

RES101

Invariants fonctionnels

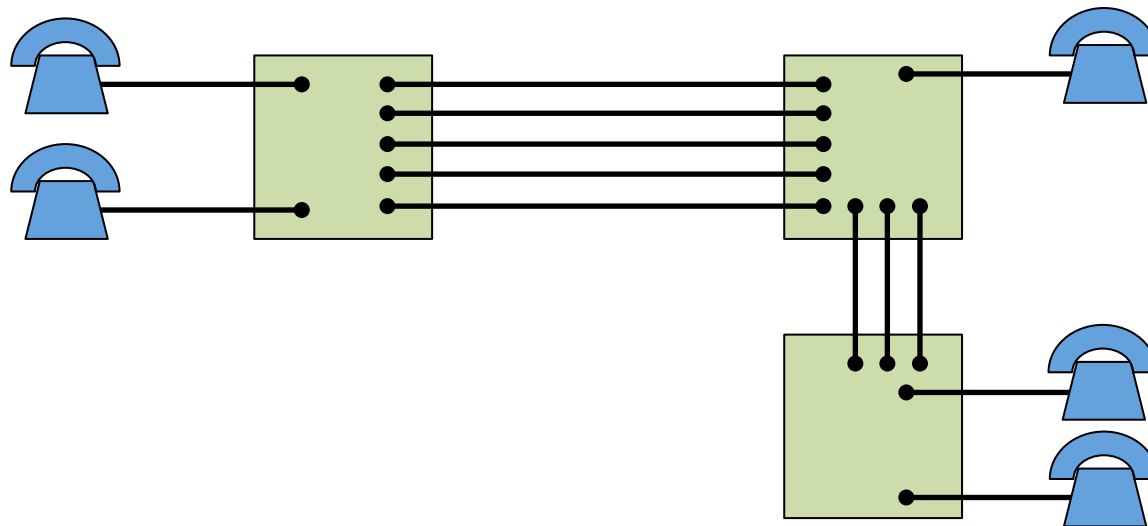
## **4. ACHEMINEMENT BOUT-EN-BOUT**

# Commutation

- La plupart des équipements terminaux ne sont pas reliés par des liaisons point-à-point
- Ils sont alors reliés par un réseau de commutation comprenant
  - Des équipements terminaux
  - Des commutateurs
  - Des liaisons
- Il existe différents types de commutation
  - De circuit
  - De messages
  - De paquets
  - De cellules

# Commutation de circuit

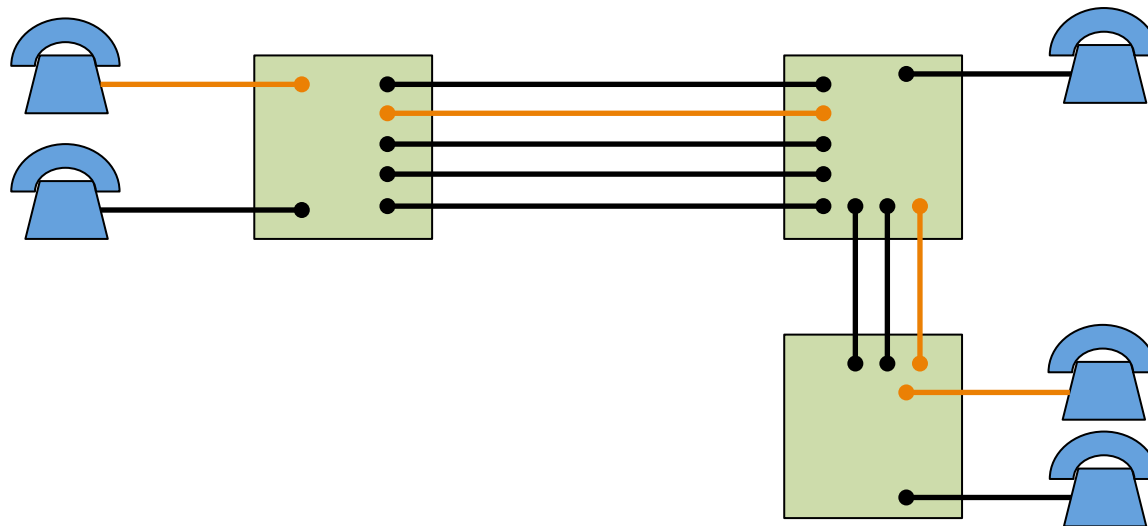
- **Principe :**
  - Dans un réseau à commutation de circuit, les commutateurs sont reliés par plusieurs liaisons appelées circuits
  - Pour établir une communication entre 2 équipements terminaux, il faut réserver un chemin à travers le réseau, i.e. un circuit entre chaque paire de commutateur traversée



- **Exemple :** réseau téléphonique commuté

# Commutation de circuit

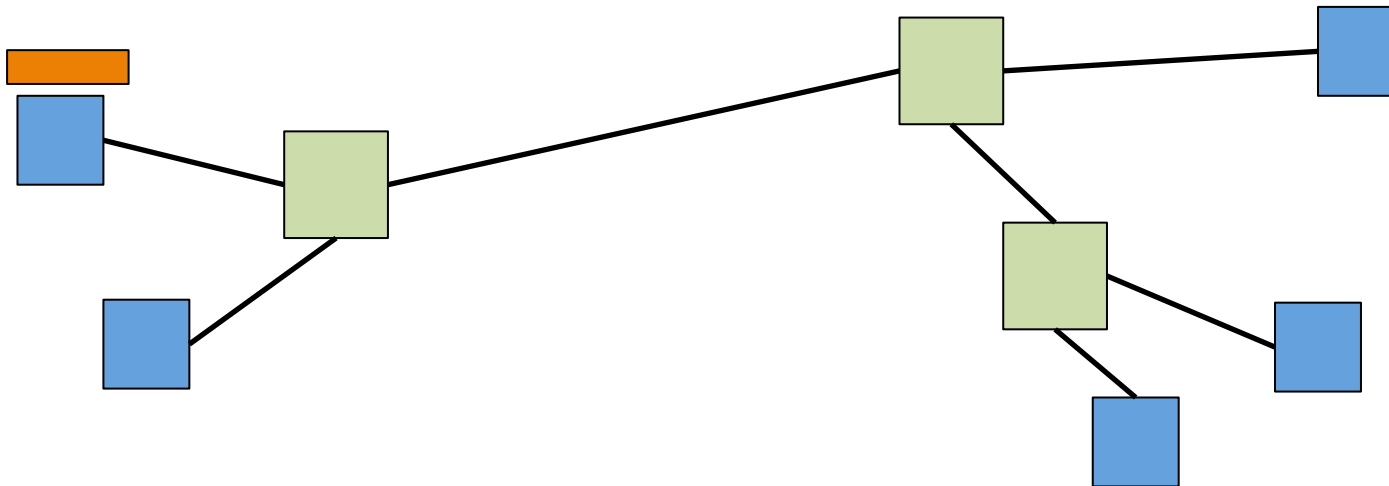
- **Principe :**
  - Dans un réseau à commutation de circuit, les commutateurs sont reliés par plusieurs liaisons appelées circuits
  - Pour établir une communication entre 2 équipements terminaux, il faut réserver un chemin à travers le réseau, i.e. un circuit entre chaque paire de commutateur traversée



