

# Échanges de données et routage. Exemples.

## I. Contexte et modèle par couches

Objectifs généraux 1 [RESO 1.5.1] Pour un réseau d'envoi de données on peut vouloir:

- De la **fiabilité**: que les données reçues soient les mêmes que les données envoyées, et dans le même **ordre**. Cela peut nécessiter la **détection** et la **correction** d'erreurs.
- Une bonne **allocation de ressources**: éviter le **gaspillage** de ressources ainsi que la **congestion** du réseau.
- Des **capacités d'évolution** du réseau: faciliter l'ajout de nouveaux **usagers** et de nouveaux **usages**.
- Des **garantie de sécurité** telles que la **confidentialité** des données, ou la **disponibilité** des services, y compris **tolérance aux pannes**.

Définition 2 [RESO 2.5.4] La **commutation de circuits** est une technique d'échange de données qui réserve, en exclusivité, un cheminement des données entre la source et la destination.

Exemple 3 La commutation de circuits a historiquement été très utilisée dans les réseaux de téléphonie fixe.

Définition 4 [RESO 2.5.4] L'**envoi de paquets** est une alternative à la commutation de circuits, où les données à envoyer sont découpées en **paquets** de taille prévisible, et le chemin emprunté est déterminé dynamiquement.

Exemple 5 **Internet** utilise de l'envoi par paquets.

Compromis 6 La commutation de circuits est plus facilement fiable, mais peut présenter un gaspillage de bande passante. L'envoi de paquets présente une meilleure tolérance aux pannes, ainsi qu'une gestion plus fine de la congestion du réseau.

Remarque 7 Dans la suite de la leçon, nous ne traiterons que l'envoi de paquets dans le cadre d'internet.

Définition 8 [RESO 1.6.2] Pour faciliter la conception et l'implémentation d'un réseau d'échange de données, ainsi que son

adaptabilité à de nouveaux usages, on le décrit souvent avec un **modèle par couches** qui voit le réseau comme l'interaction entre différents **protocoles** appartenant à différentes **couches**. Chaque couche procure des **services** et un niveau d'**abstraction** à la couche au dessus.

Définition 9 **TCP/IP** est le modèle par couches le plus répandu.

Application	Services rendus directement aux utilisateurs		
Exemples	HTTP	SMTP	NTP
Transport	Offrir des compromis fiabilité/rapidité		
Exemples	TCP		UDP
Internet	Acheminer les paquets entre source et destination		
Exemples	IP		
Liaison	Lien entre deux équipements voisins dans le réseau		
Exemples	Fibre	WiFi	Ethernet

## II. Échanger des données: la couche transport

Définition 10 Les **numéro de port** sont des identifiants permettant de distinguer différents services et différentes connexions au sein d'une même machine (e.g. SMTP et IMAP pour serveur mail).

Définition 11 Les **protocoles de la couche transport** fournissent l'abstraction des numéros de port. Il existe deux protocole possibles pour cette couche, qui offrent différents compromis entre services rendus, rapidité et simplicité d'utilisation:

- le protocole **TCP** pour *Transmission Control Protocol*
- et le protocole **UDP** pour *User Datagram Protocol*.

Définition 12 [RESO 6.4.1] Le **protocole UDP** repère quand 1 bit du message est erroné ou quand le message est tronqué, mais ne corrige pas l'erreur. Il est rapide et simple à utiliser.

16 bits		16 bits	
port source			port destination
taille			checksum
données			

**Exemple 13** Historiquement, les services de **vidéo en streaming** privilégiaient la rapidité de transmission par dessus la fiabilité de la transmission; Ces services utilisaient donc le protocole UDP.

**Définition 14** [RESO 6.5.4] Le **protocole TCP** offre des garanties en plus par rapport au protocole UDP: il permet de vérifier que tous les paquets ont été reçus, dans le bon ordre. De plus, il adapte de le débit dynamiquement, pour éviter la congestion à la fois à la destination et dans le réseau.

DEV

16 bits		16 bits	
port source		port destination	
numéro de séquence			
numéro d'acquittement			
taille		fenêtre	
checksum		pointeur urgent	
options			
données			

**Exemple 15** Pour l'affichage d'une page web il est important que les différents parties du document HTML soient dans le bon ordre: on a besoin de garanties de fiabilité de la transmission. Le protocole HTTP utilise donc le plus souvent TCP.

**III. Routage des paquets: la couche internet**

**Définition 16** Un **routeur** [RESO 5.1.1] est un équipement réseau avec plusieurs interfaces réseau (connecté donc à plusieurs autres routeurs ou machines), qui reçoit et transmet des paquets.

