Depuis trente ans environ des systèmes informatiques (ou informatisés) échangent des données à travers des réseaux de télécommunications. La maigre qualité de ces réseaux a conduit dès l'origine les concepteurs de ces moyens de communications entre ordinateurs ou ordinateurs et terminaux à se poser le problème de la **sécurisation** de ces systèmes. Durant très longtemps le terme de sécurité a été équivalent à "fiabilité des données transportées"; les seuls problèmes envisagés (du moins dans les applications civiles ...) étaient dus à la dégradation des signaux, supports de l'information transmise.

Plus récemment les utilisateurs (et les concepteurs) se sont aussi souciés des agressions volontaires commises à l'encontre des données transportées: vol ou destruction. Les techniques de sécurisation ont donc dues être étendues à ces nouvelles difficultés (sûreté), au moment où les progrès technologiques permettaient des moyens de télécommunications offrant une fiabilité accrue pour l'information.

En fait les concepteurs et les fournisseurs de moyens de communication ont d'abord essayé de fournir des systèmes de qualité suffisante pour en développer l'usage, puis de leur adjoindre des fonctionnalités supplémentaires au fur et à mesure que les usages faisaient apparaître des besoins suffisants.

Nous allons exprimer de manière synthétique **les besoins des utilisateurs** (spécifications opérationnelles), puis **décrire les fonctions et activités nécessaire**s pour répondre à ces besoins (spécifications fonctionnelles) **et les architectures standards** permettant d'organiser et d'implanter ces activités.

1. SPECIFICATIONS OPERATIONNELLES

Les besoins des utilisateurs peuvent être résumés dans la proposition suivante :

Transférer en temps utile des informations fiables d'une source vers un ou plusieurs collecteurs de manière transparente, sûre, économique et conviviale.

1.1. L'information

Dans les systèmes de communication, son sens n'est pas pris en compte. Elle est considérée comme une grandeur mesurable issue d'une source ou transmise à un collecteur.

Cette grandeur abstraite est liée à l'occurrence plus ou moins probable d'un événement: état d'un capteur, idée dans le cerveau, tout fait non déterministe, etc. L'information est codée dans

un message qui peut être transmis. Sa grandeur est liée à la probabilité d'occurrence de cet événement ou du contenu du message.

Le "Petit Robert" définit l'information par :

- *- **Mesure de la densité de renseignement** contenu dans un message (pour un nombre de signes donnés). **Concept entièrement différent de celui de sens, de signification**, opposé à redondance.*

· *Elément ou système pouvant être transmis par un signal ou une combinaison de signaux.*

et un message par :

- *- Elément matériel par lequel un ensemble d'informations organisées selon un code circule d'un émetteur à un récepteur....*

Si X est un événement qui se produit avec la probabilité P(X) l'information I(X) vaut :

$$I(X) = \log_2 \frac{1}{P(X)} \text{ bits}$$

Cette information, grandeur abstraite, pour être transmise doit être supportée par des signaux susceptibles d'être transmis de la source au(x) collecteur(x). Le support de transmission peut être un objet (par exemple une disquette ou une lettre ..) ou un signal acoustique ou **électromagnétique**.

1.2. Transfert, distance et temps utile.

Le message à transférer code des signaux qui sont transmis sur un **réseau de télécommunications.** Celui-ci utilise différents types de supports capables de transporter le signal : câble téléphonique, fibres optiques, signaux radio par exemple.

Il est classique de classer les réseaux en **réseaux locaux et réseaux étendus,** voire en sous-classes : réseaux locaux, réseaux de campus, réseaux métropolitains, réseaux nationaux et internationaux. Cette classification repose sur des critères technologiques qui n'existaient pas avant 1980 environ et tendent à disparaître avec l'avènement des technologies à très haut débit, par exemple les réseaux ATM (Asynchronous Transfert Mode = TTA : Transfert Temporel Asynchrone) qui conçus pour des communication à moyenne ou longue distance apparaissent d'abord dans des réseaux locaux.

La classification réseau local vs. réseau étendu est plus stable et repose actuellement sur des **critères plutôt économiques et juridiques**.

Dans le domaine local le réseau de télécommunications est sous l'entière responsabilité des utilisateurs (entreprise). A grande distance (réseau étendu) il utilise les services d'un opérateur externe public ou privé (ou de plusieurs) selon la juridiction nationale, qui offre(nt), en location, un ensemble plus ou moins restreint de services. Si on considère la distinction national vs. international, on ne distingue pas de différences technologiques, les coûts plus élevés sont dus à des distances plus

grandes; les problèmes posés sont essentiellement d'ordre légal, en particulier dans le domaine de la sûreté.

Ces réseaux de télécommunications constituent ainsi un environnement imposé pour le système de communication.

L'utilisateur doit transférer ses informations en **temps utile**, c'est à dire de manière que le collecteur puisse en disposer quand il en a encore besoin (Il est parfaitement inutile d'être prévenu à 11 heures que votre rendez-vous de 10 heures était supprimé, et même de recevoir ce message à 9 heures 55 la plupart du temps ....).

Le temps mis pour transférer un message est la somme des temps d'émission, de propagation et de réception.

Le temps de propagation est lié à la nature du support et à la distance. Sur un câble, par exemple, il vaut 5µs par km (soit 5ms pour 1000km). La propagation se faisant à la vitesse de la lumière, ce temps est incompressible. On doit lui ajouter les retards induits par la traversée des équipements électroniques intermédiaires.

