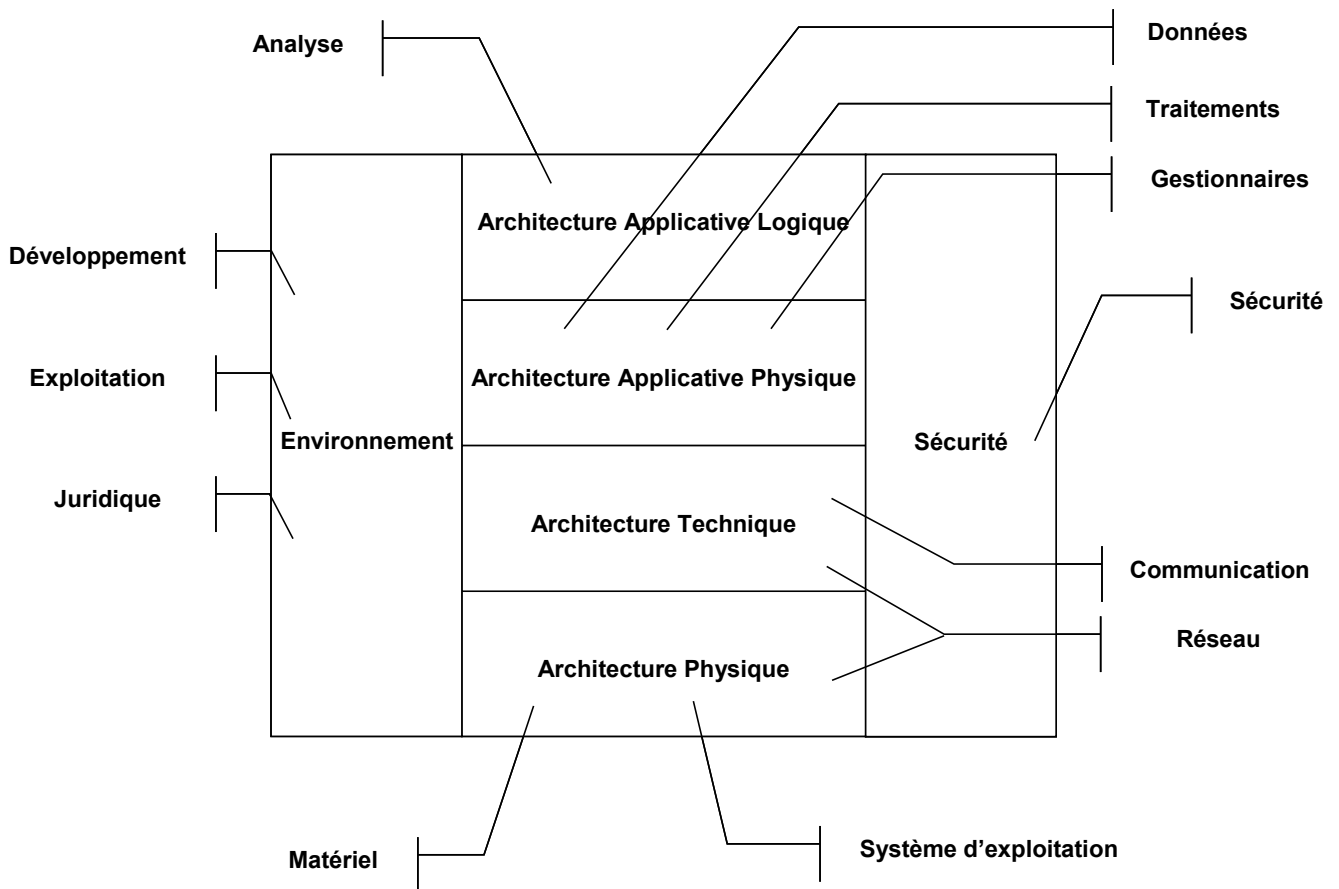


ARCHITECTURE INFORMATIQUE

IRA Analystes

L'étude de l'architecture informatique comprend l'ensemble des éléments techniques nécessaires à la réalisation d'un projet informatique. Cette phase ne peut intervenir qu'après l'analyse conceptuelle.

Il est possible de la modéliser selon le schéma ci-dessous, ce qui permet de traiter de l'ensemble des composants techniques, afin d'adopter une démarche complète et cohérente.



Plan de cours

Introduction

1. Chronologie de l'informatique
Informatique générale
Impact sur les types d'architectures
2. Architecture physique (schéma d'infrastructure)
Localisation des composants et Centralisée/Distribuée
Evolutions vers le distribué : Techno, Débits, Systèmes ouverts
3. Système d'information
Système de production : transactionnel/industriel
Système de diffusion : décisionnel/infocentre
4. Le modèle d'architecture
Présentation d'une démarche
Présentation d'un devoir d'architecture
5. Exemple sur le choix d'architecture

I) Architecture physique

6. Les matériels et logiciels associés
Poste de travail et périphériques
Réseaux
Serveurs
7. Le système d'exploitation
8. Les réseaux (couches basses)
Le modèle OSI
Les réseaux (LAN, MAN, WAN)
Le réseau local Ethernet
Les protocoles TCP/IP

II) Architecture Technique

9. Les réseaux (couches hautes)
Les services de communication
La communication inter applications

III) Architecture Applicative physique

10. La répartition des données et traitements
11. Les choix d'architecture (le client serveur)
12. Les gestionnaires de données (SGBD) et de traitements (MT)

IV) Thèmes transversaux

13. La sécurité

La sécurité du système d'information : généralités

La sécurité physique

La sécurité logique

La sécurité réseau

14. Le contexte juridique

15. Les langages de développement

16. L'Administration et la gestion du système

V) Conclusion

17. Préparation à l'écrit

18. Préparation à l'oral

Sommaire détaillé

Sommaire détaillé.....	4
Crédit	8
Chronologie de l'informatique.....	9
<i>Informatique générale.....</i>	<i>9</i>
I Les composants de bases.....	9
A. Processeur.....	9
B. Barrette mémoire.....	9
C. Carte mère	10
II Des périphériques.....	11
A. Disque dur	11
B. Carte réseau	11
C. Périphériques en vrac.....	11
III Serveurs.....	12
A. Serveur à architecture X86 au format rack (plat)	12
B. Serveur Blade	12
Lames.....	12
C. Serveur Mainframe	12
IV Eléments réseau.....	13
A. Hub	13
B. Commutateur	13
C. Routeur.....	13
D. Schéma de réseau.....	13
V Chronologie informatique, en quelques dates.....	14
<i>Types d'architectures.....</i>	<i>18</i>
I Les types d'architecture	18
II Description des architectures.....	18
III Schéma des différentes architectures :.....	19
Architecture physique (schéma d'infrastructure).....	20
<i>Localisation des composants et Centralisée/Distribuée : traitements/données.....</i>	<i>20</i>
I Architecture technique globale.....	20
A/ Centralisée.....	20
B/ Répartie.....	20
II Eléments de l'architecture.....	21
A/ Données.....	21
1) Centralisées.....	21
2) Distribuées.....	21
B/ Traitements	21
1) Centralisée.....	21
2) Répartie.....	21
C/ Utilisateurs.....	21
1) Centralisée.....	21
2) Répartie.....	21
<i>Évolution vers le distribué</i>	<i>22</i>
I Technologies, débits, systèmes ouverts.....	22
A/ Evolution des technologies	22
B/ Evolution des débits des réseaux :	22
C/ Evolution vers les systèmes ouverts :	22
II Notion de downsizing, upsizing et rightsizing	23

Système d'information.....	24
Cadre de réponse pour le domaine Architecture.....	27
Introduction.....	27
Démarche.....	27
Résultat attendu.....	28
Exemple sur le choix d'architecture.....	30
Les matériels et logiciels associés.....	32
I Poste de travail et périphériques.....	32
A/ Utilisation sédentaire.....	32
B. Utilisation nomade	32
C. Logiciel	32
II Réseaux.....	32
A/ Câblage	32
B/ Éléments d'interconnexion :.....	33
III Serveurs.....	33
A/ Serveur à architecture X 86 (monde Intel et AMD)	33
B. Serveur de type gros système*	33
IV Point sur la redondance	34
Le système d'exploitation.....	35
I Introduction.....	35
A. Définition du terme Système d'exploitation (SE).....	35
B. Rappel sommaire sur l'architecture interne d'un ordinateur :	35
C. Place du SE dans le monde des logiciels	35
D. Une notion de base des SE : le monotâche et le multitâche.....	37
II Gestion des processus.....	37
A. Description de la notion de processus.....	37
B. Les interruptions.....	38
C. Traitement d'une interruption	38
D. Comment attribuer le temps CPU ?.....	38
E. Synchronisation des processus	39
III Gestion de la mémoire.....	39
A. Une mémoire découpée.....	39
B. La mémoire virtuelle.....	40
IV Gestion des Entrée/Sorties.....	40
A. Description des périphériques	40
B. Comment se fait la gestion des entrées/sorties.....	41
V Gestion de fichiers.....	41
A. Manipulation des fichiers.....	42
B. Mode d'accès à un fichier.....	42
C. Rôle de la gestion des fichiers.....	42
D. Gestion des blocs.....	43
VI Gestion des travaux.....	43
A. La communication avec le système d'exploitation.....	43
B. Prise en charge des demandes.....	43
C. Comptabilité et statistiques.....	43
VII Thèmes associés	44
A. Classification des Systèmes d'exploitation	44
B. Evolution des Systèmes d'Exploitation.....	44
C. Vocabulaire.....	45
Les réseaux.....	46

<i>Le modèle OSI</i>	46
I Les couches.....	46
A. Présentation du modèle.....	46
B. Les couches du modèle OSI	46
II Principe de fonctionnement du modèle	47
A. Protocole et interface.....	47
B. Fonctionnement.....	47
<i>Les réseaux (LAN, MAN, WAN)</i>	50
I Typologie des réseaux.....	50
A. Généralités.....	50
B. Les types de réseaux : LAN, MAN, WAN.....	50
1) Réseau local (LAN: Local Area Network).....	50
2) Réseau métropolitain (MAN: Metropolitan Area Network).....	51
3) WAN: Wide Area Network.....	51
II Principes des réseaux.....	51
A. La topologie.....	51
1) Le bipoint	51
2) Le multipoint	52
B. Le signal et le câblage	52
1) le signal.....	52
2) le câblage et connecteur associé (couche 1 du modèle OSI).....	54
C. Les éléments d'interconnexion	56
1) le segment.....	56
2) le répéteur.....	56
3) le pont	56
4) le routeur.....	56
5) la passerelle	56
6) le multiplexeur	56
7) le concentrateur	56
8) le commutateur.....	56
<i>Les réseaux locaux</i>	58
I Introduction au réseau local.....	58
A. Méthode d'accès.....	58
B. Schéma d'échange des données.....	58
II Types de réseau local.....	58
A. Ethernet.....	58
B. Les réseaux en anneau.....	59
1) Anneau à jeton Token ring.....	59
2) FDDI.....	60
<i>Les réseaux : les protocoles TCP/IP</i>	61
I Notion de commutation	61
A/ Commutation de circuit	61
B/ Commutation de message	61
C/ Commutation de paquets	61
D/ Commutation de Cellule	62
II Principes de IP	62
A/ Description fonctionnelle.....	62
B/ Adressage.....	62
1) Classe d'adresse	62
2) Attribution d'adresse.....	63

3) Résolution d'adresse : fonctionnement du DNS.....	64
C/ Fragmentation.....	64
III Le protocole TCP.....	65
A/ Caractéristiques.....	65
B/ Interfaces.....	65
C/ Fonctionnement.....	65
IV Autres protocoles	66
A/ UDP (couche 4).....	66
B/ ARP (couche 3)	66
C/ ICMP (couche 4).....	66
V Pour aller plus loin.....	66
.....	66
A/ L'en-tête IP	66
B/ Les fonctionnalités de TCP.....	69
1) Transfert de données de base.....	69
2) Contrôle d'erreur.....	70
3) Contrôle de flux.....	70
4) Multiplexage.....	70
5) Connexions.....	70
C/ L'en-tête TCP.....	71
D/ Schéma d'encapsulation TCP/IP.....	73

Crédit

- Informatique générale : Wikipédia, www.histoire-informatique.org
- Le système d'exploitation : fascicule PSE-CRA, fascicule d'architecture de la préparation analyste.
- Les réseaux (LAN, MAN, WAN) : fascicule d'architecture de la préparation analyste
- La suite des protocoles TCP/IP : site www.guill.net
- Les gestionnaires de données (SGBD) et de traitements (MT) : fascicule PSE-CRA
- La sécurité, généralités : fascicule Analyste
- La sécurité réseau : intervention de S. Laigle
- Le contexte juridique : intervention de M. Soykurt
- Les langages de développement : document de C. Antunes

Fiche thème 1	Chronologie de l'informatique
---------------	-------------------------------

Fiche thème 1-1	Informatique générale
-----------------	-----------------------

Préambule : Cette première partie est destinée à ceux qui, comme moi, ont commencé l'informatique dans l'administration sans avoir jamais ouvert un PC de leur vie...

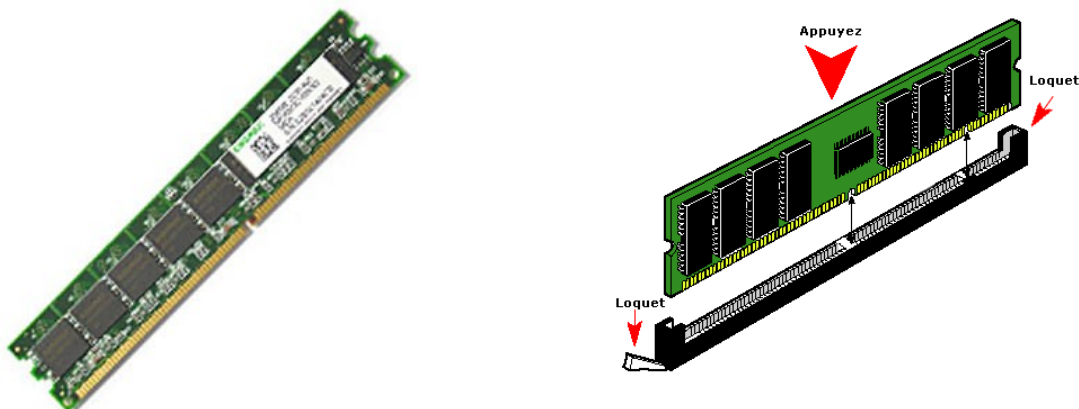
I Les composants de bases

Pour les puristes, l'unité centrale est composée du processeur (CPU), de l'horloge, du bus, et de la mémoire. Par abus de langage, on appelle souvent unité » centrale le boîtier qui contient ces éléments, mais celui-ci contient aussi des périphériques... Comme la carte réseau. Tous les autres éléments devrait être considérés comme « périphériques ». Cependant, quand on parle de périphérique, on entend surtout : écran, clavier, souris, imprimante scanner...

A. Processeur



B. Barrette mémoire



La mémoire vive, ou RAM est ce qui permet à l'ordinateur de stocker des informations volatiles : en tapant le document, la mémoire vive est sollicitée. Plus un PC a de mémoire vive, plus il pourra répondre rapidement au demande de l'utilisateur (standard actuelle barrette de 1Gigaoctet)

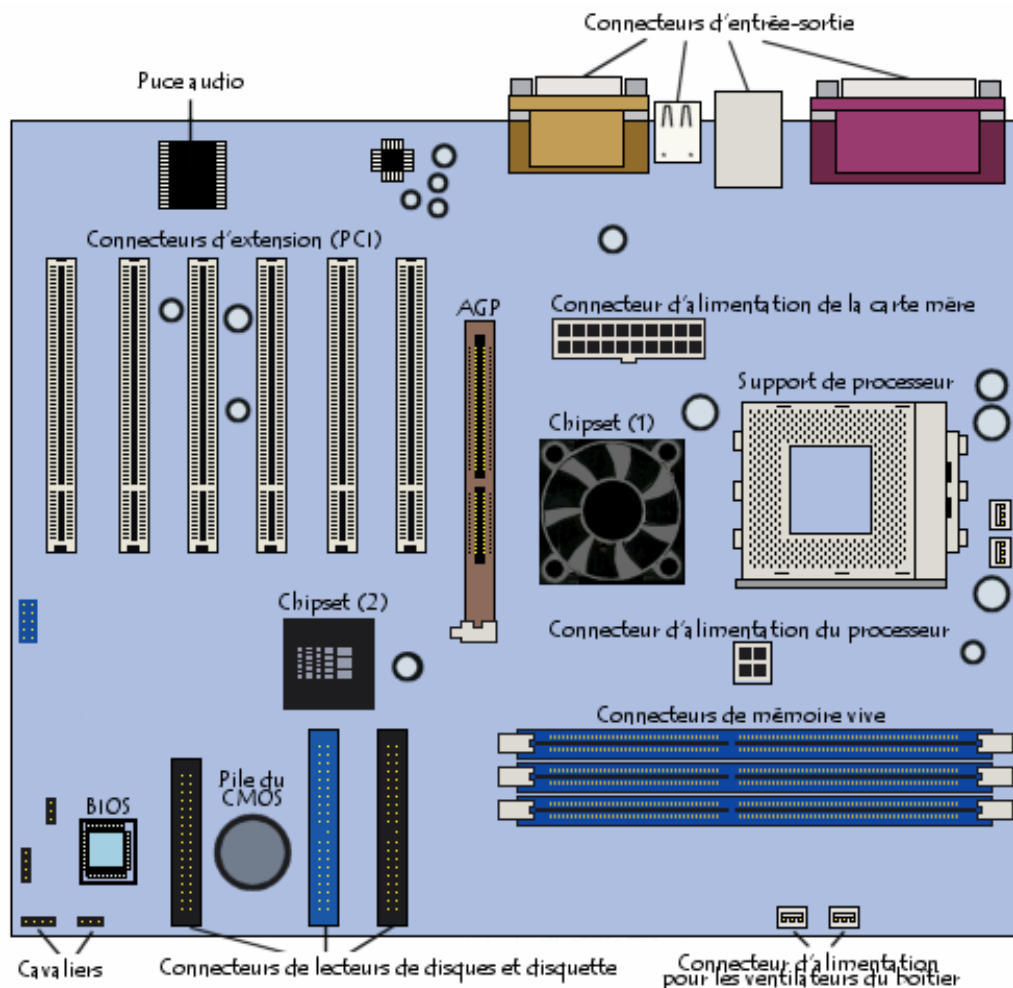
C. Carte mère

Le processeur, la mémoire, le bus sont dessus. Les barrettes mémoires s'enfichent dans un connecteur, le processeur semble être posé (dessus, il y souvent un système de refroidissement qui le cache), divers autres connecteurs sont présents (PCI par exemple...)

La plupart des ports sont portés par cette carte : ports USB, port RJ45 – la carte réseau est embarqué sur la carte mère, porte VGA pour l'écran...)



