Support de cours Réseaux

Dr Yani-Athmane BENNAI

Université de Béjaia

Faculté des Sciences Exactes

Département d'Informatique

27 octobre 2023

Table des matières

Table des figures

Chapitre 1

Chapitre 1 : Rappels et couche Liaison de données

1.1 Définition

Un réseau informatique peut être défini comme un système composé d'appareils inter-connectés, tels que des ordinateurs, routeurs et commutateurs, qui sont capables de communiquer entre eux en utilisant des protocoles de communication spécifiques. Ces protocoles sont essentiellement des règles et des conventions qui permettent aux appareils du réseau de s'entendre sur la manière dont l'information doit être échangée.

1.2 Objectifs des réseaux

Les réseaux informatiques sont essentiels dans le monde d'aujourd'hui pour plusieurs raisons fondamentales. Ils permettent la communication, le partage de ressources et la collaboration de manière efficace. Voici quelques-unes des principales raisons pour lesquelles les réseaux existent :

1. Partage de fichiers : Les réseaux permettent aux utilisateurs de partager des fichiers de données de manière fluide. Que ce soit au sein d'une entreprise, d'une université ou à travers Internet, la mise en réseau d'ordinateurs facilite le transfert de documents, de médias et d'autres informations.
2. Partage de matériel : Un autre avantage majeur des réseaux est la possibilité de partager du matériel. Les utilisateurs peuvent connecter des périphériques tels que des imprimantes, des scanners, des disques durs et d'autres équipements, ce qui maximise l'utilisation des ressources tout en réduisant les coûts.
3. Communication : Les réseaux sont un moyen puissant de communication. Les utilisateurs peuvent échanger des informations via des e-mails, des groupes de discussion, des vidéo-conférences, des jeux en ligne, etc. Cela favorise la collaboration, la coordination des projets et la connectivité entre les individus, quel que soit leur emplacement géographique.

1.3 Les architectures

Il existe deux principales architectures de réseaux informatiques, chacune avec ses propres caractéristiques distinctes :

1. Architecture entre pairs "Peer to Peer" (P2P)

   Dans une architecture P2P, tous les ordinateurs connectés au réseau ont des pouvoirs équivalents. Il n'y a pas de serveur central qui coordonne les activités. Chaque appareil peut à la fois agir en tant que client et serveur. Cette structure favorise la décentralisation et l'interconnexion directe entre les pairs.
   Un exemple courant de l'architecture P2P est le système de partage de fichiers "Torrent". Dans ce cas, les participants du réseau sont répartis en deux catégories :
   Peers : Ce sont les utilisateurs qui n'ont pas encore téléchargé l'intégralité du fichier en question.
   Seeders : Les seeders ont déjà téléchargé le fichier complet ou sont la source initiale du partage. Lorsqu'un utilisateur termine le téléchargement, il peut également devenir un seeder pour partager le fichier avec d'autres utilisateurs. Il est à noter qu'un seeder maintient un fichier complet et passe d'un état de téléchargement à un état de partage, appelé "seeding".

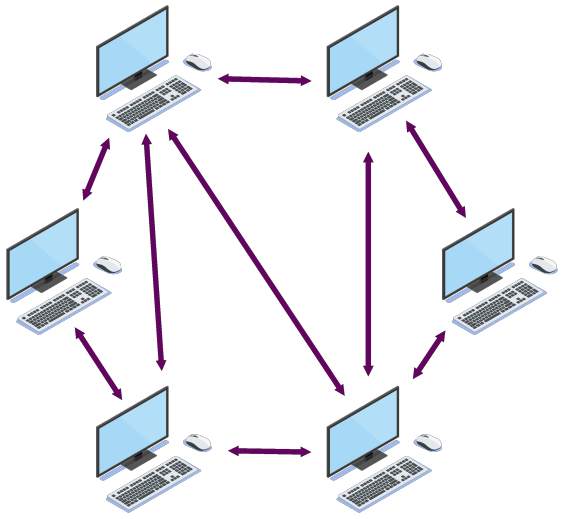

FIGURE 1.1 – Architecture pair à pair

2. Architecture client-serveur

   Dans l'architecture client-serveur, un serveur informatique central offre des services et des ressources qui sont accessibles via le réseau. Le serveur est une machine spécialisée dans l'exécution d'opérations en réponse aux requêtes émises par d'autres ordinateurs, appelés "clients".
   Un exemple courant d'architecture client-serveur est le fonctionnement d'un navigateur Web. Le client, qui est généralement un navigateur Web, envoie une re-

quête au serveur Web pour obtenir le contenu d'une page. Le serveur répond en renvoyant le résultat de la requête, c'est-à-dire la page Web demandée.
Cette architecture est couramment utilisée pour fournir des services en ligne, tels que la consultation de sites Web, la gestion de bases de données, la messagerie électronique, et de nombreux autres services réseau.

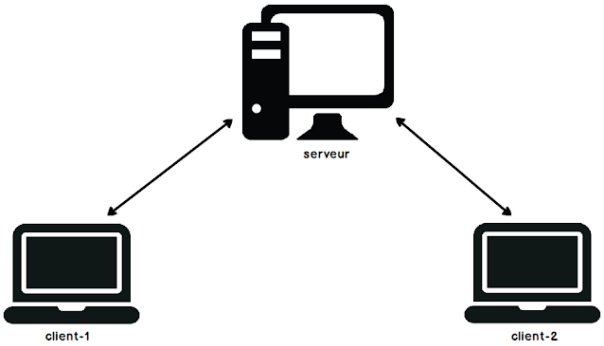

FIGURE 1.2 – Architecture Client/Serveur

1.4 Le modèle OSI

L'Organisation internationale de normalisation (ISO) est une organisation multinationale qui joue un rôle essentiel dans l'établissement de normes mondiales. Elle regroupe la participation de 75% des pays du monde, ce qui en fait une institution internationale majeure. L'objectif principal de l'ISO est de créer des accords et des standards mondiaux dans divers domaines, y compris les normes de communication dans les réseaux.

L'une de ces normes les plus importantes est le modèle de référence des systèmes ouverts OSI (Open Systems Interconnection), qui a été conçu pour la première fois à la fin des années 1970. Le modèle OSI est un cadre fondamental dans le domaine des réseaux informatiques, car il fournit un ensemble de protocoles qui permettent à différents systèmes de communiquer entre eux sans avoir à modifier leur architecture de base.

Ce modèle se compose de sept couches interdépendantes qui facilitent le transfert d'informations à travers un réseau. Chaque couche du modèle a un rôle spécifique, allant de la gestion de la connexion physique à l'interaction des applications. Le modèle OSI est essentiel pour comprendre comment les données sont transmises à travers un réseau et comment différents équipements et logiciels peuvent interagir de manière cohérente.

Le but ultime du modèle OSI est de garantir que deux machines peuvent se comprendre et interagir efficacement, même si elles sont fabriquées par des fournisseurs différents et utilisent des technologies diverses. Cela favorise la communication transparente à l'échelle mondiale.

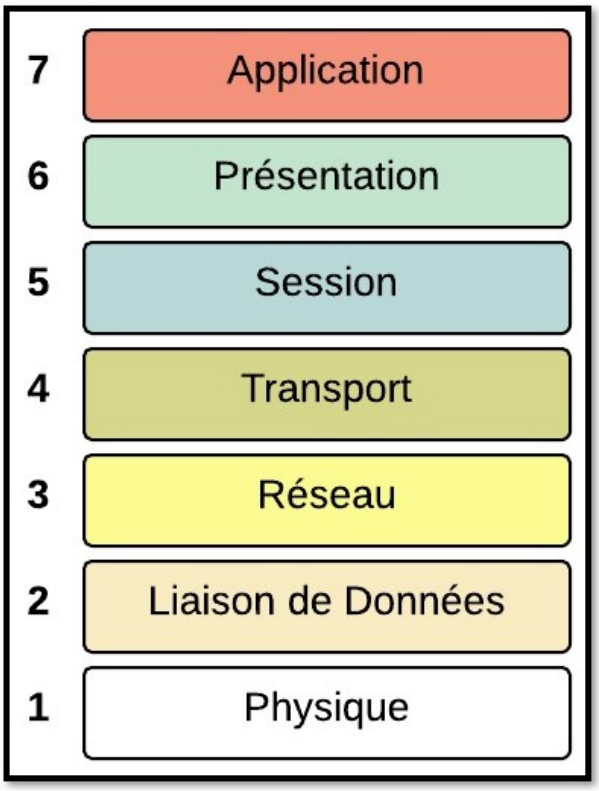

FIGURE 1.3 – Le modèle OSI

Avant les années 1990, le modèle OSI était dominant dans la littérature sur la communication de données. À l'époque, il suscitait de grandes attentes selon lesquels il deviendrait la norme utilisée pour les communications de données dans le monde entier.

