

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche  
Scientifique  
Université Constantine 2 - Abdelhamid Mehri



Faculté des Nouvelles Technologies de l'Information et la Communication

Polycopie de Cours  
destiné aux étudiants de License2 en Informatique

Intitulé du cours  
*Introduction aux Systèmes d'Information*

Préparé par  
Mr. Abdelhafid ZITOUNI  
Département TLSI

# Table des Matières

## Chapitre 1: Méthodologie d'analyse et de conception des SI

Introduction.....	1
1.1 Généralité et définitions.....	1
1.1.1 Processus.....	1
1.1.2 Information.....	2
1.1.3 Systèmes.....	2
1.1.4 Système d'information ...	2
1.1.5 Modèle.....	2
1.1.6 Analyse et conception...	2
1.2 Différentes approches .....	3
1.2.1 Approche cartésienne .....	3
1.2.2 Approche systémique.....	3
1.3 Méthodes générales d'analyse et de conception d'un SI .....	4
1.3.1 Notion de méthode.....	4
1.3.2 Familles de méthodes.....	5
1.3.3 Enjeux du développement informatique.....	6
1.4 Ingénierie des systèmes d'information.....	7
1.4.1 Développement des systèmes d'information .....	7
1.4.2 Méthodologie de développement .....	7
1.5 La modélisation conceptuelle .....	8
Conclusion.....	9

## Chapitre 2: L'entreprise et son système d'information

Introduction.....	10
2.1 L'entreprise (l'organisation).....	10
2.1.1 Notion d'organisation.....	10
2.1.2 Qu'est-ce qu'une entreprise?.....	10
2.1.3 Classification des entreprises.....	11
2.2 Le système d'information dans une entreprise.....	11
2.2.2 Les circuits d'information.....	12
2.2.3 Les cycles d'information.....	13
2.3 Représentation systémique d'une organisation.....	13
2.3.1 Notion de système.....	13
2.3.2 L'entreprise vue en tant que système.....	14
2.4 Le système d'information.....	15
2.5 Le système d'information automatisé.....	15
2.5.1 Actions programmées et décisions.....	15

2.5.2 Système d'Information Automatisé.....	16
2.5.3 Sous-systèmes fonctionnels du SIA.....	16
2.6 Qualités d'un système d'information informatisé .....	17
2.6 Exercice .....	18

### **Chapitre 3: La codification des informations**

Introduction.....	19
3.2 Objectifs de la codification.....	19
3.3 Qualités attendues d'une codification.....	20
3.4 Différents types de codification.....	20
3.4.1 La codification séquentielle.....	20
3.4.2 La codification par tranches.....	20
3.4.3 La codification articulée.....	21
3.4.4 La codification à niveaux (ou hiérarchique).....	21
3.4.5 La codification mnémonique.....	22
3.5 Conduite des opérations de codification .....	22
3.6 Exercice.....	23

### **Chapitre4: Méthodologie MERISE**

Introduction.....	24
4.1 Qu'est ce que MERISE.....	24
4.2 Les niveaux d'abstraction.....	25
4.2.1 Le niveau conceptuel.....	25
4.2.2 Le niveau organisationnel.....	25
4.2.3 Le niveau physique.....	25
4.2.4 Opérateurs parallèle.....	27
4.3 La démarche préconisée par MERISE.....	27
4.3.1 L'étude préalable.....	27
4.3.2 L'étude détaillée.....	28
4.3.3 La réalisation.....	28
4.3.4 La phase de mise en œuvre.....	28
4.5 Les trois axes d'un projet informatique.....	28
Conclusion.....	29

### **Chapitre 5: Le Modèle Conceptuel des Données (MCD)**

Introduction.....	30
5.1 Présentation des concepts et du formalisme du MCD.....	30
5.1.1 Concepts de base.....	30
5.1.2 Règles de base pour la construction d'un MCD.....	35
5.2 La construction du modèle conceptuel de données (MCD).....	40
5.3 Exercices.....	43

### **Chapitre 6: Le Modèle Conceptuel des Traitements (MCT)**

Introduction.....	46
6.1 Le Modèle Conceptuel des Traitements (MCT).....	46

6.1.1 Généralités.....	46
6.1.2 Exemple de MCT.....	47
6.1.3 Concepts .....	47
6.1.4 Construction du MCT.....	49
6.1.5 Règles de vérification d'un MCT.....	53
6.2 Exercices.....	54

## **Chapitre 7: Le Modèle Logique des de Données (MLD)**

Introduction.....	56
7.1 Problématique.....	56
7.2 Les concepts relationnels.....	57
7.3 Règles de passage du MCD au MLD.....	57
7.4 Les formes normales.....	59
7.5. Contrats structurels.....	59

<b>Références bibliographiques.....</b>	<b>61</b>
---	-----------

# CHAPITRE 1

## Méthodologie d'Analyse et de Conception des SI

### Introduction

Dans le cadre de la modélisation des Systèmes d'Information (SI), de l'expression des besoins à la conception et à la réalisation de solutions logicielles, de nombreuses représentations souvent hétérogènes sont utilisées. Elles se différencient pour exprimer différents niveaux (externe, conceptuel, logique, etc.) ou différents points de vue (statique/dynamique/fonctionnel, comportement externe, contexte, architecture du logiciel,...). Ces représentations constituent des modèles (ou un modèle) du système d'information. Ces modèles ont différents objectifs selon le moment où on les écrit ou celui où on les utilise : on parlera de modèle de spécification, de conception, d'implantation, etc. Les nombreux acteurs qui interagissent (décrivent, évaluent, utilisent, etc.) dans cette modélisation ont des compétences et souvent aussi des objectifs différents.

Les concepts et les notations utilisés par ces représentations sont variés. Certaines représentations s'appuient sur des langages libres ou un peu structurés, d'autres sur des langages précis et formels. Dans le domaine des systèmes d'information, ce sont essentiellement des représentations graphiques qui sont utilisées ; elles sont qualifiées de langages semi-formels.

### 1.1 Généralité et définitions

**1.1.1 Processus :** Ensemble d'opérations, logiquement liées, aboutissant à certains résultats. En conception de systèmes d'information, selon la méthode Merise, le processus se situe au niveau du modèle conceptuel de traitement.

Étapes ou phases de la méthode d'exécution ou de fonctionnement de quelque chose, que ce soit dans les systèmes commerciaux ou techniques, mettant en cause les actions de plus d'une personne, d'une unité ou d'une division. Système d'opérations dans la production de quelque chose. Série de mesures, de changements ou de fonctions qui produisent un résultat final.

**1.1.2 Information :** encore appelée donnée. Élément de connaissance susceptible d'être codé pour être conservé, traité ou communiqué.

**1.1.3 Systèmes :** Un système est un assemblage d'éléments reliés entre eux compris dans un ensemble plus grand. En latin et en grec, le mot « système » veut dire combiner, établir, rassembler. Un sous-système est un système faisant partie d'un autre système. Généralement, un système est constitué de composants (ou d'éléments) organisés ensemble dans le but de faciliter le flux d'informations, de matières ou d'énergie. ...

**1.1.4 Système d'information :** ou (information system) est un ensemble de composants de traitement de l'information et de communication, ainsi que l'environnement dans lequel ils opèrent. (En Anglais: MIS ou Management of Information Systems).

**Ensemble structuré:**

(1) de données, de leurs traitements et de leurs communications, décrit à l'aide de structures, de procédures et de protocoles → dictionnaire de données (en Anglais: DD ou Data Dictionary)

(2) de moyens techniques: Le système informatique (matériel et logiciel) et de communication de documents ayant pour but de générer, mémoriser, traiter, transférer et exploiter des informations dans le cadre d'objectifs définis.

**1.1.5 Modèle** (référence à la modélisation du SI) est une représentation de la réalité. Le modèle permet d'**observer un système d'information**, en permettant de **tester sa réaction** aux divers événements de la vie de l'organisation.

**Un modèle** est. la mise en œuvre d'un certain nombre d'étapes (méthodologiques) :

- une démarche
- des principes
- des outils
- un vocabulaire

**1.1.6 Analyse et conception :**

**Analyse :** consiste d'une part à **comprendre et modéliser le fonctionnement d'un domaine d'étude** du système d'information de l'organisation, et d'autre part à **préparer et concevoir la solution informatique** adéquate.

On distingue deux types d'analyse :

- l'analyse de l'existant (étude du domaine),
- l'analyse de conception (préparation d'une solution technologique).

**Conception :** Création d'un objet, d'un système : action qui donne naissance à quelque chose qui n'existe pas.

- L'analyse et la conception des systèmes d'information automatisé (CSI, ACSI, ACSIA, CSIA) consiste un ensemble de technique et de méthode destinés à améliorer l'efficacité du SI en fonction des objectifs des organisations correspondantes.

## 1.2 Différentes approches

### 1.2.1 Approche cartésienne

Cartésien —→ **Descartes** : " Diviser pour mieux résoudre "

Cette approche consiste à découper l'ensemble des procédures de gestion de l'entreprise en application indépendant qui pouvant être étudié séparément sans tenir compte des autres applications.

**Exemple :** 1- Décomposition de l'université en faculté

- 2- Dans une entreprise commerciale :
- Gestion du personnel.
  - Gestion des clients.
  - Gestion des fournisseurs.
  - Gestion des stocks.

#### **A-Avantage :**

1. Simplicité de mise en œuvre.
2. Possibilité du traitement des applications en parallèle.
3. Pas de modification profonde de structure de l'entreprise.
4. Facilité de maintenance.
5. Facilité d'estimation des coûts de fonctionnement.

#### **B-Inconvénients :**

1. Difficulté de mettre en pratique des entités indépendantes
2. Peut augmenter les coûts de développement
3. Problème d'arrêt de la décomposition
4. Pas de modification de la structure alors qu'elle peut être source de dysfonctionnement.

### 1.2.2 Approche systémique

**Constat :** Une organisation ne peut pas toujours se décomposer en application indépendante.

La résolution des différents sous problèmes indépendants n'impliquent pas forcément la résolution du problème global.

- l'approche systémique consiste donc à considérer les sous systèmes aussi indépendant que possible et à les traiter en tenant compte de leur interaction.

**Exemple :** Dans une faculté on peut considérer :

La bibliothèque, la gestion du personnel, les départements...etc.

- a- Comme des sources indépendantes (approche cartésienne)
- b- Comme des sources en interaction (approche systémique).

## **A-Avantage**

1. Meilleure prise en compte de la réalité
2. possibilité de remise en question de l'organisation existante
3. solution intégrée et coopérative

## **B-Inconvénients :**

1. Plus complexe à mettre en œuvre
2. remise en cause de l'organisation existante
3. Plus difficile de traiter en parallèle.

## **1.3. Méthodes générales d'analyse et de conception d'un SI**

### **1.3.1 Notion de méthode :**

#### **Qu'est ce qu'une méthode ?**

- Une démarche reflétant une philosophie générale
- Propose des outils spécifiques pour manipuler des concepts aptes :
  - . à donner une représentation fidèle des systèmes étudiés,
  - . à favoriser l'innovation
  - . à présenter un suivi efficace grâce à des aides documentaires.

Ainsi donc une méthode est un ensemble de démarches raisonnées suivies pour parvenir à un but.

L'objectif des méthodes de conception et de développement de SI est de représenter une démarche et un ensemble de modèles permettant de mettre en place un nouveau système.

La méthode est faite donc d'une part pour pallier la démarche intuitive, et d'autre part, pour maîtriser la complexité des problèmes à résoudre.

#### **Une méthode pourquoi ?**

##### **+ Maîtriser le développement des systèmes**

- Contrôler les coûts, et les délais
- Répondre aux besoins nés de l'évolution de l'entreprise
- Intégrer les innovations techniques
- Augmenter le service rendu par la structure.

##### **+ Faciliter la communication**

- Normaliser le vocabulaire
- Homogénéiser la présentation de solutions
- Améliorer l'expression des besoins

##### **+ Assurer la pérennité des systèmes**

- Apporter des aides à la maintenance
- Améliorer les performances des systèmes.



Une méthode de conception de systèmes d'information est nécessaire pour formuler le problème informationnel et maîtriser la résolution pour construire un SI cohérent, pertinent, complet, fiable, flexible et adaptatif.

Toute méthode met en oeuvre 4 composants indissociables:

- **Des modèles** : Ensemble de concepts et règles pour utiliser ces derniers destinés soit à expliquer et représenter les phénomènes organisationnels, soit à expliquer et à représenter les éléments qui composent le SI et leurs relations.

- **Un langage** : Ensemble de constructions qui permettent de décrire formellement les images du SI élaborées aux différents stades du processus de conception, éventuellement en faisant appel à des méthodes.

- **Une démarche** : C'est un processus opératoire par lequel s'effectue le travail de modélisation, de description, d'évaluation et de réalisation du SI.

- **Des outils** : Ce sont les outils logiciels supportant la démarche (outils de documentation, d'évaluation, de simulation, d'aide à la conception ou à la réalisation).

### **1.3.2 Familles de méthodes :**

Dans le domaine des systèmes d'information, les méthodes concernent :

- Le schéma directeur ;
- La conception de système d'information, l'analyse détaillée et la réalisation (avec, souvent, l'accent mis sur l'un de ces trois aspects) ;
- La conduite de projet et l'amélioration de la qualité.

Les méthodes de conception sont elles-mêmes présentées selon l'ordre suivant :

- **Les anciennes méthodes analytiques ou cartésiennes (Corig, RSA,...)** : elles permettent plus de décrire un système existant que de le critiquer ou de concevoir un nouveau système. Certaines de ces méthodes précisent un processus de développement, mais négligent la signification de l'information. Dans le domaine de la gestion, la plupart des méthodes analytiques ne sont plus utilisées ;
- **Les méthodes systémiques (MERISE, Rémora,...)** : elles considèrent l'entreprise comme un système à part entière. Comme dans tout système, on distingue trois sous systèmes : système de pilotage, système d'information et système opérant. En outre, ces méthodes se caractérisent par la description des relations entre informations, une

modélisation du domaine concerné de l'entreprise, une circulation des informations correspondant à la pyramide de la décision et un cycle d'abstraction allant du plus général au plus précis.

- **Les méthodes à objets (OMT, Grady Booch, UML, ...)** : l'approche objet est moins intuitive que l'approche fonctionnelle. Contrairement aux techniques classiques qui s'intéressent à décomposer un problème informatique sous forme d'une hiérarchie de fonctions atomiques et de données, l'approche objet se concentre principalement sur l'identification des objets du domaine d'application et de leurs interactions. Le terme « orienté objet » signifie que le système est organisé comme une collection d'objets associés comportant à la fois une structure de données et un comportement.

Il est à noter que toutes ces familles de méthodes, ayant le même objectif, diffèrent par la façon d'appréhender le projet d'informatisation d'un SI. Ces différences résident au niveau des éléments suivants :

- Cadre général de réflexion (principes fondamentaux)
- Démarche

Etapes de mise en œuvre

- Raisonement
  - Langages
  - Modèles
- Moyens de mise en œuvre
  - Organisation
  - Partenaires et rôles définitifs
  - Outils logiciels...

### 1.3.3 Enjeux du développement informatique

Répondre mieux et plus vite aux besoins des utilisateurs finaux en assurant la maîtrise des coûts, tout en garantissant la pérennité des investissements, voilà les impératifs majeurs auxquels doivent répondre les décideurs dans tous les domaines.

L'informatique se caractérise de plus par une forte évolutivité des besoins et des techniques qui conduit à une visibilité restreinte sur l'avenir. Il faut donc absolument prendre en compte les points clés suivants :

- Les demandes des utilisateurs se renouvellent et s'accroissent, en quantité comme en qualité.
- Les matériels et les logiciels de base sont en mutation permanente et rapide.
- Les ressources pour satisfaire les demandes diminuent plus qu'elles n'augmentent.

La mise en place d'un système informatisé, c'est à dire l'informatisation du système d'information (SI) de l'entreprise ne peut être efficace que si celle ci est dirigée du début à la fin par une méthode rationnelle et compréhensible par les différents acteurs qui interviennent tout au long de la vie du projet.

Par la définition d'un certain nombre de règles strictes, la modélisation du système à l'aide de la méthodologie **MERISE**, qui tire partie des principes de la systémique, garantie d'une part la cohérence des données et traitements et donc la validité d'une application et d'autre part la pérennité des développements réalisés grâce à une "maintenabilité" accrue.

L'utilisation d'une telle méthode est donc primordiale pour tout développement informatique actuel afin d'assurer la cohérence entre le cahier des charges initial, issu de la volonté des dirigeants et l'application livrée à l'utilisateur final.

## **1.4 Ingénierie des systèmes d'information**

L'utilisation de la modélisation conceptuelle dans le développement des systèmes d'information a pour objectif une prise en compte plus adéquate des besoins des applications dans leur environnement d'utilisation. La modélisation conceptuelle consiste à représenter de manière abstraite, c'est-à-dire en termes de concepts familiers aux domaines d'application et indépendamment des technologies d'implémentation.

### **1.4.1 Développement des systèmes d'information**

#### **A. Crise du Logiciel**

Malgré les progrès constants et considérables des technologies de l'informatique, les systèmes d'information complexes répondent souvent bien moins que parfaitement aux besoins qu'ils sont supposés satisfaire. Le terme de « génie logiciel » (software engineering), discipline qui peut se définir comme l'ensemble des processus méthodologiques et des produits pertinents au développement de logiciels à grande échelle, a d'ailleurs été inventé pour susciter une recherche méthodologique plus appropriée.

La crise du logiciel a plusieurs causes:

- La complexité toujours croissante des systèmes d'information modernes: les progrès constants de la technologie informatique suscitent et rendent possibles des applications toujours plus ambitieuses;
- La sous-estimation traditionnelle de la difficulté et donc, du coût du développement de logiciels (les méthodes largement intuitives de la programmation individuelle ne s'étendent pas à la construction de grands systèmes complexes), avec comme conséquence un processus de développement qui se révèle être souvent complexe, long et coûteux.
- une fiabilité et une maturité plus faibles pour les technologies du logiciel que pour celles du matériel, alors que l'importance relative du logiciel dans la réalisation des fonctions de systèmes complexes ne cesse de croître.

### **1.4.2 Méthodologie de développement**

Les questions méthodologiques sont plus complexes que dans des disciplines plus anciennes et donc plus mûres. En informatique, il est possible de commencer la programmation avant que la phase de conception soit terminée, parce que des modifications incrémentales aux programmes sont moins onéreuses que les modifications

correspondantes dans des productions plus tangibles (génie civil ou fabrication mécanique, par exemple).

Les méthodes modernes de conception de logiciel s'accordent sur une définition générale des phases de ce qu'on appelle le cycle de vie du logiciel:

- L'analyse précise des besoins à satisfaire par le futur système, on parle aussi de phase d'ingénierie des besoins; il s'agit d'appréhender le domaine du problème posé et de spécifier le comportement externe (boîte noire) d'un système qui peut résoudre ce problème.

Cette phase implique une modélisation adéquate des concepts pertinents du monde réel, la modélisation conceptuelle;

- La conception d'une solution au problème posé et de la structure générale d'un système qui va le résoudre;

- L'implémentation de l'architecture à l'aide de la technologie choisie (typiquement, un langage de programmation);

- Le test et la validation de l'implémentation, et la mise en opération du système;

- L'adaptation, ou maintenance, du système opérationnel aux changements des besoins et de l'environnement d'utilisation.