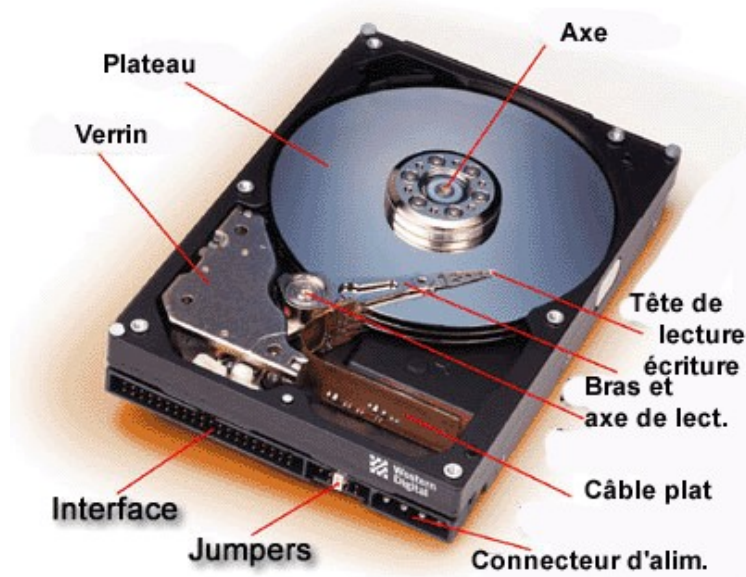
Schéma d'une carte mère



II Des périphériques

A. Disque dur

Il est relié à la carte mère par une sorte de ruban en plastique appelée nappe.



B. Carte réseau

Aujourd'hui souvent embarqué dans la carte mère sous forme de circuit intégrée, la carte réseau ressemble à ça :



C. Périphériques en vrac

Clavier, souris, écran, imprimante, modem, scanner...

III Serveurs

A. Serveur à architecture X86 au format rack (plat)

Plus puissant qu'un PC, il a la même architecture interne. Ces serveurs sont « empilés » dans des armoires, ce qui permet un gain de place importants par rapport au format tour (ressemble à des PC à s'y méprendre).



B. Serveur Blade



Des serveurs à l'architecture particulière sont glissés dans un châssis qui embarque de nombreuses fonctionnalités (réseau entre autre, administration...) Ces solutions permettent de multiplier facilement les serveurs, pour des infrastructures qui nécessitent beaucoup de machines identiques (forte charge : serveur web, serveur annuaire...)

C. Serveur Mainframe

Serveur très puissant ayant une architecture interne spécialement pensée pour les travaux de masse.



IV Eléments réseau

A. Hub



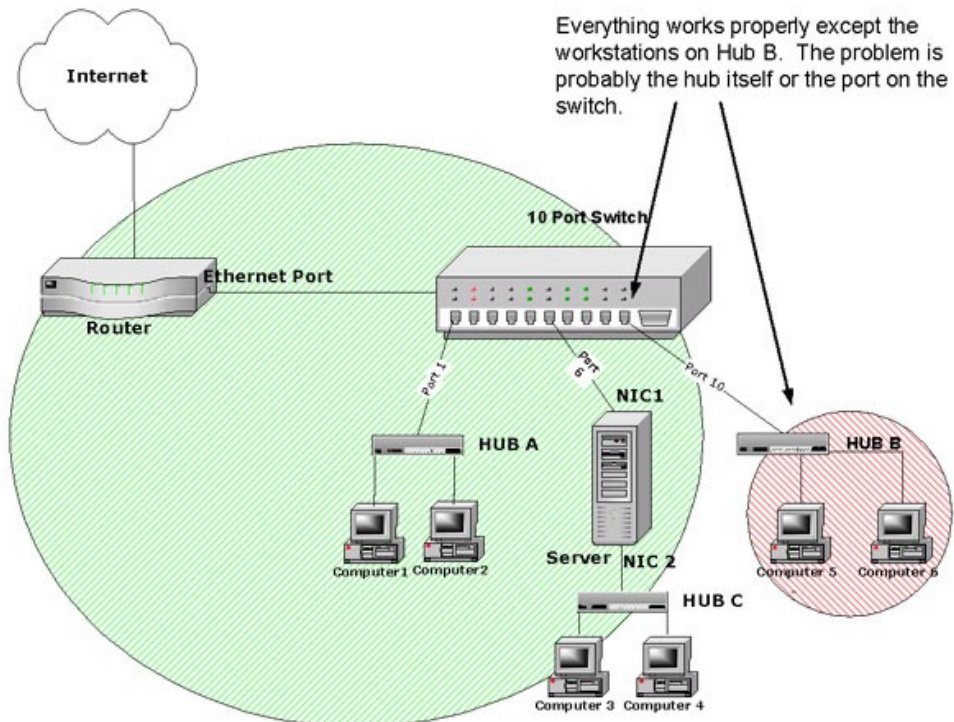
B. Commutateur



C. Routeur



D. Schéma de réseau



V Chronologie informatique, en quelques dates...

(En rose, quelques dates à retenir)

-1750 : **Code d'HAMMOURABI** : Le roi de Babylone (Mésopotamie), nommé HAMMOURABI, a fait graver cette stèle. Celle-ci composée d'un ensemble de sentences royales sous la forme :

SI {personne} ET {action} ALORS {sentence}

-330 : **Logique** : Elle est définie par le philosophe grec ARISTOTE.

820 : **Travaux du mathématicien arabe AL KHOWARIZMI**

1000 **Zéro** : Inventé en Inde et rapporté en Occident par les invasions arabes, le zéro trouvera un ardent défenseur en la personne de Gerbert d'AURILLAC qui tentera de l'imposer lorsqu'il deviendra le pape Sylvestre II. Mais, ce n'est que vers le XIV^{ème} siècle, que le monde occidental l'acceptera définitivement.

1614 : **Logarithmes** par NEPER : Grâce aux travaux de L'Ecosais NEPER, la multiplication et la division peuvent être ramenées à deux opérations très simples: l'addition et la soustraction.

1623 **Machine à calculer de W. SCHICKARD**

Composée de 6 cylindres Népériens , de réglottes coulissantes et de 6 disques opérateurs, cette machine était capable d'effectuer les reports de retenues dans un sens (addition) ou dans l'autre (soustraction). Détruite en 1624, elle ne sera reconstruite qu'en 1960 d'après les plans originaux.

1641 : **Pascaline: Machine à calculer de B. Pascal**

Cette machine, qu'il a construite afin d'aider son père, est la première qui a réellement fonctionné. Elle servira de référence pour les machines futures.

1697 : **Introduction du binaire en Europe** par G Leibniz : Passionné par la *Dyadique*, LEIBNITZ fut conforté dans ses idées lorsqu'il apprit que le binaire avait été inventé par les Chinois plusieurs millénaires auparavant. Il exposera devant l'Académie des Sciences de Paris ses idées qui seront publiées dans "Explication de l'arithmétique binaire avec des remarques sur son utilité et sur le sens qu'elle donne des anciennes figures chinoises de Fou-Hi".

1840 : **Principe des machines à calculer** par A. Lovelace.

Pour Ada Lovelace, une machine à calculer doit comporter:

- Un dispositif permettant d'introduire les données numériques (cartes perforées, roues dentées...)
- Une mémoire pour conserver les valeurs numériques entrées
- Une unité de commande grâce à laquelle l'utilisateur va indiquer à la machine les tâches à effectuer
- Un "moulin" chargé d'effectuer les calculs
- Un dispositif permettant de prendre connaissance des résultats (imprimante...)

Ces principes seront, un siècle plus tard, à la base des premiers ordinateurs

1843 : **Théorie de la programmation** par A Lovelace : ici, Ada définit le principe d'itérations successives dans l'exécution d'une opération. En l'honneur du mathématicien arabe

AL KHOWARIZMI, elle appelle "algorithme" le processus logique permettant l'exécution d'un programme.

1854 : **la logique binaire** de G Boole : dans "Les lois de la pensée", il explique que l'on peut coder les démarches de la pensée à l'aide de système n'ayant que deux états: ZERO-UN; OUI-NON; VRAI-FAUX...

1941 : **L'ABC** (Atanasoff Berry Computer : premier calculateur à utiliser le code binaire)

1943 : **L'Harvard Mark 1** : cette machine pesait 5 tonnes et avait besoin de plusieurs tonnes de glace par jour pour la refroidir. Le Colossus était une machine conçu pour décoder des messages cryptés.

1945 : **L'ENIAC** (Electronic Numerical Integrator And Calculator) : commandé par l'armée par l'armée des États-Unis en 1943 pour effectuer les calculs de balistique, il remplaçait 200 personnes chargées auparavant de calculer les tables de tir. Il occupait 23 m³, pesait 30 tonnes, coûtait un demi-million de dollars et consommait presque 200 kilowatts.

Création du premier langage de programmation : le **Plankalkül** par l'ingénieur Konrad Zuse

1947 : Invention du **transistor**.

1951 : **L'UNIVAC** : Il utilise des bandes magnétiques en remplacement des cartes perforées

1954 : Création du **FORTRAN**, premier langage de programmation à être implémenté sur un ordinateur

1956 : IBM sort le premier **disque dur** : le RAMAC 305

1958 : Le **circuit intégré**

1963 : Invention de la souris

1964 : Le PDP-8 de DEC est le **premier mini-ordinateur**.
Invention du langage de programmation **BASIC**

1965 : **La loi de Moore** : « La complexité des processeurs doublera tous les 18 mois. »

1969 : Création d'**Unix**

Internet : projet **Arpanet** de l'armée américaine, qui est à l'origine de l'Internet tel qu'on le connaît aujourd'hui.

1970 : l'Altair IV : **premier micro-ordinateur**.

1971 : Le **premier microprocesseur** : l'Intel 4004.
Création du langage C et du Pascal

1973 : l'**Alto** : ce prototype, pensé pour devenir le bureau du futur, est un condensé des idées proposées par les chercheurs réunis par XEROX au Palo-Alto Research Center (PARC). Il est

le premier à introduire l'idée de fenêtres et d'icônes que l'on peut gérer grâce à une souris. Principalement, en raison de son coût, cet ordinateur ne connaîtra qu'un succès d'estime.

Le protocole TCP/IP.

1976 : Le **premier super-calculateur** : le Cray I

1977 : Apple Computer : lancement de l'Apple II, premier ordinateur à recevoir un succès grand public, c'est une machine qui permet à ses utilisateurs de créer leurs propres logiciels d'application.

1980 : La loi française « **Informatique et Libertés** » entre en vigueur, Invention du Compact Disc (CD).

1981 : L'IBM PC, Microsoft : **MS DOS**

1983 : Création du **langage C++ et du Turbo Pascal**

1984 : Apple Computer : Sortie du Macintosh et de Mac OS,
Création de la Free Software Fondation, **lancement de la licence GPL** et du projet GNU

1985 : Le premier Amiga : l'Amiga 1000, L'AmigaOS 1.0, L'Atari ST
Microsoft : **Windows 1.0**
Le CD-ROM

1989 : **World Wide Web** : concept mis au point par Tim Berners-lee du C.E.R.N, c'est un système de recherche documentaire de données. L'utilisateur se connecte grâce un client (Navigateur ou Browser) sur un serveur désigné par l'URL (Uniform Resource Locator - c'est à dire l'adresse du site).

1991 : **Linux** : **Premier noyau linux 0.01**

1993 : Lancement de l'Intel Pentium,
Microsoft : MS Windows NT,
Internet : **Mosaic**, le premier navigateur web

1994 : Apple Computer : Lancement du Power Macintosh à base de processeur PowerPC
Internet : Création de Netscape

1995 : **Le DVD**,
Microsoft : Sortie de MS Windows 95,
Création du langage de programmation **Java**

1996 : Microsoft : Sortie de MS Windows NT4,
Microsoft : **Naissance de MS Internet Explorer.**

1998 : Apple Computer : **Lancement de l'iMac.**
Microsoft : Sortie de Windows 98.

2000 : **Microsoft** : **Sortie de Windows 2000** et de Windows ME.

2001 : Linux : Sortie du noyau 2.4,
Microsoft : Sortie de MS Windows XP,
Apple Computer : Sortie de Mac OS X 10

2002 : Microsoft : Sortie de la famille MS Windows 2003 Server

2003 : Apple Computer : Lancement du Power Mac G5

2004 : **Linux : Sortie du noyau 2.6**, la dernière version stable actuellement.

2005 : Apple Computer : annonce du passage de l'architecture Power PC au x86 d'Intel.

I Les types d'architecture

On oppose généralement les architectures centralisées et réparties (décentralisée, éclatée...). On parle d'architecture purement centralisée lorsque les 4 éléments (utilisateurs, matériels, données et traitements) sont centralisés.

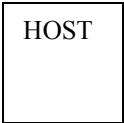
Aujourd'hui, cependant, on constate que les architectures sont souvent mixtes, avec par exemple des données centralisées et des traitements répartis.

II Description des architectures

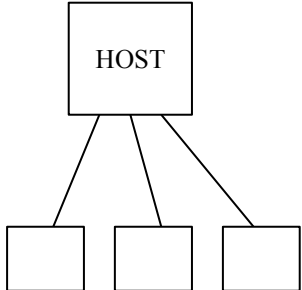
Type d'architecture	Descriptif	Exemple
Centralisée	Une machine héberge tous les traitements, données... les utilisateurs doivent travailler sur cette unique machine.	Un PC qui n'est pas en réseau représente une architecture purement centralisée.
Maître/Esclave	Il y a une ligne par terminal passif. Les coûts sont donc importants et le déploiement des terminaux ne peut être complet. Le terminal passif n'a pas d'intelligence, c'est-à-dire qu'il ne fait que de l'affichage. Les traitements, les données sont toujours sur le maître.	Minitel DAB
Hiérarchique	Le concentrateur de terminaux permet de réduire le nombre de ligne à payer aux sociétés de telecom. De plus, en réalisant des traitements aux niveaux du concentrateur (édition par exemple) on diminue le trafic réseau et le temps de réponses sont améliorée.	Des applications métier comme RAR à la DGCP (gestion du recouvrement contentieux)
Client/Serveur	Avec la micro-informatique, le PC se démocratise : la miniaturisation des équipements permet une baisse des coûts, une installation aisée, et l'avènement des systèmes d'exploitation graphique permet à tous les utilisateurs de se servir d'un PC. De plus, on utilise la capacité du PC à réaliser des traitements, stocker des données... le client est en mesure de demander au serveur des services.	Toute application web (même un site) fonctionne en mode client serveur.
Peer to Peer	On désire exploiter les ressources de toutes les machines (en capacité de traitement ou en stockage de donnée) ainsi que le développement des réseaux. Cela revient à faire de chaque PC un client <u>et</u> un serveur.	Le voisinage réseau de Microsoft Emule

III Schéma des différentes architectures :

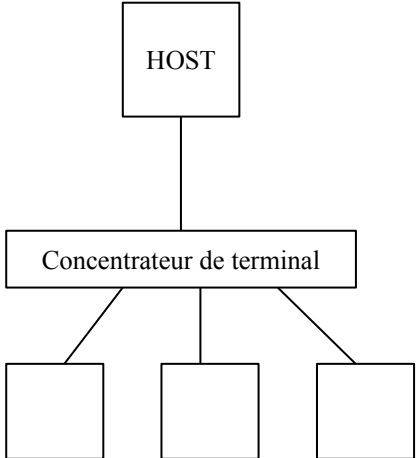
1/Architecture centralisée



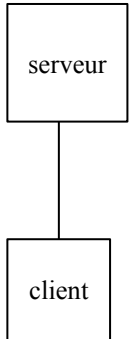
2/ Maître Esclave



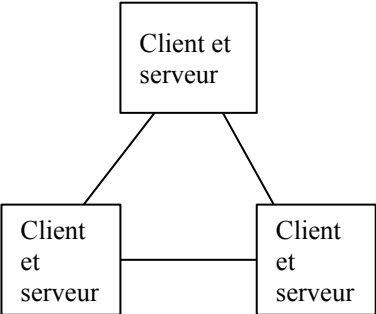
3/ Hiérarchique



4 /Client serveur



5/ Peer to peer



Fiche thème 2	Architecture physique (schéma d'infrastructure)
---------------	---

Fiche thème 2-1	Localisation des composants et Centralisée/Distribuée : traitements/données
-----------------	--

Nb : cette partie du cours est réalisée sous la forme d'un plan de fiche très classique.

I Architecture technique globale

A/ Centralisée

On centralisera pour diverses raisons :

- une exploitation + facile et + simple
- la formation des agents
- enfin sécuriser un site où tout est au même endroit est bcp + facile et surtout bcp plus fiable.

Toutefois cette centralisation a des limites. Par ex en cas de panne, « tout peut tomber », ce qui a pu être le cas pour des opérateurs de télécoms ou pour la SNCF.

Ceci nous oblige :

- d'une part à mettre en place une réplique du système avec en prévision un site de rechange.
- et d'autre part, prévoir un plan de reprise d'activité (PRA). Exemple : cas du crédit lyonnais qui suite à l'incendie du siège à Paris a du arrêter quelques jours son système informatique, en l'absence de PRA.

B/ Répartie

La mise en place d'une architecture répartie a trois avantages :

- la mutualisation des matériels, des hommes et globalement des compétences
- une souplesse dans la définition de l'architecture qui peut être modulée.
- enfin un accès à l'information bcp + rapide ce qui améliore le temps de réponse ;

Cependant ce type d'architecture a aussi des inconvénients :

- du fait de cette multiplication, la gestion et la formation des agents sont plus lourdes et plus complexes
- ce type de réseau pose un gros problème de sécurité

Chaque architecture génère un coût. Il n'y a donc pas de solution idéale. C'est pourquoi on évolue vers une mixité des 2 types d'architecture en ayant à l'esprit 3 objectifs indissociables et incontournables : améliorer le temps de réponse, sécuriser le système d'information, finaliser à moindre coût.

II Eléments de l'architecture

A/ Données

1) Centralisées

Le fait d'avoir accès aux données en un point unique permet :

- d'avoir une maintenance + facile et une meilleure protection
- de maintenir cohérence et intégrité des données.

2) Distribuées

- l'accès aux données est plus rapide (la donnée est plus proche de son utilisateur final) ;
- l'organisation des données est plus cohérente en terme de besoin local ou national, les données peuvent se trouver seulement là où elles sont nécessaires.
- Sécurité possible grâce à la redondance des données. Attention : il faudra synchroniser les données de façon à ce qu'elles restent cohérentes

B/ Traitements

1) Centralisée

- maintenance facilitée grâce à l'unicité de versions déployées (un seul site à mettre à jour...)
- diminution du trafic réseau en ce qui concerne l'application en elle même,
- par contre, les résultats produits par l'application, eux, circuleront sur le réseau.

2) Répartie

- on pourra optimiser le temps de traitement puisqu'on n'aura pas de trafic réseau pour le résultat.
- en revanche le développement sera très compliqué.

C/ Utilisateurs

1) Centralisée

C'est la meilleure façon de connaître et d'identifier les utilisateurs. Cette relation de proximité facilitera d'une part l'assistance aux utilisateurs et d'autre part leur formation.