Les temps d'émission et de réception sont liés :

- à la bande passante du support utilisé
- à sa qualité (rapport signal sur bruit)
- aux équipements électroniques et informatiques situés aux extrémités du circuit

Le débit informationnel maximal d'un support est donné par le théorème de Shannon-Hartley-Tuller :

$$D = F_s \times \tfrac{1}{2} \log_2 \left(1 + \frac{S}{B}\right) \text{ bit / seconde}$$

avec $F_S = 2\,F_C$ (Théorème de Nyquist)

où S est la puissance du signal, B le bruit parasite
F_S le nombre maximal de signaux transmissibles par seconde
et Fc la bande passante du support

Les hauts ou très hauts débits nécessitent donc:

- des supports à large bande très peu perturbés (par exemple à fibre optique)
- des équipements d'extrémité performants.

1.3. Fiabilité

Nous abordons ici un volet de l'exigence de sécurité, abordé très tôt dans la conception des réseaux étant donné la faible qualité, il y a quelques décennies, des supports de télécommunications disponibles.

Ce qui est recherché ce n'est pas la fiabilité des équipements, qui est supposée acquise, mais la **fiabilité des informations transmises**.

Celle-ci doit être :

- **sans pertes**
- **sans duplications**
- **avec un taux d'erreurs négligeable.**

Si l'exigence de non-perte et non-duplication peut être satisfaite, Shannon a montré qu'il n'était pas possible de garantir la transmission d'un message sans erreurs (avec un taux d'erreurs résiduelles nul) en un temps fini.

Actuellement, selon les applications, un taux d'erreurs résiduelles de 10^{-10} à 10^{-14} peut être considéré comme négligeable.

Les supports de télécommunications de type téléphonique analogique offrent actuellement des taux d'erreurs de l'ordre de 10^{-4}. Sur un support numérique (RNIS : Réseau Numérique à Intégration de Services) ou sur un réseau local, ce taux peut descendre à 10^{-7} ou mieux. Sur un support optique il peut être meilleur que 10^{-12} pour des distances de quelques kilomètres.

En fonction des exigences des applications et de la nature et qualité du support utilisé il conviendra donc de ramener par correction ce taux d'erreurs à une valeur acceptable.

1.4 Transparence

L'utilisateur doit disposer au collecteur de **données directement utilisables**. Lorsque les systèmes communiquants sont identiques ou suffisamment semblables (homogènes) ceci est réalisé sans aucune action. Avec la mise en place de réseaux supportant de plus en plus d'utilisateurs (sans doute plus de 2 millions de systèmes reliés au réseau Internet actuellement), il est possible d'accéder à des applications ou des données extrêmement variées qui sont supportées par des systèmes complètement hétérogènes à celui de l'utilisateur. Celui-ci ne peut disposer de manière spécifique de tous les outils d'adaptation nécessaires ni même les connaître. **Cette fonction doit être totalement prise en charge par le réseaux.** Ceci ne peut être réalisé que par une normalisation stricte de la manière de coder les données utilisateurs qui doivent être transférées. De gros efforts sont fait en ce sens depuis une dizaine d'années pour ce qui est accessible au système de communication au sens strict.

Pour des applications qui doivent collaborer, ceci est du ressort de groupement d'utilisateurs autour d'un type d'application; l'échange de données informatisées (EDI; Electronic Data Interchange) relève de cette problématique comme l'Architecture de Documents Ouverts (ODA, Open Document Architecture). Ces applications bénéficient des recherches faites pour les systèmes de communication dans le cadre de l'OSI (Organisation de Standardisation Internationale) et du CCITT, partie de UIT (Union Internationale des Télécommunications, Commission Consultative Internationale du Télégraphe et du Téléphone) et utilise les mêmes outils.

1.5. Sûreté

Nous abordons ici l'autre aspect de la sécurité: la protection contre les agressions.

Ces agressions peuvent être :

- passives. Il s'agit alors de vol de données ou de logiciels
- actives : destruction volontaire ou involontaire de données, de logiciels ou de matériels ou perturbation (blocage) des communications.

En général les agressions passives ne mettent pas en cause l'intégrité des systèmes mais seulement la **confidentialité** des données et des systèmes (logiciels, utilisateurs) et passent souvent inaperçues car on dispose toujours des données volées (contrairement aux biens matériels).

Les agressions actives mettent en cause l'**intégrité des données et des systèmes**. Elles sont d'autant plus difficiles à contrer qu'elles sont réalisées par des personnes habilitées ont généralement considérées comme redoutables ....

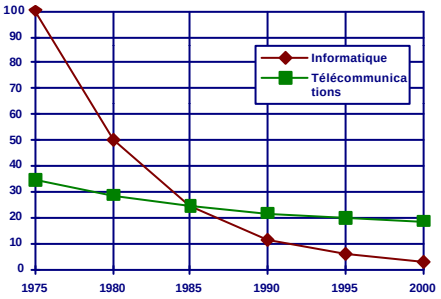
Les systèmes sont souvent considérés comme des points vulnérables pour la sécurité des systèmes répartis. Ce reproche est souvent justifié et cette sûreté repose souvent sur les moyens mis en oeuvre dans les systèmes interconnectés. Cependant les réseaux offrent parfois des moyens de sécurité méconnus et sous-utilisés.

