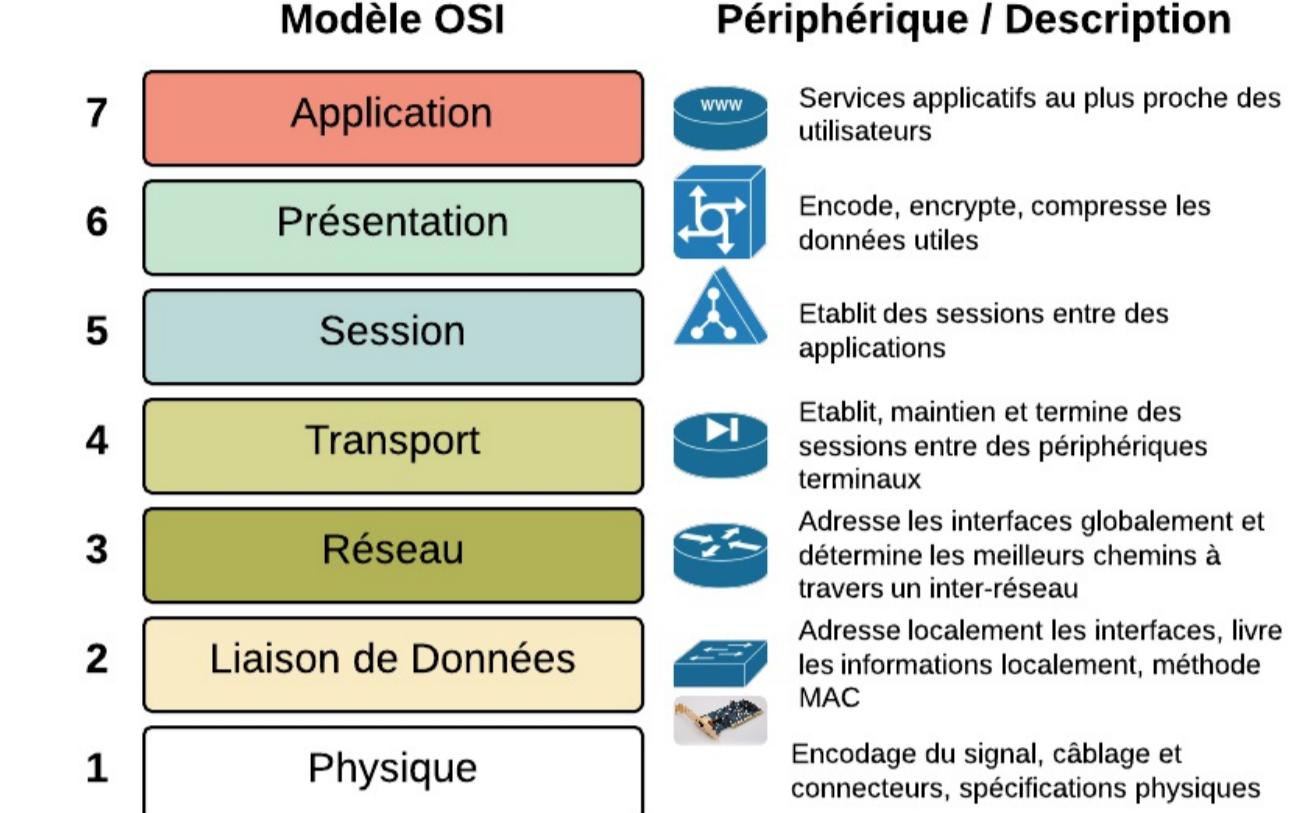
Objectif Descriptif → Le modèle TCP/IP

il décrit la façon dont se passe la communication entre deux hôtes. En d'autres termes, si vous voulez comprendre comment se déroule la communication « sur le terrain », prenez le modèle TCP/IP.

RFC1122

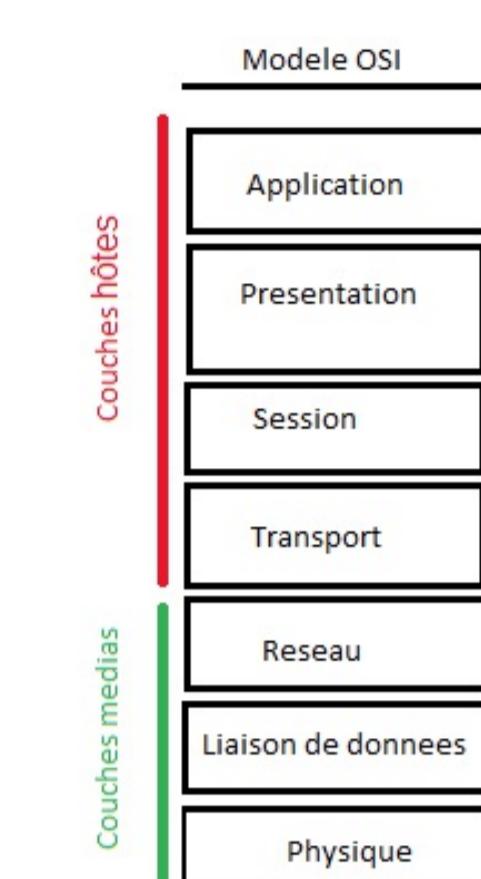


7 COUCHES : Pour Le Réseau Tout Se Passe Automatiquement (PLRTSPA)



Les couches application, présentation, session et transport sont dites « couches hôtes » (host layers en anglais) car « concernent » directement les hôtes.

