

# Analyse et modélisation

Séance 1: Introduire au développement de systèmes d'information

420-A56-GG

Par

Ange Adrienne Nyamen Tato

## Rappels :

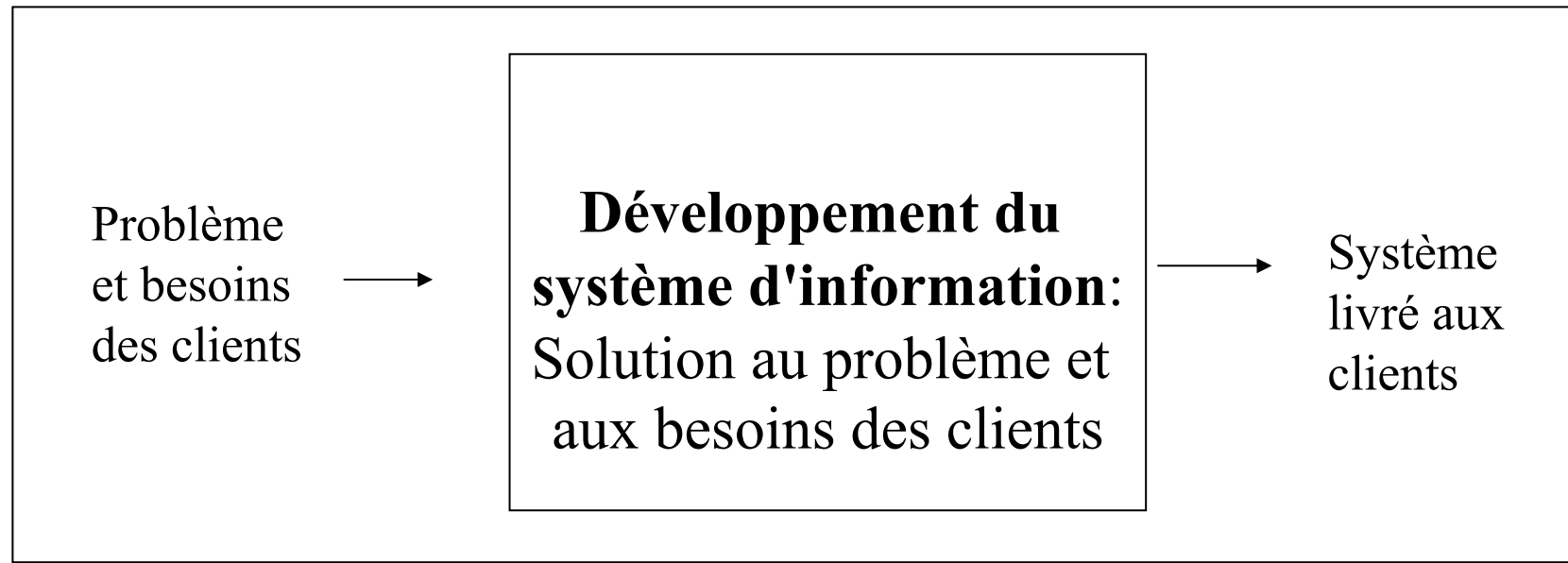
- Connaître ce qu'est un système d'information (SI)
- Son rôle dans l'organisation
- Maîtriser la notion de processus de développement des SI
  
- Savoir utiliser un diagramme de Gantt
- Savoir utiliser un diagramme Pert
- Savoir utiliser l'outil MS-Project

# Système d'information (SI)

- Un **système d'information** (ou système informatisé) un ensemble de ressources, à la fois **humaines**, **matérielles** et **immatérielles** dont le rôle est de **collecter**, **stocker**, **traiter** et **distribuer** de l'information.
- Les systèmes d'information sont très variés: système comptable, système de gestion de la paye, système de gestion des bulletins, système de gestion d'un inventaire, ...

# Le but ultime d'un SI

- A la base d'un SI, il y a des clients.
- Le but ultime d'un SI est de développer un système qui répond aux besoins des clients.



# Le développement d'un SI

- Le développement d'un projet en SI respecte les mêmes étapes que pour n'importe quel autre projet mais l'étape de la réalisation est beaucoup plus complexe
- Dans l'étape de réalisation du SI, on retrouve les 5 **phases** suivantes:
  1. L'analyse (spécifications) du SI
  2. La conception du SI
  3. La mise en œuvre du SI
  4. Le débogage et les tests sur le SI
  5. Le soutien et les mises à jour du SI

# L'analyse d'un SI

- Le but de la phase de l'analyse d'un SI est d'analyser et de décrire la problématique ainsi que les besoins des clients.
- Habituellement réalisé par des analystes, des analystes programmeurs et des experts en la matière.
- Par exemple, pour analyser un site Web relié à l'emploi, les experts sont des personnes qui œuvrent déjà dans le domaine de l'emploi.