- **Exemple :** réseau téléphonique commuté

# Commutation de messages

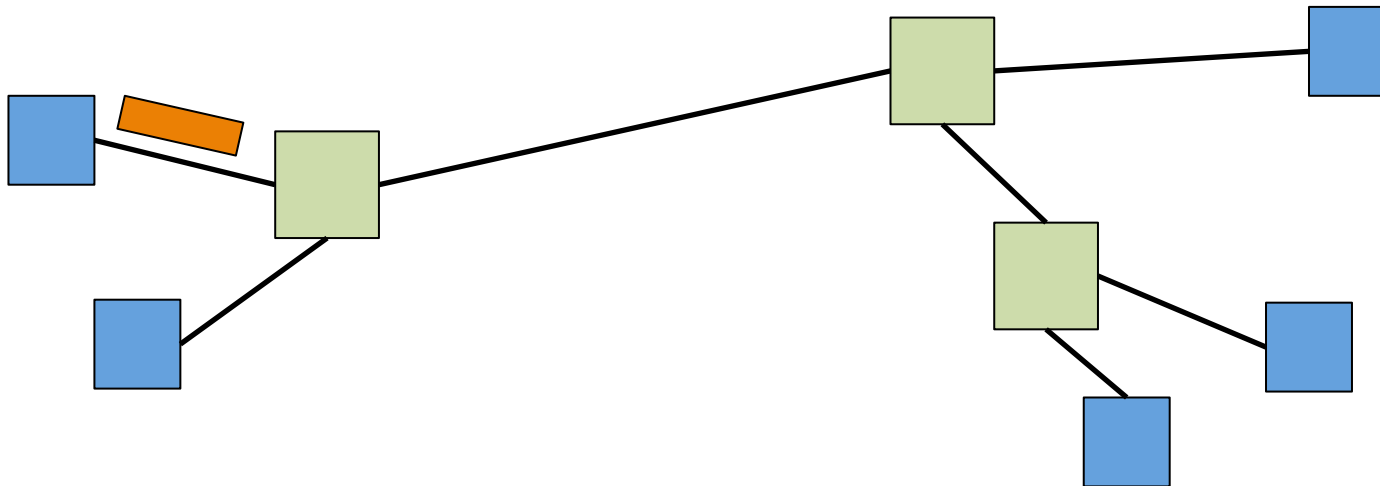
- **Principe :**
  - Les commutateurs sont reliés 2 à 2 par une liaison de données
  - Un message forme un tout logique pour les équipements terminaux (un fichier, un mail...)
  - Le message est émis avec une adresse destination
  - Chaque commutateur traversé analyse l'adresse et envoie le message vers le prochain saut adéquat
  - Le message conserve son intégrité tout le long du chemin



- **Exemple :** Telex

# Commutation de messages

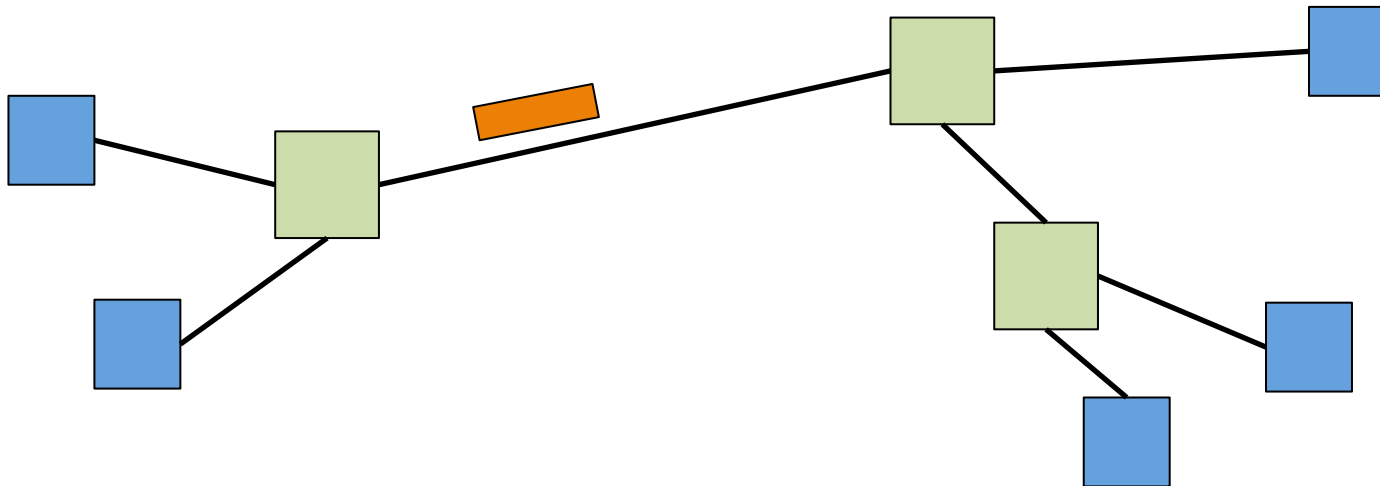
- **Principe :**
  - Les commutateurs sont reliés 2 à 2 par une liaison de données
  - Un message forme un tout logique pour les équipements terminaux (un fichier, un mail...)
  - Le message est émis avec une adresse destination
  - Chaque commutateur traversé analyse l'adresse et envoie le message vers le prochain saut adéquat
  - Le message conserve son intégrité tout le long du chemin



- **Exemple :** Telex

# Commutation de messages

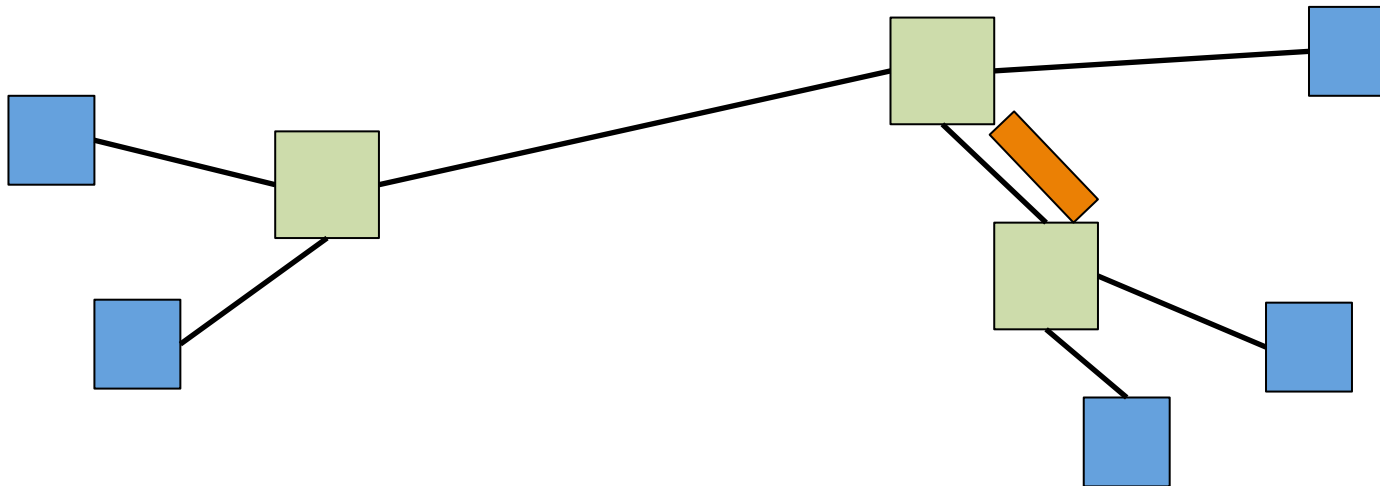
- **Principe :**
  - Les commutateurs sont reliés 2 à 2 par une liaison de données
  - Un message forme un tout logique pour les équipements terminaux (un fichier, un mail...)
  - Le message est émis avec une adresse destination
  - Chaque commutateur traversé analyse l'adresse et envoie le message vers le prochain saut adéquat
  - Le message conserve son intégrité tout le long du chemin



- **Exemple :** Telex

# Commutation de messages

- **Principe :**
  - Les commutateurs sont reliés 2 à 2 par une liaison de données
  - Un message forme un tout logique pour les équipements terminaux (un fichier, un mail...)
  - Le message est émis avec une adresse destination
  - Chaque commutateur traversé analyse l'adresse et envoie le message vers le prochain saut adéquat
  - Le message conserve son intégrité tout le long du chemin

