

Modèle en couches

Table des matières

1. Introduction.....	2
2. Services avec connexion et sans connexion.....	3
3. Modèles de référence.....	4
3.1. Le modèle de référence OSI/OSI.....	4
3.2. Le modèle de référence TCP/IP.....	8
4. Comparaison des modèles OSI/OSI et TCP/IP.....	11
4.1. Critique du modèle de référence OSI/OSI.....	11
4.2. Critique du modèle de référence TCP/IP.....	12
5. Protocoles du modèle OSI/OSI.....	13

Lors de la conception des premiers réseaux d'ordinateurs, les ingénieurs se sont surtout concentrés sur l'aspect matériel, reléguant l'aspect logiciel au second plan. Cette stratégie ne tient plus. Les logiciels de réseaux sont aujourd'hui hautement structurés.



1. Introduction

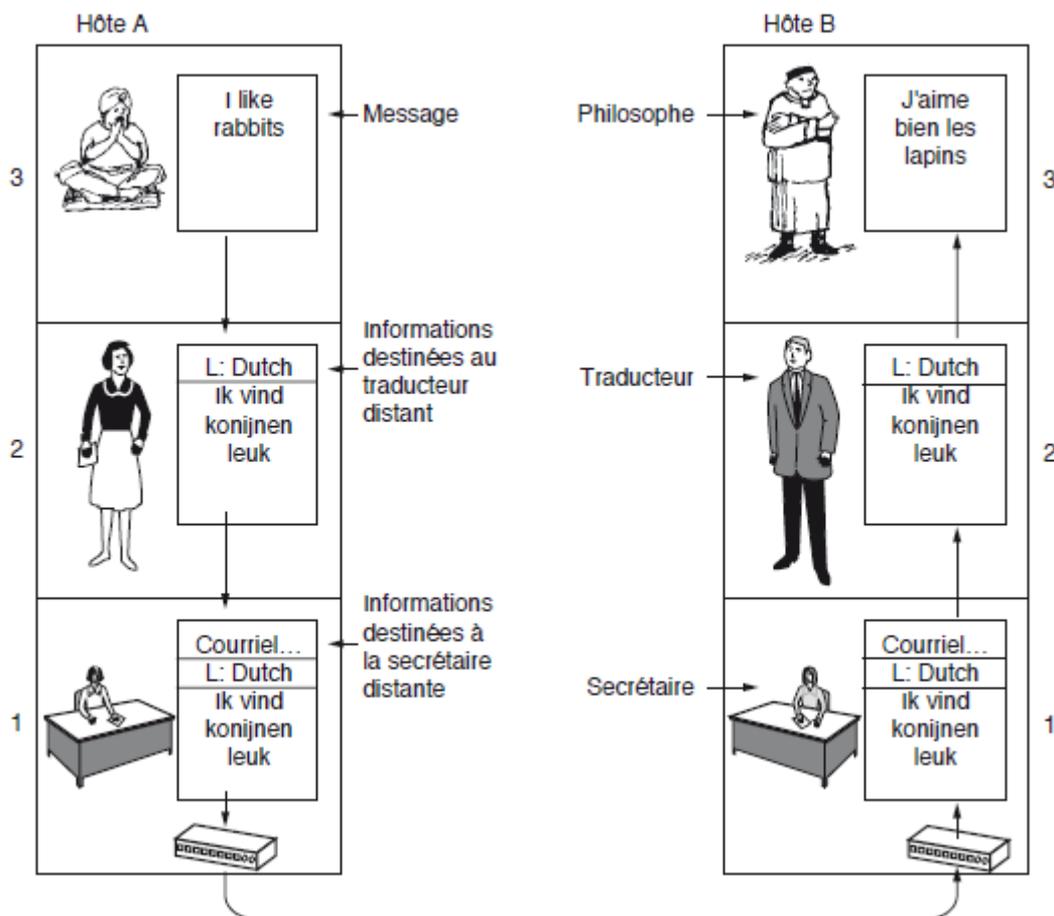
Afin de réduire la complexité de conception, la plupart des réseaux sont organisés en strates, appelées couches ou niveaux, chacune étant placée au-dessus de la précédente. Le nombre de couches ainsi que le nom, le contenu et la fonction de chacune d'elles diffèrent selon les réseaux. Le rôle de chaque couche est de fournir des services à la couche immédiatement supérieure tout en lui dissimulant les détails d'implémentation. En un sens, on peut dire que chaque niveau représente une boîte noire qui offre certains services au niveau supérieur.

Lorsque la couche n d'une machine dialogue avec la couche n d'une autre machine, les règles et les conventions qui gouvernent cette communication sont regroupées sous le nom de protocole de couche n . En substance, un protocole est une convention acceptée par les parties communicantes sur la façon dont leur dialogue doit avoir lieu.

Un ensemble de couches et de protocoles forme une architecture de réseau. La spécification d'une architecture doit contenir suffisamment d'informations pour qu'un développeur puisse écrire le programme (ou construire le matériel) pour chaque couche de façon que celui-ci obéisse correctement au protocole concerné. Il n'est même pas utile que les interfaces soient les mêmes sur toutes les machines d'un réseau, du moment que chaque machine peut utiliser correctement tous les protocoles. L'ensemble des protocoles utilisés par un système donné, avec un protocole par couche, s'appelle une pile de protocoles.