Les idées sur les qualités souhaitables des méthodes de développement d'applications ont beaucoup évolué. Par exemple, un apport important de la technologie des objets est que la structure des logiciels développés avec cette technologie peut refléter bien mieux celle du problème posé, ce qui simplifie le développement et sa compréhension.

L'évolution, toujours en cours, des idées sur la méthodologie de développement des systèmes d'information tend donc à consacrer une part plus grande de l'effort global aux phases en amont du processus (c'est-à-dire l'analyse des besoins et la modélisation conceptuelle). C'est un changement important par rapport aux pratiques, où, souvent, ces phases en amont ne sont considérées que comme des aides servant à maîtriser une phase de programmation complexe, considérée comme centrale au processus de développement.

## **1.5 La modélisation conceptuelle**

L'idée qui sous-tend la modélisation conceptuelle est très simple: il s'agit de développer un modèle d'un domaine d'application particulier en termes des concepts familiers aux acteurs de ce domaine.

Un bon modèle conceptuel implique un processus d'abstraction, de simplification d'une situation perçue comme complexe dans le monde réel et de suppression des détails dont l'effet sur la solution au problème posé est minime ou inexistant. La solution au problème est donc aussi simplifiée, et son implémentation rendue plus performante.

Les modèles conceptuels pourraient remplir plusieurs fonctions dans le cycle de vie d'un produit logiciel:

- améliorer la compréhension des structures et fonctions d'un domaine du monde réel, grâce à un modèle basé sur des concepts clairs et intuitifs proches de ceux utilisés par les acteurs du domaine. Un modèle conceptuel peut être utile pour mieux comprendre la complexité;

- fournir un vecteur de communication précis entre modélisateurs et développeurs du système d'une part, et spécialistes du domaine d'application et utilisateurs finaux d'autre part;
- spécifier et guider les phases aval du cycle de vie du système telles l'implémentation,

## **Conclusion**

La modélisation conceptuelle consiste donc à construire des représentations abstraites de certains aspects des systèmes physiques et sociaux et de leur environnement dans le monde qui nous entoure. La modélisation est une entreprise interdisciplinaire, où les informaticiens se doivent d'étudier les domaines d'application suffisamment pour en comprendre les concepts importants et les termes dans lesquels se posent les problèmes, et les spécialistes des domaines d'application se doivent de pouvoir appréhender les possibilités de la technologie du moment.

La modélisation conceptuelle nécessite des notations, outils et techniques de représentation de l'information et des traitements concernant un domaine d'intérêt. Ces techniques de modélisation peuvent également être intégrées dans des outils d'aide à la conception d'applications informatiques, ou ateliers de génie logiciel.

Des progrès en modélisation conceptuelle consistent à définir des langages de modélisation simples, puissants et de haut niveau, qui permettent de réduire la distance entre les concepts familiers aux acteurs dans les domaines d'application et leur représentation dans les modèles.

# CHAPITRE 2

## L'entreprise et son système d'information

### Introduction

Nous nous plaçons dans une optique d'informatique au service de l'entreprise. Il est donc nécessaire de nous interroger sur la nature de l'information et sur la façon dont elle circule dans l'entreprise.

### 2.1 L'entreprise (l'organisation)

#### 2.1.1 Notion d'organisation

Une organisation est un ensemble de personnes, de techniques, de procédés, de méthodes, de matériels, etc. , mis en œuvre de façon coordonnée dans un but économique, social, administratif,...etc.

- Organisation économique : entreprise
- Organisation Sociale : caisse d'assurance
- Organisation Administrative : APC, rectorat, Daïra,...etc.

#### 2.1.2 Qu'est-ce qu'une entreprise?

Il n'existe pas de définition universelle de l'entreprise. Elle dépend de l'approche que l'on choisit. Il existe trois types d'approche :

- L'approche économique
- L'approche juridique

#### Sur le plan économique :

Selon l'approche économique, l'entreprise est une unité de production des biens et services destinés aux marchés de biens de consommation (grand public) et aux marchés de biens de productions (les autres entreprises).

#### Exemples :

- La mission d'une entreprise sidérurgique est de produire de l'acier pour satisfaire les besoins de l'économie nationale : production de biens.

- La mission d'une entreprise de tourisme est de satisfaire les besoins de détente de la population : production de services.

Les moyens sont de trois sortes :

- ➔ humains (cadres, employés, etc.),
- ➔ matériels (terrains, bâtiments, machines, matières premières, etc.),
- ➔ et financiers (banques, etc.).

La mission peut être définie comme la production de biens ou de services nécessaires à la satisfaction des besoins individuels et collectifs.

**Sur le plan juridique** : l'entreprise apparaît comme une entité dont les éléments sont des individus et des personnes morales. Leurs rapports sont régis par des règles de droit.

### **2.1.3 Classification des entreprises**

Les entreprises sont classées selon différents critères. Les plus couramment admis sont le secteur d'activité et la forme juridique.

- Selon l'activité : on distingue les entreprises industrielles, commerciales, prestataires de services, etc.
- Selon la forme juridique : on distingue les entreprises publiques et privées.

## **2.2 Le système d'information dans une entreprise**

Le SI intervient d'une manière fondamentale dans le fonctionnement d'une organisation. On ne peut pas donc le définir ou le décrire sans définir ou décrire ce qu'est une organisation.

Nous allons donc étudier l'entreprise sous son aspect organisationnel et comment l'information circule-t-elle à travers cette organisation.

L'entreprise en tant qu'organisation doit être envisagée selon deux points de vue :

- C'est d'une part un réseau de secteurs d'activités (vente, comptabilité, approvisionnement, etc.) reliés entre eux par des circuits dans lesquels circulent des flux (monétaires, de matière ou d'information).
- C'est d'autre part une organisation à plusieurs niveaux de responsabilité où chaque niveau possède un certain pouvoir de décision (direction générale, direction financière, ..., département, service, section, etc.). Les décisions engendrent des actions. En fonction des résultats obtenus, on entreprend d'autres actions.

En résumé, l'entreprise est vue comme une organisation complexe où diverses fonctions sont effectuées afin d'atteindre certains objectifs :

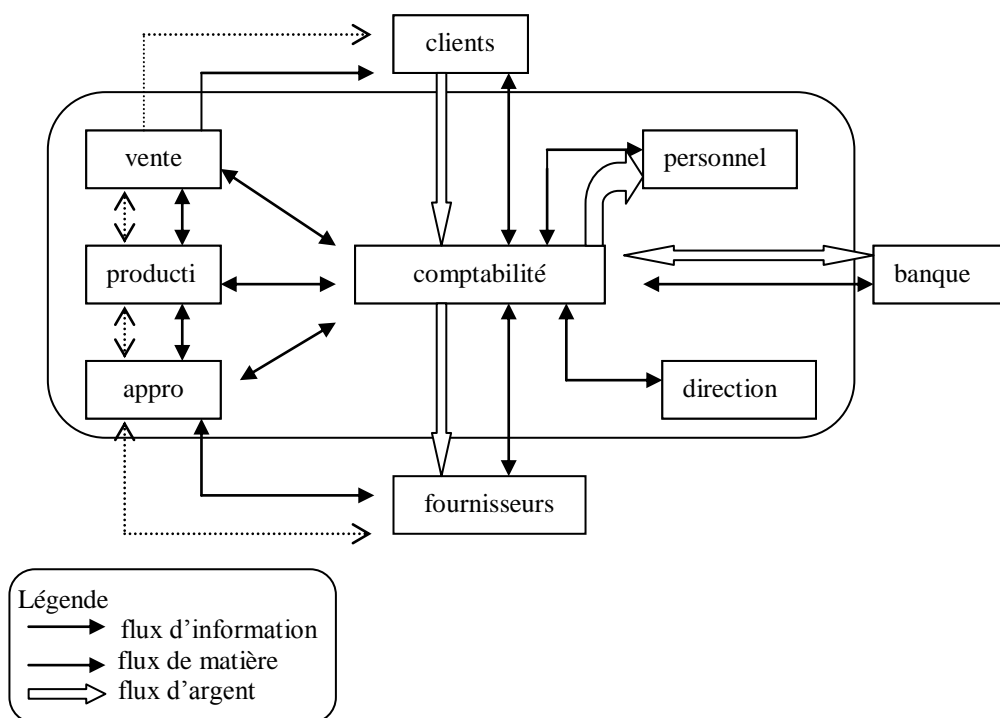
- Ces fonctions sont assumées par différents secteurs d'activité reliés entre eux et échangeant toutes sortes d'information ➔ il s'agit d'étudier les circuits d'information.

- De plus, pour atteindre les objectifs fixés, chaque niveau de responsabilité prend des décisions → il s'agit d'étudier les liens entre décisions et actions : les cycles d'information.

## 2.2.1 Les circuits d'information :

Exemple : Acheter des matières premières pour en faire un produit que l'on vend met en jeu trois grandes fonctions de l'entreprise : l'approvisionnement, la production et la vente. Il existe d'autres fonctions de type administratif, personnel, etc. Cet ensemble de secteurs d'activités n'est pas isolé, l'entreprise est en effet ouverte sur le monde extérieur et entretient des relations constantes avec la banque, ses clients et ses fournisseurs.

On peut représenter schématiquement l'entreprise de la façon suivante :



Les secteurs d'activité sont reliés par des circuits dans lesquels circulent des flux. Ce sont principalement les flux d'information, de matière et d'argent.

Tout type de flux (autre que celui d'information) doit être accompagné d'un flux d'information. Les circuits peuvent être internes ou externes.

A travers ces circuits, l'information joue un rôle prépondérant puisqu'elle est nécessaire à la connaissance de tous les flux qui y circulent ⇒ il est alors indispensable de la normaliser. La normalisation implique généralement :

- une codification de l'information,
- et une standardisation de ses supports.



La normalisation permet de réduire les risques d'erreurs et de faciliter les opérations à effectuer. Néanmoins, il subsiste toujours des erreurs, donc avant de stocker les données dans la base, il faut les contrôler.

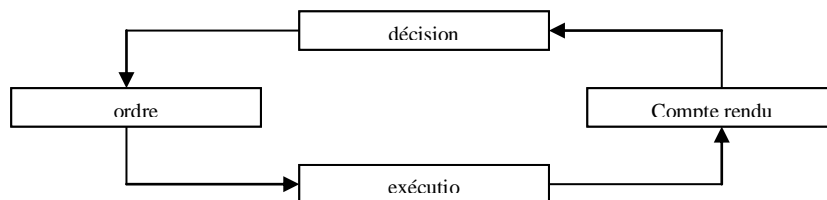
### 2.2.1 Les cycles d'information :

Pour voir le rôle fondamental que joue l'information dans l'organisation, il faut considérer non plus les circuits d'information mais le cycle qui la caractérise et relie la décision à l'action. L'étude de ce nouvel aspect est basée sur les notions de décisions et d'objectifs. Quel est l'objectif à atteindre ? est-il atteint ? autant de questions que l'on se pose à tous les niveaux de responsabilité de l'organisation.

#### Exemple :

Le Directeur Général d'une entreprise décide de lancer une campagne de publicité pour relancer les ventes d'un produit donné. Chargé d'agir en conséquence, le directeur commercial exécute lui-même certaines tâches ou les confie à une firme spécialisée. Au bout de deux mois, il fournit au DG divers éléments d'appréciation sur les premiers résultats de cette action publicitaire. En fonction de ces résultats (par exemple accroissement des ventes), le DG peut décider d'augmenter ou non la part du budget relative à la publicité de ce produit. Le cycle sera complet lorsque d'après les éléments d'appréciation fournis, le DG maintiendra, modifiera ou même annulera sa décision.

Le cycle d'information peut être schématisé de la façon suivante :



## 2.3 Représentation systémique d'une organisation

### 2.3.1 Notion de système

Un système peut être défini comme un ensemble composé d'éléments en interaction permanente, organisé et ouvert sur son environnement auquel il doit s'adapter en permanence pour sa survie. Envisager une entreprise en tant que système consiste à la considérer comme un ensemble organisé, composé de différentes fonctions, services, individus en permanente interaction, ayant tous des objectifs pouvant être contradictoires.

L'entreprise en tant que système est ouverte sur son environnement externe, source de menaces à appréhender mais aussi d'opportunités à saisir. L'entreprise doit s'y adapter en permanence pour sa survie et son développement. Les composantes de l'environnement sont très diversifiées : technologie, social, culturel, juridique, économique, politique, écologique, concurrence, clients, fournisseurs.

### 2.3.2 L'entreprise vue en tant que système

L'examen de l'entreprise selon les deux points de vue : les circuits d'information et les cycles d'information, permet de considérer cette organisation comme un système complexe formé de trois sous-systèmes (fig.:

- le système de décision (ou de pilotage)
- le système opérant
- et le système d'information.

Les deux derniers systèmes coopèrent afin d'atteindre les objectifs tracés par le système de décision. Cette coopération est effectuée en échangeant des informations lesquelles forment la mémoire de l'organisation.

Cette mémoire est constituée d'informations de décision, d'informations générées par le système opérant et d'informations externes provenant de l'environnement du système.

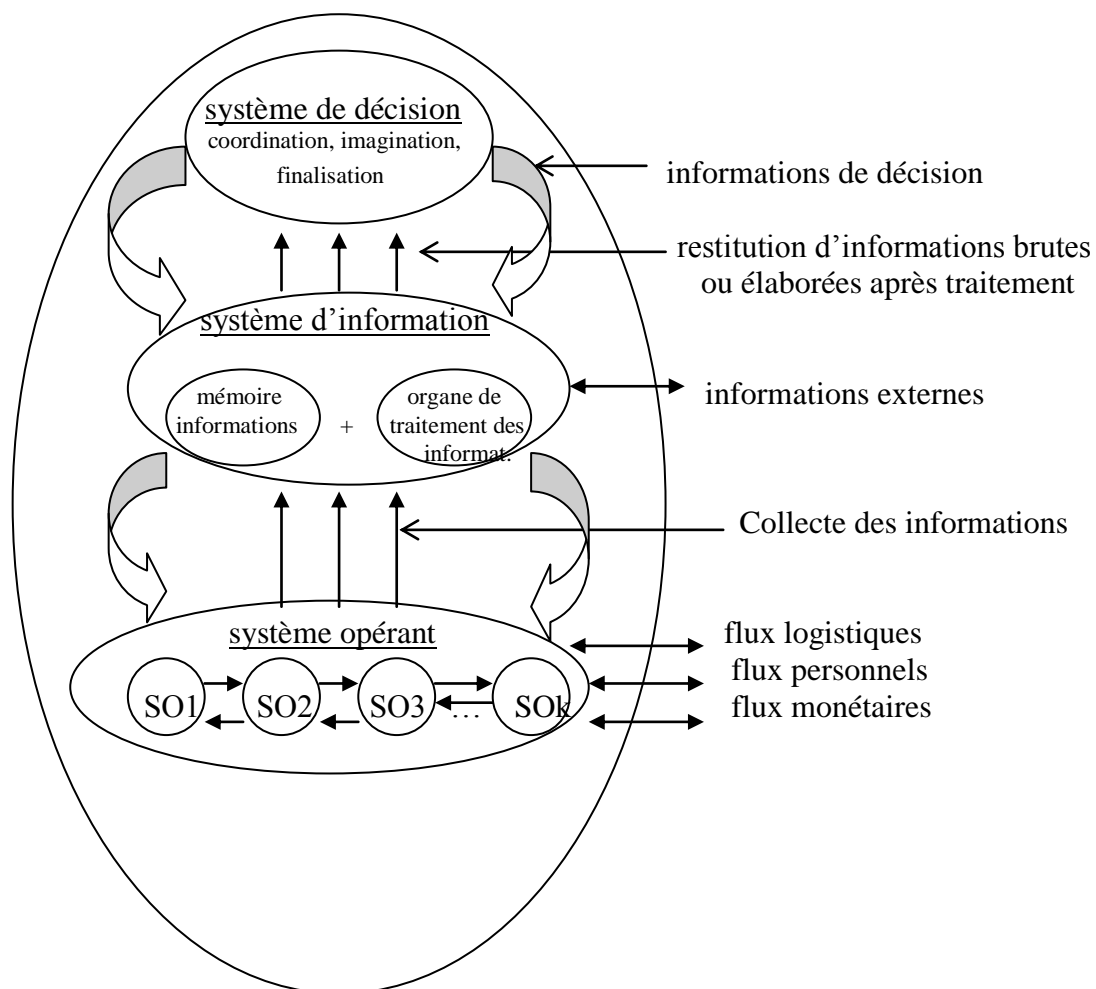


Figure SCHEMA SYSTEMIQUE D'UNE ENTREPRISE

## 2.4 Le système d'information

Le SI a pour mission essentielle de construire puis de garder en mémoire une représentation de l'activité du système opérant au sein de l'environnement afin de la mettre à la disposition des acteurs du système de décision pour qu'ils puissent piloter, coordonner et finaliser le comportement du système opérant.

Le couplage du système de décision avec le système opérant s'effectue essentiellement grâce à l'accès à la mémoire des représentations de l'activité (fig. 3.6).

Les quatre grandes fonctions du système d'information d'une organisation sont :

- générer les informations représentant l'activité du système opérant au sein de son environnement → les informations primaires,
- mémoriser les informations primaires ainsi que sur la demande du système de décision, les informations externes, de décision et élaborées,
- assurer l'accès à la mémoire et la communication des informations,
- traiter à la demande du système de décision certaines informations.

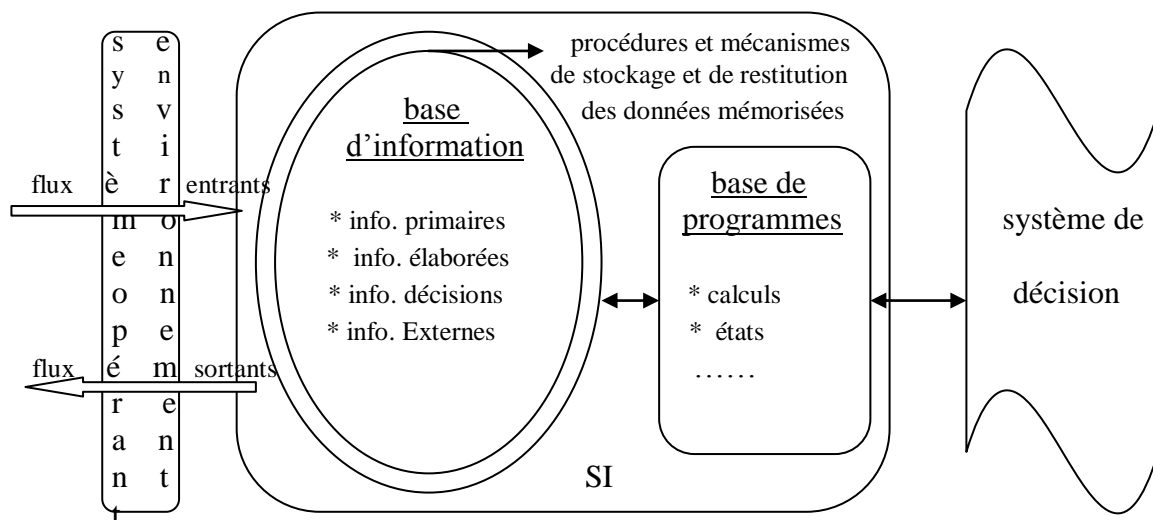


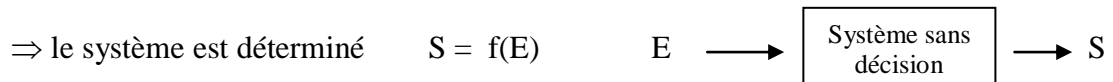
figure 3.6 SCHEMA DU SYSTEME D'INFORMATION

## 2.5 Le système d'information automatisé

### 2.5.1 Actions programmées et décisions

Dans un système, les actions programmées déterminent de manière unique les sorties à partir des entrées.