**Problème 17** **Routage.** Pour chaque paquet, un routeur doit décider vers laquelle de ces interfaces réseau l'envoyer.

**Définition 18** La **couche internet** est celle qui s'assure pour un paquet donné, qu'il soit acheminé de sa source vers sa destination.

**Définition 19** Le **protocole IP**, pour *Internet Protocol*, est utilisé par la couche internet. Il utilise des adresses IP qui identifient l'émetteur et le destinataire d'un paquet; ainsi qu'une *durée de vie* qui contribue à limiter la congestion du réseau.

**Remarque 20** Il existe deux types d'adresses IP couramment utilisées:

- Les adresses IPv4 en 4 octets (32 bits), sous la forme aaa.bbb.ccc.ddd où aaa ... ddd sont les octets, notés en décimal. Leur nombre s'est avéré insuffisant (dernières adresses IPv4 distribuées en 2019).
- Les adresses IPv6 en 16 octets (128 bits) notés sous la forme aaaa:bbbb:cccc:dddd:eeee:ffff:gggg:hhhh, où chacun des aaaa ... hhhh correspond à 2 octets, notés en hexadécimal.

**Définition 21** [RESO 5.7.1] **En-tête IPv4**

16 bits		16 bits	
Version, LET, ...		Longueur totale	
identification		position fragment	
durée de vie	protocole	checksum de l'entête	
adresse source			
adresse de destination			
options			

**Remarque 22** [RESO 5.7.2] Les **adresses IP** sont **hiérarchiques**: un réseau correspond à une plage d'adresses avec un préfixe commun. Un administrateur réseau peut décomposer une telle plage en plusieurs plages plus petites. Cet aspect hiérarchique participe à l'adaptabilité du réseau, car il permet de déléguer et décentraliser la gestion des adresses IP.

**Remarque 23** Les **adresses IP** suivent un **principe topologique**: deux machines proches dans le réseau doivent avoir des adresses proches (au sens lexicographique).

**A. Le plan des données**

**Remarque 24** Les aspects hiérarchique et topologique des adresses IP, permettent aux routeurs de procéder à un **routage par préfixes** où les routeurs disposent de **tables de routage** leur indiquant vers où diriger les paquets en fonction de différents préfixes possibles de l'adresse de destination.

**Définition 25** [RESO 5.6.1] Le **plan des données** est la partie du routeur qui exécute le protocole IP, en suivant une logique (telle que des tables de routage) *déjà établie au préalable*.

**Exemple 26** Une table de routage peut ressembler à

Préfixe	Interface
185.26.126.0/23	1
151.80.16.0/20	2
Autres	3

Où 185.26.126.0/23 correspond aux adresses dont les 23 premiers bits coïncident avec ceux de 185.26.126.0.

**Exemple 27** Avec la table précédente, un paquet destiné à 185.26.126.225 serait envoyé sur l'interface 1.

**B. Le plan de contrôle**

**Définition 28 [RESO 5.6.1] Plan de contrôle** Le plan de contrôle est la partie de l'architecture d'un routeur qui définit la topologie d'un réseau, sa logique et décide comment retransmettre les données, par exemple en définissant et en emplissant les table de routage. Il peut utiliser des protocoles tels que RIP ou OSPF.

**1. Algoritihme de routage**

**Définition 29 [RESO 5.7.6] OSPF (Open Shortest Path First)** Il s'agit d'un protocole inter-domaine répondant à diverses exi-  
gences comme par exemple:

- Supporter une variété de distance métrique
- Être dynamique
- Équilibrer les charges
- etc...

**2. Commandes système [NSIT 23.3]**

**Remarque 30** Sous les systèmes Unix, diverses commandes permettent d'afficher et de modifier la configuration réseau de la machine. Tous les réseaux n'étant pas forcément gérés par RIP ou OSPF, il est en effet possible de procéder à des configurations manuelles. On parle alors de **routage statique**.

**Commande 31 [MAN]** La commande `ip`, qui fait partie de Linux, permet d'afficher des informations sur différents périphériques. Par exemple:

- `ip addr` affiche les adresses assignées aux interfaces réseau
- `ip route` affiche la table de routage

**Commande 32 [NSIT 23.3]** La commande `ping` permet de mesurer le temps d'aller-retour vers une machine cible. Elle procède par l'envoi d'un paquet test.