2) Répartie

L'éparpillement d'utilisateurs augmentera les points d'entrée et de sorties de l'information

I Technologies, débits, systèmes ouverts

Les architectures ont évolué vers des architectures distribuées (ou mixtes) grâce à trois facteurs : l'évolution des technologies, l'augmentation des débits des réseaux, et le développement de systèmes dit ouverts.

A/ Evolution des technologies

Fréquence des processeurs : 3 GHz

Multiprocesseurs : jusqu'à 64 processeurs sur une machine type gros système.

Stockage : la capacité de stockage est presque illimitée — on peut avoir des péta-octets (10^{15} octets) de données.

La micro-informatique elle-même est de plus en plus puissante.

Face à ce phénomène, les gros systèmes résistent encore car leur architecture interne est optimisée pour gérer des travaux de masse.

B/ Evolution des débits des réseaux :

LAN (local area network) : 10 Mbits/s 100 Mbits/s 1000 Mbits/s.
Aujourd'hui, le standard est 100 Mbits/s.

WAN (wide area network) :

RTC (réseau téléphonique commuté) : débits de 56 kbits/s.

Numéris de 32 kbits/s à 2 Mbits/s

Lignes spécialisées jusqu'à 34 Mbits/s

Réseau très haut débits (protocole FDDI, support optique) : jusqu'à 650 Mbits/s promis. Actuellement, on atteint 155 Mbits/s.

Remarque sur l'ADSL : cela fonctionne grâce à de la compression de flux. Il s'agit d'un protocole asymétrique, puisque le débit en réception (élevé) et le débit en émission (faible) sont différents. Le débit dépend de la distance à laquelle se trouve le modem ADSL (sur le poste de la personne connectée) du DSLAM (équipement chez l'opérateur téléphonique) auquel il est raccordé. Le débit n'est pas garanti.

Ce genre de solution peut être choisi dans un devoir avec la remarque : vérifier que le lieu géographique est éligible à l'ADSL (= présence des équipements nécessaires).

Réseau sans fil : Infrarouge et WiFi. C'est le WiFi qui se développe actuellement (norme 802.11 : 802.11a : 54 Mbits/s et 802.11b : 11 Mbits/s attention, ici c'est un débit partagé).

C/ Evolution vers les systèmes ouverts :

Un système est dit ouvert s'il répond aux caractéristiques d'interopérabilité et de portabilité :

— Interopérabilité : caractéristique des systèmes qui indique la facilité à échanger des données et à communiquer. Sont interopérables des ensembles logiciels qui respectent un certain nombre de standard.

— Portabilité : un programme est écrit sur un système. S'il est portable, il n'est pas nécessaire de réécrire l'application pour l'utiliser sur un autre système.

Remarque : définition des logiciels libres.

Définition primaire : libre de droit (=gratuit) et d'usage (=accès au code source).

Aujourd'hui, cette définition tend de plus en plus à se réduire à l'aspect libre d'usage (= accès aux sources)

Un freeware est un logiciel gratuit dont on n'a pas le code source.

Un shareware est un logiciel bridé dans le temps et/ou ses fonctionnalités.

II Notion de downsizing, upsizing et rightsizing

L'évolution vers les architectures distribuées se retrace selon trois phénomènes :

Lorsque les petits ordinateurs ont pu avoir des performances intéressantes, de nombreuses applications de type micro-informatique ont été déployées. Il s'agit du phénomène de downsizing qui consiste à envoyer données et traitements sur des petites machines disséminées sur l'ensemble des sites informatisés. Aujourd'hui, cela se fait surtout pour les traitements.

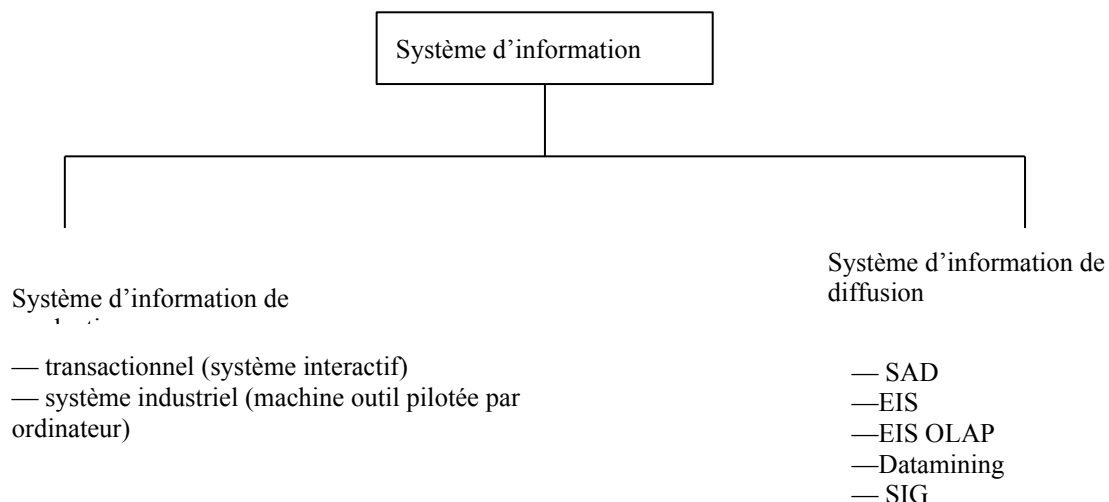
Puis, lorsque l'on a constaté que la gestion de ce genre d'architecture complètement distribuée posait des problèmes, un mouvement de re-concentration a opéré : il s'agit de l'upsizing, qui consiste à remettre, dans les architectures, des systèmes centralisés. Cela se fait tout particulièrement pour les données, qui sont souvent consolidées au niveau d'une base centrale.

Enfin, pour essayer d'optimiser au mieux les architectures, le rightsizing essaie de déterminer, grâce à des études, quels types de machines doivent être mis dans l'infrastructure.

Remarque : Le système d'information contient le système informatique.

Le système d'information modélise un ou plusieurs objets du monde réel, les propriétés de ces objets, les relations entre ces objets, les opérations sur ces objets. (Exemple d'objet : un utilisateur, un compte bancaire).

Ces dernières sont de deux natures : les opérations de lecture et les opération d'écriture. Les opération d'écriture caractérisent les systèmes d'information de production (= on génère de la donnée) ; les opérations de lecture caractérisent les systèmes d'information de diffusion (= on consulte les données).



Système d'information de diffusion (SID) : fonctionnellement, un SID (aussi appelé système décisionnel) se décompose en 4 parties :

- L'alimentation (entrée de l'information)
- Recherche (identification de l'information)
- Visualisation (rapatriement de l'information)
- Administration système

On compte 5 générations de SID, découpé en 3 familles :

1^{ère} famille

1/ le DSS ou SAD (système d'aide à la décision) : production d'état statistique papier, sans possibilité de filtre.

2/ EIS (Executive Information Service) : c'est un SAD automatisé. On peut réaliser un filtre sur un critère. Le résultat fourni est électronique.

3/ EIS OLAP : il s'agit d'un EIS dans lequel on peut faire des filtres multicritères. (C'est ce qu'on appelle un espace de recherche multidimensionnel)

2^{ème} famille

4/ Dataminig : recherche automatisée selon tous les critères possibles. Cela permet l'identification de lien entre information, non soupçonnés jusque là (exemple : organisation des magasins)

3^{ème} famille

5/ Système d'Information Géographique (SIG) : les information sont organisées selon une approche géographique (exemple de logiciel : cartabase).

Point sur l'infocentre : on parle d'infocentre à partir de l'EIS.

Un infocentre repose sur la technologie de datawarehouse. Le Datawarehouse est une infrastructure qui permet d'assurer trois fonctions de base :

- L'alimentation grâce à la modélisation des données.
- Le stockage des données, grâce à un SGBD optimisé pour ce genre d'application (ex : HP Datacenter).
- L'accès aux données, via EIS OLAP.

Dans un projet, l'objectif est la réalisation d'un infocentre ; le moyen pour le faire est le datawarehouse. L'infocentre permet de mettre un service à disposition des partenaires de l'entreprise (ou de l'administration) : ce service est un accès à une information que l'on peut valorisée et exploitée. Les clés du succès de telles structures sont la formation et l'assistance aux utilisateurs.

Datamart : entrepôt de données pour quelques utilisateurs particuliers.

Dataweb : entrepôt de données sur le web.

Comparaison entre Système d'information de diffusion et Système d'information de production.

	Système d'information de diffusion	Système d'information de production
Utilisateur	Chercheur d'information	Producteur d'informations
Mise à jour des données	Pas important	Fondamental ! (notion de temps réel)
Accès	Libre en partie	Sécurisé, utilisateur connu
Format des données	Données retravaillées (lisible)	Données brutes (codes...)
Origine des données	Externes	Internes
Besoin en données	Inconnu	Identifié
Gestion administration	Travail continu d'adaptation	Routinier

systeme		
Assistance utilisateur	Très difficile (on ne sait rien sur ex)	A besoin de réactivité, mais connu à l'avance

Le cadre présenté ci-après doit permettre de répondre aux questions générales sur la détermination d'une architecture informatique dans un examen de type qualification Analyste Développeur.

Cependant, suivant le niveau de détail (quantification, qualification) du sujet, la réponse attendue pourra être de très succincte à relativement complète (cf. sujet analyste développeur du 19 mai 2000)

Introduction

L'architecture informatique ne peut être déduite qu'après l'analyse fonctionnelle, ou plus précisément, par rapport à l'examen écrit, après la note de synthèse de présentation du système, le MCD ou tout modèle en tenant lieu et éventuellement les autres graphes ou modèles concernant les flux et les traitements.

(Pour plus de détails sur la méthode, cf. le cours « Architecture » au paragraphe « architecture applicative logique »)

Démarche

Les étapes de la démarche sont :

- ✓ **Contraintes initiales** : étude du sujet sous l'angle de considérations générales sur l'organisation de l'entreprise (siège social, centre informatique, bureaux régionaux, etc), des choix techniques stratégiques de type Schéma Directeur et d'un existant informatique.
- ✓ Synthèse macroscopique du système: il s'agit de relever dans le sujet ou de déduire des questions d'analyse précédentes des quantifications et des qualifications relatives aux trois objets : utilisateurs, traitements et données.
 1. Les **utilisateurs** : type (saisie en production, décisionnels), localisation (local, régional, national), nombre, fonctions utilisées, données manipulées
 2. Les **traitements** ou **fonctions** : liste des fonctions du système, type (temps réel ou différé), fréquence d'activation, niveau de partage de chacune des fonctions par les différents types d'utilisateurs (permet de réaliser un choix du niveau de répartition), flux maximum générés par la réalisation des fonctions par les utilisateurs par site (permet de calculer le réseau)
 3. Les **données** : volume par entité principale et volume total (permet de choisir l'outil de gestion des données et la répartition par niveau)
- ✓ **Etude des quantifications** et calculs complémentaires pour la détermination des choix d'architecture. Il peut être nécessaire de compléter les déductions d'hypothèses de travail quand rien ne permet de statuer dans le sujet.

Hypothèses raisonnables permettant de réaliser les calculs, en ne recherchant systématiquement que les valeurs maximums :

Données :

Taille moyenne d'une entité de données comprenant le nom, prénom, adresse, commentaires : 1Ko

Taille moyenne d'une image non compressée : 100 Ko

Tenir compte des durées de conservation en ligne.

Fonctions :

Temps moyen de traitement d'un dossier (enregistrement d'une demande de location par exemple) par un utilisateur : 15 minutes

Durée moyenne d'ouverture d'un service : 250 jours

Durée de travail d'une personne : 220 jours, 7 heures par jour

Ces indications doivent permettre de déterminer le nombre de postes par type de site, et ainsi le besoin de réseau local et par la suite le réseau longue distance.

Exemple de calcul : si une organisation doit traiter 400 000 dossiers par an sur 10 sites
→ $400000 / 10 = 40\ 000$ dossiers par an et par site soit 160 dossiers par jour et par site soit à 15' par dossier 2400' ou 40 heures par jour et par site, soit pour 7 heures de travail un nombre de postes de travail au minimum de 6 -> d'où besoin de réseau local de type Ethernet 10/100 BT avec hub et routeur d'accès.

Ce chiffre 6 est ensuite utilisé pour calculer le besoin de débit d'accès au réseau WAN. Considérant pour le site délocalisé un taux de simultanéité total de 6 sur des requêtes qui font appel aux objets de données les plus complexes soit par exemple 1 Ko, l'accès réseau dans notre exemple serait de $6 * 1 = 6$ Koctet soit 48 Kbit/s.

Ainsi on peut évaluer le besoin d'accès réseau WAN à un accès RTC ou un accès de base RNIS à 64 Kb/s.

Côté central par exemple, on agrège les 10 sites : un besoin de $10 * 48$ Kb/s soit 480 Kb/s qui correspond à un accès primaire RNIS ou à un accès de type LS si le site central est raccordé à un réseau à valeur ajoutée (de type OLEANE par exemple).

Résultat attendu

Un « beau » schéma peut être un plus, mais n'est pas indispensable.

N.B. : Les seuils de valeur précisés ci-après sont indicatifs : « rien ni jamais » ne remplace le bon sens et une bonne compréhension du sujet.

		Indications
ARCHITECTURE PHYSIQUE	Poste de travail	Nombre et type : PC sous Windows 9x, NT ou terminal passif
	Serveurs : web, de données et de traitements	Nombre et type par site : mini de type UNIX, Windows NT, ou gros système de type MVS, GCOS
	Réseau LAN et WAN (protocole de communication TCP/IP)	LAN : 10/100 BT de type Ethernet ou plus rarement FDDI WAN : <ul style="list-style-type: none"> • RTC si besoin de débit d'accès < 56 Kb/s ou nombre de sites très élevé • RNIS si 64 Kb/s < besoin de débit d'accès < 2 Mb/s • LS si > 2 Mb/s si pas de problème de coûts
ARCHITECTURE TECHNIQUE	Messagerie	X.400 ou SMTP/POP3/IMAP4
	Annuaire	X.500 ou DNS, et client LDAP
	Transfert de Fichier	FTP
	Web	Serveur web de type IIS de MS sur Windows NT ou Apache sur Linux
ARCHITECTURE APPLICATIVE PHYSIQUE	SGBD	SGBD micro de type Access ou Paradox, si la taille de la base est d'environ 100 Mb pour deux ou trois utilisateurs, ou SGBDR gros système de type ORACLE ou INFORMIX
	Moniteur transactionnel	Si nombre d'utilisateurs élevé (>500) répartis sur tout le territoire et si le système est fonctionnellement de type transactionnel → moniteur transactionnel de type TUXEDO, TDS ou CICS
	Répartition des données et des traitements	<ul style="list-style-type: none"> • Client/serveur de données si le système est implanté en réseau local ou si le réseau WAN est de type LS haut débit et le nombre d'utilisateurs par site est peu élevé • Sinon client/serveur WEB dite architecture 3 tiers ou multi-tiers si nombre d'utilisateurs élevé • Solution WEB-to-HOST si accès via un navigateur web à des applications sur systèmes propriétaires • Si système national de production transactionnel intensif → moniteur transactionnel et systèmes propriétaires
ADMINISTRATION SÉCURITÉ ET	Sécurité physique : réseau et système Sécurité logique : sauvegarde, organisation, etc.	Ne pas oublier si accès grand public ou par des partenaires externes.

Et si la question était, "vous présenterez les différentes architectures existantes et vous en choisirez une adaptée à l'étude"

L'architecture informatique peut se décliner à travers divers éléments, tels que par exemple, les utilisateurs, les données, les traitements ou encore les matériels.

Une architecture sera qualifiée de centralisée si 1 (ou plusieurs) de ces éléments le sont, a contrario elle sera distribuée (ou répartie). Une architecture peut donc être analysée de façon différente selon l'élément étudié. Ce qui explique que la majorité des architectures soit "mixte".

Les besoins des utilisateurs et les technologies de communication ont conduit à la quasi-disparition des architectures purement centralisées et à une diversité d'architectures mixtes ou purement distribuées.

4 typologies d'architecture cohabitent aujourd'hui :

Le modèle maître esclave, historiquement le plus ancien, qui conjugue les avantages d'une solution centralisée pour les données et traitements, et répartie pour les utilisateurs et matériels. Cette solution présente l'inconvénient majeur du trafic généré entre le serveur et le terminal passif. Par ailleurs le maître est souvent un système propriétaire afin de gérer au mieux tous les périphériques, les accès et données.

Le modèle hiérarchique, qui par rapport au précédent permet de déporter quelques fonctions de bases près de l'utilisateur et ainsi de réaliser une économie de bande passante. Ce modèle n'apporte que très peu de souplesse pour l'utilisateur, et les données ou traitements sont toujours centralisés.

Pour répondre à ce besoin et grâce à la micro informatique et à l'essor des réseaux, s'est développé le modèle Client/Serveur, avec ses différentes déclinaisons basées sur la répartition de 3 éléments (données, traitement et présentation) entre un client et un serveur. Dans ce modèle le poste de travail devient "client" c'est à dire qu'il n'est plus soumis à la seule capacité du maître, c'est lui qui initie la demande.

Le dernier modèle est dit "égal-égal", l'objectif est de tirer parti de toute la potentialité de la station de travail et d'offrir la plus grande souplesse de communication à l'utilisateur. Le poste de travail est tout à la fois client ET serveur. La contrainte de ce système est de type organisationnelle, il est difficile de maîtriser la gestion de cette architecture.

La solution client serveur est celle qui répond le mieux aux exigences du cas proposé car
(Lien avec les contraintes du sujet)

Cette solution a connu plusieurs générations :

C/S de données avec la mise en place des SGBD

C/S de traitement avec l'utilisation d'un moniteur transactionnel

C/S de données et traitement, c'est l'architecture 3 tiers

C/S Web, (n tiers) qui intègre un serveur web en "frontal" des serveurs de données ou traitement

C'est cette dernière solution qui sera retenue, en effet cette architecture permet un accès aux données depuis n'importe quelle station, hormis les restrictions mises en place par les dispositifs de sécurité, c'est du client léger dans le sens où seul un navigateur doit être installé sur la station, à la différence des autres types de C/S. Enfin, cette architecture permet d'optimiser les flux hors d'un réseau local, car seule la requête HTTP (l'affichage) circule sur le réseau public...payant.

Remarque préliminaire : les prix indiqués sont ceux du catalogue DPMA sur lequel se font les achats de l'administration centrale du Ministère des Finances, pour les postes de travail et périphériques. Les autres prix proviennent de marché DGCP.

I Poste de travail et périphériques

A/ Utilisation sédentaire

Unité centrale

Ecran

Imprimantes

Besoin spéciaux : scanners, graveurs CD, graveurs DVD...