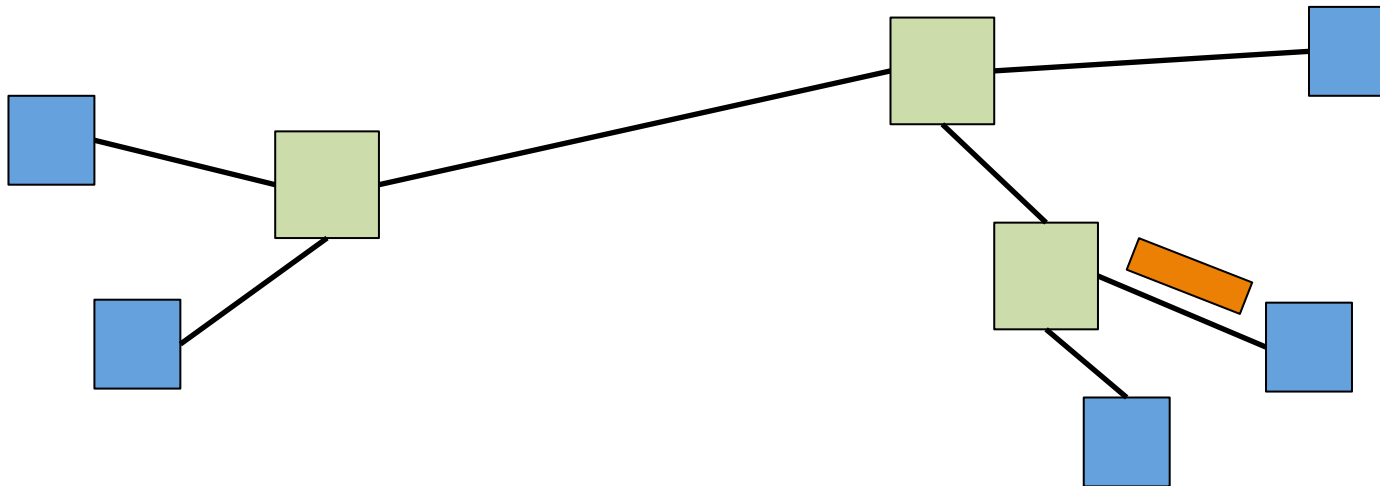


- **Exemple :** Telex



# Commutation de messages

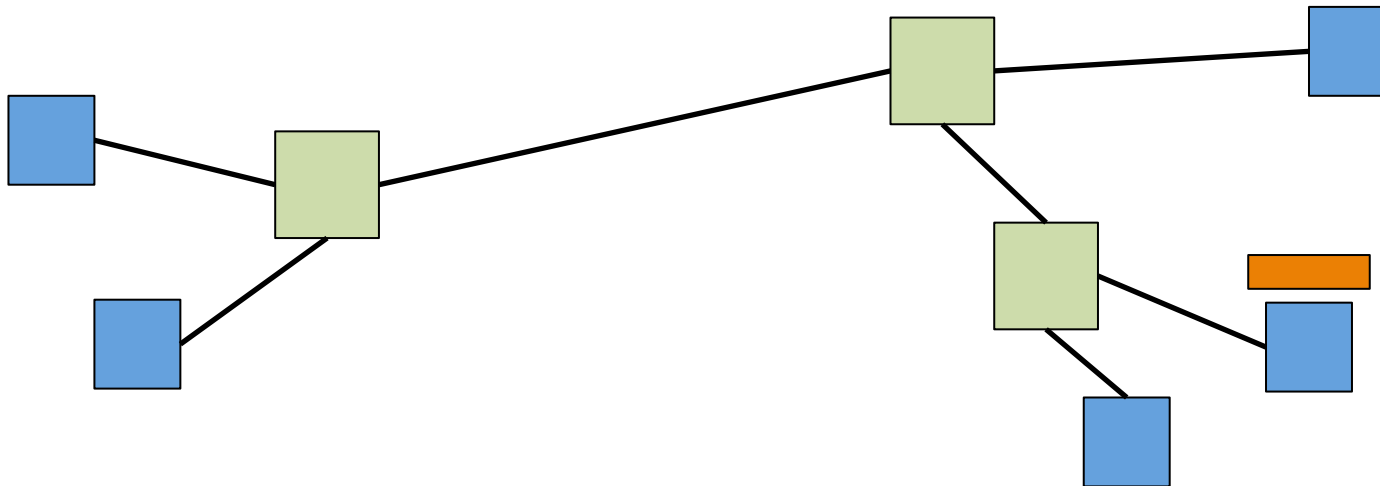
- **Principe :**
  - Les commutateurs sont reliés 2 à 2 par une liaison de données
  - Un message forme un tout logique pour les équipements terminaux (un fichier, un mail...)
  - Le message est émis avec une adresse destination
  - Chaque commutateur traversé analyse l'adresse et envoie le message vers le prochain saut adéquat
  - Le message conserve son intégrité tout le long du chemin



- **Exemple :** Telex

# Commutation de messages

- **Principe :**
  - Les commutateurs sont reliés 2 à 2 par une liaison de données
  - Un message forme un tout logique pour les équipements terminaux (un fichier, un mail...)
  - Le message est émis avec une adresse destination
  - Chaque commutateur traversé analyse l'adresse et envoie le message vers le prochain saut adéquat
  - Le message conserve son intégrité tout le long du chemin

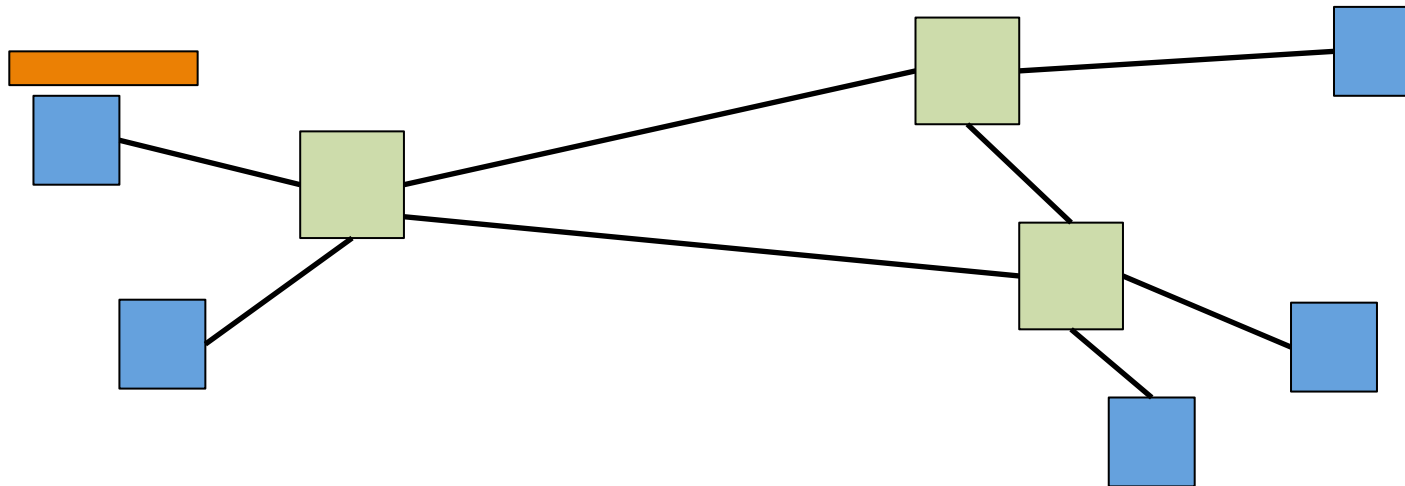


- **Exemple :** Telex

# Commutation de paquets

- **Principe :**

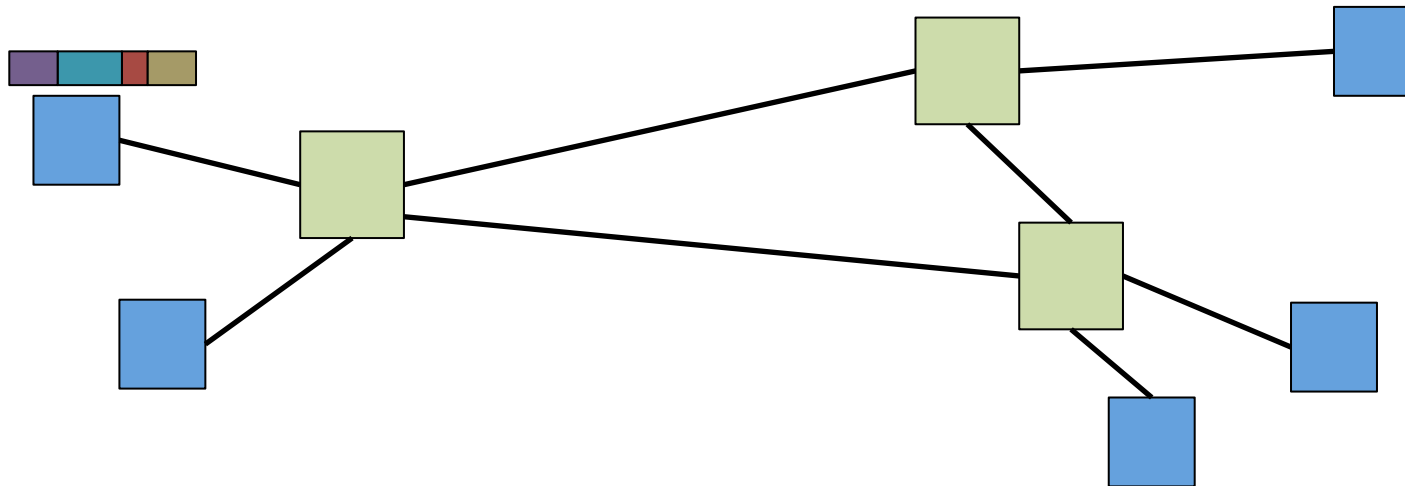
- Les commutateurs sont reliés 2 à 2 par une liaison de données
- Les messages sont découpés en petits paquets de tailles différentes au moment de l'émission
- Chaque paquet contient l'adresse destination
- Chaque commutateur traversé par un paquet analyse l'adresse et choisit le prochain saut auquel envoyer le paquet
- Les paquets ne passent pas tous forcément par le même chemin et n'arrivent pas forcément dans l'ordre dans lequel ils ont été envoyés
- Le récepteur réorganise tous les paquets pour reconstituer le message