# La conception d'un SI

- Le but de la phase de conception d'un SI est d'analyser et de décrire la solution "informatisée" au problème et aux besoins du client.
- Modèle du système à développer
- Habituellement réalisé par des analystes-programmeurs (conception des bases de données, des fonctionnalités du système, etc.) et des designers (mise en page, ergonomie, esthétisme, logos, etc.)
- Par exemple, pour concevoir le site Web relié à l'emploi, il faut des designers pour faire le design du site Web. Le site Web va également être relié à une base de données. Il faut des analystes programmeurs pour concevoir cette base de données.

# La mise en œuvre d'un SI

- Le but de la phase de la mise en œuvre est de "programmer" ce qui a été analysé et conçu dans les phases précédentes.
- Habituellement réalisé par des programmeurs-analystes (en partie) et par des programmeurs.
- Dans les grandes entreprises, les programmeurs n'ont habituellement rien "à concevoir". Ils ne font que "programmer" ce qui a déjà été analysé et conçu dans les phases précédentes par les analystes, par les analystes-programmeurs et par les designers.



# Le débogage et les tests sur le SI

- Le but de la phase de débogage et de tests est de vérifier si le système répond bien aux exigences des clients et si le système fonctionne bien.
- Souvent, on organise des rencontres avec des clients cibles afin de tester le système intensivement.
- C'est également dans cette phase que la documentation technique du système (à l'intention des informaticiens) est finalisée et que le manuel de l'utilisateur est produit (à l'intention des clients).
- Habituellement, réalisé par l'équipe d'informatique au complet.
- Parfois, cette phase est intégrée à l'intérieur de la phase de mise en œuvre.

# Le soutien et les mises à jour du SI

- Le but de la phase de soutien et de mises à jour est d'assurer un suivi sur le système auprès des clients.
- Pour les gros systèmes, l'équipe qui assure le soutien aux utilisateurs est constitué d'un grand nombre de personnes.
- Régulièrement, les utilisateurs trouvent des bogues et/ou font des propositions pour améliorer le système. On assiste alors à des mises à jour du système.
- Dans un mise à jour, habituellement, lorsque le premier numéro change c'est qu'il s'agit d'une mise à jour majeure (Exemple: 1.0, 2,0). Lorsque les autres numéros après le point changent, c'est qu'il s'agit d'une mise à jour mineure (Exemple: 2.1, 2,2, 2,3).

# Méthodologie dans le développement d'un SI

- Une méthodologie est une suite de procédures qui doit être réalisé dans un certain ordre et d'une certaine manière.
- C'est la démarche complète qu'il faut suivre pour réaliser chaque tâche dans le cycle chronologique de développement de système.
- Dans de cette démarche, on utilise toujours:
  - des modèles
  - des outils informatiques
  - des techniques
- La méthodologie de développement utilisée peut varier d'une entreprise à l'autre et d'un système à l'autre

# Modèle

**Déf** : Représentation abstraite de "quelque chose" de réel.

- Pour représenter le "réel" de manière abstraite, on utilise des éléments graphiques (schéma, diagramme, organigramme, dessins, symboles, figures géométriques, etc...)
- **Exemples de modèles pour le gestion de projet** :
  - Modèle de la suite des tâches à exécuter selon le réseau **PERT**,
  - Modèle de la suite des tâches à exécuter selon le diagramme de **GANTT**

# Modèle : Diagramme GANTT

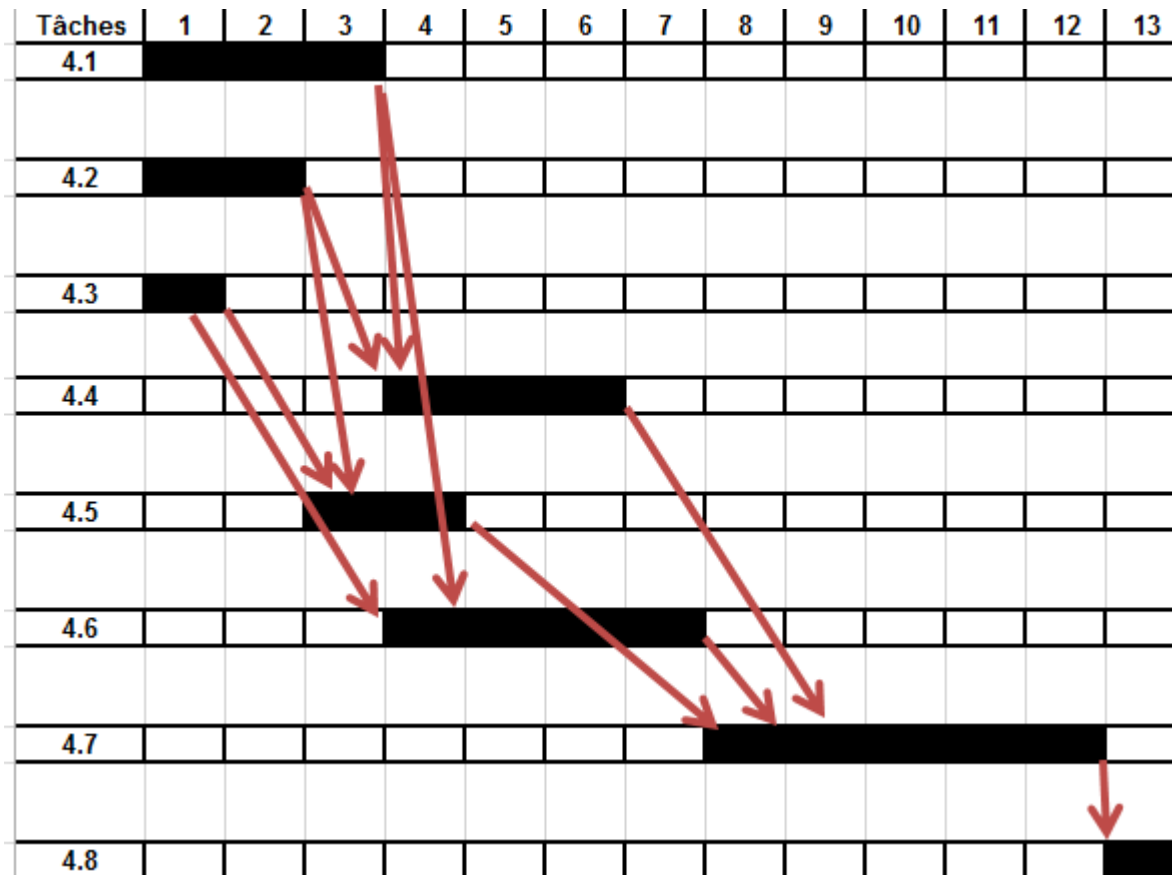
- Le **diagramme de Gantt** est une représentation graphique de la suite des tâches qui sont nécessaires pour réaliser un projet.

