

24.1.2	Attributs communs . . . . .	406
24.1.3	Attributs de contrôle . . . . .	408
24.1.4	Attributs associés aux options des sockets . . . . .	410
24.1.5	Fonctions membre . . . . .	411
24.2	Allocation et libération . . . . .	411
24.2.1	Allocation . . . . .	411
24.2.2	Gestion du compteur de référence . . . . .	413
24.2.3	Libération . . . . .	414
24.3	Gestion des données associées . . . . .	416
24.3.1	Initialisation des données . . . . .	416
24.3.2	Verrouillage . . . . .	418
24.3.3	Initialisation du délai . . . . .	419
<b>25</b>	<b>Réception des datagrammes UDP</b>	<b>421</b>
25.1	Implémentation Linux générale d'UDP . . . . .	421
25.1.1	Implémentation Linux de l'en-tête UDP . . . . .	421
25.1.2	Calcul de la somme de contrôle . . . . .	422
25.2	Réception des datagrammes UDP . . . . .	428
25.2.1	Fonction de traitement principale . . . . .	428
25.2.2	Vérification rapide de la somme de contrôle . . . . .	431
25.2.3	Consultation de la table de hachage UDP . . . . .	433
25.2.4	Traitement du datagramme UDP . . . . .	436
25.2.5	Mise dans la file d'attente de réception UDP . . . . .	438
<b>26</b>	<b>Installation de la famille de protocoles IPv4</b>	<b>441</b>
26.1	Manipulation des familles de protocoles . . . . .	442
26.1.1	Familles de protocoles . . . . .	442
26.1.2	Les types de communication . . . . .	444
26.1.3	Les protocoles de la suite TCP/IPv4 . . . . .	449
26.2	Implémentation . . . . .	449
26.2.1	Tableau des familles de protocoles . . . . .	449
26.2.2	Enregistrement d'un type de communication IPv4 . . . . .	452
26.2.3	Installation de la suite de protocoles TCP/IPv4 . . . . .	453
<b>27</b>	<b>Implémentation des fichiers de type socket</b>	<b>459</b>
27.1	Implémentation générale des fichiers . . . . .	460
27.1.1	Système de fichiers virtuel . . . . .	460
27.1.2	Super-bloc . . . . .	461
27.1.3	Nœud d'information . . . . .	463
27.1.4	Descripteur de fichier . . . . .	465
27.1.5	Répertoire . . . . .	466
27.1.6	Types de fichiers . . . . .	466
27.1.7	Déclaration d'un système de fichiers . . . . .	467
27.1.8	Enregistrement d'un système de fichiers . . . . .	467
27.2	Descripteur de socket . . . . .	468
27.2.1	Définition du type . . . . .	468
27.2.2	Les états d'une socket . . . . .	469
27.2.3	Les drapeaux . . . . .	469
27.2.4	Le type opérations sur une socket . . . . .	469
27.3	Déclaration du système de fichiers des sockets . . . . .	470

27.3.1	La structure . . . . .	470
27.3.2	Enregistrement . . . . .	470
27.4	Système de fichiers de sockets . . . . .	471
27.4.1	Obtention du super-bloc . . . . .	471
27.4.2	Libération d'un super-bloc . . . . .	474
27.5	Opérations sur les super-blocs . . . . .	475
27.5.1	Ensemble des opérations . . . . .	475
27.5.2	Statut du système de fichiers . . . . .	475
27.5.3	Allocation d'un descripteur de nœud d'information . . . . .	476
27.5.4	Libération d'un descripteur de nœud d'information . . . . .	477
27.6	Opérations sur les répertoires de socket . . . . .	477
27.7	Opérations sur les fichiers de type socket . . . . .	477
27.8	Premières implémentations de fonctions . . . . .	479
27.8.1	Implémentation du positionnement . . . . .	479
27.8.2	Implémentation de la scrutation . . . . .	479
27.8.3	Implémentation du non mappage en mémoire vive . . . . .	480
27.8.4	Implémentation de l'ouverture de fichier . . . . .	481
27.8.5	Implémentation des événements asynchrones . . . . .	481
27.8.6	Implémentation de la libération de fichier . . . . .	482
<b>28</b>	<b>Création et fermeture des sockets</b>	<b>485</b>
28.1	Création d'une socket . . . . .	486
28.1.1	Traitement général . . . . .	486
28.1.2	Cas de la famille de protocoles IPv4 . . . . .	495
28.2	Semi-arrêt d'une socket . . . . .	500
28.2.1	Traitement général . . . . .	500
28.2.2	Cas de IPv4 . . . . .	502
28.2.3	Fonction de déconnexion spécifique à UDP . . . . .	503
28.3	Fermeture d'une socket . . . . .	506
<b>29</b>	<b>Initialisation d'un serveur</b>	<b>509</b>
29.1	Partie générale . . . . .	510
29.1.1	Fonction d'appel . . . . .	510
29.1.2	Passage espace utilisateur/espace noyau . . . . .	510
29.2	Cas de IPv4 . . . . .	511
29.2.1	Fonction spécifique d'initialisation du serveur . . . . .	511
29.2.2	Détermination du type d'adresse . . . . .	513
29.3	Obtention et vérification du port dans le cas UDP . . . . .	514
<b>30</b>	<b>Réception d'un datagramme par une socket</b>	<b>519</b>
30.1	Types liés aux fonctions d'entrée-sortie . . . . .	520
30.1.1	Vecteurs d'entrée-sortie . . . . .	520
30.1.2	En-tête de message . . . . .	520
30.1.3	En-tête de contrôle . . . . .	520
30.2	Implémentation générale de la réception des datagrammes . . . . .	521
30.2.1	Fonction d'appel . . . . .	521
30.2.2	Réception de message de socket . . . . .	522
30.2.3	Passage espace noyau/espace utilisateur . . . . .	523
30.2.4	Fonction interne de réception d'un message . . . . .	524
30.3	Cas de IPv4 . . . . .	525