- **Exemple :** Internet UDP/IP

# Commutation de paquets

- **Principe :**
  - Les commutateurs sont reliés 2 à 2 par une liaison de données
  - Les messages sont découpés en petits paquets de tailles différentes au moment de l'émission
  - Chaque paquet contient l'adresse destination
  - Chaque commutateur traversé par un paquet analyse l'adresse et choisit le prochain saut auquel envoyer le paquet
  - Les paquets ne passent pas tous forcément par le même chemin et n'arrivent pas forcément dans l'ordre dans lequel ils ont été envoyés
  - Le récepteur réorganise tous les paquets pour reconstituer le message

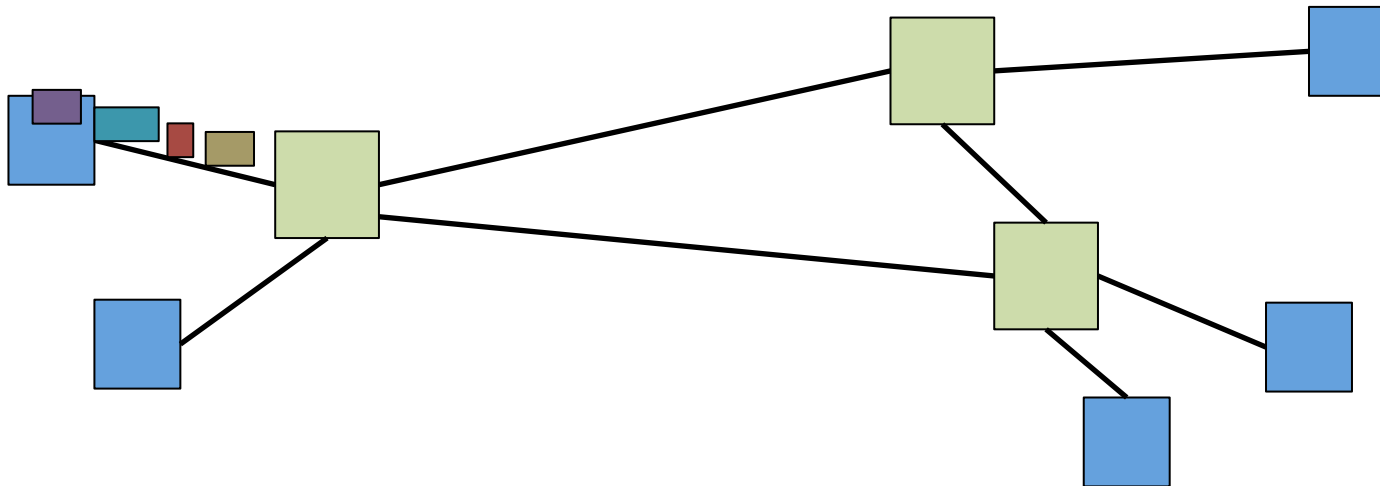


- **Exemple :** Internet UDP/IP

# Commutation de paquets

- **Principe :**

- Les commutateurs sont reliés 2 à 2 par une liaison de données
- Les messages sont découpés en petits paquets de tailles différentes au moment de l'émission
- Chaque paquet contient l'adresse destination
- Chaque commutateur traversé par un paquet analyse l'adresse et choisit le prochain saut auquel envoyer le paquet
- Les paquets ne passent pas tous forcément par le même chemin et n'arrivent pas forcément dans l'ordre dans lequel ils ont été envoyés
- Le récepteur réorganise tous les paquets pour reconstituer le message

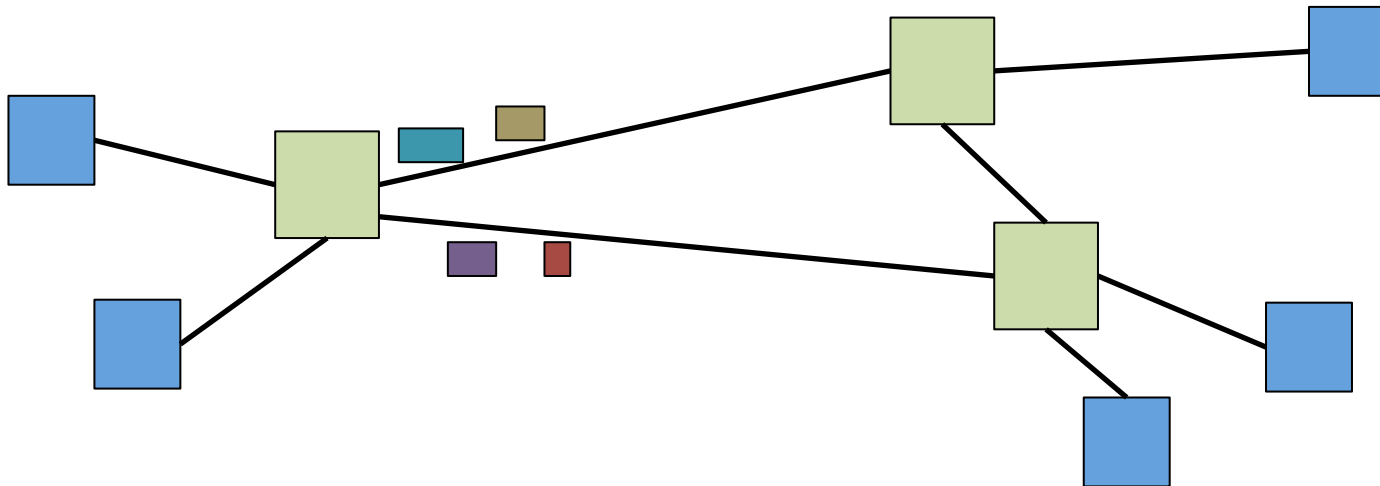


- **Exemple :** Internet UDP/IP

# Commutation de paquets

- **Principe :**

- Les commutateurs sont reliés 2 à 2 par une liaison de données
- Les messages sont découpés en petits paquets de tailles différentes au moment de l'émission
- Chaque paquet contient l'adresse destination
- Chaque commutateur traversé par un paquet analyse l'adresse et choisit le prochain saut auquel envoyer le paquet
- Les paquets ne passent pas tous forcément par le même chemin et n'arrivent pas forcément dans l'ordre dans lequel ils ont été envoyés
- Le récepteur réorganise tous les paquets pour reconstituer le message

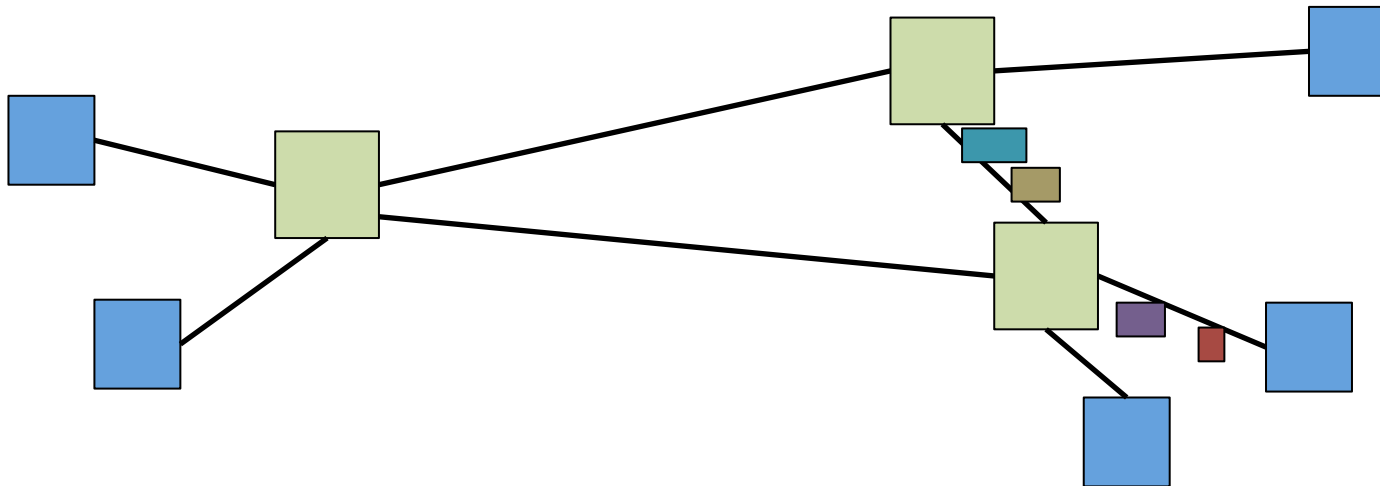


- **Exemple :** Internet UDP/IP

# Commutation de paquets

- **Principe :**

- Les commutateurs sont reliés 2 à 2 par une liaison de données
- Les messages sont découpés en petits paquets de tailles différentes au moment de l'émission
- Chaque paquet contient l'adresse destination
- Chaque commutateur traversé par un paquet analyse l'adresse et choisit le prochain saut auquel envoyer le paquet
- Les paquets ne passent pas tous forcément par le même chemin et n'arrivent pas forcément dans l'ordre dans lequel ils ont été envoyés
- Le récepteur réorganise tous les paquets pour reconstituer le message