## Caractéristiques :

1. Chaque tâche est représentée par une droite horizontale
  - le début de la droite indique le moment où la tâche doit commencer;
  - la fin de la droite indique le moment où la tâche doit se terminer;
  - la longueur de la droite est proportionnelle au temps d'exécution de la tâche.
2. Chaque tâche débute après la fin de la plus longue des tâches préalables (après le prédécesseur qui a été terminé en dernier)
3. L'utilisation de flèches de séquence facilite la compréhension du Gantt

# Modèle : Diagramme GANT

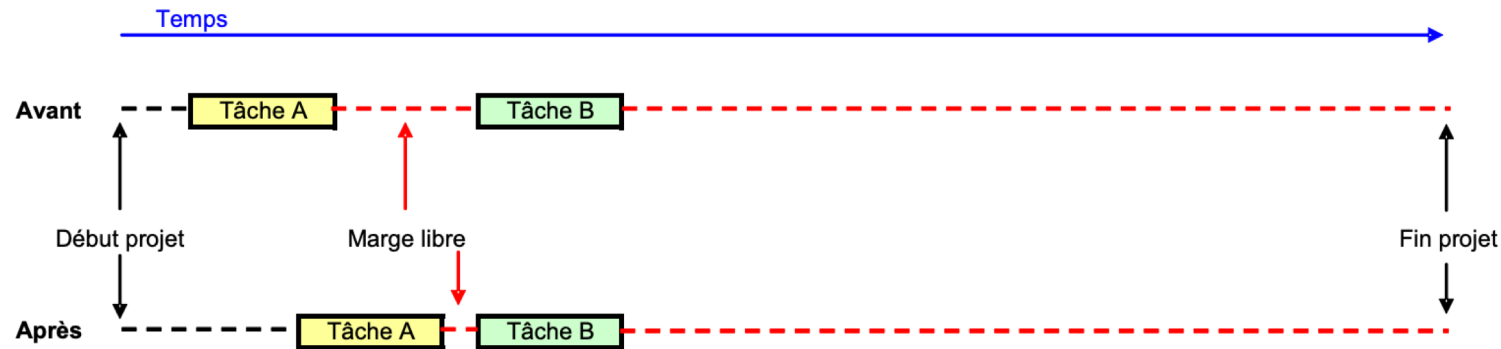
Exemple :