Cependant, ces attentes ne se sont pas réalisées. Le modèle OSI n'a pas réussi à devenir la norme prédominante. Au lieu de cela, c'est le modèle TCP/IP (Transmission Control Protocol/Internet Protocol) qui a émergé comme l'architecture commerciale dominante dans le domaine des réseaux. La popularité du modèle TCP/IP a été renforcée par le fait qu'il était largement utilisé et testé sur Internet, qui connaissait une croissance exponentielle à l'époque.

Le modèle OSI, malgré ses avantages, n'a jamais été réelement mis en œuvre à grande échelle. Il a été largement perçu comme compliqué et coûteux à implémenter. En conséquence, les acteurs de l'industrie et les organismes de normalisation ont adopté le modèle TCP/IP, qui était plus adapté au progrès et à la diversité des besoins de communication.

Ainsi, le modèle TCP/IP est maintenant l'architecture dominante, grâce à son utilisation répandue sur Internet.

1.5 Tout d'abord, qu'est ce qu'un protocole ?

Imaginons deux personnes qui se rencontrent pour la première fois et souhaitent établir une communication formelle. Dans ce contexte, le protocole de serrage de main joue un rôle crucial. Il définit une séquence claire d'événements :

Personne A tend la main vers Personne B.

Personne B reçoit la main de Personne A et la serre fermement.

Personne A fait de même en serrant la main de Personne B.

Une fois cette poignée de main accomplie, les deux personnes savent qu'elles peuvent engager une conversation formelle. Ce protocole de serrage de main assure que les deux individus comprennent exactement comment se comporter et communiquer dès le début de leur rencontre. Il élimine toute ambiguïté et établit un accord sur les normes sociales à suivre.

De manière similaire, les protocoles en réseaux informatiques fonctionnent comme des séquences d'étapes spécifiques pour que les appareils puissent établir, maintenir et conclure des connexions. Ces protocoles sont essentiels pour échanger des données de manière fluide et sans ambiguïté. Ils assurent que les appareils dans un réseau comprennent comment communiquer entre eux, quels formats utiliser pour les données, comment gérer les erreurs, et bien d'autres aspects. Pour résumer, les protocoles en réseaux informatiques définissent les règles du jeu pour une communication efficace.

Dans le domaine des réseaux informatiques, un protocole est essentiellement un ensemble de règles et de conventions qui régissent la manière dont différents appareils et systèmes communiquent entre eux de manière organisée et standardisée. Ces règles sont cruciales pour garantir que la communication se déroule efficacement et sans ambiguïté.

Un protocole peut définir divers aspects de la communication, notamment :

— Échange d'information : Il précise le format dans lequel les données doivent être structurées pour être échangées entre les appareils. Cela inclut des détails sur la manière dont les informations sont codées et présentées.

— Séquence d'événements : Il décrit l'ordre dans lequel les étapes de communication doivent se dérouler. Il établit une séquence cohérente d'actions à suivre, de l'initiation de la connexion à la conclusion de la communication.

— Gestion des erreurs : Les protocoles définissent également comment les erreurs doivent être détectées et gérées pendant la communication. Cela inclut des mécanismes pour la retransmission de données en cas de perte ou de corruption.

1.6 La pile de protocoles TCP/IP

Le modèle TCP/IP est un ensemble de protocoles inter-connectés qui jouent un rôle central dans la communication et l'échange de données sur les réseaux informatiques. Cette pile de protocoles est fondamentale pour le fonctionnement d'Internet et est également utilisée dans les réseaux locaux (LAN).

La pile de protocoles TCP/IP est organisée en plusieurs couches, chacune ayant des responsabilités spécifiques dans le processus de transmission des données. Chaque couche est composée de protocoles, qui sont des ensembles de règles et de normes conçues pour des tâches particulières.

Les couches du modèle TCP/IP travaillent en synergie pour garantir une communication efficace et fiable.

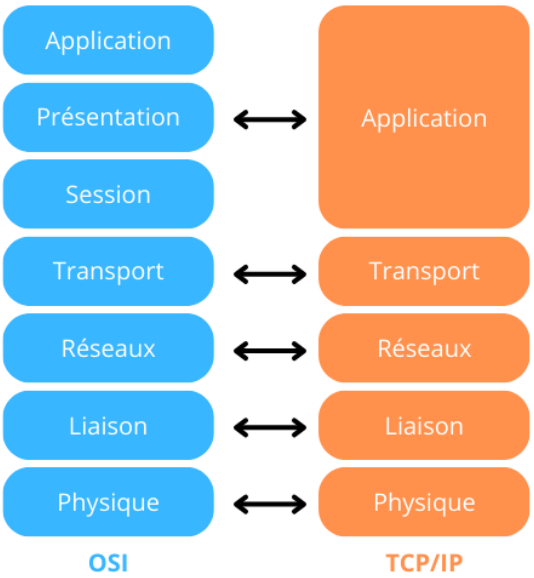

FIGURE 1.4 – modèle OSI vs TCP/IP

L'une des différences les plus notables est que les couches de Session et de Présentation du modèle OSI sont absentes dans le modèle TCP/IP. Au lieu de diviser ces fonctions en couches distinctes, le modèle TCP/IP les regroupe sous la seule couche d'Application.

Cela signifie que dans le modèle TCP/IP, la couche Application englobe les responsabilités de la couche Application, de la couche Présentation et de la couche Session du modèle OSI. Cette approche simplifiée permet de mieux refléter la structure des applications réelles utilisées sur Internet.

Deux raisons fondamentales ont conduit à la décision d'organiser le modèle TCP/IP différemment de celui de l'OSI :

— Multiples Protocoles de Couche de Transport : Le modèle TCP/IP est utilisé dans

un contexte où plusieurs protocoles de couche de transport peuvent coexister. Par exemple, outre le TCP (Transmission Control Protocol) largement utilisé, il existe également l'UDP (User Datagram Protocol). Ces protocoles de couche de transport offrent des fonctionnalités distinctes et sont adaptés à différents types d'applications. Certains d'entre eux intègrent des éléments de contrôle de session. Cette flexibilité permet au modèle TCP/IP de se conformer à divers besoins de communication, éliminant ainsi le besoin de la couche de session.

— Couche d'Application Flexible : La couche d'Application du modèle TCP/IP est conçue pour être plus souple et adaptable. Contrairement au modèle OSI, qui divise strictement les responsabilités entre les couches de Session, de Présentation et d'Application, le modèle TCP/IP permet aux applications de gérer elles-mêmes les aspects liés à la session et à la présentation. Ainsi, si une application spécifique nécessite des fonctionnalités spécifiques de ces couches (qui ne sont pas disponibles dans TCP/IP), elle peut les intégrer directement dans son propre logiciel, sans avoir besoin de s'appuyer sur des couches supplémentaires. Cela favorise la création de nombreuses applications différentes pour communiquer sur le réseau.

1.7 L'encapsulation

L'encapsulation des données est un concept clé dans les réseaux informatiques. Il s'agit du processus d'ajout d'informations supplémentaires, appelées en-têtes, à un élément de données lorsqu'il est transmis dans le modèle TCP/IP, de la source à la destination. Ces en-têtes confèrent à la donnée des caractéristiques spécifiques nécessaires à sa transmission correcte.

Ce processus commence du côté de l'expéditeur, à la couche Application, et se poursuit à travers les différentes couches jusqu'à atteindre la couche Physique. À chaque couche, des informations de protocole supplémentaires sont ajoutées à l'en-tête ou au pied de page des données. Chaque couche reçoit les informations encapsulées de la couche précédente, puis ajoute ses propres données pour encapsuler davantage l'information, avant de les transmettre à la couche suivante.

L'encapsulation des données est un mécanisme essentiel pour plusieurs raisons :

— Détection des Erreurs : Les informations ajoutées à chaque couche peuvent inclure des mécanismes de contrôle d'erreur pour garantir que les données sont reçues sans changement.

— Contrôle du Flux : Les en-têtes peuvent contenir des informations sur la gestion du débit, aidant à contrôler la vitesse à laquelle les données sont transmises.

— Routage des Données : Les informations d'en-tête indiquent souvent le chemin à suivre pour atteindre la destination, en facilitant le routage efficace des données.

— Sécurité et Authentification : Des informations de sécurité et d'authentification peuvent être ajoutées aux données pour garantir leur intégrité et leur transmission.

D'un autre côté, la désencapsulation des données est le processus inverse, qui se produit du côté du récepteur, à la couche Application. Les informations d'en-tête et de fin

ajoutées lors de l'encapsulation sont retirées pour obtenir les données originales.