**Commande 33 [NSIT 23.3]** La commande `traceroute` permet de déterminer la route empruntée par un paquet IP pour atteindre la machine cible.

**3. Performance réseau [RESO 6.7]**

**Définition 34** Le **débit** est la quantité de données qui peuvent être envoyées ou reçues dans un intervalle de temps donné. Il est principalement influencé par la capacité des fils (et autres liaisons) dans le réseau. Il se mesure en mégabits par seconde.

**Définition 35** La **latence** est le temps d'attente nécessaire pour la réception de données. Elle est influencée par la distance que doit parcourir le signal. Elle se mesure en millisecondes.

**Définition 36** La **qualité d'expérience utilisateur** est une mesure de satisfaction de l'utilisateur vis-à-vis de la connexion internet.

**Remarque 37** Les besoin en terme de débit et de latence (et de leur dérivées) peuvent beaucoup varier d'un usage à l'autre. D'où l'utilité de mesurer une qualité d'expérience en plus de mesurer le débit et la latence.

**Exemple 38** Lors de **streaming vidéo** les utilisateurs souhaitent un débit élevé, mais tolèrent une certaine latence, pourvu qu'elle soit constante.

**Exemple 39** Lors d'**appels vocaux** (par exemple en « voix sur IP »), les utilisateurs souhaitent une latence très faible, mais ne requièrent pas nécessairement d'un débit très élevé.

Échanges de données et routage. Exemples.	
<b>I. Contexte et modèle par couches</b>	
1 Objectifs généraux [RESO 1.5.1] Pour un réseau d'envoi de données	9 Def TCP/IP
2 Def [RESO 2.5.4] La commutation de circuits	<b>II. Échanger des données: la couche transport</b>
3 Ex Réseaux téléphoniques	
4 Def [RESO 2.5.4] L'envoi de paquets	
5 Ex Internet	10 Def Les numéro de port
6 Compromis Commutation circuits vs envoi paquets	11 Def Les protocoles de la couche transport
7 Rem Ici, paquets internet	12 Def [RESO 6.4.1]Le protocole UDP
8 Def Modèle par couches	
13 Ex Streaming en UDP	
14 Def [RESO 6.5.4]Le protocole TCP	
15 Ex Pour l'affichage d'une page web	21 Def [RESO 5.7.1]En-tête IPv4
<b>III. Routage des paquets: la couche internet</b>	22 Rem [RESO 5.7.2] Les adresses IP sont hiérarchiques:
16 Def Un routeur	23 Rem Les adresses IP suivent un principe topologique: A. Le plan des données
17 Prob Routage.	24 Rem Routage par préfixe
18 Def La couche internet	25 Def [RESO 5.6.1] Le plan des données
19 Def Le protocole IP,	
20 Rem IPv4 vs IPv6	31 Commande [MAN] La commande ip,
26 Ex Une table de routage	32 Commande [NSIT 23.3] La commande ping
27 Ex Routage d'un paquet B. Le plan de contrôle	33 Commande [NSIT 23.3] La commande traceroute
28 Def [RESO 5.6.1]Plan de contrôle	34 Def Le débit
29 Def [RESO 5.7.6]OSPF (Open Shortest Path First)	35 Def La latence
30 Rem	36 Def La qualité d'expérience utilisateur
	37 Rem Exigences utilisateur variables
	38 Ex Lors de streaming vidéo
	39 Ex Lors d'appels vocaux

## Remarques générales

- Nous avons choisi d'utiliser essentiellement le [RESO] car il est en français et épargne donc l'étape de traduction. Une autre très bonne référence pour cette leçon est bien sûr le [KURO] .
- Le [RESO] parle de « commutation de paquets ». Nous préférons parler de « envoi de paquets ».
- Le [RESO] parle de « couche réseau » plutôt que de « couche internet ». C'est un choix qu'on approuve, mais qui ne correspond pas *strictu sensu* au modèle TCP/IP, qui est au programme complémentaire (et au programme de seconde ?).
- À la question « Citer un exemple de protocole pour la fibre optique » on peut répondre « SONET » [RESO 1.6.2] . De même, WiFi -> 802.11 .

## Bibliographie

[RESO] A. Tanenbaum & N. Feamster & D. J. Wetherall, *Réseaux, 6° édition, en français.*

[NSIT] T. Balabonski & S. Conchon & J. Filliâtre & K. Nguyen, *Numériques et Sciences Informatiques Terminale.*

[MAN] Linux, *Man page.*

[KURO] J. F. Kurose & K. W. Ross, *Computer Networks, a top-down approach, 8th edition.*