30.4	Cas de UDP . . . . .	526
30.5	Lien avec les tampons de socket . . . . .	529
30.5.1	Instantiation de descripteur de tampon en réception . . . . .	529
30.5.2	Copie d'un datagramme dans l'espace utilisateur . . . . .	533
30.5.3	Libération d'un tampon de datagramme . . . . .	537
<b>V</b>	<b>Envoi</b>	<b>539</b>
<b>31</b>	<b>Envoi d'un datagramme par une socket</b>	<b>541</b>
31.1	Niveau implémentation des sockets . . . . .	542
31.1.1	Fonction d'appel . . . . .	542
31.1.2	Envoi de message de socket . . . . .	543
31.1.3	Fonction interne d'envoi de message . . . . .	543
31.2	Cas de IPv4 . . . . .	544
31.2.1	Appel à la fonction spécifique à la couche de transport . . . . .	544
31.2.2	Attribution automatique de numéro de port . . . . .	545
<b>32</b>	<b>Envoi de datagrammes ordinaires sous UDP</b>	<b>547</b>
32.1	Fonction d'envoi spécifique à UDP . . . . .	548
32.1.1	Description . . . . .	548
32.1.2	Implémentation . . . . .	548
32.2	Regroupement de données UDP . . . . .	555
32.2.1	Fonction principale . . . . .	555
32.2.2	Instantiation d'un tampon de socket pour l'émission . . . . .	561
32.2.3	Récupération des fragments . . . . .	564
32.3	Constitution du datagramme . . . . .	566
32.4	Envoi du datagramme à la couche réseau . . . . .	569
<b>33</b>	<b>Envoi de paquets ordinaires sous IPv4</b>	<b>571</b>
33.1	Première étape: routage d'un paquet ordinaire . . . . .	572
33.1.1	Recherche dans le cache de routage . . . . .	572
33.1.2	Recherche étendue à la FIB . . . . .	573
33.2	Seconde étape: constitution du paquet IP . . . . .	581
33.2.1	Implémentation des options IP . . . . .	581
33.2.2	Récupération des fragments . . . . .	581
33.3	Transfert de la couche réseau à la couche inférieure . . . . .	585
33.3.1	Première sous-étape: détermination de la fonction de transfert . . . . .	585
33.3.2	Deuxième sous-étape: fragmentation éventuelle . . . . .	586
33.3.3	Troisième sous-étape: positionnement du type du paquet . . . . .	586
33.3.4	Quatrième sous-étape: passage à la couche inférieure . . . . .	586
<b>34</b>	<b>Envoi de trames</b>	<b>589</b>
34.1	Transmission à la carte réseau: partie générale . . . . .	590
34.1.1	Première étape: mise en file d'attente . . . . .	590
34.1.2	Deuxième étape: récupération des paquets . . . . .	596
34.1.3	Troisième étape: envoi . . . . .	601
34.2	Partie spécifique à la carte . . . . .	603
34.2.1	Étude générale . . . . .	603
34.2.2	Cas de la carte 3Com 501: Mise en file d'attente . . . . .	604

34.3	Réaction à l'envoi . . . . .	607
34.3.1	Réponse du gestionnaire d'interruption . . . . .	607
34.3.2	Action lorsque le délai est écoulé . . . . .	609
<b>35</b>	<b>Désactivation et retrait</b>	<b>611</b>
35.1	Désactivation . . . . .	612
35.1.1	Cas général . . . . .	612
35.1.2	Cas de la carte 3Com 501 . . . . .	614
35.2	Retrait d'un périphérique réseau . . . . .	615
35.2.1	Démontage du système de fichier . . . . .	619
35.2.2	Démontage du système de fichiers . . . . .	619
35.2.3	Attente des références . . . . .	619
<b>VI</b>	<b>Complément sur IP : la fragmentation</b>	<b>621</b>
<b>36</b>	<b>Fragmentation pour la couche réseau : le cas de IPv4</b>	<b>623</b>
36.1	Fragmentation . . . . .	624
36.2	Réassemblage . . . . .	632
36.2.1	Cache de fragment . . . . .	632
36.2.2	Traitement du réassemblage . . . . .	634
36.2.3	Traitement lors de l'expiration du délai . . . . .	643
36.3	Initialisation de la fragmentation . . . . .	645
<b>VII</b>	<b>Appendices</b>	<b>647</b>
	<b>Bibliographie</b>	<b>649</b>
	<b>Index</b>	<b>654</b>

# Table des figures

1	Hiérarchie entre les chapitres . . . . .	vi
1.1	Graphe complet . . . . .	3
1.2	Réseau avec commutateur central . . . . .	4
1.3	Réseau distribué . . . . .	6
1.4	Pile réseau . . . . .	8
2.1	Niveaux logiques . . . . .	23
2.2	Synchronisation . . . . .	23
2.3	Codage Manchester . . . . .	24
2.4	Réseau étendu . . . . .	26
2.5	Réseau machine à machine . . . . .	28
2.6	Réseau en bus . . . . .	29
2.7	Réseau en anneau . . . . .	29
2.8	Réseau en étoile . . . . .	30
2.9	Croissance d'Arpanet : (a) décembre 1969 (b) juillet 1970 (c) mars 1971 (d) avril 1972 (e) septembre 1972 . . . . .	31
2.10	Épine dorsale NSFNET en 1988 . . . . .	32
2.11	Architecture du premier réseau Ethernet . . . . .	33
3.1	Structure d'une trame . . . . .	36
3.2	Décomptage des octets . . . . .	37
3.3	Fanion de signalisation . . . . .	38
3.4	Confirmation et infirmation . . . . .	39
3.5	Fenêtre glissante . . . . .	41
4.1	Types de câblages Ethernet . . . . .	46
4.2	Types de connexions Ethernet . . . . .	47
4.3	Sous-couches MAC et LLC . . . . .	49
5.1	Numéros de port bien connus des serveurs . . . . .	63
7.1	Données ancillaires . . . . .	89
9.1	Fragmentation d'un paquet IP . . . . .	105
9.2	Table de hachage . . . . .	106
9.3	Pseudo en-tête UDP . . . . .	112
12.1	Tampon de socket et son descripteur . . . . .	150
12.2	File d'attente de paquets . . . . .	157

19.1	Gestion des protocoles de réseau . . . . .	306
21.1	Table de routage . . . . .	328
22.1	Structure du cache de routage . . . . .	368
36.1	Cache de fragment . . . . .	632

## Première partie

# Couches physique et de liaison



Nous avons décidé, après mûre réflexion, d'inclure cette première partie dans cet ouvrage (consacré à l'implémentation du sous-système réseau de Linux) à titre de complétude sur l'étude des réseaux. Elle ne joue aucun rôle dans le sous-système réseau des systèmes d'exploitation. Elle peut donc être passée sans problème pour la lecture de la suite.



## Chapitre 2

# La couche physique

La *couche physique* est chargée de la transmission des bits à l'état brut sur un canal de communication. Les problèmes de conception concernent les interfaces mécaniques (le nombre de broches d'un connecteur et leur rôle) et électriques (le nombre de volts à fournir pour représenter un 1 et un 0), la synchronisation ainsi que le support physique de transmission (*medium* en anglais). La prise en compte de cette couche concerne donc le matériel, les concepteurs des cartes réseau, des micro-ordinateurs (et des routeurs) et non les logiciels que sont les sous-systèmes réseau des systèmes d'exploitation. Par conséquent Linux n'est pas partie prenante de la couche physique.