Prix :

Unité centrale : NEC VL5 Tour AMD Athlon XP2800 512Mo 40Go CD NEC /// 510,00 €

Ecran CRT : écran cathodique 17 pouces /// 100,00 €

Ecran Plat : écran plat TFT 17 pouces /// 200,00 €

Imprimante jet d'encre portable /// 250,00 €

Imprimantes laser couleur réseau a4 r/v /// 1 050,00 €

Imprimantes laser monochrome réseau groupe de travail r/v /// 400,00 €

B. Utilisation nomade

Ordinateur portable

PDA

Prix :

portable gamme normale 256 Mo RAM 40go disque TFT 15" /// 1 031,00 €

portable gamme supérieure 512 Mo RAM 30 go disque, DVD, TFT 15" /// 1 180,00 €

C. Logiciel

Système d'exploitation : Windows XP professionnel cd Microsoft /// 28,00

Windows XP professionnel doc Microsoft /// 28,00 €

Windows XP professionnel licence maj Microsoft /// 155,00 €

Suite bureautique : office pro 2003 cd Microsoft /// 28,00 €

Office pro 2003 doc Microsoft /// 28,00 €

Office pro 2003 licence Microsoft /// 362,00 €

II Réseaux

A/ Câblage

câblage cuivre (paire torsadé) : catégories de câbles, selon les débits qu'ils peuvent assurer.

On compte 150€ la prise (connecteurs + câble derrière).

B/ Éléments d'interconnexion :

hub, commutateur, routeurs (pour un modèle type routeur d'agence 1000€)

III Serveurs

A/ Serveur à architecture X 86 (monde Intel et AMD)

utilisé pour des services web (DNS, LDAP...) de la bureautique.

Matériel (deux disques en standard, possibilité de faire du RAID – voir infra...) :

Prix :

Configuration basse : 2000 € (1 go de mémoire, 2 disques, 140 go, 1 processeur 2,8 GHz, avec carte RAID)

Configuration haute : 5000 € (1 go de mémoire, évolutif 24 go, jusqu'à 12 disques durs, évolutif quadri-processeur, avec carte RAID)

Serveur Blade : châssis, 3500 € , lame 1800 €

Système d'exploitation serveurs :

Windows server 2003, licence Microsoft /// 598,00 €

RedHat 3.0 entreprise server es standard licence linux /// 752,00 €

B. Serveur de type gros système*

*gros système Unix et Mainframe = machine à architecture propriétaire sur lesquels ne s'installent que les système d'exploitation dédié : mainframe Bull + système d'exploitation Bull : utilisé pour des travaux de masses.

Prix :

Serveur Unix mono noeud configuration pour un serveur de développement

4 proc, 200Go DD, 4Go RAM

Robotique 40 alvéoles

200 000 €

Serveur intégration Unix 3 nœuds avec l'OS (AIX)

380 000 €

Haut de gamme Unix

32 proc, 64Go RAM, 1,2To DD

1 200 000 €

Serveur ZOS 1 moteur activé

1 800 000 € (prix public)

700 000 € par moteur

Ajouter maintenance et logiciels 15%

IV Point sur la redondance

La redondance est un concept qui permet d'aller vers la haute disponibilité d'un service. Il s'agit de dédoubler les équipements, à n'importe quel niveau :

Le matériel peut être redondant : dans un serveur, on aura 2 alimentations, 2 ventilateur, pour éviter les pannes bloquantes. On aura également des disques configurés en RAID (redundant array of inexpensive disk) grâce à un élément matériel particulier, la carte RAID. La technologie RAID permet de reconstruire un disque grâce à la présence de données sur un autre disque.

Il existe différents niveaux de RAID. Le RAID 5 par exemple, consiste à répartir les données exploitables et des données appelées dites « de parité » sur l'ensemble des disques. Lorsqu'un disque tombe, ses données exploitables sont reconstruites grâce aux données de parité présentes sur les autres disques.

Cluster : il s'agit ici de relier des serveurs entre eux, de façon à ce que le service qu'ils rendent soit toujours assuré au moins par l'un d'entre eux.

Réplication de plateforme (sur le même site) : il s'agit là de répliquer le cluster que l'on a mis en place pour parer aux pannes qui bloqueraient le premier cluster.

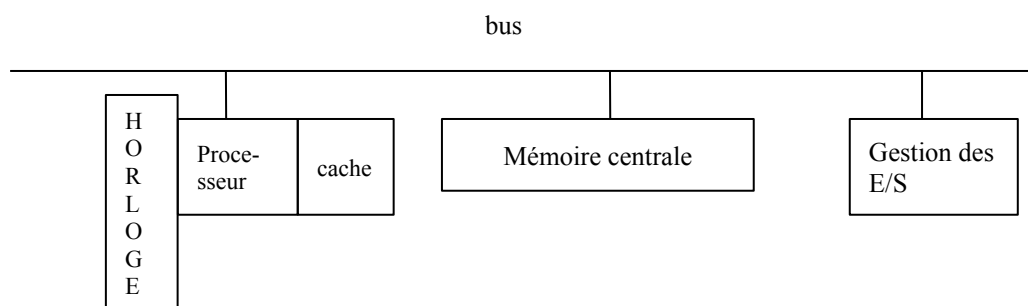
Réplication inter site : notion de site de secours, ou de back up. Ces solutions permettent de ne pas cesser l'activité même en cas de sinistre sur le site d'exploitation courant (incendie, pannes général...)

I Introduction

A. Définition du terme Système d'exploitation (SE)

- Un SE est un programme agissant comme intermédiaire entre l'utilisateur et la machine.
- Ensemble de programme conçus dans le but de faciliter l'exploitation de l'ordinateur et d'en optimiser le fonctionnement.

B. Rappel sommaire sur l'architecture interne d'un ordinateur :



Bus : permet l'échange d'informations entre les différents éléments

Processeur : unités de calcul + mémoire cache

Horloge : cadence le rythme de travail du processeur

Mémoire centrale :

Gestion des entrées/sorties (E/S) : assure la communication avec les périphériques.

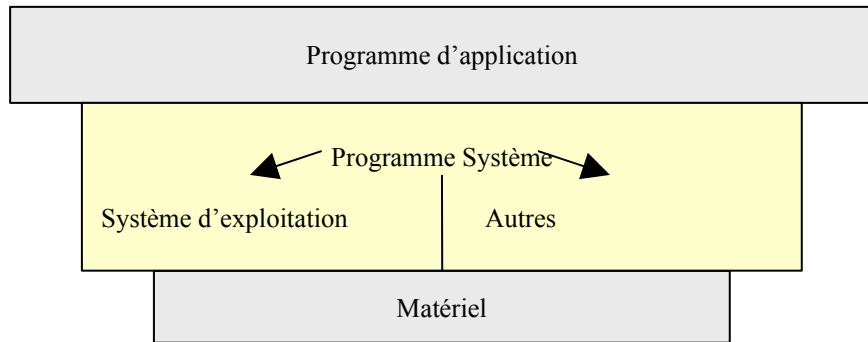
Pour faire fonctionner tout cela ensemble, il y a le Système d'Exploitation.

Le SE a deux grands rôles : masquer l'aspect matériel (gestion des entrées sorties, des interruptions, de la mémoire, de l'horloge...) et prendre en charge la gestion des ressources (à tout moment, il connaît l'utilisateur d'une ressource, contrôle l'accès et évite les conflits d'accès entre plusieurs programmes ou utilisateurs)

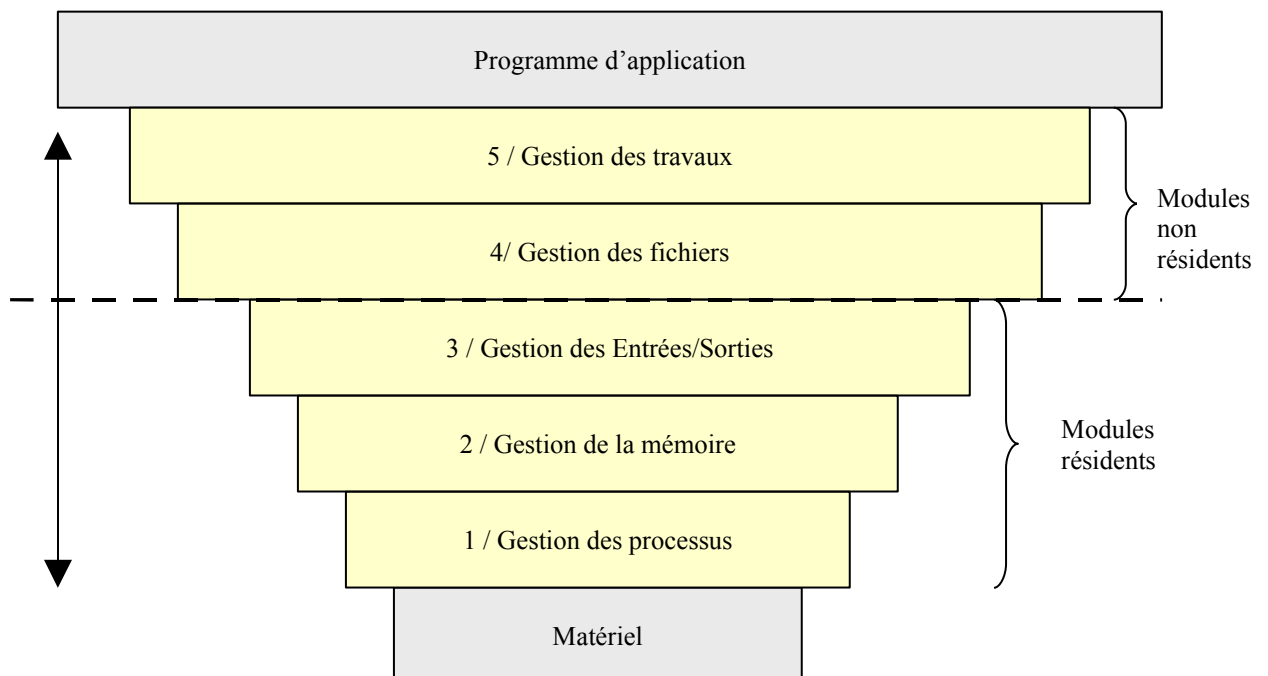
Ressources : le temps du processeur (temps CPU), la mémoire, les périphériques d'entrées-sorties... sont des ressources.

C. Place du SE dans le monde des logiciels

Il existe les logiciels d'application qui résolvent les problèmes des utilisateurs et les logiciels système. Dans ce dernier groupe il y a le système d'exploitation, et d'autres logiciels systèmes (interpréteur, compilateurs...).



On peut détailler le schéma précédent de la façon suivante :



1 + 2 + 3 : fonctions basses du SE, elles font l'interface avec le matériel.

4 + 5 : fonctions hautes du SE, interface avec l'utilisateur.

Les modules résidents sont toujours chargés en mémoire centrale (dès le démarrage). Les modules non résidents sont chargés seulement quand le besoin s'en fait sentir.

Le SE assure aussi un rôle dans sa propre sécurité (gestion des utilisateurs, permission des fichiers, possibilité de sauvegardes du système et restauration...). Cet aspect sera évoqué dans les thèmes associés.

D. Une notion de base des SE : le monotâche et le multitâche

On parle de système monotâche et de système multitâche.

Monotâche : à un temps t , une seule application et le SE sont chargés en mémoire centrale. Les systèmes monotâches sont forcément mono-utilisateur (ex : MS DOS).

Multitâche : à un temps t , plusieurs programmes sont chargés en mémoire centrale avec le SE. Le temps CPU (= la ressource qu'est le temps du processeur) est partagé entre plusieurs programmes, et ceux-ci semblent s'exécuter en même temps. Cependant, toujours à un temps t , le processeur ne peut réellement traiter qu'un seul processus : il y a donc des systèmes de gestion du temps CPU pour donner l'illusion que plusieurs programmes sont exécutés en même temps. Les systèmes multitâches peuvent être multi-utilisateur.

Cela nous amène à la première des activités du SE, la gestion des processus.

II Gestion des processus

Un programme en cours d'exécution est formalisé par la notion de processus.

A. Description de la notion de processus

Le processus est l'unité élémentaire d'exécution par le processeur.

Un processus comprend le programme en langage binaire et son contexte d'exécution (nom du processus et de l'utilisateur, répertoire de travail, liste des fichiers ouverts, zone de communication, priorité...)

Un processus peut avoir différents états : cela est lié au fait que le processeur ne peut exécuter qu'un seul processus à un temps t .

Etat des processus :

- Actif : en exécution par le processeur, il bénéficie du temps CPU.
- Prêt : il attend le temps CPU, mais tout est là pour qu'il soit exécuté.
- En attente : il n'est pas prêt à être exécuté : il attend le résultat d'une entrée/sortie par exemple.

Il existe d'autres états des processus mais ils sont pour la plupart des déclinaisons autour de ceux présentés.

L'état d'un processus ainsi que d'autres informations liées sur ce dernier (sa priorité, ses zones mémoire allouées...) sont stockés dans le Process Control Block. Ce dernier permet la restauration des processus quand le système reprend leur exécution.

B. Les interruptions

Une interruption est un événement qui arrête l'exécution d'un processus par le processeur. Les interruptions sont nécessaires pour que le processeur passe d'un processus à un autre et donc, donne l'impression d'exécuter plusieurs processus à la fois.

Exemple d'interruption :

- le processus a terminé son exécution,
- le processus a besoin d'une entrée-sortie,
- il y a un battement d'horloge,
- signal émis par un autre processus,
- appel système,
- erreur physique ou logique.

Notion d'appel système : le programme fait appel au système pour lui demander d'exécuter des routines que seul le système est autorisé à faire. Par exemple, un appel système permet au processus de communiquer avec un autre processus, de demander une entrée-sortie, une allocation mémoire...

Les appels système s'adresse à une couche logicielle appelée API (Application Programming Interface). Les API, non standards, spécifiques à chaque système, sont responsable de la non portabilité des systèmes propriétaires.

C. Traitement d'une interruption

Lors d'une interruption le système change de mode d'exécution. En effet, il existe deux modes d'exécution :

- le mode utilisateur (dans ce mode tout se passe comme s'il n'y avait qu'un seul processus en cours d'exécution, ce processus dispose de droits restreints, et est interruptible).
- le mode privilégié ou mode noyau : dans ce mode, le système peut atteindre la quasi-totalité des ressources, ce mode est ininterruptible.

Voici la séquence d'une interruption :

L'interruption survient → le système passe en mode privilégié → il arrête et sauvegarde le processus à interrompre → les routines qui peuvent traiter l'interruption prennent le control → un module système (l'ordonnanceur) prend le control : il choisit le prochain processus à exécuter par mi ceux qui sont prêts → le processus choisi est restauré pour obtenir le temps CPU → le système bascule en mode utilisateur pour exécuter le processus actif jusqu'à la prochaine interruption.

Deux modules du système interviennent lors d'une interruption : l'ordonnanceur et l'allocateur. L'Allocateur (ou dispatcher) donne le temps CPU. L'ordonnanceur (ou scheduler) attribue les priorités aux processus et les fait varier dans le temps de façon à ce que les processus qui auraient toujours une priorité haute ne monopolisent pas le CPU.

D. Comment attribuer le temps CPU ?

Pour attribuer le temps CPU, il existe deux possibilités :

- en multitâche coopératif, les processus s'exécutent les uns après les autres, sans interruption de l'horloge, jusqu'à ce qu'ils daignent rendre le temps CPU au système

d'exploitation. L'horloge n'a aucun rôle interruptif dans ce mode. Les applications 16 bits fonctionnent comme cela.

— en multitâche préemptif, le système interrompt le processus en fonction d'un quantum de temps attribué par l'horloge et le sauvegarde avec son contexte d'exécution. Dans ce cas l'horloge est une interruption. Les applications 32 bits fonctionnent dans ce mode.

E. Synchronisation des processus

Les ressources (entrées-sorties, données...) sont mises à disposition des processus par le système comme un pot commun. Donc, les processus peuvent se retrouver en conflit d'accès à une ressource. Par exemple, on peut se retrouver en interblocage si un processus P1 détient la ressource A et ne la libérera que quand B sera disponible et un processus P2 détient la ressource B et ne la libérera que quand A sera disponible.

Pour éviter cela, le système d'exploitation peut utiliser différentes méthodes comme les sémaphores.

III Gestion de la mémoire

La mémoire centrale est un élément matériel (barrette de mémoire) essentiel. C'est une ressource toujours rare (si sa taille n'a pas cessé d'augmenter, la taille des programmes aussi augmente). C'est le lieu de résidence momentanée des programmes et des données pour l'exécution. En effet, seul les instructions stockées en mémoire centrale peuvent être exécutées par le processeur.

On a vu que le processeur donnait l'impression d'exécuter plusieurs programmes en même temps. En fait, il en exécute un, puis, suite à une interruption, va en chercher un autre, dans la mémoire.

Comment est organisée la mémoire pour contenir plusieurs processus ?

A. Une mémoire découpée

La mémoire est découpée. Il y a plusieurs façons de découper la mémoire :

Partition de taille fixe de la mémoire :

La mémoire est découpée en morceaux de tailles différentes mais fixées une bonne fois pour toute. Cela est gênant lorsqu'on ne connaît pas bien les processus qui vont être exécutés : y aura-t-il un morceau de mémoire assez grand pour recevoir de gros programmes ? De plus on perd forcément de la place, puisqu'il y a peu de chance qu'un processus est exactement la taille d'un morceau... Ce système n'est intéressant que pour des ordinateurs qui font toujours tourner les mêmes processus, parfaitement connus (monde de l'industrie).

Partition de taille variable de la mémoire:

Cette méthode essaie d'optimiser l'utilisation de la mémoire et de laisser le moins de trou possible. Il y a déplacement physique des programmes dans la mémoire : de temps en temps le système suspend l'exécution et effectue un compactage de la mémoire. Mais on a toujours le problème des programmes volumineux.

On a donc eu l'idée de découper les programmes eux-mêmes en segments.

La segmentation des programmes et de la mémoire

La mémoire et les processus sont divisés en segments (= petits blocs indépendants les uns des autres). Le système se charge de placer en mémoire les segments du processus nécessaire à l'exécution de ce dernier. Il y a aussi réorganisation périodique de la mémoire afin de rassembler les segments d'un même processus dans un même espace contigu et de libérer des zones libres de mémoire plus importantes. On verra ci-dessous la méthode de pagination qui est un type de segmentation de la mémoire.

B. La mémoire virtuelle

Ce concept permet d'utiliser une partie du disque (appelée le swapfile) comme une sorte d'extension de la mémoire physique (la barrette de mémoire). Cela permet l'exécution de processus dont la taille dépasse celle de la mémoire physique disponible à un moment donné.

La plupart des systèmes à mémoire virtuelle font ce que l'on appelle de la pagination : la mémoire virtuelle est divisée en petites unités, les pages. A chaque page correspond en mémoire centrale une case mémoire. Les pages et les cases ont la même taille.

Cette méthode implique que le système doit choisir quelles sont les informations dont le processus a besoin immédiatement pour s'exécuter et qui tiennent dans l'espace mémoire disponible, et quelles sont les informations pouvant rester momentanément sur disque avant d'être appelée. Les informations feront un va et vient entre le disque et la mémoire.

Le processus commence à s'exécuter : au début, il a besoin de la page 1 et de la page 2 qui sont en mémoire centrale, mais bientôt, il va avoir besoin de la page 3 qui est restée sur le disque, dans le swapfile. Pour éviter l'interruption d'un processus, il faut anticiper ce phénomène et donc faire de la place dans la mémoire. Le système doit alors évaluer quelles pages peuvent être remplacés ou non. Pour cela, il existe plusieurs méthodes :

- remplacement d'une page non récemment utiliser
- remplacement d'une page en premier entrée premier sortie (= à l'ancienneté)
- remplacement de la page la moins récemment utilisée.