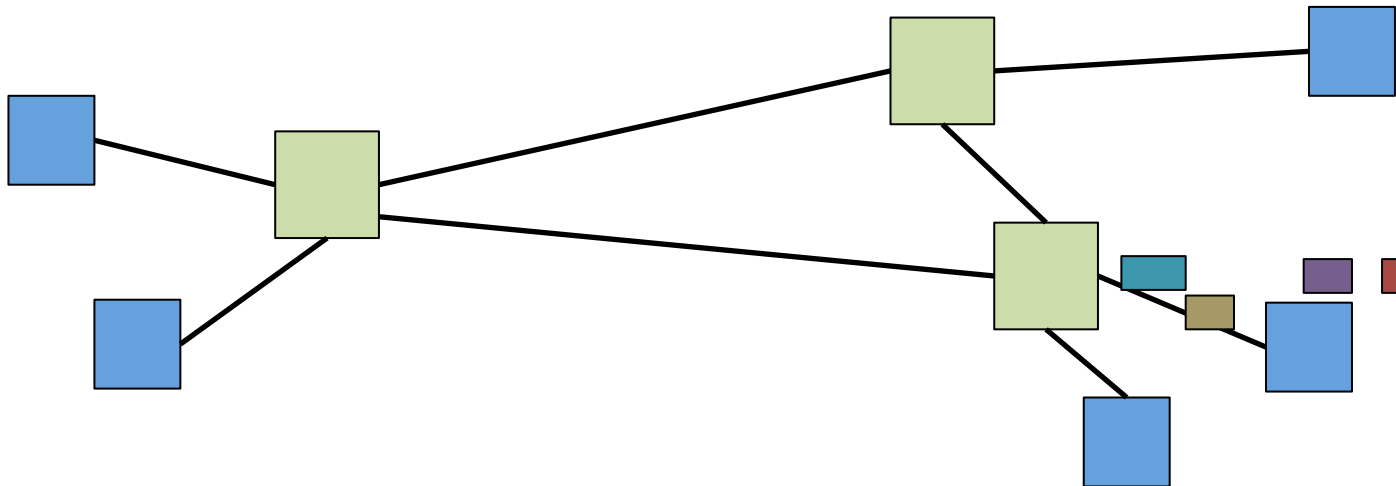


- **Exemple :** Internet UDP/IP

# Commutation de paquets

- **Principe :**

- Les commutateurs sont reliés 2 à 2 par une liaison de données
- Les messages sont découpés en petits paquets de tailles différentes au moment de l'émission
- Chaque paquet contient l'adresse destination
- Chaque commutateur traversé par un paquet analyse l'adresse et choisit le prochain saut auquel envoyer le paquet
- Les paquets ne passent pas tous forcément par le même chemin et n'arrivent pas forcément dans l'ordre dans lequel ils ont été envoyés
- Le récepteur réorganise tous les paquets pour reconstituer le message



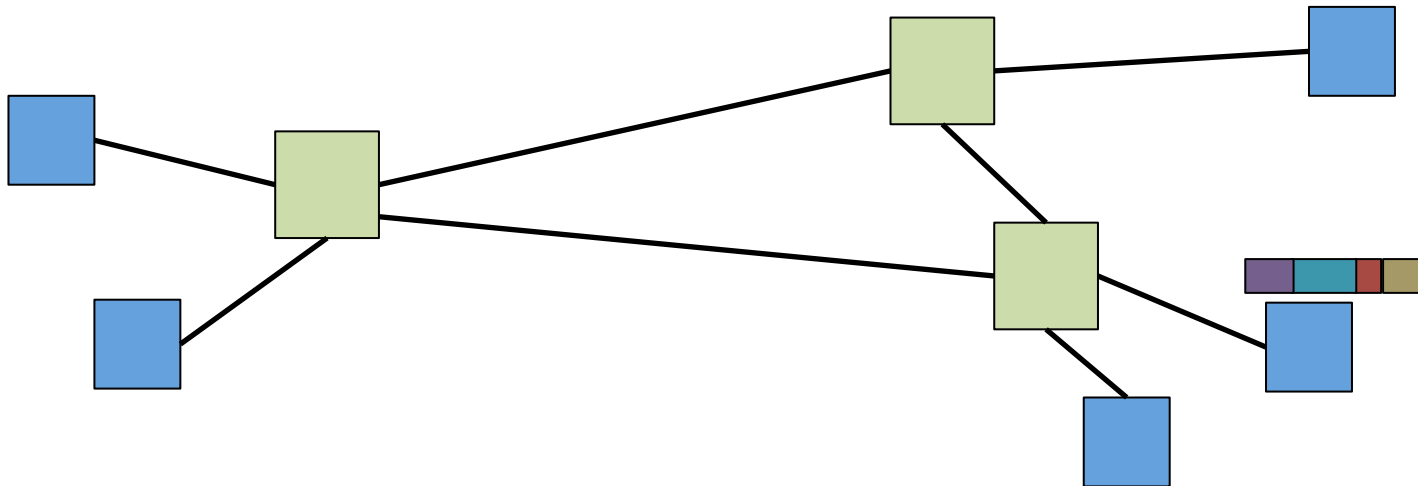
- **Exemple :** Internet UDP/IP



# Commutation de paquets

- **Principe :**

- Les commutateurs sont reliés 2 à 2 par une liaison de données
- Les messages sont découpés en petits paquets de tailles différentes au moment de l'émission
- Chaque paquet contient l'adresse destination
- Chaque commutateur traversé par un paquet analyse l'adresse et choisit le prochain saut auquel envoyer le paquet
- Les paquets ne passent pas tous forcément par le même chemin et n'arrivent pas forcément dans l'ordre dans lequel ils ont été envoyés
- Le récepteur réorganise tous les paquets pour reconstituer le message

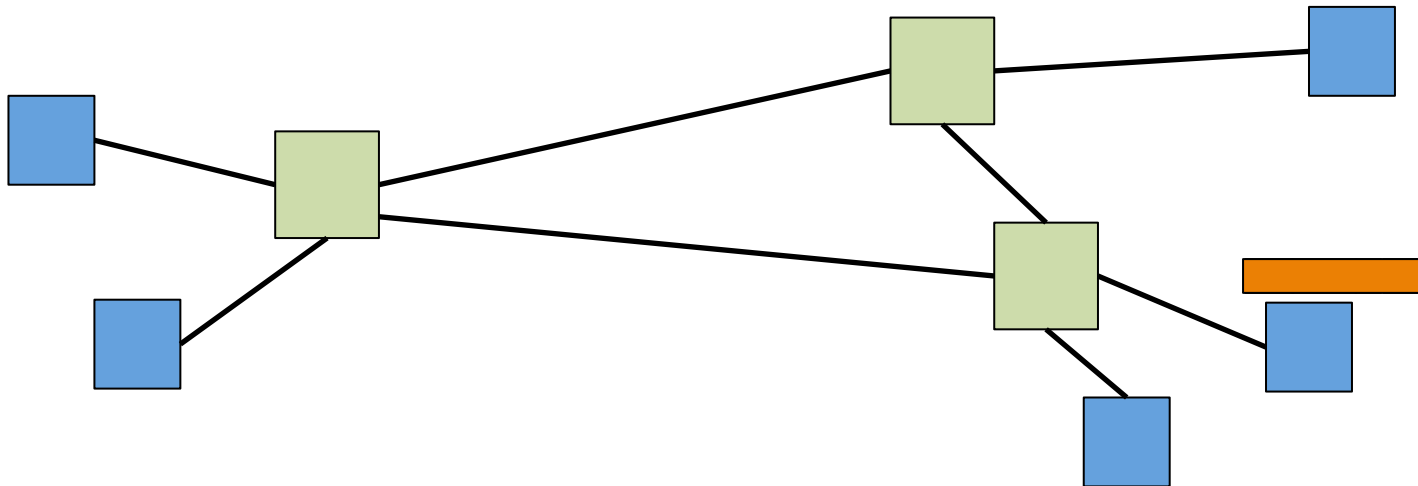


- **Exemple :** Internet UDP/IP

# Commutation de paquets

- **Principe :**

- Les commutateurs sont reliés 2 à 2 par une liaison de données
- Les messages sont découpés en petits paquets de tailles différentes au moment de l'émission
- Chaque paquet contient l'adresse destination
- Chaque commutateur traversé par un paquet analyse l'adresse et choisit le prochain saut auquel envoyer le paquet
- Les paquets ne passent pas tous forcément par le même chemin et n'arrivent pas forcément dans l'ordre dans lequel ils ont été envoyés
- Le récepteur réorganise tous les paquets pour reconstituer le message