Cette fragilité est aussi due aux contraintes auxquelles devaient satisfaire les premiers concepteurs de certains systèmes; issus du milieu universitaire, ils étaient peu sensibilisés aux problèmes de protection, pensaient n'avoir rien ou presque à cacher et ont mis tout en oeuvre pour faciliter l'usage et l'extension des réseaux dans la communauté scientifique.

Dans ce domaine, il ne faut pas ignorer les contraintes juridiques qui restreignent le champ de liberté du concepteur ou de l'administrateur de réseaux.

1.6. Economie

Les communications, à longue distance en particulier, ont des coûts élevés et ceux-ci baissent peu au fil des années contrairement aux coûts informatiques qui peuvent décroître de moitié chaque année, à puissance constante.



Pour certaines applications, les coûts de télécommunications dépassent largement les coûts informatiques purs. Dans ces conditions, on doit d'abord rechercher une réduction des coûts de fonctionnement/amortissement sur les télécommunications. (Par exemple General Motors a estimé en 1984 que les coûts des communications dans ses systèmes de production devenaient équivalents aux coûts informatiques; ceci a conduit à l'architecture MAP : Manufacturing Automation Protocol

qui conduisait à standardiser les communications dans ce domaine applicatif pour en permettre une réduction des coûts par rationalisation).

1.7. Convivialité

Cette contrainte est apparue beaucoup plus récemment avec l'arrivée des systèmes de communication auprès de personnels sans formation ou culture informatique.

Un bon système de communication doit être "invisible" pour les utilisateurs. Ceux-ci doivent avoir les mêmes usages, les mêmes facilités, les mêmes comportements que les systèmes de traitement ou les données soient locaux ou distants.

Les systèmes de messagerie par exemple doivent être très faciles à utiliser pour être bien acceptés et remplacer rapidement les communications par courrier papier ou par fax. Il doit en être de même pour les échanges de données informatisées.

Nous venons de balayer les besoins des utilisateurs. Ils ne sont pas tous de même priorité ou urgence. On considère en général qu'un transfert fiable en temps utile constitue le minimum indispensable. L'exigence économique est apparue depuis une quinzaine d'années et celle de sûreté était liée à certains types d'utilisateurs ou d'application qui acceptaient de prendre en compte le facteur économique à un niveau secondaire.

Actuellement les utilisateurs demandent de plus en plus une optimisation de leur système de communications qui doit leur fournir, au meilleur prix, la qualité de service qu'ils souhaitent en terme de fiabilité des données, de délais ou de débits de transmission, de confidentialité ou d'intégrité avec une interface homme-machine conviviale.

Nous allons examiner quelles activités et fonctions doivent être mises en oeuvre pour atteindre ces objectifs.

2. SPECIFICATIONS FONCTIONNELLES

2.1. Avant-propos : activités et composants

Etant donnés les coûts et la complexité des systèmes de communications, ils ne peuvent être totalement spécifiques ou dédiés à une application, mais ils doivent se partager le plus d'éléments réutilisables possible.

Une activité est un ensemble cohérent d'actions élémentaires unies dans la poursuite d'un but défini. Cette notion est récursive et une activité peut être décomposée en un ensemble d'activités plus élémentaires. Elle généralise la notion de processus d'un système informatique. Pour s'exécuter une activité utilise des composants qui en constituent le support. Cette notion est proche de celle de ressource. Ainsi une activité peut aussi être considérée comme l'association évolutive de composants vers un but commun. On distingue les activités :

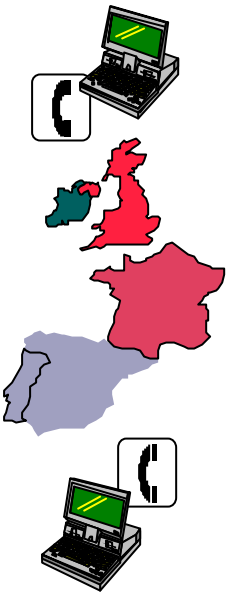
- allocatrices
- utilisatrices

Les premières répartissent à la demande les composants nécessaires aux secondes ce qui peut induire des attentes occasionnelles. Ainsi un système de communication peut être considéré comme un vaste système de files d'attentes. Une activité est multiple si elle fait intervenir plusieurs composants. Sa commande peut être centralisée ou distribuée selon qu'elle est confiée à un seul ou à plusieurs composants. Un système de communication peut être décrit comme assemblage de composants distribués permettant la coopération harmonieuse de différentes activités. L'interaction d'un ensemble de composants est réalisée à leur interface.

La coopération d'un ensemble de composants est difficile à maîtriser et, en général, on ne traitera que des couples de composants. D'une manière générale, un protocole définit les relations entre les comportements des différents composants d'une activité au travers de l'interface.

2.2. Activités de base

2.2.1. Pour une liaison directe fiable entre systèmes homogènes



Le système de communication le plus simple consiste à relier directement deux systèmes informatiques par une liaison point-à-point. Cette liaison peut être constituée par un simple câble ou par un circuit téléphonique spécialisé ou commuté. En effet après établissement de la communication sur un réseau en commutation de circuits (téléphonie analogique ou numérique) les signaux sont transmis sans stockage intermédiaire comme sur un simple câble.

La fiabilisation de l'information suppose que le système de communication est lui-même fiable et permet de transférer des données sans pertes ou duplications, avec un taux d'erreurs négligeable. Ce taux doit être compatible avec les applications supportées, et dans le meilleur des cas, comparable à celui rencontré dans les systèmes de traitement interconnectés.