Pour mieux comprendre ce que représente la communication en couches, imaginez la situation suivante. Deux philosophes (processus de couche 3) souhaitent s'entretenir, mais l'un parle ourdou et anglais, et l'autre chinois et français. Faute de pouvoir partager une langue commune, ils engagent chacun un traducteur (processus de couche 2), qui fait appel à son tour à une secrétaire (processus de couche 1). Le premier philosophe transmet un message (en anglais) à son traducteur par l'intermédiaire de l'interface 2/3, en indiquant « I like rabbits ». Comme les traducteurs se sont accordés sur une langue commune, en l'occurrence le néerlandais, le message est converti en « Ik vind konijnen leuk ».

Le traducteur passe le message à sa secrétaire pour transmission, par exemple, par courrier électronique (protocole de couche 1). Quand le message parvient à l'autre secrétaire, il est transmis au traducteur local, qui le traduit en français et le transmet par l'intermédiaire de l'interface 2/3 au deuxième philosophe. Notez que les protocoles sont complètement indépendants les uns des autres tant que les interfaces ne sont pas changées. Les traducteurs pourraient passer au finnois, pourvu qu'ils soient tous les deux d'accord et qu'aucun ne change ses interfaces avec les couches 1 et 3. De même, les secrétaires pourraient choisir le téléphone à la place du courrier électronique, sans déranger, ni même aviser, les autres couches. Chaque processus peut ajouter certaines informations à l'intention du processus pair uniquement, et celles-ci ne seront pas transmises à la couche supérieure.



Le but d'un système en couche est de séparer le problème en différentes parties (les couches) selon leur niveau d'abstraction.

Chaque couche du modèle communique avec une couche adjacente (celle du dessus ou celle du dessous). Chaque couche utilise ainsi les services des couches inférieures et en fournit à celle du niveau supérieur.

Entre deux appareils reliés, les couches doivent être les mêmes et pouvoir communiquer avec le même protocole.

2. Services avec connexion et sans connexion

Les couches peuvent offrir deux types de services différents aux couches supérieures : avec connexion et sans connexion. Cette section examine ces services ainsi que leurs différences.

Un service avec **connexion** suit le modèle du système téléphonique. Pour converser avec quelqu'un, vous décrochez le téléphone, composez le numéro, parlez, puis raccrochez. De même, pour utiliser un service réseau avec connexion, l'utilisateur doit d'abord établir une connexion, l'utiliser, puis la libérer. L'aspect essentiel d'une connexion est qu'elle agit tel un tuyau : l'émetteur y injecte des objets (des bits) à une extrémité et le récepteur les récupère à l'autre bout. Dans la plupart des cas, l'ordre d'envoi des bits est préservé.

À l'inverse, un service **sans connexion** suit le modèle du système postal. Chaque message (lettre) contient l'adresse de destination complète, et chacun est routé via les nœuds intermédiaires dans le système indépendamment de tous les messages suivants. Les messages portent des noms différents

selon le contexte : un paquet est un message au niveau de la couche réseau. Lorsque les nœuds intermédiaires reçoivent un message en entier avant de le transmettre au prochain nœud, on parle de commutation en mode différé (store-and-forward). Normalement, lorsque deux messages sont envoyés à une même destination, le premier envoyé sera le premier à arriver. Toutefois, des retards peuvent se produire et faire en sorte que le second arrive avant.

Chaque service peut encore être caractérisé par sa **fiabilité**. Certains services sont qualifiés de fiables au sens où ils ne perdent jamais de données. En principe, un service fiable est implémenté au moyen d'un système de notification dans lequel le récepteur **accuse réception** de chaque message afin que l'émetteur soit certain qu'il est arrivé. Le processus d'acquiescement provoque une surcharge de service et des délais, qui valent souvent la peine qu'on les supporte mais sont parfois indésirables.

Dans certaines applications, les délais introduits par les acquiescements sont inacceptables. L'une d'elles est le transport de la voix numérisée, pour la voix sur IP (VoIP). Il est moins gênant pour les usagers d'entendre du bruit sur la ligne de temps à autre que de supporter un retard occasionné par l'attente des acquiescements. Dans le même ordre d'idées, quelques pixels défectueux ne posent pas de problèmes lors d'une conférence vidéo, alors qu'il serait irritant d'avoir des images saccadées en raison des arrêts de transmission liés à la correction d'erreurs.

Un service non fiable sans connexion (c'est-à-dire non acquiescé) est souvent appelé service de datagramme, par analogie avec le service télégraphique qui n'envoie pas non plus d'accusé de réception à l'émetteur.

Dans d'autres situations, il est souhaitable pour des questions pratiques d'éviter d'établir une connexion pour envoyer un message, mais la fiabilité est essentielle. Le service de datagramme acquiescé peut alors être utilisé. Il s'apparente à l'envoi d'une lettre recommandée avec accusé de réception.

	Services	Exemples
Avec connexion	Flot de messages fiable	Suite de pages
	Flot d'octets fiable	Téléchargement d'un film
	Connexion non fiable	Voix sur IP
Sans connexion	Datagramme non fiable	Prospectus électronique
	Datagramme acquiescé	Messagerie avec accusé de réception
	Demande-réponse	Interrogation d'une base de données

3. Modèles de référence

Bien que les protocoles associés au modèle OSI soient rarement utilisés aujourd'hui, le modèle en lui-même est tout à fait général et reste valide : les fonctionnalités examinées au niveau de chaque couche sont toujours très importantes. Avec le modèle TCP/IP, c'est l'inverse qui se produit : le modèle n'est pas très employé alors que ses protocoles sont largement déployés.