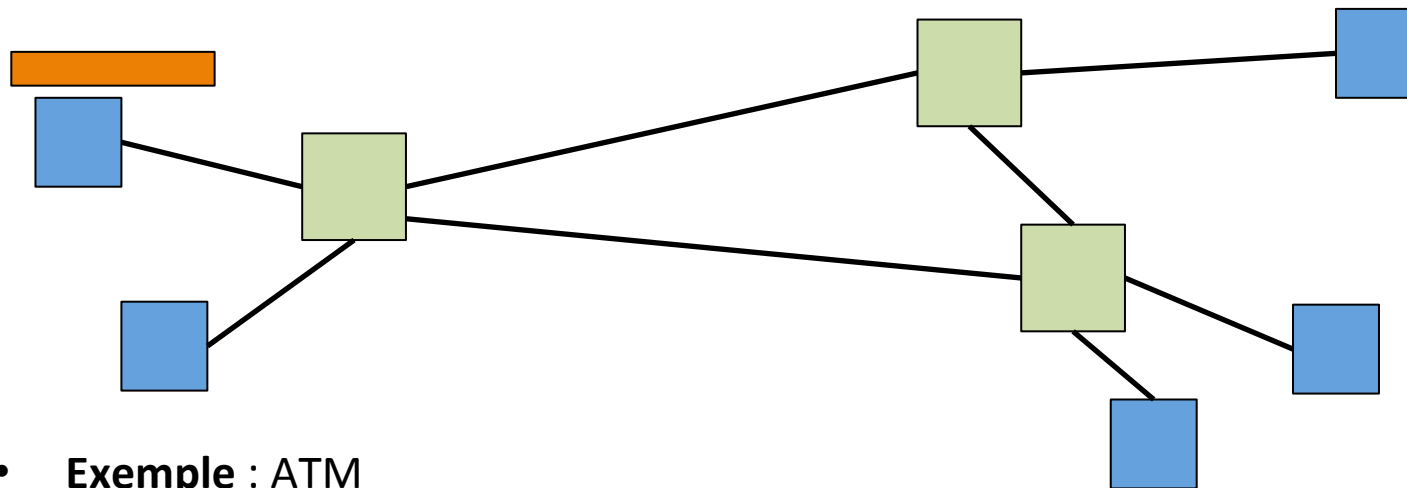


- **Exemple :** Internet UDP/IP

# Commutation de cellules

- **Principe :**

- Les commutateurs sont reliés 2 à 2 par une liaison de données
- Les messages sont découpés en petits paquets de taille identique appelés cellules au moment de l'émission
- Un chemin ou circuit virtuel est établi à travers le réseau entre l'émetteur et le récepteur
- Chaque cellule contient l'identifiant du chemin réservé
- Les cellules passent toutes par le même chemin et restent dans le même ordre
- Le récepteur rassemble les cellules pour reconstituer le message

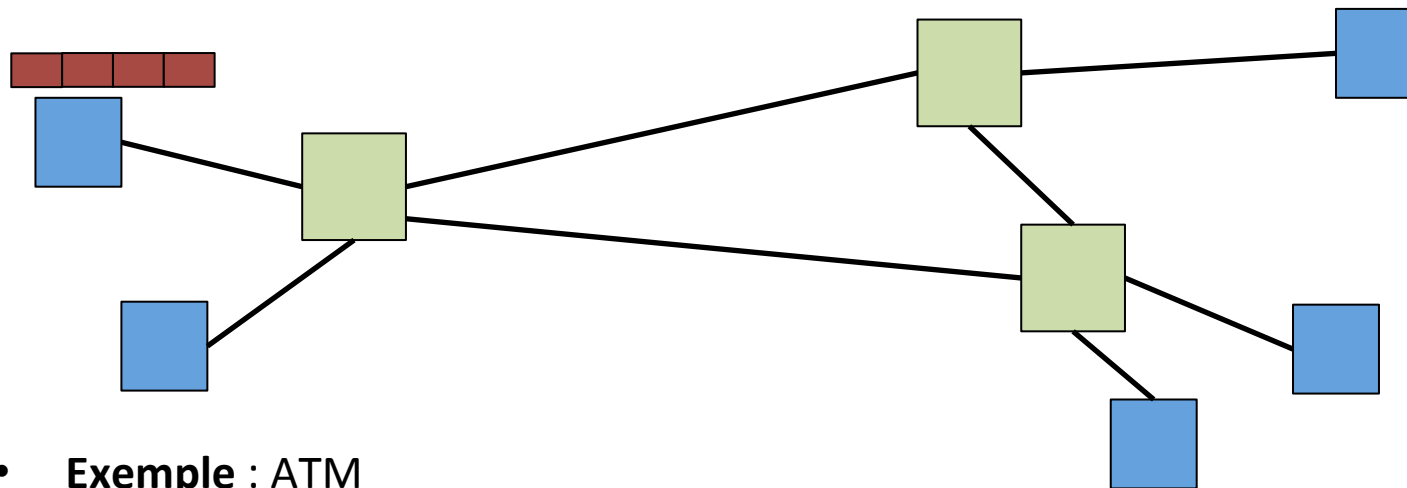


- **Exemple : ATM**

# Commutation de cellules

- **Principe :**

- Les commutateurs sont reliés 2 à 2 par une liaison de données
- Les messages sont découpés en petits paquets de taille identique appelés cellules au moment de l'émission
- Un chemin ou circuit virtuel est établi à travers le réseau entre l'émetteur et le récepteur
- Chaque cellule contient l'identifiant du chemin réservé
- Les cellules passent toutes par le même chemin et restent dans le même ordre
- Le récepteur rassemble les cellules pour reconstituer le message

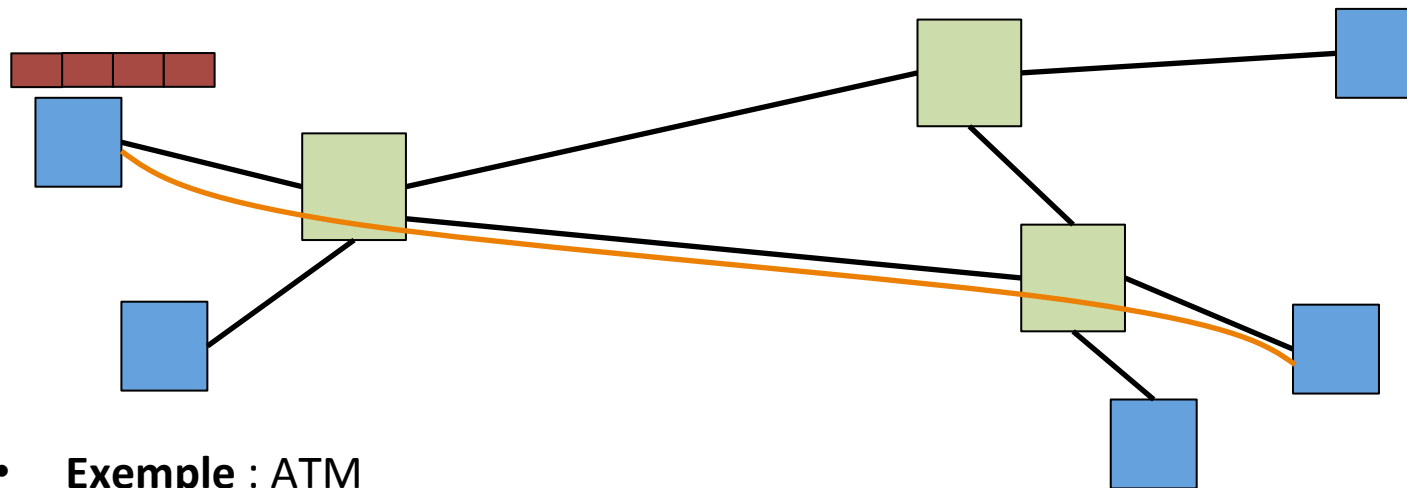


- **Exemple : ATM**

# Commutation de cellules

- **Principe :**

- Les commutateurs sont reliés 2 à 2 par une liaison de données
- Les messages sont découpés en petits paquets de taille identique appelés cellules au moment de l'émission
- Un chemin ou circuit virtuel est établi à travers le réseau entre l'émetteur et le récepteur
- Chaque cellule contient l'identifiant du chemin réservé
- Les cellules passent toutes par le même chemin et restent dans le même ordre
- Le récepteur rassemble les cellules pour reconstituer le message