Un taux d'erreurs résiduel de l'ordre de quelques 10^{-10} à 10^{-12} est généralement recherché. Il peut devoir être mille fois plus faible... Sur un système de transmission, le taux d'erreurs observé varie de 10^{-4} à 10^{-7} environ, selon la nature du support. Il peut parfois atteindre 10^{-2} (pour un très mauvais support) ou 10^{-12} (pour une fibre optique de quelques dizaines de kilomètres). D'autre part les erreurs de transmissions ne sont pas réparties de manière complètement aléatoire (erreurs isolées) mais regroupées en "paquets d'erreurs" qui eux apparaissent aléatoirement. Pour assurer une fiabilité suffisante, le taux d'erreurs doit être réduit d'un facteur de l'ordre de 10^6. Ceci est réalisé par un système de correction des erreurs et un système de séquencement contrôlé des blocs de données transmis.

2.2.1.1. Correction des erreurs de transmissions

Lorsque l'information transmise est très redondante un système de correction directe des erreurs peut être mis en oeuvre. Le codage de l'information étant binaire, il suffit de trouver l'emplacement des bits erronés pour pouvoir les corriger (en les inversant 0 -> 1 ou 1 -> 0). Cette technique, utilisée dans les systèmes de traitement (mémoire protégée) est lent et/ou coûteux dans les systèmes de communication (Pour la protection des mémoires, le système décrit ci-dessous ne peut être utilisé...).

Dans les systèmes de transmission on utilise une technique de détection des erreurs et répétition des blocs de données erronés. Pour détecter qu'un bloc de données est perturbé, on peut analyser le signal porteur (qui en constitue l'emballage) et refuser tout bloc transmis sur un signal trop perturbé. On utilise la "détection de qualité de signal" optionnelle sur les systèmes de transmission. Cette technique conduit à un ralentissement (généralement négligeable) par rejet de blocs sans erreurs.

Le système normalement utilisé consiste à ajouter une redondance légère à chaque bloc transmis (2 ou 4 octets) qui permet de déceler qu'un bloc est erroné dans presque tous les cas et de rejeter ce bloc. Les blocs de données erronés sont donc perdus. Le système collecteur constate cette perte en analysant un numéro de séquence affecté à chaque bloc et en demande la retransmission. (Ces messages de demande de répétition peuvent aussi être perturbés...). Ce système de répétition peut engendrer la duplication de bloc (que le contrôle de séquencement permet de détruire). D'autre part on ne tente pas de retransmettre ad aeternam le même bloc erroné. La retransmission est abandonnée après quelques tentatives infructueuses et une anomalie signalée. Des blocs peuvent ainsi être perdus sur ce défaut de transmission.

2.2.1.2. Contrôle de flux

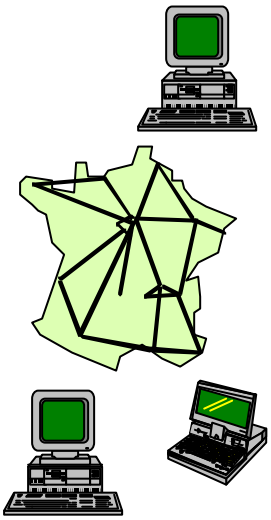
La cause principale de pertes de données ne provient pas du système de correction des erreurs mais de la saturation du collecteur par un flux de données trop important pour ses capacités de réception : La source doit être asservie au collecteur pour éviter de lui envoyer trop rapidement ses blocs de données. Cette fonction, essentielle, constitue le contrôle de flux. Elle est mise en oeuvre entre les systèmes communiquant mais aussi à l'intérieur de chaque système, aux interfaces entre les sous-ensembles matériels ou logiciels.

On peut mettre en oeuvre un système matériel pour les accès au système physique de transmission. Le plus souvent on utilise un système de crédit: Le collecteur indique à la source, parfois implicitement, combien de blocs de données il est capable de recevoir; la source ne peut alors dépasser ce crédit qui doit être renouvelé par des messages de service. (Le "protocole Xon-Xoff" (ctrlS-ctrlQ) est un mécanisme de contrôle de flux utilisé pour réguler l'affichage sur les terminaux. Il est assez mal adapté à la transmission de données.....!) Entre systèmes, le contrôle de flux de type "crédit" (ou fenêtre) utilise le séquencement mentionné ci-dessus.

2.2.2. Pour une liaison par un réseau entre systèmes homogènes

Nous considérons ici un réseau de communications comprenant des équipements informatiques intermédiaires : routeurs ou commutateurs de paquets : Réseaux de type X25 (Transpac par exemple) ou Réseaux de type IP comme le réseau Internet.

2.2.2.1 Adressage



les fonctions ci-dessus assurent un transfert sûr entre deux systèmes connectés directement par une liaison physique point-à-point. Dans la plupart des cas, l'interconnexion est beaucoup plus complexe et met en oeuvre un ou plusieurs réseaux locaux éventuellement reliés par un ou des réseaux étendus.