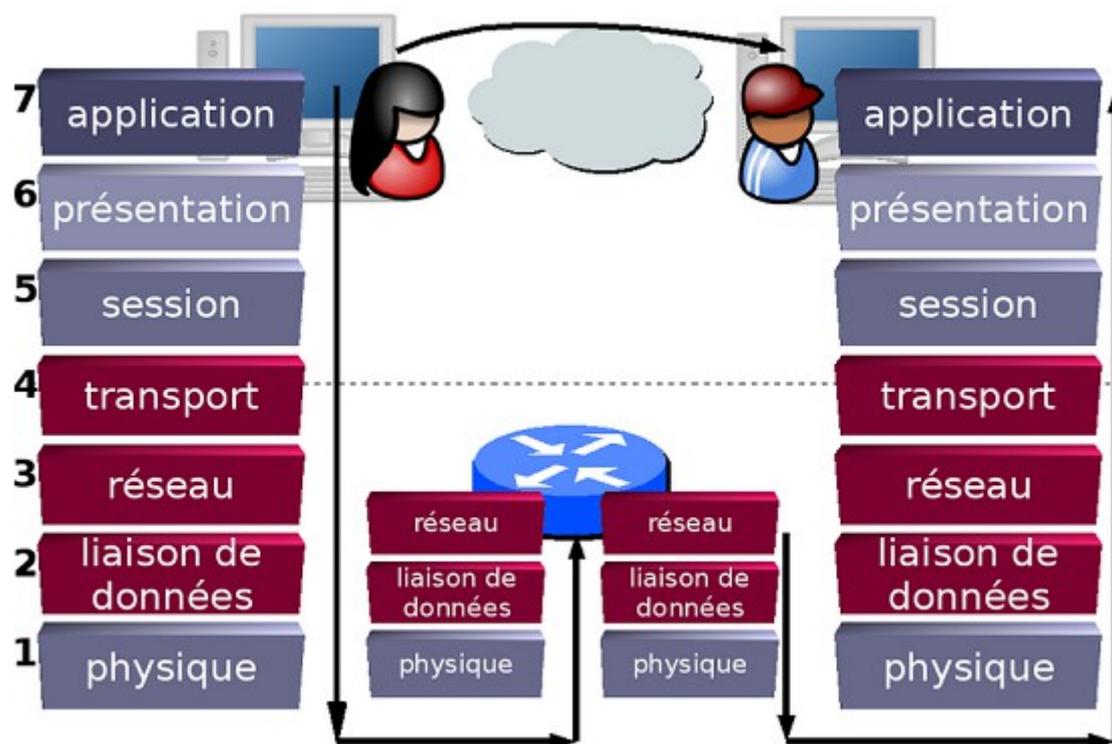
3.1. Le modèle de référence OSI/OSI

Le modèle OSI (Open Systems Interconnection) s'appuie sur une proposition qui a été développée par l'ISO (Organisation internationale de normalisation) comme une première étape vers la

normalisation internationale des protocoles utilisés dans les diverses couches. Il a été révisé en 1995. On l'appelle modèle de référence OSI car il traite des systèmes ouverts, c'est-à-dire des systèmes ouverts à la communication avec d'autres systèmes.

Les données transmises sont appelées unité de données (data unit en anglais). On les nomme parfois **PDU** (Protocol Data Unit : « unité de données de protocole ») ; dans ce cas, leur nom sera précédé de l'initiale de la couche dont ces données sont issues. Par exemple dans la couche applicative, elles prennent le nom d'APDU (Application Protocol Data Unit : « unité de données de protocole d'application »). Dans la couche de session, elles s'appelleront donc... SPDU (Session Protocol Data Unit : « unité de données de protocole de session »)...

Ce modèle se compose de sept couches :



1. La couche physique

La couche physique se charge de la **transmission de bits à l'état brut** sur un canal de communication. L'un des objectifs de conception de ce niveau est de s'assurer qu'un bit à 1 envoyé sur une extrémité arrive aussi à 1 de l'autre côté et non à 0. Les questions que l'on se pose portent généralement sur les signaux électriques à utiliser pour représenter un 1 et un 0, le nombre de nanosecondes que doit durer un bit, la possibilité de transmission bidirectionnelle simultanée, la façon dont une connexion initiale est établie puis libérée lorsque les deux extrémités ont fini, ou encore le nombre de broches d'un connecteur et le rôle de chacune. Les problèmes de conception concernent principalement les interfaces mécaniques et électriques et la synchronisation, ainsi que le support physique de transmission sous-jacent à la couche physique.

2. La couche liaison de données

Le rôle principal de la couche liaison de données est de faire en sorte qu'un moyen de communication brut apparaisse à la couche réseau comme une liaison exempte d'erreurs de

transmission non détectées. Pour ce faire, elle masque les erreurs réelles afin que la couche réseau ne les voie pas en implantant des codes de détection d'erreurs.. S'il s'agit d'un **service fiable**, le récepteur confirme la bonne réception de chaque trame en envoyant à l'émetteur une trame d'acquiescement.

Un autre point important à résoudre dans cette couche, mais aussi dans la plupart des couches supérieures, est **d'éviter la saturation** d'un récepteur lent par les données d'un émetteur rapide. Recourir à des mécanismes de contrôle du flux afin d'éviter l'engorgement de données.

3. La couche réseau

La couche réseau contrôle le fonctionnement du sous-réseau. C'est dans la couche réseau que les ordinateurs connectés au réseau sont identifiés et qu'est déterminé la façon dont les paquets sont routés de la source vers la destination. Les routes peuvent être choisies au moyen de tables statiques « câblées » dans le réseau et rarement modifiées ou, plus souvent, mises à jour automatiquement pour éviter les composants défectueux. Elles peuvent également être déterminées au début de chaque conversation, par exemple, lors d'une connexion à partir d'un terminal (comme une ouverture de session sur une machine distante). Enfin, elles peuvent aussi être très dynamiques et déterminées à nouveau pour chaque paquet, afin de prendre en compte la charge actuelle du réseau.