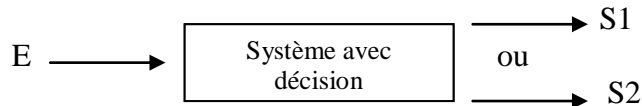
**Exemple 1.** La connaissance des écritures comptables détermine de manière unique la nouvelle balance → il existe une règle unique, formalisable permettant de déduire la nouvelle balance (sortie) de l'ancienne à partir des écritures passées (entrées).



Cependant, un système peut se trouver en situation d'information incomplète

⇒ dans ce cas, une même entrée peut conduire à plusieurs sorties.

Le choix de la sortie se fait par la prise d'une décision.



**Exemple 2.** La connaissance du niveau de stock ne détermine pas les quantités à commander au fournisseur, le service achat devra effectuer un choix.

Des éléments non formalisables (intuition, expérience professionnelle, intérêts personnels, habitudes, etc.) peuvent intervenir dans un choix.

## 2. 5.2 Système d'Information Automatisé

A quelles conditions un système d'information peut être pris en charge par un ordinateur ?

La réponse : le système ne doit comporter que des actions programmées.

Les choix ne sont pas formalisables et donc non automatisables.

**Définition :** un SIA est un sous-système d'un SI dans lequel toutes les transformations significatives d'information sont effectuées par des ordinateurs.

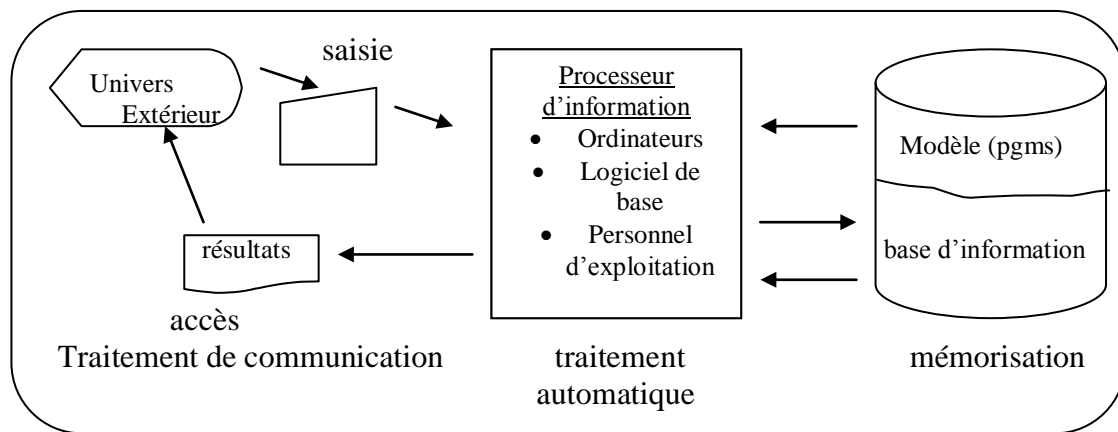
Le SIA permet la conservation et le traitement automatique des informations.

### 2. 5.3 Sous-systèmes fonctionnels du SIA

#### Présentation

Dans un SIA, le processeur d'information est constitué par un ou plusieurs ordinateurs pilotés par le personnel d'exploitation, le logiciel de base, en liaison avec l'univers extérieur au moyen des unités périphériques de communication, des supports de saisie et des personnels de saisie, et en liaison avec le modèle et la base d'information au moyen des unités périphériques de stockage. Le modèle et la base d'information sont stockés sur des mémoires externes.

Le processeur d'information va chercher dans le modèle le ou les programmes à exécuter et les structures de données à respecter et procède à la consultation ou à la mise à jour de la base d'information. Il élabore des résultats qu'il communiquera à l'univers extérieur.



SCHEMA DU SIA

On distingue quatre sous-systèmes fonctionnels:

- deux sous-systèmes internes au SIA :
  - ➔ le traitement automatique
  - ➔ la mémorisation
- deux sous-systèmes interfaces avec l'univers extérieur :
  - ➔ la saisie
  - ➔ l'accès

La mémorisation : fonction de stockage des informations

- ➔ stockage des programmes et de la structure des données ⇒ mémorisation du modèle,
- ➔ stockage des données ⇒ mémorisation de la base d'information.

Le traitement automatique : fonction qui consiste à manipuler des données mémorisées ou provenant de l'extérieur. Il est effectué par l'ordinateur.

Exemples de traitements automatiques : contrôles, mises à jours, recherches, calculs.

La saisie : introduit les entrées externes dans le SIA.

L'accès : transforme les résultats issus d'un traitement automatique en sorties externes vers l'univers extérieur.

## 2.6 Qualités d'un système d'information informatisé

Pour être efficace, le système d'information informatisé devra notamment assurer :

- La rapidité et la facilité d'accès aux informations

Un système trop lent ou trop compliqué à utiliser peut décourager les utilisateurs et diminuer l'efficacité ou la pertinence des décisions

Il faut donc des machines et des réseaux performants et des **interfaces** conviviales et pratiques à utiliser.

- La fiabilité, la pertinence et l'intégrité des informations

Les informations doivent être sûres et fiables, le système doit fournir des informations à jour.

Il est important de noter que cette caractéristique est surtout liée à la promptitude des saisies, donc de l'attitude des humains. Côté machines, le système doit être disponible quand on en a

besoin. Les indispensables opérations de maintenance auront donc lieu de préférence en dehors des heures de travail.

L'intégrité des informations implique que le système sait réagir à des situations qui risquent de rendre les informations incohérentes. Par exemple, si la communication est coupée entre deux ordinateurs qui doivent synchroniser leurs données, le système doit être capable de reconstituer une situation correcte et ce pour les deux ordinateurs.

- la sécurité et la confidentialité des informations.

La sécurité du système est assurée par des dispositifs qui permettent de sauvegarder régulièrement les données. Si le système est **critique**, on utilisera des machines à **tolérance de panne** élevée.

Le système doit également être protégé de la malveillance et des attaques extérieures grâce à des dispositifs matériels (routeurs filtrants) ou logiciels (identification, anti-virus, pare-feu, détecteurs d'intrusion...).

La confidentialité des données est un autre aspect important de la sécurité des systèmes d'information. Elle peut être assurée soit par des moyens matériels (lecteurs de cartes, d'empreintes...) soit par des moyens logiciels (identification, permissions sur des fichiers ou des bases de données...).

## 2.7 Exercice : Circuit d'information.

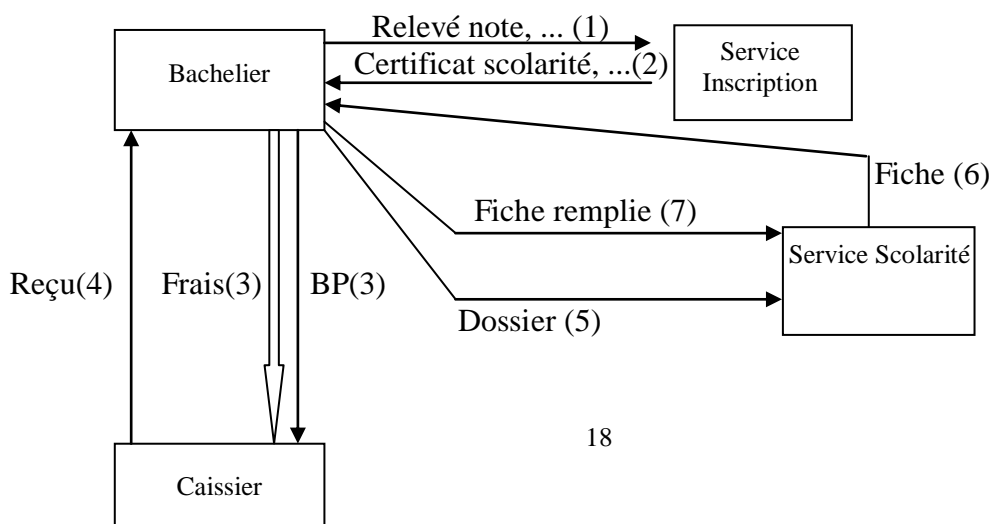
Nous voulons représenter par un circuit d'information le processus d'inscription d'un nouveau bachelier à la Faculté des NTIC.

Le nouveau bachelier doit en premier déposer son relevé de note original, son affectation définitive et les frais d'inscription au niveau du service inscription du vice-rectorat de la pédagogie. Le responsable lui remet un reçu de paiement, un certificat de scolarité et sa carte d'étudiant.

Il se présente ensuite au service scolarité de la Faculté NTIC pour déposer un dossier contenant un acte de naissance, une copie du relevé de notes du BAC, une copie de l'affectation définitive et 2 photos. La secrétaire lui remet une fiche d'information qu'il remplit et la remet à la secrétaire.

Etablir le circuit d'information correspondant à ce problème.

### Solution :



# CHAPITRE 3

## La codification des informations

### Introduction

Durant l'étude d'une application, beaucoup d'informations sont étudiées et définies en vue de leur traitement par ordinateur. L'intérêt essentiel de l'ordinateur est son aptitude à traiter automatiquement de grandes quantités d'information, à grande vitesse et dans d'excellentes conditions de sécurité. Mais cet avantage déterminant ne peut être obtenu que par un travail préalable très important d'organisation de ces informations. **C'est le rôle de la codification**

#### **But :**

La structuration de l'information en vue de son traitement automatique et sa définition de manière claire et précise, pour cela, on associe aux différents objets ou concepts rencontrés dans une application des codes permettant de les caractériser d'une manière unique

#### **Pourquoi codifier les informations**

- Pour être traitées par ordinateur
- Les informations ont besoin d'être structurées
- Association des codes aux différentes informations manipulées par SI
- Désigner une information de manière claire et unique.

**Définition1** : la codification est une opération d'optimisation, elle substitue à une information sous forme naturelle, un symbole conventionnel (code) qui est mieux adapté aux objectifs de l'utilisateur et au traitement automatique.

**Définition2** : un code est un nom abrégé ou une représentation de l'information permettant de désigner un objet ou un concept de manière claire et unique.

### 3.2 Objectifs de la codification

La codification offre plusieurs avantages :

- ➔ identifier sans ambiguïté un individu dans un ensemble ;
- ➔ réaliser des gains de place et de temps ;
- ➔ représenter certaines propriétés d'un objet ;
- ➔ permettre certains contrôles de forme sur l'information.

### 3.3 Qualités attendues d'une codification

→ **Unicité** : le code doit permettre d'isoler un objet sans ambiguïté à l'intérieur dans un ensemble,

→ **Pérennité** : un code doit être utilisé normalement très longtemps,

→ **Souplesse** : possibilité d'extension et d'insertion

-**Extension** : l'ensemble des objets codifiés peut s'accroître sans remettre en cause la codification choisie.

-**Insertion** : les objets nouveaux peuvent s'insérer entre les objets existants sans remettre en cause la codification choisie

→ **Concision** : l'intérêt d'une codification est d'éviter d'avoir à manipuler des informations trop longues, pour être efficace un code doit donc comporter le moins de caractères possibles.

→ **Stabilité** : le code doit être aussi stable que possible, c'est-à-dire qu'on aura pas à changer à chaque fois qu'un nouvel objet à codifier arrive dans le système.

### 3.4 Différents types de codification

Les possibilités de codification sont en principe illimitées. On ne présentera que quelques types de codes très connus.

**3.4.1 La codification séquentielle** : consiste à attribuer des numéros consécutifs aux objets à codifier d'un même ensemble.

Exemple : les employés d'une entreprise sont codifiés selon l'ordre chronologique de leur recrutement: 001, 002,..., 112. Les nouveaux recrutés se verront attribuer les codes 113, 114,...

**Avantages :**

- Simplicité
- Non-ambiguïté
- Possibilité d'extension

**Inconvénients :**

- Non significatif
- Impossibilité d'insertion
- Pas de regroupements possibles

**3.4.2 La codification par tranches** : consiste à diviser l'ensemble d'objets à codifier en plusieurs catégories où chaque catégorie d'objets se verra attribuer une tranche de codes. A l'intérieur des tranches, les codes sont généralement séquentiels.



**Exemple :** codification des produits d'un stock de quincaillerie (sur 3 positions numériques)

Les n° 001 à 088 désignent les vis

Les n° 100 à 285 désignent les écrous /\* de nouveaux écrous peuvent être insérés

Les n° 300 à 357 désignent les boulons entre 285 et 300 \*/

**Avantages :**

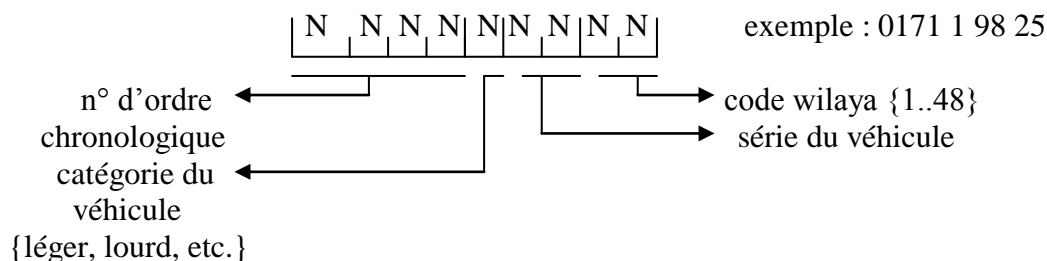
- Non ambiguïté
- Simplicité
- Possibilité d'extension et d'insertion.

**Inconvénients :**

- Non significatif sans table de correspondance.
- Nombre de code dans une tranche difficile à fixer.
- La répartition en catégories n'est pas toujours évidente

**3.4.3 La codification articulée :** les codes sont décomposés en plusieurs zones où chaque zone a une signification particulière.

**Exemple :** codification des numéros d'immatriculation des véhicules,



**Avantages :**

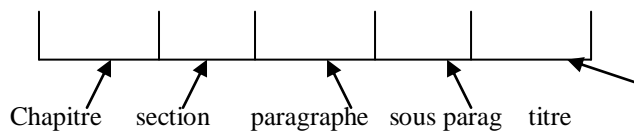
- Non ambiguë
- Possibilité d'insertion et d'extension
- Très répondue et Très utilisée
- Significative à condition d'un choix efficace des descriptions
- Possibilité de regroupement selon un critère donné.

**Inconvénients :**

- Code long et lourd à manipulé
- Risque de saturation
- L'instabilité : le changement d'une caractéristique de l'objet peut remettre en question toute la codification.

**3.4.4 La codification à niveaux (ou hiérarchique) :** utilisée dans le cas où il existe des relations d'inclusion entre les différents ensembles. Le code est découpé en plusieurs zones (niveaux) où chacune représente un ensemble d'objets. Considérés de gauche à droite, ces zones représentent des ensembles de plus en plus restreints. Cette codification peut être considérée comme un cas particulier de la codification articulée.

**Exemple :** contenu d'un livre



**Avantages :**

- Même avantage que la codification articulée
- Facilité de recherche arborescente

**Inconvénients :**

- Même inconvénients que la codification articulée.

**3.4.5 La codification mnémonique :** consiste à abrégé la désignation d'un objet à l'aide d'un ensemble réduit de caractères qui doit être évocateur de l'objet codifié. Il existe 2 sortes :

- des codes dits « consonants », obtenus par suppression des voyelles,  
exemple : FACTURE → FCTR,
- des codes dits « abrégatifs » : obtenus par abréviation de la désignation de l'objet,  
exemple : FACTURE → FACT.

**Avantages :**

- Simple à comprendre
- Possibilité d'extension et d'insertion

**Inconvénients :**

- Risque d'ambiguïté.

## **3.5 Conduite des opérations de codification**

La codification est une étape clé dans la préparation des informations car les erreurs à ce niveau sont souvent très lourdes de conséquences et peu faciles à déceler. Il est alors recommandé de suivre la démarche suivante :

- définir la population à codifier (type individu, nombre individus, leur croissance, maximum, etc.),
- étudier éventuellement les ventilations statistiques
- vérifier d'abord s'il existe des codifications déjà proposées par des organismes spécialisés,
- vérifier la stabilité des propriétés sur lesquelles est basé le code,
- définir la codification avec les personnes qui auront par la suite à l'utiliser,
- mettre en œuvre le code (prévoir des consignes d'utilisation de ce code et le tester avec les utilisateurs.

### 3.6 Exercice.

Le 3<sup>ème</sup> salon de l'Informatique qui vient d'avoir lieu a prévu plusieurs manifestations (Expositions, Conférences, Ateliers, Compétitions).

Les expositions sont de 2 types (externe à l'Université ou interne à l'Université). Les exposants sont au nombre de 13.

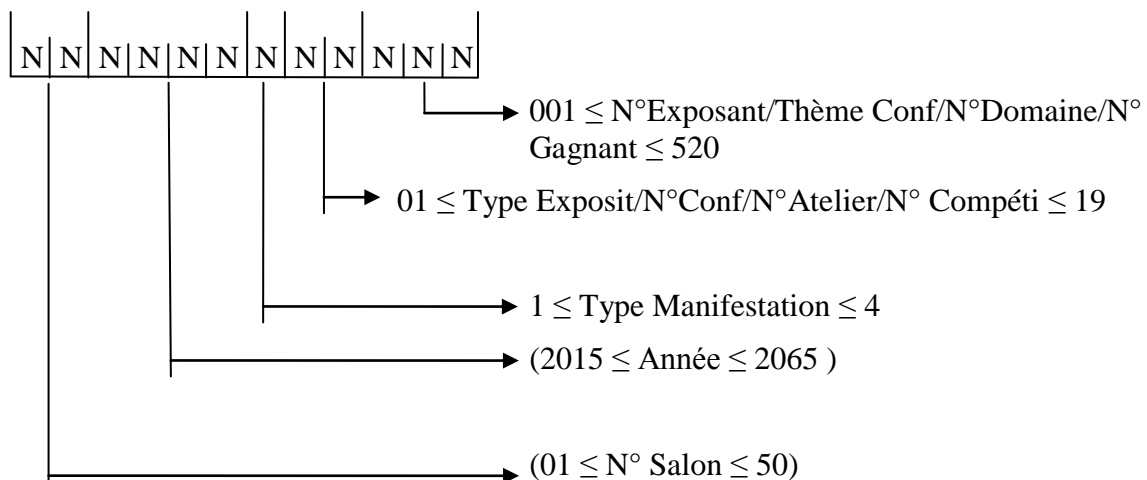
Les conférences sont au nombre de 3, chacune avec son thème.

