❶ Système d'information et informatique

- **Le système d'information d'une organisation (au sens large: société, entreprise, institution, club, groupe structuré ...) regroupe tout ce qui à quelque niveau que se soit traite ou stocke des informations relatives à l'organisation concernée.**

- **Le système d'information comprend des informations relatives:**

  - aux flux : produits en stocks, produits commandés, bons de livraison, factures, bons de commandes...
  - à l'univers extérieur: clients, fournisseurs...
  - à l'organisation de l'entreprise: que se passe-t-il entre l'enregistrement d'une commande et sa livraison?
  - aux contraintes légales: lois, règlements, paramètres financiers...
  - etc.

- **Dans l'approche d'un système d'information, on distingue :**
  - la formalisation des données (informations de toute nature) présentes à un moment ou un autre dans le système,
  - la formalisation des traitements (i.e des processus dynamiques) intervenant dans le système.
- **La formalisation des données mène à l'élaboration d'un modèle de données .**
- **La formalisation des traitements se traduit par la définition de modèles de traitements .**
- **Ce travail de formalisation d'un système d'information constitue ce que l'on appelle : une analyse informatique.**

- **De manière générale seule une partie du SI peut être automatisée. Cette partie sera appelée Système Automatisé d'Information (SAI)**
- **Qu'est-ce qui peut être informatisé?**
  - **Mémorisation d'informations:**
    - » stockage des données,
    - » structuration des données,
    - » ...
  - **Traitements:**
    - » contrôle des données,
    - » mise à jour,
    - » recherches,
    - » calculs.
  - **Interface entre l'univers et le SI: saisie, accès.**
- **Le SAI communique avec son environnement extérieur par des saisies et des accès. Il contient une partie mémorisation et une partie traitements.**

Pourquoi l'analyse informatique?

- La difficulté essentielle dans la réalisation du SAI réside dans le fait qu'elle **concerne un nombre important de personnes**, de caractéristiques très variées (la direction, le service informatique, les responsables de service, les utilisateurs terminaux) d'où la nécessité **d'une méthode**.

- **Les premières informatisations se sont souvent révélées catastrophiques (la liste suivante n'est pas exhaustive):**

  - logiciel ne fonctionnant pas ou ne réalisant pas ce pourquoi il avait été prévu,
  - logiciel incapable d'évoluer avec le matériel, la taille de l'entreprise, les changements dans l'entreprise,
  - études correctes sur le papier mais impossibles à implanter,
  - logiciel correct mais mettant trop de temps à délivrer les résultats,
  - informatisation bouleversant l'organisation humaine de l'entreprise, rejetée par le personnel...

L'approche classique de mise en place d'une application informatique

➢ **Dans une entreprise** L'approche **classique** de mise en place d'une **application informatique** consistait le plus souvent à :

- ✓ l'**écriture** d'un certain nombre de **programmes**
- ✓ **programmes** destinés à **l'exploitation** d'un ensemble **de fichiers** qu'il fallait **aussi créer**.

➢ **Les principaux problèmes** posés par cette démarche sont :

- **la redondance** des informations
- **la dépendance entre les données et les programmes qui les manipulent.**

La Redondance des informations

⇨ **Redondance** = une même information peut se trouver dans **plusieurs** fichiers distincts.

Ex : l'identité d'un employé (nom, prénom, adresse, etc.)

- ➢ peut figurer dans un fichier **F1** propre à une application **P1** calculant la paie,
- ➢ et dans un fichier **F2** propre à une autre application **P2** de gestion des remboursements des frais médicaux.

⌦ **Cause :**

- création **anarchique** de fichiers
- **absence de coordination** entre les groupes de développement **entraîne une duplication non contrôlée** des informations

Conséquences :

La redondance **complique** les opérations de mise à jour

- il est nécessaire pour **une même information mise à jour dans un fichier** de la **mettre à jour dans tous les fichiers** où elle est supposée exister.