Le système d'adressage devra permettre de désigner non seulement le système à atteindre ou le système origine mais en général son point d'accès au réseau étendu auquel il est relié. Sur ce système peuvent coexister plusieurs applications se déroulant simultanément: On devra aussi pouvoir désigner l'application ou le terminal qui l'utilise. Enfin, à l'intérieur du système d'interconnexion, chaque réseau intermédiaire doit être repéré. A cet effet on peut utiliser un codage hiérarchique des adresses. L'utilisateur doit avoir un moyen clair et commode de désigner, de manière unique, le système ou l'application distante. Une mise en correspondance (mapping) de ces noms ou adresses logiques avec les adresses réelles de chaque composant devra être réalisée. Pour accélérer les communications en diminuant les volumes transmis, des adresses temporaires abrégées sont utilisées par les logiciels de communication**.**

Sur les réseaux étendus on peut par exemple utiliser les adresses codées selon le standard X121. Une telle adresse est codée sur au maximum **15 chiffres décimaux**. Le premier indique le type d'adresse : 0 = adresse internationale
1 = adresse nationale

avec une adresse nationale le champ suivant indique le pays (France 208 0)
Les chiffres suivant donnent le point d'accès au réseau et les derniers peuvent être utilisés pour une sous-adresse local.

Exemple : 0 2080 69 001 886 06

Sur le réseau Internet, on utilise une *"adresse IP"* codée sur **32 bits.** Celle-ci est notée comme une suite de 4 octets représentée par leur valeur décimale.

Exemple : 134.214.100.21

Pour augmenter la convivialité on utilisera une fonction de nommage (voir § 2.3.4)

2.2.2.2. Acheminement - Routage

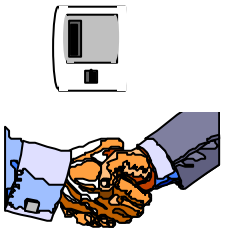
Les données transmises doivent atteindre leur destination, désignée par son adresse, en utilisant le chemin le plus rapide ou le plus économique.

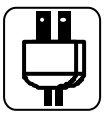
Cet itinéraire doit être déterminé soit de manière statique pour tout couple source-collecteur (acheminement déterministe fixe), soit déterminé avant chaque communication en fonction de la qualité de service requise (délais, taux d'erreurs, coûts, etc.) ou de la charge du réseau. La détermination de ces chemins à priori, à la demande ou de manière automatique (acheminement adaptatif) utilise des algorithmes appelés "procédure de routage" qui permettent de mettre à jour des tables d'acheminement dans chaque site. Cette fonction est appelée routage. Elle peut comprendre aussi une phase dans laquelle un système qui se raccorde au réseau détermine automatiquement tous les itinéraires qu'il peut établir ou le noeud le plus proche qui peut lui fournir ces itinéraires.

Lorsque un message transite dans un site intermédiaire du réseau il est acheminé (aiguillé) sur la liaison convenable en fonction de ces tables de routage; c'est la fonction d'**acheminement**.

2.2.3. Pour négocier les options de la communication : Connexion - Libération ou Association-Rupture

Les fonctions ci-dessus permettent de relier logiquement deux applications. Toutefois, avant qu'elles engagent un dialogue il est nécessaire qu'elles se mettent d'accord sur les modalités de leurs échanges. Elles doivent négocier la qualité du service à mettre en oeuvre (débit, délai, taux d'erreurs, taille des blocs de données par exemple) pour ne pas engager un dialogue sur une liaison qui ne garantirait pas la qualité souhaitée. Elles doivent aussi décider de la syntaxe commune (codage spécifique des données) utilisée par les applications





communicantes (voir § ci-dessous). Cette négociation est réalisée dans une phase préliminaire de connexion ou d'association d'applications. Durant cette phase on initialise les variables du système de communication, par exemple de séquencement, et on réalise la mise en correspondance des adresses.

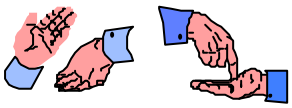
En fin de dialogue, la communication peut être rompue abruptement. Il est souvent commode de la terminer par un échange de messages particuliers pour arrêter proprement le système. Enfin la terminaison peut être négociée de manière à n'être effective que lorsque les deux applications ont traité tous leurs besoins.

2.2.4. Pour assurer la transparence entre systèmes hétérogènes

Les données transférées d'une source vers un ou plusieurs collecteurs doivent être directement utilisables. Lorsque les systèmes d'exploitation sont identiques ou très compatibles, les systèmes communiquant sont dits homogènes. Dans ce cas si les logiciels d'application utilisés pour traiter les données sur ces systèmes sont identiques, il suffit que le système de communication soit transparent aux données qui le traversent. Ce système ne doit filtrer ou modifier aucun code.

Dans le cas de systèmes hétérogènes, les données ne sont pas directement utilisables sans une adaptation. Pour ne poser aucun problème à un utilisateur qui peut dialoguer avec de nombreux systèmes différents, cette traduction doit être prise en compte par le système de communication. L'analyse des applications à supporter a montré qu'il n'était pas possible d'utiliser une syntaxe unique commune pour coder de manière efficace la grande variété des données à transférer (création d'un "espéranto" informatique).

Pour chaque classe d'application : messagerie "interpersonnel", transfert et gestion de fichiers, transactionnel, accès à des données distantes, applications (messagerie) industrielles, etc., on crée une "syntaxe de transfert" spécifique à cette application. De telles syntaxes peuvent aussi être créées de manière privée. Lors de la phase d'association des applications la syntaxe qui va être utilisée est signalée au correspondant. Pour transmettre ce type d'information, il est nécessaire de disposer d'un langage commun, une syntaxe de transfert, sur les systèmes qui dialoguent. Ces syntaxes de transfert sont construites à partir de la **seule syntaxe abstraite ASN.1** (Abstract Syntax Notation n°1) normalisée actuellement (ISO 8824 CCITT X208). Pour cela elle utilise des règles de codage de base (ISO 8825 - CCITT X209). Elle sert aussi de base pour la création des syntaxes de transfert spécifiques des classes d'Application.