Les ateliers sont au nombre de 19, chacun dans un domaine. Le nombre maximum de domaines est de 25.

Le salon a prévu 4 compétitions. Il y a 520 participants pour la première compétition, 315 pour la 2<sup>ème</sup>, 52 pour la 3<sup>ème</sup> et 266 pour la 4<sup>ème</sup>. Chaque compétition a un gagnant.

Proposer **une codification** qui permet de connaître toutes les informations sur chaque type d'évènement et sur le salon. Cette codification doit être valable pour tous les salons organisés pendant les 50 prochaines années

#### Solution:



# CHAPITRE 4

## Méthodologie MERISE

### INTRODUCTION

Toute résolution de problème quel qu'il soit, nécessite une, voir plusieurs phases de réflexion plus ou moins longues en fonction de l'ampleur et du type du problème.

Lors d'un projet informatique, cette réflexion doit pouvoir être comprise et reprise par toute personne intervenant sur le projet.

C'est pourquoi, ont été définies des méthodes d'analyse. Certaines disparaissent laissant la place à d'autres méthodes plus adaptées, d'autres évoluent dans le temps en fonction des différentes technologies. Chaque méthode a ses qualités et ses défauts. Il est donc parfois utile et nécessaire en fonction de l'étape d'analyse du projet d'appliquer des méthodes différentes. Chaque méthode est adaptée au type de projet ( objet, industrielle, gestion ) et aux outils ( SGBD, Oracle ...).

*Quelques méthodes :*

- ✓ MERISE, MERISE/2
- ✓ SADT
- ✓ SART
- ✓ OMT
- ✓ UML (bien que UML n'est pas une méthode mais un langage de modélisation unifiée )

### 4.1 Qu'est ce que MERISE :

MERISE est née en 1979 au Centre Technique Informatique du ministère de l'industrie. Ces principaux créateurs sont Hubert Tardieu, Georges Panet et Gérard Vahée . Elle a réellement été introduite dans les entreprises entre 1983 et 1985. Depuis, elle a connu des évolutions en fonction des avancées technologiques avec dernièrement MERISE/2 tournée vers l'objet. Elle reste encore une méthode très utilisée même si UML/OMT est en train d'inverser la tendance.

La méthode Merise admet :

Une vision globale de l'entreprise: mise en place d'un SI, qui est liée à la refonte de l'organisation, et une vision systémique de l'entreprise :

MERISE propose une double approches données-traitements menée en parallèle tout au long du projet ainsi qu'une démarche méthodologique de développement d'un système d'information.

« MERISE » : Méthodes pour Rassembler les Idées Sans Effort.

Méthode d'Etude et de Réalisation Informatique pour les Systèmes d'Entreprise

Un aspect important de Merise consiste à mener l'étude des données et celle des traitements. Ces deux études sont séparées dans un premier temps.

De même, la méthode Merise traite en profondeur l'aspect organisationnel.

Nous nous intéresserons à l'approche MERISE, c'est à dire aux outils de représentation que nous nommerons modèles et qui sont des outils d'analyse (représentation du système existant) et de conception (représentation du système futur).

Chacun de ces modèles correspond à un niveau d'abstraction et se représente selon un formalisme bien défini et traduit un certain choix.

## **4.2 Les niveaux d'abstraction :**

Pour chacun des problèmes de modélisation (données / traitements)

- Procéder de manière progressive....
  - .. du plus stable au plus technique

*Il y a trois niveaux d'abstraction:*

- 1) Le niveau conceptuel
- 2) Le niveau organisationnel
- 3) Le niveau physique

### **4.2.1 Le niveau conceptuel :**

Il consiste à répondre à la question QUOI ?

C'est à dire quoi faire ?, avec quelles données ?

A ce niveau, on ne se préoccupe pas de l'organisation du travail ni du matériel utilisé.

Les deux modèles sont le Modèle conceptuel des données (MCD) et le Modèle conceptuel des traitements (MCT).

### **4.2.2 Le niveau organisationnel :**

Il consiste à répondre à la question QUI?, OU?, QUAND ?

C'est à ce niveau que sont intégrés les critères d'organisation de travail.

On tient compte (ou on propose) des choix d'organisation de travail comme la répartition des traitements entre l'homme et la machine, le mode de fonctionnement (temps réel, temps différé).

Les modèle de représentation sont les modèles logiques des données.(MLD) et physiques des traitements (MOT).

### **4.2.3 Le niveau physique :**

Il consiste à répondre à la question COMMENT ?

On étudie les solutions techniques (mode de stockage pour les données, découpage des programmes pour les traitements(MPD)).

Les modèles étudiés à ce niveau sont les modèles physiques des données ( et physiques des traitements (MPT)).

Chacun de ces trois niveaux ne sont pas indépendants. Il va de soi, que les choix techniques ( niveau physique ) peuvent être imposés dès le début du projet. Les modèles conceptuels et organisationnels seront alors étudiés en tenant compte de ces contraintes. Chacun des modèles seront affinés au cours de la vie du projet.

### Les modèles au niveau Conceptuel

- Le Modèle Conceptuel des Données (MCD)
  - Description des données et des relations en termes de
    - Entité ou Individu
    - Relation ou Association
    - Propriétés ou d'Attributs
- Le Modèle Conceptuel des Traitements (MCT)
  - Description de la partie dynamique du SI en termes de
    - Processus
    - Opérations

### Les modèles aux niveaux Organisationnel

- Le Modèle logique de donnée (MLD)
  - Le modèle «CODASYL» si une orientation base de données réseau est choisie
  - Le modèle «relationnel» si une orientation base de données relationnelle est choisie
  - Le modèle «hiérarchique»
- Le Modèle Organisationnel des Traitement (MOT)
  - Permet de représenter par procédure les phases et les tâches effectuées par chaque poste de travail

### Les Modèles au niveau Physique

- Le Modèle Physique des Données (MPD)
- Spécifie les organisations physiques de données
- Le Modèle Physique des Traitements (MPT)
- Décrit les traitements réalisés pour chaque transaction (temps réel) ou chaque unité de traitement (temps différé)

Le tableau suivant résume les modèles que nous allons aborder tout au long de ce cours :

NIVEAU	DONNEES	TRAITEMENTS	CHOIX PRIX EN COMPTE
<b>Conceptuel</b>	Modèle Conceptuel des Données (MCD)	Modèle Conceptuel des Traitements (MCT)	Choix de gestion Quoi ?
<b>Organisationnel</b>	(MLD) Modèle Logique des Données	Modèle Organisationnel des Traitements (MOT)	Choix d'organisation Qui ?, Où ?, Quand ?
<b>Physique</b>	Modèle Physique des Données (MPD)	Modèle Physique des traitements (MPT)	Choix techniques Comment ?

## 4.2.4 Exemples de niveaux d'abstraction

### – Conceptuel

Le client effectue une demande de service à la compagnie pour assurer son véhicule.  
Cette dernière lui propose un devis

### – Organisationnel

Un client effectue une demande de service à l'agence de son choix, par courrier, pour assurer un véhicule. Un agent de service concerné, si le client est fiable (consultation d'un fichier central inter assurances), prend contact par téléphone pour une visite à domicile (après 17 heures) afin d'examiner plus précisément ses besoins et établir un devis

### – Physique

Le fichier central inter assurances est accessible par internet. Les agences sont connectées au siège de la compagnie par liaison ADSL. Chaque agence dispose de micro ordinateurs de type PC et peut traiter ses données en local grâce au SGBD Access

## 4.3 La démarche préconisée par MERISE :

MERISE préconise un découpage du projet en quatre étapes. Chacune des étapes correspond à des phases d'avancement du projet et donc à une étude bien précise.

Les études effectuées au cours de chacune de ces étapes utilisent les différents niveaux d'abstraction.

Chaque étude doit déboucher sur un dossier qui doit être validé par toutes les personnes concernées par le projet (utilisateur et informaticiens).

*Ces étapes sont au nombre de quatre :*

- 1) Etude préalable
- 2) Etude détaillée
- 3) Réalisation
- 4) Mise en œuvre

### 4.3.1 L'étude préalable :

Cette phase a pour but de bien cerner le système à étudier : QUOI ? Dans quel but ?, Qu'est ce qui existe ? ...et à proposer une première solution ainsi qu'une évaluation tant qualitative que quantitative du projet.

Elle est primordiale et conditionne le bon déroulement des phases suivantes.

La première des actions à effectuer est le recueil des données c'est à dire toute information nécessaire à cerner le projet, à le comprendre et à en effectuer une première ébauche.

Tous ces renseignements sont généralement obtenus au cours d'entretiens. Chaque entretien doit par la suite être consolidé.

L'objectif de la consolidation est de préparer les étapes suivantes en identifiant d'ores et déjà certains concepts ( règles de gestion, de calcul et d'organisation, tâche, domaines, acteurs, processus, flux d'information ... ) mais aussi de constater les points restés obscurs.

Il est généralement utile et nécessaire de formaliser les échanges d'information par un diagramme de flux de données ou par un modèle Flux-acteurs.

Toutes les informations ainsi obtenus constituent le dossier d'étude préalable.

#### 4.3.2 L'étude détaillée:

Basée sur le dossier d'étude préalable elle a pour but de décrire complètement, au plan fonctionnel, la solution à réaliser.

Elle débouche sur un dossier de spécifications détaillées.

#### 4.3.3 La réalisation:

Elle concerne la production du code informatique correspondant aux spécifications définies dans l'étude détaillée. Elle débouche sur un dossier de réalisation.

#### 4.3.4 La phase de mise en œuvre :

Son but est d'exécuter toutes les actions (formation, documentation, installation des matériels, initialisation des données ...) qui permettront aux utilisateurs d'utiliser le logiciel fourni.

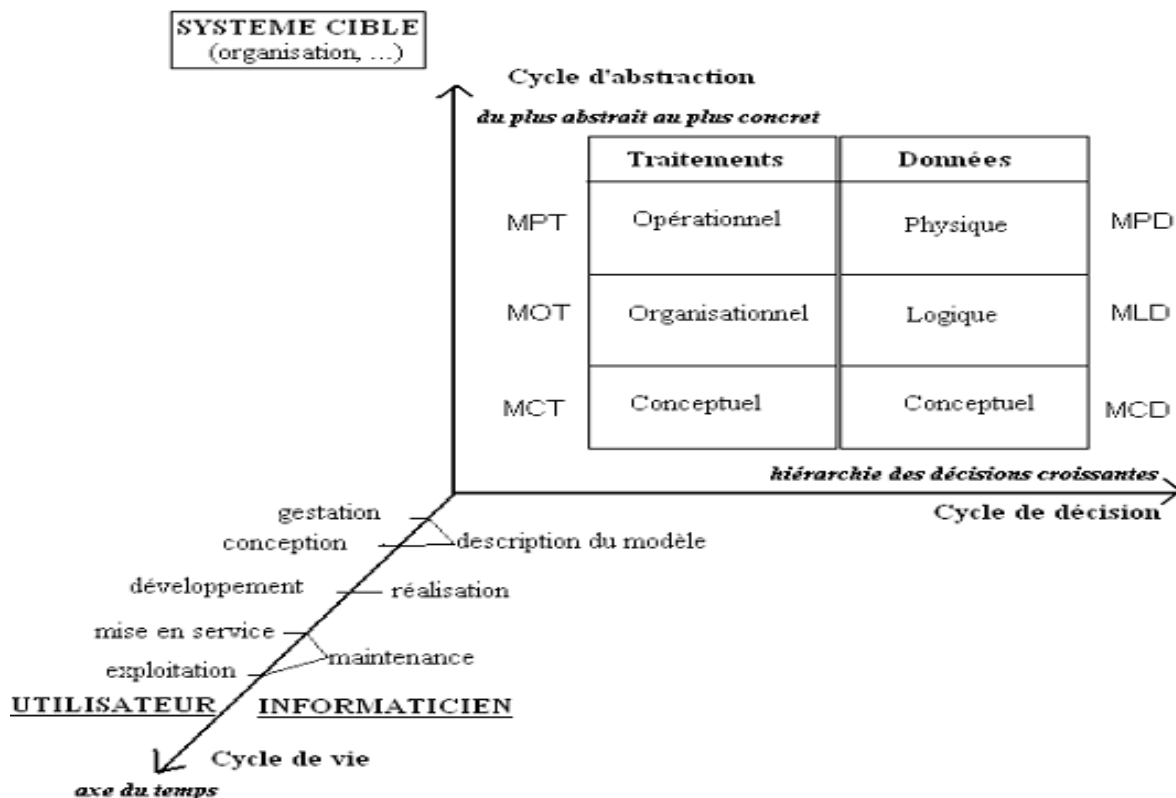
### 4.5 Les trois axes d'un projet informatique :

De l'abstraction à la réalisation d'un Système d'informations, on va devoir observer sous plusieurs angles de vues l'organisation que l'on étudie.

Ces angles de vues sont appelés **cycles**.

MERISE présente dans sa démarche d'analyse trois cycles fondamentaux :

- le cycle d'abstraction,
- le cycle de vie,
- le cycle de décision.





L'axe d'abstraction correspond aux niveaux conceptuels, organisationnels et physiques.  
L'axe vie correspond à l'étape d'avancement de l'étude dans laquelle on se trouve.  
L'axe de correspond aux différents choix de gestion, d'organisation et technique effectués.

A un instant  $t$ , la position sur un axe ne préjuge pas absolument de la position précise sur l'un des deux autres.

En effet, à tout instant, on peut être amené à affiner un modèle conceptuel. De même, les choix techniques peuvent être imposés dès le départ (on ne remet pas en cause le matériel à chaque démarrage d'un nouveau projet).

## **Conclusion**

Même si la méthode MERISE étant, avant tout, une méthode de conception de systèmes d'information, et non de systèmes informatiques, il apparaît aujourd'hui que les systèmes d'information sont largement gérés par des applications informatiques. Les modèles MERISE doivent donc être utilisés pour faciliter le développement de ces applications en s'appuyant sur les technologies logicielles actuelles telles que les bases de données relationnelles et/ou l'architecture client-serveur.

Ce cours s'inscrit dans cette logique : il ne détaillera donc pas les étapes de la méthode Merise dans le processus d'informatisation, mais sera axé sur les formalismes et concepts de Merise utiles aux descriptions statique et dynamique du système d'information à automatiser. Relativement à ces descriptions (encore appelées modèles) la méthode Merise préconise 3 niveaux d'abstraction

# CHAPITRE 6

## Le Modèle Conceptuel des Traitements (MCT)

### Introduction

Contrairement aux modèles de données, qui présentent la partie statique du système, le modèle conceptuel de traitement, ou MCT, en représente la partie dynamique et ceci, sans tenir compte des contraintes liées à l'organisation et des moyens mis en œuvre pour exécuter ces traitements (les choix technologiques ne doivent pas apparaître).

Concrètement, ce modèle décrit les actions qui sont appliquées sur les données dans le but d'obtenir le résultat escompté.

Le MCT représente les activités exercées par le domaine, sa constitution est basée sur la prise en compte des flux échangés entre le domaine et son environnement, de ce fait, le point de départ de l'élaboration du MCT peut être constitué par le modèle des flux.

### 6.1 Le Modèle Conceptuel des Traitements (MCT)

#### 6.1.1 Généralités

Le modèle conceptuel des traitements (MCT) permet de mettre en évidence les liens sémantiques entre les traitements.

Il s'agit de dégager *les actions* que mènent l'entreprise, indépendamment de leur organisation.

⇒ répondre à la question « QUOI »

## 6.1.2 Exemple de MCT

### Exposé du problème

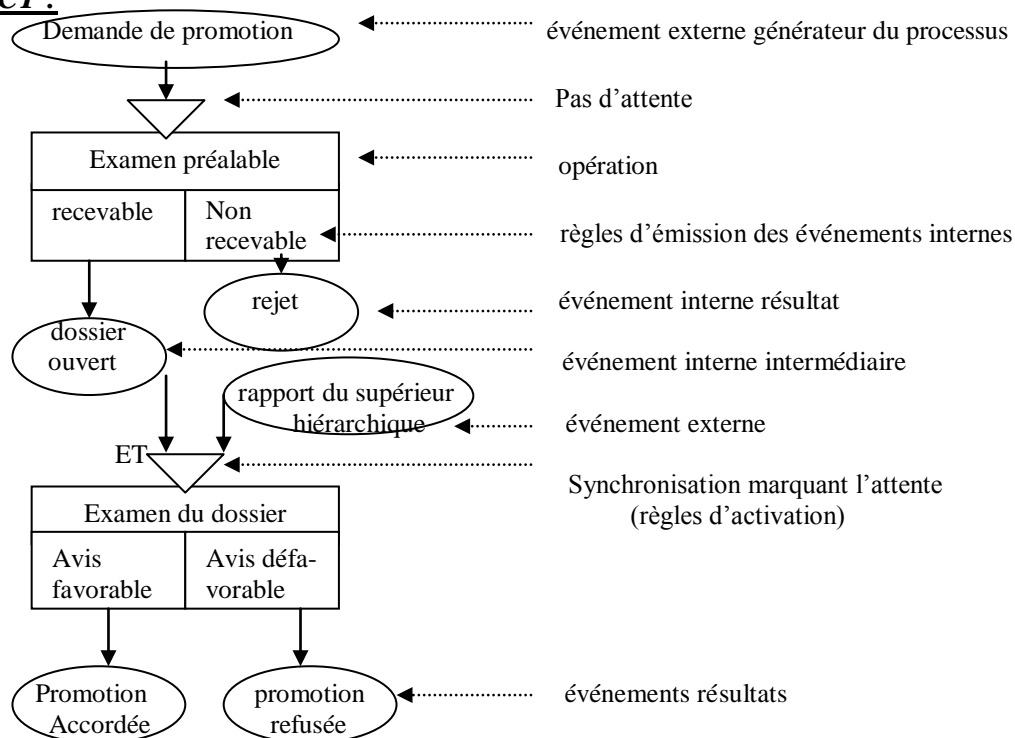
Dans une grande administration, les demandes de promotion sont traitées selon les règles de gestion suivantes :

R.G.1 : toute demande de promotion doit subir un examen préalable permettant de déterminer si elle est recevable ou non ;

R.G.2 : l'examen du dossier d'une demande recevable ne peut se faire qu'après rapport du supérieur hiérarchique ;

R.G.3 : après examen du dossier par l'autorité compétente, la promotion sera accordée ou refusée.

### MCT :



## 6.1.3 Concepts

Les actions sont issues de l'étude de l'existant. Pour chaque action :

- événement qui la déclenche ;
- règles de gestion selon lesquelles elles se déroulent ;
- résultats qu'elle produit.

Les éléments utilisés pour la formalisation d'un MCT sont les suivants :

### l'événement :

Interne ou Externe au Système d'information il s'agit d'un déclencheur pour le lancement d'une opération ou le résultat d'une opération à destination du monde extérieur

Un événement externe → provoque une réaction du SI sous la forme d'une opération.