Lorsqu'une interruption d'un processus survient suite à l'absence d'une page, on parle de faute de page. Le vol de page correspond au fait de remplacer un page par une autre dont on va avoir besoin.

Pour éviter que le système ne face plus de pagination qu'autre chose, le système alloue aux processus un espace vitale (c'est-à-dire un nombre de page minimal et maximal en mémoire).

IV Gestion des Entrée/Sorties

Les entrées/sorties sont un ensemble de transferts de données entre la mémoire centrale et les périphériques.

A. Description des périphériques

Il existe deux types de périphériques :

- Les périphériques blocs qui mémorisent l'information dans des blocs de taille fixe : les disques

— Les périphériques caractères qui acceptent les informations sous forme d'un flot de caractère sans se soucier d'organisation en bloc (ex : imprimante, souris, carte réseau...)

Les périphériques, quelque soit leur type, sont attachés à l'ordinateur via une liaison physique : les canaux, ou bus. De plus, certains matériels évolués sont en plus pilotés par un contrôleur.

Exemple de périphériques : clavier, souris, imprimantes, disques durs, carte réseaux...

Les disques écrans clavier... sont très lents par rapport au processeur. Transférer les informations octet par octet entre la mémoire centrale et le périphérique serait un gaspillage de temps : pour éviter cela, il y a une mémoire tampon associée à chaque périphérique. Cette mémoire tampon est généralement une zone de la mémoire centrale. Les informations ne sont envoyées que lorsque la mémoire tampon est pleine : le bus de données est moins sollicité. On perd moins de temps.

B. Comment se fait la gestion des entrées/sorties

La gestion des entrées/sorties consiste à envoyer des commandes d'E/S aux périphériques, à intercepter les interruptions, et à gérer les erreurs éventuelles.

Le système d'exploitation s'efforce de traiter tous les périphériques de la même manière, en confinant tous les traitements particuliers dans des modules spécialisés, appelés gestionnaires périphériques (= driver = handler).

La gestion des E/S est une tâche difficile suite à :

- La diversité fonctionnelle des périphériques (disques, imprimantes),
- La diversité des vitesses,
- La diversité de codage des informations échangées,
- Et bien d'autres...

Cette gestion complexe est réalisée par le logiciel d'entrée/sortie, qui est découpé en 4 couches :

Gestion des interruptions : permet la réalisation d'une commande d'E/S sans qu'un autre processus ne puisse accéder aux périphériques pendant ce temps.

Les drivers (ou handler): gèrent la communication avec les contrôleurs des périphériques. Ce sont eux qui envoient les commandes et qui assurent le bon ordonnancement des commandes.

La partie du logiciel d'E/S ne dépendant pas du matériel : elle effectue les fonctions d'E/S communes à tous les périphériques et fournit une interface aux logiciels utilisateurs. Elle fait la liaison entre le nom symbolique du périphérique et le périphérique lui-même.

La partie du logiciel d'E/S intégrée dans l'espace utilisateur : elle gère les procédures standards utilisées par les logiciels des utilisateurs (par exemple, elle gère le spool = les files d'attente pour les impressions, les périphériques réseau...)

V Gestion de fichiers

Remarque préliminaire : ici, nous ne parlerons que de fichiers inscrits sur des supports tels que les disques durs qui sont des périphériques blocs.

La partie du système d'exploitation qui prend en charge les fichiers est le système de fichier. Un système de fichier comporte un ensemble de principe et de règle de gestion et manipulation des fichiers.

Chaque système d'exploitation a son système de fichier privilégié bien qu'il puisse en utiliser d'autre :

- MS DOS : FAT 16
- Windows 95 et 98 : Fat 32
- Windows NT, 2000, et XP : NTFS
- Linux : Ext3.

Un fichier peut être défini comme l'unité élémentaire de regroupement des informations sur une mémoire auxiliaire de stockage.

A. Manipulation des fichiers

Les opérations de manipulations des fichiers demandées par les utilisateurs sont gérées par le système. Il s'agit de :

- La création
- La destruction
- L'ouverture
- La fermeture
- La lecture/écriture

B. Mode d'accès à un fichier

- séquentiel : il faut lire tous les blocs avant le bloc souhaité pour pouvoir traiter ce dernier.
- direct : on va directement sur le bloc intéressant : il faut que le support des données soit adressable.
- indexé : création d'un index pour se positionner à un endroit X et aller, en séquentiel, jusqu'à X + n

C. Rôle de la gestion des fichiers

La fonction fondamentale d'un système de fichier est d'établir un lien entre le modèle logique des données (avec lequel travail l'utilisateur) et la réalité physique du dispositif de stockage.

On parle d'enregistrement logique (ensemble de données qui a un sens pour l'utilisateur) et d'enregistrement physique (unité de stockage manipulé par le système).

Pour établir le lien entre ces deux aspects, il y a plusieurs étapes :

- Pour retrouver les blocs dans lesquels sont inscrits des informations du fichier, il y a les techniques de la table, utilisée entre autre par Windows (dans cette table, il y a une entrée par bloc : sur bloc numéro n, il y a un bout de tel fichier, etc.) et la technique du nœud d'information (ou i-node) utilisé par Unix : le nœud contient toutes les caractéristiques d'un fichier, y compris les blocs qu'il occupe.
- Mais les numéros de blocs ne parlent pas à l'utilisateur : il veut retrouver son fichier par un nom logique : pour cela, il y a le catalogue. Ce dernier fait la liaison entre le nom symbolique du fichier et son emplacement physique sur disques. C'est dans le catalogue

qu'est défini l'organisation des fichiers : aujourd'hui, l'organisation la plus connue est l'organisation arborescente (autant de niveau de répertoire que l'on veut)

D. Gestion des blocs

La gestion des blocs par le système doit prendre en compte :

- l'organisation optimale des blocs pour améliorer le temps d'accès
- la gestion des blocs libres
- la contrôle de cohérence des blocs
- la gestion des blocs défectueux se fait par les contrôleurs de disques

VI Gestion des travaux

La gestion des travaux assure :

- La communication avec le système d'exploitation notamment par la rédaction et la formulation de la demande (interface utilisateur)
- La prise en charge de la demande (affectation des ressources et planifications)
- La comptabilité et les statistiques d'utilisation des ressources

A. La communication avec le système d'exploitation

L'utilisateur communique avec le SE par l'intermédiaire d'un langage de commande (qui est masqué aujourd'hui par les interfaces graphiques).

Séquence de demande d'un utilisateur :

Utilisateur → demande en langage de commande → interprétée par l'interpréteur de commande → qui oriente les requêtes aux services appropriés.

La plupart des langages de commande sont propriétaires car ils reflètent la structure interne de l'ordinateur. Pour Unix, système ouvert par excellence, ce n'est pas le cas : on peut remplacer son langage de commande (le shell) par un autre. Dans ce cas, le langage de commande est traité comme externe au système.

B. Prise en charge des demandes

La prise en charge des demandes recoupe la gestion des processus avec le rôle fondamental de l'allocateur et de l'ordonnanceur.

C. Comptabilité et statistiques

A partir des données d'allocation des ressources, le système peut réaliser des statistiques sur l'utilisation des ressources.

VII Thèmes associés

A. Classification des Systèmes d'exploitation

- Ouvert / propriétaire : un système ouvert est portable sur différentes machines, indépendamment de leur architecture matériel. Unix en est l'exemple type. Un système propriétaire est spécifique à une machine ou une gamme de machine : MVS pour les gros systèmes IBM, GCOS pour les gros système Bull.

Limite de cette classification : Unix, récupéré par les grands constructeurs s'est déclinés en version d'Unix propriétaires, comme AIX, chez IBM, Solaris chez Sun...

- Selon la famille d'ordinateur auxquels le SE est destiné :

Les micro-ordinateurs : MAC OS, Windows 95/98, Windows 2000, Linux

Les mini-ordinateurs : AS 400 d'IBM

Les gros systèmes (aux caractéristiques bien particulières : multiprocesseurs, exploitation en mode batch, plusieurs centaines d'utilisateurs simultanés...) : MVS, GeCOS, UNIX et ses déclinaisons

B. Evolution des Systèmes d'Exploitation

Très logiquement, les systèmes d'exploitation ont suivi les évolutions de l'informatique et en particulier, avec l'accroissement des réseaux, de nouveaux types de SE sont apparus : on parle de NOS (network operating System).

Les **NOS** partagent les caractéristiques des systèmes plus classique et se voient doter de caractéristiques particulières :

- sécurité : la sécurité sur le réseau consiste à s'assurer que celui qui modifie ou consulte des données du système en a l'autorisation et peut le faire correctement car il en a la possibilité. Pour cela, on a des systèmes d'authentification (mots de passe), des systèmes de fichier gérant les droits d'accès, la possibilité de surveiller les menaces (par exemple par l'enregistrement de mot de passe invalide), la liste des contrôles (qui accèdent à quoi quand...)

- accès aux ressources distantes : cela se fait via le protocole RPC (Remote Procedure Call)

- système de fichier réparti : il s'agit d'un système de fichier dans lequel les clients, serveur et unité de stockages sont dispersés, malgré cela il présente un système de fichier centralisé à l'utilisateur. Exemple : nfs.

Les NOS mettent en jeu des systèmes clients et serveurs. Les systèmes d'exploitation serveur mettent à disposition des services pour les machines clientes. Ils acceptent un nombre illimité de connexion.

Support des multiprocesseurs : les machines peuvent aujourd'hui avoir plusieurs processeurs, d'où une gestion plus complexe de l'activité du processeur. Les systèmes d'exploitation doivent pouvoir supporter ce type d'architecture matériel.

Support du clustering : un cluster est l'ensemble de plusieurs serveurs.

Evolution du moniteur transactionnel : il gère les accès multiples des utilisateurs aux ressources et se greffe sur le SE tout en le déchargeant de ces tâches.

C. Vocabulaire

Ouvert = portable + interopérable

Portabilité : capacité pour un logiciel (système d'exploitation ou autre) de fonctionner sur différentes plates-formes matériels.

Interopérabilité : capacité des différents systèmes à travailler ensemble

Libre : libre de droit et d'usages : c'est-à-dire que l'on a accès au code source (avant compilation) et que l'on peut le modifier, le tout gratuitement. Attention, aujourd'hui, le monde « libre » est de moins en moins gratuit : des éditeurs vendent des services annexes (formation, support) et des développements particuliers : exemple RedHat qui diffuse le noyau linux entouré de ses propres développements.

Freeware : c'est un logiciel gratuit mais dont on n'a pas les sources.

Shareware : version bridée d'un logiciel (soit dans le temps, soit en ce qui concerne ses fonctionnalités). Il faut payer pour avoir la version complète.

Fiche thème 8	Les réseaux
---------------	-------------

Fiche thème 8-1	Le modèle OSI
-----------------	---------------

Le modèle OSI a été conçu pendant les années 70 par l'ISO (international standard organisation). Il s'agit d'un modèle qui ne traite que de communication entre système, quelque soit les systèmes.

Au début de l'informatique, les mondes des différents constructeurs (BULL, IBM, DEC...) ne pouvaient communiquer entre eux que grâce à des passerelles. Cependant, pour des raisons de positionnement sur le marché informatique, le développement de telles solutions n'étaient pas une priorité pour les constructeurs.

Pour autant le besoin en communication entre les différents systèmes devenait réel. Le modèle OSI va permettre de créer un cadre pour le développement de solution de communication entre systèmes hétérogènes.

I Les couches

A. Présentation du modèle

L'objectif de ce modèle est de fournir une base commune pour le développement de standard de communication. Il s'agit de fournir une référence pour encadrer les développements ; suivre la référence qu'est le modèle OSI doit permettre la communication entre système hétérogène.

L'organisation ISO est composée de pays membres. Elle est ainsi indépendante face aux constructeurs et éditeurs. Cela lui permet de développer des normes applicables à tous les environnements.

Remarque :

Norme : vient d'un organisme reconnu et officiel qui édite des règles, comme l'ISO ; la norme a pour but de devenir un standard.

Standard : quelque chose qui est adopté par tout le monde, sans pour autant avoir été forcément produit par un organisme de normalisation.

Le modèle OSI décline toute opération de communication en 7 étapes qui sont les 7 couches du modèle OSI.

B. Les couches du modèle OSI

Couche Physique : elle s'occupe de l'établissement de la communication, de faire passer les flots de bits sur le média (elle gère des influx électriques, de la connectique...)

Couche Liaison : elle contrôle les erreurs éventuels de la couche physique et organise l'accès au média (j'envoie tout le temps de l'info, pas tout le temps ?)

Couche Réseau : elle choisit et réserve la route (les paquets de données passeront par ici, puis ici...).

Couche Transport : elle assure la fiabilité du transport et le contrôle de l'information de bout en bout.

Couche Session : elle gère le dialogue entre deux entités communicantes : établissement de la connexion, transfert des données, fin de la connexion.

Couche Présentation : elle présente les données échangées (structure, format).

Couche Application : elle gère la partie communicante de l'application de l'utilisateur.

II Principe de fonctionnement du modèle

A. Protocole et interface

Fonctionnement du modèle : chaque couche rend des services à la couche précédente.

Ce travail s'organise via des interfaces. Les interfaces sont des verbes pour dialoguer entre deux couches différentes.

Les protocoles sont des règles de communication entre couches de même niveau.

On utilise une passerelle si les protocoles utilisés par deux couches d'un même niveau sont différents.

B. Fonctionnement

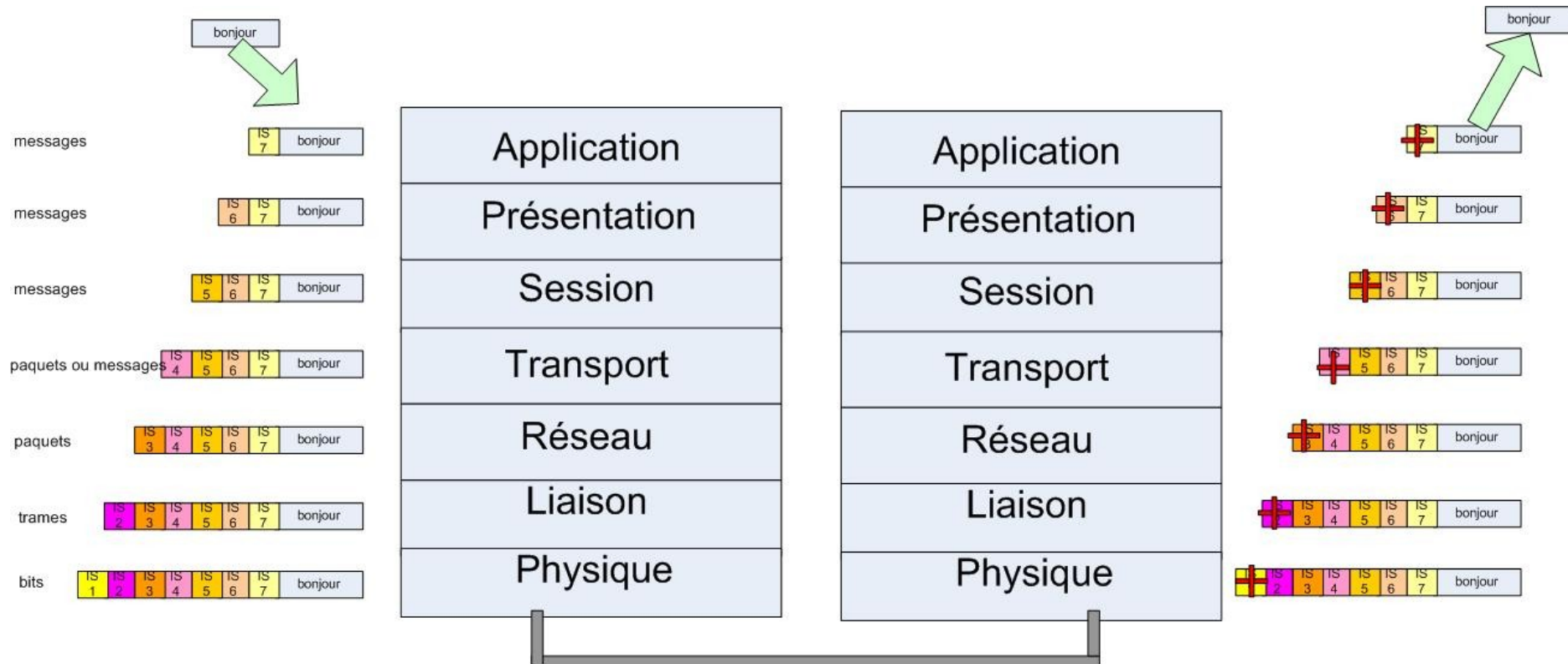
Chaque couche rajoute au message initial une information de service, dite information de service de couche 7 pour la couche 7, de couche 6 pour la couche 6...

Ces informations de services contiennent entre autre les informations liées aux protocoles qu'utilise la couche et aux services que peuvent demander les couches.

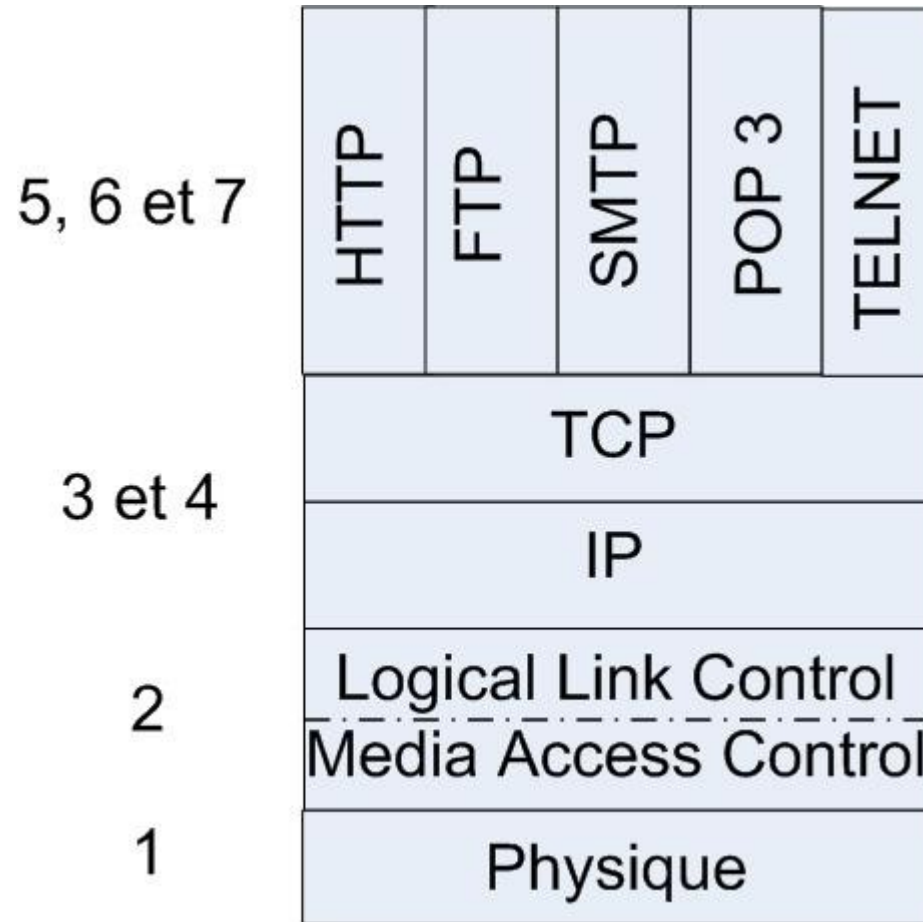
Pour l'émetteur, chaque couche rajoute une information de service, en partant de la couche 7 vers la couche 1. Le récepteur reçoit le message au niveau de la couche 1 : chaque couche, à partir de cette couche lit l'information de service qui lui correspond, puis va l'enlever de façon à « passer » à l'application le message initial sous une forme qu'elle peut comprendre.