Ce qu'il faut retenir est que les données sont encapsulées du côté de l'expéditeur dans chaque couche pour leur transmission, puis désencapsulées du côté du destinataire dans la même couche, permettant aux informations de parcourir avec succès l'ensemble du modèle TCP/IP tout en maintenant leur intégrité, leur sécurité et leur routage approprié. Ce processus d'encapsulation et de désencapsulation est essentiel pour garantir la communication fiable à travers les réseaux informatiques.

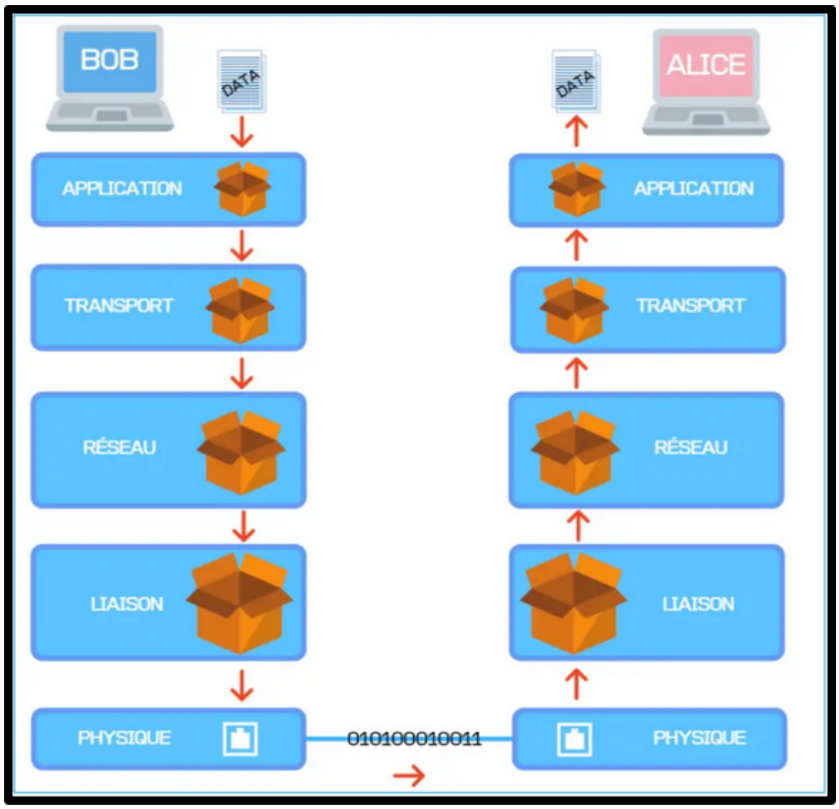

FIGURE 1.5 – Cheminement des données de l'émetteur vers le récepteur

1.8 La couche Physique

La couche physique est la couche la plus basse du modèle TCP/IP. Comme son nom l'indique, elle est responsable de tout ce qui est physique dans le réseau, c'est-à-dire des éléments matériels et de la transmission des signaux physiques. Cette couche gère le transport des données d'un point à un autre sur le réseau, sans se soucier du contenu ou de la signification des données elles-mêmes.

Les principales responsabilités de la couche physique incluent :

— Transmission des Signaux : Elle assure la transmission des signaux électriques, optiques ou radioélectriques qui portent les données à travers les câbles, les fibres optiques, les ondes radio, ou d'autres médias de transmission.

— Caractéristiques Physiques : Elle gère les aspects physiques des connexions réseau, tels que les types de câbles, les connecteurs, les niveaux de tension, les fréquences, et d'autres spécifications techniques.

— Topologie du Réseau : Elle définit la topologie physique du réseau, c'est-à-dire la manière dont les appareils sont physiquement connectés les uns aux autres, que ce soit en étoile, en bus, en anneau, ou d'autres configurations.

La couche physique du modèle TCP/IP est responsable de la manière dont les données sont physiquement transmises d'un ordinateur à un autre au sein d'un réseau. Elle gère les aspects concrets de la transmission, tels que l'utilisation de câbles, de signaux électriques ou d'ondes radio pour envoyer et recevoir les informations. En d'autres termes, cette couche se concentre sur le moyen matériel par lequel les données sont déplacées entre les appareils.

Pour accomplir cette tâche, la couche physique fait usage de divers supports de transmission. Ces supports peuvent inclure des câbles en cuivre, des fibres optiques, des liaisons sans fil, des ondes radio, et d'autres technologies de transmission adaptées à différentes situations.

Dans le modèle TCP/IP, il est important de noter que contrairement à certaines autres couches, il n'y a pas de protocole spécifique défini pour la couche physique. Au lieu de cela, le modèle TCP/IP est conçu pour être compatible avec différents protocoles standards utilisés au niveau physique. Il offre une flexibilité considérable en permettant l'utilisation de diverses technologies de transmission, telles que les câbles en cuivre, les fibres optiques, le Wi-Fi, etc.

Au niveau de la couche physique, la communication se fait entre deux nœuds, que ce soient des ordinateurs, des routeurs ou d'autres dispositifs. L'unité de communication de base est le bit, la plus petite unité de données, qui peut représenter 0 ou 1. Lorsqu'une connexion est établie entre ces deux nœuds, un flux de bits circule entre eux. Cependant, il est essentiel de noter que la couche physique traite chaque bit individuellement, s'occupant des aspects concrets de la transmission, tels que la modulation des signaux, l'amplification, le câblage ou les ondes radio, pour garantir que les bits sont correctement acheminés d'un point à un autre.

La couche physique dans le modèle TCP/IP agit comme un traducteur essentiel entre les ordinateurs et les supports de transmission. Son rôle est de convertir les données provenant des ordinateurs en signaux physiques compréhensibles pour les supports de transmission, et vice versa. C'est comme si la couche physique utilisait des "astuces" pour que les ordinateurs et les câbles puissent se comprendre et communiquer harmonieusement.

Cette conversion est cruciale car les ordinateurs comprennent et manipulent les données sous forme de bits (0 et 1), tandis que les supports de transmission, tels que les câbles, transmettent ces données sous forme de signaux électriques, optiques ou radioélectriques, en fonction de la technologie physique utilisée.

Ainsi, la couche physique assure que les informations peuvent voyager à travers dif-

férents types de câbles ou de médias de transmission en adaptant les signaux pour correspondre à ce que chaque support physique comprend. Cela garantit que les données sont transmises d'un point à un autre dans un réseau, quels que soient les types de supports utilisés.

Qu'est ce que la topologie d'un réseau ?

La topologie réseau est le schéma ou la structure qui définit comment les appareils informatiques sont inter-connectés au sein d'un réseau, qu'il s'agisse d'un réseau local (LAN) dans une entreprise, d'un réseau étendu (WAN) ou même d'un réseau mondial comme Internet. Elle est essentiellement la "forme" du réseau, déterminant comment les appareils sont positionnés et connectés pour permettre la communication entre eux.

La topologie réseau peut prendre plusieurs formes, chacune ayant des avantages et des inconvénients en fonction des besoins spécifiques du réseau. Voici quelques-unes des topologies réseau les plus courantes :

1. **La topologie en bus** est un type de configuration réseau dans laquelle tous les appareils sont connectés à une seule ligne de communication principale, souvent appelée le "bus". Les données sont transmises le long de cette ligne, et chaque appareil sur le réseau décide s'il doit accepter ou ignorer les données en fonction de son adresse. C'est une topologie simple à mettre en œuvre, mais elle peut être sujette à des collisions de données.

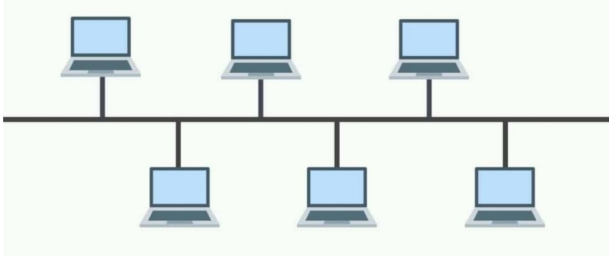

FIGURE 1.6 – Topologie en Bus

Avantages de la topologie en bus :

Facilité d'ajout d'appareils : L'un des avantages majeurs de cette topologie est qu'il est relativement simple d'ajouter une nouvelle machine au réseau. Vous pouvez simplement connecter un câble au bus existant pour ajouter un nouvel appareil, ce qui en fait une option flexible pour les réseaux en croissance.

Moins de longueur de câble : Comparé à certaines autres topologies, la topologie en bus nécessite moins de longueur de câble, car vous n'avez pas besoin de connecter chaque appareil directement à tous les autres. Cela peut réduire les coûts de câblage.

Inconvénients de la topologie en bus :

Vulnérabilité aux pannes de câble : Un inconvénient majeur de la topologie en bus est qu'elle est vulnérable aux pannes de câble. Si le câble principal cesse de fonctionner en raison d'une défaillance ou d'une interruption, tout le réseau peut être mis hors service. Cela peut causer des interruptions significatives dans la communication et nécessiter des efforts pour localiser et réparer la panne.