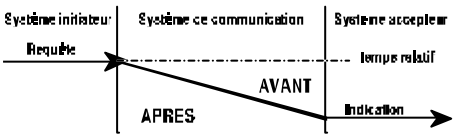
2.3. Activités d'optimisation

Les fonctions décrites jusqu'ici assurent un transfert d'informations fiables entre deux systèmes hétérogènes interconnectés. Pour rendre cette communication plus efficace, moins coûteuse ou plus rapide, il est possible d'utiliser quelques fonctions supplémentaires.

2.3.1. Optimiser la fiabilité

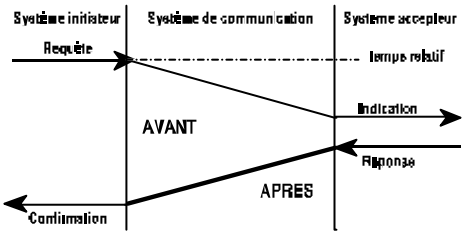
2.3.1.1 Synchronisation

Pour contrôler le bon déroulement des échanges et pouvoir reprendre un dialogue ou un transfert sur incident ou arrêt volontaire, il est nécessaire de poser des points de repères temporels au cours des échanges. Ces repères sont appelés points de synchronisation. Etant donné les temps de transfert des blocs de données, liés essentiellement au débit de transmission, il n'est pas possible d'utiliser une mesure classique du temps, en particulier sur un réseau étendu. Les points de synchronisation doivent être posés par des messages spécifiques intercalés dans les échanges normaux. Durant une période donnée, ces points ne peuvent être posés que par un seul des interlocuteurs.



Le schéma ci-contre illustre une synchronisation simple. Celle-ci ne permet pas une séparation absolument étanche de deux tranches temporelles.

Pour avoir une séparation sûre, il est nécessaire d'utiliser un service de synchronisation confirmé, illustré sur le schéma suivant.



2.3.1.2 Reprises sur défaut - Journalisation

A la suite d'une anomalie, le dialogue ou le transfert en cours doit être rétabli. Il peut toujours être repris au début mais ceci peut être très coûteux en temps ou financièrement. Pour améliorer l'efficacité du système on utilise une fonction de reprise qui négocie l'endroit, repéré par un point de synchronisation, où doit être rétablie la communication et qui redémarre celle-ci.

Pour pouvoir réaliser cette fonction ces points de contrôle doivent être mémorisés dans une mémoire non volatile : disque ou EEPROM. Les informations nécessaires pour reconstituer les données qui doivent être retransmises après le rétablissement de la communication doivent aussi être enregistrée par une fonction de *journalisation*.

2.3.1.3. Gestion de la communication

Pour gérer les échanges, il est souhaitable, en particulier, de pouvoir désigner le système qui, à un moment donné, est habilité à

- émettre des données
- poser un point de synchronisation
- terminer la communication.

Dans un système *maître-esclave*, ces caractéristiques peuvent être définies à priori; l'activité de gestion de communication n'est alors pas nécessaire. Dans un système *équilibré* ces fonctions sont gérées de manière dynamique (par le passage de jetons). Il faut aussi décider qui peut donner ces droits notamment en début d'une communication ou après son rétablissement à la suite d'une rupture sur anomalie.

2.3.1.4. Contrôle de congestion

Malgré la mise en oeuvre du contrôle de flux, certains réseaux peuvent être "congestionnés" et ne plus pouvoir transmettre assez rapidement les données. Un contrôle global des flux dans le réseau doit être mis en place et certaines données éventuellement réacheminées sur d'autres liaisons ou bloquées (temporairement) à l'entrée du réseau. En cas de blocage, la guérison passe généralement par une **purge** plus ou moins complète des données en transit, donc de leur destruction...

2.3.2. Optimiser les performances

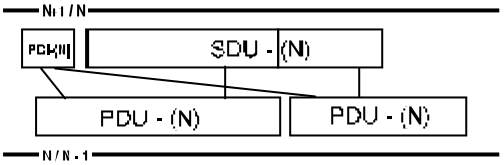
2.3.2.1. Eclatement - recombinaison

A l'inverse, un support de communication peut offrir un débit trop faible pour assurer les délais de transmission requis (correspondant à la qualité de service souhaitée). Le transfert peut alors être éclaté entre plusieurs liaisons utilisées en parallèle. Les données doivent être recombinées correctement sur le système collecteur. Cette fonction optimise les performances.

2.3.2.2 Fragmentation - Regroupement

Sous cette dénomination sont regroupées trois fonctions voisines. L'utilisateur doit manipuler des blocs de données conformes à son application: ce peut être des fichiers très longs aussi bien que des messages très courts. D'autre part le rendement de la transmission est optimisé si on utilise des trames de données de quelques dizaines ou centaines d'octets: pour des trames très courtes, les octets de contrôles diminuent beaucoup le débit utile; des trames trop longues seront très probablement reçues erronées et devront être répétées, réduisant là encore le débit utile. Enfin, à l'intérieur du système de communication, il est souvent plus efficace de manipuler des blocs de

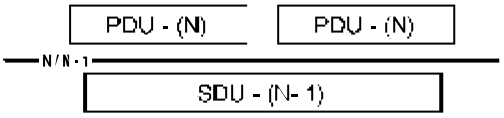
quelques milliers d'octets, par exemple pour réduire les accès disques ou les manipulations de données sur des interfaces.