Nous allons cependant mettre en avant un minimum de connaissances concernant la couche physique qui nous seront utiles, sans être à proprement nécessaires, pour la suite.

## 2.1 Les supports physiques

Un bit peut être représenté par une donnée mécanique, électrique (par une certaine différence de potentiel), hydraulique, pneumatique (c'est-à-dire par une certaine pression), optique ou autre. Actuellement, et sauf à titre tout à fait expérimental, les bits sont toujours représentés au niveau de l'ordinateur électroniquement en jouant sur une certaine différence de potentiel. Ce n'est pas le cas dans les canaux de communication où l'on utilise une fréquence (cas du réseau téléphonique ou des lignes spécialisées), des ondes radio ou des rayons lumineux (dans le cas des fibres optiques). Cependant les signaux sont toujours transformés à l'arrivée et, au niveau de l'ordinateur, on n'a à tenir compte que de différences de potentiel.

Nous renvoyons, par exemple, au chapitre deux de la dernière édition en cours de [TAN-81] pour une vue d'ensemble des medias physiques utilisés pour la couche physique. Pour ce qui nous concerne, nous allons nous contenter de les citer :

- les supports avec guide physique (*transmission filaire*) :
  - avec une âme en cuivre comprennent les paires torsadées et les câbles coaxiaux ;
  - remplacés pour les distances moyennes par des fibres optiques ;
- les supports sans guide physique (*transmission sans fil*) utilisent les ondes radio :
  - *via* des relais hertziens ;
  - ou des satellites de communication.

## 2.2 Interface support physique/ordinateur

Le modèle que l'on peut prendre pour l'interface entre le support transportant les données réseau et l'ordinateur est un flux de bits, comme dans le cas d'une liaison série. La liaison série fut d'ailleurs longtemps utilisée, par exemple dans le cas d'un accès au réseau *via* un modem (avant l'arrivée du câble et de ADSL).

### 2.2.1 Rappels sur les liaisons série

Nous nous permettons de renvoyer au chapitre 24 de notre livre [CEG-03] pour une étude des liaisons série. Rappelons-en ce qui est essentiel pour la suite.

Dans une **liaison série**, les données sont transmises un bit à la fois (et non pas un ou plusieurs octets à la fois). On distingue deux types de communications séries : les *communications séries asynchrone* et les *communications séries synchrones*.

#### 2.2.1.1 Liaisons série asynchrone

Dans une **communication série asynchrone**, les octets sont transmis de façon irrégulière ; il faut donc un **marqueur** pour indiquer le début de l'envoi d'un certain nombre de bits, convenu à l'avance, et un marqueur pour en indiquer la fin.

Une communication série asynchrone se caractérise par la nature et le nombre de ses bits de début et d'arrêt et par son taux de transmission :

**Bit de début et bit d'arrêt.** La technique courante pour les communications série asynchrones consiste à maintenir le signal à un niveau haut (une **marque**, en anglais *mark*) jusqu'à ce qu'une donnée soit transmise. Chaque caractère transmis doit commencer par un bit de niveau 0 (un **espace**, en anglais *space*), appelé le **bit de début** (en anglais *start bit*). Il

est utilisé pour synchroniser le transmetteur et le receveur. La donnée, d'un octet, est alors envoyée en commençant par le bit de poids le plus faible. On termine par un ou deux **bits d'arrêt** (en anglais *stop bit*) à niveau haut.

Exemple.- Le schéma 2-1 ([UFF-87], p. 515) montre les niveaux logiques pour la transmis-

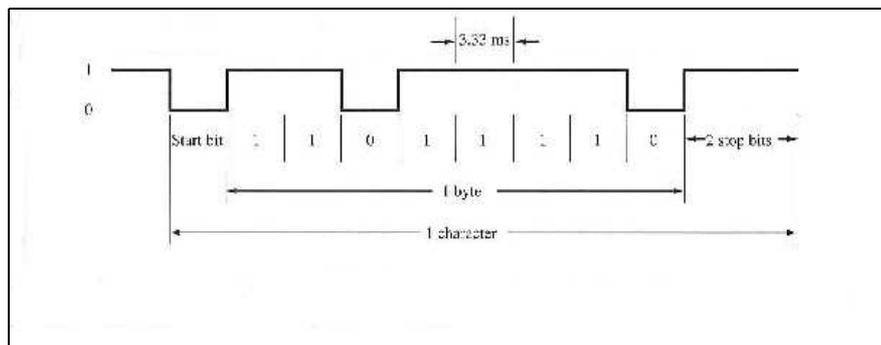


FIG. 2.1 – Niveaux logiques

sion de l'octet 7Bh dans le cas de deux bits d'arrêt.

**Taux de transmission.** Le **taux de transmission**, ou **débit** (en anglais *data rate*), est exprimé en bits par seconde (**bps**) ou en caractères par seconde.

### 2.2.1.2 Synchronisation du transmetteur et du receveur

La transmission serait facile si les débits du transmetteur et du receveur étaient parfaitement synchronisés. Mais il y a peu de chance que ceci soit réalisé.

La figure 2-2 ([UFF-87], p. 519) montre le résultat lorsque le débit est trop rapide ou trop

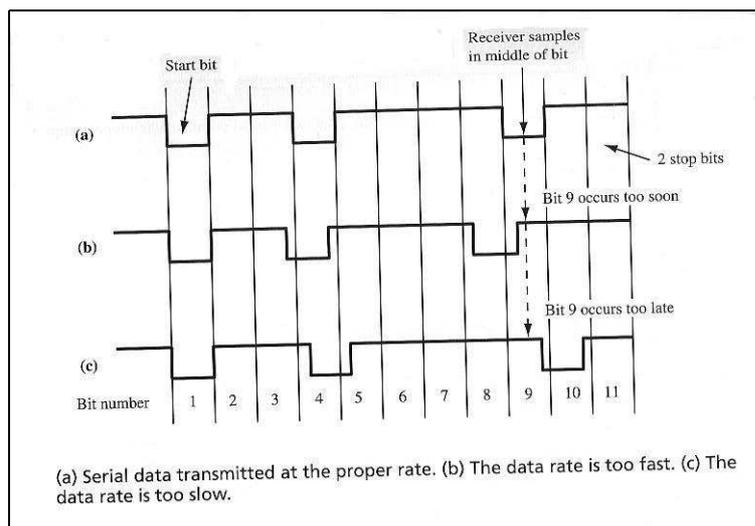


FIG. 2.2 – Synchronisation

lent. Dans ces deux cas, les erreurs s'accumulent. Si on regarde ce qui se passe au milieu de la transmission d'un bit, l'erreur maximum permise fait que le neuvième bit sera décalé à droite ou à gauche d'un demi DELAY. Si tous les bits sont décalés de la même façon, la quantité d'erreur sur un bit sera de  $1/2 \text{ durée} / 9 = 1/18 \text{ durée d'un bit}$ . Ceci signifie que les débits du transmetteur et du receveur doivent coïncider avec une erreur d'au plus 5.6 %.