Lorsqu'un paquet doit passer d'un réseau à un autre pour atteindre sa destination, de nombreux problèmes peuvent surgir : la technique d'adressage du second réseau peut être différente de celle du premier, le second réseau peut refuser le paquet s'il est trop grand, les protocoles peuvent être différents, etc. La couche réseau doit gérer tous ces problèmes pour que des réseaux hétérogènes puissent être interconnectés.

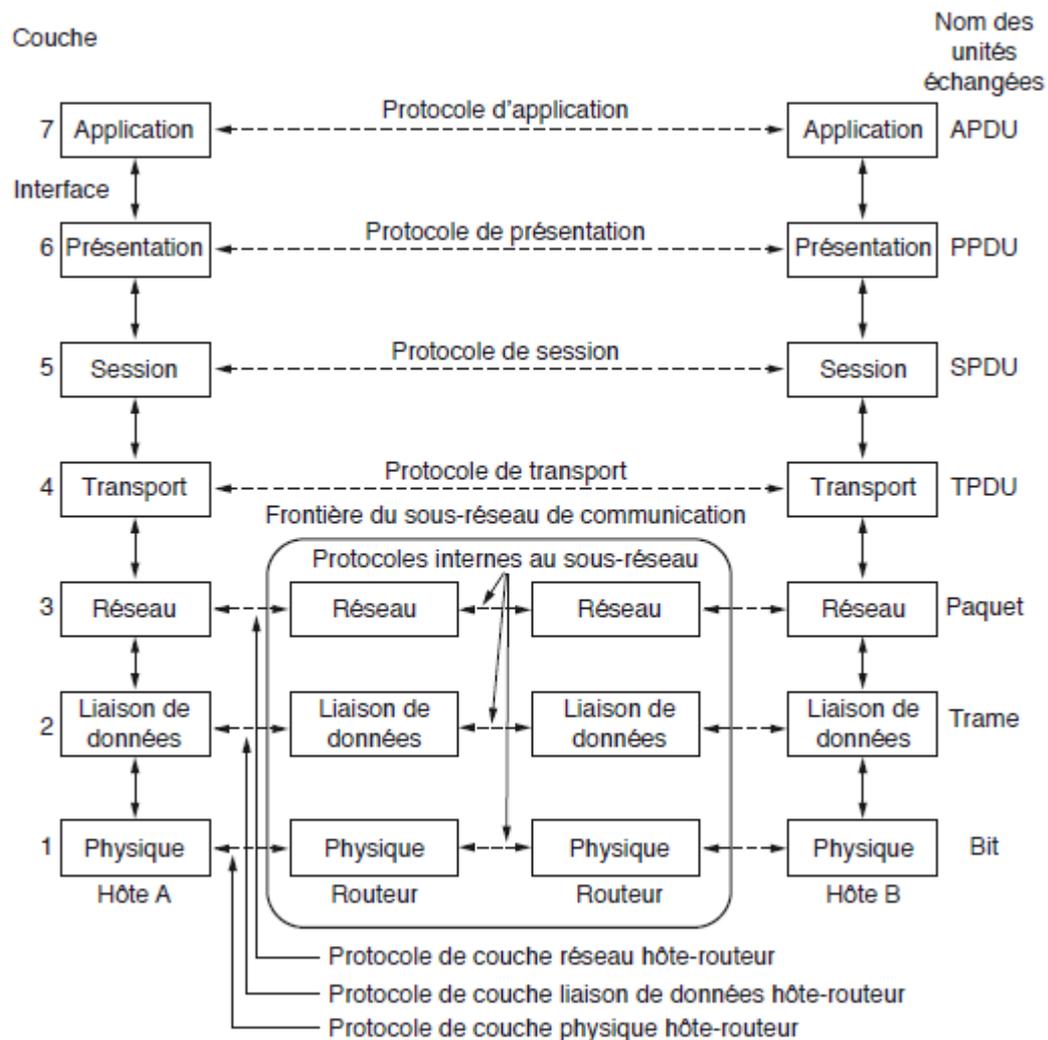
4. La couche transport

La fonction de base de la couche transport est d'accepter des données de la couche supérieure, de les diviser en unités plus petites si nécessaire, de les transmettre à la couche réseau et de s'assurer qu'elles arrivent correctement à l'autre bout.

La couche transport est une vraie couche de bout-en-bout : elle transporte les données tout du long, de la source à la destination. Autrement dit, un programme sur la machine source entretient une conversation avec un programme similaire sur la machine de destination, en utilisant les en-têtes des messages et des messages de contrôle. Dans les couches plus basses, chaque protocole prend place entre une machine et ses voisins immédiats, non entre les machines source et de destination, qui peuvent être séparées par de nombreux routeurs.

C'est dans la couche transport que sont corrigées les erreurs de transmission et qu'est vérifié le bon acheminement des données. Elle optimise l'utilisation de la couche réseau et assure des travaux de type fragmentation de message. Si une connexion est interrompue, elle va décider de réinitialiser la connexion et de reprendre le transfert des données.

La différence entre les couches 1 à 3, qui sont chaînées, et les couches 4 à 7, qui sont de bout-en-bout, est illustrée ci-dessous :



5. La couche session

La couche session permet aux utilisateurs de différentes machines d'établir des sessions entre eux. Une session offre divers services, parmi lesquels la gestion du dialogue (suivi du tour de transmission), la gestion du jeton (empêchant deux participants de tenter la même opération critique au même moment) et la synchronisation (gestion de points de reprise permettant aux longues transmissions de reprendre là où elles en étaient, à la suite d'une interruption).