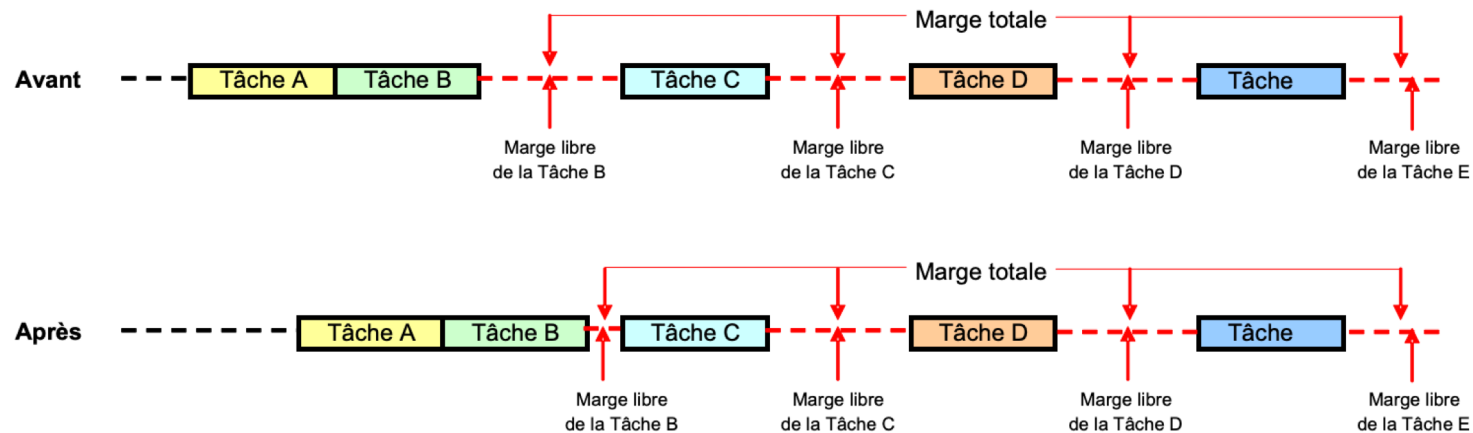
	Durée	Pré-décécesseur
4.1	3	Aucun
4.2	2	Aucun
4.3	1	Aucun
4.4	3	4.1 et 4.2
4.5	2	4.2 et 4.3
4.6	4	4.1 et 4.3
4.7	5	4.4, 4.5 et 4.6
4.8	1	4.7

# Modèle: Diagramme Gantt

**Marge libre** : la marge à disposition après une certaine tâche sur laquelle on peut empiéter sans retarder aucune autre tâche successeur.

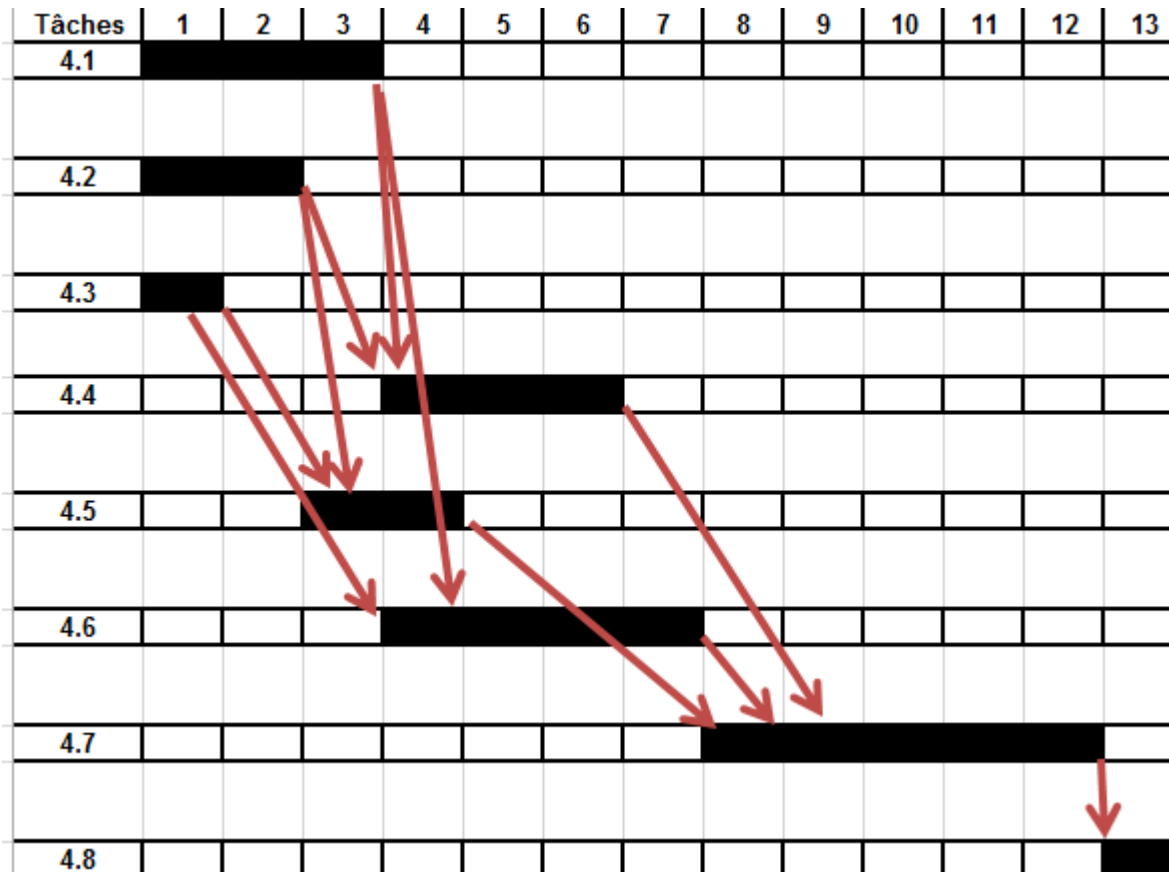


**Marge totale**: Retard permis à une tâche qui n'affectera pas la durée prévue du projet, si les autres tâches se réalisent dans le temps prévu.