Conclusion : le modèle OSI a 20 ans, pour autant est-il périmé ? Non, il reste la seule référence en matière de communication, et se retrouve, très adapté, à travers le modèle TCP/IP

Schéma de fonctionnement du modèle OSI



Evolution du modèle OSI : adaptation au standard TCP/IP



L'association de l'informatique et des télécommunications a conduit à l'apparition des réseaux.

I Typologie des réseaux

A. Généralités

Un réseau téléinformatique est composé de nœuds qui constituent des unités de traitement de l'information.

Ces unités de traitement échangent de l'information par l'intermédiaire des liens qui relient les nœuds et qui sont des canaux de transmission.

En pratique, les nœuds peuvent être des ordinateurs ou des équipements terminaux (écran/clavier, imprimante ...) et les canaux de transmission sont les câbles, ou les ondes, dans le cadre d'un réseau sans fil.

Un réseau d'ordinateurs est par conséquent un ensemble d'ordinateurs (et d'équipements terminaux), géographiquement dispersés, reliés entre eux par un ou plusieurs liens afin de permettre les échanges d'informations.

Les ordinateurs d'un réseau peuvent appartenir à diverses catégories allant du super-ordinateur (Mainframe, gros système Unix...) jusqu'au micro-ordinateur et évidemment être de marques différentes.

Un réseau a pour objet d'offrir un certain nombre de services à ses utilisateurs, basés sur l'échange d'informations à distance :

- accès à des informations (programmes, données) stockées sur d'autres ordinateurs du réseau ;

- accès à d'autres ordinateurs (par exemple, un super-ordinateur ou un ordinateur spécialisé) ;

- permettre l'échange d'informations entre les utilisateurs, comme par l'intermédiaire des messageries électroniques.

B. Les types de réseaux : LAN, MAN, WAN

Suivant le diamètre d'un réseau, c'est-à-dire l'éloignement maximal entre les nœuds, on peut le classer (par ordre croissant de capacité) dans une des catégories suivantes :

1) Réseau local (LAN: Local Area Network)

Un LAN est un ensemble d'ordinateurs appartenant à une même organisation et reliés entre eux par un réseau dans une aire géographique restreinte. Un réseau local est donc un réseau sous sa forme la plus simple, la vitesse de transferts de données d'un réseau local peut s'échelonner entre 10 Mb/s et 1000 Mb/s.

2) Réseau métropolitain (MAN: Metropolitan Area Network)

Il s'agit d'un réseau dont les noeuds se situent dans la même métropole. Les fibres optiques sont souvent utilisées pour la réalisation d'un tel réseau (exemple ADER : réseau reliant les administrations centrales, sur Paris).

Il peut arriver que l'on veuille relier deux LAN entre eux sans que la vitesse ne soit affectée. Pour relier des LAN géographiquement éloignés, il est possible d'utiliser un MAN. Les MAN utilisent des lignes téléphoniques spécialisées (ou bien des équipements spéciaux) dont le taux de transfert est équivalent à celui d'un LAN, sur des distances importantes. Un MAN permet ainsi à deux LAN distants de communiquer comme s'ils faisaient partie d'un même réseau local. Toutefois, les lignes qu'utilise le MAN sont totalement différentes de celles d'un LAN, car elles permettent de transmettre des données sur de très grandes distances, c'est la raison pour laquelle le coût d'un MAN est considérablement supérieur à celui d'un LAN.

3) WAN: Wide Area Network

Il s'agit d'un réseau dont les noeuds sont géographiquement très éloignés les uns des autres (plusieurs centaines ou milliers de kilomètres). Ce type de réseau utilise généralement les réseaux publics (les lignes téléphoniques par exemple) ou des liaisons louées (lignes spécialisées...).

On recourt à des WAN lorsque les distances entre deux LAN deviennent trop importantes pour les relier à leur vitesse de transfert. L'accès au WAN est limité en terme de vitesse de transfert à cause des lignes téléphoniques qui représentent un goulet d'étranglement, étant donné que leur débit est limité à 56 kb/s. On est alors bien loin des 10 ou 100 Mb/s du LAN. Même les lignes spécialisées des opérateurs téléphoniques ont une bande passante qui n'excède pas 1,5 Mb/s.

Les WAN fonctionnent grâce à des routeurs qui choisissent le trajet le plus approprié pour atteindre un nœud.

II Principes des réseaux

Plusieurs éléments permettent de décrire les réseaux :

A. La topologie

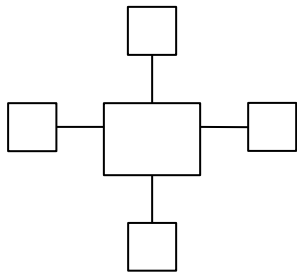
Il s'agit de la représentation géographique du réseau

On dénombre deux types de topologies : le bipoint et le multipoint.

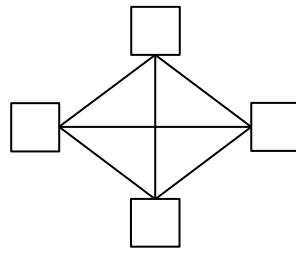
1) Le bipoint

il y a une liaison entre un émetteur et un récepteur.

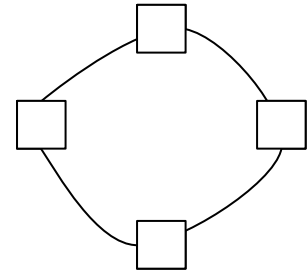
Exemple de réseau bipoints :



Réseau en étoile



Réseau maillé



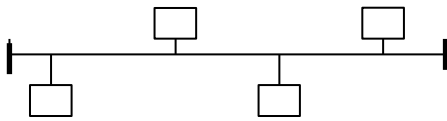
Réseau en boucle

Les réseaux WAN ont des topologies bipoints.

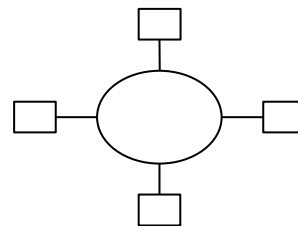
2) Le multipoint

il y a un émetteur et n récepteurs, connectés sur une seule ligne.

Exemple de réseau



Réseau en bus



Réseau en anneau

Les réseaux LAN ont des topologies multipoints.

B. Le signal et le câblage

1) le signal

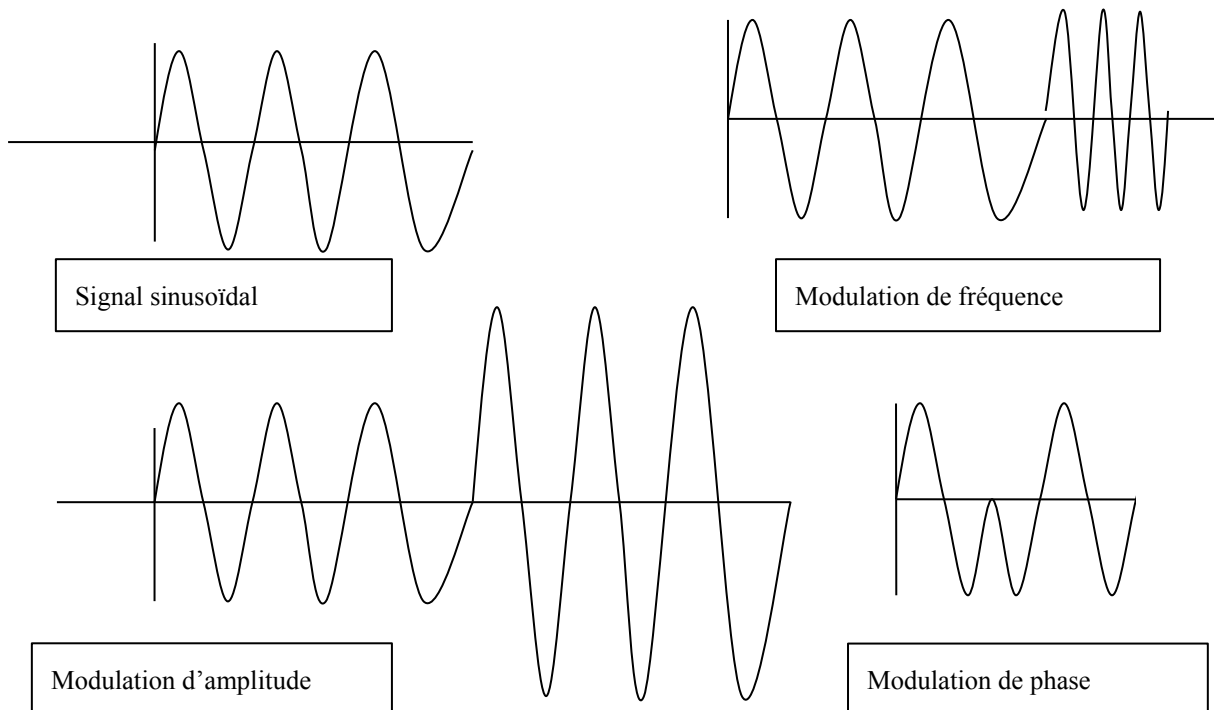
a) Type de signal : Signal sinusoïdal et Signal numérique

Le signal est la façon dont sont transportés les bits, électriquement.

On est donc ici au niveau le plus bas des couches OSI (couche 1). Il s'agit de coder des 1 et des 0, sous forme électriques. Le signal électrique peut être, entre autre, sous forme sinusoïdale ou sous forme carrée.

Un signal sinusoïdal se transporte mieux qu'un signal carré. On préférera donc cette forme pour transporter les bits sur des réseaux longues distances (bien que des solutions pour transporter le signal sous forme carré existent et se développent).

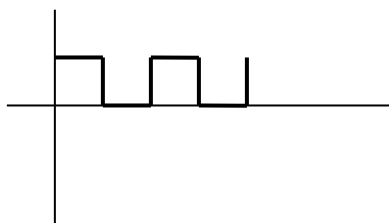
Pour coder les 1 et les 0 en signal sinusoïdal, on fait intervenir la notion de modulation : le signal change selon qu'il exprime un 1 ou un 0. Il existe la modulation d'amplitude, la modulation de fréquence, la modulation de phase.



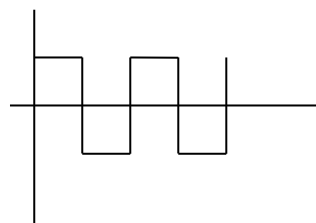
Notion de bande passante

La bande passante est la plage de fréquence disponible sans déformation du signal sur un support de communication. Elle s'exprime en Hertz : Hz (équivalent de 1/ seconde). Plus la bande passante est large, plus le débit (en bit par seconde) peut être élevé. Elle s'applique aux signaux sinusoïdaux.

Le signal carré est aussi dit numérique : il est utilisé sur les réseaux locaux, à l'intérieur de l'ordinateur. Le signal carré peut être utilisé sous la forme dite de Non retour à Zéro, qui permet d'identifier les problèmes de transmissions sur le réseau.



Signal carré



Signal carré, non retour à zéro

b) Transmission

— Parallèle et Série

La transmission en parallèle améliore le débit, mais implique une visibilité des pannes réduites (ex : les anciennes imprimantes, les disques IDE). La transmission en parallèle consiste à envoyer des données simultanément sur plusieurs canaux (fils). Les ports parallèle présents sur les ordinateurs personnels permettent d'envoyer simultanément 8 bits (un octet) par l'intermédiaire de 8 fils.

La transmission en série permet de visualiser les pannes directement (ex : les vieilles souris séries, les disques S-ATA avec plusieurs fils !). Le terme série désigne un envoi de données via un fil unique: les bits sont envoyés les uns à la suite des autres. Les ports séries bidirectionnels ont donc besoin de deux fils pour effectuer la communication.

— Synchrones et asynchrones

Synchrone : émetteur et récepteur adoptent la même unité de temps pour découper les octets et reconnaître ce qui est transmis. Devant le premier octet, 3 bits de synchronisation sont insérés, ce qui permet aux deux éléments de se caler.

Asynchrone : émetteur et récepteur n'ont pas la même unité de temps ; pour identifier le début et la fin de chaque octet, deux bits start et un bit stop sont insérés de part et d'autre de l'octet.

— Sens de la transmission : simplexe, half duplex, full duplex.

Le mode simplexe : il est unidirectionnel (ex : principe de la radio) $E \rightarrow R$

Le mode Half Duplex : les deux sens de communication sont autorisés (dialogue possible) mais chacun à son tour (principe du talky walky) $A \rightleftarrows B$

Le mode Full duplex : communication dans les deux sens en même temps. $A \leftrightarrow B$

Application en informatique : ces notions se rencontrent lorsqu'on paramètre une carte réseau. Par exemple, les cartes de stations de travail sont paramétrées en half duplex, et les cartes réseaux de serveur sont en full duplex.

2) le câblage et connecteur associé (couche 1 du modèle OSI)

a) Câble de cuivre

Composés de 4 paires de fils torsadés.

Avantages : pas cher, facile à installer

Inconvénients : sensibles aux perturbations électriques et magnétiques. Pour éviter cela on a créé des câbles en cuivre blindés les STP (Shielded twisted pair) en opposition au UTP (Unshielded twisted pair).

Les câbles en cuivre se répartissent en 3 catégories :

Catégorie	Débit
Catégorie 3	10 Mb/s
Catégorie 5	100 Mb/s
Catégorie 6	> 100 Mb/s

Ainsi qu'en trois classe :

Classes	Fréquence max	Longueur max	Utilisé pour
A	100 kHz	3 km	Téléphone
B	1 MHz	700 m	RNIS
C	16 MHz	160 m	Ethernet

Connecteur associé : RJ 45 pour les câbles réseaux, RJ 11 pour le téléphone

b) Câble coaxial

Avantage : moins sensible aux perturbations électromagnétiques que le câble en cuivre.
Longueur : 1 km, avec débit de 10 mb/s ; il a une meilleure résistance au passage du courant électrique.

Inconvénient : un peu plus cher que le cuivre, et plus difficile à connecter.

Connecteur associé : BNC (british navy connector)

c) Fibre optique

Avantage : beaucoup plus rapide que les autres types (on va jusqu'au gigabits/s) et aussi plus sûr, car il est impossible de s'y connecter directement.

Inconvénients : difficile à poser (contrainte due à la lumière)

Utilisée pour les backbone (ossature) des réseaux.

Connecteur associés : modèle ST (même genre que BNC)

Modèle MTRJ (même genre que RJ 45)

d) Réseau sans fil

— réseau local : le Wi-Fi (une borne est présente dans la salle, et les ordinateurs sont munis d'antennes) et l'infrarouge

— radio messagerie : transmission de message court sans fil. (Tamtam, bipper)

— réseau 3RD : réseau radioélectrique de données : utilisés dans le domaine des transport (SNCF)

— radiotéléphonie :

- cordless telephone 2^{ème} génération (premier téléphone portable, le bibop fonctionnait sur ce protocole)
- GSM (global System mobil) : transporte la voix et les données ; téléphone portable.
- GPRS : peut aller jusqu'à 3 fois plus vite que le GSM (téléphone portable)
- UMTS : a pour objectif d'atteindre les 2 Mb/s
- I-pode : a atteint les 2 Mb/s avant l' UMTS

Ne pas oublier les systèmes de communication par satellite.

C. Les éléments d'interconnexion

1) le segment

c'est le réseau ou la portion de réseau qui n'intègre pas d'élément d'interconnexion

2) le répéteur

il permet de prolonger un segment quand on a atteint la longueur maximale du câble. Il copie les données bit à bit, c'est donc un élément de couche OSI 1.

3) le pont

il connecte plusieurs segments ensemble ; il a pour objectif d'étendre un segment (plusieurs segments reliés par un pont forment un « segment étendu »), d'augmenter le nombre de stations sur un segment et de réduire le trafic. C'est un élément de couche 2. Dans un LAN, le pont est appelé hub. Le pont est capable de faire du filtrage sur les adresses MAC (adresses qui identifient un composant réseau : dans un ordinateur, c'est la carte réseau qui a une adresse MAC, cette adresse est composée d'un numéro identifiant le constructeur et d'un numéro identifiant le matériel lui-même). Le pont fait de la redirection au niveau des trames.

4) le routeur

il est utilisé pour relier des segments étendus mais surtout pour connecter des réseaux qui utilisent des protocoles de couches 3 différents (par exemple IP et X 25). Plusieurs segments étendus reliés par un routeur forment un réseau étendu. Pour diriger les données vers il travaille avec des adresses de niveau 3, comme l'adresse IP. Le routeur fait de la redirection au niveau des paquets. Le routeur exécute ses fonctions de routage grâce à des composants logiciels, ce qui le rend lent à traiter l'information qui transite par lui.

5) la passerelle

élément des couches hautes de l'OSI qui permet de faire dialoguer des systèmes ayant des groupes de protocoles session-présentation-application différents : par exemple, pour une application de messagerie, la passerelle doit pouvoir faire communiquer une entité ayant choisi la norme X 400 pour sa messagerie, et une entité ayant choisi le standard SMTP.

6) le multiplexeur

composant réseau qui acceptent en entrée n lignes et qui a en sortie une ligne dont le débit est égale à la somme des débits des lignes en entrée. (utilisé pour accéder au host dans un système centralisé, c'est à dire en entrée du site central)

7) le concentrateur

composant réseau qui acceptent en entrée n lignes et dont le débit en sortie est inférieur à la somme des débits des lignes en entrée (utilisé pour sortir vers un réseau d'opérateur de Télécom, c'est à dire en sortie d'un site central)

8) le commutateur

composant réseau de niveau 3 comme le routeur, il est géré par électronique et non par logiciel, et est plus rapide pour traiter les informations qui transitent par lui que le routeur. Il offre une large bande de réseau, et permet de segmenter le réseau local en VLAN (Virtual Local Area Network). Les VLAN permettent une sécurisation des réseaux et une diminution de l'encombrement du réseau en limitant le passage des messages en diffusion broadcast. Nb : il existe aussi des commutateurs de niveau 2.