Un événement interne → provoque une nouvelle réaction du SI ou constitue un résultat pour l'univers extérieur.

Un événement peut être porteur de propriétés.

### ***l'opération :***

Liste des actions à réaliser si la synchronisation associée est réalisée. L'ensemble des actions de l'opération s'exécute sans interruption ni attente d'événement.

Une opération produit en sortie de nouveaux événements.

### ***La synchronisation :***

Règle indiquant les événements et l'enchaînement de ces derniers nécessaires au lancement d'une opération. Il s'agit d'une expression logique composée essentiellement de OU et de ET

La synchronisation de la première opération peut se faire sans attente dès l'apparition d'un événement. Mais la synchronisation de toute opération suivante doit correspondre à une attente.

### ***Emission :***

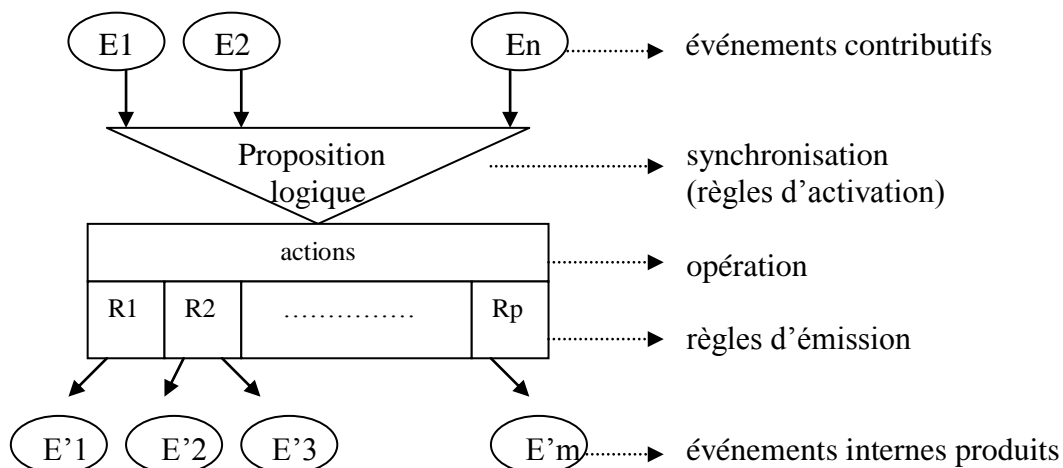
Expression logique indiquant selon le résultat de l'opération quels événements internes au Système d'Information sont créés

### ***Processus***

Le MCT est décomposé généralement en processus.

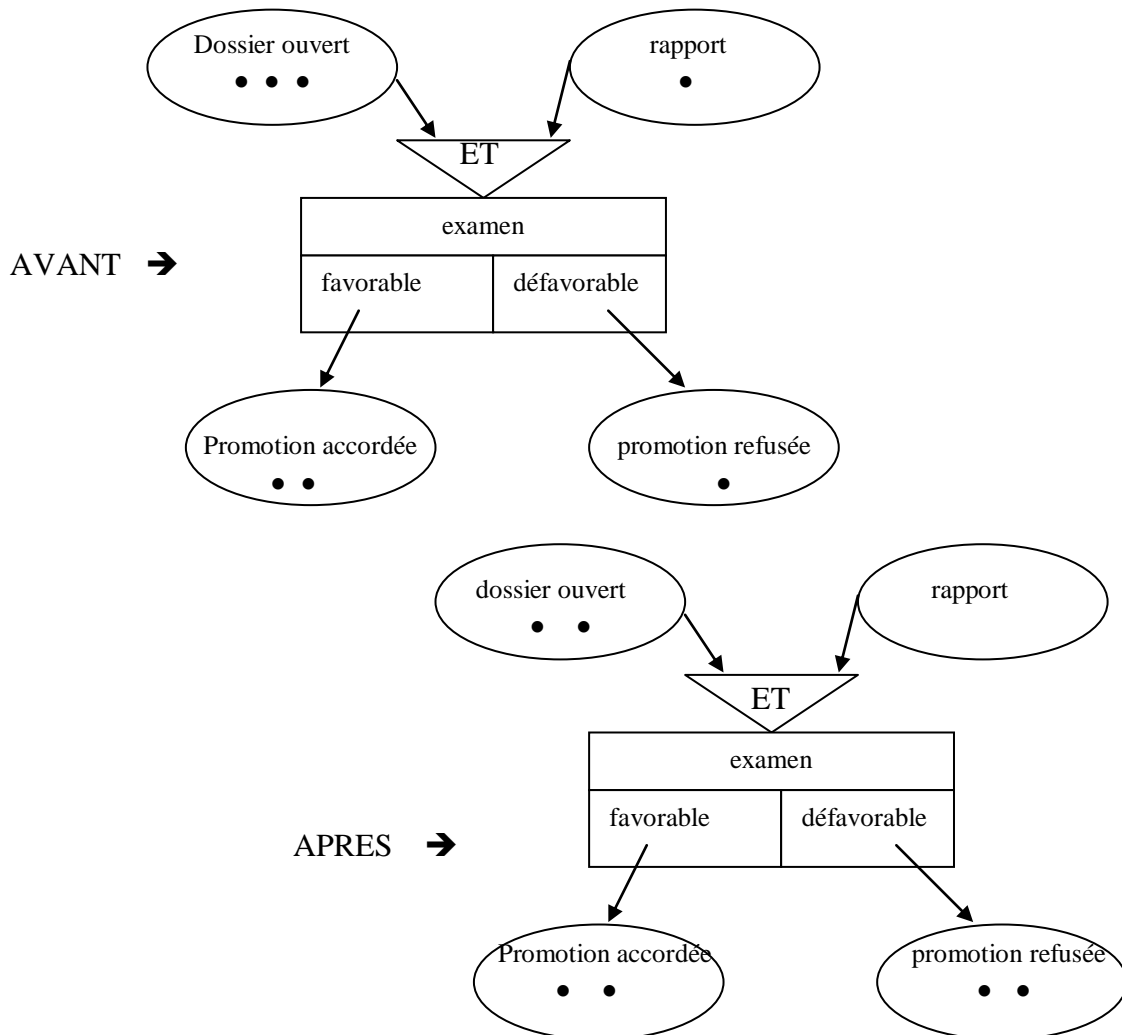
Un processus est un enchaînement d'opérations incluses dans un même domaine d'activité.

### ***Schéma de fonctionnement :***



### Consommation des occurrences des événements contributifs :

Chaque occurrence d'un événement contributif qui active la synchronisation est consommée.



Sur ce schéma, les occurrences d'événements sont matérialisées par des jetons. Des jetons sont consommés, et d'autres sont produits.

### **6.1.4 Construction du MCT**

#### Présentation :

Soit 3 processus liés: traitement des commandes clients, tenus de stock et approvisionnement. Le processus considéré est le traitement des commandes clients.

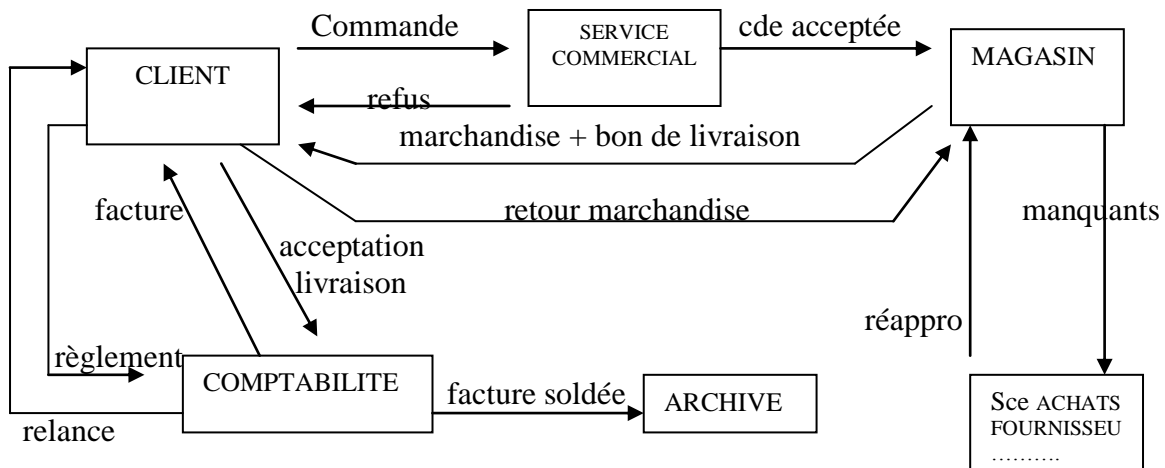
#### Règles de gestion :

- R.G.1 toute commande de client non solvable est refusée ;
- R.G.2 les commandes non disponibles sont mises en attente et devront déclencher un réapprovisionnement par le fournisseur ;
- R.G.3 les commandes en attente seront déclarées disponibles lorsque le réapprovisionnement sera suffisant ;

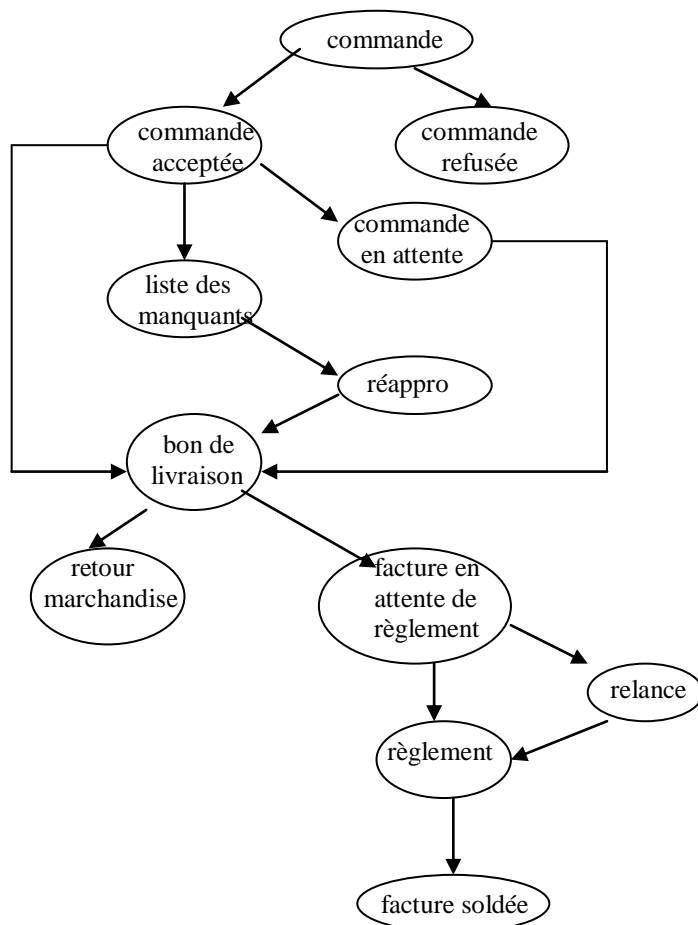
- R.G.4 les commandes disponibles donnent lieu à la livraison au client ;  
 R.G.5 les livraisons refusées par le client donnent lieu à un retour de marchandise ;  
 R.G.6 les livraisons acceptées donnent lieu à des factures qui sont conservées jusqu'à complet règlement ;  
 R.G.7 toute facture non réglée à l'échéance donne lieu à une relance.

### Graphe Acteurs/Flux (schéma de circulation)

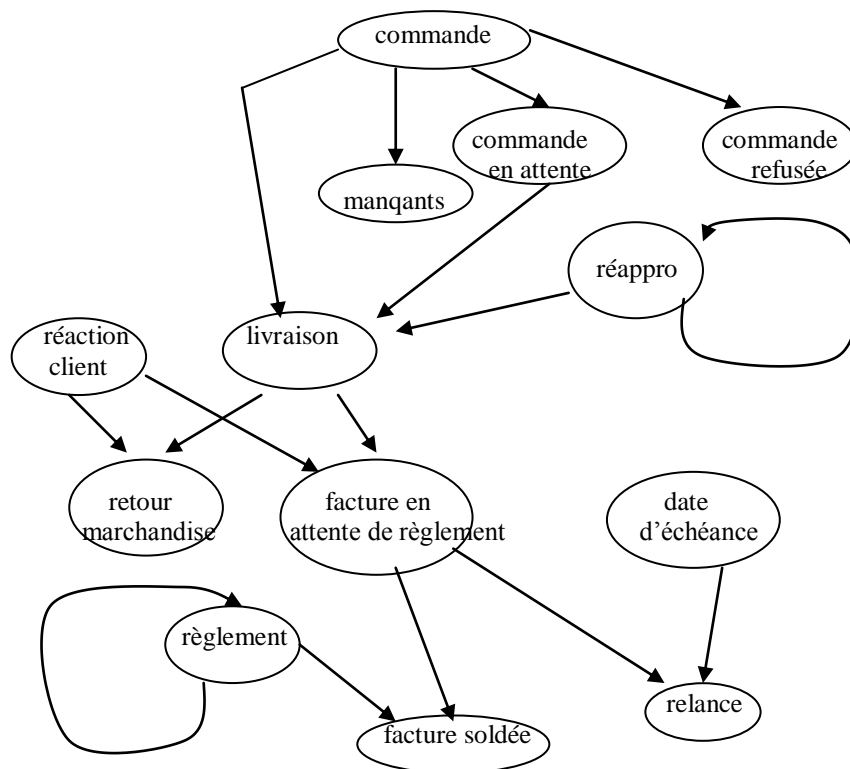
Le schéma de circulation est obtenu à partir des interviews des postes de travail actuels et de l'étude des circuits existants.



### graphe d'ordonnancement des flux

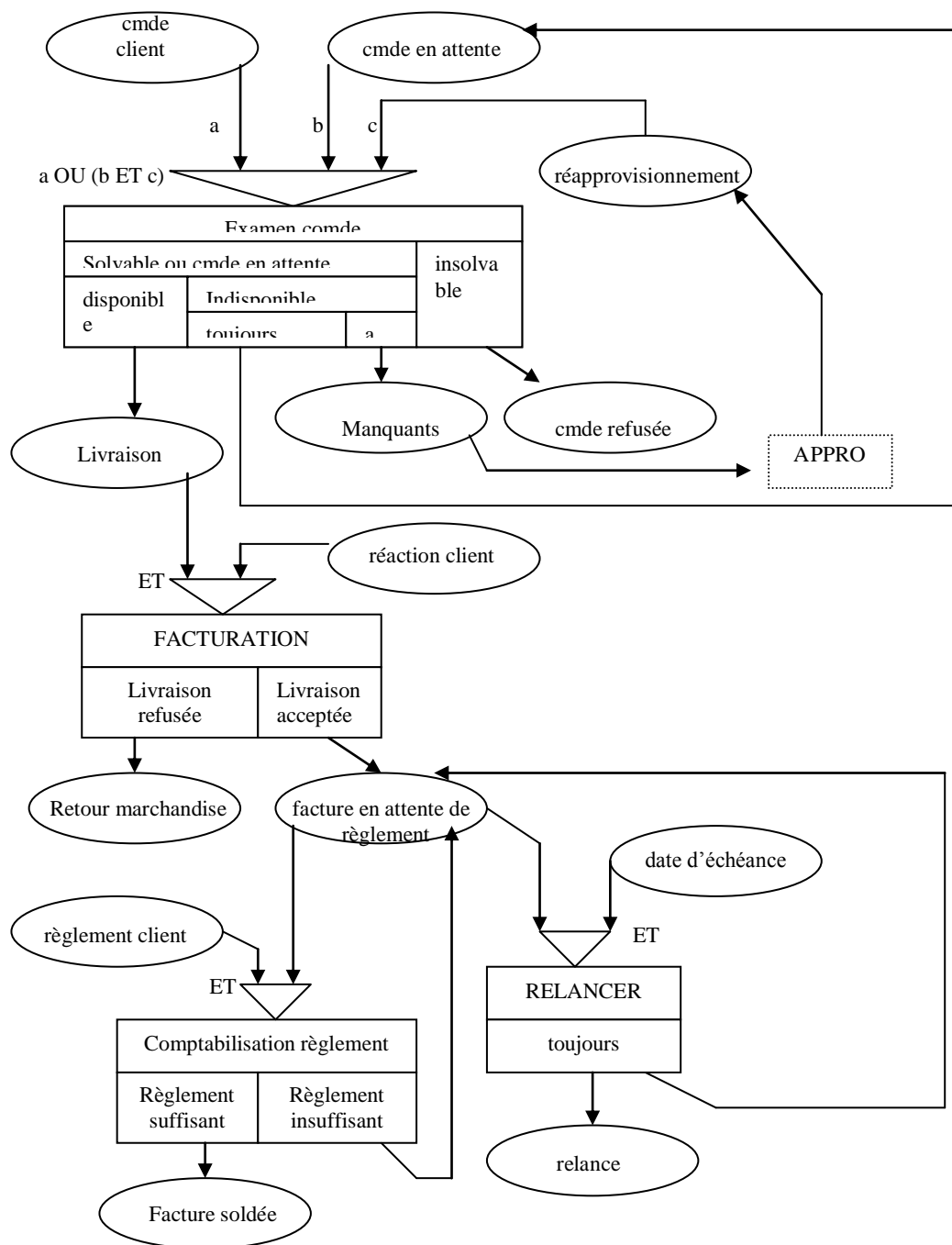


- Des événements générateurs d'attente conceptuelle sont à ajouter :
  - L'événement « réaction client » qui détermine si la livraison est acceptée ou non ;
  - L'événement « date d'échéance » pour l'envoi d'une relance.
- Pas d'attente conceptuelle entre les événements :
  - Commande acceptée et manquants ; commande acceptée et livraison
    - ⇒ liée à l'organisation (le temps que le magasinier consulte ses rayons ..)
    - ⇒ l'événement commande acceptée doit être éliminée du graphe.
- Eviter des expressions telles que « bon de livraison » et « liste des manquants » qui font penser plutôt à l'organisation.



MCT définitif

→ schéma de fonctionnement





### **6.1.5 Règles de vérification d'un MCT**

-Une opération est une suite de traitements qui ne peut pas être interrompue, de ce fait, un événement externe intervenant en cours de traitement ne peut pas influencer sur les conditions de déroulement de celui-ci. Si tel est le cas, l'opération doit être scindée en deux opérations distinctes.

-A l'intérieur d'une même opération, il ne peut pas y avoir de conditions influant sur le déroulement des séquences du processus en cours.

-Les opérations mises en œuvre dans un processus ne doivent pas être redondantes.

-On ne doit pas introduire une synchronisation OU entre événement interne au processus avec un événement externe.

-Lorsque le domaine d'activité comprend un grand nombre d'opérations, il est conseillé de découper le MCT globale en plus petits processus. Quand deux opérations ne s'enchaînent pas, il y a découpage en deux processus différents.

## 6.2 Exercices

### Exercices 1

Nous voudrions modéliser le processus de transfert de fonds dans une banque.

Pour demander un transfert de fond de son compte vers le compte d'une autre personne, le client doit se déplacer jusqu'à la banque.