La technique asynchrone doit **s'auto-synchroniser** après chaque caractère. Bien sûr le prix à payer est que la longueur de chaque octet de donnée doit être accrue de deux bits, soit de 25 %.

### 2.2.2 La synchronisation et le codage Manchester

Pour résoudre le problème de la synchronisation, Ethernet, par exemple, n'emploie pas directement un codage binaire simple avec des tensions de 0 et V volts pour coder les valeurs de bit 0 et 1, respectivement, mais le **codage Manchester**. Il s'agit d'une méthode qui permet au récepteur d'identifier avec exactitude le début, la fin ou le milieu de chaque bit, sans devoir recourir à une horloge externe.

Chaque période représentative d'un bit est divisée en deux intervalles égaux. Pour coder un bit à 1, on envoie une tension haute pendant le premier intervalle, suivie d'une tension basse durant le second intervalle. Pour coder un bit à 0, on fait l'inverse: on envoie d'abord une tension basse, puis une tension haute. Cette stratégie garantit que chaque période de codage d'un bit comprend une transition en son milieu, ce qui permet au récepteur de se synchroniser plus facilement avec l'émetteur.

La figure 2.3 illustre le codage Manchester.

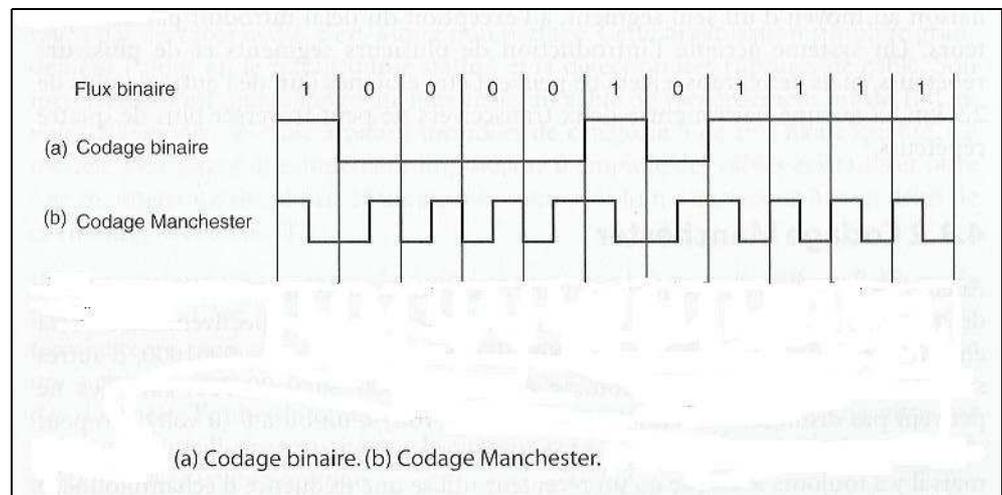


FIG. 2.3 – Codage Manchester

L'inconvénient de cette technique est qu'elle nécessite un débit double par rapport à un codage binaire simple, car chaque impulsion n'occupe qu'une moitié de période significative.

Tous les systèmes Ethernet utilisent le codage Manchester en raison de sa simplicité. Le signal de tension haute est +0,85 V et celui de tension basse - 0,85 V, ce qui donne une valeur de courant direct de 0 V.

## 2.3 Hiérarchie des réseaux

Nous avons déjà parlé du premier réseau de Tom Marril reliant deux ordinateurs et du réseau ARPANET réparti sur tous les États-Unis. Les réseaux informatique ont connu un tel succès que les entreprises ont utilisées des réseaux au sein de celles-ci (pour partager les fichiers et les ressources matérielles telles que les imprimantes). Depuis le début du vingt-et-unième siècle, même les particuliers utilisent un réseau dans leur appartement.

### 2.3.1 Hiérarchie suivant la taille

On hiérarchise les réseaux, plus ou moins suivant la distance entre les ordinateurs, de la façon suivante :

- Les **réseaux personnels** (**PAN** pour l'anglais *Personal Area Network*) se destinent à une personne. Ils utilisent des techniques de réseau sans fil, en particulier le protocole *Bluetooth*.
- Les **réseaux familiaux** se destinent à une famille dans un appartement. Ces réseaux, apparus à la fin du vingtième siècle, ont d'abord utilisé les techniques des réseaux locaux puis on est passé à des techniques de réseau sans fil, et en particulier à l'un de ses protocoles appelé **Wi-FI** (pour *Wireless Fidelity*).
- Les **réseaux locaux** (**LAN** pour l'anglais *Local Area Network*) sont des réseaux privés implantés dans un même immeuble ou dans un même campus dont la taille peut atteindre quelques kilomètres.

Ils se distinguent d'autres catégories de réseaux par trois caractéristiques: leur taille, leur technologie de transmission et leur topologie. Nous venons de parler de leur taille. Il s'agit de réseau de diffusion, et non de réseau point à point, technique sur laquelle nous reviendrons dans la sous-section suivante. Nous reviendrons également sur leur topologie ci-dessous.

- Les **réseaux métropolitains** (**MAN** pour *Metropolitan Area Network*) couvrent une ville. L'exemple le plus connu de MAN est le réseau de télévision par câble que l'on trouve dans nombre de villes. Celui-ci s'est développé à partir de l'ancien système d'antenne collective qui était utilisé dans les zones souffrant d'une mauvaise réception: on plaçait une grosse antenne en haut d'une colline pour ensuite conduire le signal par câble jusqu'au domicile des abonnés.

Il s'agissait au départ de systèmes *ad hoc* dont la conception ne répondait qu'à des besoins locaux. Lorsque les entreprises se sont attaquées au marché, elles ont obtenu des gouvernements des contrats leur permettant de câbler des villes entières. L'étape suivante a été la programmation d'émissions de télévision qui a donné naissance à des chaînes expressément conçues pour le câble. Lorsque le public a afflué sur l'internet, les cablo-opérateurs ont réalisé qu'à l'aide de peu de changements, ils pourraient offrir un service d'accès à l'internet bidirectionnel dans certaines parties inutilisées du spectre.