C'est au niveau de la couche session que sont ouverts les ports de communication.

6. La couche présentation

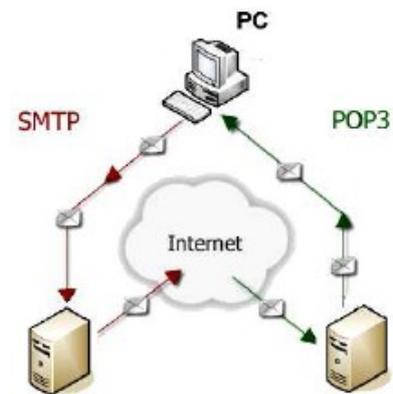
À la différence des couches les plus basses, qui sont principalement concernées par le transport des bits, la couche présentation s'intéresse à la syntaxe et à la sémantique des informations transmises. Pour permettre à des ordinateurs utilisant des représentations de données internes différentes de communiquer.

C'est dans cette couche que sont converties les informations d'un format à un autre (ASCII par exemple) afin d'assurer l'indépendance entre l'utilisateur et le transport.

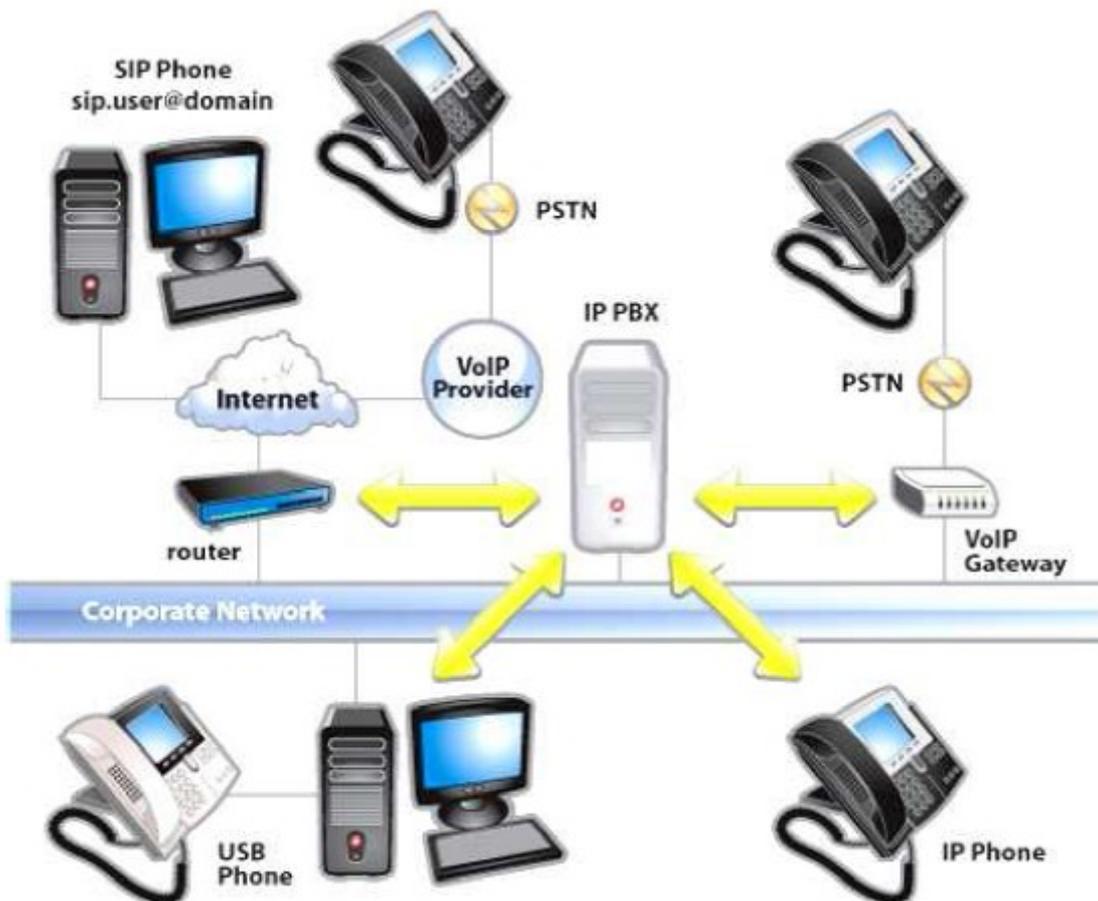
Les opérations de conversion, de cryptage et de compression y sont réalisées.

7. La couche application

La couche application contient différents protocoles dont les utilisateurs ont couramment besoin. HTTP (HyperText Transfer Protocol), qui forme la base du World Wide Web, est un protocole d'application largement utilisé. Lorsqu'un navigateur veut afficher une page web, il transmet son nom au serveur qui l'héberge au moyen du protocole HTTP, et le serveur envoie la page en réponse. D'autres protocoles d'application sont utilisés pour le transfert de fichiers, le courrier électronique et les nouvelles (news).



Remarque : L'ensemble des trois couches application, présentation et session assure la collecte des données au niveau de l'utilisateur et leur mise en forme afin d'assurer leur transmission à l'application de destination.



3.2. Le modèle de référence TCP/IP

Venons-en maintenant au modèle de référence utilisé par l'ancêtre des réseaux étendus, l'ARPAnet, et par son célèbre descendant, l'Internet mondial. À l'origine, il s'agissait d'un réseau destiné à la recherche, subventionné par le ministère de la Défense des États-Unis (DoD, Department of Defense). Il a fini par connecter des centaines de sites universitaires et gouvernementaux au moyen de liaisons téléphoniques louées. Lorsque les réseaux satellitaires et radio ont fait leur apparition, on a dû développer une nouvelle architecture de référence pour permettre aux protocoles existants

d'interagir avec eux.