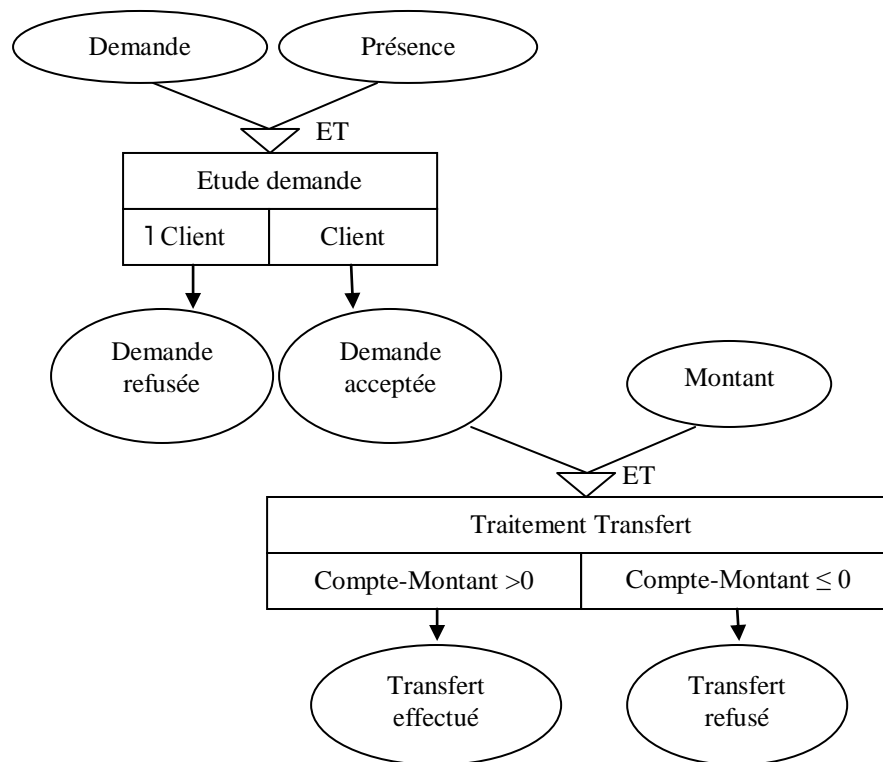
Le caissier doit vérifier que ce client possède bien un compte dans cette banque, sinon la demande est rejetée.

Si le client possède un compte, le gestionnaire doit connaître le montant de transfert.

Si le montant relatif au compte de ce client diminué du montant à transféré est positif, le gestionnaire des comptes clients effectue le transfert, sinon le transfert est refusé.

Il vous est demandé d'établir le MCT de cette description.

### Solution



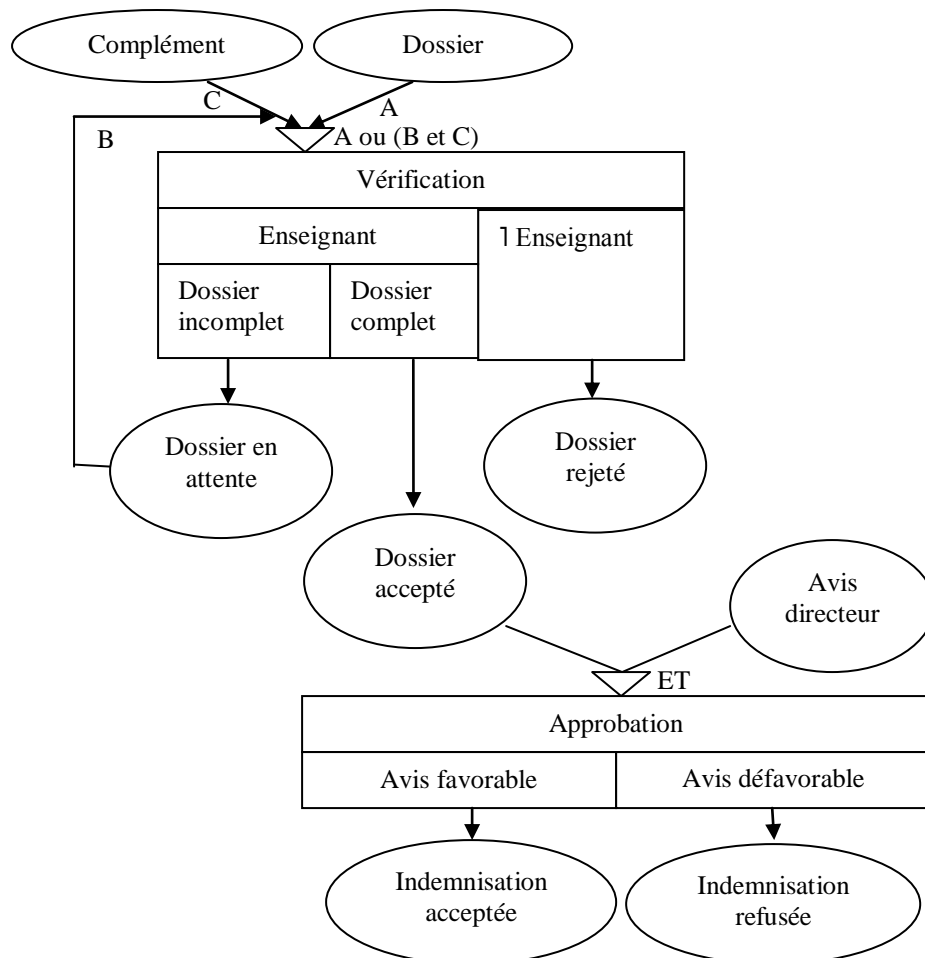
### Exercice2

Les enseignants de l'université peuvent bénéficier de plusieurs indemnités des œuvres sociales (mariages, décès, circoncision, etc..).

- Tout enseignant qui voudra bénéficier d'une indemnité, doit déposer un dossier au niveau du secrétariat des œuvres sociales.
- La secrétaire doit vérifier que le demandeur est bien un enseignant de cette université et que le dossier remis est complet.
- Si le dossier est incomplet, il reste en attente jusqu'à ce que l'enseignant en question ramène les documents manquants.
- Tout dossier complet est envoyé au directeur des œuvres sociales pour approbation.
- Si le directeur a donné un avis favorable, le secrétaire indemnise l'enseignant, sinon il ne le sera pas.

Afin d'aider la secrétaire à faire le suivi de ce dossiers, construire le MCT (Schéma de fonctionnement) qui va permettre l'automatisation de ce processus.

### Solution



# CHAPITRE 7

## Le Modèle Logique des de Données (MLD)

### Introduction

La description conceptuelle a permis de représenter le plus fidèlement possible les réalités de l'univers à informatiser. Mais cette représentation ne peut pas être directement manipulée et acceptée par un système informatique, car le MCD est une représentation des données dans un formalisme compris par les concepteurs et pas par la machine. Il est donc nécessaire de passer du niveau conceptuel à second un niveau plus proche des capacités des systèmes informatiques. Ce niveau, appelé niveau logique, consiste à choisir l'un des trois modèles suivants :

- modèle hiérarchique
- modèle réseau,
- ou modèle relationnel

Chacun de ces modèles repose sur des techniques d'organisation des données particulières que des logiciels seront capables de gérer, dans ce chapitre l'accent sera mis sur le modèle relationnel.

### 7.1 Problématique :

- Définir l'organisation logique des traitements à partir du MCD validé
- Optimiser cette organisation propre aux besoins des traitements.

*Choix du formalisme individuel : clarté, richesse, etc.*

Mais ne peut être supporté par la machine  $\Rightarrow$  choix d'un modèle de données

Exemple : le modèle relationnel (en fonction des choix techniques).

## 7.2 Les concepts relationnels

**Domaines** : ensemble de valeurs. Chaque domaine est caractérisé par un nom. Il est défini soit en extension par la liste de ses éléments, soit en compréhension par l'énoncé d'une condition nécessaire et suffisante d'appartenance.

**Exemple** :

Le domaine « nom » : ensemble de mots (suite de lettres)

Le domaine « sexe » : {masculin, féminin}.

**Relations** : sous-ensemble d'un produit cartésien de domaines, muni d'un nom. C'est donc un ensemble de tuples pour lesquels on indique par le nom de la relation, la cause de leur existence.

**Exemple** : la relation « ETUDIANT » est un sous-ensemble de :

« nom » x « prénom » x « no-carte » x « sexe ».

**Remarque** : la relation du formalisme individuel est différente de la relation du modèle relationnel.

**Attributs** : sous-ensemble d'un domaine caractérisé par un nom. Il correspond à la donnée élémentaire (c'est la colonne d'un tableau).

**Exemple** : l'attribut « nom » de la relation « ETUDIANT ».

**Schéma d'une relation** : ensemble constitué du nom de la relation et de tous les couples (attribut, domaine) sur lesquels elle est définie.

Exemple : le schéma de la relation ETUDIANT est :

{ETUDIANT, (nom, « Zarour, Zitouni, .. »), (prénom, « Karim, Samir, .. »), ..., (sexe, « M,F »)}.

**Clé de la relation** : ensemble minimum d'attributs dont la connaissance des valeurs identifie un tuple de façon unique.

## 7.3 Règles de passage du MCD au MLD

Ces règles sont de type algorithmique et peuvent donc être mises en œuvre par des outils de génie logiciel. La traduction des concepts de base du modèle conceptuel est régie par les trois règles suivantes :

**Propriété** : une propriété du modèle individuel devient un attribut du modèle relationnel. L'identifiant de l'entité devient clé primaire de la relation.

**Individu** : un individu du modèle interne devient une relation (minimum 3<sup>ème</sup> forme normale) du modèle relationnel. L'identifiant de l'individu devient la clé primaire de la relation correspondante.

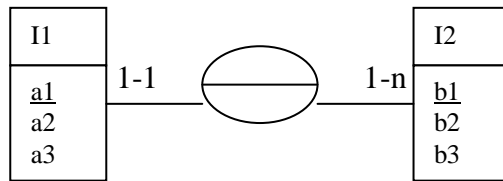
**Relation** :

*Sans propriétés propres*

:

- Si elle est binaire fonctionnelle  $\implies$  la relation-type disparaît

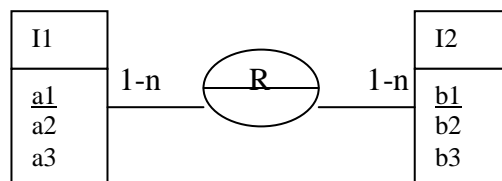
Exemple :



on obtient le modèle relationnel suivant :  $I1(\underline{a1}, a2, a3, b1)$  ;  $I2(\underline{b1}, b2, b3)$

- Si elle est binaire non-fonctionnelle  $\implies$  la relation-type du MI devient une relation relationnelle.

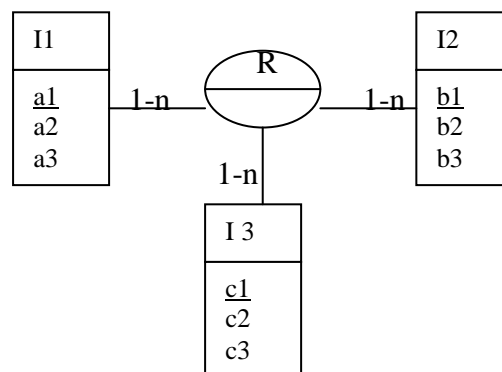
Exemple :



On obtient le modèle suivant relationnel suivant :  
 $I1(\underline{a1}, a2, a3)$  ;  $I2(\underline{b1}, b2, b3)$  ;  $R(\underline{a1}, \underline{b1})$

- Si elle est n-aire  $\implies$  elle devient une relation relationnelle

Exemple :



On obtient le modèle relationnel suivant :

$I1(\underline{a1}, a2, a3)$  ;  $I2(\underline{b1}, b2, b3)$  ;  $I3(\underline{c1}, c2, c3)$  ;  $R(\underline{a1}, \underline{b1}, \underline{c1})$

*Avec propriétés propres* : la relation-type est équivalente à une relation relationnelle, en plus de l'exemple précédent, les propriétés propres deviennent des attributs de la relation.

Remarque : toutes les relations obtenues dans le modèle relationnel sont minimum en 3<sup>ème</sup> FN.

## 7.4 Les formes normales :

### *Première forme normale (1FN) :*

Une entité ou une association est dite en 1FN si elle possède un **identifiant** et si toutes ses propriétés sont atomiques

### *Deuxième forme normale (2FN) :*

Une association (ou entité) est en 2FN si elle est 1FN et si toutes les DF entre ses propriétés sont complètes.

En d'autres termes, on ne doit pas trouver une dépendance fonctionnelle entre une propriété et une partie de l'identifiant.

Cette règle concerne donc :

- 1) toutes les associations porteuses de propriétés
- 2) toutes les entités ayant un identifiant composé d'un groupe de propriétés

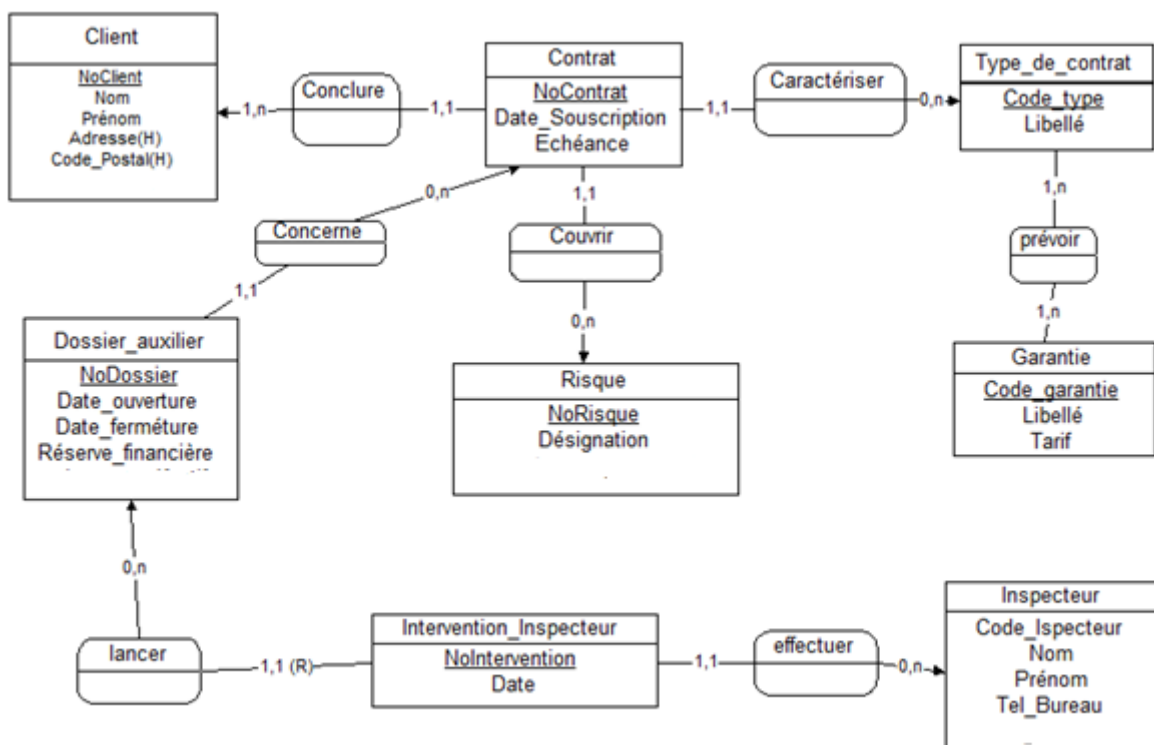
### *Troisième forme normale :*

Un entité ou une association est en 3FN si elle est en 2FN et 1FN et si toutes les DF entre ses propriétés sont directes.

En d'autres termes les dépendances fonctionnelles transitives ne sont pas autorisées.

## 7.5 Exercices

Voici un MCD qui représente de façon très simplifiée la gestion d'une compagnie d'assurances. Transformez le MCD en MLD en respectant toutes les règles de passage MCD à MLD.



**Le MLD correspondant est le suivant**

**Client** (N°client, nom, prenom, adresse, code\_postale)

**Contrat** (N°Contrat, Date\_Souscription, Echéance, # N°client, # NoRisque, # Code\_type)

**Type\_de\_contrat** (Code\_type, Libellé)

**Dossier\_auxillier** (N°Dossier, Date\_ouverture, Date\_fermeture, réserve\_financière,  
#N°Contrat)

**Risque**(NoRisque, Designation)

**Garanti** (Code\_garantie, Libellé, tarif)

**Intervention\_Inspecteur** (NoIntervention, Date, #N°Dossier, # Code\_Inspecteur)

**Inspecteur** (Code\_Inspecteur, Nom, Prénom, Tel\_Bureau)

**Prevoir** (Code\_type, Code\_garantie)



# CHAPITRE 5

## Le Modèle Conceptuel des Données (MCD)

### Introduction

L'objectif d'une modélisation des données, est d'atteindre une Base de Donnée physique (*modèle interne*) contenant des informations relatives à l'activité de l'organisation.

Les données que l'on aura à utiliser au niveau de l'organisation constituent *le réel perçu*.

Le passage du réel perçu au modèle interne se fait au moyen d'un niveau intermédiaire : *le niveau conceptuel*.

Le modèle conceptuel des données (MCD) a pour but de représenter de façon structurée les données qui seront utilisées par le système d'information. Le modèle conceptuel des données décrit la sémantique c'est à dire le sens attaché à ces données et à leurs rapports et non à l'utilisation qui peut en être faite.

On établit le MCD après avoir recensé et donné un nom à l'ensemble des données du domaine étudié. Ensuite on étudie les relations existantes entre ces données (les dépendances fonctionnelles), pour aboutir au MCD.

**N.B :** La formalisation des données est le point majeur de la méthode Merise. Il est important de bien saisir ce chapitre.

### 5.1 Présentation des concepts et du formalisme du MCD

#### 5.1.1 Concepts de base :

Ensemble de concepts et de règles qui permettent d'exprimer le modèle conceptuel de données (MCD) et les modèles externes.

1. **Propriété :** particule élémentaire d'information appelée aussi caractéristique ou attribut. Elle décrit un individu ou une relation. Les propriétés: sont les informations de base qui décrivent les éléments(les entités) d'un SI.

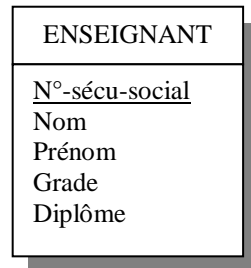
**Exemple:** numéro client, nom clients, prénom client, adresse client sont des propriétés qui décrivent l'élément(l'entité) Client.

Chaque propriété dispose d'un type (alphabétique, alphanumérique, numérique, date, logique,...).

2. **Entité** : Une entité représente un objet du SI (acteur, document, concept, ...), ou plus exactement un ensemble d'objets ayant les mêmes caractéristiques.

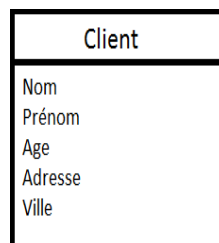
Dans une entité, on met les informations nécessaires et suffisantes pour caractériser cette entité. Ces informations sont appelées propriétés. Les propriétés sont collectées lors de l'établissement du dictionnaire des données (voire partie suivante). Les propriétés prennent des valeurs pour chaque occurrence d'une entité. On représente une entité par un rectangle dans lequel on place le nom de l'entité, son identifiant (souligné) et la liste des propriétés.