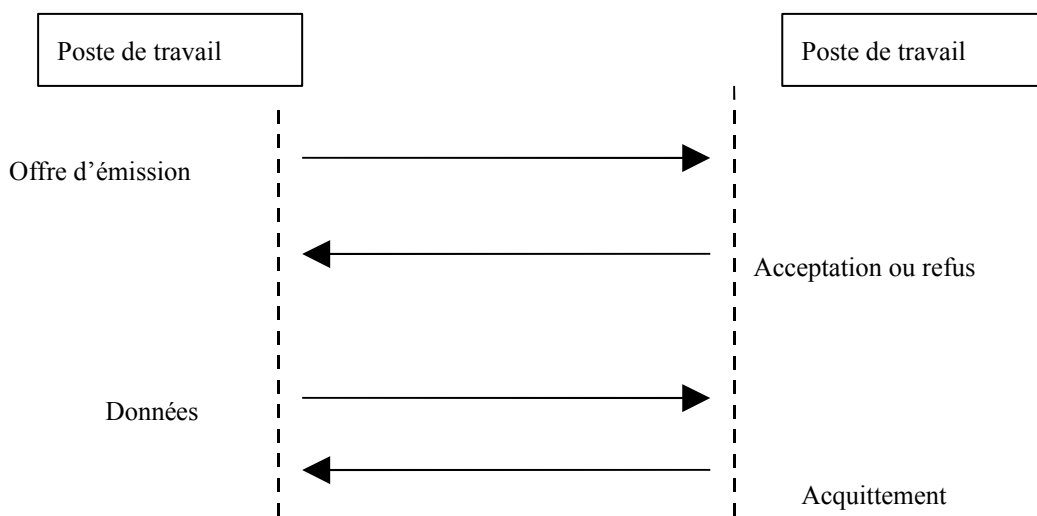
I Introduction au réseau local

A. Méthode d'accès

En matière de LAN, les méthodes d'accès sont dites par « contention » : tous les ordinateurs utilisent le même support physique pour communiquer.

De plus, ces méthodes sont dites d'égal à égal, (peer to peer) : chaque station a les mêmes droits pour accéder au support de communication

B. Schéma d'échange des données



II Types de réseau local

A. Ethernet

Ethernet est une norme de réseau local.

La méthode d'accès utilisée est CSMA/CD (carrier sense multiple access with collision detection).

Le principe de cette méthode d'accès est simple : avant de parler, l'émetteur écoute : s'il n'y a rien sur la ligne, il parle, si quelqu'un parle déjà, il attend. De plus, s'il génère une collision, il s'arrête.

CSMA/CD est dite non déterministe et aléatoire. Non déterministe signifie qu'on ne peut pas donner de priorité à une machine sur le réseau : toutes les machines ont les mêmes droits pour parler. Cela peut-être ennuyeux pour un serveur par exemple, qui doit logiquement plus envoyer d'information qu'une simple station.

Ethernet est la norme la plus utilisée pour les réseaux locaux : il représente plus de 90 % des installations.

CSMA/CD, qui le sous-tend, est un protocole rapide, mais qui présente des limites lorsqu'on atteint un certain nombre de stations sur le réseau local. Il peut y avoir rapidement dégradation à cause d'une augmentation des collisions de trames sur les réseaux.

Pour éviter ce genre de problème, on segmente les LAN en VLAN.

La topologie d'un réseau Ethernet est une topologie en Bus.

De plus, il existe plusieurs type de réseaux Ethernet, décrit selon l'expression : nBx ou $n =$ débit du réseau en Mb/s, $B =$ la modulation de base, et $x =$ le type de câblage. Les tableaux suivants résument les différentes valeurs que peuvent prendre n et x :

n en Mb/s	Non du réseau Ethernet
10	
100	Fast Ethernet
1000	Gigabit Ethernet

x	Type de câble
5	Coaxial, 1,5 de diamètre
2	Coaxial, 0,5 diamètre
T	Paire torsadée
F	Fibre Optique

Les types les plus courants sont : le 10BT et 100BT. Le 1000BT tend de plus en plus à se généraliser.

La norme Ethernet indique que le nombre maximal de station de station par segment est de 100

La longueur du réseau Ethernet est limité à 100 m (il existe des évolutions de cette norme)

En fibre optique cette dernière contrainte est levée.

B. Les réseaux en anneau

1) Anneau à jeton Token ring

Les réseaux locaux en anneau repose sur une méthode d'accès de type Token ring et une topologie en anneau.

Le principe de la méthode d'accès Token ring repose sur un jeton : lorsque la premières stations s'allument, elle envoie sur le réseau un « jeton » (une trame particulière). Lorsque une station A veut parler à une station C, elle colle les données qu'elle veut envoyer à C sur le jeton. Seul C, destinataire de l'information, peut décharger le jeton.

En somme les stations ne peuvent émettre que lorsqu'elles arrivent à récupérer le jeton vide.

Token Ring est paramétrable : on peut donner des priorité à l'émission à des stations, en leur offrant une fenêtre de temps pour émettre plus importantes que celles laissées aux autres stations du réseau. Ce protocole permet d'éviter les problèmes de collision de trames sur le réseau. Cependant, c'est aussi un protocole moins rapide qu'Ethernet.

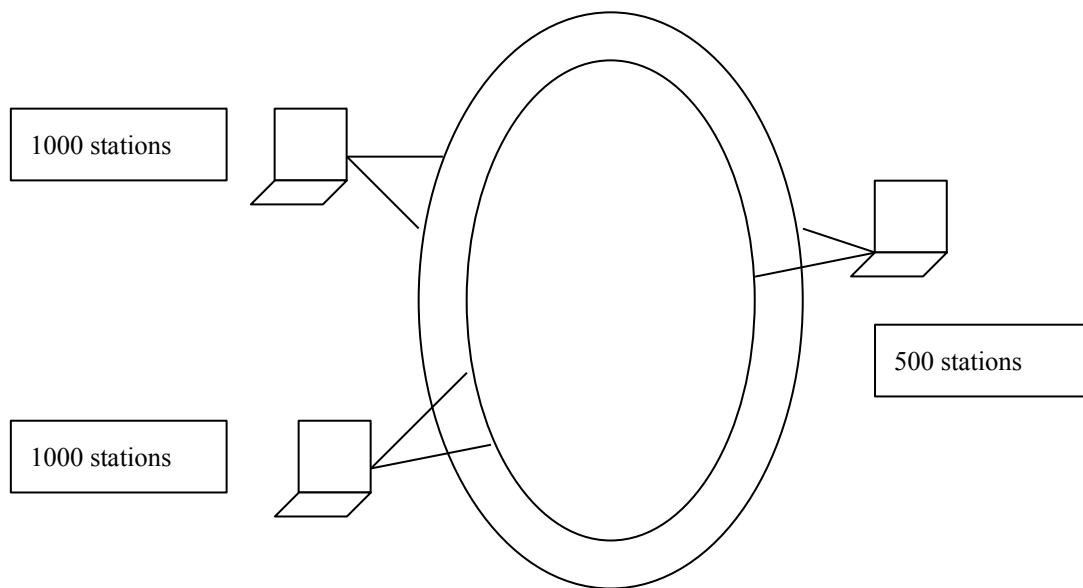
Un anneau à jeton accueille au maximum 260 stations (théoriques).

De tels réseaux représentent moins de 10% des réseaux locaux installés.

2) FDDI

FDDI (fiber distributed data interface) est une évolution de Token ring. Il s'agit d'un double anneau en fibre optique, capable d'accueillir jusqu'à 2500 stations.

Le FDDI permet une sécurisation du réseau grâce à la redondance de l'anneau. De plus, il permet des vitesses importantes et évitent les collisions.



TCP/IP est une grande famille de protocole, communément appelé, protocole de l'Internet. Lorsqu'on parle de cette famille, on entend aussi bien les deux principaux protocoles IP et TCP que les protocoles de même niveau (couche 3 et 4) UDP, ARP... et des protocoles de couches supérieures SMTP, FTP...

Ici, nous aborderons les protocoles IP et TCP ; tout d'abord pour comprendre comment fonctionnent ces protocoles sur le réseau, nous évoquerons la notion de commutation.

I Notion de commutation

La commutation est la **manière** d'acheminer les informations sur un réseau.

A/ Commutation de circuit

On ouvre un circuit et on fait tout passer par ce circuit (= le téléphone)

On construit la liaison physiquement, c'est-à-dire qu'on l'établit nœud pas nœud :

— le RTC en est l'exemple type (réseau téléphonique Commuté = 56 kbits/sec)

Il faut un débit de transmission le plus constant possible.

— Les liaisons spécialisées ou louées utilisent aussi la commutation de circuit : les débits vont de 64 kb/s à 2 Mb/s

— le RNIS (englobe les 3 couches basses, avec en protocole de couche 2 HDLC) fait aussi de la commutation de circuit.

— bien sûr, les DSL, qui circulent sur le réseau RTC font aussi de la commutation de circuit.

B/ Commutation de message

exemple, le télégramme

C/ Commutation de paquets

On fait de la commutation de paquets lorsque chaque nœud réceptionne le paquet et le commute à un autre nœud.

Il existe deux méthodes de commutation de paquets : le mode Datagramme et le mode circuit virtuel.

Mode datagramme : chaque paquet peut prendre une route physiquement différente.

Les paquets passent par des lignes en fonction de l'encombrement. Le récepteur doit reconstruire le message, parce que les paquets sont arrivés dans le désordre ; TCP/IP fonctionne en mode datagramme.

Mode circuit virtuel : au début de la communication, il y a réservation des ressources pour la transmission. Tous les paquets qui constituent le message utilisent le même circuit.

On conserve l'ordre des paquets, pas de perte de temps pour recomposer le message. La vitesse d'accès peut-être différente aux deux bouts. La longueur des paquets dépend de l'accès, c'est à dire que c'est l'opérateur de télécoms qui va fixer la longueur des paquets. Le trafic n'est pas optimisé. X25 est un protocole orienté circuit virtuel.

L'offre France Télécom de commutation de paquets était TRANSPAC.

Qualité de Transpac : fiable, maîtrisée depuis longtemps, accessible via une ligne RTC ou RNIS, mais vieux protocole, très verbeux, adressage complexe et surtout protocole franco-français.

D/ Commutation de Cellule

C'est du frame relay avec des trames de taille fixe, appelées cellules.

Asynchronous Transfer Mode : ATM

Cellule de 48 octets.

II Principes de IP

A/ Description fonctionnelle

Le rôle du protocole IP est d'acheminer les datagrammes à travers un ensemble de réseaux interconnectés.

Ceci est réalisé en transférant les datagrammes d'un « module Internet » à l'autre jusqu'à atteindre la destination. Les modules Internet sont des programmes exécutés dans des hôtes et des routeurs du réseau Internet. Les datagrammes sont transférés d'un module Internet à l'autre sur un segment particulier de réseau selon l'interprétation d'une adresse. De ce fait, un des plus importants mécanismes du protocole IP est la **gestion de cette adresse Internet**.

Lors de l'acheminement d'un datagramme, les datagrammes peuvent éventuellement avoir à traverser une section de réseau qui admet une taille maximale de paquet inférieure à celle du datagramme. Pour surmonter ce problème, un mécanisme de **fragmentation** est géré par le protocole IP.

B/ Adressage

Une distinction doit être faite entre noms, adresses, et chemins. Un nom indique ce que nous cherchons. Une adresse indique où cela se trouve. Un chemin indique comment y aboutir.

Le protocole IP s'occupe essentiellement des adresses. C'est à des protocoles de niveau plus élevé que revient la tâche de lier des noms à des adresses. Une adresse IP d'une machine se connectant directement à Internet doit **toujours être unique**. L'affectation d'adresse IP unique est gérée par le ICANN (Internet Corporation for Assigned Names and Numbers). Les adresses IP dites privées, qui sont affectées aux machines des réseaux d'entreprise ne sont pas « vues » sur Internet : plusieurs réseaux privés peuvent donc avoir des adresses similaires. L'adresse qui sera unique sera celle de la porte de sortie de ces réseaux vers Internet de ces machines (le Proxy).

1) Classe d'adresse

Les adresses IP version 4 ont une longueur fixe de 4 octets soit 32 bits*.

Une adresse commence toujours par un numéro de réseau, suivi d'une adresse locale codant l'adresse de la machine sur ce réseau. Le masque de sous réseau permet de spécifier quels sont les octets (voir les bits) qui définissent le réseau.

* il existe une version 6 d'IP, dont les adresses sont sur 128 bits, et qui apporte la possibilité de générer plus d'adresse IP, ainsi que de nouvelles fonctionnalités.

Exemple : adresse IP 182.30.25.13, masque de sous réseau : 255.255.0.0 : schématiquement les octets valant 255 définissent la partie de l'adresse qui indique le réseau donc pour l'adresse IP de l'exemple, le numéro de réseau, avec ce masque, est 182.30.

Il existe trois formats ou classes d'adresses Internet :

- classe A, le bit de poids fort vaut zéro, les 7 bits suivants désignent le réseau, les derniers 24 bits désignent l'adresse locale de la machine
- classe B, les deux bits de poids fort valent 1 et 0, les 14 bits suivants désignent le réseau et les 16 derniers bits l'adresse locale de machine
- classe C, les trois bits de poids fort forment le schème 110, les 21 bits suivants forment l'adresse réseau et les 8 derniers bits l'adresse locale

Adresse IP x.y.z.t	X	Y	Z	t
Classe de Réseau et Masque				
Classe A	De 0 à 127	indifférent	indifférent	indifférent
Masque classique	255	0	0	0
Classe B	De 128 à 191	De 0 à 255	indifférent	indifférent
Masque classique	255	255	0	0
Classe C	De 192 à 223	De 0 à 255	De 0 à 255	indifférent
Masque classique	255	255	255	0

On peut noter qu'une machine peut avoir plusieurs adresses IP : si elle a plusieurs cartes réseaux, ou si on met en place des systèmes d'alias d'adresse IP.

2) Attribution d'adresse

L'attribution des adresses IP peut se faire de deux manières différentes : soit de façon statique, soit de façon dynamique.

— Attribution statique : l'adresse IP de chaque équipement est définie de manière fixe ; l'adresse IP est alors renseignée « en dur » sur la machine (par exemple, dans les propriétés du protocole TCP/IP sur Windows). La gestion de l'affectation des adresses est réalisée par un administrateur réseau. Avantage : l'administrateur sait quelle machine a quelle adresse : il connaît donc parfaitement son réseau. Inconvénient : il faut gérer les changements d'adresses IP, essayer d'affecter au mieux les adresses, ne pas générer de doublon (conflit)... et cela prend du temps.

— Attribution dynamique : DHCP (Dynamic Host Control Protocol, monde Windows) et BOOTP (monde UNIX). Le DHCP est paramétré avec les plages d'adresses disponibles : lorsque les machines ont besoin d'une adresse IP, le DHCP affecte une adresse inutilisée.

Avantage : gestion automatisée, facilitée. Inconvénient : problème lorsque le DHCP est en panne, souci quand il s'agit de retrouver une machine via son adresse IP.

3) Résolution d'adresse : fonctionnement du DNS

Lorsqu'un utilisateur souhaite accéder à une machine, il ne connaît pas son adresse IP. Il connaît un nom. Il existe plusieurs possibilités pour résoudre les noms en adresses : ici, nous ne parlerons que de la résolution DNS (Domain Name Server *ou* Service).

Un DNS permet de relier une adresse IP (192.1.3.5) à un nom (ftp.bercy.cp). Grâce à un système d'alias, plusieurs noms peuvent correspondre à une même adresse : par exemple, <ftp2.d075.cp> correspondra à <ftp.bercy.cp>, lui-même relié à 192.1.3.5.

Chaque DNS gère un « domaine ». Il y a un domaine par suffixe : .fr, .com, .org... Sur Internet, lorsque l'on tape une adresse logique de type www.google.fr, on s'adresse à un DNS qui permettra la résolution de l'adresse. Si le premier DNS auquel la question est posée n'a pas la réponse, il s'adressera à un autre DNS et ainsi de suite jusqu'à ce que la réponse soit trouvée.

La résolution de nom peut aussi être faite en local sur le poste de travail, pour un certain nombre d'adresses : il s'agit de renseigner un fichier appelé HOST, avec les adresses IP et les noms qui y correspondent.

C/ Fragmentation

La fragmentation du datagramme Internet devient nécessaire dès lors qu'un datagramme de grande taille arrive sur une portion de réseau qui n'accepte que la transmission de paquets plus courts.

Un datagramme IP peut être spécifié "non fractionnable" Un tel datagramme IP ne doit jamais être fragmenté quelques soient les circonstances. Si un datagramme IP non fractionnable ne peut être acheminé jusqu'à sa destination sans être fragmenté, alors il devra être rejeté.

Les procédures de fragmentation et réassemblage doivent pouvoir "casser" un datagramme IP en un nombre de "fragments" arbitraire et quelconque pourvu que le réassemblage soit possible. Le récepteur des fragments utilise le champ d'identification pour s'assurer que des fragments de plusieurs datagrammes ne peuvent être mélangés. Des champs présents dans l'en-tête du datagramme permettent sa recomposition (voir infra, description d'un datagramme IP).

Pour fragmenter un long datagramme, un module Internet (par exemple, dans un routeur), crée deux nouveaux datagrammes et copie le contenu des champs d'en-tête Internet originaux dans les deux nouvel en-têtes. Les données du datagramme original sont divisées en deux portions, la première d'une taille multiple de 8 octets (64 bits) (la taille de la seconde portion n'est donc pas nécessairement un multiple de 8 octets).

Cette procédure peut être généralisée à une fragmentation en n fragments, plutôt que les deux décrits ci-dessus.

Pour ré-assembler les fragments d'un datagramme IP, un module Internet (par exemple dans un hôte destinataire) recombine les datagrammes dont les valeurs des quatre champs suivants sont identiques : identification, source, destination, et protocole. L'ordre des données peut être retrouvé grâce à d'autres champs.

III Le protocole TCP

Le protocole TCP est défini dans le but de fournir un service de transfert de données de **haute fiabilité** entre deux ordinateurs raccordés sur un réseau de type "paquets commutés", et sur tout système résultant de l'interconnexion de ce type de réseaux.

A/ Caractéristiques

TCP est un protocole **orienté connexion** conçu pour s'implanter dans un ensemble de protocoles multicouches, supportant le fonctionnement de réseaux hétérogènes. TCP fournit un moyen d'établir une communication fiable entre deux tâches exécutées sur deux ordinateurs autonomes raccordés à un réseau de données.

Le protocole TCP s'affranchit le plus possible de la fiabilité intrinsèque des couches inférieures de communication sur lesquelles il s'appuie. TCP suppose donc uniquement que les couches de communication qui lui sont inférieures lui procurent un service de transmission de paquet simple, dont la qualité n'est pas garantie.

TCP s'intègre dans une architecture multicouche des protocoles, juste au-dessus du protocole IP. Ce dernier permet à TCP l'envoi et la réception de segments de longueur variable, encapsulés dans un paquet Internet appelé aussi "datagramme". Le datagramme Internet dispose de mécanismes permettant l'adressage des services TCP source et destination, quelle que soit leur position dans le réseau. Le protocole IP s'occupe aussi de la fragmentation et du réassemblage des paquets TCP lors de la traversée de réseaux de plus faibles caractéristiques.