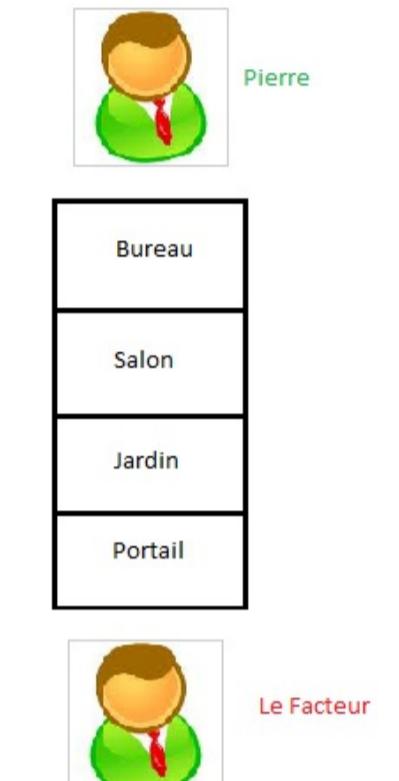
Les couches réseau, liaison et physique, elles, sont des couches de médias (media layers) : elles sont plus liées au média qu'à l'hôte.



Comprendre le modèle par l'exemple du facteur

Pierre garde une lettre dans son bureau. Il veut la donner au facteur, qui devra la porter dans la boulangerie. La lettre est destinée à Jacques, mais Pierre n'a pas le droit d'entrer dans le bureau de Jacques. Jacques non plus n'a pas le droit de sortir de son bureau. Si le facteur peut entrer dans le bureau de Jacques pour livrer la lettre, mais il lui est interdit d'aller dans celui de Pierre pour la chercher.

La maison de Pierre est mal construite : il n'y a pas de couloir, juste un alignement vertical de pièces séparées par une porte. Pour aller du bureau au portail, Pierre doit traverser le salon et le jardin. Schématiquement, cela donne ceci :



Les unités de données

Chaque information transmise est une DONNÉE. La DONNÉE prend un nom différent à chaque COUCHE mais ça reste la MEME DONNÉE.

On peut la trouver sous le nom PDU pour Protocol Data Unit

Si on utilise la terminologie en PDU alors le nom sera précédé de l'unité de la couche dont la donnée est issue.

Dans la couche applicative -> APDU (application data.)

Dans la couche présentation -> SPDU

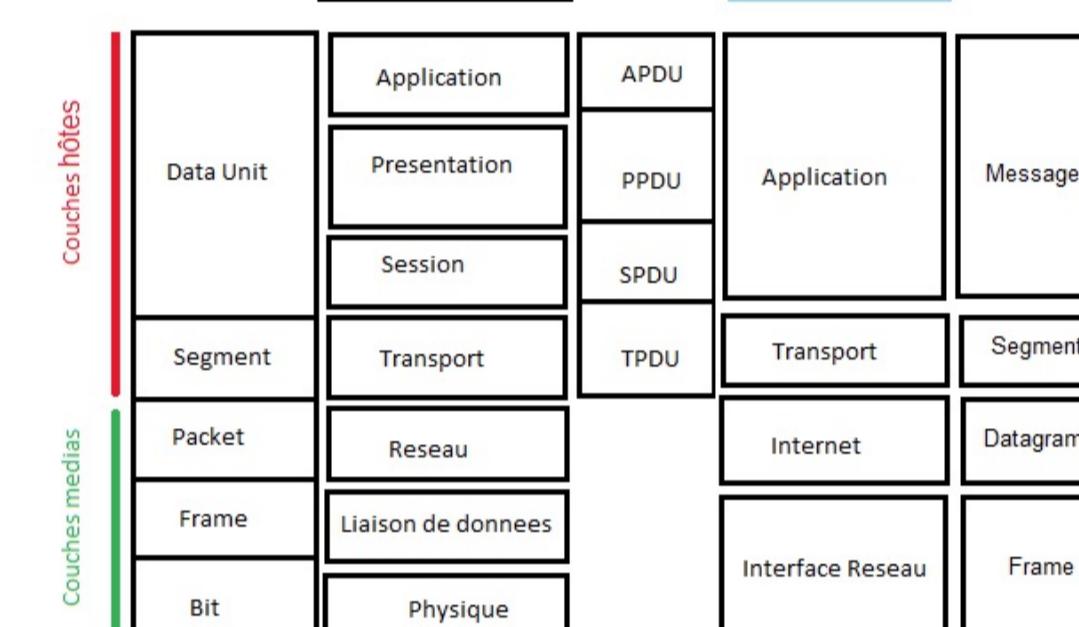
Dans la couche session -> SPDU

Dans la couche transport -> TPDU

Pas de terminologie PDU pour les couches médias. A la place

PACKET FRAME BIT pour OSI

DATAGRAM FRAME pour TCP/IP



Les couches du modèle OSI

Couche APPLICATIVE

- interface logicielle, etc qui donne accès aux services réseaux.