**Exemple :**



3. **Occurrence** : une occurrence d'un individu-type est un membre de la famille d'objets « individu-type ».

Chaque occurrence doit être distinguable au sein de sa famille. Toutes les occurrences doivent être dotées des mêmes propriétés sur toute l'étendue du réel.



**Exemples d'occurrences d'une entité: Client**

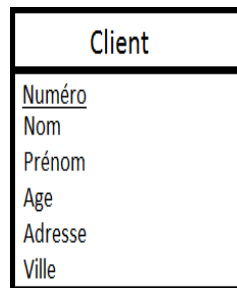
Client	Client	Client
Nom : Amin	Nom : Mohamed	Nom : Tarik
Prénom : Jamil	Prénom : Salim	Prénom : Tribek
Age : 33	Age : 45	Age : 26
Adresse : Rue Farah	Adresse : Rue M5	Adresse : Rue Maarif
Ville : Nador	Ville : Oujda	Ville : Casa

4. **Identifiant** : Une propriété particulière, appelée identifiant, permet de distinguer sans ambiguïté toutes les occurrences de l'entité. L'identifiant est toujours souligné. L'identifiant est une propriété qui ne peut pas changer au cours du temps pour une occurrence. L'identifiant est une propriété qui permet de connaître de façon unique et sûre les occurrences d'une entité donnée

- **Question:** dans les occurrences précédentes de l'entité Client, est-ce qu'il est possible d'avoir 2 occurrences avec le même nom et prénom? C'est-à-dire, avoir 2 clients différents mais avec le même nom et prénom!
- **La réponse : est oui !!!**

Dans la question de l'exemple précédent, nous allons ajouter une propriété au début de l'entité qui s'appelle numéro client (ce numéro doit être unique) pour faire la différence entre deux occurrences qui peuvent avoir le même nom et prénom

- Exemple d'une entité avec identifiant: **Numéro**

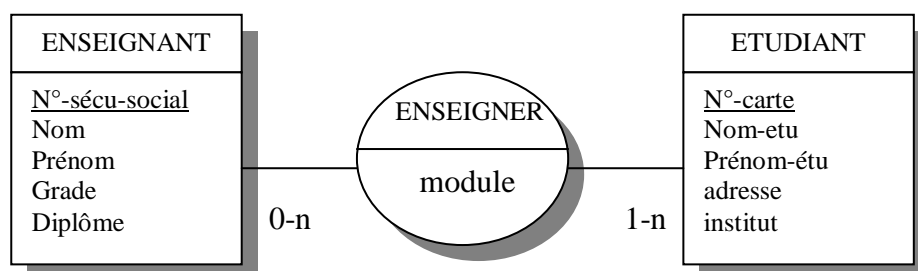


## 5. Notion de relation

Une relation représente une possibilité de "lien" entre deux ou plusieurs occurrences d'individus. Pour une relation donnée, les individus sont fixés. La relation est entre les individus et a besoin d'individus pour exister. Elle est représentée par une ellipse reliée par des traits aux rectangles représentant les individus.

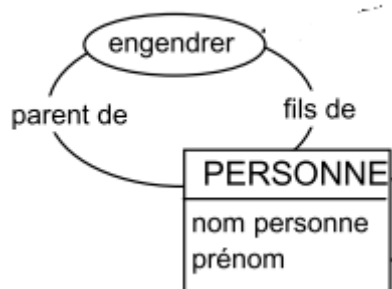
- **Relation-type** : c'est une association
  - de deux ou plusieurs individu-types
  - qui n'a pas d'existence propre
  - chacune de ses occurrences est distinguable
  - toutes les occurrences d'une relation-type sont caractérisées par la même liste des propriétés.
- **Collection** : d'une relation-type est la liste des individus-type que cette relation-type associe.
- **Dimension** : d'une relation-type est le nombre d'individus-type de sa collection.
- **Entité** : d'une relation-type est la liste de son identifiant et de ses propriétés propres.
- **Occurrence** : d'une relation-type composée d'une seule occurrence de chacun des individus-type de sa collection.

Exemple :

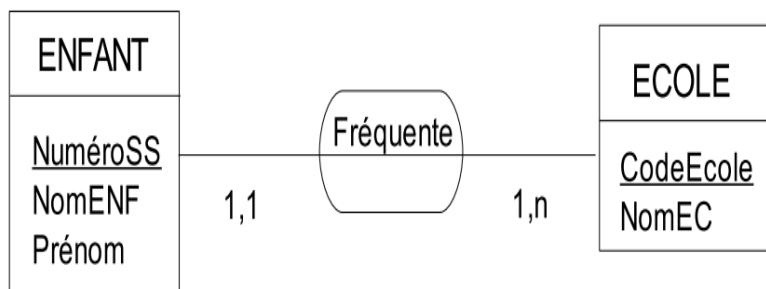


## Types de relation:

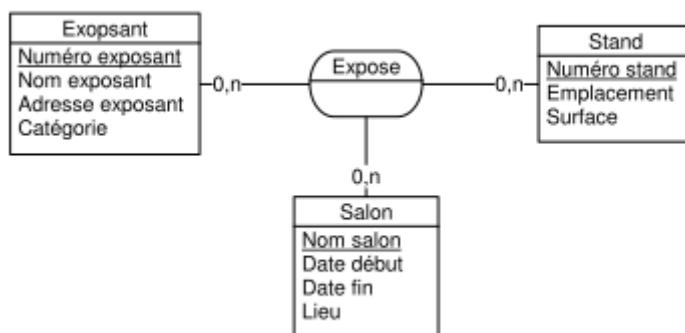
### a. Relation binaire réflexive



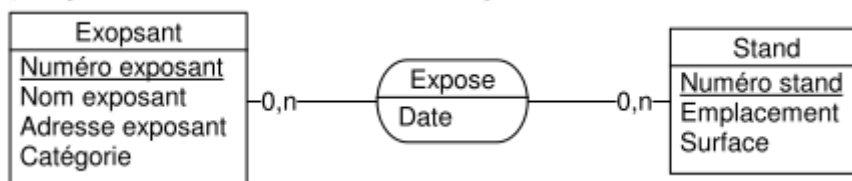
### b. Relation binaire



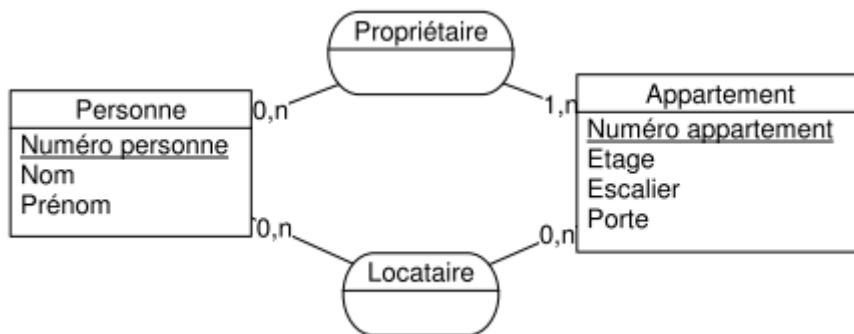
### c. Relation n-aire



## Une association type peut avoir des propriétés



**Il peut y avoir plusieurs associations type liant les m<sup>^</sup>emes entités si la s<sup>^</sup>emantique est différente**



## 6. Cardinalités

Cardinalité d'une association est le nombre de fois minimal et maximal qu'une occurrence d'une des entités associée peut intervenir dans l'association

– Cardinalité minimale

**Elle peut prendre deux valeurs :**

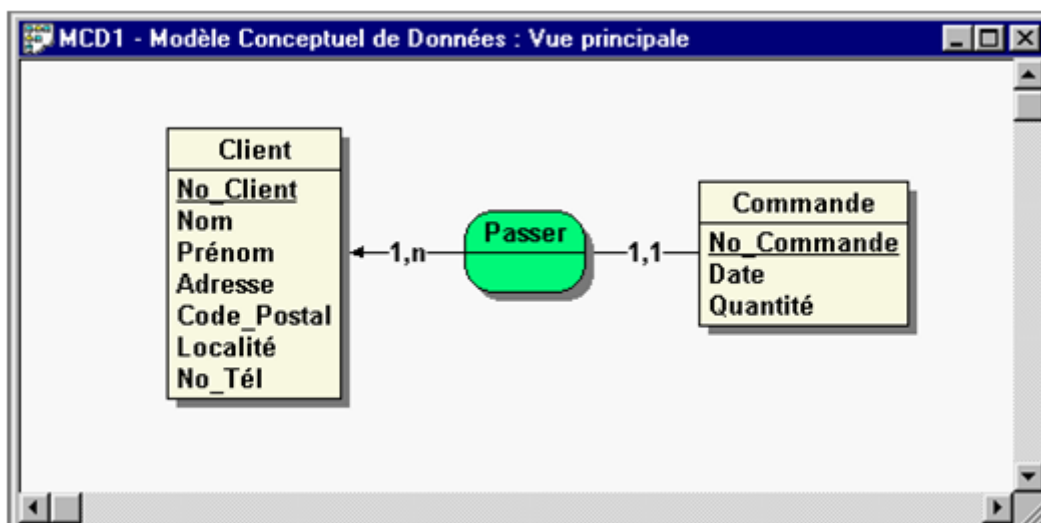
- **0**, signifie que l'entité peut ne pas intervenir dans l'association.
- **1**, signifie au contraire qu'elle intervient obligatoirement une fois.

– Cardinalité maximale

**Elle peut prendre deux valeurs :**

- **1**, signifie que l'entité ne peut intervenir plus d'une seule fois dans l'association-type ;
- **n**, signifie au contraire qu'elle peut intervenir plusieurs fois dans l'association

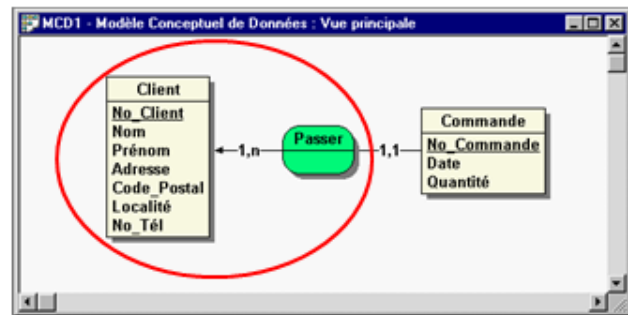
**Que signifie Les cardinalités ?**



Entre l'entité *Client* et la relation *Passer*, nous avons :

Cardinalité **minimale = 1** , ce qui veut dire que chaque client passe au moins une commande

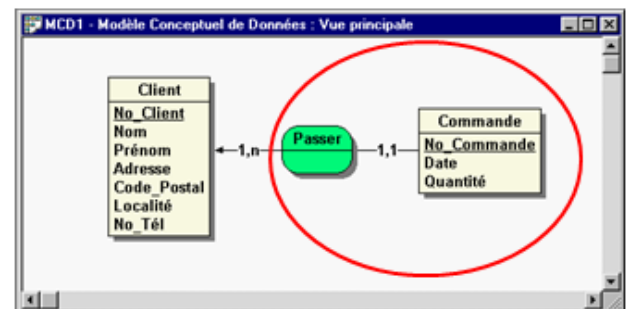
Cardinalité **maximale = n** , ce qui veut dire que chaque client peut passer plusieurs (n) commandes



Entre l'entité *Commande* et la relation *Passer*, nous avons :

Cardinalité **minimale = 1** , donc chaque commande est passée par au moins un client

Cardinalité **maximale = 1** , chaque commande est passée au maximum par un seul client



## 5.1.2 Règles de base pour la construction d'un MCD

### a. Règles sur les individus

Un individu est un objet répondant aux conditions suivantes :

- 1 - être d'intérêt pour l'organisation
- 2 - être distinguable parmi les autres individus de son type, c'est-à-dire qu'il existera un identifiant par individu (numéro, code, référence...)
- 3 - avoir une existence propre, "c'est-à-dire une existence concevable sans hypothèse sur l'existence d'autres éléments du réel perçu".
- 4 - être doté d'un ensemble unique de propriétés. Toutes les informations ont une seule valeur, une seule occurrence pour une occurrence d'individu.

### b. Règles sur les propriétés des individus

- 1 - Pour qu'il n'y ait pas d'ambiguïté possible, une propriété ne peut pas apparaître plusieurs fois dans un MCD. Autrement dit, toutes les propriétés d'un MCD doivent être différentes les unes des autres.
- 2 - Une seule occurrence d'information par individu. La valeur d'une information est unique pour un individu (nommé aussi ensemble unique de propriétés d'un individu)

### c. Règles concernant les associations

- 1- L'identifiant d'une association est implicitement formé par la concaténation des identifiants des entités liées.
- 2- On ne représente pas cet identifiant au niveau du MCD. Deux occurrences d'association ne peuvent pas avoir le même identifiant. Pour une occurrence, l'identifiant ne doit jamais changer de valeur (l'identifiant est une propriété constante).
- 3- Une association peut avoir des propriétés, mais ce n'est pas obligatoire. impliquée dans l'association, au minimum et au maximum.

### Les contraintes d'intégrité

- Sur les propriétés : elles sont de deux types :

- syntaxique : la date qui s'écrit JJ-MM-AA
- sémantique : l'âge d'un employé doit être compris entre 18 et 62 ans.

- Sur les relations :

\* Cardinalités individuelles : d'un individu-type au sein d'une collection d'une relation-type, les nombres minimum et maximum de fois qu'une même occurrence de cet individu-type peut apparaître dans les occurrences de la relation-type.

Pour l'exemple précédent, les couples (0-n) et (1-n) indiquent respectivement :

- qu'un enseignant n'enseigne aucun étudiant ou enseigne plusieurs étudiants ;
- qu'un étudiant est enseigné au moins par un enseignant ou par plusieurs enseignants.

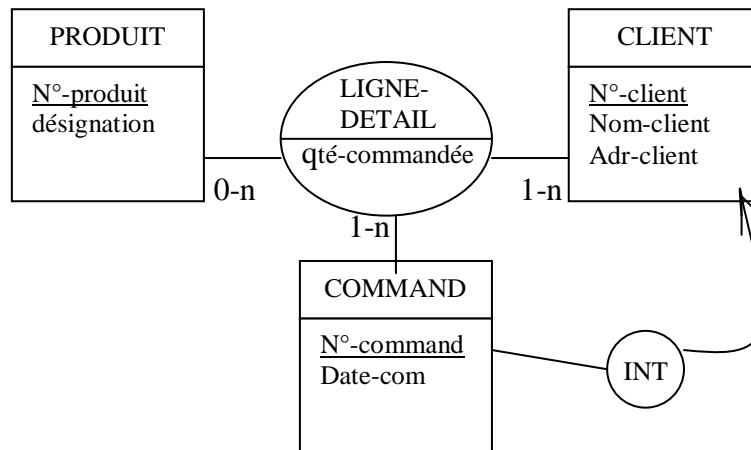
Les couples de valeurs possibles sont : (0-1) ; (1-1) ; (0-n) ; (1-n)

\* Cardinalités fonctionnelles ou contraintes d'intégrité fonctionnelles (CIF): dans une relation-type, la CIF indique l'existence d'une dépendance fonctionnelle entre un sous-ensemble de la collection et un individu-type de la collection.

La CIF « INT » définie sur la relation LIGNE-DETAIL dans l'exemple ci-dessous signifie qu'une commande donnée n'est émise que par un client donné et un seul.

Remarque : les cardinalités (0-1) ou (1-1) dans une relation binaire imposent directement une CIF sur cette relation.

Exemple :



□ Décomposition d'une relation-type

La décomposition est un processus réversible qui vise à simplifier le modèle conceptuel *sans en modifier le sens*.

Elle permet de diminuer la dimension d'une relation-type par l'utilisation des contraintes d'intégrité définies sur les relations-type.

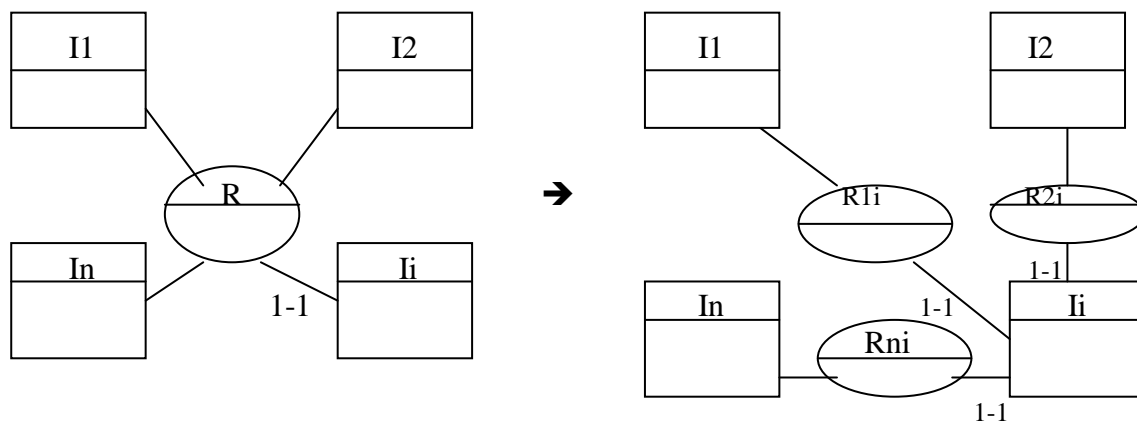
On distingue deux sortes de décomposition : *endogène et exogène*.

• La décomposition endogène :

➔ en utilisant les cardinalités individuelles :

soit une relation R de dimension n et de collection (I1, I2 , ..., In) et un individu-type Ii de cette collection telle que sa cardinalité individuelle dans R soit de (1-1).

La décomposition endogène en utilisant la cardinalité individuelle (de Ii) permettra de transformer la relation R de dimension n en (n – 1) relations binaires.