B/ Interfaces

TCP fournit un service de communication de processus à processus, dans un environnement réseau. TCP s'interface avec un processus utilisateur ou applicatif (couches supérieures) et un protocole de niveau inférieur du type IP (couche 3).

L'interface avec les applicatifs consiste en un ensemble de commandes. Par exemple, on trouvera des commandes pour établir et rompre une communication, pour envoyer ou recevoir des données sur une connexion ouverte.

TCP assume un fonctionnement avec un large ensemble de protocoles réseau. Dans ce document, nous nous limiterons au fonctionnement avec IP.

C/ Fonctionnement

TCP est conçu pour fournir un service de **transmission de données fiable** entre deux machines raccordées sur un réseau de paquets. Pour pouvoir assurer ce service même au dessus d'une couche de protocole moins fiable, les fonctionnalités suivantes sont nécessaires:

- Transfert de données de base
- Correction d'erreur
- Contrôle de flux
- Priorité et Sécurité ...

Ces fonctionnalités sont décrites en dans les paragraphes « Pour aller plus loin » qui suivent.

IV Autres protocoles

A/ UDP (couche 4)

Protocole de la couche transport qui fonctionne en mode non connecté et n'assure pas le contrôle de l'arrivée de l'information.

B/ ARP (couche 3)

Permet de faire la correspondance entre les adresses IP et les adresses MAC (adresses physiques des cartes)

C/ ICMP (couche 4)

Le protocole **ICMP** (*Internet Control Message Protocol*) est un [protocole](#) qui permet de gérer les informations relatives aux erreurs aux machines connectées. Etant donné le peu de contrôles que le [protocole IP](#) réalise, il permet non pas de corriger ces erreurs mais de faire art de ces erreurs aux protocoles des couches voisines. C'est ce protocole qu'utilise la commande PING (permettant de vérifier qu'une adresse répond, et donc que la machine concerné est bien active sur le réseau)

V Pour aller plus loin

A/ L'en-tête IP

0		16		32 bits
Ver.	LET	Type de service	Longueur totale	
Identification			Flags	Fragment Offset
Durée de vie	Protocole		Checksum d'en-tête	
Adresse source				
Adresse destination				
Option + Bourrage				
Data				

Version : 4 bits

Le champ Version renseigne sur le format de l'en-tête Internet. Ce document décrit le format de la version 4 du protocole (existe maintenant IP v6, qui doit permettre de pallier le problème du manque d'adresses IP)

Longueur d'En-Tête : 4 bits

Le champ Longueur d'En-Tête (LET) code la longueur de l'en-tête Internet, l'unité étant le mot de 32 bits, et de ce fait, marque le début des données. Notez que ce champ ne peut prendre une valeur en dessous de 5 pour être valide.

Type de Service : 8 bits

Le Type de Service donne une indication sur la qualité de service souhaitée, qui reste cependant un paramètre "abstrait". Ce paramètre est utilisé pour "guider" le choix des paramètres des services actuels lorsqu'un datagramme transite dans un réseau particulier. Certains réseaux offrent un mécanisme de priorité, traitant préférentiellement un tel trafic par rapport à un trafic moins prioritaire (en général en acceptant seulement de véhiculer des paquets d'un niveau de priorité au-dessus d'un certain seuil lors d'une surcharge momentanée). Principalement, le choix offert est une négociation entre les trois contraintes suivantes : faible retard, faible taux d'erreur, et haut débit.

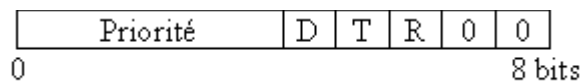
Bits 0-2 : Priorité.

Bit 3 : 0 = Retard standard, 1 = Retard faible.

Bits 4 : 0 = Débit standard, 1 = Haut débit.

Bits 5 : 0 = Taux d'erreur standard 1 = Taux d'erreur faible.

Bit 6-7 : Réservé.



Priorité

111 - Network Control

110 - Internetwork Control

101 - CRITIC/ECP

100 - Flash Override

011 - Flash

010 - Immediate

001 - Priority

000 - Routine

L'utilisation des indications en termes de retard, débit, et qualité de transmission peut augmenter le "coût" (d'un certain point de vue) du service. Dans la plupart des réseaux, de meilleures performances pour l'un de ces paramètres s'obtient au prix d'une dégradation des performances pour un autre. A moins d'une situation exceptionnelle, il sera préférable de ne pas activer plus de deux optimisations sur les trois.

Le "Type de Service" sert à préciser le traitement effectué sur le datagramme pendant sa transmission à travers Internet. Des exemples d'association de ce code aux améliorations de service proposées par des réseaux existants comme AUTODIN II, ARPANET, SATNET, et PRNET sont données dans la RFC 795 "Service Mappings" [8].

La priorité dite "Network Control" est stipulée comme étant une priorité à l'intérieur d'un seul réseau. Le fait d'utiliser cette option instaure une priorité pour chaque section traversée. La priorité "Internetwork Control" n'est gérée que par les routeurs. Si l'utilisation de ces priorités a une signification particulière ou supplémentaire pour l'un des réseaux, il est de la responsabilité de ce dernier de lire et d'interpréter les présentes informations.

Longueur Totale : 16 bits

Le champ "Longueur Totale" est la longueur du datagramme entier y compris en-tête et données, mesurée en octets. Ce champ ne permet de coder qu'une longueur de datagramme d'au plus 65,535 octets. Une telle longueur rendrait de toute façon les datagrammes impossible à gérer pour la plus grande partie des réseaux. Les hôtes devront au moins pouvoir accepter des datagrammes d'une longueur jusqu'à 576 octets (qu'il s'agisse d'un datagramme

unique ou d'un fragment). Il est de même recommandé que des hôtes ne décident d'envoyer des datagrammes de plus de 576 octets que dans la mesure où ils sont sûrs que la destination est capable de les accepter.

Le nombre 576 a été choisi pour permettre à un bloc de données de taille raisonnable d'être transmis dans un datagramme, tenant compte des données à ajouter pour constituer les en-têtes de protocole. Par exemple, cette taille permet la transmission d'un bloc de 512 octets, plus 64 octets d'en-tête dans un datagramme unique. (NdT : je rappelle ici que la taille de 512 octets correspond à un secteur sur la plupart des supports de stockage) La taille maximale d'un en-tête Internet étant de 60 octets, et sa taille typique étant de 20 octets, ce nombre permet de conserver une bonne marge pour les données protocolaires de plus haut niveau.

Identification : 16 bits

Une valeur d'identification assignée par l'émetteur pour identifier les fragments d'un même datagramme.

Flags : 3 bits

Divers commutateurs de contrôle.

Bit 0 : réservé, doit être laissé à zéro

Bit 1: (AF) 0 = Fragmentation possible, 1 = Non fractionnable.

Bit 2: (DF) 0 = Dernier fragment, 1 = Fragment intermédiaire.

0	AF	DF
---	----	----

Fragment Offset : 13 bits

Ce champ indique le décalage du premier octet du fragment par rapport au datagramme complet. Cette position relative est mesurée en blocs de 8 octets (64 bits). Le décalage du premier fragment vaut zéro.

Durée de vie : 8 bits

Ce champ permet de limiter le temps pendant lequel un datagramme reste dans le réseau. Si ce champ prend la valeur zéro, le datagramme doit être détruit. Ce champ est modifié pendant le traitement de l'en-tête Internet. La durée de vie est mesurée en secondes. Chaque module Internet doit retirer au moins une unité de temps à ce champ, même si le traitement complet du datagramme par le module est effectué en moins d'une seconde. De ce fait, cette durée de vie doit être interprétée comme la limite absolue maximale de temps pendant lequel un datagramme peut exister. Ce mécanisme est motivé par la nécessité de détruire les datagrammes qui n'ont pu être acheminés, en limitant la durée de vie même du datagramme.

Protocole : 8 bits

Ce champ indique quel protocole de niveau supérieur (TCP, UDP...) est utilisé dans la section données du datagramme Internet. Les différentes valeurs admises pour divers protocoles sont listées dans la RFC "Assigned Numbers" [rfc1060].

Checksum d'en-tête : 16 bits

Un Checksum calculé sur l'en-tête uniquement. Comme certains champs de l'en-tête sont modifiés (ex : durée de vie) pendant leur transit à travers le réseau, ce Checksum doit être recalculé et vérifié en chaque point du réseau où l'en-tête est réinterprétée.

L'algorithme utilisé pour le Checksum est le suivant :

On calcule le complément à un sur 16 bits de la somme des compléments à un de tous les octets de l'en-tête pris par paires (mots de 16 bits). Lorsque l'on calcule le Checksum, on considère une en-tête dont le champ réservé pour ce même Checksum vaut zéro.

L'algorithme de Checksum peut paraître élémentaire mais l'expérimentation a montré que cette technique était suffisante. Il se peut que cet algorithme soit plus tard remplacé par un calcul de type CRC, suivant la nécessité future.

Adresse source : 32 bits

L'adresse Internet de la source.

Adresse destination : 32 bits

L'adresse Internet du destinataire.

Options : variable

Les datagrammes peuvent contenir des options. Celles-ci doivent être implémentées par tous les modules IP (hôtes et routeurs). Le caractère "optionnel" concerne leur transmission, et non leur implémentation.

Dans certains environnements, l'option de sécurité peut être obligatoire dans tous les datagrammes.

Le champ d'option est de longueur variable. Un datagramme peut comporter zéro ou plus options. Voici les deux formats possibles d'une option :

Cas 1: Une option codée sur un seul octet.

Cas 2: Un octet codant le type d'option, un octet donnant la taille de l'option, les octets de données d'option.

La taille de l'option compte tous les octets de l'option y compris le type, son propre octet et tous les octets de donnée d'option.

L'octet de type d'option est composé de trois champs de bits :

1 bit indicateur de recopie

2 bits classe d'option

5 bits numéro d'option.

B/ Les fonctionnalités de TCP

1) Transfert de données de base

TCP est capable de transférer un flux continu de données entre deux ordinateurs, en découpant ce flux en paquets. En général, TCP décide de lui-même là où le flux de données doit être coupé.

Parfois les utilisateurs ont besoin de savoir que toutes les données soumises à TCP ont bien été émises. La fonction "push" a été prévue à cet effet. Pour s'assurer de la transmission complète de données jusqu'à un point spécifié, l'utilisateur activera la fonction "push" de TCP. Cette fonction oblige TCP à transmettre rapidement les données situées avant le point spécifié vers le destinataire. Il n'est nul besoin de fournir un marqueur spécifique pour ce point, dans la mesure où le destinataire accepte ces données comme une transmission normale.

2) Contrôle d'erreur

TCP doit considérer et traiter les cas de données perdues, erronées, dupliquées, ou arrivées dans le désordre à l'autre bout de la liaison Internet. Ceci est réalisé par l'insertion d'un numéro de séquence, et par l'obligation d'émission d'un "accusé de réception" (ACK) par le TCP destinataire. Si l'accusé de réception n'est pas reçu au bout d'un temps prédéfini, le paquet sera réémis. Côté récepteur, les numéros de séquence sont utilisés pour reconstituer dans le bon ordre le flux original, et éliminer les paquets dupliqués. L'élimination des erreurs physiques de transmission se fait par encodage d'un Checksum à l'émission, re-calcul de ce Checksum par le destinataire, et élimination des paquets pour les quels les deux valeurs ne correspondent pas.

Tant que TCP fonctionne correctement, et que le réseau Internet n'est pas saturé, aucune faute de transmission ne devrait transparaître dans la communication. TCP est donc sensé récupérer les erreurs de la transmission Internet.

3) Contrôle de flux

TCP fournit un moyen au destinataire pour contrôler le débit de données envoyé par l'émetteur. Ceci est obtenu en retournant une information de "fenêtre" avec chaque accusé de réception indiquant la capacité de réception instantanée en termes de numéros de séquence. Ce paramètre noté "window" indique le nombre d'octets que l'émetteur peut envoyer avant une autorisation d'émettre ultérieure.

4) Multiplexage

Pour permettre à plusieurs tâches d'une même machine de communiquer simultanément via TCP, le protocole définit un ensemble d'adresses et de ports pour la machine. Une "socket" est défini par l'association des adresses Internet source, destinataire, ainsi que les deux numéros de port à chaque extrémité. Une connexion nécessite la mise en place de deux sockets. Une socket peut être utilisée par plusieurs connexions distinctes.

L'affectation des ports aux processus est établie par chaque ordinateur. Cependant, il semble judicieux de réserver certains numéros de ports pour des services caractérisés et souvent utilisés. Ces services standards pourront alors être atteints via ces ports "réservés". L'affectation, l'utilisation et l'apprentissage des ports associés à d'autres services moins courants ou propriétaires nécessiteront l'utilisation de mécanismes plus dynamiques.

5) Connexions

Les mécanismes de fiabilisation et de contrôle de flux décrits ci-dessus imposent à TCP l'initialisation et la maintenance de certaines informations pour chaque communication. La combinaison de ces informations, dont les sockets, les fenêtres, et les numéros de séquence formeront ce que nous appelons une connexion. Chaque connexion est identifiée de manière unique par sa paire de sockets, définissant chacun des deux sens de la communication. Lorsque deux processus désirent communiquer, leur TCP respectifs doivent tout d'abord négocier et établir une connexion (initialiser ces informations d'état de part et d'autre). Lorsque la communication s'achève, elle sera fermée, en libérant ses ressources à d'autres usages.

Dans la mesure où l'on considère que les ordinateurs, ainsi que le réseau Internet n'est pas d'une fiabilité absolue, on utilise une méthode d'initialisation par négociation bilatérale basée sur une horloge pour les numéros de séquence.

Les paquets TCP sont envoyés sous forme de datagrammes Internet. L'en-tête IP transmet un certain nombre de paramètres, tels que les adresses Internet source et destinataires. L'en-tête TCP est placée à la suite, contenant les informations spécifiques au protocole TCP. Cette division permet l'utilisation de protocoles autres que TCP, au dessus de la couche IP.

C/ L'en-tête TCP

En-tête TCP

0		16										32bits									
Port Source										Port Destination											
Numéro de séquence																					
Accusé de réception																					
Data Offset		Réservé				U	A	P	R	S	F	Fenêtre									
Checksum										Pointeur données urgentes											
Option										Bourrage											
Data																					

Port source (16 bits) : le numéro de port de la source.

Port Destination (16 bits) : Le numéro de port du destinataire.

Numéro de séquence (32 bits) : Le numéro du premier octet de données par rapport au début de la transmission (sauf si SYN (*bit de control*) est marqué). Si SYN est marqué, le numéro de séquence est le numéro de séquence initial (ISN) et le premier octet à pour numéro ISN+1.

Accusé de réception (32 bits) : si ACK (*bit de control*) est marqué ce champ contient le numéro de séquence du prochain octet que le récepteur s'attend à recevoir. Une fois la connexion établie, ce champ est toujours renseigné.

Data Offset (4 bits): La taille de l'en-tête TCP en nombre de mots de 32 bits. Il indique là où commence les données. L'en-tête TCP, dans tous les cas à une taille correspondant à un nombre entier de mots de 32 bits.

Réservé (6 bits) : réservés pour usage futur. Doivent nécessairement être à 0.

Bits de contrôle = U, A, P, R, S, F (6 bits - de gauche à droite):

- URG: Pointeur de données urgentes significatif
- ACK: Accusé de réception significatif
- PSH: Fonction Push
- RST: Réinitialisation de la connexion
- SYN: Synchronisation des numéros de séquence
- FIN: Fin de transmission

Fenêtre (16 bits): le nombre d'octets à partir de la position marquée dans l'accusé de réception que le récepteur est capable de recevoir.

Checksum (16 bits) : le Checksum est constitué en calculant le complément à 1 sur 16 bits de la somme des compléments à 1 des octets de l'en-tête et des données pris deux par deux (mots de 16 bits). Si le message entier contient un nombre impair d'octets, un 0 est ajouté à la fin du message pour terminer le calcul du Checksum. Cet octet supplémentaire n'est pas transmis. Lors du calcul du Checksum, les positions des bits attribués à celui-ci sont marquées à 0.

Le Checksum couvre de plus une pseudo en-tête de 96 bits préfixée à l'en-tête TCP. Cette pseudo en-tête comporte les adresses Internet source et destinataires, le type de protocole et la longueur du message TCP. Ceci protège TCP contre les erreurs de routage. Cette information sera véhiculée par IP, et est donnée comme argument par l'interface TCP/Réseau lors des appels d'IP par TCP.

La longueur TCP compte le nombre d'octets de l'en-tête TCP et des données du message, en excluant les 12 octets de la pseudo en-tête.

Pointeur de données urgentes (16 bits) : Communique la position d'une donnée urgente en donnant son décalage par rapport au numéro de séquence. Le pointeur doit pointer sur l'octet suivant la donnée urgente. Ce champ n'est interprété que lorsque URG est marqué.

Options (variable): les champs d'option peuvent occuper un espace de taille variable à la fin de l'en-tête TCP. Ils formeront toujours un multiple de 8 bits. Toutes les options sont prises en compte par le Checksum. Un paramètre d'option commence toujours sur un nouvel octet. Il est défini deux formats types pour les options:

Cas 1: Option mono-octet.

Cas 2: Octet de type d'option, octet de longueur d'option, octets de valeurs d'option.

La longueur d'option prend en compte l'octet de type, l'octet de longueur lui-même et tous les octets de valeur et est exprimée en octets.

La liste d'option peut être plus courte que ce que l'offset de données pourrait le faire supposer. Un octet de remplissage (padding) devra être dans ce cas rajouté après le code de fin d'options. Cet octet est nécessairement à 0.

Définition des options spécifiques

Fin de liste d'options

Ce code indique la fin du champ d'options. Sa position peut ne pas coïncider avec l'indication du début du champ de données marqué dans l'Offset de données. Il doit être placé après toutes les options, et non après chaque option. Il ne doit être utilisé que dans le cas où la fin des options ne coïncide pas avec le début du champ de données.

No-Operation

Cette option peut être utilisée entre deux options, par exemple pour aligner le début d'une option sur un début de mot de 16 bits. L'utilisation de ce séparateur n'est pas une obligation. L'implémentation doit donc prévoir de pouvoir prendre en compte une option même au milieu d'un mot.

Taille maximale de segment

Donnée d'option : Taille maximale de segment: 16 bits

Si cette option est présente, elle communique à l'émetteur la taille maximale des segments qu'il pourra envoyer. Ce champ doit être envoyé dans la requête de connexion initiale (avec SYN marqué). Si cette option est absente, le segment pourra être pris de n'importe quelle taille.

Bourrage (padding): (variable)

Les octets de bourrage terminent l'en-tête TCP:

- de sorte que le nombre d'octet de celle-ci soit toujours multiple de 4 (32 bits)
- de sorte que l'offset de données marqué dans l'en-tête corresponde bien au début des données applicatives.

D/ Schéma d'encapsulation TCP/IP