- Les **réseaux longue distance** ou **réseaux étendus** (**WAN** pour *Wide Area Network*) s'étendent sur une vaste zone géographique (un pays, voire même un continent).

On ne sait pas concevoir un WAN comme une seule entité. Il s'agit toujours de plusieurs réseaux locaux regroupés au moyen d'une méthode de connexion rapide (ligne téléphonique à haute vitesse, câble réseau spécialisé et très rapide, liaison hertzienne, fibre optique, etc.) appelée **dorsale** (*backbone* en anglais). Les réseaux locaux formant le réseau étendu s'appellent **sous-réseaux**.

Un réseau étendu composé de deux réseaux locaux peut être vu de la façon schématisée à la figure 2.4 :

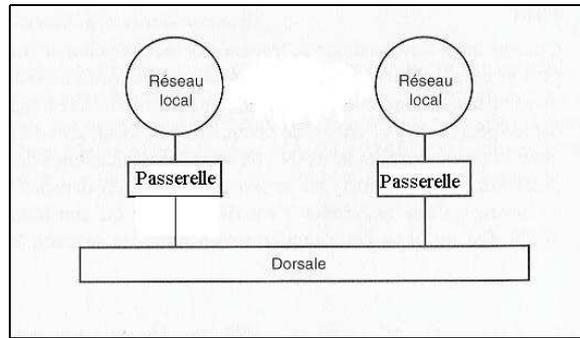


FIG. 2.4 – Réseau étendu

- chacun des réseaux locaux possède un ordinateur spécialisé, appelé **passerelle** (*gateway* en anglais);
- les deux passerelles sont reliées par une dorsale.

Le rôle d'une passerelle est triple :

- lorsqu'un message est envoyé d'un ordinateur A vers un ordinateur B du réseau étendu, la passerelle du réseau sur lequel se trouve l'ordinateur A vérifie si le message est destiné à un ordinateur d'un autre réseau local, si c'est le cas elle l'envoie sur la dorsale;
  - avant de l'envoyer, elle transforme les données du format de la suite réseau du réseau local en un format commun du réseau étendu si les deux réseaux utilisent des protocoles différents;
  - lors de l'arrivée sur le réseau local de l'ordinateur B, elle transforme les données du format commun en des données du format local.
- L'étape suivante consiste à relier tous les réseaux entre eux à l'échelle planétaire pour obtenir un **interréseau** (*internet* avec un 'i' minuscule en anglais).

Du point de vue de la technologie, un interréseau n'est rien d'autre qu'un énorme WAN.

Il a existé plusieurs interréseaux avec de jolis noms tels que ARPANET, BITNET ou INTERNET (avec un 'i' majuscule). Ils utilisent tous des systèmes ouverts, mais différents les uns des autres. Comme le lecteur le sait sûrement, Internet demeure pratiquement le seul interréseau à l'heure actuelle, s'étendant sur toute la planète. **Internet** se caractérise par l'utilisation de la suite de logiciels TCP/IP.

### 2.3.2 Réseau à diffusion et réseau point-à-point

On distingue deux types de réseau du point de vue de leur **technologie de transmission** :

- Un **réseau à diffusion** (*broadcast* en anglais) dispose d'un seul canal de transmission qui est partagé par tous les équipements qui y sont connectés. Sur un tel réseau, chaque message envoyé est reçu par toutes les machines du réseau. Le message doit donc comporter un indice permettant d'identifier le destinataire réel. À réception de ce message, une machine cherche l'indice et procède au traitement du message si celui-ci la concerne, ou l'ignore dans le cas contraire.

- Par opposition, un **réseau point-à-point** consiste en un grand nombre de connexions, chacune faisant intervenir deux machines. Pour aller de sa source à sa destination, un message peut alors transiter par plusieurs machines intermédiaires. Ce type de réseau offre souvent plusieurs **routes** possibles de longueurs différentes pour atteindre la même destination.

Bien entendu, le débit souffre de l'utilisation d'un réseau à diffusion. D'une manière générale, le système à diffusion sera utilisé sur des petits réseaux géographiquement limités alors que le système point-à-point sera plutôt implanté pour les points d'accès aux sous-réseaux d'un réseau étendu.

Le type de transmission concerne à la fois la couche physique (nous avons vu comment les bus permettaient la diffusion) et la couche liaison. En effet la transmission de bits n'est pas suffisante à cause de l'indice que le message doit contenir quant à l'adresse du destinataire dans le cas d'un réseau à diffusion. Nous verrons qu'on utilise les trames pour résoudre ce problème : l'adresse est un des champs de l'en-tête de chaque trame.

Une transmission point-à-point entre un expéditeur et un destinataire est appelée **diffusion individuelle** (*unicast sending* en anglais).

## 2.4 Réseaux locaux

### 2.4.1 Technologie des réseaux locaux

Dans un réseau local les divers composants ont été reliés successivement dans le temps par le port série, un câble coaxial puis maintenant une paire torsadée **Ethernet**.

Les diverses technologies employées font que chaque élément ne doit pas être éloigné de l'élément auquel il est relié directement de plus de quelques centaines de mètres, comme l'indique le tableau suivant :

Câble	Longueur maximum
série	15 m
coaxial fin	185 m
10base T (paire torsadée)	100 m
coaxial gros (jaune)	500 m

On comprend que les composants d'un réseau local sont rassemblés dans une même pièce ou un même immeuble. Il s'agit quelquefois même de la définition d'un réseau local, qui peut alors utiliser des supports physiques réservés aux réseaux étendus (la fibre optique en particulier).

Tous les composants d'un réseau local sont interconnectés par le biais d'un type unique de câble réseau.

### 2.4.2 Topologie des réseaux locaux

La disposition du réseau, c'est-à-dire les composants eux-mêmes et la façon dont ils sont interconnectés, est appelée la **topologie du réseau**.

Bien qu'il existe de nombreuses topologies de réseaux locaux, on en compte trois principales : machine à machine, en bus, en anneau et en étoile.