# Modèle: Diagramme Gantt

Exemple marge totale :



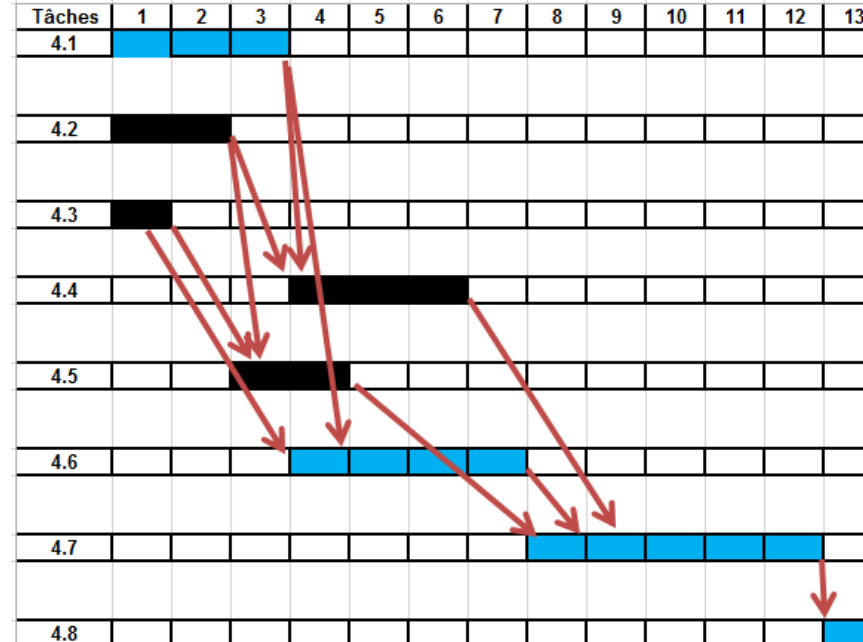
	Durée	Marge libre	Marge totale
4.1	3	0	0
4.2	2	0	2
4.3	1	1	2
4.4	3	1	1
4.5	2	3	3
4.6	4	0	0
4.7	5	0	0
4.8	1	0	0



# Modèle: Diagramme Gantt

**Tâche critique** : Tâche dont la marge totale est nulle.

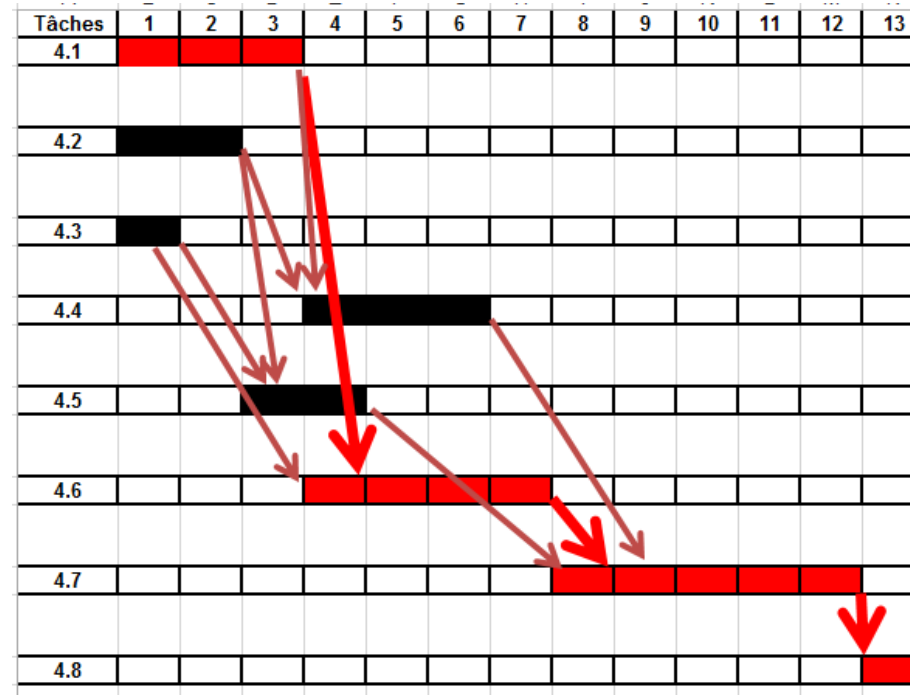
- Par exemple, la tâche 4.1 est une tâche critique.
- La tâche 4.6 est également une tâche critique.
- Les tâches 4.2 et 4.3 ne sont pas des tâches critiques.



# Modèle: Diagramme Gantt

**Chemin critique:** Suite des tâches critiques (dont la somme des temps est la plus longue ou dont la marge totale est nulle).

- Tout retard dans les tâches qui font partie du chemin critique retarde le projet.



# Modèle: Diagramme Gantt

## Avantages

- En général, facile à réaliser.
- Facile à comprendre par tous.
- Visuel (si le nombre de tâches n'est pas trop élevé).
- À l'échelle, comme un calendrier... facile à suivre.