Ex :

- si on **change l'adresse** d'un employé dans le **fichier F1**,
- **il faut faire aussi ce changement** dans tous les fichiers contenant **l'adresse de cet employé** .

- Ceci devient **de plus en plus difficile** dès que le **nombre de fichiers** est **élevé**
- Souvent on a **tendance à retarder** **la mise à jour** dans les autres **fichiers** ce qui engendre une situation **incohérente**

situation incohérente :

- l'interrogation de la même information par deux **utilisateurs différents** **chacun** sur **son propre fichier** fournira **2 résultats différents.**

La Dépendance entre les données et les programmes

⌦ Généralement, la **création d'un fichier** sous entend la réalisation d'un **certain nombre d'opérations** telles que :

- La structuration des enregistrements :

  définir les champs, leurs types et leur ordre

- Le choix d'une organisation :

  séquentielle, séquentielle indexée, directe, etc.

- Le choix du support :

  disque dur, bande magnétique, etc.

☝ Cette **création** était donc **figée** car faite en fonction d'un ou de plusieurs programmes.

- Les **données contenues** dans les **fichiers** sont **directement associées aux programmes** qui les exploitent
- par le biais d'une description contenue dans ces programmes eux-mêmes.

Exemple :

- Pour mettre en place une application de gestion de prêts dans laquelle on suppose qu'un **emprunteur peut emprunter jusqu'à trois (3) ouvrages**, on doit créer un fichier dont les enregistrements auront à peu près la structure suivante :

Nom	Prénom	Adresse	Ouvrage 1	Ouvrage 2	Ouvrage 3

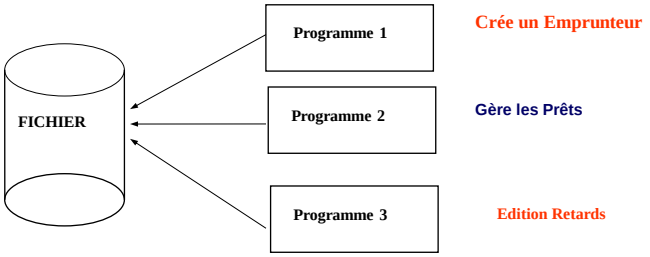
- Les programmes qui utilisent un tel fichier en donnent la description à l'intérieur du programme même.

Exemple :

Si on se réfère par exemple au langage COBOL, une telle description aurait l'allure suivante :

```
01      PRET
        02      NOM             PICTURE         A(20)
        02      PRENOM          PICTURE         A(20)
        02      OUVRAGE         OCCURS   3      TIMES
                03              COTE_OUVRAGE    9(6)
```


- Si **ce fichier** est utilisé par les programmes P1, P2, P3



- **Si on doit le modifier pour le besoin de P2** : prise en compte de l'âge de l'emprunteur par exemple,
  - ⇨ il **faudra modifier P1 et P3** en conséquence.

- Ceci est dû au fait que tous les programmes **voient** le fichier de la même façon **bien qu'ils n'aient pas tous besoin des mêmes informations.**

⌦ un **changement de l'organisation** du fichier sur disque pour le besoin d'un programme **Pi** (souci d'optimisation des temps d'accès de **Pi**)

- **obligera les autres programmes à prendre en compte ce changement même s'ils n'ont pas vraiment besoin de ce type d'organisation.**

➢ une **réorganisation des champs de l'enregistrement** (changement de l'ordre des champs à l'intérieur de l'enregistrement) entraînera une modification dans les programmes et une reconstruction du fichier sur le support.

⌦Ceci montre qu'il **existe une dépendance étroite** entre les données et les programmes qui les manipulent

➢ **Elle se traduit par le fait suivant :**

il n'est **pas possible** de changer la structure et **l'organisation** des données **sans modifier** en conséquence les **programmes**.

⌦il **manque** un **moyen de filtrer les informations** afin de ne **retenir** que **celles** qui sont **utiles** à **un programme donné**.