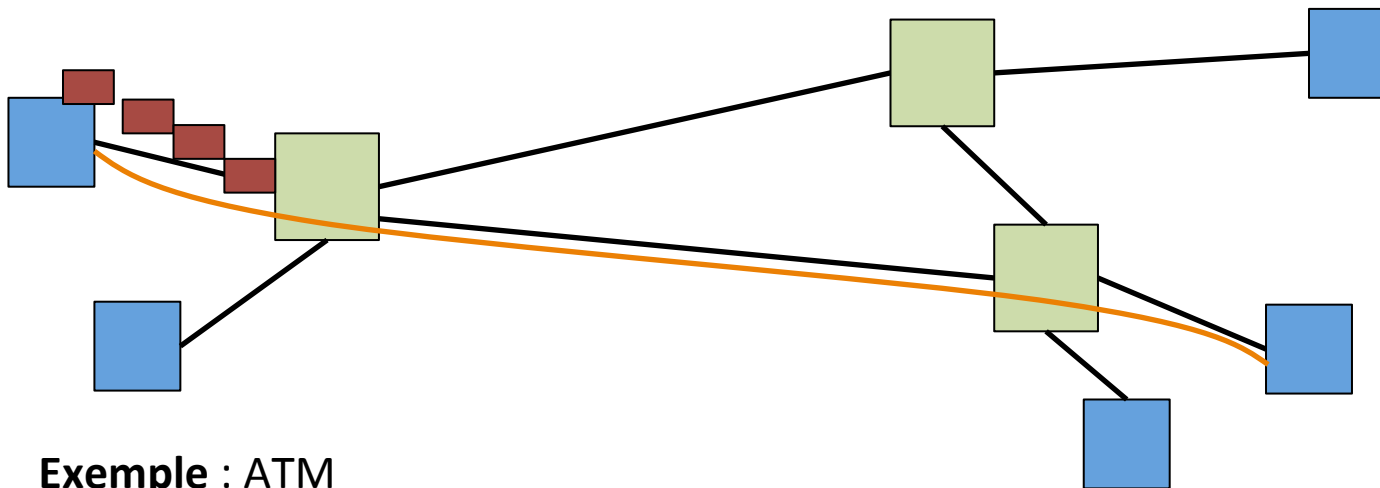


- **Exemple : ATM**

# Commutation de cellules

- **Principe :**

- Les commutateurs sont reliés 2 à 2 par une liaison de données
- Les messages sont découpés en petits paquets de taille identique appelés cellules au moment de l'émission
- Un chemin ou circuit virtuel est établi à travers le réseau entre l'émetteur et le récepteur
- Chaque cellule contient l'identifiant du chemin réservé
- Les cellules passent toutes par le même chemin et restent dans le même ordre
- Le récepteur rassemble les cellules pour reconstituer le message

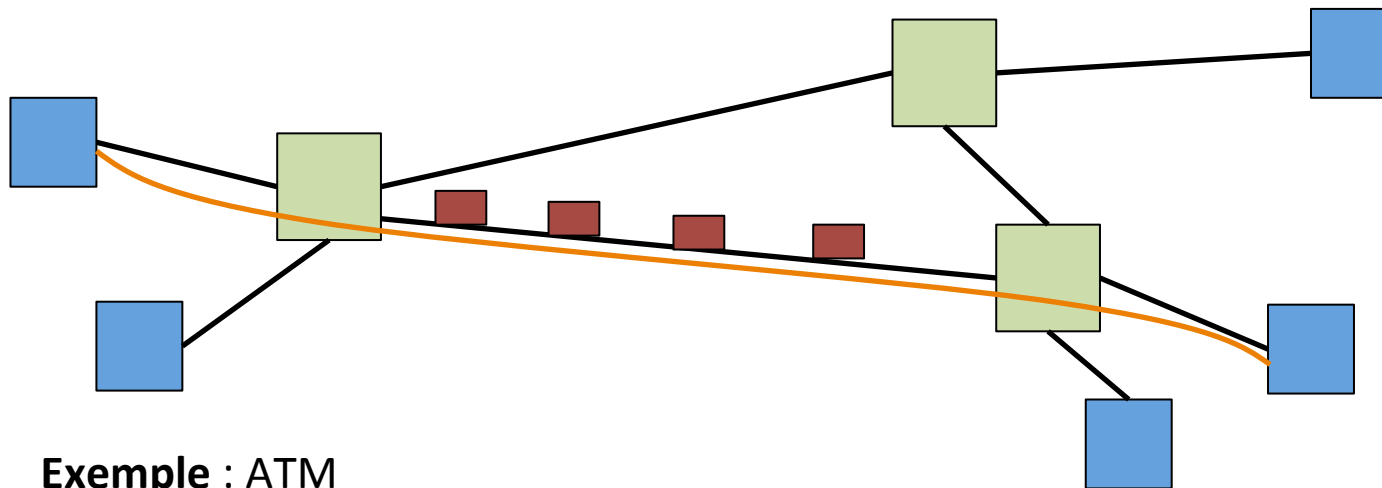


- **Exemple : ATM**

# Commutation de cellules

- **Principe :**

- Les commutateurs sont reliés 2 à 2 par une liaison de données
- Les messages sont découpés en petits paquets de taille identique appelés cellules au moment de l'émission
- Un chemin ou circuit virtuel est établi à travers le réseau entre l'émetteur et le récepteur
- Chaque cellule contient l'identifiant du chemin réservé
- Les cellules passent toutes par le même chemin et restent dans le même ordre
- Le récepteur rassemble les cellules pour reconstituer le message

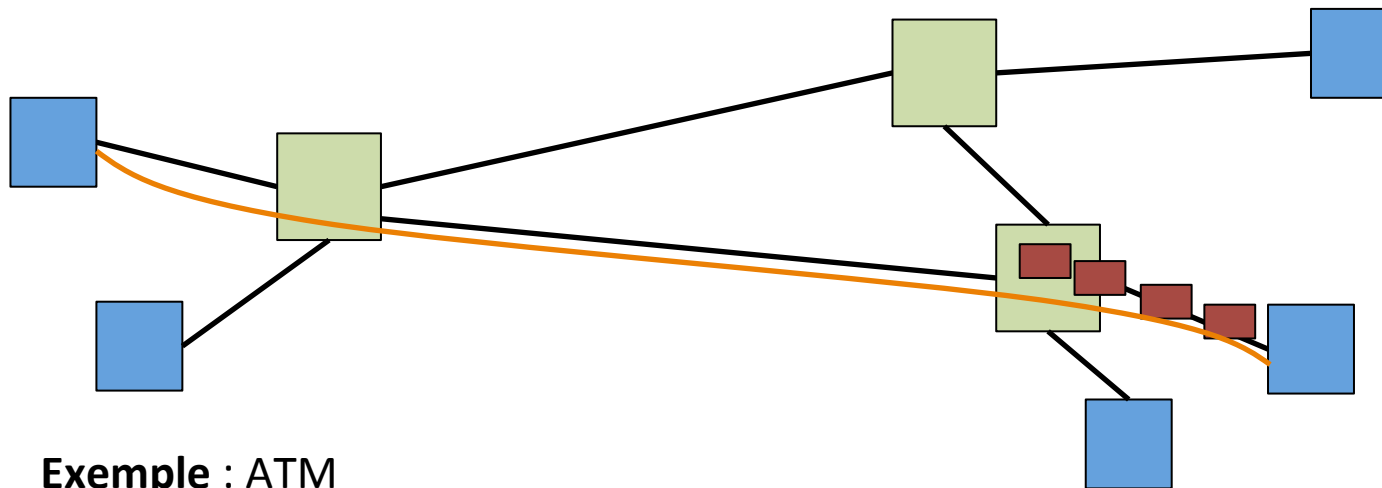


- **Exemple : ATM**

# Commutation de cellules

- **Principe :**

- Les commutateurs sont reliés 2 à 2 par une liaison de données
- Les messages sont découpés en petits paquets de taille identique appelés cellules au moment de l'émission
- Un chemin ou circuit virtuel est établi à travers le réseau entre l'émetteur et le récepteur
- Chaque cellule contient l'identifiant du chemin réservé
- Les cellules passent toutes par le même chemin et restent dans le même ordre
- Le récepteur rassemble les cellules pour reconstituer le message