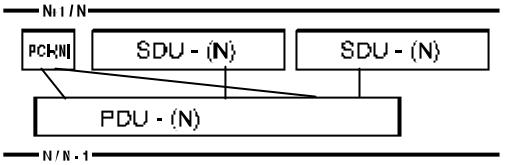
- La **segmentation** avec **réassemblage** des données est nécessaire pour que l'utilisateur puisse traiter des blocs de données très longs.

- La **concaténation** suivie d'une **séparation** permet de transférer ensemble des messages trop courts. Il en est de même des fonctions de **groupage - dégroupage** qui permettent de regrouper, dans la mesure du possible, des messages de service. Il faut veiller à ne regrouper que des données que l'on saura séparer ou dégrouper (La distinction entre concaténation et groupage est plutôt liée à la manière de les réaliser dans les logiciels de communication).

Concaténation - Séparation

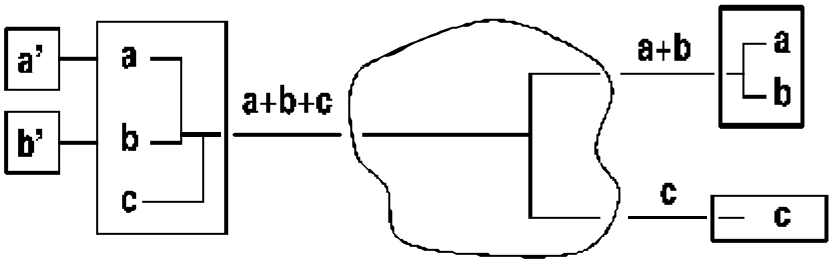


Groupage - Dégroupage



2.3.3. Optimiser les coûts : Multiplexage - Démultiplexage

Très souvent les capacités de transfert des liaisons utilisées offrent des débits très supérieurs à celui requis par une application. On peut alors faire partager ce support par plusieurs communications, éventuellement vers des destinations différentes, à travers un accès commun à un réseau. On crée ainsi des liaisons logiques multiplexées souvent appelées circuits virtuels. En fait, sur un circuit virtuel entre deux systèmes interconnectés par un réseau, il est possible de multiplexer les communications de plusieurs applications qui coopèrent deux à deux sur ces systèmes. Cette fonction réalise une optimisation économique.



2.3.4. Optimiser la convivialité

2.3.4.1. Nommage

Il est difficile de se souvenir des adresses des différents systèmes distants à utiliser (sauf d'un nombre très réduit des plus fréquemment utilisés). On utilise donc une fonction de nommage permettant de désigner de manière claire un système. Une table fait correspondre chaque nom à une adresse. Un système peut avoir aussi des noms synonymes (alias). Ce nom comporte en général un nom de système et un nom de domaine administratif plus ou moins hiérarchisé (établissement+pays).

Ainsi sur le réseau Internet un système peut être désigné par

système.établissement.pays

par exemple

serveurif1.insa-lyon.fr qui recevra comme alias *if.insa-lyon.fr*

En donnant le même alias à un autre système on peut reconfigurer le réseau sans en changer les noms utilisés.

2.3.4.2. Annuaire

Les tables de traduction des noms en adresses peuvent être situées sur chaque système. Cependant sur des réseaux de grande taille ou même simplement moyenne leur mise à jour est complexe. Il est alors préférable de disposer ces tables sur un ou quelques systèmes "serveurs de noms" qui traitent ces fonctions d'annuaire. Sur chaque système il suffira de connaître l'adresse d'un ou deux (secours) serveurs de noms. Un logiciel d'annuaire permettra de rechercher l'adresse d'un correspondant dont on connaît le nom en recherchant cette adresse sur le serveur en lui communiquant son nom.

La fonction de détermination automatique de tous les systèmes qui peuvent être connectés à un système donné et de détermination automatique des itinéraires sont en pratique des fonctions destinées à accroître la convivialité pour les administrateurs de réseaux.

2.3.5. Assurer la sûreté des communications

Cette fonction est chargée de protéger les systèmes de traitement (et le système de communication) contre les attaques passives (espionnage des données en transit) ou actives (essai d'intrusion dans un système, destruction ou perturbation de données en transit). Les attaques actives sont souvent réalisées involontairement par des correspondants ayant le droit d'accéder aux systèmes mais le faisant incorrectement... Certaines informations confidentielles peuvent être déduites simplement de l'analyse du trafic ou de graves incidents créés en perturbant celui-ci.

L'usage de liaisons par satellites de télécommunications entraîne la diffusion des informations sur une zone géographique immense. L'espionnage de ces données est donc simple si l'on dispose d'un système de réception....

La protection contre les attaques passives, pour assurer la **confidentialité** des données, passe par le *chiffrement de ces données*.

Contre les intrusions on doit **identifier** le système appelant pour réaliser un **contrôle d'accès** grâce à un mot de passe associé à cet identificateur. Il faut aussi vérifier que l'utilisateur appelant correspond bien à l'identificateur (vol d'identité). Pour cela on doit **authentifier** sa signature. Il est aussi nécessaire que chaque système qui se connecte à un serveur distant contrôle qu'il s'agit réellement du système souhaité pour éviter les techniques de *déguisement (mascarade)*. Par cette méthode un système pirate se fait passer auprès d'un client pour un serveur pour capter son identificateur et le mot de passe associé (même s'il est chiffré ...) et rejouer cet identification plus tard auprès du serveur à pénétrer. (*authentification réciproque*).