## Inconvénients

- Parfois long à réaliser surtout s'il y a plusieurs tâches.
- Parfois difficile de bien visualiser les interdépendances entre les tâches.
- Si les durées sont variées, difficiles à produire.
- Parfois difficile de déterminer la marge totale allouée à chaque tâche.
- Parfois difficile d'identifier les tâches critiques ainsi que le chemin critique.

# Modèle: Diagramme Gantt

**Exercice:**

# Modèle : Réseau PERT

**PERT**: Project Evaluation and Review Technique

- Cette méthode consiste en une représentation graphique de la suite des tâches d'un projet mais sous la forme d'un **réseau** (et non pas sous la forme d'un diagramme).
- Contrairement à Gantt, on insiste sur la marge totale de chacune des tâches (et non pas sur la durée de chacune des tâches).

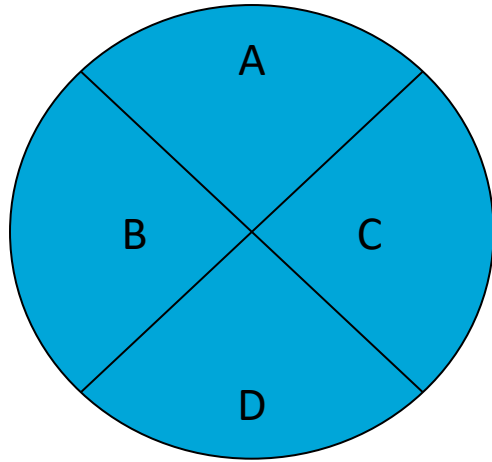
# Modèle : Réseau PERT

## Caractéristiques :

- Chaque tâche est représentée par un **nœud** (un cercle).
- Les tâches se suivent de gauche à droite en respectant les prédécesseurs.
- Les tâches parallèles sont situées, autant que possible, sur le même plan vertical (plus facile à visualiser)
- Les flèches représentent les liens entre les tâches (on indique la durée entre les nœuds)
- La longueur de la flèche est indépendante de la durée de la tâche et elle peut être courbe.
- Tout réseau **PERT** commence nécessairement par un début et se termine par un fin.

# Modèle : Réseau PERT

Nœuds du réseau :



A: Numéro de la tâche

B: Temps d'achèvement de la tâche au plus tôt (ATO)

C: Temps d'achèvement de la tâche au plus tard (ATA)

D: Marge totale de la tâche (MT)

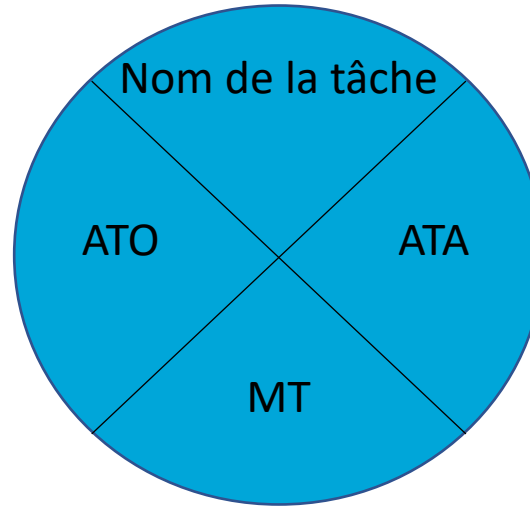
# Modèle : Réseau PERT

Calcul :

- B: Calcul du temps d'achèvement au plus tôt (ATO):
  - Quand *au plus tôt* peut se terminer chaque tâche?
- C: Calcul du temps d'achèvement au plus tard (ATA):
  - Quand *au plus tard* peut se terminer chaque tâche?
- D: Calcul de la marge totale (MT):
  - $MT = ATA - ATO$

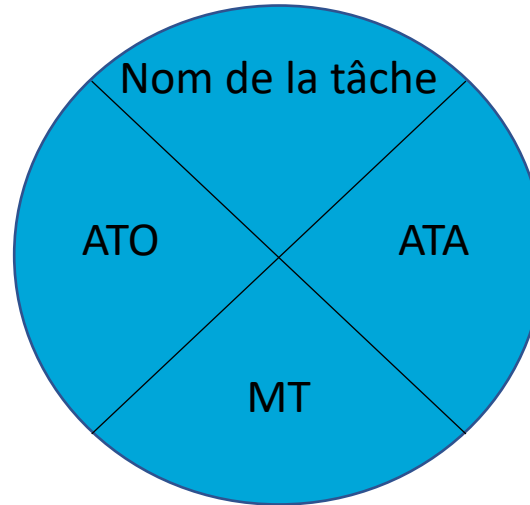


# Modèle : Réseau PERT



1. On commence par dessiner le réseau de Pert de gauche à droite en commençant par le nœud vide de début. On inscrit le nom de chaque tâche et sa durée.
2. On calcule les ATO. En allant de gauche à droite.
3. On calcule les ATA. En allant de droite à gauche.
4. On calcule les MT.