Collisions de données : Étant donné que tous les appareils partagent la même ligne de communication, il peut y avoir des collisions de données. Cela se produit lorsque deux appareils tentent de transmettre des données en même temps, ce qui peut entraîner des conflits et une perte de données. Pour gérer ces collisions, les réseaux en bus utilisent souvent des protocoles de détection de collision.

2. Dans la **topologie en anneau**, les appareils sont connectés dans un arrangement en forme de cercle fermé, formant un anneau. Les données circulent le long de cet anneau dans un sens spécifique, passant par chaque appareil sur leur chemin jusqu'à ce qu'elles atteignent leur destination. Chaque appareil reçoit les données, les examine, et les transmet au suivant dans l'anneau.

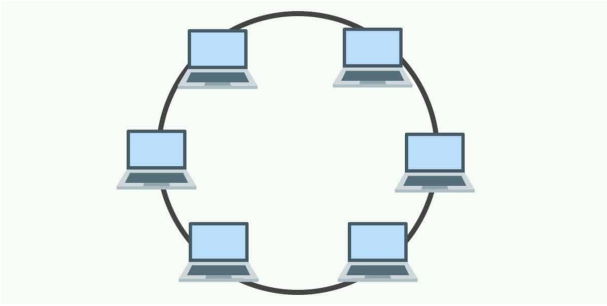

FIGURE 1.7 – Topologie en Anneau

Avantages de la topologie en anneau :
Signal fort : L'un des avantages clés de la topologie en anneau est que le signal de transmission reste fort. Cela est dû au fait que l'anneau est une boucle infinie, et chaque appareil régénère et renforce le signal avant de le transmettre au suivant. Par conséquent, les données restent de haute qualité tout au long de leur voyage dans l'anneau.

Inconvénients de la topologie en anneau :
Vulnérabilité aux pannes d'appareils : Un inconvénient majeur de la topologie en anneau est sa vulnérabilité aux pannes d'appareils. Si une seule machine dans l'anneau cesse de fonctionner, cela peut entraîner une panne de tout le réseau, car les données ne peuvent pas continuer à circuler. La défaillance d'un seul appareil peut donc avoir un impact significatif.

Complexité de l'ajout d'appareils : L'ajout de nouveaux appareils à un réseau en anneau peut être complexe. Il nécessite généralement de couper temporairement l'anneau pour ajouter un nouvel appareil, puis de rétablir la connexion. Cela peut

être fastidieux et nécessiter des compétences techniques.

3. Dans la **topologie en étoile**, tous les appareils du réseau sont connectés à un point central, créant une structure qui ressemble à une étoile. Ce point central peut être un commutateur (switch) ou un concentrateur (hub).

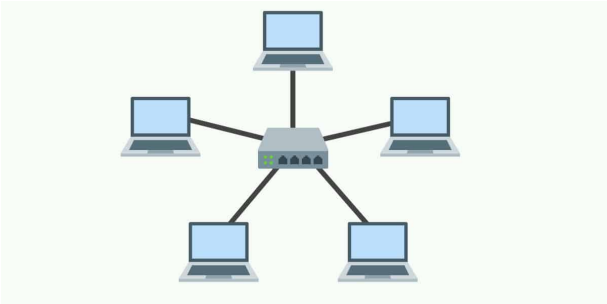

FIGURE 1.8 – Topologie en étoile

Avantages de la topologie en étoile :

Robustesse locale : Un avantage majeur de la topologie en étoile est que les problèmes locaux, tels que des pannes d'appareils individuels, n'arrêtent pas l'ensemble du réseau. Si un appareil échoue ou rencontre un problème, cela n'affecte que cet appareil particulier, tandis que les autres continuent de fonctionner normalement.

Facilité de détection des problèmes : Étant donné que tous les appareils sont reliés au point central (serveur central), il est relativement facile de détecter quelle machine a un problème. Si une machine ne répond pas aux paquets ou présente des problèmes de communication, il est plus simple de localiser l'origine du problème.

Inconvénients de la topologie en étoile :

Coût plus élevé : La mise en place d'une topologie en étoile est généralement plus coûteuse que certaines autres topologies. Cela est dû au fait qu'elle nécessite un serveur central (commutateur ou concentrateur) qui coordonne la communication entre les appareils. L'achat et la maintenance de ce matériel central peuvent représenter un investissement significatif.

Dépendance au serveur central : La topologie en étoile est entièrement dépendante du serveur central. Si le serveur central échoue ou rencontre des problèmes, l'ensemble du réseau peut être hors service. Par conséquent, le bon fonctionnement du serveur central est critique.

4. Dans une **topologie en maillage**, chaque appareil est connecté à tous les autres appareils du réseau. Cette configuration crée un réseau hautement redondant, car il existe de multiples chemins pour que les données atteignent leur destination. En d'autres termes, chaque nœud du réseau peut communiquer directement avec

tous les autres nœuds, ce qui offre une grande flexibilité en matière de routage des données.

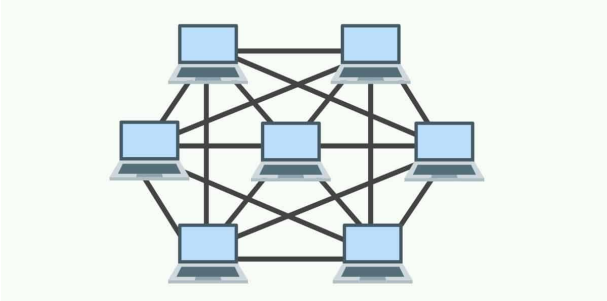

FIGURE 1.9 – Topologie maillée

Avantages de la topologie en maillage :

Adapté aux fortes charges : La topologie en maillage est particulièrement bien adaptée aux environnements à forte charge de trafic. Étant donné que les nœuds se partagent la tâche de transmission des données, la charge est répartie, ce qui permet au réseau de gérer efficacement un grand volume de données.

Sécurité et fiabilité : En raison de sa décentralisation, la topologie en maillage est généralement plus sécurisée que d'autres configurations. En cas de défaillance d'un nœud, la communication peut souvent être réacheminée par d'autres chemins, ce qui rend le réseau résistant aux pannes et difficile à pirater.

Inconvénients de la topologie en maillage :

Coût élevé du câblage : Un inconvénient majeur de la topologie en maillage est son coût élevé. Le câblage de chaque appareil pour qu'il soit connecté à tous les autres peut représenter une grosse dépense, en particulier dans les grands réseaux.

Complexité : La configuration et la gestion d'un réseau en maillage peuvent être complexes, en particulier à grande échelle. La nécessité de gérer de nombreux câbles et connexions peut rendre l'administration du réseau plus exigeante.

1.9 La couche Liaison de données

La couche de liaison de données, contrairement à la couche physique qui gère le transport brut des données, offre une liaison fiable en prenant en charge plusieurs responsabilités cruciales :

1. Découpage des données en trames : La couche de liaison de données découpe les morceaux de données provenant de la couche réseau en petits groupes appelés trames. Ce découpage permet de segmenter les données en unités gérables pour la

transmission.

2. Utilisation d'adresses MAC : Chaque trame réseau contient un en-tête qui indique l'adresse MAC du destinataire. Cela permet aux appareils du réseau de déterminer l'émetteur et le destinataire de la trame. Les adresses MAC sont essentielles pour acheminer efficacement les trames.

3. Contrôle de flux : La couche de liaison de données gère le contrôle de flux pour garantir que le destinataire peut traiter les données à un rythme approprié. Si le destinataire ne peut pas lire rapidement les données, il communique au "parleur" d'arrêter temporairement la transmission pour éviter de le submerger.

4. Contrôle d'erreurs : En cas de problème avec les données, comme des erreurs de transmission ou des pertes de données en cours de route, la couche de liaison de données intervient pour tenter de les renvoyer. Cela garantit une communication fiable en corrigeant les éventuelles erreurs.

5. Contrôle d'accès : Lorsque plusieurs appareils souhaitent communiquer en même temps, la couche de liaison de données aide à déterminer l'ordre de priorité pour la transmission. Cela réduit les collisions et permet une utilisation efficace du réseau.

Si les supports de transmission étaient parfaits (pas de pertes de données, délais courts, débit infini, etc.) on n'aurait pas eu besoin de la couche liaison. Sauf que les réseaux sont imparfaits, les services de cette couches sont donc très importants.

Mais quels types de services existent dans cette couche ?