### 2.4.2.1 Réseau machine à machine

Le **réseau machine à machine** (*peer-to-peer* en anglais) est le plus simple de tous. Un ordinateur est relié à un de ses voisins en utilisant un **câble Ethernet fin** (*Thin Ethernet* en anglais, semblable au câble coaxial utilisé pour la télévision) ou un câble à paires torsadées (semblable aux câbles téléphoniques). Ce type de réseau consiste en une certaine longueur de câble coaxial qui court d'une machine à une autre. Chaque composant est connecté directement au câble à l'aide d'un connecteur en forme de T sur la carte interface du réseau, appelé **connecteur BNC** (pour *Bayonet Neill Concelman*). Celui-ci connecte la machine à deux de ses voisins au moyen d'un câble pour chaque voisin. Aux extrémités du réseau, là où il n'y a qu'un seul voisin, on ajoute une terminaison sur l'un des connecteurs en T. La figure suivante ([PAR-96], p. 7) montre le schéma de ce type de réseau :

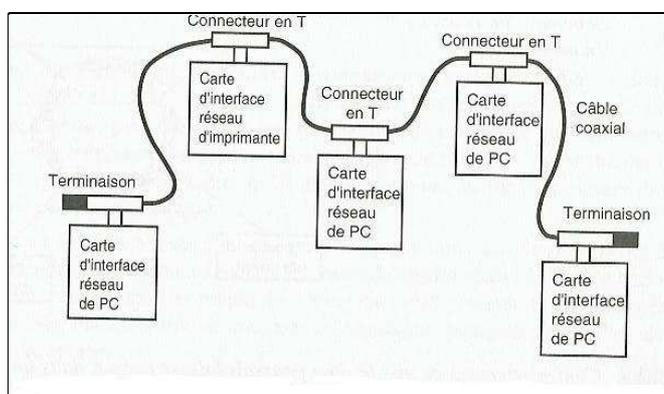


FIG. 2.5 – Réseau machine à machine

L'avantage des réseaux machine à machine tient à leur simplicité. Pour lui ajouter une nouvelle machine, il suffit d'installer une carte réseau et de connecter la nouvelle machine à un emplacement logique sur la dorsale. Ils ont aussi l'avantage d'être peu coûteux : c'est la topologie de réseau local la moins chère.

L'inconvénient principal de cette topologie est que si une machine est coupée du câble réseau, si la carte réseau ne fonctionne plus, ou si l'ordinateur est tout simplement éteint, la dorsale est coupée et doit être remise en place d'une façon ou d'une autre pour que le réseau fonctionne à nouveau.

De plus ce type de réseau n'est pas capable d'atteindre la vitesse des topologies suivantes en raison du câble réseau utilisé. Le débit des réseaux machine à machine fut longtemps limité à 10 Mbps (Mégabits par seconde). Des solutions telles que **Ethernet rapide** (*Fast Ethernet* en anglais) permettent des débits allant jusqu'à 100 Mbps.

### 2.4.2.2 Réseau en bus

Le **réseau en bus** permet de résoudre le principal inconvénient du réseau machine à machine. Il existe un unique chemin de communication principal, appelé **bus** ; chaque composant est rattaché au bus (matérialisé par un câble) par le biais d'un périphérique actif appelé **transmetteur/récepteur**, en abrégé **transcepteur** (*transceiver* en anglais) ou boîte de jonction. Le bus est également appelé une **dorsale** (*backbone* en anglais) car sa topologie fait penser à une épine dorsale. De chaque transcepteur du bus part un autre câble (très court la plupart du temps) qui se connecte à la carte réseau du composant. La figure 2.6 ([PAR-96], p. 6) montre le schéma d'un

réseau en bus. Chaque extrémité du bus est terminée par une résistance, appelée **terminaison**, qui marque électriquement la fin du bus. Chaque composant du chemin principal dispose d'un nombre identificateur – ou **adresse** – spécial qui lui permet de savoir si les informations qui lui arrivent lui sont bien destinées.

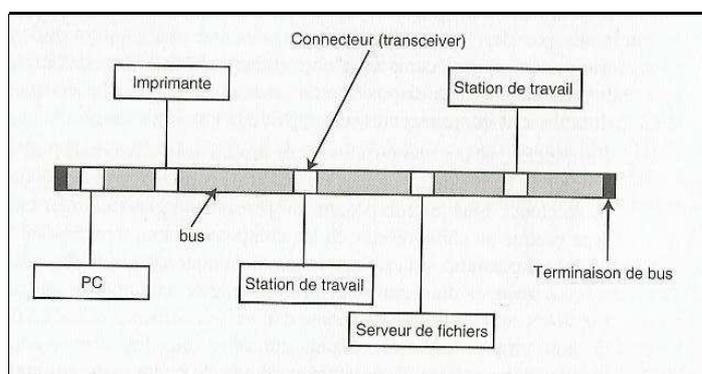


FIG. 2.6 – Réseau en bus

Le principal avantage de cette topologie est qu'elle permet un bus à haute vitesse. Autre avantage, le réseau en bus n'est généralement pas affecté si la carte réseau d'un des composants a des problèmes. En effet, le transcepteur assure l'intégrité du trafic à travers la dorsale, qu'un composant lui soit attaché (et est allumé) ou non.

Pour sa mise en place effective, un réseau en bus est rarement composé d'un câble rectiligne. Il suit les contours des murs et la disposition des immeubles en fonction des besoins. Ce qui résulte de la topologie est que son chemin est unique et les deux extrémités en sont marquées par un dispositif de terminaison. La figure ci-dessus montrait une représentation logique du réseau, en n'utilisant que des lignes droites et en ne respectant pas l'échelle.

### 2.4.2.3 Réseau en anneau

Comme son nom le suggère, un **réseau en anneau** (*ring* en anglais) est souvent représenté sous la forme d'un anneau, comme le montre la figure 2-7 ([PAR-96], p. 8).

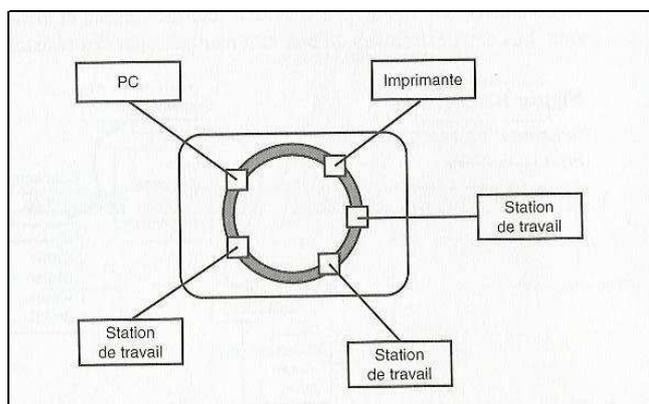


FIG. 2.7 – Réseau en anneau

Contrairement à ce que le nom pourrait laisser penser, dans un réseau en anneau les extrémités de la dorsale ne se rejoignent pas pour former une boucle; le terme d'anneau s'applique à l'unité de contrôle centrale, qui dépend de la couche liaison et non de la couche physique.