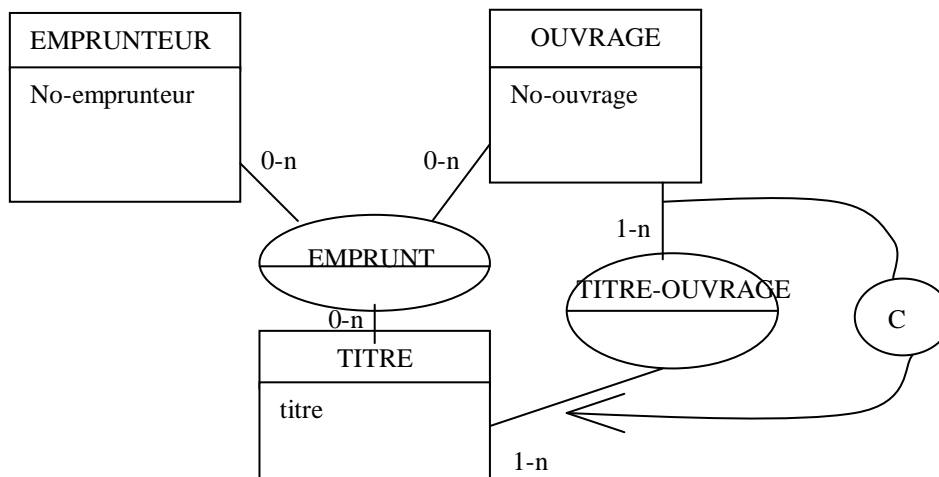


➔ en utilisant les CIFs :



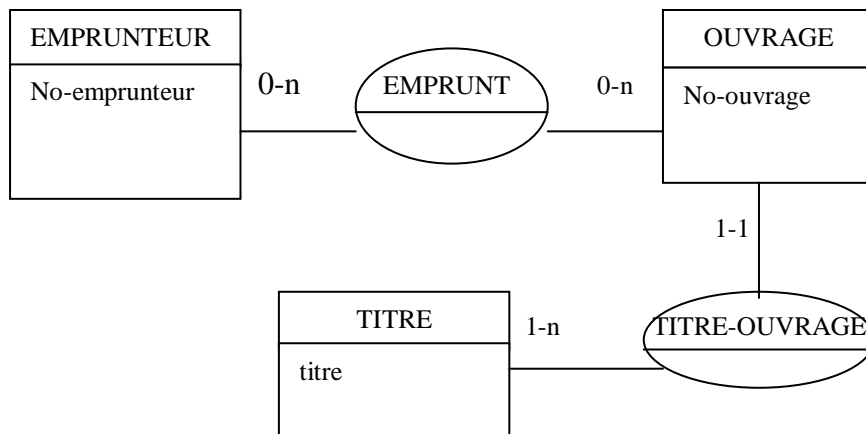


Exemple :



la CIF C exprime qu'un ouvrage ne possède qu'un et un seul titre.

Le modèle décomposé sera :



## 5.2 La construction du modèle conceptuel de données (MCD)

Soit l'exemple suivant :

No-bon .....	Date .. /.. /..
Nom client .....	
Adresse client .....	
Nom représentant .....	

Référence	désignation	quantité	prix unitaire	montant
.....	.....	.....	.....	.....
.....	.....	.....	.....	.....
total .....				

### a) recueil des informations

- rassembler les exemplaires de tous les documents ;
- expliciter clairement les règles de gestion.

### Exemple :

R.G.1 : un client peut passer une, plusieurs ou aucune commande ;

R.G.2 : une commande peut concerner un ou plusieurs produits ;

R.G.3 : une commande est passée à un représentant qui n'est pas toujours le même pour un client donné.

On établit la liste des propriétés à partir des documents et des fichiers.

### b) constitution du dictionnaire de données

Le dictionnaire de données est représenté dans le tableau suivant :

NOM	SIGNIFICATION	TYPE	LONGUEUR	NATURE		REGLE DE CALCUL OU INTEGRITE
				E/CO/CA	M/SIG/SIT	
No-bon	N° de bon de cmde	N	4	E	M	
Date	Date cmde	N	6	E	M	JJMMAA
Cocli	Code client	?	?	E	SIG	A créer
Nomcli	Nom client	A	30	E	SIG	
Adresse	Adresse client	AN	60	CO	SIG	Rue+ville
montant	Montant ligne	N	8	CA	M	P.U x QTE
.....	.....	.....	.....	.....	.....	.....

E : élémentaire ; CO : concaténé ; CA : calculé

M : mouvement ; SIG : signalétique ; SIT : situation.

### c) épuration du dictionnaire de données

des difficultés peuvent apparaître entre les signifiés et les signifiants .

- ➔ Signifié : objet abstrait ou concret que l'on veut qualifier ;
- ➔ Signifiant : mot employé pour représenter un signifié.

Il faut alors éviter les cas suivants :

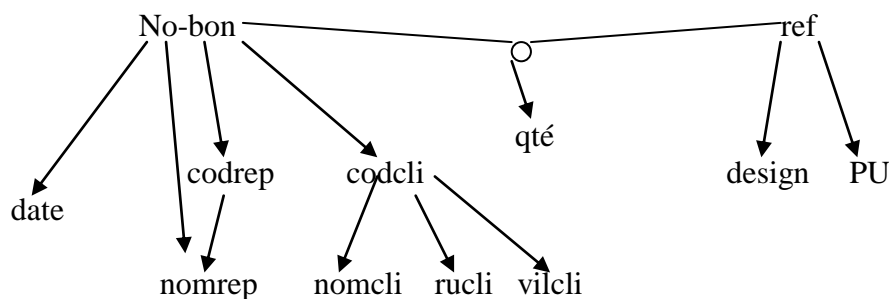
- ➔ synonymes : deux signifiants pour un même signifié,  
exemple : n°-client et code-client.
- ➔ polysèmes : un signifiant pour deux signifiés,  
exemple : nom pour nom du client et nom du fournisseur.

### d) graphe de dépendances fonctionnelles

Extraire du DD la liste des propriétés non calculées et non concaténées.

Dans l'exemple précédent, montant et total sont calculés et remplacer adresse par rue et ville.

Etablir la liste de dépendances fonctionnelles (DFs) dont le domaine de départ ne contient qu'une seule propriété (non concaténée) à partir de l'examen des documents et des identifiants évidents. Cette liste de DFs peut être représentée par le graphe suivant :

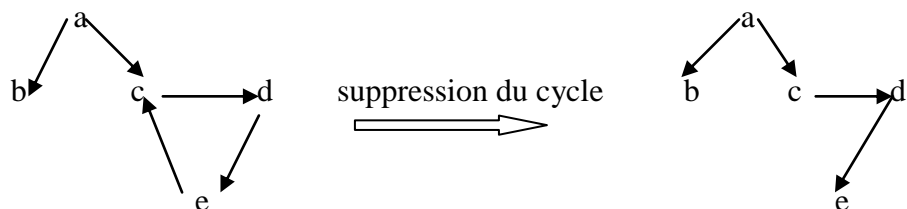


pour les propriétés isolées, on cherche des DFs qui conduisent à ces propriétés à partir des propriétés concaténées. Si on n'en trouve pas, la propriété reste isolée, sinon la propriété devient non-isolée.

Exemple : (no-bon + ref) → qté    qté est une propriété non-isolée.

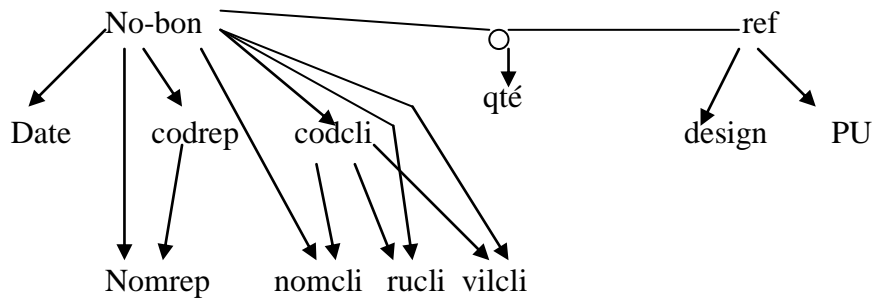
Si le graphe comporte *des cycles* alors supprimer une DF pour éviter cette anomalie.

Exemple :



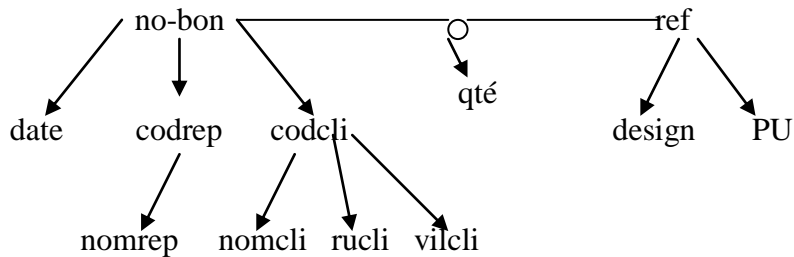
Etablir ensuite la liste de toutes les DFs qui découlent du graphe précédent.

On établit alors la fermeture des DFs :



On vérifie qu'il n'y a pas de cycle ; on élimine les transitivités ;

⇒ on obtient alors la structure d'accès théorique (SAT) ou la couverture minimale :



#### e) établissement du MCD

Les arcs terminaux obtenus à partir des propriétés élémentaires définissent les entités.

Les origines de ces arcs sont les identifiants.

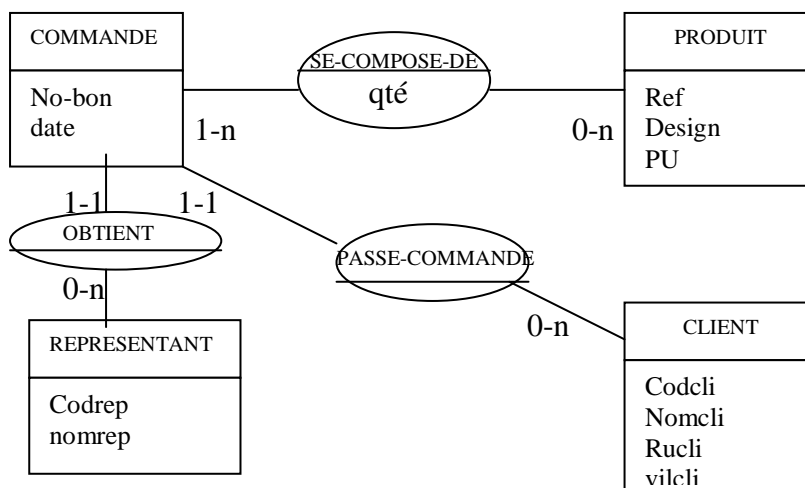
Les propriétés correspondantes constituent les individus-type.

Les arcs restants mettent en évidence les relations-type.

Les propriétés non-isolées restantes sont affectées à des relations-type.

Les propriétés isolées doivent constituer des entités isolées.

Les règles de gestion doivent permettre de trouver les cardinalités et éventuellement d'autres relations-type non fonctionnelles.



Il faut s'assurer que les règles de vérification, de normalisation et de décomposition sont respectées. Le dernier graphe en est le cas.

### 5.3 Exercices

#### Exercice N°2

Pour la structuration du système d'informations d'un groupe de sociétés, on a recueilli les informations sur les sujets suivants:

CLIENT: N°client, nom

SOCIETE: Nom-société, activité, capital

DEPOT: Nom-dépôt, adresse, superficie

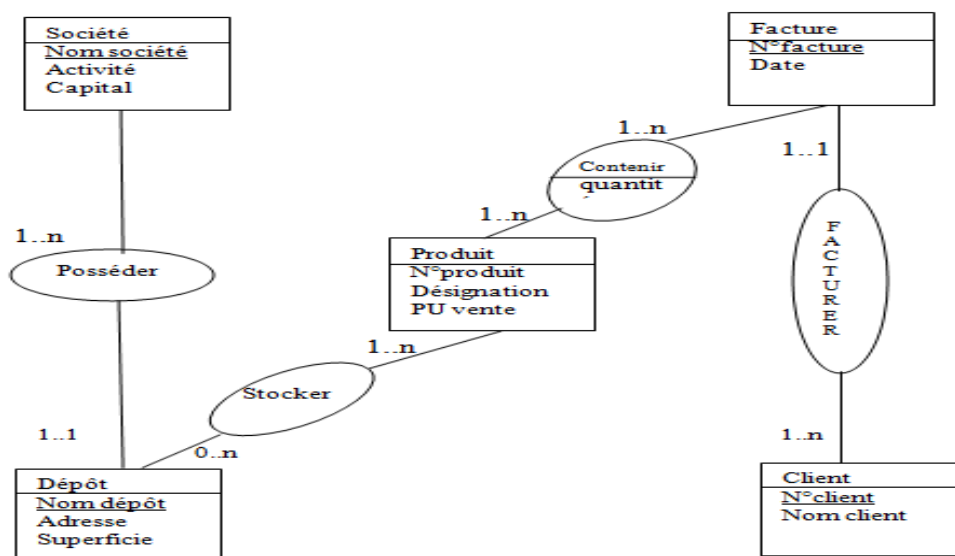
FACTURE dont l'image est la suivante:

N°facture			date
N°client			
N°produit	Désignation	PU vente	Quantité
-----	-----	-----	-----
-----	-----	-----	-----
-----	-----	-----	-----

Représenter le modèle conceptuel (MCD) de données avec les propriétés et les cardinalités, sachant que :

- une société possède plusieurs dépôts,
- un dépôt peut contenir plusieurs ou aucun produit,
- un produit peut être entreposé dans plusieurs dépôts.

#### Corrigé exercice



## Exercice N°2

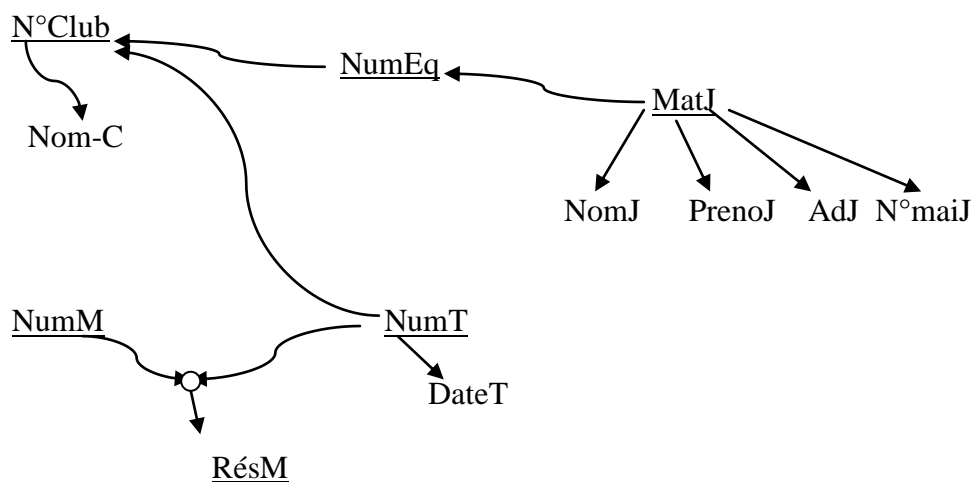
Une fédération sportive désire informatiser l'organisation de ces tournois. Les club de la fédération sont dotés d'un numéro (unique) et d'un nom. Chaque joueur d'un club appartient à une seule équipe (junior, sénior...) de celle-ci. Un joueur est décrit par un numéro matricule attribué par la fédération, son nom, prénom et adresse, son numéro de maillot. Un tournoi est décrit par un numéro unique et sa date. Chaque tournoi est organisé par un club de la fédération. Au cours d'un tournoi les différentes équipes s'affrontent dans des matchs. Un match est décrit par un numéro unique du tournoi, et son résultat (victoire de l'équipe A, victoire de l'équipe B ou match nul).

-a)Etablir les MCD associé au problème (DD épuré, Graphe de dépendances fonctionnelles, MCD).

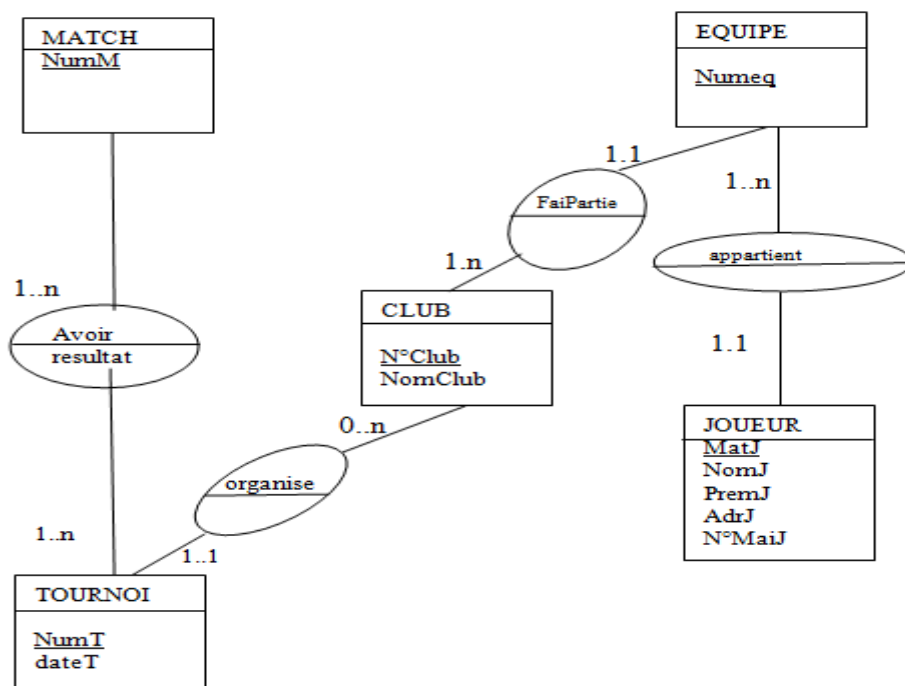
Etablissement du DD

Nom	Désignation	Nature
N°Club	Numéro club	E
Nom-C	Nom du club	E
MatJ	Matricule du joueur	E
NomJ	Nom du joueur	E
PrenoJ	Prénom du joueur	E
AdJ	Adresse du joueur	E
N°maiJ	Numéro maillot du joueur	E
NumT	Numero de tournoi	E
DateT	Date de tournoi	E
NumEq	Numero d'équipe	E
NumM	Numéro Match	E
RésM	Résultat Match	E

Graphe de dependances Fonctionnelles



## Etablissement du MCD





# Bibliographie

1. Le développement de SI- Une méthode intégrée à la transformation des processus, Suzanne Rivard (Auteur) - Presses de l'Université du Québec. 07/2013
2. V. Bertalanfy, Théorie générale des systèmes. Dunod. X. Castellani, Méthode générale d'analyse d'une application informatique.
3. Tardieu et al. , « la méthode merise : principes et outils », éd. d'organisation, 1983.
4. Tardieu et al. , « la méthode merise : démarche et pratique » éd. d'organisation, 1985.
5. Tabourier, « de l'autre côté de Merise », éd. d'organisation, 1986.
6. J. P. Mathéron, « Comprendre Merise », 1990
7. Solutions d'entreprise, Date de parution : 03/01/2013 (2<sup>e</sup> édition)
8. Cours : Introduction aux SI., Pr. Zarour Nacer eddine, <http://www.univ-constantine2.dz/facntic>
9. La méthode Merise : H. Tardieu, A. Rochfeld, R. Coletti aux Ed. d'organisation AMC\*Designor : Mise en œuvre de merise – Gilles GUEJ aux Editions Eyrolles [www.commentcamarche.net](http://www.commentcamarche.net): La méthode Merise.
10. Hubert Tardieu, Arnold Rochfeld, René Colletti. La méthode Merise, Editions
11. ] Bertrand Bisson. Modèles de données, Etudes conceptuelle et relationnelle, Economica, 2005.
12. ] Joseph Gabay. Merise et UML pour la modélisation des systèmes d'information. Dunod, 2004
13. [1] Cauvet C., rosenthal-Sabroux C., Ingénierie des systèmes d'information, Informatique et système d'information, Hermès Science Publications, 2001.
14. [2] Karsenti G., La fin du paradoxe de l'informatique, Éditions d'Organisation, Paris, 1999.
15. [3] Morley C, Hugues J, Leblanc B, UML pour l'analyse d'un système d'information, Dunod Informatiques, 2000.