Service sans connexion et sans accusé de réception :
Dans ce service, la machine source envoie ses données à une machine de destination sans établir de connexion entre elles et sans attendre d'accusé de réception. Il n'y a aucune garantie que les données atteindront leur destination ni que toutes les trames seront reçues. Si une trame est perdue en cours de route, il n'y a aucun mécanisme pour la récupérer. Ce type de service est couramment utilisé dans des applications où le temps réel est plus important que la qualité de la transmission, telles que la Voix sur IP (Viber, WhatsApp, Discord, etc.).

Service sans connexion et avec accusé de réception :
Ce service est plus fiable que le précédent, car chaque trame envoyée est acquittée par le récepteur. Il n'y a toujours pas de connexion établie entre les deux machines communicantes, mais si une trame n'est pas acquittée dans un certain délai, elle sera renvoyée. Ce service est particulièrement utile dans des canaux peu fiables, tels que les liaisons sans fil 802.11 (WIFI), où la perte de trames est fréquente.

Service avec connexion et avec accusé de réception :
Avant d'envoyer des données, une connexion logique est établie entre les deux machines communicantes. Chaque trame envoyée est numérotée, ce qui garantit que chaque trame

est reçue une seule fois et dans l'ordre. Ce service est utilisé dans les transmissions longues et non fiables, comme les liaisons par satellite, par exemple le satellite Nilesat pour la télévision.

les trames dans la couche de liaison de données sont structurées avec des en-têtes (headers) et des pieds de page (footers) qui contiennent des informations essentielles pour garantir une communication fiable et efficace.

Côté émetteur :
La couche de liaison de données reçoit des paquets de données de la couche réseau, qui sont plus gros.
Elle convertit ces paquets en trames en ajoutant des en-têtes et des pieds de page. Les en-têtes contiennent des informations cruciales, telles que l'adresse MAC du destinataire et d'autres métadonnées de contrôle.
Une fois les trames prêtes, la couche de liaison de données les transmet à la couche physique, qui est responsable de leur transmission sur le support physique.

Côté récepteur :
La couche de liaison de données reçoit des bits de la couche physique, qui sont la forme brute des trames.
Elle effectue le processus inverse en convertissant ces bits en trames en analysant les en-têtes et les pieds de page pour extraire les données pertinentes.
Les trames ainsi reconstituées sont ensuite transmises à la couche réseau, où elles sont traitées comme des paquets de données prêts à être acheminés vers leur destination finale.

1.9.1 Le découpage en trames

Lorsque nous manipulons des données, il est important de savoir que des erreurs peuvent survenir en cours de transmission. Des altérations accidentelles peuvent se produire, telles que des bits qui changent de valeur ou qui sont manquants. Cependant, pour garantir l'intégrité des données, la couche de liaison de données joue un rôle crucial en détectant et, dans certains cas, en corrigeant ces erreurs.

Elle accomplit cette tâche en divisant les données en morceaux distincts appelés trames (frames). Chaque trame est équipée d'un mécanisme de vérification d'erreur qui permet à la couche de liaison de données de repérer toute altération indésirable survenue pendant la transmission. Une fois détectée, cette couche peut prendre des mesures pour corriger ces erreurs si les mécanismes de détection et de correction des erreurs sont en place.

La division des données en trames, combinée à l'ajout d'en-têtes et de vérifications, présente certains avantages :

Détection d'erreurs : Chaque trame commence généralement par un en-tête qui contient des informations essentielles, y compris des bits de contrôle pour la détection d'erreurs. Ces bits de contrôle sont calculés en fonction du contenu de la trame. Lorsque la trame est reçue, les destinataires comparent les bits de contrôle calculés avec ceux reçus. Si ces

bits ne correspondent pas, cela indique une erreur dans la trame. La couche de liaison de données peut alors demander une retransmission de la trame.

Correction d'erreurs (si possible) : En plus de la détection d'erreurs, certaines trames peuvent inclure des mécanismes de correction d'erreurs. Ces mécanismes permettent de corriger automatiquement les erreurs mineures sans nécessiter une retransmission. Cela améliore l'efficacité de la transmission en évitant des retards inutiles dus à des erreurs mineures.

Transmission individuelle : Chaque trame est transmise individuellement sur le canal de communication. En cas d'erreur sur une trame, seules les données de cette trame sont affectées. Les autres trames ne sont pas impactées, ce qui garantit une transmission fiable et continue des données. Cette isolation des erreurs est essentielle pour minimiser les perturbations dans le réseau.

Une fois que les données ont été divisées en trames, il est essentiel de les délimiter de manière à ce que les destinataires puissent les séparer correctement. Cela équivaut à mettre des étiquettes sur des boîtes pour identifier leur contenu. Il existe trois techniques principales pour délimiter les trames :

En utilisant directement la taille des trames
Remplissage d'octets
Remplissage de Bits

Taille des trames (Comptage d'octets)

La première technique consiste à délimiter les trames en fonction de leur taille, cela peut être comparée à la façon dont un message vocal est découpé en segments avant d'être envoyé. Cela fonctionne de la manière suivante :

Imaginons que vous envoyiez un message vocal à un ami sur une application de messagerie comme Messenger. Au lieu d'envoyer l'enregistrement vocal en une seule fois, vous divisez l'enregistrement en petits segments, et chaque segment a une durée en secondes prédéfinie marquée dessus. Par exemple, vous pouvez découper l'enregistrement en segments de 10 secondes chacun. (mais ce n'est pas obligatoire que ces segments aient la même taille).

De cette manière, votre ami sait à quoi s'attendre. Lorsqu'il reçoit les segments, il peut les lire l'un après l'autre. Les segments sont délimités par leur taille, c'est-à-dire par la durée préétablie. Cela rend la réception et la lecture du message vocale plus organisées, car votre ami sait quand un segment se termine et quand le suivant commence. De plus, cela permet une meilleure gestion des données et de la bande passante, car les segments sont transmis individuellement.

la technique de délimitation des trames par la taille fonctionne en ajoutant une indication de "taille" au début de chaque trame, spécifiant combien d'octets sont inclus dans cette trame. Cette information est essentielle pour que le destinataire puisse lire correctement les données et savoir où se termine la trame. Voici comment cela fonctionne :

Ajout d'une indication de taille : Chaque trame commence par une partie de l'en-tête qui indique la taille de la trame, généralement exprimée en nombre d'octets. Cette information est essentielle pour déterminer où se termine la trame et où commence la suivante.

Lecture de l'indication de taille : Lorsque le destinataire reçoit la trame, il lit l'indication de taille dans l'en-tête pour savoir combien d'octets suivent dans la trame. Cela aide à garder les données organisées et à séparer correctement les trames.

Cependant, il existe un problème potentiel avec cette méthode. Si l'indication de taille est reçue de manière incorrecte en cours de transmission, le destinataire peut se désynchroniser par rapport à la transmission. Cela signifie que le destinataire pourrait interpréter incorrectement la taille de la trame, ce qui entraînerait des erreurs sur les prochaines trames également.

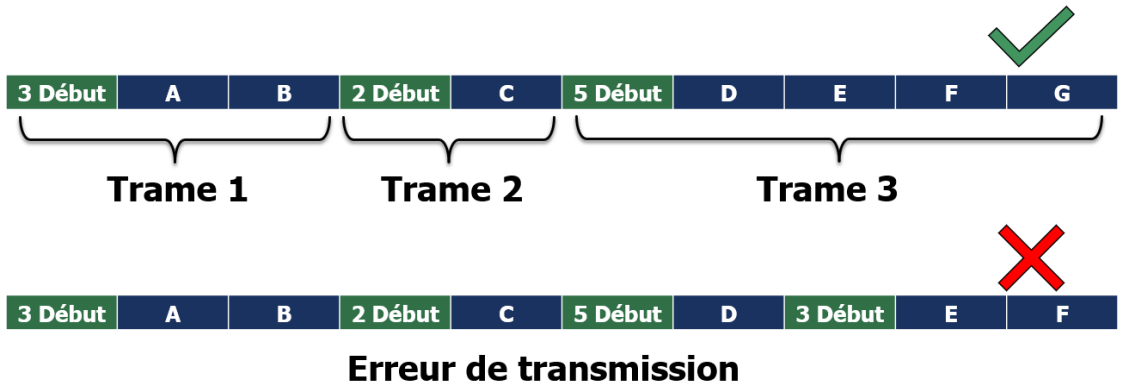

FIGURE 1.10 – Technique de taille des trames

Remplissage d'octets

La deuxième technique, le remplissage d'octets, est une méthode utilisée pour délimiter les trames en ajoutant des fanions (FAN) spéciaux au début et à la fin de chaque trame. Voici comment cela fonctionne :

Fanions au début et à la fin : Chaque trame commence par un fanion (01111110) et se termine également par un fanion (01111110). Ces fanions servent à indiquer le début et la fin de la trame, permettant ainsi de la délimiter clairement.