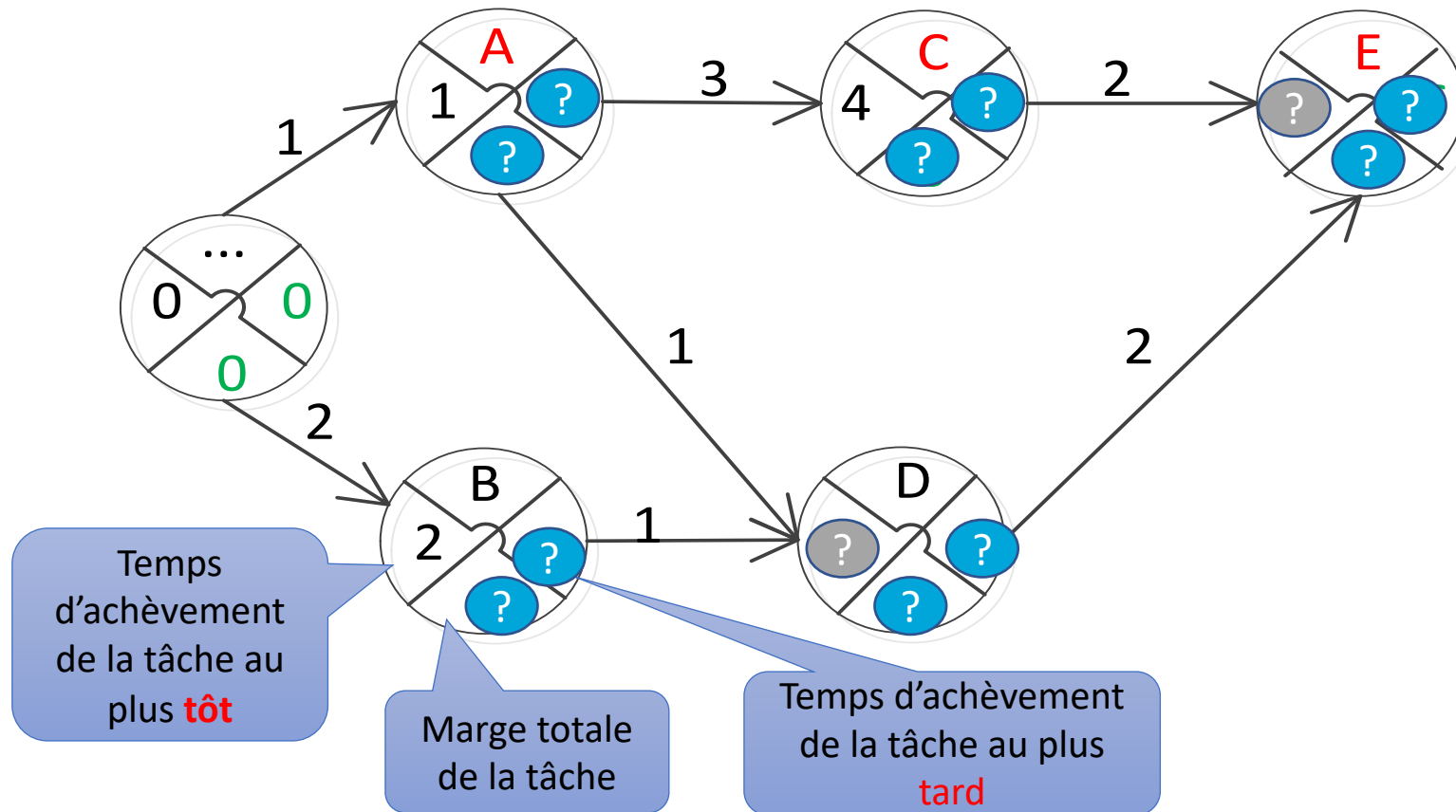
# Modèle : Réseau PERT



1. ATO => Se calcule de gauche à droite,
2.  $ATO = ATO \text{ de la tâche précédente} + \text{la durée de la tâche courante.}$
3. ATO => Si plus d'une tâche précédente, je choisis le chemin le plus long,
4. ATA => Se calcule de droite à gauche,
5.  $ATA = ATA \text{ de la tâche successeuse} - \text{durée de la tâche successeuse.}$
6. ATA => Si plus d'une tâche successeuse, je choisis le chemin le plus court.