Ainsi, dès le tout début, la possibilité d'interconnecter de nombreux réseaux de façon transparente a été l'un des objectifs de conception majeurs de l'ARPAnet. Cette architecture est devenue plus tard le modèle de référence TCP/IP, nommé d'après ses deux principaux protocoles.

Craignant que certains de ses précieux hôtes, routeurs et passerelles d'interconnexion ne soient soudainement neutralisés par une attaque de l'Union soviétique, le DoD avait un autre objectif majeur : faire en sorte que le réseau survive à la perte d'un équipement de sous-réseau, sans que les conversations existantes soient interrompues.

1. La couche réseau

Toutes ces exigences ont conduit au choix d'un réseau à commutation de paquets fondé sur une couche sans connexion transversale aux différents réseaux. La couche la plus basse de ce modèle, la couche liaison, décrit ce que les liens comme les lignes séries et les connexions Ethernet classiques doivent faire pour répondre aux besoins de cette interconnexion sans connexion. Il ne s'agit pas du tout d'une couche au sens normal du terme, mais plutôt d'une interface entre les hôtes et les liens de transmission.

Elle prend en charge les notions suivantes :

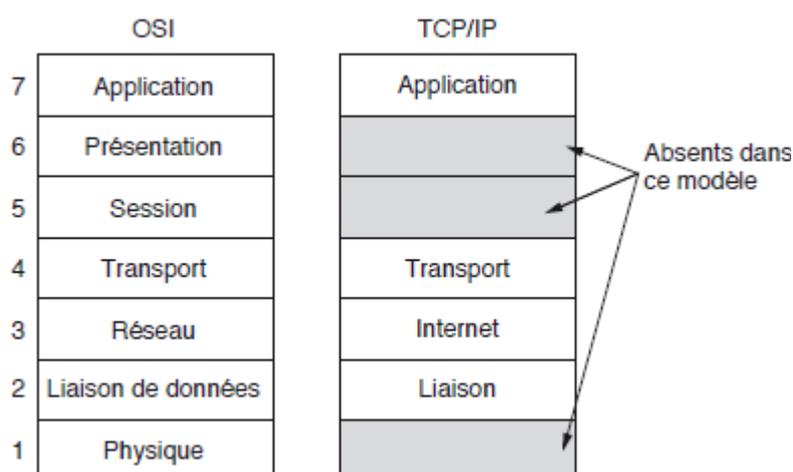
- acheminement des données sur la liaison,
- synchronisation de la transmission de données,
- format des données,
- conversion des signaux (analogique/numérique),
- contrôle des erreurs à l'arrivée.

Toutes ces spécifications sont transparentes aux yeux de l'utilisateur, car l'ensemble de ces tâches est en fait réalisé par le système d'exploitation, ainsi que par les drivers du matériel permettant la connexion au réseau (ex : driver de carte réseau).

2. La couche internet

La couche internet est l'axe central qui soutient toute l'architecture. La figure ci-contre la représente comme correspondant approximativement à la couche réseau du modèle OSI. Elle permet aux hôtes d'introduire des paquets sur n'importe quel réseau et fait en sorte qu'ils soient acheminés indépendamment les uns des autres vers leur destination (pouvant se

trouver sur un réseau différent de celui d'origine). Il est même possible que les paquets arrivent dans un ordre complètement différent, auquel cas les couches supérieures se chargeront de les réordonner si cela fait partie des exigences de livraison. Notez que le terme « internet » est employé ici au sens générique d'interréseau, même si cette couche est



présente dans l'Internet.

La couche internet définit un format de paquet officiel et un protocole nommé IP (Internet Protocol), plus un protocole compagnon, ICMP (Internet Control Message Protocol), qui l'aide à fonctionner. Son rôle est d'acheminer les paquets IP jusqu'à leur destination. Les deux préoccupations majeures à ce niveau sont donc le routage et les congestions (bien qu'IP ne se soit pas montré très efficace pour les éviter).

3. La couche transport

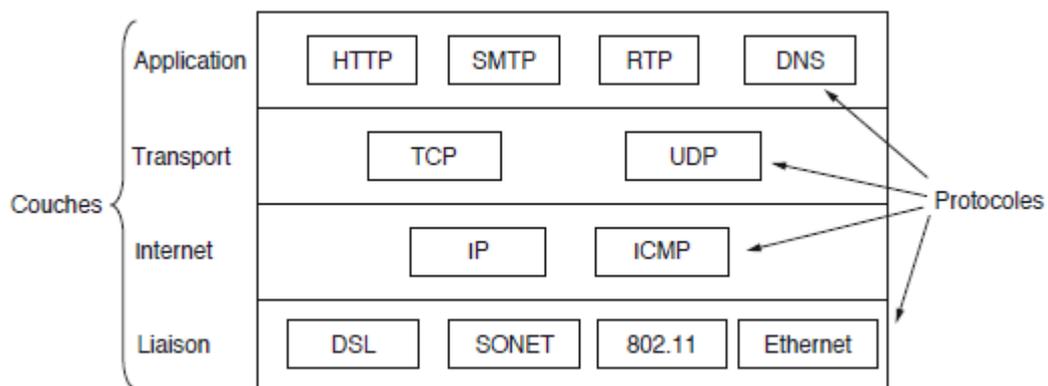
La couche directement supérieure à la couche internet dans le modèle TCP/IP se nomme aujourd'hui la couche transport. À l'instar de la couche transport du modèle OSI, son rôle est de permettre à des entités paires sur les hôtes source et de destination de mener une conversation.