Octet d'échappement (ESC) : Pour éviter toute confusion entre les vrais fanions et les données qui pourraient ressembler à des fanions, une technique d'échappement est utilisée. Si un fanion ou un octet d'échappement (ESC) est présent dans les données, un octet spécial d'échappement (ESC) est ajouté avant lui. Cela permet de distinguer les fanions de délimitation VS les données .

Élimination des octets d'échappement : Lorsque la trame est reçue, le récepteur élimine les octets d'échappement (ESC), garantissant ainsi que les fanions de délimitation ne sont pas confondus avec les données. Le premier octet d'échappement (ESC) est retiré au niveau du récepteur, laissant uniquement celui qui fait partie des données.

Pour résumer, les fanions de délimitation (FAN) entourent chaque trame, et des octets d'échappement (ESC) sont utilisés pour différencier les vrais fanions de ceux qui pourraient ressembler à des fanions dans les données. Cette technique permet de délimiter clairement les trames et de garantir que les fanions de délimitation ne sont pas confondus avec les données réelles.

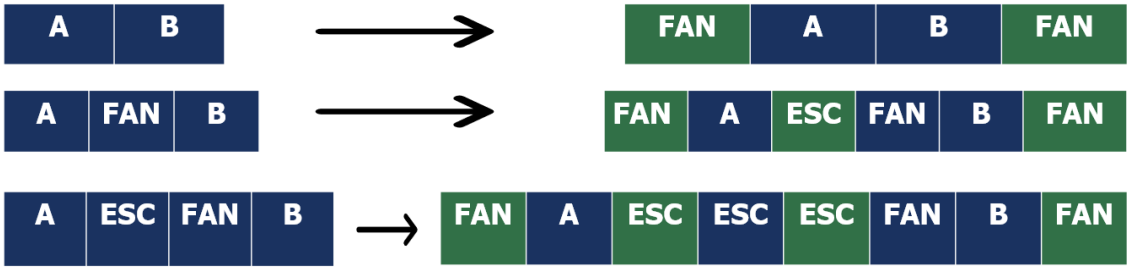

FIGURE 1.11 – Technique de Remplissage d'octets

Remplissage de bits

La troisième technique, le remplissage de bits, est une méthode utilisée pour délimiter les trames en ajoutant un fanion binaire "01111110" au début et à la fin de chaque trame. Voici comment cela fonctionne :

Fanions de début et de fin : Comme dans la technique de remplissage d'octets, chaque trame commence par un fanion "01111110" et se termine également par ce même fanion. Ces fanions servent à indiquer le début et la fin de la trame, la délimitant clairement.

Remplissage de bits : Lorsque l'émetteur détecte cinq "1" consécutifs dans les données, il ajoute un "0" après eux. Cela signifie que si les données contiennent la séquence "11111", l'émetteur la transformera en "111110". Cette transformation est réalisée pour éviter que les données ne soient confondues avec les fanions de début et de fin.

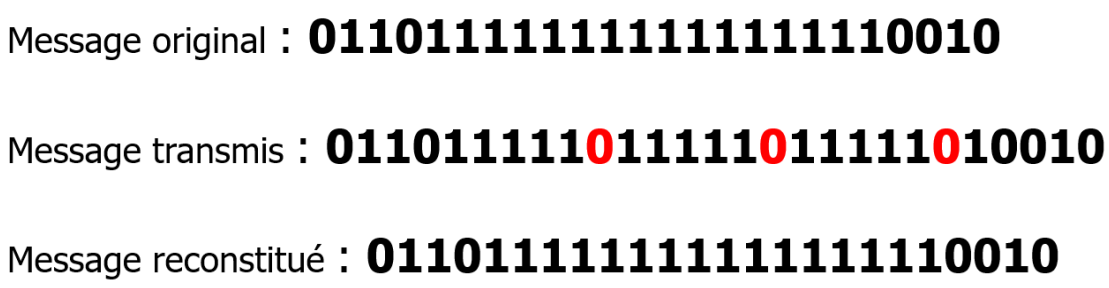

FIGURE 1.12 – Technique de Remplissage de bis

Élimination des bits de remplissage : Lorsque la trame est reçue, le récepteur vérifie la séquence "01111110" pour délimiter la trame (ce sont ls vrais fanions). Il élimine ensuite tous les "0" qui viennent après cinq "1" consécutifs. Par exemple, si le récepteur détecte la séquence "011111010" dans les données, il la transforme en "01111110".

1.9.2 Le contrôle de flux

Le contrôle de flux est une technique essentielle dans la communication de données, car elle garantit un flux de données correct de l'expéditeur au destinataire. Cela revient à assurer que l'expéditeur n'envoie pas de données plus rapidement que ce que le destinataire peut recevoir et traiter. Voici pourquoi le contrôle de flux est important :

Éviter la surcharge : Sans contrôle de flux, l'expéditeur pourrait envoyer des données de manière continue, ce qui risquerait de surcharger le destinataire. Le contrôle de flux permet au destinataire de signaler à l'expéditeur de ralentir ou d'arrêter temporairement l'envoi si sa capacité est atteinte.

Tampon (cache) : Le destinataire a besoin d'un tampon ou d'un cache pour stocker temporairement les données entrantes jusqu'à ce qu'elles puissent être traitées. Le contrôle de flux permet de gérer ce tampon de manière à éviter que les données ne soient perdues ou que le destinataire ne soit submergé.

Le contrôle de flux peut être classé en deux catégories principales, chacune avec une approche différente pour réguler le flux de données :

— Contrôle de flux basé sur les retours (Feedback) : Dans cette approche, l'expéditeur envoie simplement des données au destinataire sans restrictions majeures. Le destinataire renvoie des informations de réception à l'expéditeur pour indiquer qu'il a correctement reçu les données. L'expéditeur utilise ces retours ou confirmations pour décider de la quantité de données qu'il peut envoyer ensuite. Cela signifie que l'expéditeur envoie des données après avoir reçu des confirmations du destinataire, ajustant ainsi le débit en fonction de la capacité de réception du destinataire.

— Contrôle de flux basé sur le débit : Dans cette approche, le mécanisme de contrôle de flux est intégré dans le protocole de communication. Lorsque l'expéditeur envoie des données plus rapidement que ce que le destinataire peut les recevoir, le mécanisme limite automatiquement la vitesse globale de l'expéditeur. Cela se fait sans nécessiter de retour ou de confirmation du destinataire. Cette méthode est plus automatique, car elle ajuste la vitesse d'envoi en fonction des conditions du réseau sans nécessiter une communication directe entre l'expéditeur et le destinataire.

Le contrôle de flux peut être mis en œuvre avec différentes techniques. Voici deux de ces techniques couramment utilisées :

1. **Technique numéro 1 : STOP AND WAIT**

— Cette méthode est la plus simple.
— Le destinataire indique sa disponibilité à recevoir une trame de données.
— L'expéditeur envoie la trame uniquement lorsque l'accusé de réception est reçu pour la trame précédente.
— Ce processus se poursuit jusqu'à ce que l'expéditeur envoie le message de fin de transmission (EOT).
— Seule une trame peut être en transmission à la fois, ce qui peut être inefficace si le délai de propagation est beaucoup plus long que le délai de transmission espéré.

Avantages :
Cette méthode est très simple et facile à implémenter.
Chaque trame est vérifiée et reconnue avec précision.

Inconvénients :
Cette méthode est très lente.
On ne peut envoyer qu'une seule trame à la fois.
Elle gaspille la bande passante du réseau.

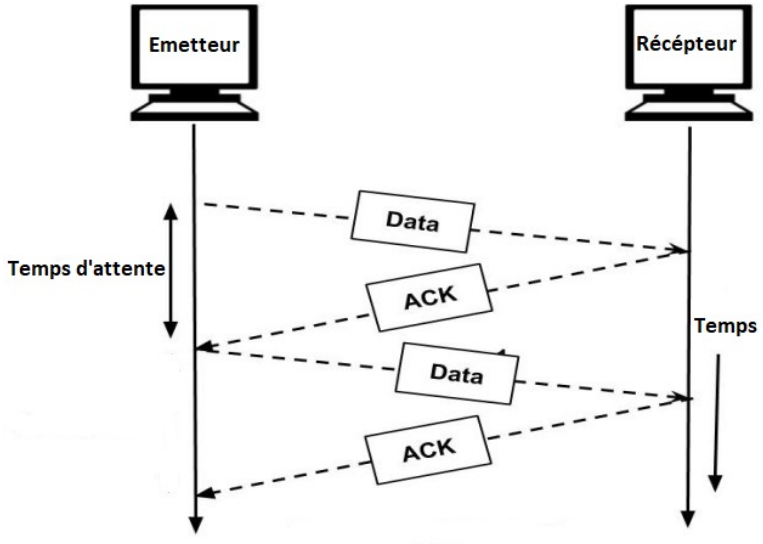

FIGURE 1.13 – Stop and Wait

2. **Technique numéro 2 : Sliding Window (Fenêtre coulissante)**

— C'est un protocole point à point qui suppose que personne d'autre ne tente de communiquer tant que le transfert de trames actuel n'est pas terminé.
— Dans cette méthode, l'expéditeur envoie plusieurs trames avant de recevoir une quelconque confirmation.
— L'expéditeur et le destinataire se mettent d'accord sur le nombre de trames à envoyer avant qu'une confirmation soit nécessaire (ce nombre est souvent appelé "window").