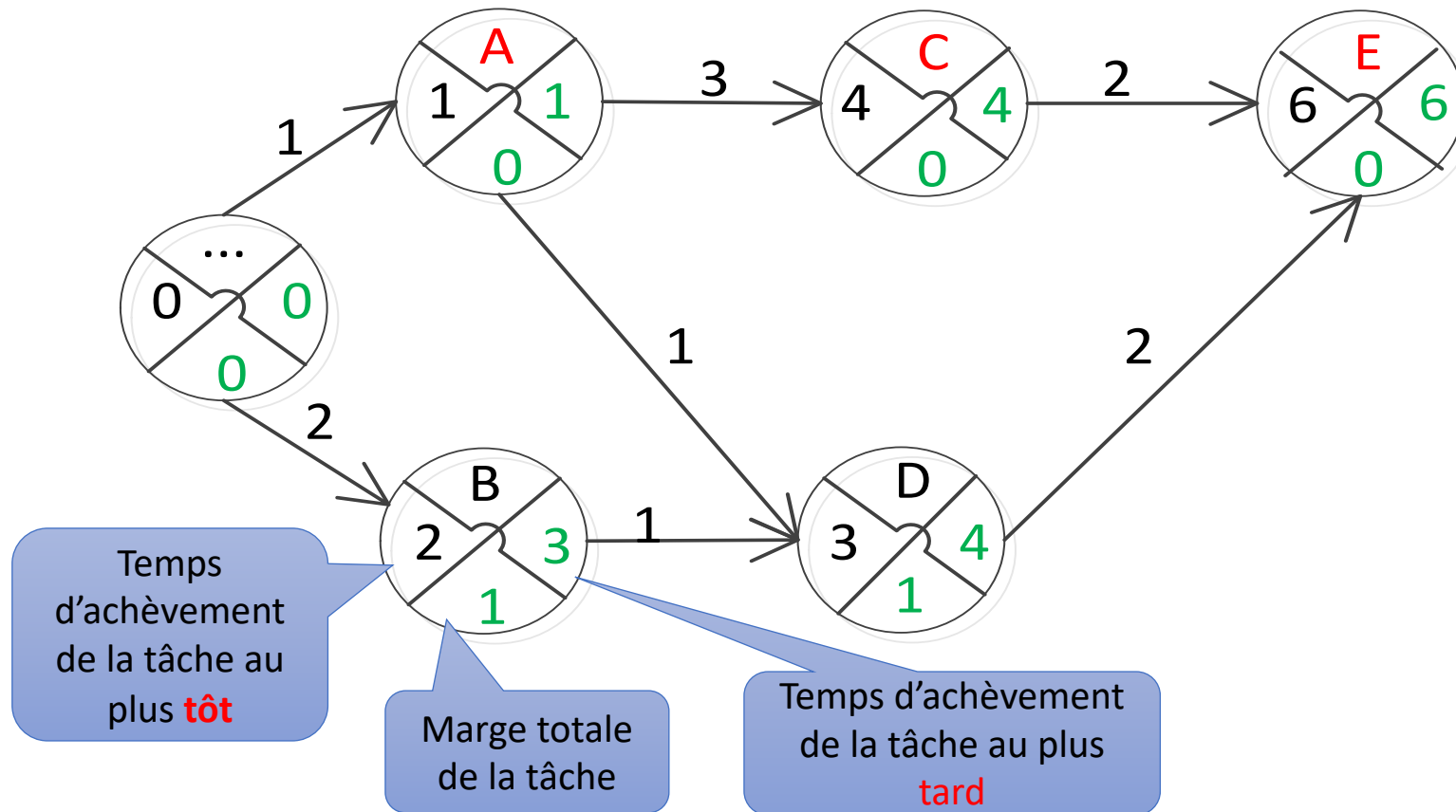
# Modèle : Réseau PERT

TÂCHES	DURÉE (EN SEMAINES)	PRÉDÉCESSEURS
A	1	Aucun
B	2	Aucun
C	3	A
D	1	A,B
E	2	C,D



# Modèle: Réseau PERT

TÂCHES	DURÉE (EN SEMAINES)	PRÉDÉCESSEURS
A	1	Aucun
B	2	Aucun
C	3	A
D	1	A,B
E	2	C,D



# Calcul des valeurs en résumé

1. En partant du nœud d'origine, évaluer les ATO en prenant la valeur maximale lorsque plusieurs chemins mènent à un nœud.
2. En partant du nœud final, évaluer l'ATA puis le MT. Dans le cas où plusieurs chemins mènent à un nœud, prendre la valeur minimale comme ATA.
3. Le chemin critique est constitué des nœuds dont le  $MT = 0$ .

# Modèle: Réseau Pert

## **Avantages:**

- Les interdépendances entre les tâches sont faciles à visualiser.
- La marge totale allouée à chacune des tâches est directe.
- Les tâches critiques sont faciles à visualiser (celles dont la marge totale est de 0).
- Le chemin critique est facile à visualiser (la suite des tâches critiques).

## **Inconvénients :**

- Long et pénible à réaliser pour de grands projets.
- Plus difficile à comprendre par tous.

# Modèle: Réseau Pert

**Exercice.**

# Modèle pour l'analyse et la conception d'un SI

Dans les phases d'analyse et de conception, les modèles utilisés pour les composantes du système d'information:

- Ordinogrammes
- Diagramme de flux de données
- Diagrammes entité-relation
- Diagramme des cas d'utilisation
- Diagramme des classes



# Outils informatiques

**Déf :** Logiciels qui aident à créer des modèles ou d'autres composantes nécessaires au SI

**Exemples:**

- **Ms-Project**: Gestion de projets
- **Visio**: Créer des ordinogrammes et des diagrammes
- **PhotoShop**: Dessin, graphisme
- **Word**: Traitement de texte
- **Excel**: Créer des tableaux
- **Access**: Pour créer et gérer la base de données
- ...