Pour lutter contre les maladresses on peut aussi négocier les paramètres de connexions et contrôler que leurs valeurs sont corrects et correspondent à des "profils" convenables. Toute la politique de sécurité doit être contrôlée par un *audit de sécurité.*

Le contrôle d'accès (après authentification réciproque) doit permettre d'éviter les intrusions. On doit aussi s'assurer de l'**intégrité** des données en transit (ou des données et programmes stockés si le contrôle d'accès est inefficace). Pour cela chaque unité de données est dotée d'une *clé d'intégrité* établie de manière secrète de manière à pouvoir vérifier si des données ont été modifiées et à les détruire dans ce cas.

Même en cas de transfert protégé, les correspondants peuvent ne pas être de bonne fois et renier leur identité. Il faut établir des fonctions de **non-répudiation d'origine** et de **non-répudiation de remise** de manière à ce qu'un système ne puisse nier avoir émis un message ou l'avoir reçu. Ces fonctions mettent en oeuvre des mécanismes de **signature** et de **notarisation**.

Toutes ces fonctions reposent sur le chiffrement de données: clés d'accès, clés d'authentification, clés d'intégrité, signatures ou données elles-mêmes.

On peut utiliser des codes à clé privée ou à clé publique.

Dans le premier cas, **clé privée**, la clé secrète est partagée par la source et le collecteur. Cette clé partagée est un élément de fragilité du système car elle est vulnérable durant son transport et son stockage en plusieurs endroits.

On peut utiliser dans ce cas le code DES: Data Encryption Standard (qui peut poser des problèmes légaux dans un cadre international).

Dans le second cas, **clé publique**, la clé de codage est décomposée en une clé secrète, créée et stockée au collecteur et une clé publique qui est transmise (sans protection) à toutes les sources possibles. Une source chiffre ses données avec cette clé publique et transmet le message chiffré au collecteur. Seul celui-ci peut le déchiffrer grâce à sa clé secrète. La connaissance du message chiffré et de la clé publique ne permet pas de décrypter le message en un temps raisonnable ou de reconstituer la clé secrète même en connaissant le message clair.

A cet effet on peut, par exemple, utiliser le code RSA (Rivest-Shamir-Adleman) basé sur la décomposition en deux nombres premiers d'un nombre-produit très grand (100 ou 200 chiffres).

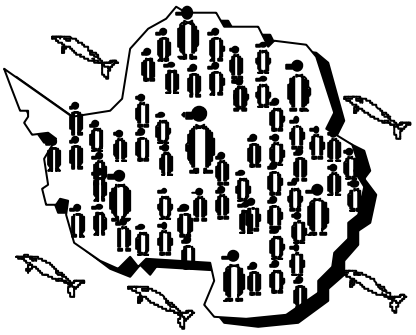
Les algorithmes nécessaires au chiffrement et déchiffrement sont très longs. Ce type de code est essentiellement applicable aux clés d'authentification, aux clés d'intégrité ou aux signatures.

Ce type de code peut aussi être utilisé pour transporter de manière sûre les clés privées (secrètes) du code DES et permet de les changer fréquemment.

2.3.6. Administrer le réseau (informatique)

Dans les entreprise, depuis quelques années, les réseaux ont subi une mutation importante: la fédération en un **réseau d'entreprise** unique de plusieurs réseaux locaux ou étendu(s). Chacun de ces (sous-)réseaux était petit et facilement maîtrisable et il pouvait être géré par un utilisateur particulier ou un ingénieur système.

L'interconnexion de ces sous-réseaux en un réseau d'entreprise de grande taille, réparti sur de nombreux sites et fortement hétérogène a posé de nouveaux problèmes, souvent très complexes. Il est donc nécessaire de mettre en place une administration de réseaux constitué d'une équipe de responsables munis des outils nécessaires à une gestion efficace du réseau.



Cette équipe, **au service des utilisateurs**, doit assumer plusieurs tâches :

• gérer les utilisateurs et les ressources. Chaque **utilisateur** doit être identifié avec son niveau de responsabilité et les équipements ou logiciels qu'il peut utiliser (ou créer ou supprimer ...).
Toutes les **ressources** doivent être identifiées et localisées. Il s'agit non seulement des ressources de télécommunications matérielles ou logicielles mais aussi des ressources informatiques partagées que l'on peut atteindre à travers le réseau (disques, imprimantes, bases de données, etc.). Dans l'esprit des utilisateurs ces ressources sont parties intégrantes du réseau et ne peuvent en être dissociées.
• assurer correctement une **maintenance** curative et une maintenance évolutive du réseau

• s'efforcer de fournir les meilleures **performances** possibles en adaptant les procédures de routage, en ajoutant de nouvelles liaisions ou en adaptant la topologie du réseau.
• évaluer les **coûts** de fonctionnement et d'ammortissement des équipements et répartir des coûts entre les utilisateurs.
• Contrôler la mise en oeuvre des mécanismes de **sûreté**, délivrer les droits d'accès aux ressources, etc.

L'accomplissement de ces tâches repose sur une connaissance de la configuration statique et dynamique du réseau et des activités qui s'y déroulent. Les informations nécessaires doivent être colléctées sur les différents équipements et mémorisées dans une base de données administrative (MIB : Management Administration Base), souvent répartie.