— Cela permet à l'expéditeur d'avoir plusieurs trames non confirmées "en vol" en même temps, ce qui améliore le débit du réseau.
— En fin de compte, l'expéditeur envoie plusieurs trames, mais le destinataire les prend (les traite) une par une.

Avantages :
Donne en général de meilleures performances que la technique 1.
Plusieurs trames peuvent être envoyées en même temps.

Inconvénients :
Plus complexe et demande plus de ressources réseau.
Les trames risquent d'arriver dans le désordre.

1.9.3 Les adresses MAC

Les adresses MAC sont essentielles dans les réseaux informatiques pour attribuer une identification unique à chaque appareil connecté. Voici quelques points importants concernant les adresses MAC :

— Unicité : Chaque carte réseau (ou adaptateur réseau) a une adresse MAC unique au monde. Cela signifie qu'aucun autre appareil ne devrait avoir la même adresse MAC, ce qui garantit une identification individuelle.
— Formats : Les adresses MAC sont généralement représentées sous forme de chiffres hexadécimaux, regroupés en paires de deux (par exemple, 00 :1A :2B :3C :4D :5E). Il existe différentes longueurs d'adresses MAC, y compris des adresses MAC 48 bits (6 octets) et des adresses MAC 64 bits (8 octets).
— Rôles : Les adresses MAC sont utilisées principalement au niveau de la couche liaison de données du modèle OSI. Elles permettent de diriger les trames vers la bonne carte réseau dans un réseau local. Chaque appareil peut avoir plusieurs interfaces réseau, chacune ayant sa propre adresse MAC.
— Indépendance de la couche réseau : Contrairement aux adresses IP, les adresses MAC sont indépendantes de la couche réseau. Cela signifie que même si un appareil change d'adresse IP lorsqu'il se déplace d'un réseau à un autre, son adresse MAC reste inchangée.

Différence en adresse MAC et adresse IP

Adresse MAC :

— Identifiant local : L'adresse MAC est spécifique à un appareil et est utilisée pour l'identifier localement au sein du réseau local (LAN).
— Attribution : Chaque carte réseau a une adresse MAC unique qui lui est assignée par le fabricant. Cette adresse est gravée dans le matériel de la carte et ne change pas.

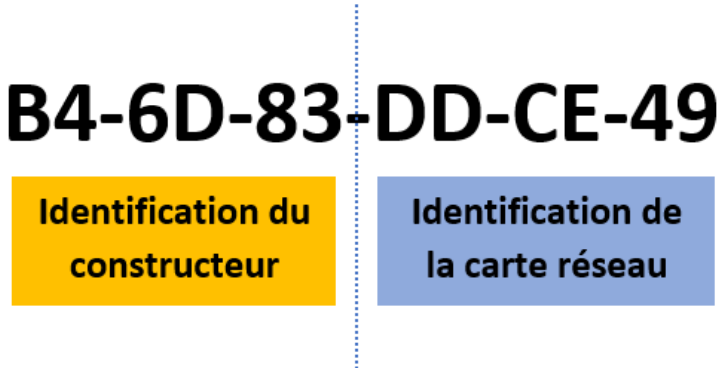

FIGURE 1.14 – Structure d'une adresse MAC

— Fonction : L'adresse MAC est principalement utilisée au niveau de la couche liaison de données pour acheminer les trames au sein du LAN. Elle permet de déterminer quelle carte réseau dans le réseau local est la destinataire d'une trame.

Adresse IP :

— Identifiant global : L'adresse IP est utilisée pour identifier un appareil sur un réseau étendu (WAN), y compris sur Internet. Elle fournit une identité globale et permet la communication entre différents réseaux.
— Attribution : Les adresses IP sont attribuées par des serveurs DHCP (Dynamic Host Configuration Protocol) ou configurées manuellement. Elles peuvent varier lorsque l'appareil change de réseau.
— Fonction : Les adresses IP sont utilisées pour acheminer les paquets de données entre différents réseaux. Elles sont essentielles pour la communication sur Internet et permettent d'acheminer les données sur de longues distances.

1.9.4 Le contrôle d'erreurs

Le contrôle d'erreurs est une pratique essentielle dans les réseaux informatiques pour garantir la précision de la transmission des données, en particulier dans des situations où la qualité des informations est cruciale. Il englobe plusieurs concepts clés.

Tout d'abord, il y a la détection d'erreurs, qui consiste à vérifier si des erreurs se sont produites lors de la transmission des données. Pour ce faire, on ajoute des bits de contrôle, appelés bits de redondance, aux données. En comparant ces bits avec les données reçues, on peut détecter les erreurs.

Dans certaines situations, il ne suffit pas de détecter les erreurs ; il est nécessaire de les corriger. C'est là que les codes correcteurs d'erreurs entrent en jeu. Ils sont conçus pour restaurer les données corrompues à leur état d'origine. Des exemples de ces techniques incluent les codes de Hamming, les codes Reed-Solomon, et les codes convolutifs.

Pour assurer la fiabilité des données dans les protocoles de communication, des mécanismes de retransmission sont utilisés. Par exemple, le protocole TCP (Transmission Control Protocol) utilise des accusés de réception pour s'assurer que les données sont correctement reçues. En cas d'erreur, les données sont renvoyées.

Dans certaines applications, comme la diffusion d'audio ou de vidéo, il est acceptable de tolérer un certain niveau d'erreurs. Cependant, même dans ces cas, des techniques de correction d'erreurs légères peuvent être employées pour améliorer la qualité de la transmission.

Les types d'erreurs

1. Erreur d'un seul bit (Single Bit Error) : C'est lorsque la transmission d'une trame est affectée par la modification d'un seul bit. Cette erreur peut se produire en raison de diverses interférences ou perturbations dans le canal de communication. C'est généralement plus fréquent qu'une erreur en rafale.

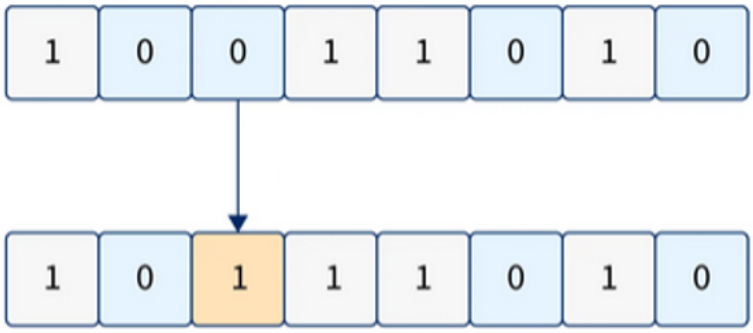

FIGURE 1.15 – Erreur sur un seul bit

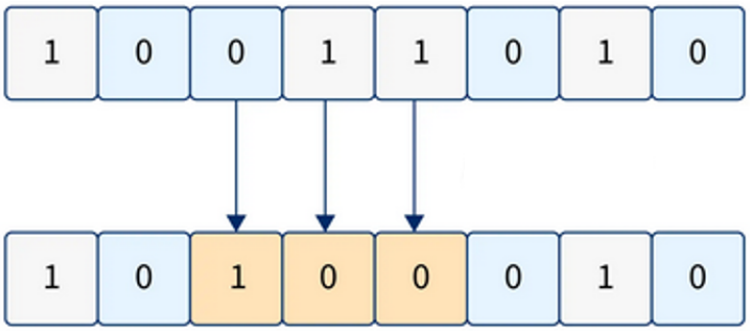

FIGURE 1.16 – Erreur en rafale

2. Erreur en rafale (Burst Error) : Ce type d'erreur est plus grave, car il implique la modification de plusieurs bits dans une trame en même temps. Les erreurs en rafale sont souvent le résultat de perturbations plus longues ou de conditions de transmission altérées. Cela peut être dû à des collisions de paquets, à un mauvais câble, à des interférences avec des apparels ou à un affaiblissement du signal. Les

erreurs en rafale ont plus de risque de gâcher complètement une trame, la rendant inutilisable.