les protocoles assurant ces services :

FTP (transfert de fichiers), Telnet (Sessions à distance), SMTP (envoi d'un mail), HTTP (visualisation page web)

Couche PRESENTATION

- emballage de l'information pour la présenter correctement, son aspect visuel

SEMANTIQUE, SYNTAXE, CRYPTAGE / DECRYPTAGE

ex : conversion d'un fichier codé en EBCDIC vers ASCII

ATTENTION le cryptage peut s'effectuer dans d'autre couches pour sécuriser les données.

Couche SESSION

- ouvrir le "bout" le "port" -> établir une session entre les applications pour qu'elles communiquent

OUVERTURE, GESTION, CLÔTURE, RECONNEXION

Couche TRANSPORT

- gère les contraintes de communication, la meilleure façon d'envoyer l'information.

- divise les données en séquences

utilisation du protocole TCP le plus souvent dans cette couche

ATTENTION le transport de données est en fait réalisé par les 4 dernières couches PLRT

Couche RÉSEAU

- routage des données du point A au point B (liaison logique)

adresse

- ajout dans l'en-tête adresse IP source et adresse IP destinataire

- choix du mode de transport connecté ou non connecté (TCP ou UDP)

Couche LIASION

- établissement d'une liaison physique entre les 2 hôtes

- transforme la couche physique en une liaison

- corrige éventuellement des erreurs de transmission

- fragmente les données en plusieurs trames envoyées une par une dans le réseau local.

Protocoles de cette couche (physique) ex : Ethernet, IEEE802.11, PPP, HDSL...

Couche PHYSIQUE

- réception des trames de la couche de liaison de données

- convertit les trames en BITS mis sur le média pour envoi

- transmission des signaux électriques ou optiques entre les hôtes en communication

Services ex : détection de collisions, multiplexing, modulation, circuit switching...

Encapsulation des données

Emballer les données à chaque couche du modèle OSI.

Quand vous écrivez une lettre (ou un mail), vous devez la glisser dans une enveloppe. C'est à peu près le même principe dans le modèle OSI :

les données sont emballées à chaque couche et le nom de l'unité de données n'est rien d'autre que le nom de l'enveloppe. Nous avons donc une sous-partie précédente que, dans la couche application, nous avons vu dans la sous-partie PDU.

Ensuite, nous avons vu que dans une couche réseau, l'unité de données était le paquet. Ces PDU forment une sorte d'enveloppe qui contient deux choses : la donnée en elle-même et l'en-tête spécifique à cette couche.

La partie « donnée » du paquet est composée de la donnée initiale, suivie des en-têtes des couches qui la précèdent. Il existe une toute petite formule mathématique définissant la relation entre les couches.

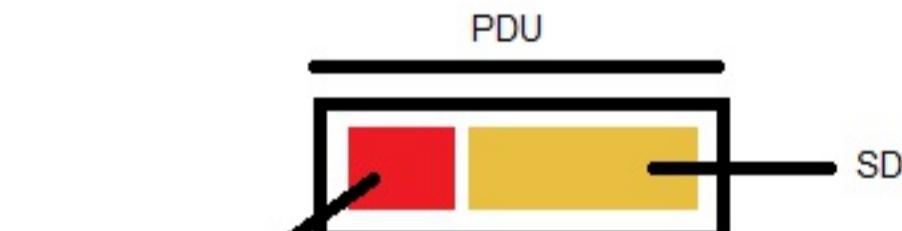


La DONNÉE que C+1 transmet à C s'appelle SDU (Standard Data Unit) tant qu'elle n'est pas encapsulée par C.

Dès que C a encapsulé sa SDU cela devient une PDU.

Le PDU de C est également à son tour encapsulé par C.

PDU C = (SDU C+1) + ENTEXTE ET/OU PIED C



L'encapsulation par analogie

Quand vous écrivez une (vraie) lettre, c'est un SDU. Vous la mettez dans une enveloppe sur laquelle est écrite une adresse. Cette lettre qui n'était qu'un SDU devient un PDU qu'elle a été enveloppée (encapsulée). Votre lettre arrive à la poste. Un agent du service postal la prend et la place dans un sac à lettres. Mais on la va plus, puisqu'elle est dans un sac. Pour l'instant, la lettre, l'enveloppe et le sac forment un SDU. L'agent du service postal va alors inscrire le code postal du destinataire sur le sac en question, qui devient alors un PDU. Si vous avez d'autres lettres partant pour la même ville, elles seront mises dans un seul sac. C'est un SDU. Tout comme on a ajouté des informations sur l'enveloppe et sur le sac, il faut également mettre un code postal sur la caisse. Cet ajout fait de cette caisse un PDU.