#### 2.4.2.4 Réseau en étoile

Un **réseau en étoile** utilise un câble principal, très proche de celui en bus, appelé **dorsale** (*backplane* en anglais); un ensemble de câbles va de la dorsale aux **concentrateurs** (*hub* en anglais), qui sont des boîtes contenant plusieurs **ports** dans lesquels sont branchés les composants. Les câbles menant à un point de connexion sont souvent appelés des **câbles de descente** (*drop* en anglais) car à l'origine ils descendaient des faux-plafonds. La figure 2.8 montre le schéma d'un réseau en étoile ([PAR-96], p. 9).

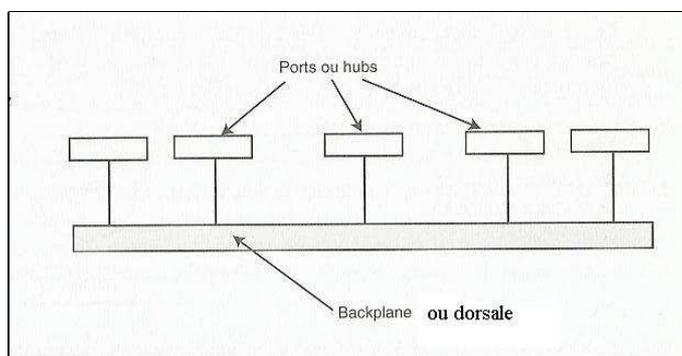


FIG. 2.8 – Réseau en étoile

Les réseaux en étoile peuvent atteindre de grandes tailles, comportant alors une dorsale à fibre optique et des câbles des descente Ethernet moins rapides. Ils peuvent également être plus petits, ne comporter qu'un seul concentrateur prenant en charge quelques composants interconnectés au moyen de câbles Ethernet standard. Le réseau en étoile est facilement extensible, il peut donc être étendu au fur et à mesure des besoins, ce qui n'est pas un mince avantage.

Les réseaux en étoile sont souvent utilisés dans les grandes installations, car ils sont faciles à mettre en place et à entretenir. Le coût en est réduit.

## 2.5 Historique

### 2.5.1 Émergence de réseaux de recherche

Le premier réseau, ARPANET, s'est rapidement développé comme le montre la figure 2.9 ([TAN-81], p. 58).

Les années 1970 ont vu l'émergence de nombreux réseaux à commutation de paquets autres que ARPANET, chacun d'eux au service d'une communauté scientifique, tels que **CSNET** (entre informaticiens), **BITNET** (entre professeurs d'économie et de science du comportement) et une myriade de petits réseaux de recherche situés dans d'autres pays que les États-Unis.

À la fin des années 1970, la NSF (*National Science Foundation*) a réalisé combien l'impact d'ARPANet sur la recherche universitaire était grand, puisqu'il permettait aux scientifiques de partager des données et de travailler sur des projets communs à travers tout le pays. Toutefois, pour pouvoir participer à ce réseau, une université devait être sous contrat avec le DoD (*Department of Defence*), ce qui n'était pas le cas pour beaucoup d'entre elles. La NSF a donc décidé de

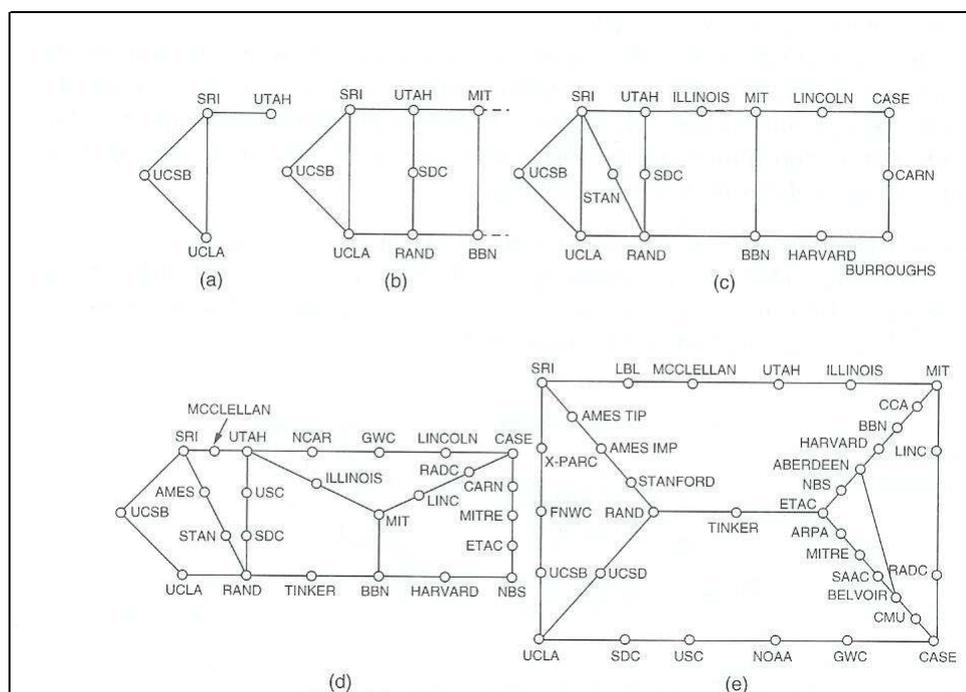


FIG. 2.9 – Croissance d'Arpanet : (a) décembre 1969 (b) juillet 1970 (c) mars 1971 (d) avril 1972 (e) septembre 1972

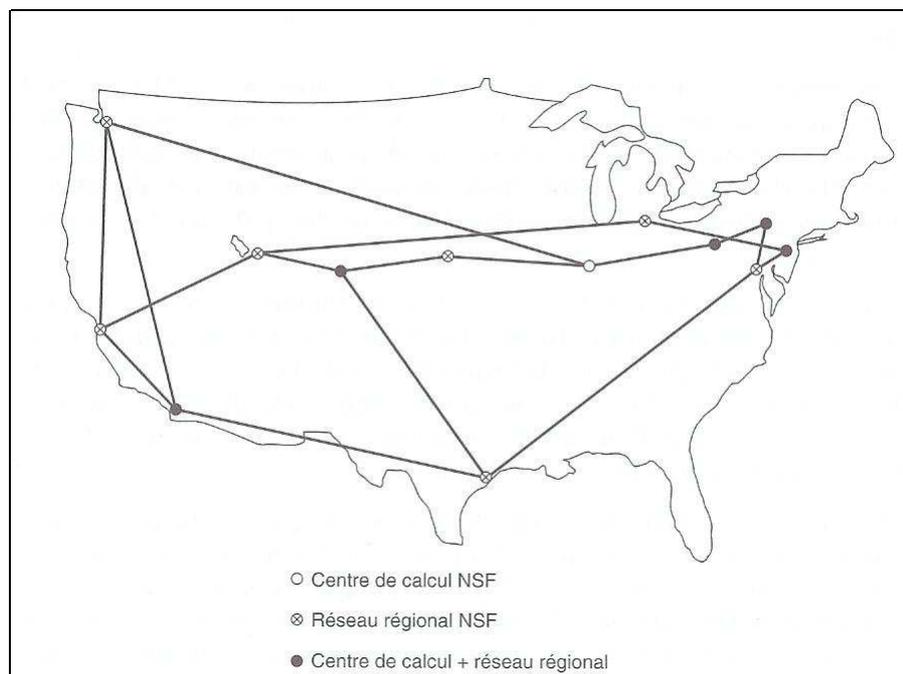
créer un réseau fédérateur pour connecter les six centres de calcul basés à San Diego, Boulder, Champaign, Pittsburgh, Ithaca et Princeton. À chacun des six supercalculateurs était associé un micro-ordinateur LSI-11 ou **fuzzball**. Ces fuzzballs étaient reliés au moyen de liaisons louées à 56 kbit/s. Sur le plan logiciel, les fuzzballs supportaient dès le départ TCP/IP.