Les techniques de détection d'erreurs

Les techniques de détection d'erreurs sont utilisées pour vérifier si des erreurs se sont produites pendant la transmission de données. Elles ajoutent des données spécifiques aux messages, généralement appelés bits de redondance, qui permettent au destinataire de détecter des altérations dans le message. Voici un aperçu des principales techniques de détection d'erreurs :

Le bit de parité, également connu sous le nom de VRC (Vertical Redundancy Check), est une méthode très utilisée de détection d'erreurs. Elle repose sur l'ajout d'un bit de parité aux données pour garantir un nombre pair ou impair de 1 dans la trame. Cela permet de détecter des erreurs simples en comptant les 1 dans la trame et en vérifiant si le bit de parité correspond à la parité attendue.

Cependant, le bit de parité présente certaines limitations. La principale limitation est qu'il ne peut détecter que des erreurs simples (un seul bit erroné ou un nombre de bits erronés impair). Il est inefficace pour détecter des erreurs plus complexes, telles que des erreurs en rafale, où plusieurs bits peuvent être altérés en même temps.

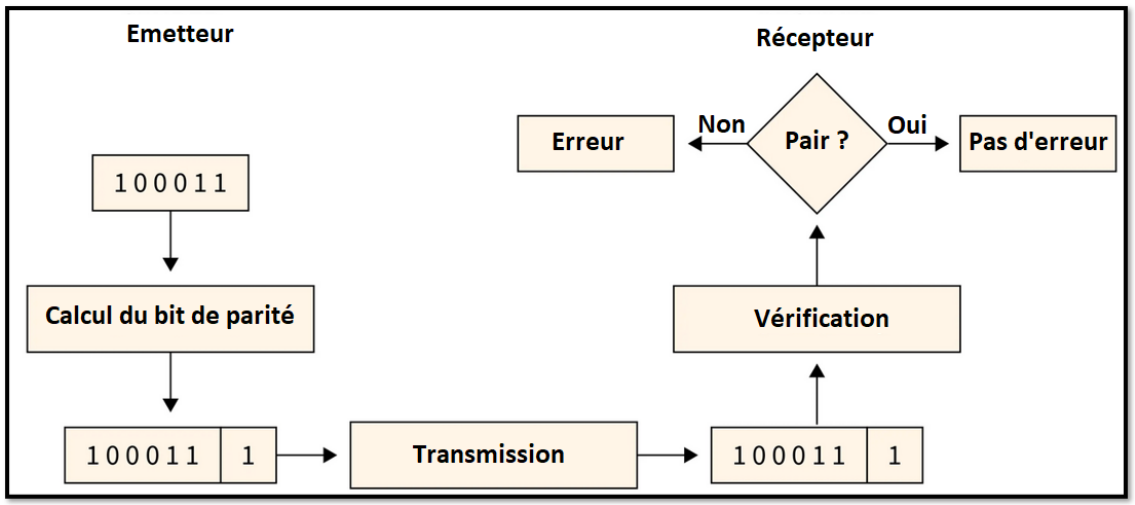

FIGURE 1.17 – Le bit de parité

LRC (Longitudinal Redundancy Check) est un mécanisme de détection d'erreurs qui organise les données en tableaux de lignes et de colonnes, ce qui permet de détecter les erreurs en rafale plus efficacement que d'autres méthodes. Voici comment il fonctionne :

1. Les données à envoyer sont organisées en une matrice de lignes et de colonnes. Chaque colonne de cette matrice a son propre bit de parité.

2. Un bit redondant est ajouté pour chaque colonne de la matrice, qui est utilisé pour détecter les erreurs.
3. Une fois que les données sont transmises, le récepteur reçoit à la fois les bits de données et les bits de parité.
4. Le récepteur utilise ces bits de parité pour vérifier s'il y a des erreurs dans les données. Si une erreur est détectée dans une colonne donnée, cela signifie que l'une des lignes dans cette colonne contient une erreur.

   Le récepteur peut ainsi demander une retransmission de la trame ou prendre d'autres mesures pour corriger l'erreur.

Le LRC est particulièrement efficace pour détecter les erreurs en rafale, car il peut identifier les colonnes entières de données incorrectes.

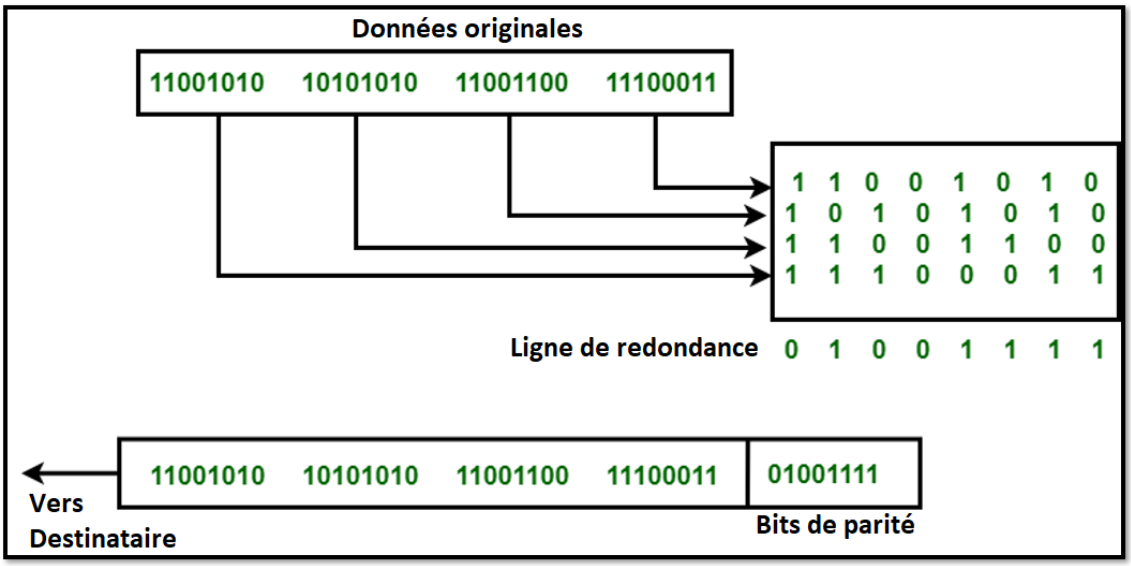

FIGURE 1.18 – LRC

CRC (Cyclic Redundancy Check) est une méthode de détection d'erreurs plus puissante que la parité. Voici comment fonctionne le CRC :

Du côté de l'émetteur : Le CRC est calculé en divisant l'unité de données par un diviseur prédéterminé appelé G(X). G(X) est un polynôme binaire qui représente le diviseur. Pour effectuer la division, des bits de 0 sont ajoutés après le bit de poids faible de la trame pour correspondre au degré de G(X). La division est effectuée, et le reste de cette division est le CRC. Ce CRC est ajouté à l'unité de données pour former une nouvelle trame, que nous appellerons T(X).

Du côté du récepteur : La donnée reçue, c'est-à-dire la trame T(X), est également divisée par le même diviseur G(X). Si le reste de cette division est égal à zéro, cela signifie que les données sont acceptées, car aucune erreur n'a été détectée. Si le reste n'est pas nul, cela indique qu'il y a une erreur, et les données sont rejetées.

Le CRC est basé sur la division binaire et repose sur la propriété que si les données sont correctes, la division à la destination ne doit pas laisser de reste. En revanche, si une erreur est survenue pendant la transmission, le reste de la division sera différent de zéro, ce qui permet de détecter l'erreur.

Correction d'erreurs

Toutes les méthodes que nous venons de voir ne servent qu'a détecter les erreurs. Il existe un code qui peut à la fois détecter et corriger. Mais il ne peut corriger qu'une seule erreur de transmission (un seul Bit). C'est le code de Hamming.

Le code de Hamming est une méthode qui permet à la fois de détecter et de corriger une seule erreur de transmission. Voici comment le code de Hamming fonctionne :

1. La trame à envoyer est composée des données originales et des bits de contrôle.
2. Les bits de contrôle sont placés à des positions spécifiques dans la trame en utilisant les puissances de 2. Par exemple, si la trame originale est "11001010", les bits de contrôle seront placés aux positions 1, 2, 4, et 8 dans la trame étendue, tandis que les autres bits sont les données originales.
3. Pour calculer les valeurs des bits de contrôle, un tableau est utilisé. Les positions des bits de contrôle dans la trame sont utilisées pour déterminer quelles données originales contribuent à chaque bit de contrôle. Les bits de contrôle sont calculés en utilisant une opération de parité.
4. Les bits de contrôle sont ensuite ajoutés à la trame, et la trame étendue est prête à être envoyée.
5. Lorsque la trame est reçue, le destinataire utilise les bits de contrôle pour vérifier s'il y a des erreurs dans la transmission. Si une erreur est détectée, le destinataire peut utiliser les informations des bits de contrôle pour corriger la trame en modifiant le bit erroné.