Le problème consiste à identifier ces applications. En effet, suivant la machine et son système d'exploitation, l'application pourra être un programme, une tâche, un processus... C'est la raison pour laquelle un système de numéro a été mis en place afin de pouvoir associer un type d'application à un type de données, ces identifiants sont appelés : **ports**.

La couche transport contient 2 protocoles permettant d'échanger des données indépendamment du type de réseau emprunté (c'est-à-dire des couches inférieures) :

Le premier, **TCP** (Transmission Control Protocol), est un protocole **fiable avec connexion** qui garantit la livraison sans erreur à n'importe quel hôte de l'interréseau d'un flot d'octets émis par une machine. Il segmente le flot d'octets entrant en messages discrets et transmet chacun d'eux à la couche internet. À l'arrivée, le processus TCP destinataire réassemble les messages reçus en un flot de sortie. TCP assure aussi un contrôle de flux pour éviter qu'un émetteur rapide ne submerge un récepteur lent de plus de messages qu'il ne peut en traiter.

Le second, **UDP** (User Datagram Protocol), est un protocole **non fiable sans connexion** qui permet aux applications d'assurer elles-mêmes le séquençage et le contrôle de flux au lieu de faire appel à TCP. Ce protocole est aussi largement utilisé par les interrogations ponctuelles de type demande-réponse dans les environnements client-serveur, et par les applications pour lesquelles la livraison des données à temps est plus importante que leur précision, comme dans la transmission de sons ou d'images. Les relations entre IP, TCP et UDP sont illustrées ci-dessous :



4. La couche application

Le modèle TCP/IP ne comprend ni couche session, ni couche présentation, leur nécessité n'ayant pas été perçue. À la place, les applications incluent simplement les fonctions

correspondantes si elles ont besoin. L'expérience du modèle OSI a confirmé la pertinence de ce choix : ces couches sont très peu utiles à la plupart des applications.

Directement au-dessus de la couche transport, on trouve la couche application, qui contient tous les protocoles de haut niveau. Les premiers à avoir été développés sont Telnet (protocole de terminal virtuel), FTP (protocole de transfert de fichiers) et SMTP (protocole d'échange de courrier électronique). Beaucoup d'autres protocoles, ont été ajoutés au fil des ans. Citons notamment DNS (Domain Name System), pour associer des noms d'hôtes à leurs adresses réseau, HTTP (HyperText Transfer Protocol), pour transférer des pages web et RTP (Real-Time Transfer Protocol) pour diffuser en temps réel des médias comme la voix et la vidéo,

4. Comparaison des modèles OSI/OSI et TCP/IP

Les modèles de référence OSI et TCP/IP ont beaucoup de points communs. Tout d'abord, ils reposent tous deux sur le concept de pile de protocoles indépendants. Ensuite, leurs couches sont à peu près identiques sur le plan fonctionnel. Par exemple, les couches inférieures, couche transport incluse, ont pour rôle d'offrir un service de transport de bout-en-bout, indépendant du type de réseau, aux processus qui souhaitent communiquer. De même, les couches au-dessus de la couche transport sont des couches orientées applications utilisatrices du service de transport.

Trois concepts sont au cœur du modèle OSI :

1. Les services.
2. Les interfaces.
3. Les protocoles.

La plus grande contribution du modèle OSI a probablement été de rendre explicite la distinction entre ces trois concepts. Chaque couche rend des **services** à la couche située au-dessus d'elle. La définition du service indique ce que fait une couche, mais pas comment elle le fait ni comment les entités supérieures y accèdent. Le service spécifie la sémantique de la couche.

L'interface d'une couche indique aux processus situés au-dessus d'elle comment accéder à cette couche. Elle spécifie les paramètres à utiliser et les résultats à attendre. Elle non plus ne dit rien du fonctionnement interne de la couche.

Enfin, les **protocoles** pairs utilisés dans une couche ne regardent qu'elle. Elle peut utiliser celui qu'elle veut du moment qu'il remplit son rôle (qu'il fournit les services prévus). Elle peut également changer de protocole sans que cela affecte le fonctionnement des couches supérieures.

4.1. Critique du modèle de référence OSI/OSI

À l'époque où la deuxième édition de ce livre a été publiée (1989), il était clair pour de nombreux experts du domaine que le modèle OSI et ses protocoles allaient s'imposer et tout balayer sur leur passage. Cela ne s'est pas produit pour différentes raisons majeures :

1. moment inopportun : Les protocoles de la pile TCP/IP étaient déjà largement déployés au sein des universités de recherche lorsque ceux du modèle OSI sont apparus. Si la vague d'investissements n'avait pas encore eu lieu, le marché universitaire était suffisamment important pour que certains constructeurs commencent prudemment à proposer des produits TCP/IP. Lorsque le modèle OSI s'est présenté, ils ne voulaient pas prévoir de support pour une seconde pile de protocoles tant qu'ils n'y étaient pas forcés. Il n'y a donc pas eu d'offre

initiale. Les entreprises ont pratiqué l'attentisme, aucune ne voulant prendre le risque d'être la première, et OSI a fait long feu.

2. mauvaise technologie : Le choix des sept couches a été plus politique que technique, et deux d'entre elles – session et présentation – sont pratiquement vides, alors que deux autres – liaison de données et réseau – sont au contraire trop pleines.

Un autre problème avec OSI est que certaines fonctions, telles que l'adressage, le contrôle de flux et le contrôle d'erreurs, réapparaissent dans chaque couche. Des études ont montré que, par exemple, pour être efficace, le contrôle d'erreurs doit être réalisé dans la couche la plus haute, si bien que le réitérer à chaque niveau est souvent inutile et inefficace.