La NSF a également financé plusieurs réseaux régionaux (une vingtaine au total), qui ont été reliés à l'épine dorsale pour permettre aux utilisateurs de milliers d'universités, de laboratoires de recherche, de bibliothèques et de musées d'accéder à n'importe quel supercalculateur et de communiquer entre eux. Le réseau complet, incluant l'épine dorsale et les réseaux régionaux, a été nommé **NSFNET**. Il a été raccordé à ARPAnet au moyen d'une liaison entre un IMP et un fuzzball dans la salle des machines de l'université de Carnegie-Mellon. La figure 2.10 ([TAN-81], p. 60) illustre l'épine dorsale NSFNET initiale.

### 2.5.2 Naissance des réseaux à diffusion : ALOHANET

Norman Abramson et ses collègues de l'université d'Hawaï tentaient de relier les utilisateurs d'îles éloignées à l'ordinateur central situé à Honolulu au début des années 1970. Cependant l'archipel ne disposait pas encore d'un système téléphonique opérationnel.

L'option de tirer des câbles sous l'océan Pacifique paraissant peu envisageable, ils ont cherché une autre solution. Ils en sont venus à utiliser des ondes radio de faible portée. Chaque terminal utilisateur était équipé d'un petit dispositif radio à deux fréquences : l'une montante (vers l'ordinateur central) et l'autre descendante (depuis l'ordinateur central). Lorsqu'un utilisateur souhaitait contacter le central, il transmettait simplement un paquet contenant les données sur le canal montant. Si personne d'autre ne transmettait au même moment, le paquet était acheminé

FIG. 2.10 – *Épine dorsale NSFNET en 1988*

vers sa destination, puis était acquitté par voie descendante. Lorsqu'une situation de conflit se présentait pour l'utilisation du canal montant, le terminal remarquait l'absence d'acquiescement et essayait à nouveau d'émettre. Il ne pouvait pas y avoir de conflit sur le canal descendant puisque seul l'ordinateur central y émettait. Ce système, baptisé ALOHANET, fonctionnait assez bien dans des conditions de faible fréquentation mais réagit par contre très mal à l'intensification du trafic montant [ABR-70, ABR-85].

### 2.5.3 Naissance des réseaux locaux : Ethernet

#### 2.5.3.1 Le PARC

À peu près à la même époque, un étudiant nommé Bob Metcalfe partait décrocher un doctorat à Harvard après avoir obtenu son diplôme de licence au MIT. Confronté aux travaux d'Abramson durant ses études, il les a trouvés si intéressants qu'une fois sorti de Harvard, il a décidé de passer l'été à Hawaï avec le chercheur avant d'intégrer le centre de recherches de Palo Alto (PARC, *Palo Alto Research Center*) de Xerox. À son arrivée au PARC, il a découvert que les chercheurs avaient conçu et construit un ordinateur personnel. Mais les machines étaient isolées. Mettant à profit sa connaissance des travaux d'Abramson, Metcalfe et son collègue David Boggs ont développé et implémenté le premier réseau local [M-R-76].

Ils l'ont appelé **Ethernet** d'après l'éther lumineux à travers lequel les scientifiques ont pensé à une époque que les ondes se propageaient. Le support de transmission n'était pas ici le vide mais un câble coaxial épais (l'Ether) long de 2,5 km avec des répéteurs tous les 500 m. On pouvait relier 256 machines au système en même temps *via* des trançeurs fixés au câble à l'aide d'une vis comme le montre la figure 2.11 ci-dessous ([TAN-81], p.72).

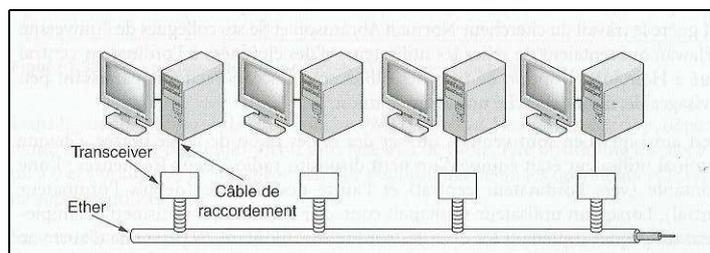


FIG. 2.11 – Architecture du premier réseau Ethernet

### 2.5.3.2 3Com

La société Xerox était déjà connue par ses inventions originales qu'elle ne commercialisait jamais, par exemple l'ordinateur personnel dont nous avons parlé ci-dessus. Comme elle ne semblait intéressée que par la normalisation d'Ethernet, Metcalfe a fondé sa propre entreprise, **3Com**, pour commercialiser des adaptateurs Ethernet pour les micro-ordinateurs. Il en a vendu plus de 100 millions.

### 2.5.3.3 Autres types de réseaux locaux

En 1977, la Datapoint Corporation lance sur le marché le système de réseau local **ARCnet**. En 1979, la société Ungermann-Bass Inc. est fondée par Ralph K. Ungermann et Charlie Bass, à Santa Clara (Californie), en vue du développement et de la vente de systèmes de réseaux locaux. Son produit Net/one sera lancé en 1980. En 1980, la société Novell Data Systems Inc. est créée à Provo (Utah).

## 2.5.4 Interréseaux

Vinton Cerf a formulé l'idée de relier entre eux les divers réseaux afin de constituer un vaste réseau mondial, qui va devenir **Internet**. La DARPA s'en charge en se fondant sur l'architecture d'ARPANET.