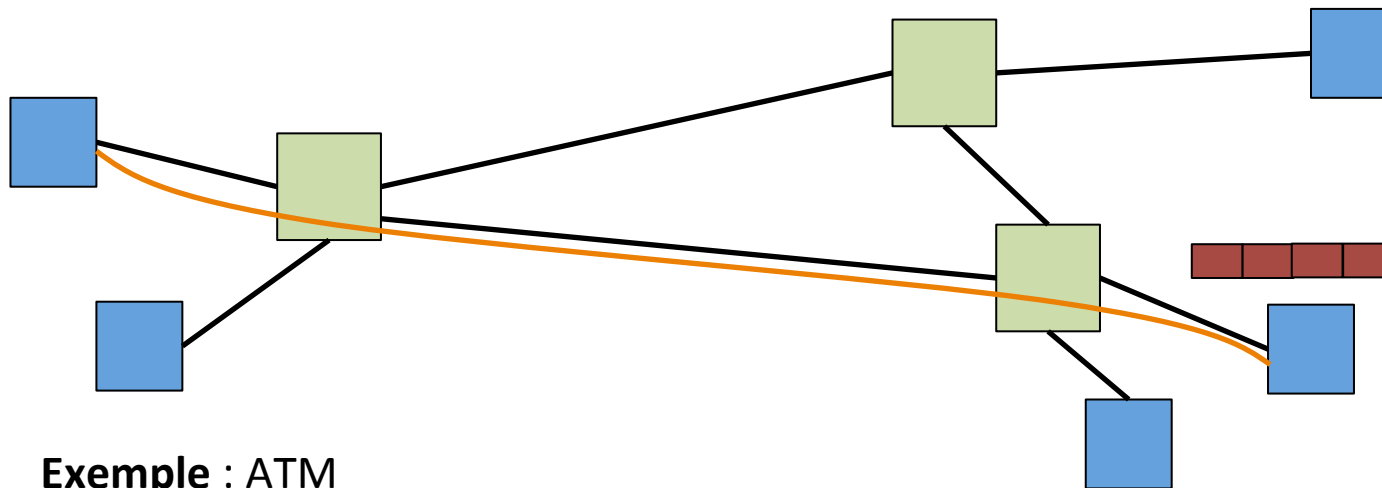
- **Exemple : ATM**



# Commutation de cellules

- **Principe :**

- Les commutateurs sont reliés 2 à 2 par une liaison de données
- Les messages sont découpés en petits paquets de taille identique appelés cellules au moment de l'émission
- Un chemin ou circuit virtuel est établi à travers le réseau entre l'émetteur et le récepteur
- Chaque cellule contient l'identifiant du chemin réservé
- Les cellules passent toutes par le même chemin et restent dans le même ordre
- Le récepteur rassemble les cellules pour reconstituer le message

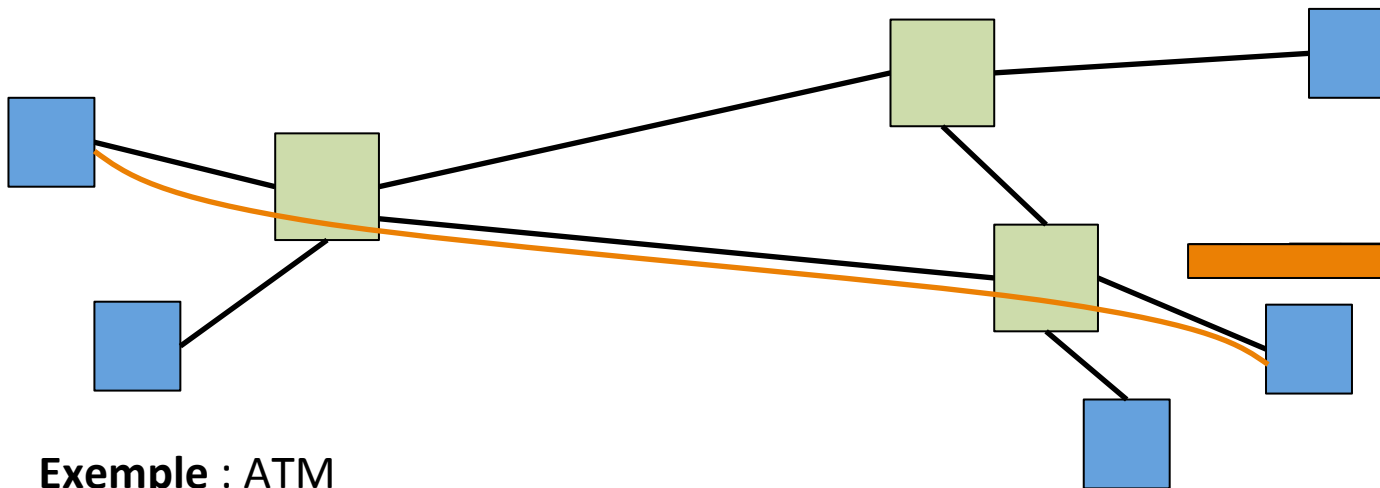


- **Exemple : ATM**

# Commutation de cellules

- **Principe :**

- Les commutateurs sont reliés 2 à 2 par une liaison de données
- Les messages sont découpés en petits paquets de taille identique appelés cellules au moment de l'émission
- Un chemin ou circuit virtuel est établi à travers le réseau entre l'émetteur et le récepteur
- Chaque cellule contient l'identifiant du chemin réservé
- Les cellules passent toutes par le même chemin et restent dans le même ordre
- Le récepteur rassemble les cellules pour reconstituer le message



- **Exemple : ATM**

# Mode connecté / non connecté

- **Mode connecté**

- Une connexion est un lien logique entre 2 équipements
- Une communication en mode connecté comporte 3 étapes :
  - **Etablissement** de la connexion : négociation des paramètres de connexion (synchronisation des numéros de trames, choix du chemin, qualité de service...), création de l'identifiant de connexion s'il existe.
  - **Transfert** des données : selon le type de connexion, les données peuvent être affranchies d'adresses source et destination si on utilise un identifiant de connexion, elles peuvent transiter par le même chemin. Le contrôle d'erreur, de flux et de congestion est possible.
  - **Libération** de la connexion : fermeture du lien logique établi, libération des ressources réservées si il y en avait.
- Avantages
  - Le routage (choix du chemin) n'est fait qu'une seule fois par communication
  - Les contrôles d'erreur, flux et congestion sont possibles de bout-en-bout

- **Mode non connecté**

- Les paquets d'une même communication sont traités de manière indépendante
- La communication peut commencer directement sans besoin d'établissement
- Avantages
  - Pas besoin d'attendre un établissement de connexion pour commencer à communiquer
  - Plus flexible

# Acheminement

- L'acheminement ou le routage est la détermination du chemin emprunté par les messages entre l'émetteur et le récepteur dans un réseau de commutation
- Cette opération a lieu
  - Soit lors de l'établissement de la connexion
  - Soit indépendamment pour chaque paquet
- Le routage peut être
  - Statique
    - Le choix du chemin entre 2 équipements est calculé une fois puis stocké
    - Adapté pour des équipements particuliers à l'intérieur d'un réseau opéré (administré par un opérateur)
  - Dynamique
    - Le choix du chemin est calculé en fonction de l'état actuel du réseau
    - Adapté pour un réseau où des liens sont susceptibles d'apparaître, être modifiés, ou disparaître
- Le choix du plus court chemin peut être fait sur plusieurs critères :
  - Moins cher
  - Plus rapide
  - Plus sécurisé...
  - Mais toujours : **sans boucle**

# Algorithmes de routage

- L'algorithme de routage peut être
  - Centralisé : un équipement dédié se charge seul du routage pour tout le réseau
  - Distribué : chaque nœud calcule les chemins dont il a besoin en fonction des informations dont il dispose
- Il existe de nombreux algorithmes de routage selon l'information dont dispose l'équipement qui l'exécute
  - La topologie complète du réseau
  - Seulement son voisinage
  - Seulement ses propres liens
  - ...
- Basés sur les algorithmes de plus courts chemins (vus en INF101)