3. mauvaises implémentations : Étant donné l'énorme complexité du modèle et des protocoles, il n'est pas surprenant que les implémentations initiales aient été volumineuses, lourdes à manipuler et lentes. Tous ceux qui s'y sont frottés s'en sont rendu compte à leurs dépens. Dans ce contexte, il n'a pas fallu longtemps pour que OSI devienne synonyme de piètre qualité.

Par contraste, l'une des premières implémentations de TCP/IP, qui faisait partie de l'UNIX de Berkeley, était tout à fait réussie (et gratuite de surcroît). Elle a rapidement fait de nombreux adeptes, ce qui a conduit à des améliorations, puis à l'élargissement de la communauté d'utilisateurs, qui a entraîné elle-même de nouvelles améliorations.

4. mauvaise stratégie : Suite à la première implémentation, nombreux sont ceux, en particulier dans le milieu universitaire, à avoir pensé que TCP/IP faisait partie d'UNIX. Et le milieu universitaire des années 1980 voyait UNIX comme le nec plus ultra. D'un autre côté, OSI a été vu comme le produit des administrations européennes des télécommunications, puis de la Communauté européenne, et plus tard du gouvernement des États-Unis.

4.2. Critique du modèle de référence TCP/IP

Le modèle TCP/IP et ses protocoles ont aussi leurs problèmes. Premièrement, le modèle ne différencie pas clairement les concepts de services, d'interfaces et de protocoles. Les bonnes pratiques du génie logiciel ont montré qu'il était essentiel de séparer la spécification de l'implémentation, ce que le modèle OSI fait soigneusement, mais que le modèle TCP/IP ne fait pas. Par conséquent, on peut difficilement s'appuyer sur ce dernier pour concevoir de nouveaux réseaux mettant en œuvre de nouvelles technologies.

Deuxièmement, ce modèle n'étant pas du tout général, il ne permet pas réellement de décrire des piles de protocoles autres que TCP/IP. Tenter de s'en servir pour décrire Bluetooth, par exemple, est totalement impossible.

Troisièmement, la couche liaison n'est pas du tout une couche au sens où on l'entend habituellement dans le contexte des protocoles en couches, mais une interface (entre les couches réseau et liaison de données). Cette distinction entre une interface et une couche est essentielle, mais le modèle TCP/IP n'en tient pas compte.

Quatrièmement, le modèle TCP/IP ne distingue (ou n'évoque même) pas les couches physique et liaison de données qui sont pourtant complètement différentes. La couche physique est concernée par les caractéristiques de transmission du câble en cuivre, de la fibre optique et des canaux radio, tandis que la couche liaison de données est chargée de déterminer le début et la fin des trames et de les transmettre entre deux extrémités avec le degré de fiabilité souhaité. Un modèle correct devrait inclure ces deux couches en les séparant nettement, chose que ne fait pas le modèle TCP/IP.

Enfin, alors que les protocoles IP et TCP ont été soigneusement conçus et bien implémentés, le développement des autres protocoles de la suite a été improvisé et généralement confié à des doctorants. Leurs implémentations étant distribuées gratuitement, ces protocoles se sont largement imposés au point qu'il est encore difficile aujourd'hui de les remplacer. À titre d'exemple, le protocole de terminal virtuel, Telnet, a été conçu pour un terminal Télétype mécanique à 10 caractères par seconde et ignore donc tout des interfaces graphiques et de la souris. Malgré cela, il est toujours largement utilisé trente ans plus tard.

5. Protocoles du modèle OSI/OSI

couche	protocoles
7. Application	BGP · DHCP · DNS · FTP · Gopher · H.323 · HTTP · IMAP · IRC · NFS · NNTP · POP3 · RTSP · SILC · SIMPLE · SIP · SMTP · SNMP · SMB-CIFS · SSH · TCAP · Telnet · TFTP · VoIP · XMPP · WebDAV
6. Présentation	AFP · ASCII · ASN.1 · MIME · NCP · SSP · TDI · TLV · Unicode · UUCP · Vidéotex · XDR
5. Session	AppleTalk · DTLS · H.323 · RSerPool · SOCKS · TLS
4. Transport	DCCP · RSVP · RTP · SCTP · SPX · TCP · UDP
3. Réseau	ARP · Babel · BOOTP · CLNP · ICMP · IGMP · IPv4 · IPv6 · IPX · IS-IS · NetBEUI · OSPF · RARP · RIP · X.25
2. Liaison	Anneau à jeton · Token Bus · ARINC 429 · ATM · AFDX · Bitnet · CAN · Ethernet · FDDI · Frame Relay · HDLC · I ² C · IEEE 802.3ad (LACP) · IEEE 802.1aq (SPB) · LLC · LocalTalk · MIL-STD-1553 · PPP · STP · Wi-Fi · X.21
1. Physique	4B5B · ADSL · BHDn · Bluetooth · Câble coaxial · Codage bipolaire · CSMA/CA · CSMA/CD · DSSS · E-carrier · EIA-232 · EIA-422 · EIA-449 · EIA-485 · FHSS · IEEE 1394 · HomeRF · IrDA · ISDN · Manchester · Manchester différentiel · Miller · MLT-3 · NRZ · NRZI · NRZM · Paire torsadée · PDH · SDH · SDSL · SONET · T-carrier · USB · VDSL · V.21-V.23 · V.42-V.90 · Wireless USB · 10BASE-T · 10BASE2 · 10BASE5 · 100BASE-TX · 1000BASE-T